UNIVERSITY OF PANNONIA

DOCTORAL THESIS

Matrix-based project planning method for multi-level project environments

Author:
Gergely Lajos NOVÁK

Supervisor:
Prof. Dr. habil. Zsolt Tibor KOSZTYÁN

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

Doctoral School in Management Sciences and Business Administration
Department of Quantitative Methods

December 15, 2022

Declaration of Authorship

I, Gergely Lajos NOVÁK, declare that this thesis titled, "Matrix-based project planning method for multi-level project environments" and the work presented in it are my own. I confirm that:

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

Signed:
....
Date:
....

Matrix-based project planning method for multi-level project environments

Thesis for obtaining a PhD degree in the Doctoral School in Management Sciences and Business Administration of the University of Pannonia

in the branch of Social Sciences on the subject of Management and Business Studies

Author: Gergely Lajos NOVÁK
Supervisor: Prof. Dr. habil. Zsolt Tibor KOSZTYÁN

Propose acceptance (yes/no) ....
Reviewer

As reviewer, I propose acceptance of the thesis:

Name of reviewer: .... yes / no

....
(Reviewer)

Name of reviewer: .... yes / no

....
(Reviewer)

The PhD candidate has achieved ....% at the public discussion.

Veszprém, ...
(Chairman of the Committee)

The grade of the PhD Diploma ....

Veszprém, ...
(Chairman of UDHC)

UNIVERSITY OF PANNONIA

Abstract

Doctoral School in Management Sciences and Business Administration
Department of Quantitative Methods

Doctor of Philosophy

Matrix-based project planning method for multi-level project environments

by Gergely Lajos NOVÁK

Flexible methods, such as agile, hybrid and extreme project management, are increasingly replacing traditional project management. The popularity of flexible approaches is growing significantly in non-software areas as well. Only a few methods, however, can address projects of a flexible nature, and no project databases or topological, time-, or resource-related indicators are available in the literature.

To represent the different types and attributes of project plans, such as individual and multiple projects, programs, single- and multiple execution modes and demands, this study proposes a unified matrix-based model (UMP). Using the model, a compound matrix-based project database (CMPD) is constructed that combines 15 existing heterogeneous project and multiproject libraries from the literature, including 32 datasets, both artificial and real-life data, with the ability to support flexibility.

A flexible structure generator (FSG) is also proposed to generate supplementary tasks and flexible dependencies using a specified parameter and to analyze the effects on project databases. Topology and demand-related indicators were adapted, and new indicators were proposed to address flexibility. A case study for a software development multiproject plan from the automotive industry was used to validate the model and the corresponding parameters.

Using correlation graphs, the relations between structure- and demand-related indicators were analyzed. Comparing the simulated and real-life databases revealed the contrast in indicator value ranges, that can be improved with flexibility. Indicators show that flexibility decreases complexity and serial execution with lower project duration and higher resource demands. Decisions in the planning phase can be significantly improved by considering minimal and maximal (demand) structures generated by the FSG. Using metaheuristic optimization, the different flexible structures were solved to near-optimality. The constructed open database allows users to test both traditional and flexible project scheduling algorithms.

Keywords: Multilevel project management; Project database; Flexibility; Topology, Time- and resource related indicators

UNIVERSITY OF PANNONIA

Zusammenfassung

Doctoral School in Management Sciences and Business Administration
Department of Quantitative Methods

Doctor of Philosophy

Matrix-based project planning method for multi-level project environments

von Gergely Lajos NOVÁK

Traditionelles Projektmanagement wird zunehmend durch flexible Methoden wie agiles, hybrides und extremes Projektmanagement ersetzt. Auch in Nicht-Software-Bereichen nimmt die Popularität flexibler Ansätze deutlich zu. Allerdings können nur wenige Methoden Projekte mit flexiblem Charakter adressieren, und in der Literatur sind keine Projektdatenbanken oder topologische, zeit- oder ressourcenbezogene Indikatoren verfügbar.

Um die verschiedenen Typen und Attribute von Projektplänen, wie Einzel- und Mehrfachprojekte, Programme, Einzel- und Mehrfachausführungsmodi und Anforderungen darzustellen, schlägt diese Studie ein einheitliches Matrix-basiertes Modell (UMP) vor. Unter Verwendung des Modells wird eine Verbundmatrix-basierte Projektdatenbank (CMPD) konstruiert, die 15 kombiniert vorhandene heterogene Projekt- und Multiprojektbibliotheken aus der Literatur, einschließlich 32 Datensätze, sowohl künstliche als auch reale Daten, mit der Fähigkeit, Flexibilität zu unterstützen.

Außerdem wird ein flexibler Strukturgenerator (FSG) vorgeschlagen, um ergänzende Aufgaben und flexible Abhängigkeiten anhand eines vorgegebenen Parameters zu generieren und die Auswirkungen auf Projektdatenbanken zu analysieren. Topologie- und nachfragebezogene Indikatoren wurden angepasst und neue Indikatoren wurden vorgeschlagen, um die Flexibilität zu adressieren. Um das Modell zu validieren und den entsprechenden Parametern wurde ein Softwareentwicklungs-Multiprojektplan aus der Automobilindustrie in einer Fallstudie verwendet.

Anhand von Korrelationsgraphen wurden die Beziehungen zwischen struktur- und nachfragebezogenen Indikatoren analysiert. Der Vergleich der simulierten und realen Datenbanken zeigte den Kontrast in den Wertebereichen des Indikators, der durch Flexibilität verbessert werden kann. Indikatoren zeigen, dass Flexibilität die Komplexität und Serienausführung verringert, mit kürzerer Projektdauer und höherem Ressourcenbedarf. Entscheidungen in der Planungsphase können durch die Berücksichtigung minimaler und maximaler (Bedarfs-)Strukturen, die von der FSG generiert werden, erheblich verbessert werden. Mittels metaheuristischer Optimierung wurden die verschiedenen flexiblen Strukturen nahezu optimal gelöst. Mit der konstruierten offenen Datenbank können Benutzer sowohl herkömmliche als auch flexible Projektplanungsalgorithmen testen.

Stichworte: Mehrstufig Projektmanagement; Projektdatenbank; Flexibilität; Topologie, Zeit- und ressourcenbezogene Indikatoren

Acknowledgements

I am grateful beyond words to my supervisor Prof. Dr. Kosztyán Zsolt Tibor, for his inspiring hard work and continuous guidance over all these years. I would also like to thank my co-authors and members of the research group, Csaba Hegedűs, István Szalkai, Marcell Tamás Kurbucz, and Mária Kisgyörgy-Pál for their support.

I am indebted to my family; especially my parents and fiancee for their endless patience and unconditional love. Special thanks to my friend Róbert Jakab for being a mentor and Kristóf Steierlein for motivating me. Shout out to my colleagues, friends, and fellow students.

Finally, I appreciate all the valuable feedbacks and discussions from peer reviewers, proofreaders and experts.

Professional acknowledgements

PREPARED WITH THE PROFESSIONAL SUPPORT OF THE DOCTORAL STUDENT SCHOLARSHIP PROGRAM OF THE CO-OPERATIVE DOCTORAL PROGRAM OF THE MINISTRY OF INNOVATION AND TECHNOLOGY FINANCED FROM THE NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION FUND.

NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION OFFICE
HUNGARY

PROGRAM FINANCED FROM THE NRDI FUND

Contents

List of Figures

List of Tables

List of Abbreviations

ANOVA	**an**alysis **o**f **va**riance
GA	**g**enetic **a**lgorithm
MPE	**m**ulti-**p**roject **e**nvironment
AoN	**a**ctivity **o**n **n**ode
AoA	**a**ctivity **o**n **a**rrow
CPM	**c**ritical **p**ath **m**ethod
WBS	**w**ork **b**reakdown **s**tructure
PM	**p**roject **m**anager
MCS	**M**onte-**C**arlo **s**imulation
SDP	**s**oftware **d**evelopment **p**roject
IT	**i**nformation **t**echnology
ICT	**i**nformation and **c**ommunications **t**echnology
PSP	**p**roject **s**cheduling **p**roblem
RCPSP	**r**esource-**c**onstrained **p**roject **s**cheduling **p**roblem
RCPSP-AS	**r**esource-**c**onstrained **p**roject **s**cheduling **p**roblem with **a**lternative **s**ubgraphs
RCPSP-AC	**r**esource-**c**onstrained **p**roject **s**cheduling **p**roblem with **a**lternative activity **c**hains
RCMPSP	**r**esource **c**onstrained **m**ulti**p**roject **s**cheduling **p**roblem
MMRCPSP	**m**ulti**m**ode **r**esource **c**onstrained **p**roject **s**cheduling **p**roblem
MMRCMPSP	**m**ulti**m**ode **r**esource **c**onstrained **m**ulti**p**roject **s**cheduling **p**roblem
PSPLIB	**p**roject **s**cheduling **p**roblem **lib**rary
MNPG	**m**ulti-**n**etwork **p**roblem **g**enerator
EF	**e**arliest **f**inish
ES	**e**arliest **s**tart
LS	**l**atest **s**tart
LF	**l**atest **f**inish
EST	**e**arliest **s**tart **t**ime
EFT	**e**arliest **f**inish **t**ime
LST	**l**atest **s**tart **t**ime
LFT	**l**atest **f**inish **t**ime
SST	**s**cheduled **s**tart **t**ime
F-S	**f**inish-to-**s**tart relation
GPR	**g**eneralized **p**recedence **r**elation
TPT	**t**otal **p**roject **t**ime
DSM	**d**esign **s**tructure **m**atrix
DMM	**d**omain **m**apping **m**atrix

PDM	**p**roject **d**omain **m**atrix
PEM	**p**roject **e**xpert **m**atrix
LD	**l**ogic **d**omain
TD	**t**ime **d**omain
CD	**c**ost **d**omain
RD	**r**enewable resource **d**omain
ND	**n**onrenewable resource **d**omain
QD	**q**uality **d**omain
DSDM	**d**ynamic **s**ystems **d**evelopment **m**ethod
CMPD	**c**ompound **m**atrix-based **p**roject **d**atabase
UMP	**u**nified **m**atrix-based **p**roject-planning **m**odel
FSG	**f**lexible **s**tructure **g**enerator
M^4	**m**atrix-based **m**ulti-level **m**ulti-**m**ode
NPD	**n**ew **p**roduct **d**evelopment
AHP	**a**nalytical **h**ierarchy **p**rocess
R&D	**r**esearch and **d**evelopment
TPM	**t**raditional **p**roject **m**anagement
APM	**a**gile **p**roject **m**anagement
HPM	**h**ybrid **p**roject **m**anagement
xPM	e**x**treme **p**roject **m**anagement
MPx	emertxe **p**roject **m**anagement (←)
WIP	**w**ork **i**n **p**rogress
RQ	**r**esearch **q**uestion
RA	**r**esearch **a**ssumption
RT	**r**esearch **t**hesis

List of Symbols

Latin symbols

$a_i(T)$	scheduled execution time interval of task a_i
a_i	task i
D	maximal number of short arcs
EF_i, LF_i	early and late finish for task i
ES_i, LS_i	early and late start to task i
$f\%$	rate of flexible dependencies
fp	flexibility parameter, the ratio of flexible dependencies and prioritized tasks to all tasks and dependencies
FS_i	free slack of task i
k	number of task completion modes
$L(i_1, i_2)$	length of an arc between tasks i_1 and i_2, i.e., the difference between their progressive level numbers
$\overrightarrow{L}$	longest (critical) path
$l_{ij} = [\mathbf{LD}]_{ij}$	element of the logic domain, task occurrence if $i = j$, and arc that represent the precedence relation between tasks i and $j \neq i$ (in this case, $l_{ij} = 1$ means task i precedes task j)
n	number of tasks
n'_L	number of arcs with length L
m	maximal number of progressive levels
$\overrightarrow{P}$	task path (sequence)
P_i	set of immediate predecessors of task i
r_{ij}	demand of task i for renewable resource type j
$r_{ij}(\tau)$	demand of task i for renewable resource type j at time τ
$s\%$	rate of supplementary tasks
S_i	set of immediate successors of task i
t_i	duration of task a_i
TS_i	total slack of task i
w_i	width of progressive level i, $i = 1, ..., m$

Greek symbols

α_j	availability of renewable resource type j
α_w	total absolute deviation from the average width
η	number of types of nonrenewable resources
ρ	number of types of renewable resources

Calligraphic symbols

$\mathcal{A}$	set of arcs (dependencies)
$\lvert\mathcal{A}\rvert$	number of dependencies in a project structure
$\mathcal{S}$	project structure, set of (to-be-) realized tasks
$\overrightarrow{\mathcal{S}}$	project schedule of project structure $\mathcal{S}$

Chapter 1

Introduction

1.1 Motivation of the thesis

Our society is substantially projectified (Godenhjelm et al., 2015). Around 40% of the global economy is project-based and project management is a fundamental process for producing products and services (Turner, Huemann, et al., 2010). Projects of all types can contribute almost 20% of a country's GDP (Denizer et al., 2013; World Bank, 2012), and have become the standard way of doing business. According to Lova et al. (2000), 84% of companies run projects simultaneously in a multiproject context, a similar number reported already in the nineties (Payne, 1995), while their challenges are still prevailing. Several studies have shown that to increase the success of these projects (Johnson, 2020), traditional project management approaches are increasingly being replaced by flexible approaches (Ciric, Lalic, et al., 2019; Hidalgo, 2019a; Özkan and Mishra, 2019; Wysocki, 2019) not only in the IT field (see, e.g., in Stare, 2014) but also previously unconsidered fields, such as construction (Yasaman et al., 2022) and maintenance projects (Kosztyán, Pribojszki-Németh, et al., 2019).

These approaches require flexible project plans, allowing, for example, the possibility of either or both project restructuring and task reprioritization according to the customer's requirements; however, most project planning methods assume a fixed (Franco-Duran and Garza, 2019) logic plan or a limited number of scheduling alternatives (Čapek et al., 2012; Creemers et al., 2015; Hauder et al., 2020; Kellenbrink and Helber, 2015; Servranckx and Vanhoucke, 2019b; Tao and Dong, 2018). There are already a few matrix-based methods available for scheduling structurally flexible projects and multilevel projects (Kosztyán, 2015; Kosztyán and Szalkai, 2020) where certain task realizations and dependency occurrences are considered as variables during the planning phase. However, there is neither a project database that supports the design, planning, and scheduling of flexible (multi)projects nor a set of complexity and time- or resource-related indicators that are capable of characterizing flexible project plans available. It is essential to provide both scholars and practitioners with such a database and set of indicators to allow them to examine flexible projects.

Therefore, the aim of the dissertation was threefold. First, to specify a matrix-based method, which can handle single and multi-level projects, multiple execution modes, and flexible projects besides traditional ones. In addition, to collect existing heterogeneous project databases, including simulated (artificial) and real-life projects. Finally, to examine the effects of flexibility not only on the project structure but on the project demands as well.

1.2 Research questions

Considering the issues and their relevance above, the current study seeks to answer the following research questions:

RQ1: How to create a unified model that can represent the heterogeneous project and multiproject databases available in the literature?

RQ2: How the flexibility of single- and multiproject plans can be modeled?

RQ3: What characterizes the topology (structure) and the different demands of the flexible project and multiproject plans?

RQ4: How is it possible to find feasible (sub)optimal solution for the single- and multiproject plans considering flexibility?

1.3 Structure of the thesis

The dissertation is organized as follows. After a brief introduction to project management in Chapter 1, the literature is reviewed in Chapter 2 with the related works and databases. In Chapter 3, first, the applied project databases and the considered complexity, time-related, and resource-related indicators are introduced. Then, the flexibility-dependent indicators are specified. In Chapter 4, the applied project databases are compared, and the flexibility effects are examined. Chapter 5 discusses, while Chapter 6 provides validation of the results. A summary is given in Chapter 7, then Chapter 8 considers the limitations of this research and gives directions for future work.

Chapter 2

Literature review

2.1 Projects and their management

2.1.1 Definitions

There are many attempts in the literature to define projects. Most of the authors describe projects as temporary endeavors with specific objectives, uniqueness, complexity, and novelty; constrained by scope, quality specifications, budget, time, and limited resources (see e.g., Gareis, 2000; PMI, 2021; Turner, Huemann, et al., 2010; Wysocki, 2019).

The definition of project has evolved from unique tasks (Olsen, 1971) to building blocks of strategy (Cleland, 1994) and temporary organizations (Lundin and Söderholm, 1995).

Turner, Huemann, et al. (2010) defines the project as *"a temporary organization to which resources are assigned to do work to deliver beneficial change"*.

The temporary organization is *"a unique endeavor in which human, financial and material resources are organized in a novel way to undertake a unique scope of work, of given specification, within constraints of cost and time, so as to achieve beneficial change by quantitative and qualitative objectives"* (Turner, Huemann, et al., 2010)

Di Muro et al. (2021) defines projects as *"functional networks aimed at delivering solutions or business benefits"* (Sydow et al., 2004; Thiry and Deguire, 2007).

2.1.2 Project lifecycle and typology

The lifecycle of a project typically consists of several consecutive phases. The number and name of these phases often vary, but they are still relatively standard in the literature. According to Schwindt, Zimmermann, et al. (2015), there are five such phases, that also involve managerial tasks, as we will see later.

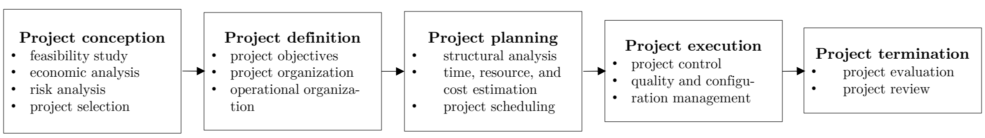


FIGURE 1: Project lifecycle (Schwindt, Zimmermann, et al., 2015)

Starting with *project conception*, the basic idea of the project is analyzed from different perspectives for feasibility. *Project definition* specifies the project objectives and organizational structure. In the *project planning* phase, the activities relations are identified, and their duration, cost, and required resources are estimated for a schedule. During *project execution*, the progress of execution is monitored and, if needed, adjusted for the previously defined plan. Finally, the *project termination* phase evaluates the project's completion and documents it (Schwindt, Zimmermann, et al., 2015). Each of the phases is shown in Figure 1 in detail. (see e.g., Archibald, 2003; Cleland, 2007; Corsten and Corsten, 2000; Görög, 2003; Leybourne, 2007; PMI, 2021).

Projects can be categorized based on many aspects, e.g., type of industry, content, size, location, complexity, technology, novelty, environment, customer, etc. One example is given by Turner (2009), where the clarity of goals and methods is used to categorize projects, shown in Figure 2. A similar matrix is proposed by Szűcs (2000). Litke (2007) takes a different approach and defines four groups based on complexity and novelty.

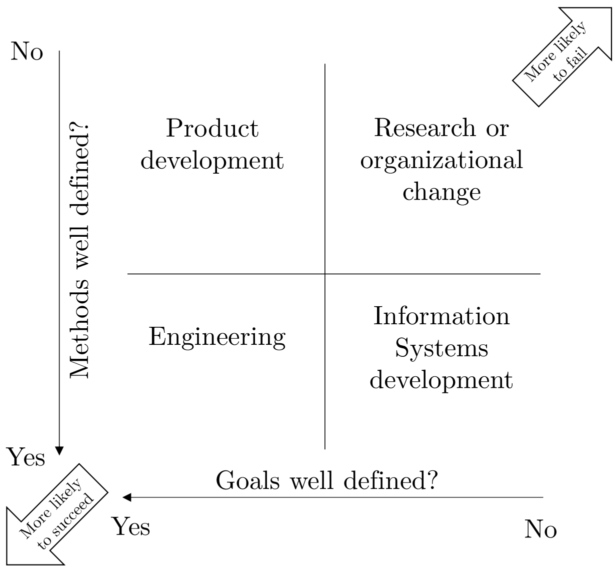


FIGURE 2: Project types based on goals and methods (based on Turner (2009))

Shenhar, Dvir, et al. (2002) gave an overview of the existing classification frameworks for projects. Shenhar and Dvir (2007) proposed their diamond model to distinguish between projects and select the right management approach based on the following dimensions: novelty, technology, complexity, and pace, as shown in Appendix A. It is possible to analyze the expected benefits and risks using the framework. The novelty dimension represents the uncertainty of either the project's goal or the market. Different technologies carry different risks and possible benefits. Complexity is related to the organization and the level of formality needed to manage it. Pace affects the planning and urgency of projects. The study of Hansen et al. (2022) found that extending the dimension of pace with impact would further improve the model.

2.1.3 Management of projects

Project management is considered the primary capability of a firm to respond to change and thereby gain competitive leverage (Pellegrinelli et al., 2015; Vuorinen and Martinsuo, 2018).

Westland (2009) defines project management as the *"skills, tools and management processes required to undertake a project successfully."*.

Turner (1993, 2009) explains the three dimensions of project management: the *project* (scope, project organization, stakeholders, quality, cost, time, risk), the *management processes*, followed at each phase of the project lifecycle (plan, organize, implement, control), and the *levels*[1] (integrative, strategic, tactical).

PMI (2021) describes five management process groups employed in a project to meet the objectives. These are: *initiating*, *planning*, *executing*, *monitoring and controlling*, and *closing*.

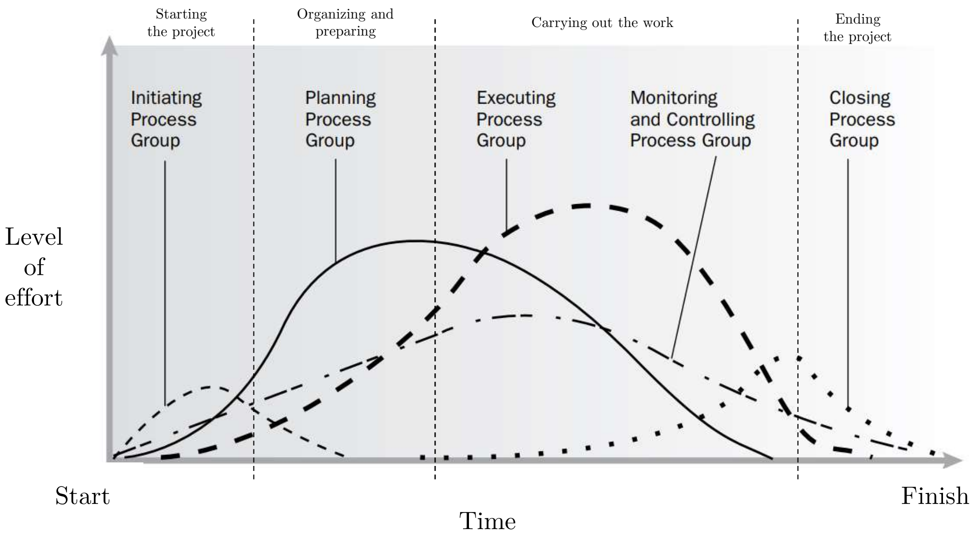


FIGURE 3: Project management process groups and efforts
Based on PMI (2021)

It is important to note that process groups are not project phases. Process groups interact with each other, usually overlap, can provide inputs and receive outputs from other processes, and can be started and iterated when needed. Figure 3 illustrates the process groups and the typical efforts over time.

In the definition of project management given by Westland (2009), one can notice a vaguely mentioned success criteria. The iron triangle, also referred to as the triple constraint or project triangle, is a fundamental and widely accepted concept (Pollack et al., 2018), an essential criterion to measure project success, i.e., if the project is

[1]Turner changed the dimension "levels" later to micro- and macro-levels, but the previous classification is intentionally used here.

delivered in time, within budget, and to the agreed level of quality, performance or scope.

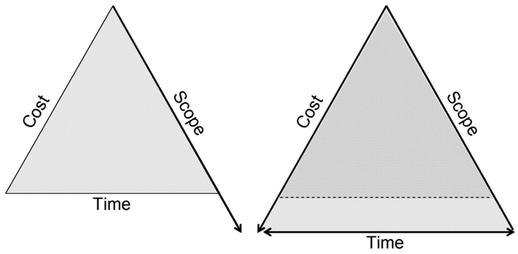


FIGURE 4: Project triangle and a possible scenario
(Van Wyngaard et al., 2012)

The iron triangle has become the standard for assessing project performance. It is typically depicted as a triangle with criteria on the vertices or sides. Change in one criterion, for example, in response to customer demands or resource limitations, can put pressure on other criteria, as they are interconnected. A break in a constraint likely leads to negative consequences (trade-offs) on one or both of the additional two constraint(s) (Van Wyngaard et al., 2012). Figure 4 shows the dynamics when time and cost are affected by the change in scope. The iron triangle is criticized mostly for not having additional criteria for soft aspects, like customer relationship and satisfaction (Pinto, 2013; Pollack et al., 2018; Williams et al., 2015).

Project success is often confused with the success of project management (Papke-Shields et al., 2010), which needs a distinction. Project management success is measured through the project's process and is evaluated at the end (Papke-Shields et al., 2010; Pollack et al., 2018). Project management's success is a subset of project success, a wider concept with many aspects (Radujković and Sjekavica, 2017).

2.1.4 Project management approaches

An important aspect in handling projects is to select the appropriate management approach (Charvat, 2003), as no single best method exists (Špundak, 2014). Project management approaches can also be categorized. A possible grouping is based on goals and solutions (Wysocki, 2019), as shown in Table 1.

The first category has a clear goal and solution and is called traditional project management (TPM). Good examples are construction projects, where requirements are stable and thus, no significant changes are expected.

In contrast, agile project management (APM) has clear goals, but it is unclear how to achieve them. Most software development projects fall into this category. Extreme project management (xPM) has vague goals and solutions, which characterizes most research and development (R&D) or new product development projects. The fourth category, emertxe (MPx), has no clear goal, but there is already a solution. An example is when the technology exists before it is applied.

TABLE 1: Project management categories
Based on Wysocki (2019)

		Solution	
		Clear	Not clear
Goal	Not clear	Emertxe PM (MPx)	Extreme PM (xPM)
	Clear	Traditional PM (TPM)	Agile PM (APM)

2.1.5 Lifecycle of project management

The lifecycle of project management approaches can be differentiated. The traditional approach has a linear or incremental lifecycle where requirements are clearly specified, and changes are not expected during the project.

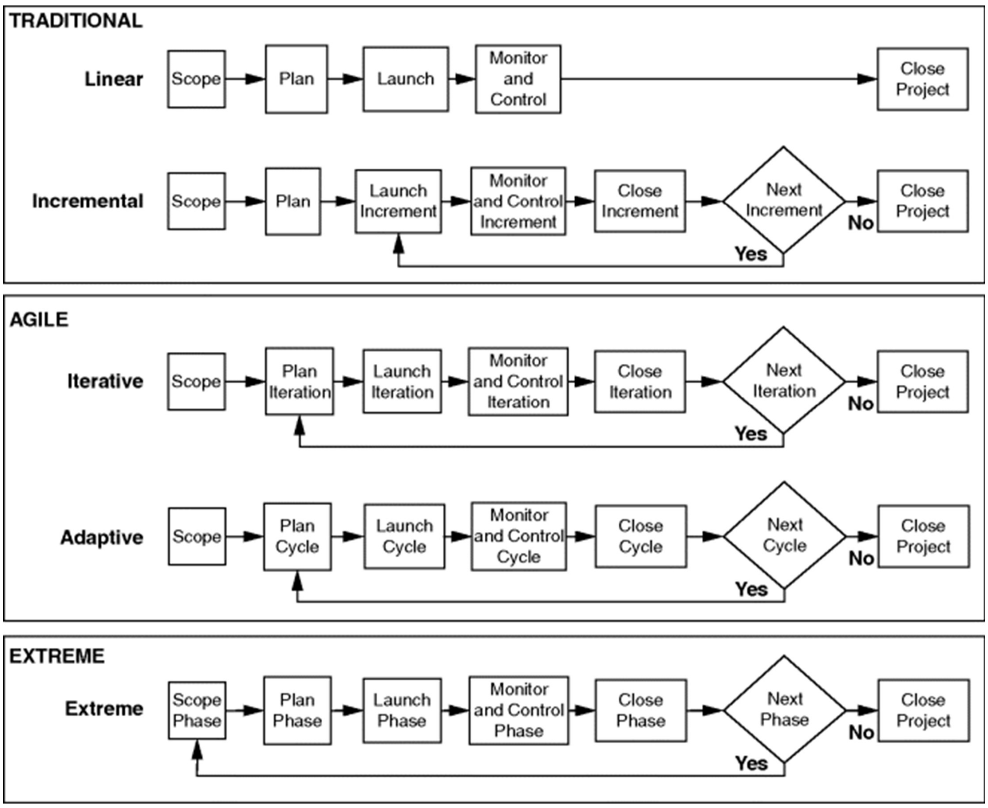


FIGURE 5: Project management lifecycles
Wysocki (2019)

Agile approaches are developed with recurring changes in mind. They have an iterative or adaptive lifecycle, frequent deliveries to handle uncertainty, and changes in requirements. Similar agile project phases are defined by DeCarlo (2004) and Highsmith (2009). The extreme approach has its dedicated lifecycle, where the main characteristic is that the scope can change after each phase. The different lifecycles defined by Wysocki (2011) for each approach are illustrated in Figure 5.

2.1.6 Traditional and agile project management

Traditional project management brings formal methods of planning and control (Conforto and Amaral, 2016; Kerzner, 2017; Salameh, 2014), however usually faces challenges with increasing level of uncertainty (Bergmann and Karwowski, 2018; Mattia et al., 2020). Highsmith (2009) defines *agility* as an ability to create and respond to change to create value in a turbulent business environment. Agile approaches are formed from the main principles of the agile manifesto (Fowler, Highsmith, et al., 2001).

Despite being significantly influenced by agile software development methods, APM has already gained popularity in a broader context (Bergmann and Karwowski, 2018; Bianchi et al., 2018; Conforto, Salum, et al., 2014; Owen et al., 2006). TPM tries to predict and minimize change (Ciric, Lalic, et al., 2019), while APM adapts to uncertainty and changes even in later phases of a project. In TPM, phases are rarely revisited, providing no feedback (Bergmann and Karwowski, 2018). APM explores and adapts to customer requirements in short cycles based on feedback, helps to identify errors early (Thesing et al., 2021), and maximizes value (Fowler, Highsmith, et al., 2001).

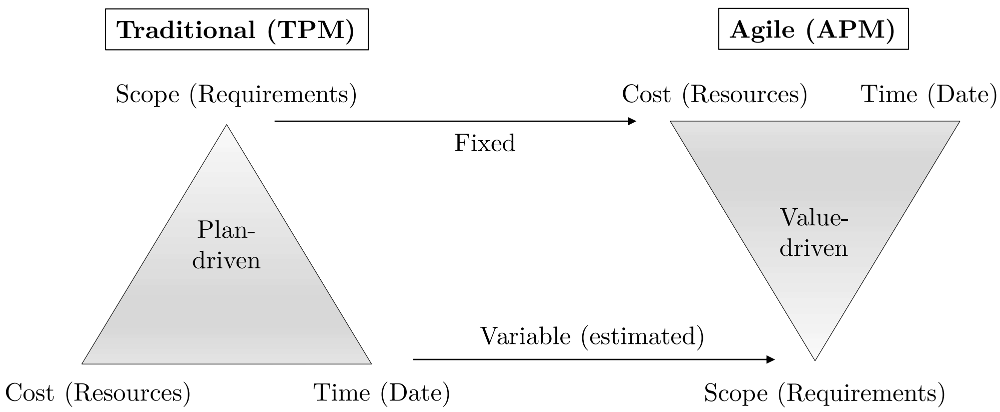


FIGURE 6: TPM and APM project triangles based on Leffingwell (2010)

APM assumes that cost, time (and quality) are fixed, and only the scope can change. APM focuses on prioritized items and requirements that offer the most business value in time. Figure 6 illustrates the difference between traditional and agile project triangles. Unlike TPM, where new features (scope) are added at the expense of cost and delivery date, APM has a rather small scope, and rapid deliveries at a high rate (Collyer and Warren, 2009), with a greater emphasis on communication over a process or plan (Fowler, Highsmith, et al., 2001). The frequent customer interaction and early concept testing result in a faster time to market and create a positive loop with higher customer satisfaction and economic benefit.

Ciric, Lalic, et al. (2019), identified possible challenges like the adaptation of organizational culture, incompatibility with organizational processes, work prioritization, and alignment among stakeholders. Implementing APM requires a change in the organizational culture which is a complicated process and needs support from management (Ciric, Lalic, et al., 2019; Loiro et al., 2019). Dumitriu et al. (2019) mentions larger organizations struggle to adopt agile e.g., due to managing interdependencies between projects and self-organizing teams.

TABLE 2: Comparison of TPM and APM characteristics
Based on Špundak (2014), Dybå and Dingsøyr (2008)

Characteristic	Traditional Project Management (TPM)	Agile Project Management (APM)
Requirements	clear initial requirements; low change rate	creative, innovative; requirements unclear
Clients	not involved	close and frequent collaboration
Documentation	explicit, formal documentation required	tacit knowledge
Project size	bigger projects	smaller projects
Management style	autocratic, prescriptive	affiliate, democratic
Organizational structure	bureaucratic, highly formalised	flexible, cooperative
Organizational support	use existing processes; bigger organizations	prepared to embrace agile approach
Team members	not accentuated; fluctuation expected; distributed team	co-located team; smaller team
System criticality	system failure consequences serious	less critical systems
Project plan	linear	complex; iterative
Quality control	planned in time in details	ongoing control of subresults with client's expectations

The ratio of projects falling into the category of agile is around 70%, according to international research done by Wysocki (2011). Despite the relatively small fraction of traditional projects, project managers continue to apply traditional methods.

Regarding the success of technology projects, Johnson (2020) highlights from the regular CHAOS report that agile projects are three times more likely to succeed[2] [3] than traditional projects, and traditional projects have double the chance to fail[4] and the trend continues. The study also highlights interesting differences regarding project size[5]. Small agile projects succeed three times more than large agile projects; large agile projects succeed twice more than large traditional projects. Small traditional projects succeed six times more than large traditional ones. Agile and traditional only come close to each other in the small project category. The study suggests that large projects should be avoided by breaking them down into smaller ones, reducing risk. The detailed success rates are shown in Table 3.

2.2 Multilevel project management

Management of individual projects is extensively studied in the project management literature (see e.g., Artto et al., 2008; Atkinson, 1999; Baccarini, 1996; Cooke-Davies, 2002; Kerzner, 2017; Munns and Bjeirmi, 1996; Pinto and Slevin, 1987), yet the research addressing multiple projects and their environment is still scarce and needs more focus, as the majority of value resides in the multiproject setting (Jerbrant, 2013; Liberatore and Pollack-Johnson, 2003; Maroto et al., 1999; Payne, 1995).

[2]Successful project means on time, on budget and satisfied customer
[3]Challenged project means late or over budget, with less than satisfied customer
[4]Failed project means canceled before it is resolved, or resolved and not used
[5]Project size is measured in hours of productive labor:
less than 10,000 considered as small; 30,000-60,000 as medium; 60,000-100,000 as large project

TABLE 3: Project success rates by method and size
Based on Johnson (2020)

Size	Method	Successful	Challenged	Failed
All	Traditional	13%	59%	28%
	Agile	42%	47%	11%
Large	Traditional	8%	56%	36%
	Agile	19%	56%	25%
Medium	Traditional	9%	66%	25%
	Agile	34%	53%	13%
Small	Traditional	45%	46%	9%
	Agile	59%	36%	5%

Several conceptual categorizations can be found in the literature regarding multiple projects. Depending on how closely the subprojects are related to each other (via goals or through the common resources used to accomplish them), one can distinguish multiprojects, programs, and project portfolios (Patanakul and Milosevic, 2009) in the multiproject environment (MPE). Multilevel project management is understood as the approach to handle not only projects, but multiprojects, programs, and portfolios.

Figure 7 shows a possible organizational setting (Patanakul and Milosevic, 2009).

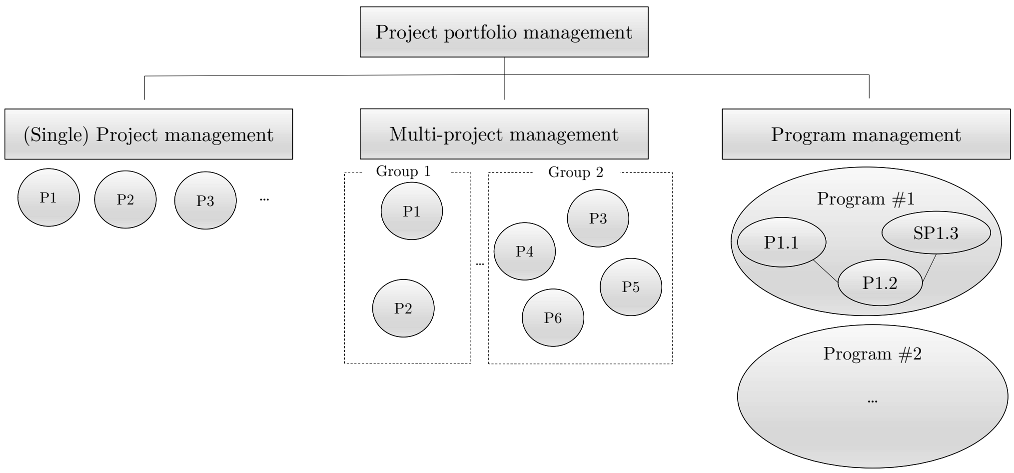


FIGURE 7: Multiproject environment (MPE)
Adapted from Patanakul and Milosevic (2009)

The topic of the dissertation focuses mostly on multiprojects. From the existing definitions, the term defined by Fricke and Shenbar (2000) is used: *"Multi-project is understood as a setting in which more than one project is carried out at the same time. The projects vary in size, importance, required skills and urgency, are in various stages of completion and are using the same pool of resources."* Furthermore, it is assumed that these projects are not necessarily or directly interrelated.

Projects in a program are *"[...] mutually dependent, they share a common goal, and they lead to a single product or service"* (Patanakul and Milosevic, 2009).

Rajegopal et al. (2007) defines a project portfolio as *"[...] a group of projects and programs and other work that are bound together to enable the effective management of work and to meet strategic goals of the business."*

Single projects are also part of the multiproject environment. The main difference, as opposed to a multiproject, is that traditional (single) project managers are responsible for coordinating activities within a project.

Thus, in organizations managing multiple projects, management faces additional challenges (Elonen and Artto, 2003). They need to carry out projects that run at the same time, share common resources and have different (often conflicting) goals, complexity and timelines. Such simultaneous handling of projects requires effective identification of the different project and organizational goals and the ability to quickly switch between projects (Patanakul and Milosevic, 2009). Consequently, multiproject managers have to deal with problems that are different from individual projects (Dooley et al., 2005; Pellegrinelli, 1997; Platje et al., 1994), including, for example, a lack of resources, which appears widely in the literature (Cooper et al., 2000, 2001; Zika-Viktorsson et al., 2006).

Balancing limited resources across projects is described as the prime challenge of multiprojects (Dooley et al., 2005; Engwall and Jerbrant, 2003) and has great importance. In a multiproject environment, projects must negotiate priority for resources on an almost daily basis. Prioritization can take many forms in an organization's project management processes. Organizations' primary goal is often to win increasingly more projects, not necessarily considering the level of available resources. Short-term problem solving is common (Delisle, 2020) that can lead to a vicious circle. The organizations cannot identify the projects that are a priority to implement, thus disrupting the implementation of other projects (Engwall and Jerbrant, 2003; Spuhler and Biagini, 1990) increasingly competing against each other (Dooley et al., 2005; Kuprenas, 2003). Fricke and Shenbar (2000) finds that factors that have low relevance for single projects, such as the assignment of resources, prioritization, and customized management style, play a significant role in the success of multiproject management.

In addition, leaders of organizations often choose short-term and easy-to-implement or low-budget projects, such as modifying or expanding products. However, this approach will reduce their future success potential and competitive advantage (Cooper et al., 2000, 2001). Elonen and Artto (2003) also drew attention to this bias in decision-making.

It is evident from the diversity of problem areas affecting multiproject management (Elonen and Artto, 2003), that a better understanding and more complex tools are necessary to manage these projects effectively. The term multilevel is used in this context to emphasize and express the different scales and their mostly interrelated challenges, which are inherently present when managing multiple projects.

2.3 Project planning and scheduling

Project planning is defined as the establishment of a predetermined course of action for the forecasted environment (Kerzner, 2017). Specifically, it covers the formulation of goals and objectives that explain the work that has to be done, the timeline for the project, and the necessary resources that are required to accomplish the objectives of the project (Zwikael, 2009). Project planning helps to reduce uncertainty, to improve the efficiency of the operation, and to obtain a better understanding of project objectives while providing a basis for monitoring and controlling work (Kerzner, 2017). The planning process (re)defines the objectives and selects the best alternatives to achieve these objectives (PMI, 2021).

Project scheduling is also an integral part of project management as it establishes an allocation of resources over time to perform a set of activities. It is typically concerned with two dependencies; activities compete for scarce resources that are carrying them out, and the precedence constraints between pairs of activities that define the order of execution (Hartmann and Briskorn, 2021).

The traditional network-based project planning tools (see e.g., Eisner, 1962; Kelley Jr, 1961; Roy, 1962; Wiest, 1981) are no longer able to fully support the strategic decisions of companies (Kosztyán, 2012). These methods cannot handle projects where certain activities must be skipped due to time, cost, or resource constraints. Network-based project planning does not consider that a project can have several possible outcomes and does not provide an opportunity to prioritize activities and subprojects. The characteristics of these methods make it impossible to consider and handle cycles in graphs (Kosztyán and Kiss, 2011). In addition, network project planning only supports the traditional project management approach and neglects other emerging types, such as agile or hybrid.

2.3.1 Matrix-based project planning

Projects are usually represented as graphs in which activities (i.e., tasks) are depicted either with arcs (activity-on-arrow [AoA] networks) (Demeulemeester, Herroelen, et al., 1996) or nodes (activity-on-node [AoN] networks) (Ren et al., 2021). The matrix representation of projects usually describes an AoN network (Minogue, 2011). Matrix-based project planning can eliminate the shortcomings of traditional methods; it is possible to plan agile and hybrid projects as well as traditional projects. The matrix-based project planning methods are often based on the design (or dependency) structure matrix (DSM) (Kosztyán, 2015; Steward, 1981). The domain mapping matrix (DMM) is an extended version of the DSM, with multiple domains (Danilovic and Browning, 2007). With the numerical DSM (NDSM), the level of dependency relationship between two activities can also be plotted (Chen et al., 2003; Tang et al., 2010). With the stochastic network planning method (SNPM) developed by Kosztyán and Kiss (2010), probabilities or priorities regarding the completion of the activities can be considered. With these methods, various possible network plans can be modeled due to the parallel or sequential completion mode of the tasks. In the case of the project expert matrix (PEM) (Kosztyán, Kiss, et al., 2010), which was created as a further development of the SNPM, the relationships between the activities can be uncertain or stochastic, as can the completion of the activities in the project scenario. The project domain matrix (PDM) proposed by Kosztyán (2015) is used to cope with multiple domains.

Kosztyán (2015) suggested a project domain matrix (PDM), that can be used for both single and multimodal project plans. PDMs allow mandatory and supplementary tasks with priorities and flexible dependencies between tasks. Kosztyán (2020) later extended this matrix-based model to address multiple projects, programs and project portfolios. This matrix-based multiple project management model is denoted M^4.

2.3.2 Resource-constrained (multi)project scheduling problem

The resource-constrained project scheduling problem (RCPSP) is one of the most studied problems in the project planning literature since the 1950s. The classical problem consists of a set of activities that need to be scheduled, subject to precedence and resource constraints to optimize an objective function, e.g., to minimize the overall duration of a project. Several researchers developed both exact and heuristic solutions and various extensions have been investigated. Hartmann and Briskorn (2021) provides an overview and classification of the most important extensions of the RCPSP.

An important extension, the resource-constrained multiproject scheduling problem (RCMPSP), deals with multiple projects using the same resources that must be scheduled without violating the resource constraints. Different variants of the resource-constrained multiproject scheduling problem have also been studied by several researchers since the first introduction of the problem.

Only a minority of the scheduling algorithms address multilevel projects, and they follow the traditional scheduling methodology. In this case, the activities still have a fixed order of execution (Pellerin and Perrier, 2019). Recent algorithms usually decompose multilevel projects into collaborative or competitive single projects, which are solved in a distributed way using agents (Liu et al., 2019; Song et al., 2018). However, these approaches also assume fixed logic plans of projects. For example, software development projects are usually run as a part of a multiproject and are flexible, such as agile, hybrid or extreme project management (Marchenko and Abrahamsson, 2008).

For a survey of the different RCMPSP extensions, see Hartmann and Briskorn (2021), Issa and Tu (2020), and Van Eynde and Vanhoucke (2020). A comprehensive, state-of-the-art survey of the different methods, variants, features, and objectives are also given in (Sánchez et al., 2022).

2.3.3 Flexibility of projects

Projects managed by traditional methods assume that the activities have a fixed order of execution (Pellerin and Perrier, 2019) in the project plans. Software development projects are typically multiprojects, having flexible characteristics like agile, hybrid, or extreme projects (Marchenko and Abrahamsson, 2008), thus the dependencies of tasks are not necessarily fixed (Kosztyán, 2015). The priorities for these tasks are set to select which tasks will be either completed in a short project (a so-called sprint), postponed, or skipped. Agile project management allows such flexible dependencies and priorities of task completion (Kosztyán, 2015; Sajad et al., 2016), while extreme projects allow new and unplanned tasks for common changes in stakeholder requirements. Hybrid approaches allow traditional trade-off methods besides flexibility with multimode task completions (Kosztyán and Szalkai, 2020). Today, flexible approaches are often used in nonsoftware development projects (Hidalgo, 2019a; Papadakis and Tsironis, 2018), such as R&D (Som de Cerff et al., 2018), new product development (NPD) projects (Ciric, Lalic, et al., 2018), construction (Arefazar et al., 2022) and maintenance (Kosztyán, Pribojszki-Németh, et al., 2019).

Broadly defined, flexibility is the magnitude of the room for scheduling decisions (for an overview of the different definitions, see Bernardes and Hanna (2009)). (Multi) project scheduling is open to several flexibility types; time-related or scheduling flexibility can result from slacks or topological floats (see Tavares (1999) and Vanhoucke, Coelho, Debels, et al. (2008)) in the project plan. This type is the most obvious, and it frequently occurs even in traditional projects. In this case, the precedence relations and the implementation modes remain the same, and only the scheduled start and finish times of the tasks change. Hauder et al. (2020) shows how this flexibility can change the logistical (storing or conveying) task duration, however, it can be implemented by defining the minimal and maximal time lags of an activity-on-node project network (Ren et al., 2021).

The second type is activity (i.e., task) or modal flexibility in which a task can be performed in several modes or the same result can be achieved by carrying out one of the different sets of tasks and utilizing different resource combinations. These alternative (sets of) tasks are modeled by Petri nets in Čapek et al. (2012), by mandatory and optional choices in the project network (Kellenbrink and Helber, 2015), or by the AND/OR network in Tao and Dong (2018). These works extended the resource-constrained (multiple) project scheduling problems (RCPSP or RCMPSP) with alternative activity chains (RCPSP-AC or RCMPSP-AC). Combined with time-related flexibility, Hauder et al. (2020) defined the problem set of the resource-constrained multiple project scheduling problem with alternative activity chains and time-related flexibility (RCMPSP–ACTF).

The third type is dependency flexibility. Some logical dependencies can be omitted if the project task technology does not require a strict sequence. Omitting a dependency lifts the restriction of sequential execution and allows the associated tasks to be performed in parallel or in an arbitrary, relative order.

The fourth type is scope flexibility, in which some low-priority tasks can be omitted or postponed to a later project. This situation reduces the resource demand and can shorten the project duration by sacrificing quality or fulfillment level. The latter two flexibility types appear typically but not exclusively in agile projects (Kosztyán, 2015). Since these flexibilities affect the logical structure of a project, i.e., which tasks are performed and according to which logical dependency they are performed, hereinafter, dependency and scope flexibility are together called structural flexibility.

While structurally flexible projects require flexible project plans, allowing the possibility of project restructuring, task reprioritization or both according to the customer's requirements, most project-planning methods assume a fixed (Franco-Duran and Garza, 2019) logic plan or a limited number of scheduling alternatives (Čapek et al., 2012; Creemers et al., 2015; Hauder et al., 2020; Kellenbrink and Helber, 2015; Servranckx and Vanhoucke, 2019b; Tao and Dong, 2018). In addition, a few matrix-based methods are available for scheduling structurally flexible projects (Kosztyán, 2015; Kosztyán and Szalkai, 2020); among these, some task realizations and dependency occurrences are treated as variables during the planning phase.

2.4 Project related databases and indicators

Project related data

Project databases play a key role in the research of different scheduling and resource allocation methods (Brucker et al., 1999; Hartmann and Briskorn, 2010, 2021) by making them comparable and developing new methods (Franco-Duran and Garza, 2019). Three types of data sources can be found in the literature: notional data (e.g., illustrational examples), artificial (generated) data, and empirical (collected) data.

Individual projects are available in various databases, such as Patterson (Patterson, 1976), SMCP and SMFF (Kolisch et al., 1995), PSPLIB (Sprecher and Kolisch, 1996), RG300 and RG30 (Debels and Vanhoucke, 2007; Vanhoucke, Coelho, Debels, et al., 2008), Boctor (Boctor, 1993), MMLIB (Peteghem and Vanhoucke, 2014), the real-life project database by (Batselier and Vanhoucke, 2015), or sets of individual or multiple projects, including MPSPLIB (Homberger, 2007), BY (Browning and Yassine, 2010a), RCMPSPLIB (Vázquez et al., 2015), and MPLIB (Van Eynde and Vanhoucke, 2020).

All these databases contain tasks and dependencies between tasks and renewable resources. However, most databases do not include costs, quality, or nonrenewable resources. Only two datasets consider structural flexibility with alternative subgraphs, the RCPSP-PS dataset (Kellenbrink and Helber, 2015) and ASLIB dataset (Servranckx and Vanhoucke, 2019a). Several databases contain only one completion mode (namely, those of Patterson (Patterson, 1976), SMCP and SMFF (Kolisch et al., 1995), PSPLIB (Sprecher and Kolisch, 1996), RG300 and RG30 (Debels and Vanhoucke, 2007; Vanhoucke, Coelho, Debels, et al., 2008), and the real-life database Batselier and Vanhoucke (2015)), while others contain multiple completion modes (namely, PSPLIB (Sprecher and Kolisch, 1996), Boctor (Boctor, 1993), and MMLIB (Peteghem and Vanhoucke, 2014)).

Some criticism has arisen regarding these simulated project databases. Peteghem and Vanhoucke, 2014 reported four shortcomings of the widely used PSPLIB. One limitation is the low diversity in the complexity of topology networks indicated by the order strength (OS) values. Some of the instances are infeasible. In general, the instances are easy to solve with older procedures. The authors also found that Boctor's dataset contains mainly serial projects, and the renewable resources are hardly restricted by the constraints.

There are further datasets that are worth mentioning, although it is not possible to list all the datasets for the different RCPSP variants. The MT dataset (Vanhoucke, 2010b) is mainly used for schedule risk analysis and earned value management, containing project structures that can be combined with ResSet, have additional resource data and result in the NetRes dataset (Vanhoucke and Coelho, 2018). DC1 (Vanhoucke, Demeulemeester, et al., 2001) and DC2 (Vanhoucke, 2010a) are studied within the context of the RCPSP with discounted cash flows. The CV set (Coelho and Vanhoucke, 2020) contains RCPSP instances that are difficult to solve. MISTA2013 (Wauters et al., 2016) is a dataset and generator for the multi-mode resource-constrained multiple project scheduling problem (MRCMPSP) and combines instances from PSPLIB. The BL (Baptiste and Pape, 2000) and PACK (Carlier and Néron, 2003) datasets are also modifications of the PSPLIB, designed for the context of highly disjunctive and cumulative scheduling of RCPSP, respectively.

Other sources of project data are project generators, such as ProGen (Kolisch et al., 1995), Progen/max (Schwindt, 1995) and Progen/πx (Drexl et al., 2000), RanGen1 and RanGen2 (Demeulemeester, Vanhoucke, et al., 2003; Vanhoucke, Coelho, Debels, et al., 2008), RiskNet (Tavares, 1999), and the multi-network problem generator (MNPG) by Browning and Yassine, 2010a. These project generators have been used to generate several project databases in a controlled manner, such as, the PSPLIB (Sprecher and Kolisch, 1996), the RG300 and RG30 (Debels and Vanhoucke, 2007; Vanhoucke, Coelho, Debels, et al., 2008), the MMLIB (Peteghem and Vanhoucke, 2014), and BY (Browning and Yassine, 2010a).

Project related indicators

Project related indicators can be used to classify existing project plans based on different characteristics and as input parameters for the random generation of artificial project plans. The indicators for project plans can be classified into two main groups. The first group characterizes the project structure, including measures of its complexity, and the second group characterizes the project demands, such as resource, time and cost. There are several indicators proposed in the literature. A general overview of indicators and databases is given by Vanhoucke, Coelho, and Batselier (2016). For multiprojects, Browning and Yassine (2010b) gives an overview of the existing indicators, which was extended by Van Eynde and Vanhoucke (2020) recently, showing the relevance and interest for the research of different indicators.

2.5 Synthesis of challenges from literature

In this chapter, the main challenges from the literature are critically reviewed and summarized providing a base for the current research in addition to the already reviewed parts. Many authors are dealing with different sources of uncertainty for projects, for example, Hazır and Ulusoy (2020) gives a classification for the different forms of such variability.

The situation gets more complicated when it comes to multiproject planning, as disturbances to one project influence the others and the situation altogether as a whole becomes less predictable (Zika-Viktorsson et al., 2006). With the high integration of parts, interdependencies (e.g., shared resources) and interactions of projects, the organization also becomes negatively affected, and the need for planning and control increases. Gustavsson (2016) suggests limiting the number of interconnected projects and tasks to avoid project overload, a situation in which fragmentation, disturbances and disruptions are highly prevalent (Zika-Viktorsson et al., 2006). Engwall and Jerbrant (2003) identifies challenges with multiple projects as a source of complexity in organizations. They identified issues with long projects which are not broken down into smaller projects, making detailed planning difficult. In addition, changing objectives during projects was also found as a problem. Similarly, Petit and Hobbs (2010) found the change in scope as the most important source of uncertainty at the portfolio level.

Agile projects address uncertainty and reduce project failure (Conboy, 2010; Johnson, 2020), however, according to Dybå and Dingsøyr (2008), because of the higher focus on flexibility and iterative work structures, they also become less predictable. Špundak (2014) suggests that a software development project's scope could change up to 30% during iterations. Hazır and Ulusoy (2020) groups requirement changes into one of the major factors of uncertainty. Due to changing work content, the project network might need to be modified (tasks and relations added or deleted). Zhu et al. (2005) categorizes sources of uncertainties in projects and considers deleted or new activities or precedence relations in projects as network disruptions.

As agile became common for single projects, the focus shifted to the effects of agile projects on multiprojects and portfolios. In this context, the agile characteristics present challenges to the traditional management of multiple projects (Jonas, 2010; Kaufmann et al., 2020; Sweetman and Conboy, 2018) and adjustments such as agility and adaptiveness need to be added (Krebs, 2008; Leffingwell, 2010) to the current state of practice (Stettina and Hörz, 2015) to avoid disjointed, incoherent conflicting agile projects. Unlike single agile projects, agile projects within a multiproject or portfolio context increase difficulties for the management (Rautiainen et al., 2011; Stettina and Hörz, 2015) and introduces a high degree of complexity, with an increased number of interactions due to changing customer needs that also need to be aligned with organizational strategy (Sweetman and Conboy, 2013, 2018). Such dynamic projects need a higher coordination effort and increased adaptiveness to align to the portfolio due to constant change, improvisation and self-organization (Highsmith and Highsmith, 2002; Schwaber and Beedle, 2002; Sweetman and Conboy, 2013). Regarding size and scalability, Uludag et al. (2018) identified 71 challenges for large-scale agile development and Hobbs and Petit (2017) mentions several challenges related to large-scale agile multiprojects. Dumitriu et al. (2019) further categorizes such challenges on project and organizational levels. In general, attempts for agility on the multiproject and portfolio level have been criticized by agile pioneers for being overly complicated (Vähäniitty et al., 2012).

TABLE 4: Summary of challenges from literature for TPM, APM for single and multiple projects
Source: own collection

Challenge cat.	Traditional project	Agile project	Traditional multiproject	Agile multiproject
Uncertainty	Hazır and Ulusoy (2020) and McLain (2009)	Hans et al. (2007) and Laslo and Goldberg (2008)	Gustavsson (2016), Hans et al. (2007), Hazır and Ulusoy (2020), and Laslo and Goldberg (2008)	Dingsøyr and Moe (2014)
Complexity	Danilovic and Sandkull (2002)	Sohi et al. (2016)	Danilovic and Sandkull (2002) and Hans et al. (2007)	Sweetman and Conboy (2018)
Dependencies	Duimering et al. (2006) and Eppinger et al. (1989)	Strode (2016)	Engwall and Jerbrant (2003) and Hans et al. (2007)	Wińska and Dąbrowski (2020)
Priorization	Miranda Mota et al. (2009)	Bakalova et al. (2011), Karlesky and Vander Voord (2008), and Racheva et al. (2008)	Elonen and Artto (2003), Engwall and Jerbrant (2003), and Fricke and Shenbar (2000)	Vähäniitty et al. (2012)
Goals/obj.	Shenhar and Dvir (2007), Van Wyngaard et al. (2012), and Williams (2005)	Serrador and Pinto (2015)	Laslo and Goldberg (2008)	Kaufmann et al. (2020)
Resource alloc.	Turner and Müller (2003)	Dybå, Dingsøyr, and Moe (2014) and Hoda and Murugesan (2016)	Elonen and Artto (2003), Engwall and Jerbrant (2003), Laslo and Goldberg (2008), and Zika-Viktorsson et al. (2006)	Hoda and Murugesan (2016) and Stettina and Smit (2016)
Coordination	Andres and Zmud (2002)	Strode et al. (2011)	Elonen and Artto (2003)	Turek and Werewka (2016)
Adoption	Gareis (1991)	Boehm and Turner (2005)	Aritua et al. (2009) and Elonen and Artto (2003)	Hobbs and Petit (2017) and Marchenko and Abrahamsson (2008)
Location	Aarseth et al. (2014), Evaristo and Scudder (2000), and Evaristo, Scudder, et al. (2004)	Abad et al. (2010), Boehm and Turner (2003), Bose (2008), and Papadopoulos (2015)	Evaristo and Van Fenema (1999)	Lee and Hur (2010)

As shown in Table 4, there are many challenges identified related to projects and their management. It is clear that the uncertainty is amplified with the size and number of projects, the number of dependencies, especially when multiple projects are interacting with each other and competing for scarce resources leading to their overload. Complexity is ever-increasing due to the number of interconnected elements, and changes in scope and objectives expand the need for coordination. Adopting agile methods alone is a complicated task for an organization, even if agile brings proven benefits, it can also bring some degree of uncertainty in a traditional environment. All things considered, the two main interest, on the one hand, the literature for flexible projects like agile and hybrid; and on the other hand, multiprojects and portfolios are the areas where the previously summarized challenges are culminating. Surprisingly, their literature is very scarce and definitely needs more focus.

Vanhoucke (2018) emphasizes several areas for further research: the balance between empirical and artificial project data; the classification, generation and structuring of project data; the incorporation of flexibility; and a better understanding of multiprojects.

2.6 Research assumptions

By revisiting the research questions formulated in Section 1.2, and critically reviewing the findings and relationships within the literature, it becomes possible to formulate the corresponding research assumptions. The four research assumptions are as follows:

RA1: A model can be created that unifies the different project and multiproject database formats from the literature, including time, cost, renewable-, nonrenewable-resource and quality demands. Existing databases can be imported and further extended with flexible tasks and dependencies into a single, matrix-based database.

RA2: Flexible project plans can be generated from existing traditional (multi)project plans and new possible structures can be added to the model to improve the planning process.

RA3: Existing project-related indicators for topology, time- and resource-related demands can be adapted for flexible projects and multiprojects to analyze the effects of flexibility.

RA4: Flexible multilevel projects can be scheduled and near-optimal solutions can be found. A simulation framework can be constructed to handle flexible dependencies and supplementary tasks.

Chapter 3

Methods

3.1 Data sources

The different datasets and libraries mentioned in this dissertation were collected from the project scheduling literature. During the research, suitable data sources were identified that are commonly used and shared by scholars to evaluate scheduling approaches and find the best solutions. The first challenge when dealing with data from literature is usually accessing the different datasets published by various researchers in the field. One of the intentions of the research was also to review, collect and share the wide range of available data.

The second challenge arises when the data must be handled, as they often have unique formatting and a structure that lacks proper documentation. This situation might lead to additional reverse engineering efforts that increase the research time and, of course, involve their own risks. Thus, there is a need to harmonize and integrate a wide range of datasets into a library that is accessible, ready to process, and respects the original content.

3.1.1 Data collection and processing

To overcome limitations such as a lack of standardization and database integration efforts, as part of the research, a parser tool was developed (a software program that reads inputs, e.g., a text file for further processing) for the commonly used datasets found in the project scheduling literature. The parser extracts all information from the existing libraries or the output of project generators in an automated and reproducible way. The resulting data are ready for research and analysis and, if needed, can be further adapted to various formats or platforms. Although the parser covers most available formats, the aim is to extend the list of supported extensions continually. The two main dataset categories considered in the current study are generated and empirical. For an overview of the selected databases, see Table 5).

TABLE 5: Selected project databases and their attributes
Source: own edit

Name	Project Plan	Completion Modes	Projects	Demands	Cited as
Patterson	Generated	Single	Single	Time, renewable resources	Patterson, 1976
PSPLIB	Generated	Single, Multiple	Single	Time, re/nonrenewable resources	Sprecher and Kolisch, 1996
RG30, RG300	Generated	Single	Single	Time, renewable resources	Vanhoucke, Coelho, Debels, et al., 2008
SMCP, SMFF	Generated	Single	Single	Time, renewable resources	Kolisch et al., 1995
Boctor	Generated	Multiple	Single	Time, renewable resources	Boctor, 1993
MMLIB	Generated	Multiple	Single	Time, re/nonrenewable resources	Peteghem and Vanhoucke, 2014
Real-life	Collected	Single	Single	Time, cost, renewable resources	Batselier and Vanhoucke, 2015
MPSPLIB	Generated	Single	Multiple	Time, renewable resources	Homberger, 2007
BY	Generated	Single	Multiple	Time, cost, renewable resources	Browning and Yassine, 2010a
RCMPSPLIB	Generated	Single	Multiple	Time, renewable resources	Vázquez et al., 2015
MPLIB1, MPLIB2	Generated	Single	Multiple	Time, renewable resources	Van Eynde and Vanhoucke, 2020

The parser was written in MATLAB (Mathworks, 2021) and works as follows. It reviews the existing project files in search of network-related data (tasks and their precedence relations); time-related and resource-related data, including demands and constraints, and if present, data of the costs and multiple modes of completion. Additional fields are captured from the original data files even if the input is not used directly for scheduling (e.g., the MPM-time field in the case of PSPLIB). The obtained data were then preprocessed into a matrix-based representation and saved to a MAT file that contains the data as variables. This container file can be easily loaded into MATLAB's workspace. The parser addresses renewable resource types, and the tool is designed such that it can be extended easily to use other types (e.g., nonrenewable and doubly constrained resource types). From all parsed libraries and datasets considered, the datasets were selected specifically for the dissertation topic. To allow a straightforward comparison of the different indicators, mainly single-mode examples were selected, and cost-related data were not considered, as only one library contains it.

In addition, the source file format is heterogeneous; therefore, if a scholar wants to test a new method in multiple databases, different parsers must first be written for each project database. An example format is shown in Figure 8 for one of the earliest and simplest project instances by Patterson (1976). However, heterogeneity is not simply a matter of format; tasks might be assigned with different requirements, such as duration, cost, or renewable and nonrenewable resources, furthermore, they can have multiple execution modes.

In Chapter 4, the instances of the real project dataset (Protrack) are compared to the simulated ones, and the effects are evaluated by introducing flexibility to implementation priority or precedence relations on the project properties.

When generating a new project, only very few structure-related, time-related and resource-related indicators can be set. Therefore, the existing project generators can only generate a few undiscovered and untested project structures. Although developing of a new project generator was not a primary aim of this study, considering flexibility extends the domain of the indicator values.

```
pat1.rcp
14  3
2   1   2
0   0   0   0   3   2   3   4
6   1   0   0   2   9   10
4   0   0   0   3   5   6   7
3   0   0   0   2   8   11
1   0   0   0   1   10
6   1   0   1   1   12
2   1   0   0   2   8   11
1   0   0   0   1   13
4   0   1   1   1   14
3   0   0   1   1   12
2   0   0   1   1   12
3   0   1   0   1   13
5   0   0   0   1   14
0   0   0   0   0
```

FIGURE 8: An example of an early project instance by Patterson (1976)

3.1.2 Data selection and construction

A matrix-based model is proposed based on the M^4 model by Kosztyán (2015, 2020), to unify the heterogeneous project databases. The decision was made according to the goal to effectively represent all features of the widely accepted databases, i.e., individual and multiple projects, single- and multimodal completions, renewable and nonrenewable resources.

The proposed matrix-based method is called the *unified matrix-based project-planning model* (UMP) and it contains two mandatory and four supplementary domains (see Figure 9).

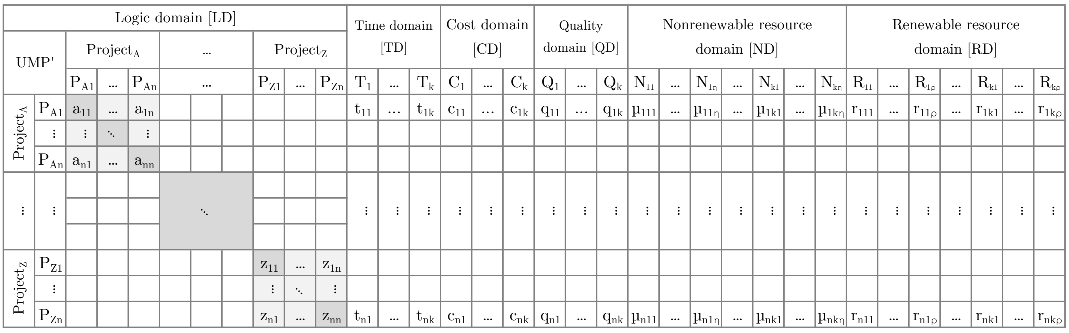


FIGURE 9: Structure of the unified matrix-based project-planning model (UMP)

LD The logic domain is an n by n matrix, where n is the number of tasks. Each cell contains a value from the [0,1] interval.

TD The time domain is an n by k matrix with positive real values, where k is the number of completion modes.

The first mandatory domain is the logic domain, $\mathbf{LD} \in [0,1]^{n\times n}$. The diagonal values in **LD** represent the task priority values. If a diagonal value is 0, the task will not be completed, and if the diagonal value is 1, the task is mandatory. If the

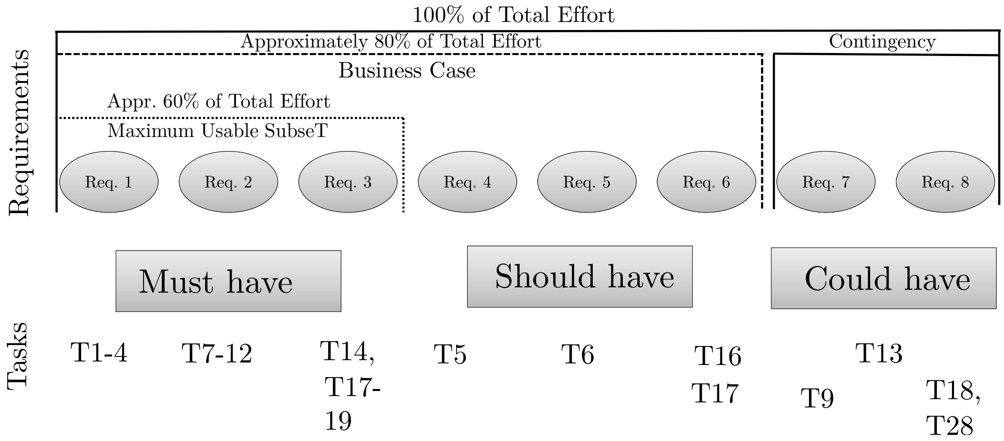


FIGURE 10: An example of MoSCoW of the prioritization of requirements and tasks (based on the guide of DSDM (Stapleton, 1997))

diagonal value is between 0 and 1, the task is supplementary, indicating that depending on the decision, it will be either completed or omitted/postponed. In the case of flexible projects, tasks are prioritized by the product owner according to their business value and the risks involved in their development (Abad et al., 2010). To help decision-makers prioritize task completion, several methods, such as MoSCoW rules, are available, and the requirements are prioritized based on their importance by sorting them into the four groups of must-have, should-have, could-have and will-not-have features. In addition to the categories, tasks can be ranked by their importance, or the importance/priority values can be calculated by the analytical hierarchy process (AHP) method (Srivastava et al., 2021). The prioritization of task completions is an essential part of all flexible, such as agile, hybrid, and extreme project management methods. Nevertheless, in this dissertation, only the rate of the existing supplementary (i.e., lower priority) tasks was analyzed; therefore, priority rankings were not studied. Figure 10 shows an example of MoSCoW prioritization of requirements applied by the agile Dynamic System Development Method (DSDM). DSDM method was one of the first, which suggests MoSCoW method prioritize task completion (Stapleton, 1997). This technique indicates that the rate of the mandatory tasks should be approximately 60%. Nevertheless, the concept of task prioritization is generally applied in most agile techniques (Dingsøyr, Nerur, et al., 2012; Govil and Sharma, 2021).

A task can fulfill more than one requirement (see T13); however, usually, to fulfill requirements, more than one task should be completed. In an agile project, only 'MUST (called Maximum Usable SubseT) have' tasks (appr. 60% of tasks and efforts) will be completed necessarily; the other tasks (appr. 40%) are supplementary tasks with a different class of priorities.

The out-diagonal values represent the dependencies between the tasks. If an out-diagonal value $a_{ij} = l_{ij} = [\mathbf{LD}]_{ij}$ $(i \neq j)$ is 1, task i precedes task j. In the case of $l_{ij} = 0$, no precedence relation exists from task i to task j. If $0 < l_{ij} < 1$, a flexible dependency exists between task i and task j, indicating that task i may

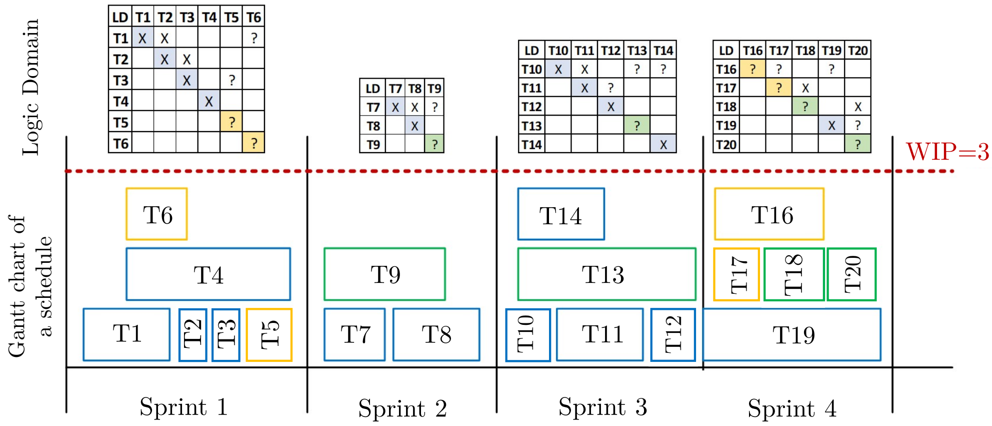


FIGURE 11: Example of a schedule of prioritized tasks with the SCRUMBAN method ('X'=1 represents mandatory (**M**ust have) tasks in diagonal, fixed dependencies in out-diagonal; 0<'?'<1 represents supplementary (either **S**hould have or **C**ould have) tasks in diagonal, or flexible dependencies in out-diagonal.

precede or follow task j depending on managers' (algorithm) decisions. All flexible techniques, such as agile, hybrid, or extreme, require flexible dependencies between tasks (Ciriello et al., 2022; Fernandez and Fernandez, 2008).

Since none of the project networks from the considered databases contains any cycles, they can be ordered topologically, and the logic domain of the topologically ordered project networks is an upper triangular matrix (formally, $l_{ij} = 0$ if $i > j$). Although the matrix-based representation does not require acyclic structures, and feedback can be resolved (see, e.g., in Kosztyán, 2015) since most indicators are defined for acyclic project structures, the upper triangular logic domain is considered for the topologically ordered tasks in the rest of this dissertation. Flexible project management allows iterations; however, the databases lack cycles; thus, we can investigate only one iteration at a time. Figure 11 shows how to schedule prioritized tasks using a SCRUMBAN method. SCRUMBAN is a combination of SCRUM, which is the first agile method suggesting iterations called sprints (Hidalgo, 2019b), and the KANBAN, which limits parallel work-in-progress (WIP) tasks (Williams, 2010).

The other mandatory UMP domain is the time-related domain. The positive values of the time domains represent the possible task durations. For each task, k kinds of durations can be assigned; the duration values may also match each other.

Matrix-based methods can also address general precedence relations (GPRs) (Minogue, 2011); however, most databases allow only finish-to-start (F-S) relations between tasks. F-S relations indicate that a successor task can be started only if all predecessor tasks have been finished. It is assumed that tasks can only have F-S relations.

The additional supplementary domains are as follows:

CD The cost domain, is an n by k nonnegative matrix of the task costs

QD The quality domain, is an n by k, nonnegative matrix of the task quality parameters, where the quality parameters are between [0,1]

ND The nonrenewable resource domain, is an n by $k \cdot \eta$ nonnegative matrix of nonrenewable resource demands, where η is the number of types of nonrenewable resources

RD The renewable resource domain, is an n by $k \cdot \rho$ nonnegative matrix of renewable resource demands, where ρ is the number of types of renewable resources

The optional domains can be either ignored or filled in with zero values. In the current research, **LD**, **TD** and **CD** domains were always used, and if there were renewable resources, the **RD** was also filled in, but if there was no information regarding resources, the **RD** was ignored. The applied database does not contain quality data; therefore, **QD** was omitted. The dissertation focuses only on the structure, time-related and (renewable) resource demands. A nonrenewable domain was not used, as only a minority of the databases have it. Since the real-life database counts of the task and resource costs can also be calculated from the multiplication of resource and time demands, **CD** was not ignored, but like nonrenewable resources, the cost was not in the scope of the analysis.

If the logic domain of the UMP contains either or both supplementary tasks and flexible dependencies, the minimal (maximal) makespan of the project (henceforth, the total project time, TPT) can be specified. When the supplementary tasks and all flexible dependencies are excluded from (included), projects (Kosztyán, 2015) are called *minimal (maximal) project structures*, denoted $\mathcal{S}_{\text{min}}$ ($\mathcal{S}_{\text{max}}$), see the example in Figure 12.

In the case of an early schedule, the maximal (minimal) resource use occurs when all supplementary tasks are included in (excluded from) the project while all flexible dependencies are excluded from (included in) the project structure. These structures are henceforth called *maximin (minimax) project structures* denoted $\mathcal{S}_{\text{maximin}}$ ($\mathcal{S}_{\text{minimax}}$) (see the left side of Figure 12 and Equations (2) through (5)).

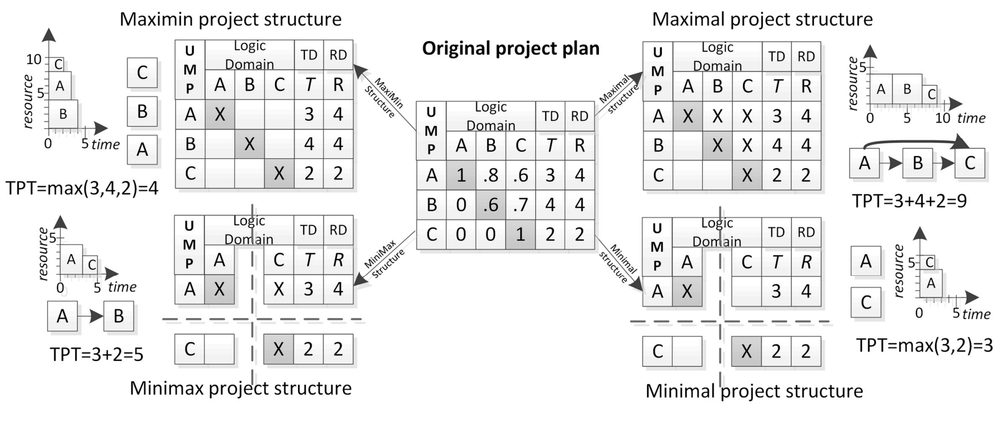


FIGURE 12: Minimal, maximal, minimax and maximin structures of the flexible project plan

To indicate that the minimal, maximal, minimax and maximin structures are the results of a decision, the mandatory tasks and fixed dependencies are represented by X, while the omitted tasks and independence are represented by empty cells.

3.2 Studied project indicators

Table 6 summarizes the applied indicators on the project plans.

TABLE 6: Applied indicators

Name	Short description	Adapted from	Besides single mode single project, applicable for			For results, see
			single mode multiproject	multi-mode single project	multi-mode multiproject	
Structural indicators						
I1	number of nodes (i.e., tasks)	Tavares, 1999; Vanhoucke, Coelho, Debels, et al., 2008	X	X	X	F15
I2	serial or parallel structure	Tavares, 1999; Vanhoucke, Coelho, Debels, et al., 2008	X	X	X	F15, F19a, F23
I3	task distribution	Tavares, 1999; Vanhoucke, Coelho, Debels, et al., 2008	X	X	X	F15
I4	rate of short arcs	Tavares, 1999; Vanhoucke, Coelho, Debels, et al., 2008	X	X	X	F15
I5	rate of long arcs	Tavares, 1999; Vanhoucke, Coelho, Debels, et al., 2008	X	X	X	F15
I6	topological float	Tavares, 1999; Vanhoucke, Coelho, Debels, et al., 2008	X	X	X	F15
T-DENSITY	total activity density	Patterson, 1976	X	X	X	F15
XDENSITY	average activity density	Patterson, 1976	X	X	X	F15
C	network complexity	Sprecher, 1994	X	X	X	F15, F19b, F23,F26
CNC	coefficient of network complexity	Davis, 1975	X	X	X	F15
OS	order strength	Mastor, 1970	X	X	X	F15
Time related indicators						
TPT	total project time	Patterson, 1976	X	X	X	F16
XDUR	average activity duration	Patterson, 1976	X	X	X	F16
VA-DUR	variance in activity duration	Patterson, 1976	X	X	X	F16
PCTSLACK	percent of activities possessing positive total slack	Patterson, 1976	X			F16
XSLACK	average total slack per activity	Patterson, 1976	X			F16
TOTSLACK-R	total slack ratio	Patterson, 1976	X			F16, F22, F25, F26
XSLACK-R	average slack ratio	Patterson, 1976	X			F16, F22, F25
PCTFREESLACK	percent of activities possessing positive free slack	Patterson, 1976	X			F16
XFREESLACK	average free slack per activity	Patterson, 1976	X			F16
Renewable resource-related indicators						
RF	resource factor (i.e., density of **RD**)	Kolisch et al., 1995	X			F17
$PCTR_j$	percent of activities that require resource type j	Patterson, 1976	X			F17
RU	resource use	Demeulemeester, Vanhoucke, et al., 2003	X			F17
$DMND_j$	the average demand resource type j	Patterson, 1976	X			F17
RC	resource constrainedness	Patterson, 1976	X			F17, F23, F25, F26
RS	resource strength	Kolisch et al., 1995	X			F17
UTIL	utilization of resources	Patterson, 1976	X			F17
$TCON_j$	constraints of resource j over time	Patterson, 1976	X			F17
$OFACT_j$	obstruction of resource j	Patterson, 1976	X			F17, F23, F25
$UFACT_j$	underutilization of resource j	Patterson, 1976	X			F17
UTIL	utilization of resources	Patterson, 1976	X			F17
NARLF′	resource distribution front/backloaded	Van Eynde and Vanhoucke, 2020	X			F29
Gini	equality distribution of resource demands	Van Eynde and Vanhoucke, 2020	X			F27, F29
Distributional indicators						
$\alpha^{dist}(...)$	variation of multiple indicator values	Labro and Vanhoucke, 2008	X			F27, F29, F29

Table 6 shows that the characterization of both the project structure and demands has several indicators. However, flexibility has no indicators, and quality and cost demands have very few indicators. None of the indicators are interval indicators. This result indicates that the result of each indicator is a scalar or, in the case of multimode completions, a vector. However, in the case of flexible projects, several possible projects have different project demands; therefore, the indicators should be specified as an interval.

Three indicator types are examined. The first group is *structural indicators*, such as complexity and flexibility indicators, which consider only the logic domain of the project domain matrices. The second group of indicators consists of *demand indicators*, which consider other domains, such as time domains (time-related indicators) and renewable resource domains (such as renewable resource-related indicators). The last group is formed by a single indicator, which uses other indicators to show their distributional properties rather than providing any characteristics on its own.

An original logic structure of a project yields an activity-on-node network, which is denoted as $G = (N, \mathcal{A})$ directed graph, where $N = \{A_1, ..., A_n\}$ (A_i is often shortened to i) is the set of nodes (i.e., tasks), and $\mathcal{A} \subset N \times N$ is the set of arcs (i.e., dependencies). $n = |N|$ is the number of tasks, and $|\mathcal{A}|$ is the number of dependencies. Furthermore, the matrix representation of the logic plan is the logic domain (LD) of the UMP matrix, where $\mathbf{LD} \in \{0,1\}^{n\times n}$, for each $i \leq n$ $[\mathbf{LD}]_{ii} = 1$, and for each $i \neq j$, we have $(A_i, A_j) \in \mathcal{A}$ if and only if $[\mathbf{LD}]_{i,j} = 1$ (otherwise $[\mathbf{LD}]_{i,j} = 0$).

Since none of the project databases considers flexible project structures, in the first step, flexible project structures are generated. Let $\mathbf{LD} \in \{0,1\}^{n\times n}$ and $\mathbf{LD}' \in [0,1]^{n\times n}$ the modified logic domain as follows:

$$l'_{ij} = [\mathbf{LD}']_{ij} := \begin{cases} u_{ij} \text{ if } l_{ij} = 1 \text{ and } v_{ij} \leq fp \\ l_{ij} \text{ otherwise} \end{cases} \tag{1}$$

where $l_{ij} = [\mathbf{LD}]_{ij}$, $u_{ij}, v_{ij} \sim U[0,1]$ are uniformly distributed random probability variables (r.v.), and $fp \in [0,1]$ is a fixed flexibility parameter set for computer runs. The goal is to have the ratio of the number of (supplementary tasks + flexible dependencies) w.r.t. the total number of LD elements is approximately fp, which is ensured by "$v_{ij} \leq fp$". The *weights* of these flexible objects are set by the r.v. u_{ij}. Note that $\mathbf{LD}'$ already contains flexible dependencies ($i \neq j$) and supplementary tasks ($i = j$). However, complexity and time-related and resource-related indicators address only fixed project structures.

The modified logic domain is used to specify only the minimal, maximal, minimax and maximin structures, as follows:

$$l_{ij}^{\min} = \left\lfloor l'_{ij} \right\rfloor, \tag{2}$$

$$l_{ij}^{\max} = \left\lceil l'_{ij} \right\rceil, \tag{3}$$

$$l_{ij}^{\text{minimax}} = \begin{cases} \left\lfloor l'_{ij} \right\rfloor & \text{if } i = j \\ \left\lceil l'_{ij} \right\rceil & \text{if } i \neq j \text{ and } \lfloor l'_{ii} \rfloor = \left\lfloor l'_{jj} \right\rfloor = 1 \\ 0 \text{ otherwise} \end{cases}, \tag{4}$$

$$l_{ij}^{\text{maximin}} = \begin{cases} \left\lceil l'_{ij} \right\rceil & \text{if } i = j \\ \left\lfloor l'_{ij} \right\rfloor & \text{if } i \neq j \end{cases}, \tag{5}$$

where $l_{ij}^{\min}, l_{ij}^{\max}, l_{ij}^{\text{minimax}}, l_{ij}^{\text{maximin}}$ are the (i,j) cells of the logic domains of the minimal, maximal, minimax and maximin structures, respectively, with $i, j = 1, 2, .., n$ (see Figure 12). [1]

Minimal, maximal, minimax and maximin structures are also included in the databases. Of course, any other possible implementation structure can be specified by rounding up or down the cell values of the logic domain. However, in the case of single completion modes and the early schedule, the minimal structure provides the minimal task duration and minimal project budget, while a maximal structure

[1] The $\lceil\cdot\rceil$ ($\lfloor\cdot\rfloor$) operators denote the rounding up (rounding down) of real numbers.

provides the highest project score (widest project scope). In addition, the minimax (maximin) structure provides the highest (lowest) renewable resource demands.

For comparability reasons, the real and artificial databases can be examined only for single projects. Individual and multiple projects can be compared by calculating the average indicators per project. However, the calculation of most indicators differs in the single and multimode cases. To avoid the confusion and additional efforts of implementing and using different definitions for the same indicator, each project instance can be analysed with fixed modes. For multimode databases, this means that the number of generated instances will be multiplied by the number of modes.

3.2.1 Structural indicators

Two structural indicator types are investigated in detail. The first group describes the rates of the flexible dependencies and supplementary tasks, and the second group describes the project structure complexity.

Structural flexibility. First, set

$$S\text{-SET} := \{l'_{ii} | l'_{ii} \sim P(0,1), 0 < l'_{ii}\} \tag{6}$$

$$F\text{-SET} := \{l'_{ij} | l'_{ij} \sim P(0,1), i \neq j, 0 < l'_{ij}\} \tag{7}$$

where $P(0,1)$ is an arbitrary continuous distribution on interval]0,1[. Then, let

fp = *flexibility parameter*, shows the total number of flexible dependencies and supplementary tasks across all tasks and dependencies as follows:

$$fp = \frac{|F\text{-SET} \cup S\text{-SET}|}{n(n+1)/2} \tag{8}$$

(The fp is set before the computer runs as the *approximate* ratio of flexible objects in Equation (1), while Equation (8) calculates the *exact* value of this ratio. Hereafter, we use this latter value of fp.)

$f\%$ = *rate of flexible dependencies* shows the sum of flexible dependencies across all dependencies as follows:

$$f\% = \frac{|F\text{-SET}|}{n(n-1)/2} \tag{9}$$

$s\%$ = *rate of supplementary tasks* shows the sum of supplementary (prioritized) tasks across all tasks as follows:

$$s\% = \frac{|S\text{-SET}|}{n} \tag{10}$$

Observe that $fp = \frac{a+b}{c+d}$ if $f\% = \frac{a}{c}$ and $s\% = \frac{b}{d}$, which has the notation $\frac{a}{c} \boxplus \frac{b}{d} = \frac{a+b}{c+d}$. For a, b, c, d positive (in this case) $\frac{a}{c} \boxplus \frac{b}{d}$ is always between $\frac{a}{c}$ and $\frac{b}{d}$, thus fp is always between $f\%$ and $s\%$, and all three depend only on Equation (1).

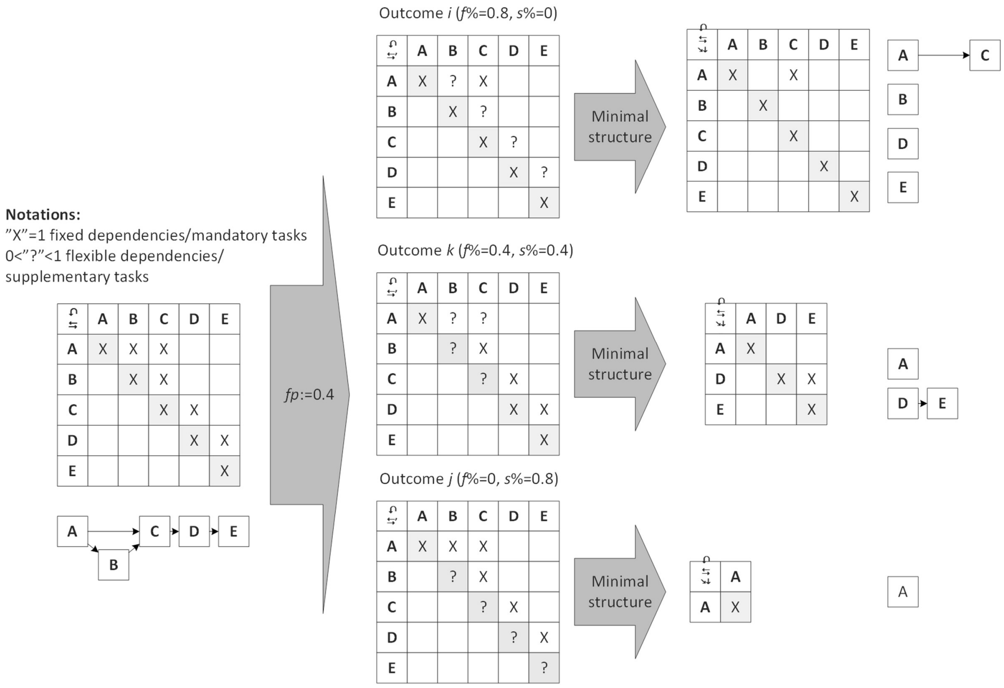


FIGURE 13: Example of generating flexibility

Figure 13 shows the mechanism of generating flexibility. The left side of the Figure 13 shows the original logic domain, where the flexibility parameter is set to be 0.4. In the first step, fixed dependencies/mandatory tasks (denoted by the "X" symbol) become flexible (denoted by "?", where "?" indicates a number between 0 to 1). The right side of Figure 13 shows the minimal structure of the project. The center of Figure 13 shows three possible outcomes from $\binom{10}{4}$. Because the number of "X" symbols is 10, we have $fp = 0.4$.

Outcome i retains all tasks, but cuts almost all dependencies, while outcome j retains only one task from the original project. In the general case, several dependencies are cut, and several tasks are omitted, e.g., in outcome k.

The following sections give exact mathematical definitions of the indicators listed in Table 6. Using these indicators, the databases can be compared and it supports the decision of researchers to find the closest databases for their case or problem to study.

Structural complexity. Denote $\mathcal{S}$ a realized project structure, $\mathbf{LD} \in \{0,1\}^{n\times n}$ of $\mathcal{S}$, $|\mathcal{A}| = \sum_{i\neq j} l_{ij}$ ($l_{ij} = [\mathbf{LD}]_{ij}$) is the total number of dependencies (arcs) between tasks.

I_1, the number of tasks (nodes), is calculated as follows:

$$I_1 := n \tag{11}$$

I_2, the serial-parallel structure, measures the closeness to a serial or parallel completion. For I_2, the following notations are needed: S_i (P_i) denotes the *set* of immediate successors (predecessors) of task i. For topologically ordered, acyclic project networks, $|S_i| = \sum_{j=i+1}^{n} l_{ij}$, $|P_i| = \sum_{j=1}^{i-1} l_{ji}$. The progressive ($PL_i$) and regressive ($RL_i$) level *numbers* of each task i can be calculated as follows:

$$PL_i := \begin{cases} 1 & \text{if } P_i = \emptyset \\ \max\limits_{j \in P_i} PL_j + 1 & \text{if } P_i \neq \emptyset \end{cases} \tag{12}$$

and

$$RL_i := \begin{cases} m & \text{if } S_i = \emptyset \\ \min\limits_{j \in S_i} RL_j - 1 & \text{if } S_i \neq \emptyset \end{cases} \tag{13}$$

where $m = \max\limits_{i} PL_i$. Next, have the following:

$$I_2 := \begin{cases} 1 & \text{if } n = 1 \\ \frac{m-1}{n-1} & \text{if } n > 1 \end{cases} \tag{14}$$

I_3, the task distribution, measures the distribution of tasks over the progressive levels by calculating the total absolute deviations.

First, define the jth progressive *level* of $j = 1, ..., m$ as follows: $\mathbf{PL}_j := \{i \leq n : PL_i = j\}$, i.e., the *set* of all tasks having progressive level number j. Then,

$$I_3 := \begin{cases} 0 & \text{if } m = 1 \text{ or } m = n \\ \frac{\alpha_w}{\alpha_{\max}} = \frac{\sum_{j=1}^{m} |w_j - \overline{w}|}{2(m-1)(\overline{w}-1)} & \text{if } 1 < m < n \end{cases} \tag{15}$$

where $w_j = |\mathbf{PL}_j|$ is the width (size) of progressive level $j = 1, ..., m$, $w = (w_1, w_2, ..., w_m)$ is the vector containing the widths of each progressive level, and $\overline{w} = n/m$, α_w is the total absolute deviation from the average width. Then, $\alpha_{\max}$ is the maximal value of α_w of a network (ranging for all possible $\mathcal{A}$); thus, [2] $\alpha_{\max} = (m-1)(\overline{w}-1) + (n - m + 1 - \overline{w}) = 2(m-1)(\overline{w}-1)$.

I_4, the ratio of *short* arcs. The length of an "arc" (called a path in graph theory) between tasks i_1 and i_2 is defined as $L(i_1, i_2) := |PL_{i_1} - PL_{i_2}|$, the difference between their progressive level numbers. Arcs of length 1 are called *short*, and $D := \sum_{j=1}^{m-1} w_j \cdot w_{j+1}$ is the *maximal* number of short arcs. n'_L denotes the number of arcs of length L for $1 \leq L \leq m-1$. Then, I_4 is calculated as follows:

[2]The maximal value of α_w is achieved (n and m are fixed, $\sum_{j=1}^{m} w_j = n$) when all levels are singletons, except for one with $n - (m-1)$ tasks; repetitive use of the inequality $|a - \overline{w}| + |b - \overline{w}| < |a - 1 - \overline{w}| + |b + 1 - \overline{w}|$ for $1 < a \leq \overline{w} \leq b < n$ proves this extrema.

$$I_4 := \begin{cases} 1 & \text{if } D = n - w_1 \\ \frac{n'_1 - n + w_1}{D - n + w_1} & \text{if } D > n - w_1 \end{cases} \tag{16}$$

I_5, the ratio of *long* arcs ($L > 1$), is calculated as follows:

$$I_5 := \begin{cases} 1 & \text{if } |\mathcal{A}| = n - w_1 \\ \frac{\left(\sum_{L=2}^{m-1} n'_L \frac{m-L-1}{m-2}\right) + n'_1 - n + w_1}{|\mathcal{A}| - n + w_1} & \text{if } |\mathcal{A}| > n - w_1 \end{cases} \tag{17}$$

I_6, the topological float, considers the differences between the regressive and progressive level numbers of task i, i.e., $|RL_i - PL_i|$, as follows:

$$I_6 := \begin{cases} 0 & \text{if } m \in \{1, n\} \\ \frac{\sum_{i=1}^{n} |RL_i - PL_i|}{(m-1)(n-m)} & \text{if } m \notin \{1, n\} \end{cases} \tag{18}$$

CNC, the coefficient of network complexity, is calculated as follows:

$$\text{CNC} = \frac{|\mathcal{A}|}{n} \tag{19}$$

OS, the order strength, is calculated as follows:

$$\text{OS} = \frac{|\mathcal{A}|}{n(n-1)/2} \tag{20}$$

C, the network complexity, is calculated as follows:

$$\text{C} = \begin{cases} \frac{\log \frac{|\mathcal{A}|}{n-1}}{\log \frac{n^2-1}{4(n-1)}} & \text{if } n \text{ is odd} \\ \\ \frac{\log \frac{|\mathcal{A}|}{n-1}}{\log \frac{n^2}{4(n-1)}} & \text{if } n \text{ is even} \end{cases} \tag{21}$$

T-DENSITY, the total activity density, is calculated as follows:

$$\text{T-DENSITY} := \sum_{i:=1}^{n} \max\{0, |P_i| - |S_i|\} \tag{22}$$

(S_i and P_i were defined immediately before I_2.)

XDENSITY, the average activity density, is calculated as follows:

$$\text{XDENSITY} := \frac{\text{T-DENSITY}}{n} \tag{23}$$

Flexibility-related structural indicators. All structural indicators depend on the realized structure ($\mathcal{S}$), i.e., on the set of the included flexible dependencies and supplementary tasks from $\mathbf{LD'} \in [0,1]^{n \times n}$. I_1 = number of tasks; therefore, $I_1(\mathcal{S}_{\min}) =$

$I_1(\mathcal{S}_{\text{minimax}}) \leq I_1(\mathcal{S}) \leq I_1(\mathcal{S}_{\max}) = I_1(\mathcal{S}_{\text{maximin}})$. Nevertheless, since the fixed dependencies between the supplementary tasks must be excluded if the supplementary tasks are excluded, the minimal (maximal) structures are the lower (upper) bounds of C. The CNC and OS indicators of these cases are those in which only mandatory tasks exist. Regarding the other structural indicators, the connection between them and the maximal-minimal structures is not obvious, and no such rules can be defined.

3.2.2 Time-related indicators

To ensure the validity of the comparison of the simulated and real-life datasets, only networks with single modes are considered. With an additional setup, multiple modes are also explored separately, however without a possible direct comparison to the real-life databases. Denote $\mathcal{S}$ a realized project structure that decides the non-mandatory tasks and dependencies *from* $\mathbf{LD}' \in [0,1]^{n\times n}$. In the following, all quantities depend on $\mathcal{S}$, but indicating $\mathcal{S}$ is omitted everywhere. For example, $\mathcal{S}$ determines $\mathbf{LD}'' \in \{0,1\}^{n''\times n''}$ from $\mathbf{LD}' \in [0,1]^{n\times n}$, However, simply denote $\mathbf{LD}''$ and n'' by $\mathbf{LD}$ and n, similarly for $\mathbf{TD}$, and $|\mathcal{A}| = \sum_{i<j} l_{ij}$ ($l_{ij} = [\mathbf{LD}]_{ij}$). Denote $t_i := [\mathbf{TD}]_{ii}$ as the duration of task i and $\overrightarrow{P} =$ "$a_1 \prec a_2 \prec \ ... \prec a_N$" a path of preceding tasks, where $a_j \prec a_{j+1}$ indicates $l_{a_j,a_{j+1}} = 1$ for $1 \leq j < N$ ($N \leq n$). $\ell(\overrightarrow{P}) := N$ is the *length* of the path, and $d(\overrightarrow{P}) := \sum_{i\in \overrightarrow{P}} t_i$ is the *duration* of path $\overrightarrow{P}$. A path $\overrightarrow{L}$ is called the *longest* or *critical* path if $d(\overrightarrow{L})$ is maximal among all paths. Next, the TPT, the total project time, is calculated as follows:

$$\text{TPT} := d(\overrightarrow{L}) \tag{24}$$

for any longest path $\overrightarrow{L}$. $\overline{\text{X}}$DUR, the average task duration, is calculated as follows:

$$\overline{\text{X}}\text{DUR} := \frac{1}{n}\sum_{i=1}^{n} t_i \tag{25}$$

VA-DUR, the variance in task duration, is calculated as follows:

$$\text{VA-DUR} := \frac{1}{n-1}\sum_{i=1}^{n} \left(t_i - \overline{\text{X}}\text{DUR}\right)^2 \tag{26}$$

PCTSLACK, the percent of tasks with positive total slack, is calculated as follows:

$$\text{PCTSLACK} := \frac{1}{n}\sum_{i=1}^{n} \begin{cases} 1 & \text{if } LS_i - ES_i > 0 \\ 0 & \text{if } LS_i - ES_i = 0 \end{cases} \tag{27}$$

where LS_i (ES_i) is the latest (earliest) start time, and $TS_i := LS_i - ES_i$ is the total slack of task i.

XSLACK, the average total slack per task, is calculated as follows:

$$\text{XSLACK} := \frac{1}{n}\sum_{i=1}^{n} TS_i \tag{28}$$

TOTSLACK-R, the total slack ratio, is calculated as follows:

$$\text{TOTSLACK-R} := \frac{\sum_{i=1}^{n} TS_i}{TPT} \tag{29}$$

XSLACK-R, the average slack ratio, is calculated as follows:

$$\text{XSLACK-R} := \frac{XSLACK}{TPT} \tag{30}$$

PCTFREESLK is the percent of tasks with positive free slack. First, the earliest finishing time of task j is $EF_j = ES_j + t_j$; then, denote $FS_i := \min_{l_{ij}=1} ES_j - EF_i$ the free slack of task i (lowest early start of successors - early finish). Here, have the following:

$$\text{PCTFREESLK} := \frac{1}{n}\sum_{i=1}^{n} \begin{cases} 1 & \text{if } FS_i > 0 \\ 0 & \text{if } FS_i = 0 \end{cases} \tag{31}$$

XFREESLK, the average free slack per task, is calculated as follows:

$$\text{XFREESLK} := \frac{1}{n}\sum_{i=1}^{n} FS_i \tag{32}$$

Flexibility impacts of the time-related indicators. Since the average task duration and variance in activity duration depend on the inclusion/exclusion of tasks but not on their dependencies (see Equations (26) and (25)), the following equations are easy to verify:

$$\overline{\text{X}}\text{DUR}(\mathcal{S}_{\text{max}}) = \overline{\text{X}}\text{DUR}(\mathcal{S}_{\text{maximin}}) \tag{33}$$

$$\overline{\text{X}}\text{DUR}(\mathcal{S}_{\text{min}}) = \overline{\text{X}}\text{DUR}(\mathcal{S}_{\text{minimax}}) \tag{34}$$

$$\text{VA-DUR}(\mathcal{S}_{\text{max}}) = \text{VA-DUR}(\mathcal{S}_{\text{maximin}}) \tag{35}$$

$$\text{VA-DUR}(\mathcal{S}_{\text{min}}) = \text{VA-DUR}(\mathcal{S}_{\text{minimax}}) \tag{36}$$

Large samples. Large samples refer to large n for which the central limit theorem (CLT) can be used. Here, some mathematical results regarding $\overline{\text{X}}\text{DUR}(\mathcal{S})$ are offered. Similar results are also used for resource indicators, such as RF, PCTR, RU, DMND, and RC in Equation (48).

$\overline{X}DUR(\mathcal{S})$ contains (finally) mandatory tasks only; thus, consider $\mathcal{S} \subseteq \mathbb{I}_n$, where denote $\mathbb{I}_n := \{1, 2, ..., n\}$, and let $s = |\mathcal{S}|$.

In the following, it is assumed that n and s are large numbers, $t_i \sim U(a,b)$ (for $i \in \mathbb{I}_n$) are uniform random variables (**r.v.**) on the *fixed* finite interval $[a,b] \subset \mathbb{R}$, and t_i are independent and identically distributed (**i.i.d.**) r.v.

STEP ONE: n and $\mathcal{S}$ are fixed. Next, $\overline{X}DUR(\mathcal{S})$ is the mean of s i.i.d. uniform r.v., and thus, the CLT yields the following:

$$\frac{\overline{X}DUR(\mathcal{S}) - \mu}{\frac{\sigma}{\sqrt{s}}} \sim \Phi(0,1) \tag{37}$$

where:

$$\mu = E\left(\overline{X}DUR(\mathcal{S})\right) = \frac{a+b}{2} \quad , \quad \sigma = D(\overline{X}DUR(\mathcal{S})) = \frac{|b-a|}{\sqrt{12}} \tag{38}$$

and $\Phi(0,1)$ is the standard normal distribution[3]

STEP TWO: n is fixed, but $\mathcal{S}$ may be any nonempty subset of $\mathbb{I}_n$, i.e., the event space is currently the power set of $\mathbb{I}_n$: $\Omega = \mathcal{P}(\mathbb{I}_n)$. Next, consider $\overline{X}DUR(\mathcal{S})$ on Ω and use the notation $\overline{\mathcal{X}}_{DUR}$ instead of $\overline{X}DUR(\mathcal{S})$. The probability of *any* $\mathcal{S}$ is $\frac{1}{n!}$, $E(\overline{\mathcal{X}}_{DUR}[\mathcal{S}]) = \mu$ and $D\left(\overline{\mathcal{X}}_{DUR}[\mathcal{S}]\right) = \frac{\sigma}{\sqrt{s}}$ when $s = |\mathcal{S}|$, which has the probability $\binom{n}{s}/2^n$; thus, having the following:

$$D\left(\overline{\mathcal{X}}_{DUR}\right) = \sqrt{\frac{1}{2^n}\sum_{s=1}^{n}\binom{n}{s}\left(\frac{\sigma}{\sqrt{s}}\right)^2} = \sigma\sqrt{\frac{1}{2^n}\sum_{s=1}^{n}\frac{\binom{n}{s}}{s}} \tag{39}$$

Finally, the following is obtained by the CLT:

$$\frac{\overline{X}DUR(\mathcal{S}) - \mu}{D\left(\overline{\mathcal{X}}_{DUR}\right)} \sim \Phi(0,1) \tag{40}$$

In the case $|\mathcal{S}|$ is limited, i.e., $c \leq |\mathcal{S}| \leq d$ is required for some fixed $c \leq d \leq n$, Equation (39) becomes the following:

$$D\left(\overline{\mathcal{X}}_{DUR}\right) = \sigma\sqrt{\frac{1}{2^n}\sum_{s=c}^{d}\frac{\binom{n}{s}}{s}} \tag{41}$$

3.2.3 Resource-related indicators

Denote $\mathcal{S}$ a realized project structure and $\mathbf{LD} \in \{0,1\}^{n\times n}$, $\mathbf{T} \in \mathbb{R}_+^n$ and $\mathbf{RD} \in \mathbb{R}_+^{n\times\rho}$ domains of the matrix representation of $\mathcal{S}$, where n is the number of tasks and $|\mathcal{A}| = \sum_{ij,i\neq j} l_{ij}$ ($l_{ij} = [\mathbf{LD}]_{ij}$). Denote $t_i = [\mathbf{TD}]_{ii}$ the duration of task i and TPT the duration of the project, and $r_{ij} = [\mathbf{RD}]_{ij}$ the resource demand of task i of resource j.

$\overrightarrow{\mathcal{S}}$ is a project *schedule* of project structure $\mathcal{S}$ if for each realized task $a_i \in \mathcal{S}$, the interval $T_i \subseteq [0, \mathrm{TPT}]$ is determined when a_i is addressed (scheduled). To ensure compatibility with other papers, the redundant notation $a_i(T) \in \overrightarrow{\mathcal{S}}$ is used.

[3]In the denominator of Equation (37), one may write $\sqrt{\text{VA-DUR}(\mathcal{S})}$ instead of $\frac{\sigma}{\sqrt{s}}$.

Denote $S(a_i(T)) \in [0, TPT - t_i]$ the start and $F(a_i(T)) \in [t_i, TPT]$ the finish time of task i. The early schedule, denoted $\overrightarrow{\mathcal{S}}_{\min}$, satisfies $\forall a_i(T) \in \overrightarrow{\mathcal{S}}_{\min}\ S(a_i(T)) = \mathrm{ES}_i$ and $F(a_i(T)) = \mathrm{EF}_i$. Denote the *resource demand* j of task i at time τ as follows:

$$r_{ij}(\tau) := \begin{cases} r_{ij} \text{ if } a_i(T) \in \overrightarrow{\mathcal{S}}, \tau \in T_i \\ 0 \text{ otherwise} \end{cases} \tag{42}$$

Furthermore, denote the total (renewable) resource *demand* of j at time τ as $r_j(\tau) = \sum_i r_{ij}(\tau)$, where $\tau \in [0, \mathrm{TPT}]$.

Nonscheduled. RF, the resource factor, is the density of **RD**, the resource matrix from a domain mapping matrix (DMM). RF gives the rate of how often resources required are from all possible resource type-activity pairings. Higher RF values indicate a more complex scheduling problem.

$$\mathrm{RF} := \frac{1}{n\rho} \sum_{i=1}^{n} \sum_{j=1}^{\rho} \begin{cases} 1 & \text{if } r_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} = \frac{1}{\rho} \sum_{j=1}^{\rho} \mathrm{PCTR}_j \tag{43}$$

where r_{ik} denotes the amount of resource type j required by task i, and PCTR_j denotes the percent of activities that require the given resource type, which gives a column-wise view of RF as follows:

$$\mathrm{PCTR}_j := \frac{1}{n} \sum_{i:=1}^{n} \begin{cases} 1 & \text{if } r_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \tag{44}$$

RU, the resource use, represents the resource use for each activity, i.e., the number of resource types used. RU varies between 0 and r (the number of resource types). It is a row-wise view of RF ($i = 1, ..., n$) as follows:

$$\mathrm{RU}_i := \sum_{j=1}^{\rho} \begin{cases} 1 & \text{if } r_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \tag{45}$$

DMND_j is the average quantity of resource j demanded when required by an activity ($j = 1, ..., \rho$) as follows:

$$\mathrm{DMND}_j := \frac{\sum_{i=1}^{n} r_{ij}}{\sum_{i=1}^{n} \begin{cases} 1 & \text{if } r_{ij} > 0 \\ 0 & \text{if } r_{ij} = 0 \end{cases}} \tag{46}$$

RC is the resource constrainedness of each resource type and is calculated as follows:

$$\mathrm{RC}_j := \frac{\mathrm{DMND}_j}{\alpha_j} \tag{47}$$

where α_j is the *availability* of renewable resource type j.

Flexibility impacts on the nonscheduled renewable resource indicators. The nonscheduled resource-related indicators are independent of the schedule. Therefore, they are independent of the rate of flexible dependencies.

All possible structures can be considered a random sample from the maximal structure if the elements of S-SET follow a uniform distribution. In this case, the following formula can be specified:

$$\frac{\mathrm{NRI}(\mathcal{S}) - Exp(\mathrm{NRI}(\mathcal{S}))}{\sqrt{Var(\mathrm{NRI}(\mathcal{S}))}} \sim \Phi(0,1) \tag{48}$$

where NRI($\mathcal{S}$) denotes any mean of the nonscheduled resource indicators, such as RF, PCTR, RU, DMND, and RC for project structure $\mathcal{S}$.

Resource-related indicators for the early schedule. The following indicators from Patterson (1976) require early scheduling ($\overrightarrow{\mathcal{S}}_{\min}$) of the activities regarding the precedence relations but not the resource constraints.

RS is the resource strength of each renewable resource type and is calculated as follows:

$$\mathrm{RS}_j := \frac{\alpha_j - r_j^{\min}}{r_j^{\max} - r_j^{\min}} \tag{49}$$

where α_j denotes the total availability of renewable resource type j, $r_j^{\min} := \max_{i=1,\ldots,n}(r_{ij})$ is the highest *individual* resource demand, and $r_j^{\max}$ denotes the peak total demand at any moment for resource type j in the precedence preserving the earliest start schedule.

UTIL_j is the utilization (rate) of resources and is measured based on the critical path length. Higher values indicate more constraints, less room for scheduling, and less possibility of changing the task starting times without increasing the TPT.

$$\mathrm{UTIL}_j := \frac{\sum_{i=1}^{n} r_{ij} t_i}{\alpha_j \cdot \mathrm{TPT}} \tag{50}$$

TCON_j is the constrainedness of (renewable) resource type j over time. In practice, it is the average utilization (UTIL_j) considering only those tasks that use that particular resource type as follows:

$$\mathrm{TCON}_j := \frac{\sum_{i=1}^{n} r_{ij} t_i}{\alpha_j \cdot \mathrm{TPT} \cdot \sum_{i=1}^{n} \begin{cases} 1 & \text{if } r_{ij} > 0 \\ 0 & \text{otherwise} \end{cases}} \tag{51}$$

OFACT_j is the obstruction factor of (renewable) resource type j and is calculated as follows:

$$\mathrm{OFACT}_j := \frac{\int_0^{\mathrm{TPT}} \max\{0; r_j(\tau) - \alpha_j\} d\tau}{\sum_{i=1}^{n} r_{ij} t_i} \tag{52}$$

UFACT_j is the underutilization factor and is calculated as follows:

$$\text{UFACT}_j := \frac{\int_0^{\text{TPT}} \max\{0; \alpha_j - r_j(\tau)\} d\tau}{\sum_{i=1}^{n} r_{ij} t_i} \tag{53}$$

ARLF (average resource loading factor) proposed by Kurtulus and Davis (1982), represents the resource distribution of projects. If resource requirements are in the first half of the project, it has a negative value, while a positive value means that resource demands are rather in the back half of the project, based on the critical path duration of each project. For multiple projects, the possible issue of averaging individual ARLF values is handled by NARLF, where normalization is done with the critical path duration of all projects (Browning and Yassine, 2010b). The formula is further improved by Van Eynde and Vanhoucke (2020) with NARLF', which, on top of this, determines if a resource demand falls in the front or the back based on the portfolio's critical path instead of each project's critical path. It uses two auxiliary variables:

$$Y_{i(\tau)} = \begin{cases} 1 & \text{if activity } a_i \text{ is active at time } \tau, \\ 0 & \text{otherwise} \end{cases} \tag{54}$$

$$Z_{i(\tau)} = \begin{cases} -1 & \text{if } \tau \leq \lceil TPT/2 \rceil, \\ 1 & \text{if } \tau > \lceil TPT/2 \rceil, \end{cases} \tag{55}$$

to get:

$$NARLF' = \frac{1}{TPT} \sum_{\tau}^{TPT} \sum_{i=1}^{n} \sum_{j=1}^{\rho} Z_{i(\tau)} Y_{i(\tau)} \left(\frac{r_{ij}}{|\{r_{ij} : r_{ij} > 0\}|} \right) \tag{56}$$

where τ is the time for the earliest start schedule, considering release dates.

Interval of the scheduled resource indicators. Since the minimax (maximin) structure requires minimal (maximal) resource demands, the following equations can be specified.

$$\text{SRI}_j(\mathcal{S}_{\text{minimax}}) \leq \text{SRI}_j(\mathcal{S}) \leq \text{SRI}_j(\mathcal{S}_{\text{maximin}}) \tag{57}$$

$$\text{SRI}(\mathcal{S}_{\text{minimax}}) \leq \text{SRI}(\mathcal{S}) \leq \text{SRI}(\mathcal{S}_{\text{maximin}}) \tag{58}$$

where SRI_j denotes the scheduled resource indicators, such as RS, UTIL, TCON, OFACT, and UFACT, of resource j, and SNI denotes the mean of a scheduled resource indicator of all resource types.

Aggregated indicators. Since the number of resource demands is very heterogeneous, the mean of the resource indicators was considered instead of calculating these values of all resources. Moreover, when the resource numbers differ across projects, the means of these indicators were used to ensure the comparability of the resource indicators. Similarly, for time related indicators, the focus is also on the means. In the following, the means are denoted without indexing.

To represent the distribution of the same indicator calculated for multiple projects, an alternative to using average values or multiproject specific formulas is the α-distance (Labro and Vanhoucke, 2008). Its value ranges from 0 (no variation) to 1 (maximal variation). For a vector of arbitrary nonnegative values $X = (x_1, \dots, x_n)$, $\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ it is calculated as: $\alpha^{dist} = \frac{\sum_{i=1}^{n}(|\overline{x}-x_i|)}{\alpha^{dist}_{max}(x_{min},x_{max})}$, where x_{max} is the upper, x_{min} is the lower bound of values of X.

$$\begin{aligned} \alpha^{dist}_{max}(x_{min}, x_{max}) =& (x_{max} - \overline{x}) \left\lfloor \frac{\sum_{i=1}^{n} x_i - n \cdot x_{min}}{x_{max} - x_{min}} \right\rfloor \\ &+ \left| x_{max} - \overline{x} + (\textstyle\sum_{i=1}^{n} x_i - n \cdot x_{max} \mod x_{min} - x_{max}) \right| \\ &+ (\overline{x} - x_{max}) \left(n - 1 - \left\lfloor \frac{\sum_{i=1}^{n} x_i - n \cdot x_{max}}{x_{max} - x_{min}} \right\rfloor \right) \end{aligned} \tag{59}$$

Gini coefficient (Gini, 1936) is used in economics to measure inequality across a population. Van Eynde and Vanhoucke (2020) used it to show how the total work content (each activity's resource demand multiplied by it's duration) is distributed amongst all activities. Value 0 means all activities have the same work content (equality), and 1 means only one activity has all the work (inequality).

$$G = \frac{1}{n}\left(n + 1 - 2\left(\frac{\sum_{i=1}^{n}(n+1-i)[WD]_i}{\sum_{i=1}^{n}[WD]_i}\right)\right) \tag{60}$$

for a population of $WD = \sum_{j=1}^{\rho}([RD]_j \cdot TD)$, $[WD]_i$ $(i = 1, ..., n)$ is a column vector, indexed in a non-decreasing order ($[WD]_i \leq [WD]_{i+1}$).

3.3 Applied multivariate analysis

In addition to the descriptive statistics, multivariate and network analyses were used to explore the relationships between the indicators. First, a correlation graph is specified between the indicators, represented by nodes, where the arcs represent the strength of the correlation between these nodes. The clustered correlation graph collects subsets of highly correlated indicators and groups them into a module by the Leiden method (Traag et al., 2019). In addition, the Force Atlas II (FA2) algorithm (Jacomy et al., 2014) arranges central indicators, which have many correlations between other variables, to the center of the module, and peripheric indicators are arranged at the edge of the correlation graph. For regression, robust statistical methods are applied such as the quantile regression (Koenker and Hallock, 2001) or the generalized least squares method (Aitken, 1936).

3.4 Proposed metaheuristic optimization framework

To schedule a multiproject, the resource-constrained multiproject scheduling problem (RCMPSP) (Pritsker et al., 1969) needs to be solved. It is a generalization of the well-known RCPSP problem (Dike, 1964) that is already proven strongly NP-hard (Lenstra and Rinnooy Kan, 1978). Due to the complex nature of these problems and a large number of activities and high resource-constrainedness, this study considers a metaheuristic optimization in line with the literature to achieve approximate best solutions without the need for high and time consuming computation efforts using the exact mathematical models. The model for this study's proposed metaheuristic optimization framework is implemented in a simple spreadsheet environment that managers are also familiar with. The simulations use the commercial optimization engine OptQuest™, which combines metaheuristic procedures including scatter search, tabu search, and neural networks (Laguna, 2011). Researchers widely use the tool, but one of the limitations is that due to commercial reasons, the details of the composite methods of the tool remain a black box (Kleijnen and Wan, 2007). For comparison with other popular software, refer to Eskandari et al., 2011 and Jafferali et al., 2005. The proposed simulation framework addresses both single and multiple projects, multiple global and local resource types, release dates, multiple execution modes for demands such as time, resource and cost, stochastic activity durations and release dates, as well as overtime costs, pre-defined and customizable objective functions and constraints. With the graphical part of the user interface it is also possible to visualize and compare the resource profiles of two separate multiproject schedules, the durations of the portfolio by subprojects and the cost profile. In this dissertation, only a subset of these features will be used and demonstrated for the case study.

3.5 Applied sensitivity analysis

A Monte-Carlo sensitivity analysis was designed to carry out simulations using the modeled spreadsheet logic within the simulation framework. Using this method the full-factorial enumeration of the parameter combinations was not necessary. The sensitivity analysis was used on the company project plan of the case study to show, understand and validate the effects of flexibility for a real project plan in a representative context. The typical output for the two relevant input parameters, the flexibility of dependencies and supplementary tasks were examined by generating random values from a simple Bernoulli distribution. The contribution of these parameters to TPT were also explored.

Chapter 4

Simulation results

4.1 Descriptive statistics and data comparison

Table 7 shows the number of projects in the 12 datasets of the 7 project databases.

TABLE 7: Descriptive statistics of the applied project databases

(a) Descriptive statistics of the single project databases

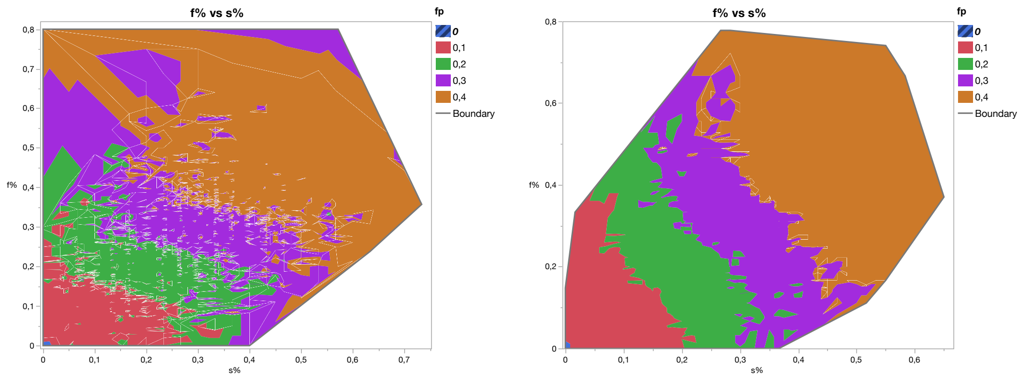
			Task number mean ($\bar{I}_1$)				
			original	minimal structures			
Database	Set	N	fp=0	0.1	0.2	0.3	0.4
Boctor	Boctor	2160	75.00	67.38	60.09	52.40	44.81
Kolisch	SMCP	1800	29.00	26.16	23.29	20.51	17.40
	SMFF	4320	30.00	26.97	23.84	21.08	17.77
MMLIB	MMLIB50	4860	50.00	45.05	40.14	35.18	29.86
	MMLIB100	4860	100.00	89.94	80.00	70.10	59.97
	MMLIB+	29160	75.00	67.50	60.05	52.54	44.85
Patterson	Patterson	990	24.02	21.73	19.51	16.85	14.91
PSPLIB	j30	5760	30.00	27.14	24.08	20.86	17.91
	j30sm	4320	30.00	27.06	24.02	21.11	17.78
Real-life	PROTRACK	1125	65.56	58.83	52.09	45.50	39.78
RG	RG30	16200	30.00	26.96	24.07	21.08	18.01
	RG300	4320	300.00	270.16	240.11	210.25	180.31

(b) Descriptive statistics of the multiproject databases

			Mean of task numbers (by projects) ($\bar{I}_1$)				
			original	minimal structures			
Database	Set	N	fp=0	0.1	0.2	0.3	0.4
BY	BY	110880	60.00	53.99	47.98	42.04	35.94
MPLIB1	Set 1	7497	360.00	324.08	287.95	251.59	216.39
	Set 2	13167	720.00	648.13	576.42	503.87	431.86
	Set 3	20286	1440.00	1296.34	1151.50	1007.89	863.79
MPLIB2	Set 1	91125	1000.00	900.19	800.08	700.10	599.91
	Set 2	77760	1000.00	900.40	800.39	700.30	600.16
	Set 3	77760	1000.00	900.01	799.77	700.09	599.89
	Set 4	69120	1000.00	899.88	800.25	700.17	600.29
MPSPLIB	MPSPLIB	1260	872.14	785.12	698.91	610.79	522.21
RCMPSPLIB	RCMPSPLIB	234	164.62	149.00	131.65	117.15	98.38

The total considered project number in a single project database was 79, 875. This value was nine times more than the original 8,875 projects. This result is due to the inclusion of both minimum and maximum structures in the database with four different flexibility parameter (fp) values. Most projects were derived from the MMLIB+ dataset (29,160) from the MMLIB database and the RG30 dataset from the RG database (16,200). The average task number within a project in the original databases was between 24 and 300 (see column $fp = 0$ in Table 7); this value decreased for minimal structures when the flexibility parameter (fp) was increased. The considered multiple project database contains 5 databases and 10 datasets. Considering demands by projects shows the same effects of increasing flexibility. Nevertheless, this database does not contain any real-life data; therefore, only simulated projects can be compared.

Figure 14 shows the relationship between the specified rate of constraints and the observed rates of the supplementary tasks and flexible dependencies.



(a) Single projects (b) Multiple projects

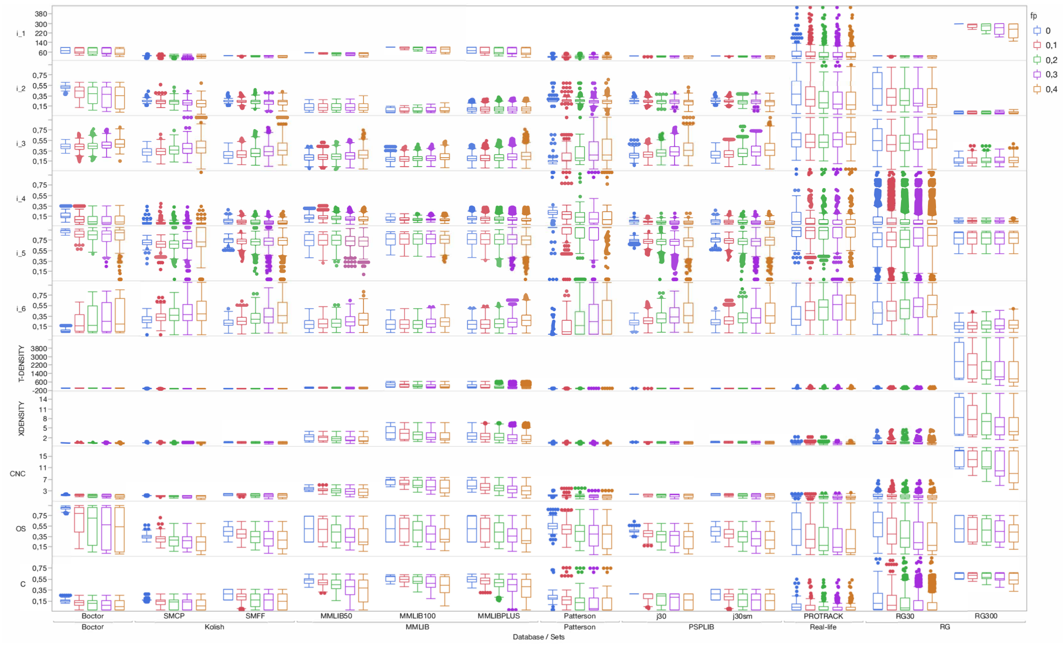
FIGURE 14: Observed rate of the supplementary tasks ($s\%$) and of the flexible dependencies ($f\%$) by the flexibility parameter rate (fp)

fp is maximized to 40% for both theoretical and practical reasons. However, the expected value of f% and s% is 40% if fp% is 40%, which is in line with the guide of the DSDM (see Figure 10), Fig 13. indicates that a further increase in the fp% above 40% might cause all tasks to be flexible and could be omitted or postponed in the minimal structure in which only mandatory tasks are completed. In addition, since we consider an iteration (sprint) as a logic plan, the number of flexible tasks may be higher than 40%. However, on average, this number should not be greater than 40%. In the case of hybrid projects, the number of flexible tasks is less than that in agile ones; therefore, fp between 0.0 to 0.4 well simulates the traditional-hybrid-agile transitions.

Figure 14 shows that the observed rates of the supplementary tasks and those of flexibility dependencies covered the most combinations of the flexibility parameters.

4.1.1 Flexibility effects on the indicators

Figure 15 compares the structural indicators in the 22 datasets with 5 different flexibility parameters.



(a) Single project database

(b) Multiple project database

FIGURE 15: Flexibility effects on the structural indicators

Figure 15 shows that the considered datasets provide various complexity values. Regarding most complexity measures, such as $I_1 - I_6$, OS, and C, the real-life database covers the greatest intervals of the structure-related and complexity related values, while regarding the CNC, T-DENSITY, and X-DENSITY indicators, the RG300 datasets cover the most possible values. Nevertheless, generally, the flexibility extends to the covered intervals of the structural indicators in all datasets. Nevertheless, the multiple project databases do not contain real-life datasets. Thus, the comparison between the simulated and real-life database can only be analyzed in a single project database.

Figure 16 compares the time-related indicators of projects from the 7 single project databases and 12 datasets.

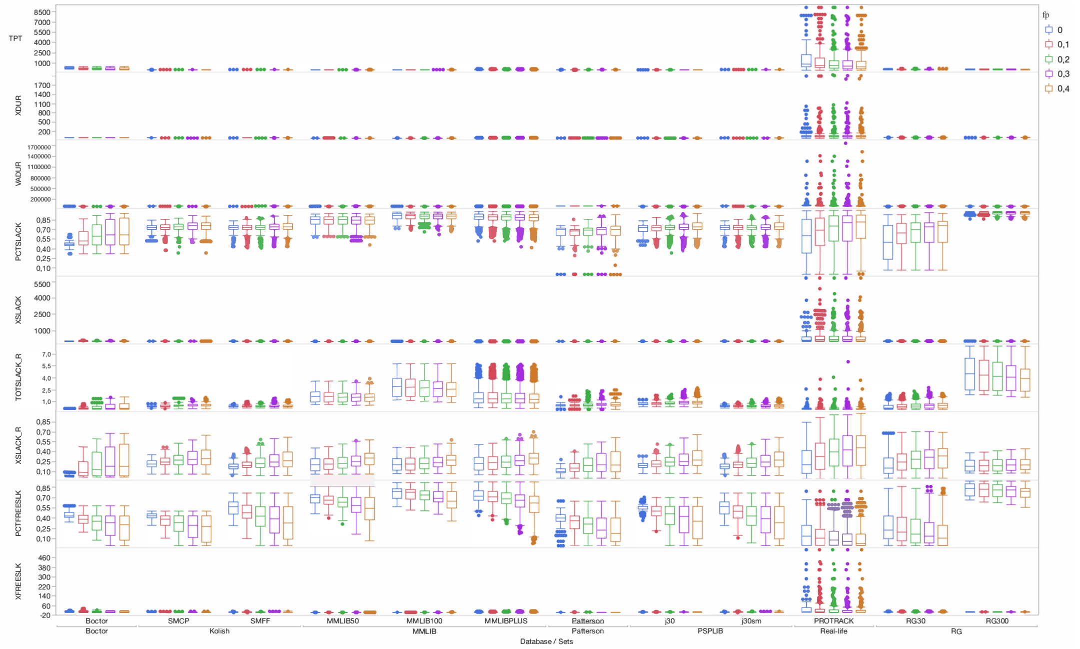


FIGURE 16: Flexibility effects on the time-related indicators

Figure 16 also shows that the real-life database and the RG300 dataset covered the most possible values of the time-related measures/indicators. Nevertheless, despite the spread of the time-related value intervals induced by considering flexibility, the real-life database covered significantly more possible values for the time-related indicators. Without considering flexibility, any single simulated database focuses on a narrow interval of time-related indicators that can be very far from real-life project values.

Figure 17 compares the project resource-related indicators from the 7 databases and their 12 datasets.

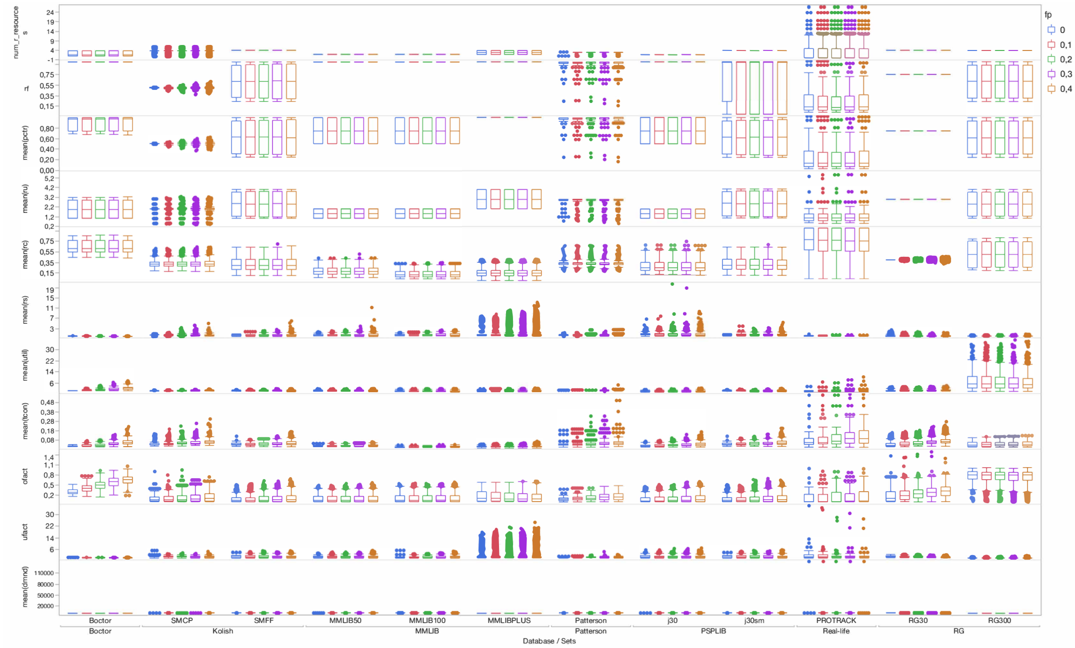
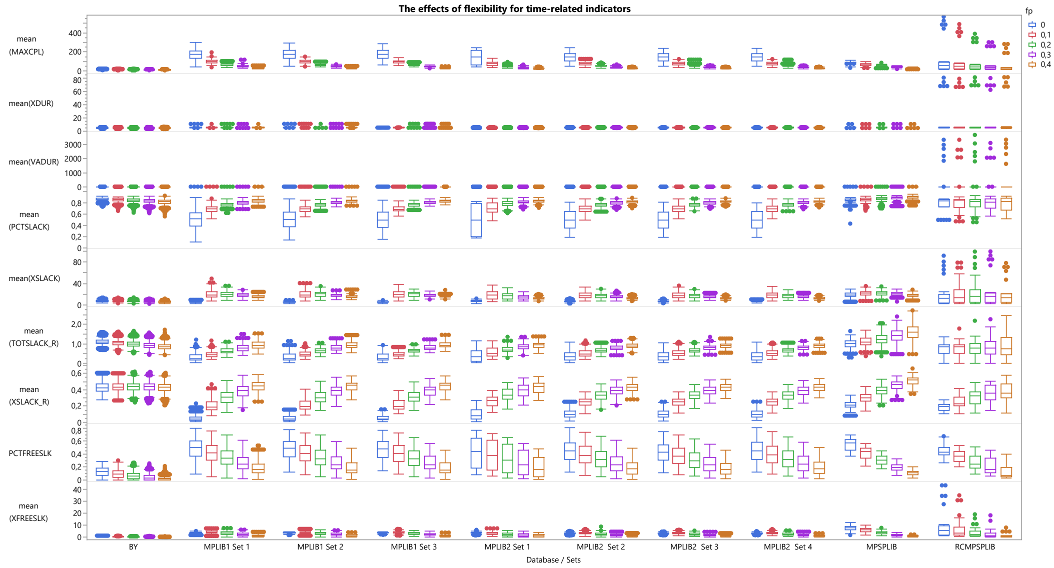
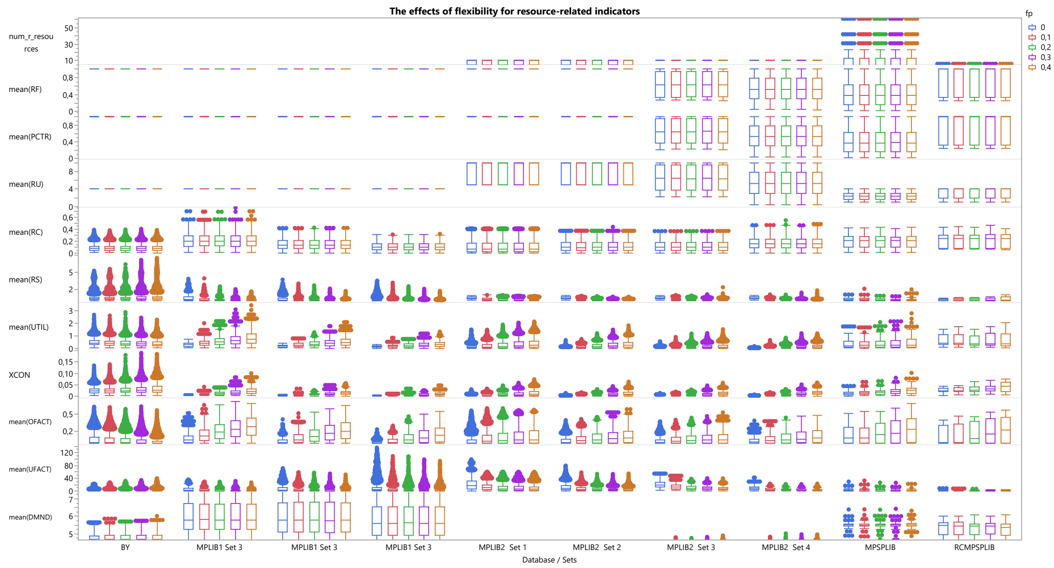


FIGURE 17: Flexibility effects on the resource-related indicators

The difference between the simulated and real-life projects based on the resource-related indicators can also be identified in Figure 17. Nevertheless, in contrast to the time-related indicators, Figure 17 shows that the MMLIB+ dataset provided resource-related indicator values, e.g., the resource strength (RS) values, that never occur in a real-life project. For example, the number of resources (num_r_resources), resource constrainedness (RC), and underutilization factor (UFACT) values varied in a wider range in the real-life database. In all cases, by introducing flexibility to the project structures and including the generated minimal structures, the interval of the possible values of the structure-related, time-related, and resource-related indicators can be widened and brought closer to the values of the real-life database. The interpretation ranges of the indicators of multiprojects are also broadened, see Figure 18.



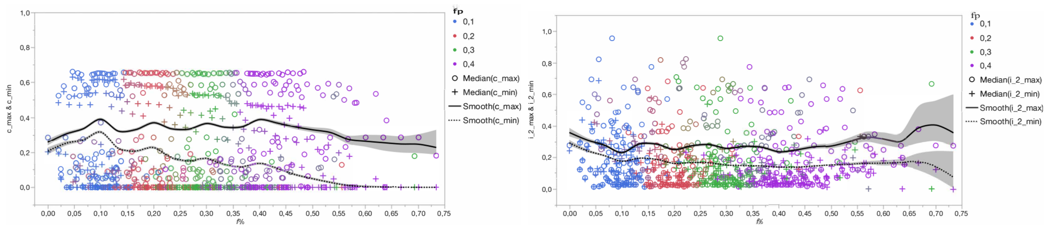
(a) The effects of flexibility for time-related indicators on multiprojects



(b) The effects of flexibility for resource-related indicators on multiprojects

FIGURE 18: Flexibility effects on the demand-related indicators among multiprojects

Figure 19 compares the complexity (C) and parallelization (I_2) values of the minimal and maximal structures regarding the ratio of flexible dependencies ($f\%$) (marked on the horizontal axis).



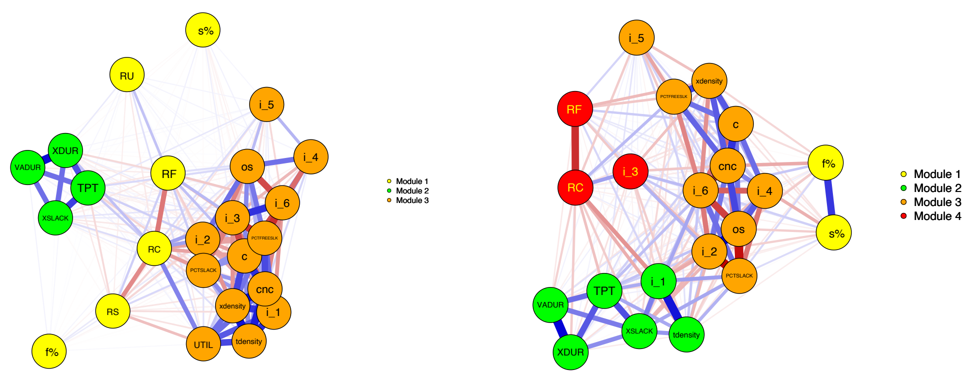
(a) Flexibility vs complexity (b) Flexibility vs parallelization

FIGURE 19: Structural changes in complexity and parallelization

Figure 19 shows that when the flexibility parameter (fp) was increased via an increase in the rate of flexibility dependencies ($f\%$) for the minimal structures, the complexity (C) decreased (see Figure 19(a)), as did the serial completions (see Figure 19(b)).

4.1.2 Flexibility effects on indicator interdependence

Figure 20 shows the clustered correlation graph between the indicators in the single-project database. Leiden's modularity specifies the modules. In the center of the modules are the indicators that correlate with most other indicators. On the periphery are the indicators correlated with relatively few other indicators, and their correlations with the remaining indicators are weak.



(a) For the simulated projects (b) For the real-life database ($RS, UTIL$ cannot be used in correlation graph)

FIGURE 20: Clustered correlation graph between the indicators[1]. Notes: The correlation strengths are proportional to the tightness of the arcs between the nodes. The blue (red) arcs indicate positive (negative) correlations.

One interpretation of Figure 20 is that several redundant indicators were highly correlated with each other. This was especially true for the topological indicators

[1]Only the significant correlations are represented. Applied grouping was accomplished using the Leiden modularity-based community detection method. The nodes are represented only where there is variance.

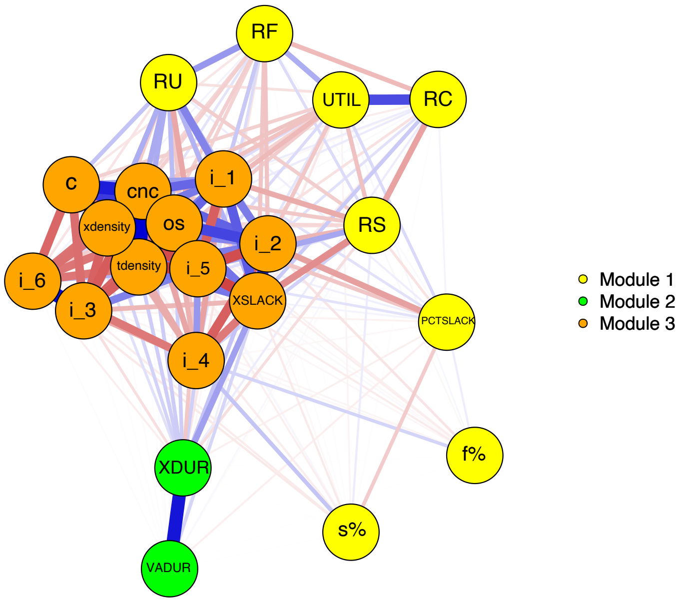


FIGURE 21: Clustered correlation graph of the multiple project database indicators. Note: indicators represent average values.

(Module 3). In comparison, the proposed ($s\%, f\%$) flexibility indicators were located on the periphery and in another module (i.e., in Module 2), which suggests that, although they are related to the other indicators, they should not be merged with them. The other finding is that the modules in the simulated datasets were quite well provided with the structure-related, time-related and resource-related indicators, where the complexity (C), resource constrainedness (RC), and project duration (TPT) played central roles. At the same time, the real-life dataset provided more mixed modules. Thus, the correlation direction did not change, four modules were specified, and at least one structural indicator was included in all the modules, which indicates the greater importance of structural indicators in real-life projects. The separation of the three modules can also be considered in the case of multiple projects (see Figure 21). When Figures 20 and 21 are compared, more significant differences can be seen between the simulated vs. real-life indicators than between the single vs. multiple project indicators. The multiple project database also produced three modules. Nevertheless, they were more mixed than were the single-project cases.

Flexibility considerations not only expand the interval of the indicator values but also specify new value pairs for the coupled indicators. Figure 22 shows the effect of including minimal structures on the complexity and time-related indicators. In all subfigures, the blue circles and plus signs represent the original pairs of indicator values. Figure 22 shows the pairs of the indicator values of the total slack ratio (TOTSLACK-R) and average slack ratio (XSLACK-R) as time-related indicators on the vertical axis and complexity (C) and parallelization (I_2) as structural parameters on the horizontal axis.

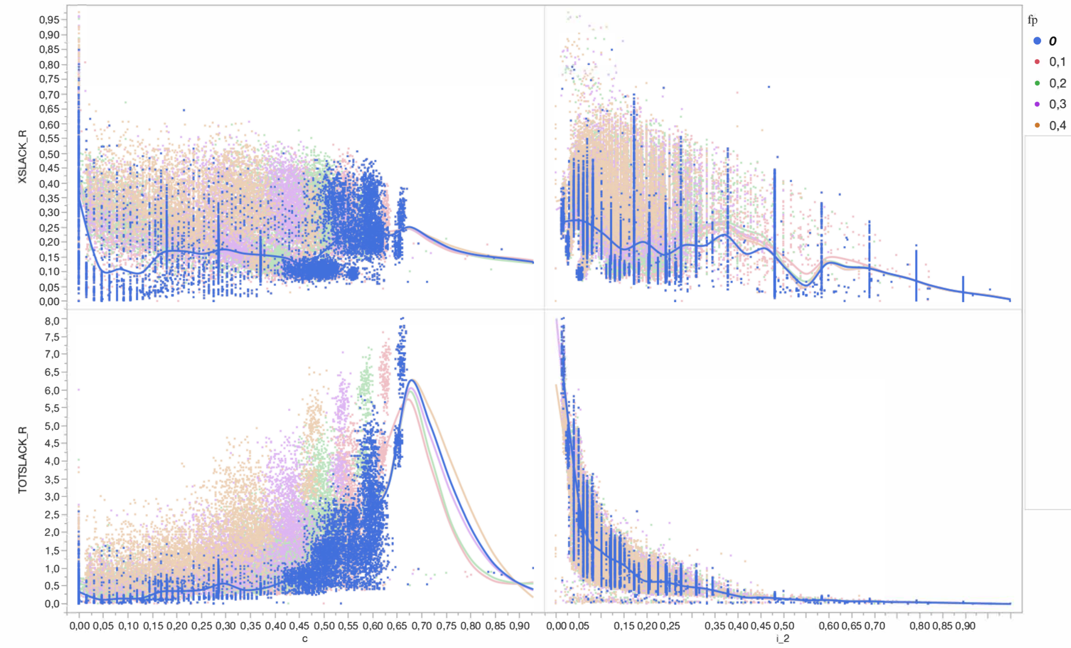


FIGURE 22: Flexibility effects on the relations between the time-related and complexity indicators

Figure 22 shows that including minimal structures helps explore new areas on the planes spanned by the structure-related and time-related indicator pairs. These combinations better cover the area of the possible value pairs. Flexibility can also be expressed in other ways as follows: the minimal structures of flexible projects have higher average slacks, which can be better utilized in resource allocation. When the minimal structures of flexible projects are included, the domain is better covered if a combination of (1) resource-related indicators, such as the mean of resource constrainedness ($\overline{\mathrm{RC}}$)/the mean of the obstruction factor ($\overline{\mathrm{OFACT}}$, and (2) a structural indicator, such as complexity (C)/parallelization (I_2), is studied (see Figure 23).

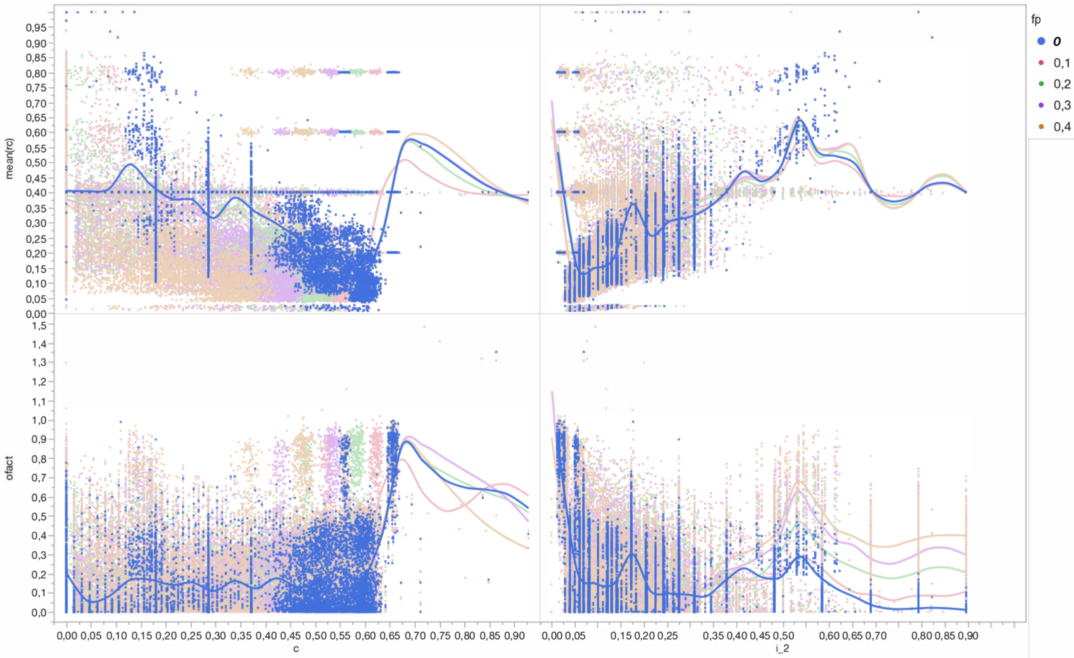


FIGURE 23: Flexibility effects on the relations between the resource-related and complexity indicators

Figure 23 shows that while minimal structures decreased the complexity (C) and increased the parallelization (i.e., decreased serialization) (I_2), they also increased the obstruction factor and the resource constrainedness.

Figure 24 shows that flexible multiprojects become more parallel and slack times increase while their overall complexity is reduced. As a result, total project time is also reduced, and resources get more constrained – the interval of indicator values for parallelity and complexity shrinks and shifts to lower values considering minimal structures.

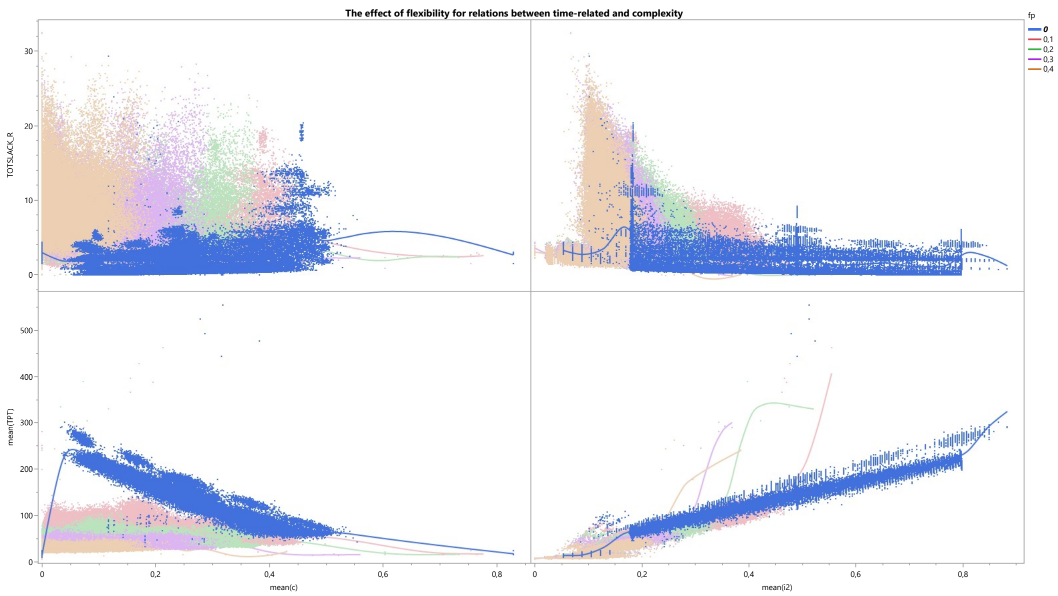


FIGURE 24: The effect of flexibility for relations between time-related and complexity indicators for multiprojects

Figure 25 shows the relations between the slack ratios (TOTSLACK-R, XSLACK-R) and the resource-related indicators in the earliest start schedule. Considering the minimal structures of flexible projects increases the slack ratio, the resource constrainedness, and the obstruction factor because of the parallelization. These combinations of time-related and resource-related indicator values occurred only in flexible project plans.

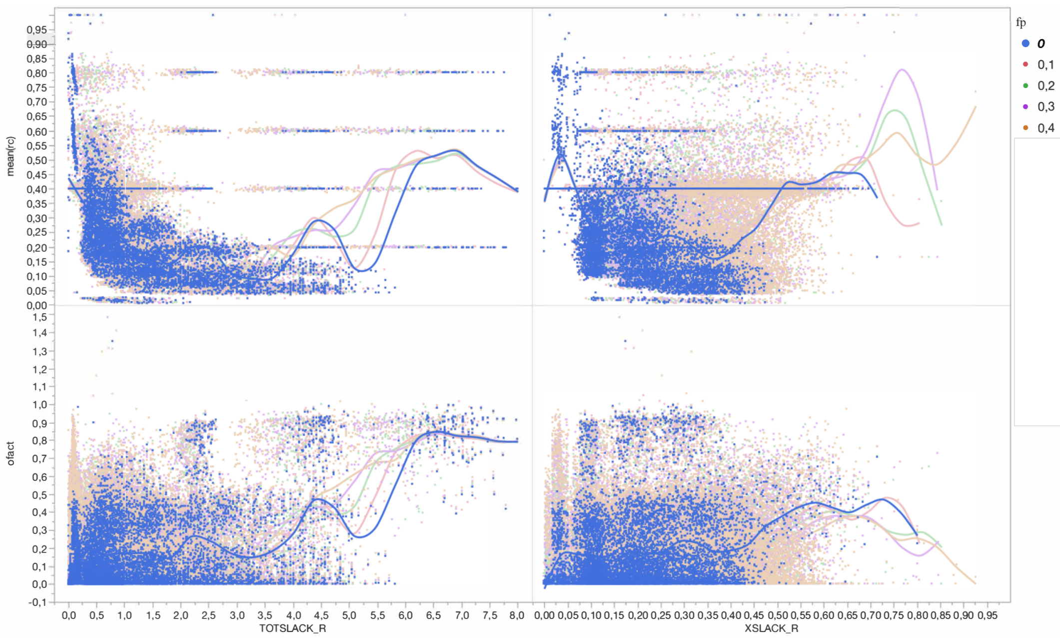
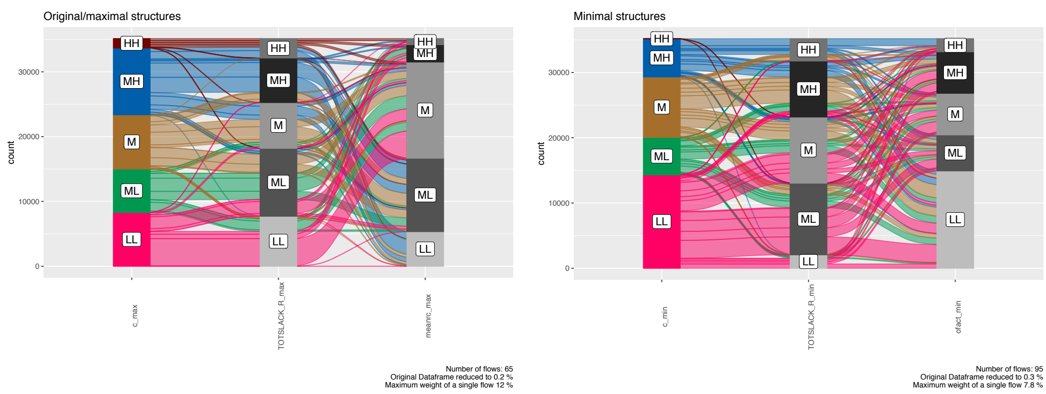


FIGURE 25: Flexibility effects on the relations between the time-related and the resource-related indicators

Figure 26 shows the mutual effect of flexibility on a structure-related (C), a time-related (TOTSLACK-R), and a resource-related (RC) indicator. Flexibility can reduce complexity (compare Figures 26(a) and (b)) while it increases the slack ratios and reduces the resource constrainedness (RC).



(a) Original/maximal structures (b) Minimal structures

FIGURE 26: Alluvial diagrams of the complexity, time-related, and resource-related indicators

Figure 27 depicts structural and resource-related indicators and gives insight into their variation on multiple levels: tasks, connected components, and projects. Variation in parallelity described by $\alpha^{dist}(I_2)$ gets closer to one (less variation) when flexibility is present and varies more when only a few projects exist.

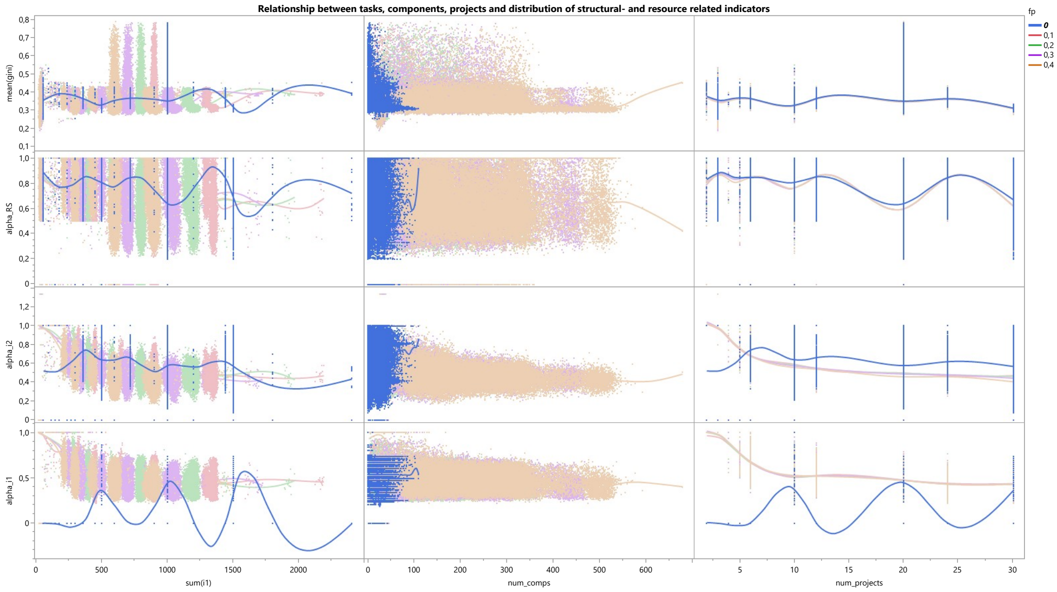


FIGURE 27: The effect of flexibility for distributions of structural and resource-related indicators on multiple levels

With increasing flexibility for minimal structures, the values of the Gini index become smaller as the work demand for resources gets more equally distributed (closer to zero) amongst projects. However, the interval of Gini indices also widens, which means a higher potential inequality in some cases. $\alpha^{dist}(RS)$ shows less variation (closer to zero) in the resource demand and availability relation when flexibility is higher. $\alpha^{dist}(I_1)$ shows that the variation in the number of tasks decreases with more projects present.

4.1.3 Flexibility effects on multiple modes

Evaluation for multiple modes is only possible for artificial single projects. MMLIB+ has the highest number of modes (9) and the amount normally varies from 3 (MMLIB50 and MMLIB100, J30mm) to 4 execution modes (as in Boctor). The modes usually have different meanings regarding resource- or time demands. For example, work content distribution gets more inequal (*Gini* indicator) for modes with a higher index in the Boctor dataset, and the opposite is true for the other group of datasets examined. Boctor has a unique meaning for mode numbers and cannot be easily ordered by meaning. For other datasets, the resource constrainedness (RC) gets higher with lower mode indices, and TPT decreases as a trade-off, shown in Figure 28. Similarly, a lower mode index means lower task durations and decreased slacks.

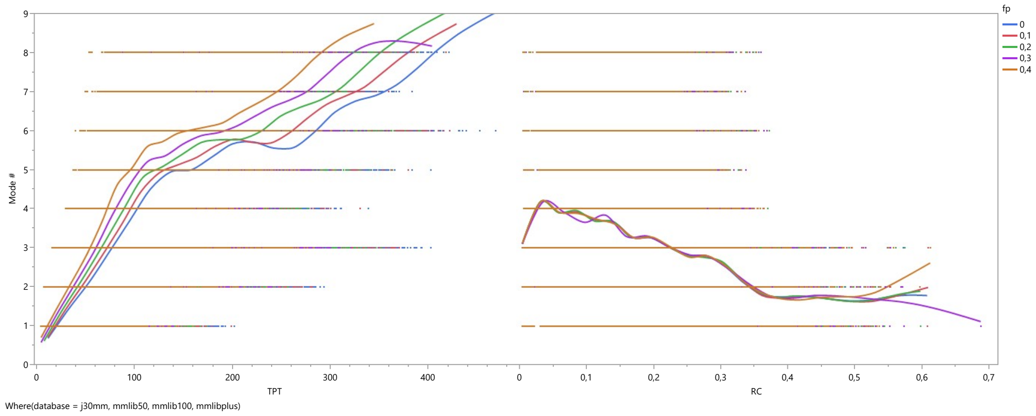


FIGURE 28: Trade-off between time- and resource demands of the different execution modes

It is visible in Figure 41 that $NARLF'$ is increasing in general and resources get from front-loaded to more evenly distributed along the project with higher modes. All related Figures can be found in Appendix B.

4.1.4 Flexibility effects on resource load on different levels

The effects of flexibility can also be visualized on the different topological levels such as tasks, sprints (represented by sets of activities) and projects. Figure 29 shows the decreasing TPT and also a decreasing number of (supplementary) activities for the multiproject databases. At the same time, the sets of activities in the network that are linked to each other, i.e., the number of connected components (Thulasiraman and Swamy, 2011) increases as a result of flexible dependencies being removed. The same decrease can also be observed by looking at project level.

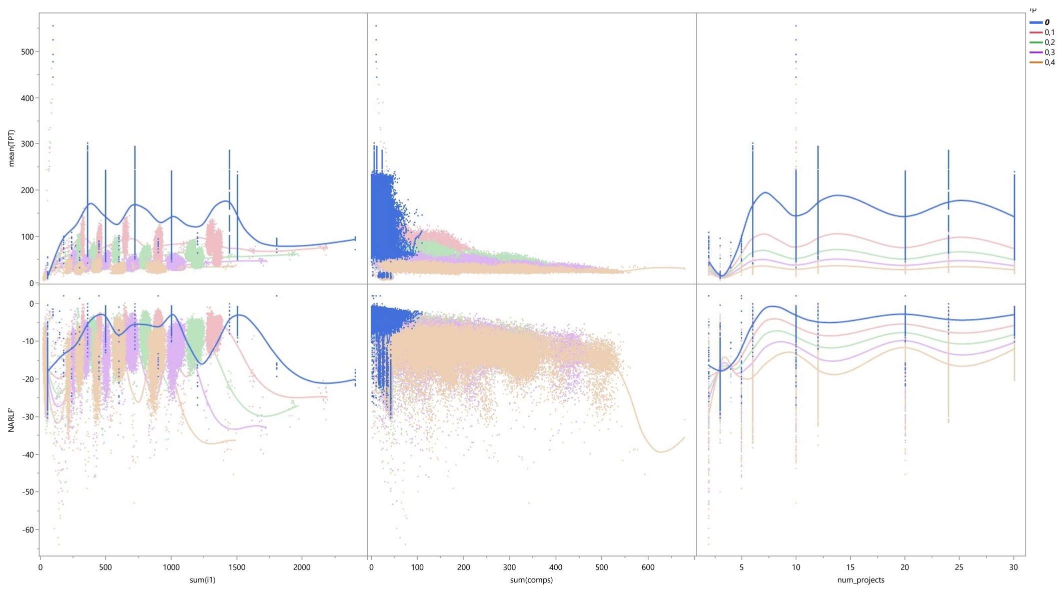


FIGURE 29: The effect of flexibility for resource loading ($NARLF'$) and total project time (TPT) on multiple topological levels

4.1.5 Flexibility effects on total project time

A regression model was built to explore further the relationship between flexibility and project duration besides existing descriptive statistics.

As a first step, the assumptions of linear regression had to be verified. The sample size is adequate (Table 7). Linearity can be assumed based on the horizontal character of the residuals vs. the fitted plot. However, normality cannot be assumed based on the histogram (Figure 43), the corresponding statistical tests (e.g., Kolmogorov-Smirnov), or by the Q-Q plot (Figure 44). Outliers are visible on the boxplot, and the residuals vs. leverage plot in Figure 44, that shows some influential observations that can affect the regression line. The homogeneity of variances assumption is violated as indicated by the scale-location and spread-level plots on Figure 44-45), and by specific tests (e.g., Levene's test, backed up with a nonparametric Fligner-Killeen's test). This implies that a robust method is necessary. To address the non-normality, a nonparametric test will also be used.

For single projects (excluding real-life database), using Spearman's rank correlation method, the relationship between fp and TPT is negative, moderate in strength ($\rho = -0.405$) and statistically significant ($p - value < .001$). For multiprojects, the same method indicates that the relationship between fp and TPT is negative, moderate in strength ($\rho = -0.573$), and statistically significant ($p - value < .001$).

Using quantile regression ($\tau = 0.5$) the pseudo-R-squared proposed by Koenker (2005) for single projects is $R1 = 0.09$, so the flexibility parameter fp explains approximately 9% of TPT, which is considered a weak effect size ($p - value < .001$). However, for multiprojects, the pseudo-R-squared is $R1 = 0.465$, which can be considered a moderate effect size in the field of this study ($p - value < .001$).

Interpreting the quantile regression results of Table 16, the fp coefficient estimate of -153.83 means that the median (0.5^{th}) quantile of TPT decreases by 153.833 for every one unit increase in fp (independent variable).

To account for the violation of homogeneity of variances, a robust GLS (generalized least squares) regression method was also applied that takes residual weight into account. As fp can take zero value, a $\log(0)$ inverse hyperbolic sine (IHS) transformation was used to avoid issues with the infamous $\log(x + 1)$ transformation practices (see, e.g., MaCurdy and Pencavel, 1986).

The resulting adjusted $R^2 = 0.396$, with details are shown in Table 17. In addition, Figure 30 shows the linear model only for reference (assumptions violated), together with the visually similar quantile and GLS regression line. The model is statistically significant ($p - value < .001$).

All data, additional tests, figures, calculation details, and results can be found in the supplementary electronic materials listed in Appendix G.

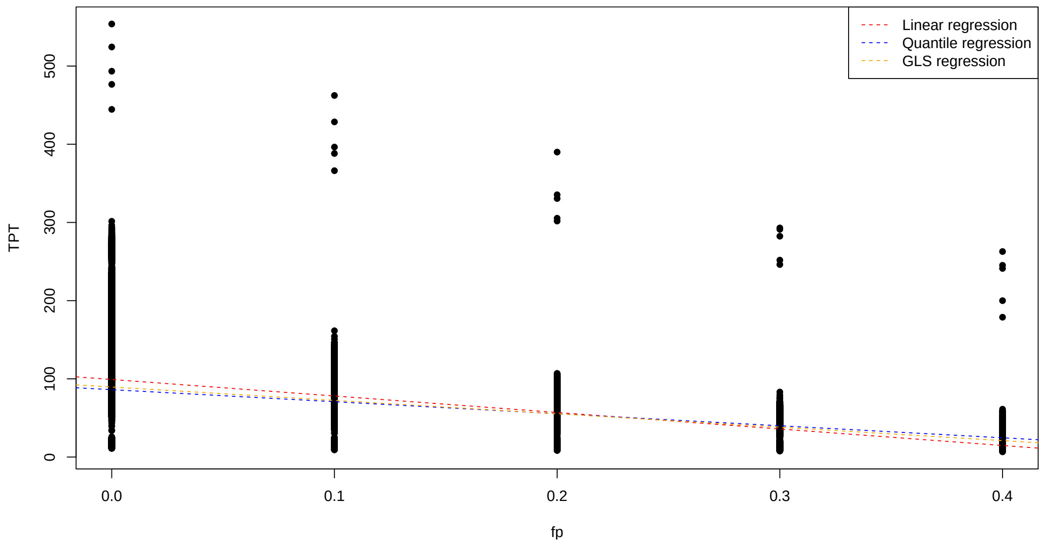


FIGURE 30: Multiproject regressions: effect of flexibility parameter on TPT

4.1.6 Flexibility effects on variation in total project time

The statistical measure of variance was used to calculate and compare how flexible methods affect total project times (TPT) compared to traditional methods in the available groups of databases. The database groups were the artificial single projects, real single projects, and artificial multiprojects. First, the variances were calculated for each flexible case (represented by $fp = \{0.1, 0.2, 0.3, 0.4\}$) relative to the traditional methods (represented by $fp = 0$) within each database group. To make these results comparable between the different groups of databases, the coefficient of variation (CV), also known as relative standard deviation shows the extent of variability in relation to the mean of the population. The higher the CV, the greater the dispersion. Table 8 shows the actual values of each descriptive statistic, including the measure CD, that is, the coefficient of dispersion (also known as variance-to-mean ratio) additionally.

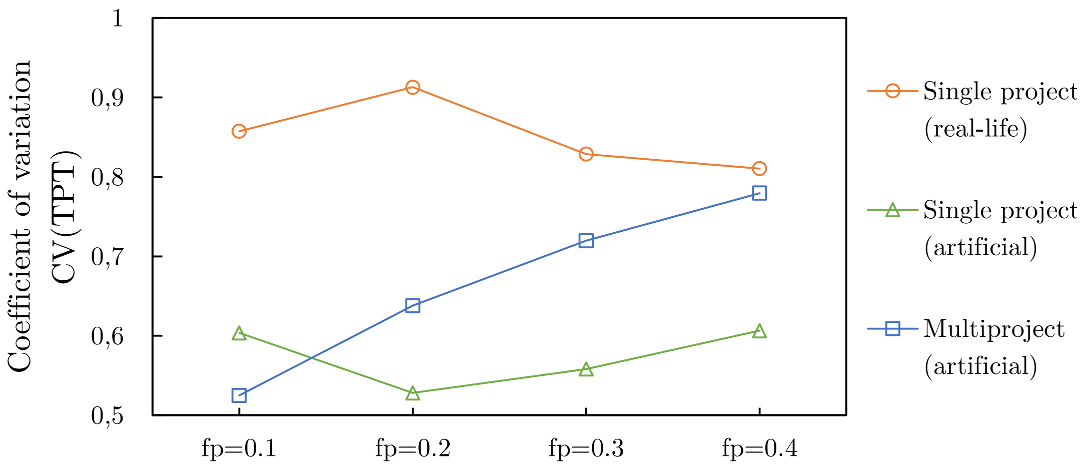


FIGURE 31: The coefficient of variations of average total project time by flexibility parameters of different database groups

As Figure 31 shows, artificial multiproject databases have the lowest initial variation for average total project time and it is similar to artificial single projects. Real-life single projects have a relatively high initial variation compared to artificial single and multiprojects. While the variation of multiprojects average total project time is continuously increasing, the shape of single projects (including both artificial and real-life) seems inconclusive.

TABLE 8: Data of TPT related descriptive statistics

	Single project artificial				Single project real				Multiproject artificial			
fp	0.1	0.2	0.3	0.4	0.1	0.2	0.3	0.4	0.1	0.2	0.3	0.4
StDev	40.06	35.04	37.04	40.24	1417.33	1508.98	1369.91	1339.48	61.08	74.18	83.73	90.68
Var	1604.64	1227.64	1372.05	1619.24	2008816.47	2277015.01	1876663.52	1794203.99	3730.63	5502.92	7010.83	8222.70
CV	0.60	0.53	0.56	0.61	0.86	0.91	0.83	0.81	0.53	0.64	0.72	0.78
CD	24.18	18.50	20.68	24.40	1215.30	1377.56	1135.35	1085.47	32.07	47.30	60.27	70.69

Chapter 5

Discussion

5.1 Evaluation of the project library comparison

Two error types can be made when testing project scheduling and resource allocation algorithms only in simulated databases. The first problem is whether new algorithms are applied to real-life projects that have different types of complexity (see Figure 15, time-related (see Figure 16) or resource-related (see Figure 17) indicator values than simulated projects in (benchmark) databases. Even if scheduling simulated projects is more difficult for the current objectives and algorithms, these algorithms may not be prepared for the challenges of the new objectives often found in real-life projects. Creating a specified database tailored to one type of problem can cause discrepancies in real-life usage because of indirect constraints rooted in unconsidered properties. Second, if the algorithms are optimized to properties of simulated projects that never appear in real life, resources are wasted. An interesting result is that the differences in the indicator values are much larger between simulated and real-life projects than they are between individual and multiple projects (compare Figures 15-17 and Figures 18(a-b)). The relationship between the indicators illustrated by the clustered correlation graph (see Figures 20 and 21) also shows significantly different results, mainly between the simulated and traditional projects. In the current research, it was not possible to include a real-life multiple project database as there is no such library available at the time of this dissertation, however, it is essential to examine real-life projects. The simulated datasets should also be combined because an individual dataset usually covers only a small range of an indicator (see Figures 15-17).

Figures 15-17, 18 also show that including minimal structures (see Figure 12) widened the indicator intervals; therefore, even if flexible structures are not studied, the extended dataset may cover larger indicator intervals.

Table 9 compares the simulated and real-life databases. The indicators from the two groups, i.e., (1) a real-life database and (2) simulated datasets, were compared by an ANOVA. Table 9 shows the number of indicators with significantly different values between these groups.

TABLE 9: Number of significantly different indicators between the simulated and the real-life databases ($p - value < 0.001$)

Indicators	$fp = 0$	$fp = 0.1$	$fp = 0.2$	$fp = 0.3$	$fp = 0.4$	All
Structural	11/13	11/13	10/13	10/13	11/13	11/13
Time-related	9/9	9/9	9/9	9/9	9/9	9/9
Resource-related	9/11	8/11	8/11	8/11	8/11	8/11

When flexibility and generating minimal structures are considered, the indicator interval can be widened; therefore, this operation should be covered in the testing of project scheduling or a resource allocation algorithm to widen the scope of the application of that algorithm. Nevertheless, considering minimal structures does not solve the problem that most complexity, time-related and resource-related measures remain significantly different between the real-life and simulated databases.

Figure 19 shows that increased flexibility reduces complexity and increases parallelization (decreases the task sequence length). These results are in line with the requirements of flexible project management approaches for reducing project complexity (Williams, 2010). However, Figure 20(a) shows that structural flexibility correlates with the resource-related indicators, especially in the simulated databases where resource constraints are prespecified. In real-life projects, structural flexibility forms a separate module. In contrast to the simulated projects, the structural flexibility indicators mainly correlated with the other structural and topological indicators; because of the lack of resource constraints, indicators RS and UTIL could not be calculated.

5.1.1 Flexibility effects on demands

Considering flexibility widens the indicator intervals and specifies new demand combinations. Figures 22-25 indicate that including minimal structures of flexible projects covered more of the domain. The new combination of indicators specified new structures that have never been tested by project scheduling and resource allocation algorithms. However, the fact that flexible projects are becoming increasingly popular implies that tasks must be prioritized and technological dependencies must be rethought. Projects supporting multiple modes can exploit a higher degree of flexibility due to alternative decisions on how the different trade-offs between time, resource or cost are varied. Indicator values of multiprojects are more sensitive to the changes in structural flexibility, especially for duration. The results shown in Figure 31 are in line with literature (Dybå and Dingsøyr, 2008), suggesting that agile multiprojects become less predictable in terms of duration (and planning) from the perspective of other projects, or the organization itself, when flexibility is increased. Minimal structures have the advantage of eliminating the need to use new algorithms. Existing algorithms can be tested in new structures generated by the FSG. Nevertheless, maintaining flexibility values, flexible project planning and scheduling algorithms can also be tested in a large set of project databases.

Chapter 6

Validation and verification

6.1 Case study - Continental Automotive

Due to the limited amount of theoretical and empirical studies related to flexible projects, a single case qualitative case study methodology (Yin, 2009) was chosen to support the quantitative research.

The case study took place at Continental Automotive Ltd. research & development center, in Hungary. The location has (among others) many projects for electronic brake systems, including software development. The case was selected because the organizational structure, type, and applied methodology of the location's projects are highly relevant to the dissertation's topic, and the access to empirical data and professionals provided a good basis for research.

6.1.1 Company overview

Continental is a multinational automotive supplier company founded in 1871, specializing in brake systems, interior electronics, automotive safety, powertrain and chassis components, tires, and many other parts for automotive. Figure 32 shows the actual structure and key data of the organization. Continental is present today in 58 countries with a generated sales of €33.8 billion and employs around 190.000 people. The scope of this case is electronic brake systems, where Continental is a first-tier supplier with typical competitors like Bosch and ZF Group.

6.1.2 Context of projects

The organization has a matrix-like structure (a mixed organizational form with vertical hierarchy overlaid by lateral authority, influence or communication (Knight, 1976) with distributed locations having several projects that form a multiproject setting. The traditional V-model (Forsberg and Mooz, 1991) is used for safety-critical embedded software development which is mixed with agile methods, such as a slightly modified version of Scrum sprint. The company is actively seeking ways to adopt agile practices for its software development activities.

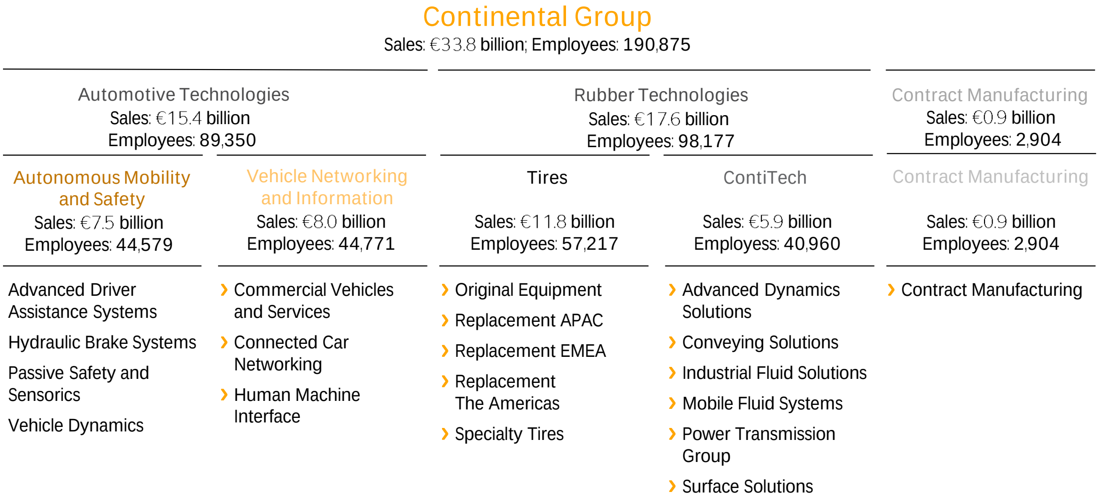


FIGURE 32: Overview of Continental group
Source: *Continental Annual Report* (2021)

6.1.3 Data collection and analysis

Data was collected with access to primary data (experts on different levels) and secondary data (project plans, manuals or other work products, intranet, ticketing, and version control system).

An empirical multiproject plan was selected to realistically represent the set of simultaneously running projects managed by the company. Key attributes of the projects were also collected, examined, and described in this chapter. Some sensitive project data, such as time and resource (cost) demands or constraints were intentionally transformed to a different unit[1].

Even though there are more than 30 projects executed simultaneously, many of these projects reached high maturity status or are already under series production. Because of this, the case study examines the 5 recently started and active software development projects that are consisting of a total number of 150 activities. It was confirmed that normally there is no logical relationship (i.e., precedence relations) between the individual projects or their activities in the multiproject.

Software projects in the company typically have four main development phases, starting from prototype development (1), to pre-production (2), pre-series production (3), and finally, series (or mass) production (4). In phase (1), basic functions are implemented and put under initial tests. In phase (2), all requested features' logic is realized with basic performance. In phase (3), additional features are tuned for performance for end customers. With phase (4), the parameters are finalized for mass production.

The so-called *features* are requested by customers usually at the beginning of the

[1] A common practice in the literature to keep sensitive information hidden without sacrificing meaningful results.

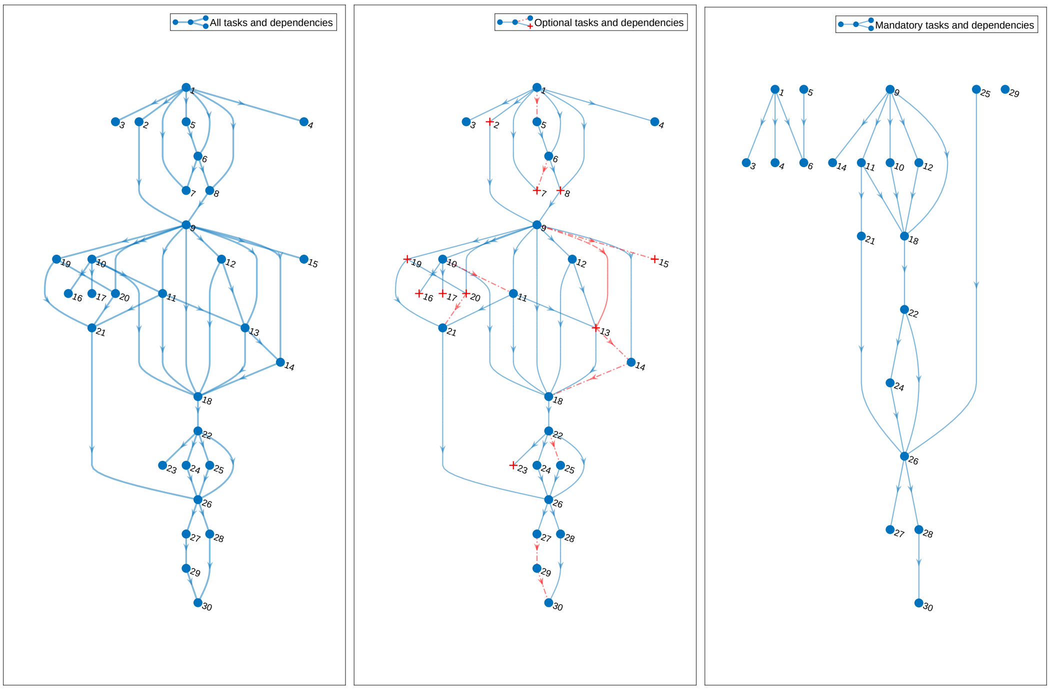
project. Traditional project plans contain all these features, assigned to specific milestones and the plans are established in advance. However, every project has a certain amount of activities that might not need to be realized, some features can change later, e.g., depending on market needs or other sources of constraints. The supplementary tasks (with flexible dependencies) within the sprints can be reordered or postponed, where the realization of features (or part of features) is done. An example of this is testing activities. For an early state, some of the tests executed can be postponed without severe consequences on quality or technical debt, so the priority of doing thorough testing initially is lower. However, with increasing maturity levels, testing gets intense focus and becomes even mandatory when approaching the final software release. Different from testing, chances of major architectural changes get less likely when software reaches a high maturity. If allowed, dependencies between activities can also be dismissed. Any of these cases can be due to internal or customer decisions, giving management flexibility. These flexible tasks and dependencies were introduced to a new matrix-based flexible project plan. The whole network logic is shown in Figure 34 in Appendix C.

	Project milestone	Activity group	Task	…	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	…
…	…	…	…	…											1							
43	Develop series production SW	Requirements, change requests	Analysis, decision and proposal		1	1	1	1	0.2	0.2	1	1	1	1								
44		Design	Architecture			0.6									1							
45		Implementation	Extended function #1				1	0.1			0.1											
46			Extended function #2					1	0.1													
47			Extended function #3						0.6	0.1												
48			Additional functions							0.8	0.2											
49			Customer function #1								1	0.5	0.3	0.3								
50			Customer function #2									0.5		0.3								
51			Logic performance										1	0.3				1				
52			Parameter performance											1	0.5							
53			Freeze development												1	1					1	
54		Test	Unit tests													0.8	0.8		1			
55			Integration test														0.9	0.9	1			
56			Robustness improvements, bugfixes															1	1			
57			Regression test																0.8	0.5		
58			System and vehicle test																	1	1	
59		Release for series production	System release																		1	1
…	…	…	…																			…

FIGURE 33: Example for a detailed software release
Source: company data

As a detailed example, the milestone *"Develop series production SW"* is shown in Figure 33. In this excerpt, the supplementary tasks and dependencies have a priority lower than 1 (<100%). Considering that the *"Release for series production"* is close, modifying the software architecture has less priority and chance (60%). However, if needed, it strictly depends on the preceding requirements analysis task (100%). Another example is the *"Customer function #2"* task, which has an equal (50%) chance to be realized, skipped or delayed further. The reason could be that it might not be requested, or a previously developed feature can be carried over instead. If it is needed, it has a (50%) priority to make it dependent on the previously developed *"Customer function #1"* task. Regarding testing, a final vehicle test is a must before release, but regression tests are needed in most of the cases (80%).

Figure 34(a) visualizes the structure of a complete single software development project plan. In Figure 34(b) the supplementary tasks and dependencies that can be omitted are highlighted, however, additional dependencies might also be deleted by



(a) All tasks and dependencies (b) Supplementary tasks and dependencies in red (c) Mandatory tasks and dependencies only

FIGURE 34: Flexible tasks within the logic network

removing supplementary tasks. Figure 34(c) shows the minimal structure when all non-mandatory tasks and dependencies are removed from the network, resulting in the minimal structure.

Table 10 shows the total project time of the previously specified structures applied for the original project plan. As expected, the minimal structure yields the shortest time necessary to finish the project, followed by the maximin structure with a duration of $TPT_{maximin} = 157$, by removing only the flexible dependencies. Minimax becomes slightly shorter by omitting only supplementary activities and their dependencies. Maximal structure demands the execution to be the most serial and thus maximizes the duration, while maximin structure does the opposite by removing all flexible dependencies and making the project more parallel while still keeping all supplementary tasks.

TABLE 10: TPT of flexible single project structures scheduled for earliest start time

Structure	TPT
Minimal	115
Minimax	147
Maximin	157
Maximal	213

Table 11 shows the rates of supplementary to mandatory tasks and dependencies in the empirical project plan. The values indicate that the company has relatively high flexibility to exploit within a project. Several dependencies can be omitted, as some functionalities are not interconnected and can be developed in parallel without technological dependency. It can also be observed from Figure 33, that supplementary tasks are typically located in the development part of the project plan, where decisions are made for the actual features to implement. Near the end of the project, rather the dependencies change. For example, the performance, quality goals or other preconditions may be already reached before the final verification, so the depending activity can be already started based on a tailored verification and not need to wait for the final results.

TABLE 11: Flexibility rates for the SW development project

	Supplementary	Mandatory	Rate
Tasks	10	30	33.33%
Dependencies	11	49	22.45%

Figure 35 shows indicator values for the case study project for maximal, minimax, maximin and minimal project structures. The shown indicators were examined previously for the different databases in detail. This evaluation aims to validate some of the assumptions and observations reported in Chapter 4. Looking at the values of TPT in 10, it is visible that the maximin structure has a higher TPT compared to the minimax structure. This suggests that in this case, the supplementary tasks have a higher impact on the project duration than the flexible dependencies, which can be explained by looking at the differences in ratios in Table 11 again. The structural complexity (C) decreases as parallelization increases (I_2). Complexity becomes the lowest with minimal structure. $DMND$ is equal for minimax and minimal structures, and maximal and maximin structures, since this indicator does not take scheduling into account and only depends on tasks and resource values based on tasks. Interestingly, the obstruction factor ($OFACT$) for resources is close in the maximin and minimax structures, because scheduling is considered in this case. Looking at $NARLF'$ it becomes visible that resource demands get more and more front-loaded with increasing flexibility of maximal to minimal structures. As the value is close to zero in the maximal (original) structure, it can be concluded that company plan has an initially good balance of resources that can be pulled ahead in the first half of the project by exploiting flexibility. Regarding slack times, the initially low slack time ratios (maximal structure) can be increased either with minimal or maximin structures. While minimal structure results in the highest average slack ratio ($XSLACK\_R$), maximin provides the highest total slack ratio ($TOTSLACK\_R$).

Sensitivity analysis of different project plans Using sensitivity analysis, not only the range, but the distribution of the total project times (TPT) can be analysed. It is also possible to measure the contribution of the flexible tasks and dependencies

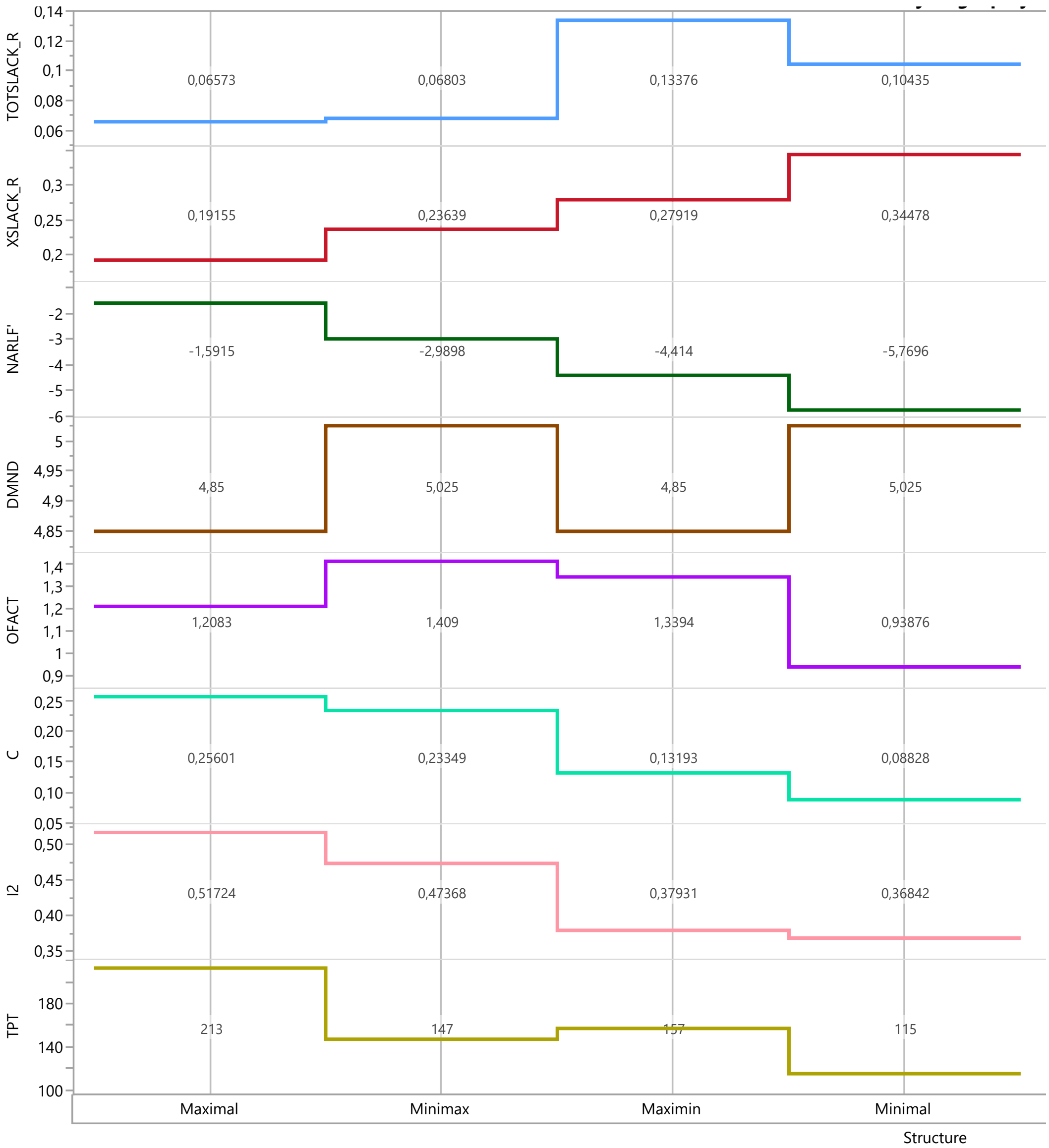


FIGURE 35: Various indicators of flexible project structures in case study
Source: company data

to the variance of TPT. To do this, a Monte Carlo simulation experiment was conducted, where each supplementary task and flexible dependency was included (excluded) in (from) the original project plan with an equal probability of 0.5 (50%), for 100.000 trial runs. The results are shown in Figure 36. Using the same approach, the contribution to the average TPT of the multiproject was also examined and shown in Figure 47 of Appendix E. For the single and multiproject, a similar result could be observed; almost the same tasks are responsible for the changes in the (average) durations. Looking at the duration of the whole portfolio of projects reveals that the last subproject's flexible tasks and dependencies have the most impact on the result, as shown in Figure 48 of Appendix E.

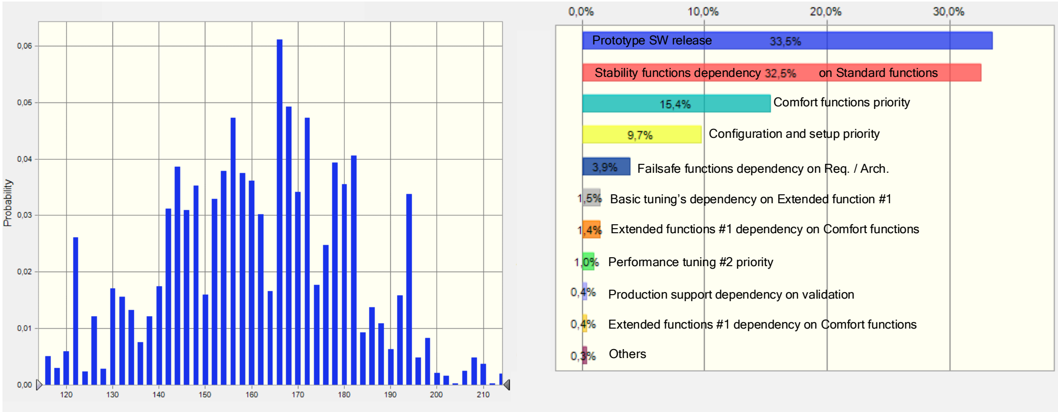


FIGURE 36: Statistical distribution of TPT (left) and contribution of flexible tasks and dependencies within the company project (right)

Finally, with the flexible structure variants, it is possible to examine also the characteristics of multiple projects. The 5 software development (sub)projects have no logical (structural) dependencies with each other. Management makes sure not to start multiple projects at the very same time, to avoid the obvious resource peak. Thus the project arrival times[2] (the earliest time the project can start its main activities) are set to 0, 50, 100, 150, 200, respectively, which means that project "C" can begin 100 time units later than project "A" can.

As mentioned by Pellerin, Perrier, and Berthaut (2020), to this day, exact methods can handle problems typically up to 60 activities and without high resource constrainedness. As the studied multiproject has a high number of activities, heuristics are preferred also because good solutions must often be determined quickly in practice. To provide the best trade-off between accuracy and computation speed, this study proposes a metaheuristic genetic algorithm to solve these instances to near-optimality. The classical objective of the RCPSP or RCMPSP is to minimize the project duration (makespan) but other (often composite) objective functions can be chosen.

Two global renewable resource types[3] ($\rho = 2$) are used: software developers, and testers. The company has 55 unit of developers ($\alpha_1 = 55$) and 45 unit of testers ($\alpha_2 = 45$), and as there is no overtime considered, this resource constraint cannot be violated at any time. The chosen objective function was to minimize the average total project time ($\overline{TPT}$) of the multiproject.

After the optimization procedure for 100.000 iterations on each instance, the results show that feasible solutions can be found for all the fixed structures, respecting all constraints, such as resource and time limits given by the company. As the optimization method is not exact, multiple suboptimal solutions are possible, and with other initial values, further tuning of the parameters could produce a better solution, but it seemed unnecessary in this case.

[2]Release date has the same meaning in the literature but to avoid confusion with software releases, it is intentionally not used here.

[3]The simulation model can also handle local resources if the problem requires it.

TABLE 12: Results for scheduling possible structures of the software development multiproject

Structure	Projects	Tasks	Dependencies	Res. constraints α_1, α_2	$\overline{TPT}_{EST}$	$\overline{TPT}_{OPT}$	$TPT^{portfolio}_{EST}$	$TPT^{portfolio}_{OPT}$
Maximal	5	150	49	55; 45	213	214*	413	413*
Maximin	5	150	38	55; 45	157	203*	357	390*
Minimax	5	100	28	55; 45	147	161*	347	361*
Minimal	5	100	22	55; 45	115	147*	315	338*
Note:	**resource-feasible solution*							

From the summarized results in Table 12, it is visible that all of the earliest start schedules, i.e., all resource-unconstrained cases[4] are infeasible because they violate the resource constraints. The maximal (original) multiproject plan can be finished in 413 time units only after finding a resource-feasible solution by rescheduling activities and causing an increase in the average total project time of the multiproject.

For the maximin and minimax structures, the company needs to invest even more time in the projects to resolve their resource conflicts. Considering flexibility, the optimized maximin structure reduces the average multiproject duration with 5.14%, the minimax reduces approximately 24.77% and a 31.31% reduction can be achieved using the minimal structures.

The approach and results presented here prove the assumption of optimizing fixed project plans that are generated from flexible ones, providing backward compatibility with the most recent algorithms, traditional approaches, and the relatively few methods that support flexible project plans. The solution is given with the scheduled start times (SST) vector for each activity in the multiproject. For reproducibility, the solution for each structure is given in Appendix F in Table 18, and Figure 50 shows an example for the multiproject and project resource profile graphs, compared with the earliest schedule times (EST) for the solution of the maximin structure as well.

6.1.4 Implications

The case study provided valuable insight into the previously observed phenomena in a real-life context. The analysis results were discussed with the relevant experts in the organization to validate its correctness, gather feedback and minimize mistakes or psychological biases. The parameter values observed from company plans validate the defined ranges used throughout the simulation process and are coherent with the empirical observations during data collection. One of the main findings of the case study is that the relatively high available flexibility ratios are not directly considered by the company, at least not on the planning level. With the help of the proposed simulation and optimization framework, it is possible to better utilize flexibility and improve the company's (re)planning processes.

[4] Also known as the critical path method (CPM) proposed by Kelley Jr and Walker (1959)

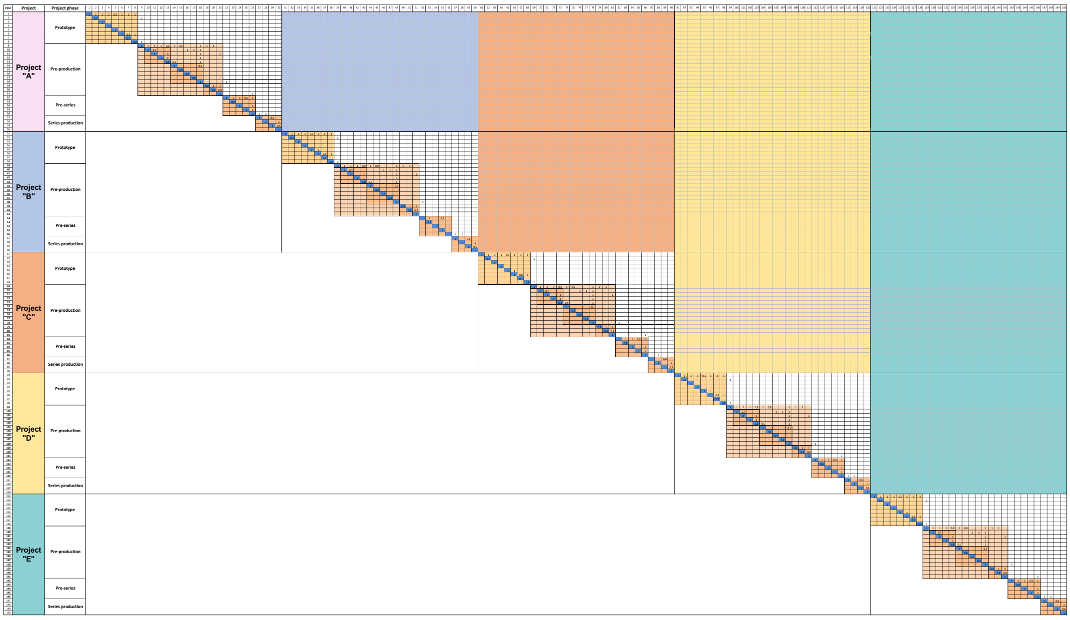


FIGURE 37: Overview of the company multiproject plan

6.2 Threats to validity

The effect of threats to validity needs to be carefully considered not only in the outcome of the research but also during research. Validity is a goal, that cannot be assured, but following a defined structure from the literature, i.e., conclusion-, internal-, construct-, and external validity (Wohlin et al., 2012), threats can be identified and addressed as a mitigation strategy. The validity analysis will be discussed in detail in the following section.

Conclusion validity: besides statistical significance, the necessary assumptions were checked for the statistical methods used in the research (e.g., sample size, independence, normality, homogeneity of variances, etc.). For the metaheuristic optimizations and stochastic simulations (e.g., sensitivity analysis), the number of iterations was defined and set according to typical values from the literature considering the number of input parameters and combinations. Model variables and effects were also analyzed ceteris paribus, changing one at a time while others held constant. The checked ranges included extremely high and low values for each parameter. In stochastic simulations, descriptive statistics were used to show the central tendency and dispersion of results from multiple runs, instead of comparing them to single values. Random numbers were generated based on an initial pseudo-random seed for comparability and reproducibility.

Internal validity: The structural flexibility of tasks and dependencies was generated based on uniform random distribution to avoid any modifying effects of a specific statistical distribution and unnecessary bias in interpreting the outcome of the simulation runs. Furthermore, the number of iterations and sample size were set according to the previously checked stochastic variability of the model. The simulation model's assumptions were peer-reviewed by experts from the field and found no direct impact on the outcome of the independent variable or similar explicit mechanisms implemented.

Construct validity: Simulation is a widely accepted and applicable method in the literature for the topics addressed in this dissertation. Several real-life databases are applied within the research to compare real parameter settings with several artificial ones from the literature. The controlled parameters' effects were visible and could be verified on implemented charts (e.g. detailed resource profiles) and data visualizations within the proposed simulation software. Interactive debugging was used to see the whole chain of effects within the model execution (e.g., spreadsheet trace dependents feature, set breakpoints, etc.). Formulas were checked to produce the correct output. The whole simulation framework was tested and iteratively reviewed during development. Unit tests were applied to support development, regression testing, and consistency. Calculations were cross-checked with examples or results available in the literature. Both academic and professional reviewers with relevant experience made peer reviews and found no missing real-world variables or relationships between them in the model.

External validity: The possibility to exploit structural flexibility, which has originally emerged from the software development environment, is growing with the application in other industries (Batselier and Vanhoucke, 2015). Furthermore, for single projects, it was possible to include a real-life database from five different industries[5]. However, this option is not available for multiprojects due to a lack of relevant empirical data in the literature. To fill this gap, a case study research was carried out in a real multiproject environment considering software development projects. The distribution of project characteristics, such as project duration was compared to real-life cases. The generalization of results is thus possible for multiple industries and contexts where agile approaches apply to some extent.

[5] The covered industries are: IT, production, construction, education, and events

Chapter 7

Summary and Conclusion

In the current dissertation, a quantitative approach supplemented with a case study was provided to evaluate the effects of flexibility on different indicators and project databases.

7.1 Research theses

According to the research questions formed in Section 1.2, four research theses were formulated, considering the results of Chapters 4, 5, and 6.

RT1: [Model] The proposed unified matrix-based project-planning model (UMP) can represent both traditional and flexible single project, multiproject, and program plans. It addresses the demands of renewable and non-renewable resources, time, cost, and quality with single and multiple execution modes.

RT2: [Structures] The flexible structure generator (FSG), is able to specify possible minimal, minimax, maximin, and maximal matrix-based structures corresponding to a defined flexibility parameter, which can be added to the model. The planning phase of projects is improved by considering these additional outcomes with their demands.

RT3: [Indicators] There is a relationship between the modeled flexibility and topology, time-, and resource-related indicators.

RT3.1: [Topology] With an increased rate of flexibility, structural indicators show reduced complexity and reduced serial completions (higher parallelity) for minimal structures.

RT3.2: [Time] As the rate of flexibility increases, time-related indicators show decreased project duration and increased average slack ratio.

RT3.3: [Resources] With increased flexibility ratio, resource-related indicators show higher average resource utilization and higher resource constrainedness considering an early schedule.

RT3.4: [Planning] Flexibility has a negative effect on multiproject planning by significantly increasing the variance of average total project times compared to the traditional method where multiproject plans are more predictable.

RT4: [Solution] With the help of the proposed minimal, minimax, maximin, and maximal structures, it is possible to specify multilevel project plans with supplementary tasks and flexible dependencies in a deterministic way, and solve them both with flexible and traditional methods and algorithms. It is possible to find a feasible, (near) optimal solution minimizing (multi)project duration or other objective functions while considering constraints on time, resources, and cost.

The previously formulated research assumptions could be verified with the results that are validated in Chapter 6, with a case study from an automotive company.

To model heterogeneous project databases, a unified matrix-based project-planning model (UMP) is proposed. To combine existing project databases from the literature, a compound matrix-based project database (CMPD) is proposed that can also handle flexibility. In addition, a flexible structure generator (FSG) is proposed to extend existing project databases with specified structures corresponding to the given flexibility parameter. Companies dealing with agile planning considering supplementary (priorized) activities and dependencies in a project often make decisions and estimates based solely on previous experience. The defined structures can enhance planning of projects by considering their attributes and demands as well. Traditional algorithms can also be tested in flexible project management environments by providing new combinations of the structural- and demand-related indicator values.

As Table 13 highlights the applicability of the proposed models and methods, the proposed UMP addresses both individual and multiple projects, single and multimodal completions, renewable and nonrenewable resources, cost and quality parameters, traditional and flexible project plans (see Table 13). The unified database contains both artificial (simulated) and real-life data sources. The offered parsers are prepared for single and multimode completion modes as well. However, to our best knowledge, there is no existing real-life database for multi-projects and multimodal completion modes. Therefore, the proposed model and methods cannot be tested in these types of real-life projectsThe proposed CMPD provides a wider range of values to test project schedules and resource allocation algorithms by introducing flexibility. These parsers, generators and indicators are available at `https://github.com/novakge/project-parsers` and `https://github.com/novakge/project-indicators`. Table 14 shows the summary of the research.

TABLE 13: Summary table of employed models, generators and databases, and limitations

Data source	Type of projects	Comletion modes	UMP Traditional	UMP Flexible	CMPD Traditional	CMPD Flexible	FSG	Analyzed? Traditional	Analyzed? Flexible
Simulated	Single project	Single-mode	X	X	X	X	X	X	X
Simulated	Single project	Multi-mode	X	X	X	X	X	X	X
Simulated	Multi-project	Single-mode	X	X	X	X	X	X	X
Simulated	Multi-project	Multi-mode	X	X	-	-	-	-	-
Real-life	Single project	Single-mode	X	X	X	X	X	X	X
Real-life	Single project	Multi-mode	-	-	-	-	-	-	-
Real-life	Multi-project	Single-mode	?	?	-	-	-	?	?
Real-life	Multi-project	Multi-mode	-	-	-	-	-	-	-

Notations: 'X' addressed, '-' not addressed, '?' partially addressed.

7.2 Contribution to literature

No databases are currently available to help design and schedule (structurally) flexible projects. This research helps fill the gap. The contributions to the literature and practice are summarized below. 1. A unified matrix-based project-planning model (UMP) is proposed to unify a set of heterogeneous single- and multiproject databases into a compound matrix-based project database (CMPD). 2. The proposed CMPD is complemented by the ability to model flexible dependencies and completion priorities. 3. Minimal, minimax, maximin, and maximal structures are generated to specify the minimal and maximal demands with the proposed flexible structure generator (FSG). 4. Structure-related, time-related, and resource-related indicators are modified to address the flexible nature of projects.

10 project databases, including 22 datasets from sources including Patterson, SMCP and SMFF, PSPLIB, RG300 and RG30, Boctor, MMLIB, MMLIB+, and a real-life project database were combined into a matrix-based project library. Further 5 multiproject databases, including 10 datasets were combined. Current research shows a way of extending the databases to address the flexible nature of the projects. The dissertation gives flexibility-dependent versions of the complexity and the time-related and resource-related indicators of individual projects that can also be applied to multiprojects. It also examines the effects of multiple modes for single projects and project flexibility.

7.3 Practical implications

The proposed matrix-based model addresses cost and nonrenewable demands and quality parameters and manages multiple completion modes and multilevel projects. It not only unifies heterogeneous databases but also allows the user to test both traditional and flexible project scheduling algorithms. The proposed simulation framework supports planning decisions with the characterization, comparison, and optimization of project plans. The open database solves the problem of data availability and reduces time to research and maintenance efforts while enabling collaboration between researchers and practitioners.

TABLE 14: Research summary

Item	Statement
RQ1:	*How to create a unified model that can represent the heterogeneous project and multiproject databases available in the literature?*
RA1:	A model can be created that unifies the different project and multiproject database formats from the literature, including time, cost, renewable-, nonrenewable-resource and quality demands. Existing databases can be imported and further extended with flexible tasks and dependencies into a single, matrix-based database.
RT1:	[Model] The proposed unified matrix-based project-planning model (UMP) can represent both traditional and flexible single project, multiproject, and program plans. It addresses the demands of renewable and non-renewable resources, time, cost, and quality with single and multiple execution modes.
RQ2:	*How the flexibility of single- and multiproject plans can be modeled?*
RA2:	Flexible project plans can be generated from existing traditional (multi)project plans and new possible structures can be added to the model to improve the planning process.
RT2:	[Structures] The flexible structure generator (FSG), is able to specify possible minimal, minimax, maximin, and maximal matrix-based structures corresponding to a defined flexibility parameter, which can be added to the model. The planning phase of projects is improved by considering these additional outcomes with their demands.
RQ3:	*What characterizes the topology (structure) and the different demands of the flexible project and multiproject plans?*
RA3:	Existing project-related indicators for topology, time- and resource-related demands can be adapted for flexible projects and multiprojects to analyze the effects of flexibility.
RT3:	[Indicators] There is a relationship between the modeled flexibility and topology, time-, and resource-related indicators.
RT3.1:	[Topology] With an increased rate of flexibility, structural indicators show reduced complexity and reduced serial completions (higher parallelity) for minimal structures.
RT3.2:	[Time] As the rate of flexibility increases, time-related indicators show decreased project duration and increased average slack ratio.
RT3.3:	[Resources] With increased flexibility ratio, resource-related indicators show higher average resource utilization and higher resource constrainedness considering an early schedule.
RT3.4:	[Planning] Flexibility has a negative effect on multiproject planning by significantly increasing the variance of average total project times compared to the traditional method where multiproject plans are more predictable.
RQ4:	*How is it possible to find feasible (sub)optimal solution for the single- and multiproject plans considering flexibility?*
RA4:	Flexible multilevel projects can be scheduled and near-optimal solutions can be found. A simulation framework can be constructed to handle flexible dependencies and supplementary tasks.
RT4:	[Solution] With the help of the proposed minimal, minimax, maximin, and maximal structures, it is possible to specify multilevel project plans with supplementary tasks and flexible dependencies in a deterministic way, and solve them both with flexible and traditional methods and algorithms. It is possible to find a feasible, (near) optimal solution minimizing (multi)project duration or other objective functions while considering constraints on time, resources, and cost.

Chapter 8

Limitations

To ensure the comparison between simulated and real-life projects, projects were examined mainly with time and renewable resource demands. However, the presence of nonrenewable resources, cost, and quality demands would open possibilities for further research. Moreover, with the introduction of both artificial and real-life multi-mode multiproject, agile single project, and agile multiproject databases, it would be interesting to compare these with existing databases as well. Currently, there are no real-life multiproject and no agile project databases (of any kind) available, thus the direct comparison between simulated (artificial) and real data is limited to single projects. Multiproject plans from other industries could further diversify the scope of analysis.

Appendix A

Appendix

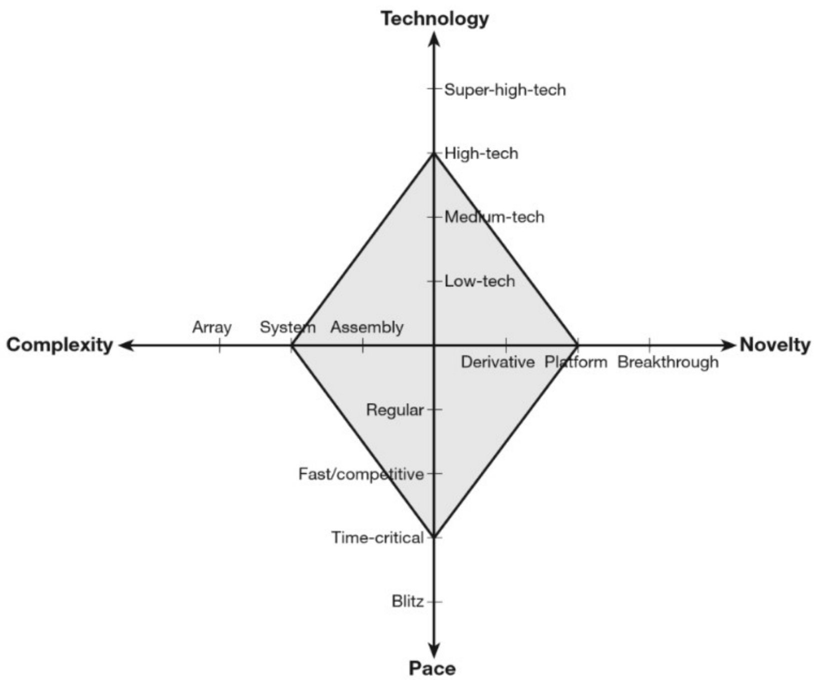


FIGURE 38: Project classification of Shenhar and Dvir (2007)

Appendix B

Results for multiple mode analysis of artificial single projects

FIGURE 39: Single project time-related indicators of multimode datasets

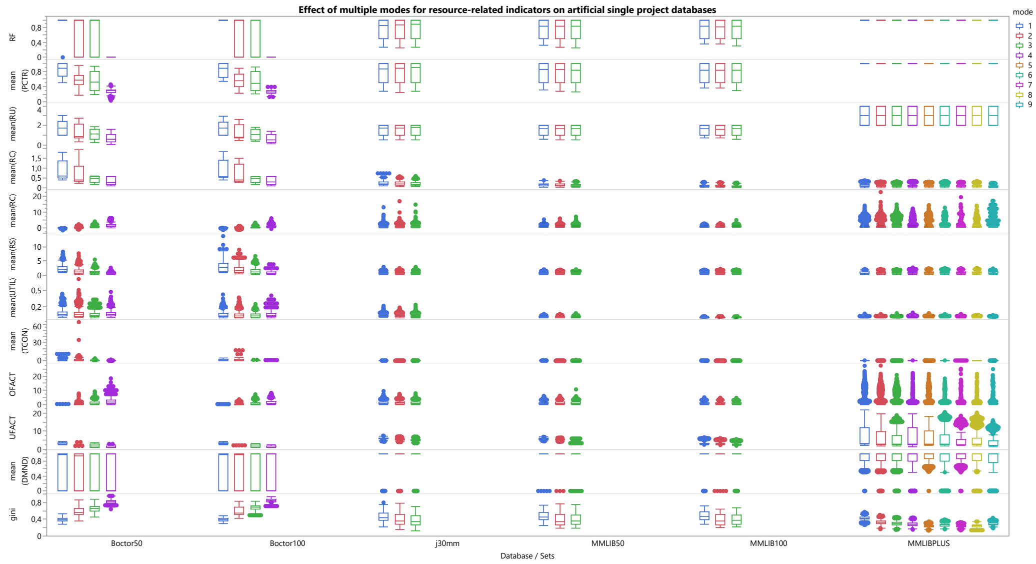


FIGURE 40: Single project resource-related indicators of multimode datasets

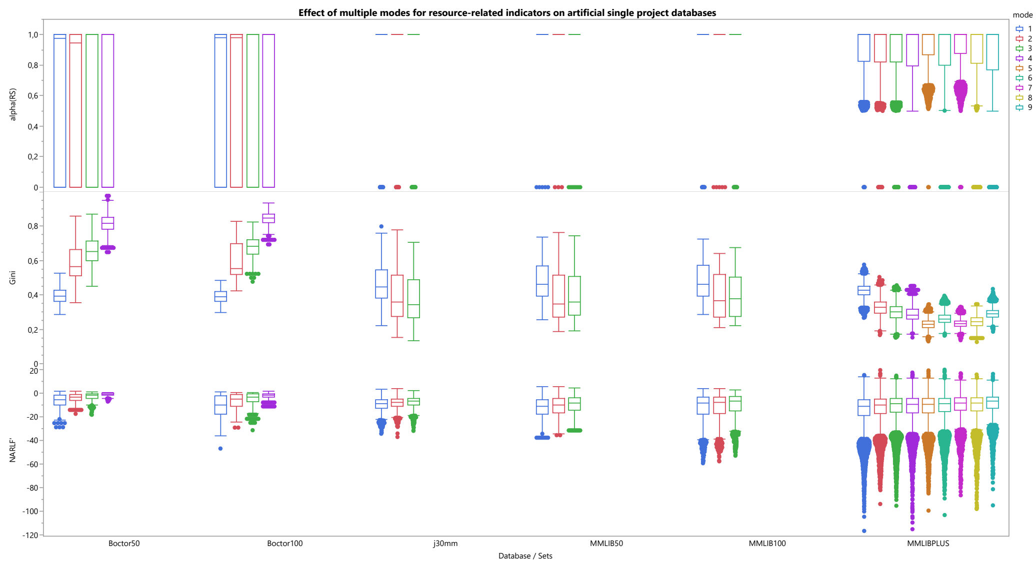


FIGURE 41: Single project distribution of resource-related indicators of multimode datasets

Appendix C

Case study: company project plan with time and resource demands

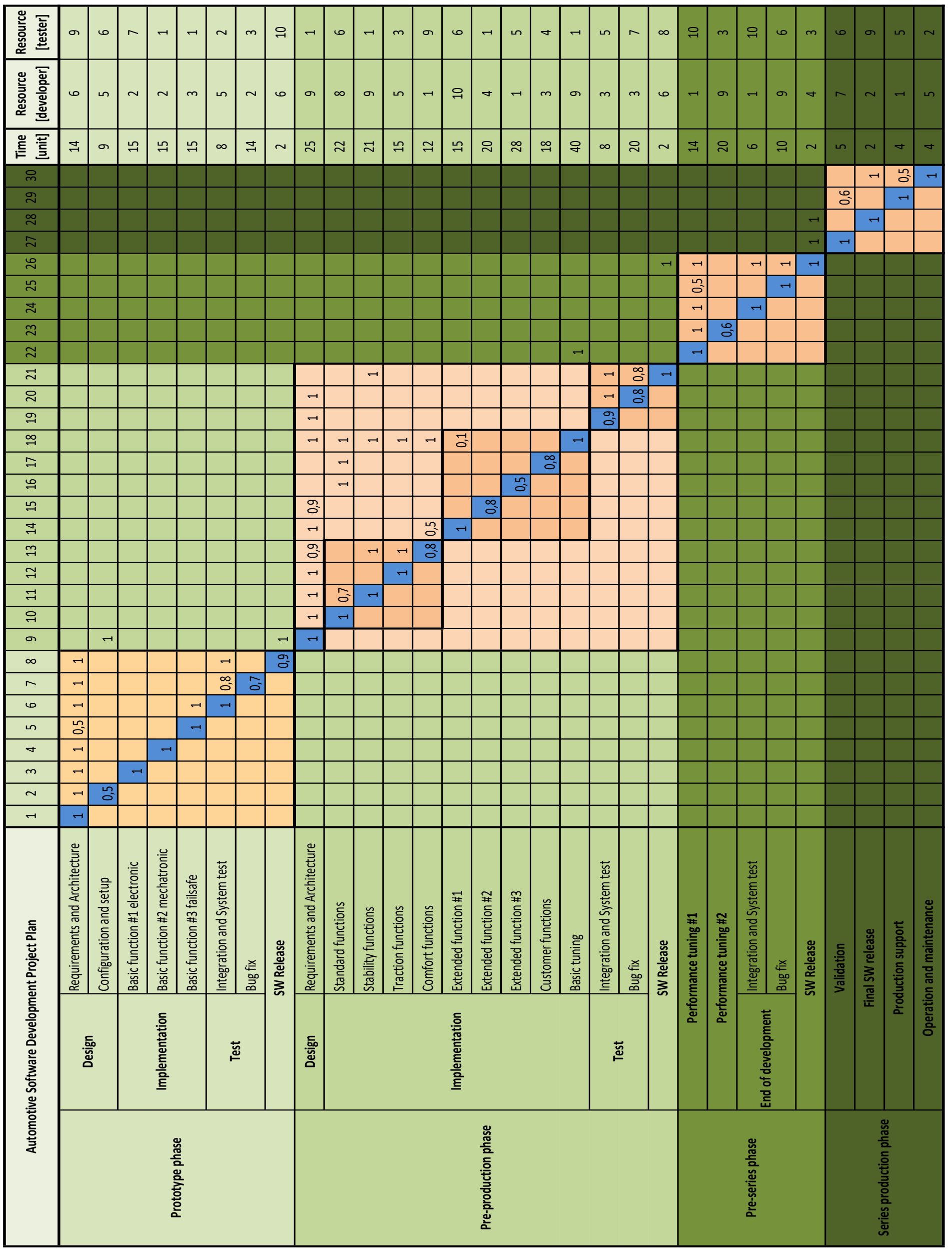


FIGURE 42: Automotive software development project plan

Appendix D

Details of regression for flexibility and duration in multiprojects

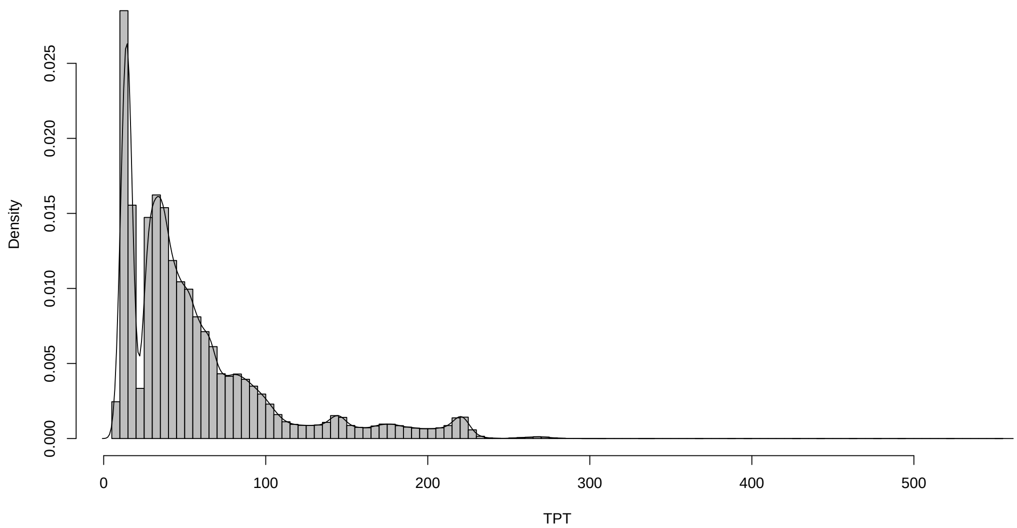


FIGURE 43: Histogram of TPT

TABLE 15: Comparing different quantile regressions for $\tau = \{0.25, 0.5, 0.75\}$

	Dependent variable:		
	TPT		
	(1)	(2)	(3)
fp	-77.500^{***}	-153.833^{***}	-235.750^{***}
	(1.286)	(0.398)	(0.629)
Constant	53.050^{***}	86.150^{***}	121.600^{***}
	(0.502)	(0.154)	(0.238)
Observations	260,690	260,690	260,690
Note:			*p<0.1; **p<0.05; ***p<0.01

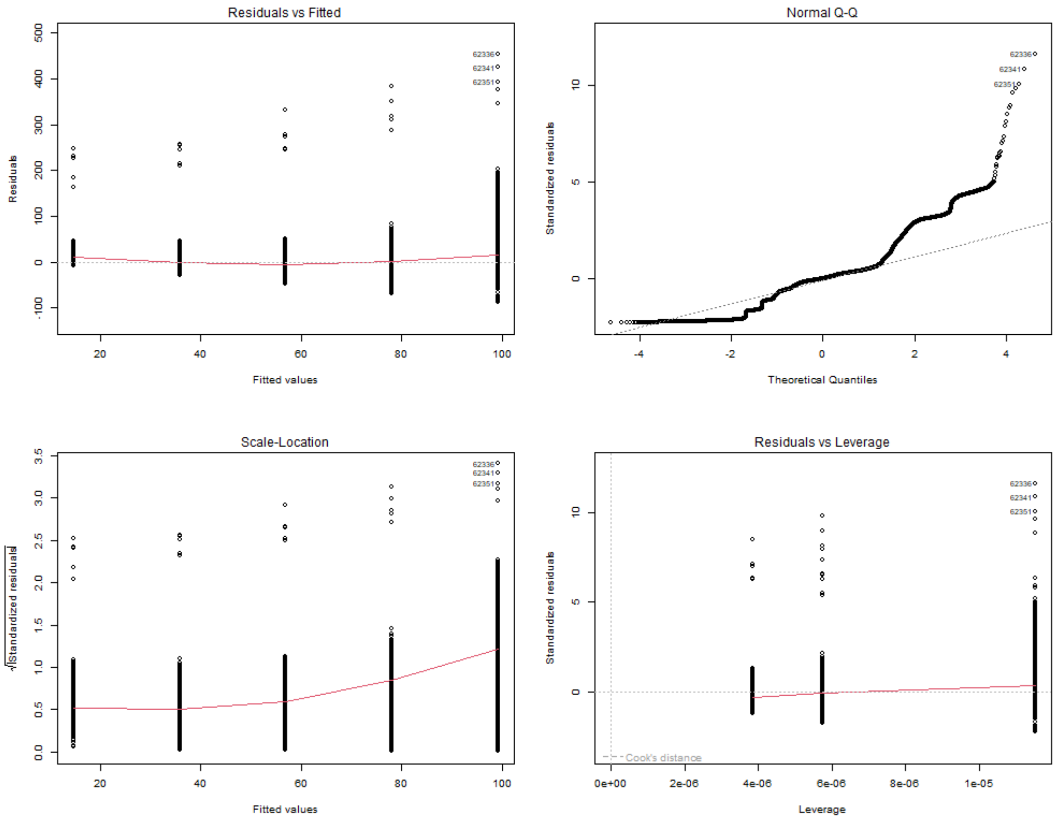


FIGURE 44: Various diagnostic plots for regression

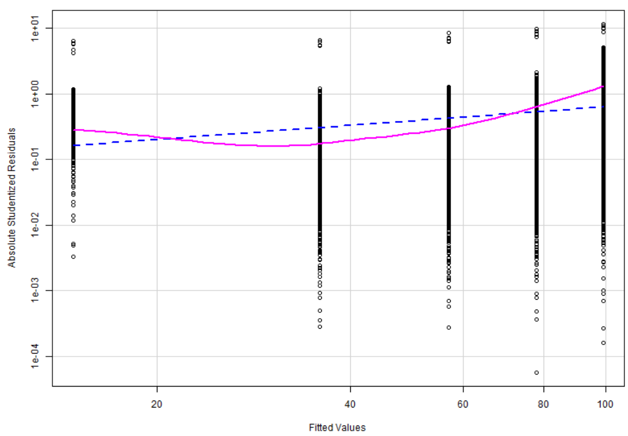


FIGURE 45: Spread-Level plot for regression

TABLE 16: Effect of flexibility on TPT in multiprojects (quantile regression)

	Dependent variable:
	TPT
fp	-153.833^{***}
	(0.398)
Constant	86.150^{***}
	(0.154)
Observations	260,690
Note:	*p<0.1; **p<0.05; ***p<0.01

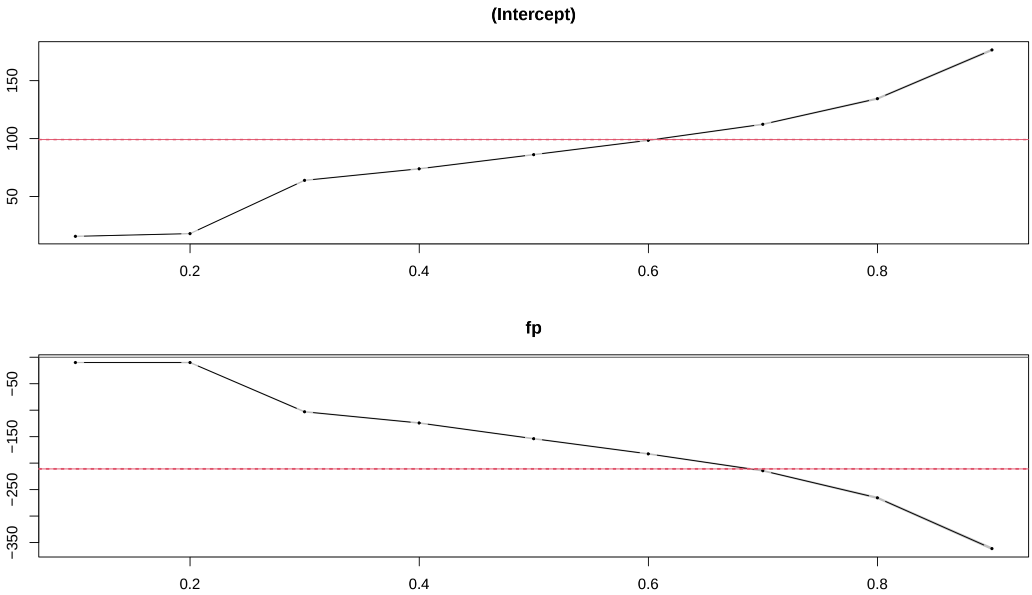


FIGURE 46: Multiple τ test for multiprojects, exploring conditional quantiles other than median

TABLE 17: Effect of flexibility on TPT in multiprojects (GLS regression)

	Dependent variable:
	TPT
fp	-171.399^{***}
	(0.414)
Constant	89.587^{***}
	(0.127)
Observations	260,690
R^2	0.396
Adjusted R^2	0.396
Residual Std. Error	8.030 (df = 260688)
F Statistic	$171{,}065.600^{***}$ (df = 1; 260688)
Note:	*p<0.1; **p<0.05; ***p<0.01

Appendix E

Sensitivity analysis of the company's multiproject

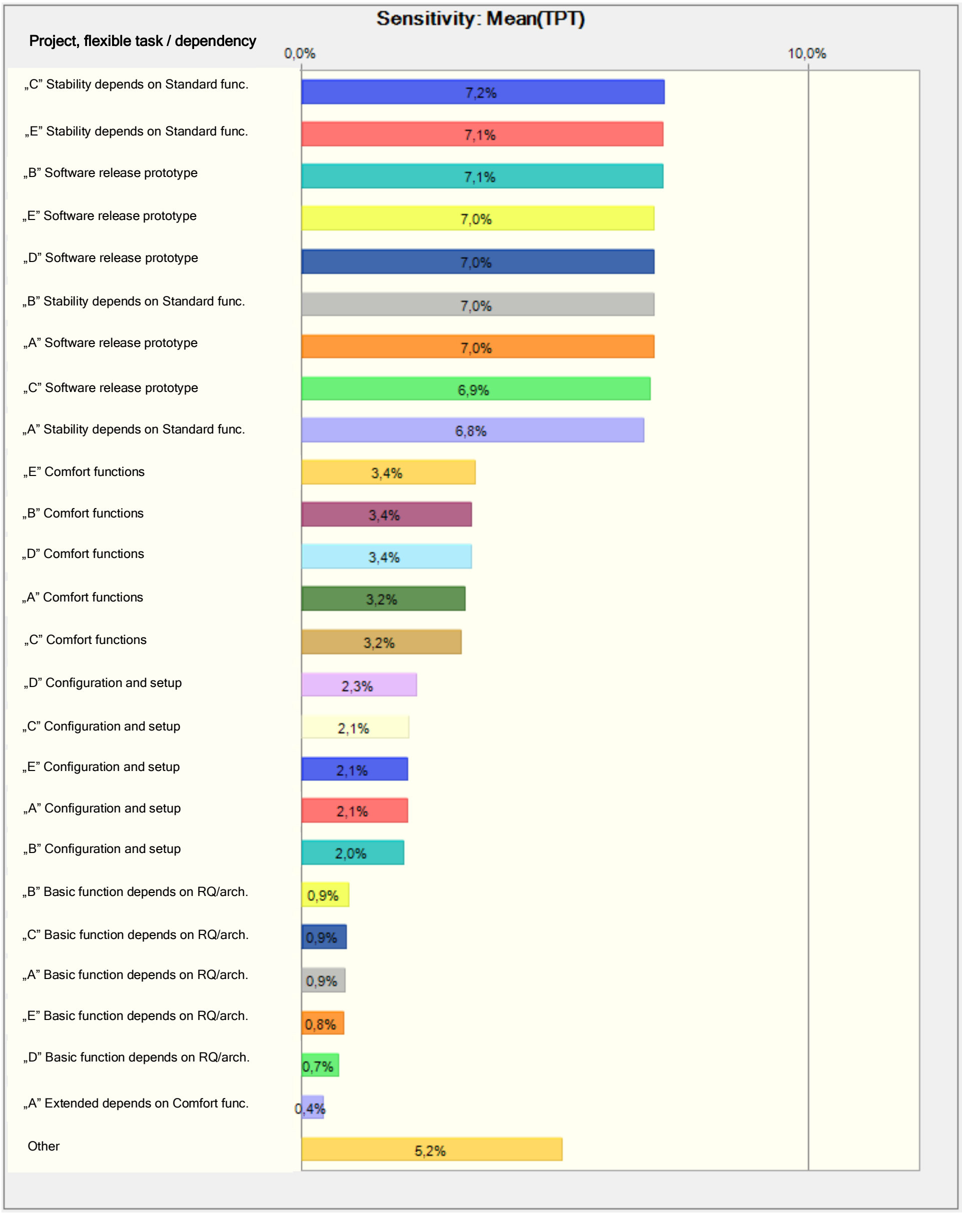


FIGURE 47: The contribution of flexible dependencies and tasks to the average duration of the company's multiproject

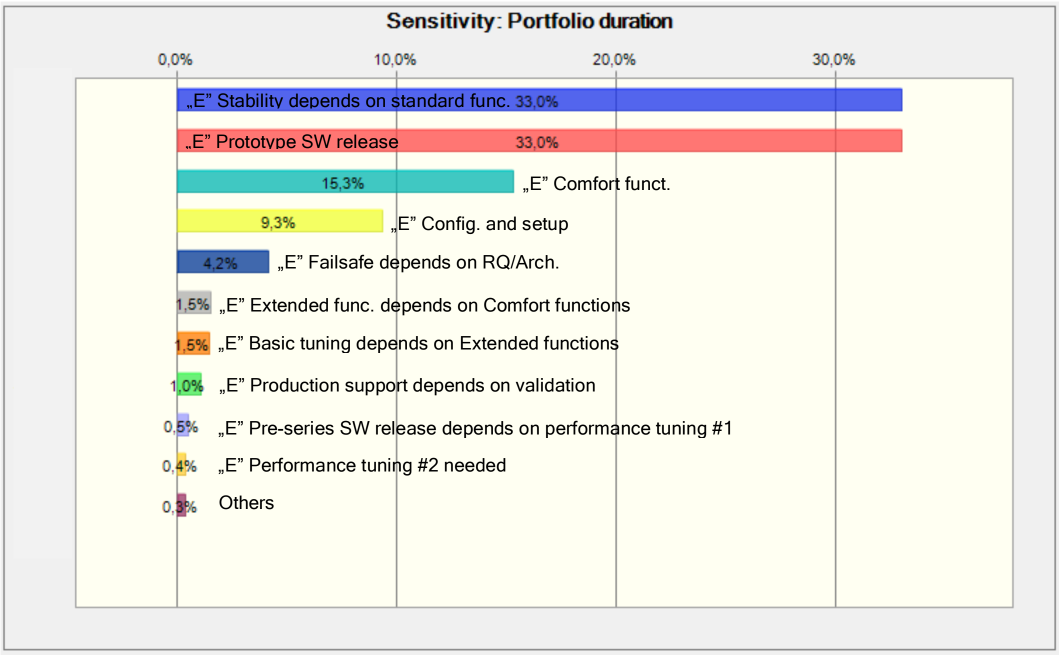


FIGURE 48: The contribution of flexible dependencies and tasks to the duration of the company's portfolio

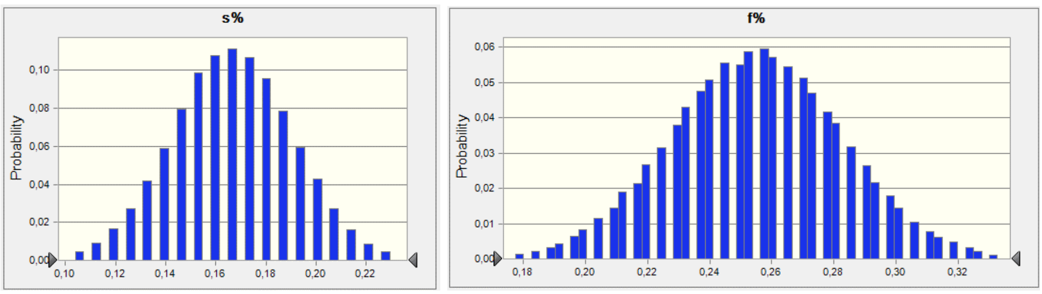


FIGURE 49: Probability distribution of the flexibility indicator values $f\%$ and $s\%$ throughout the sensitivity analysis for the company's multiproject

Appendix F

Solutions of company multiproject plans

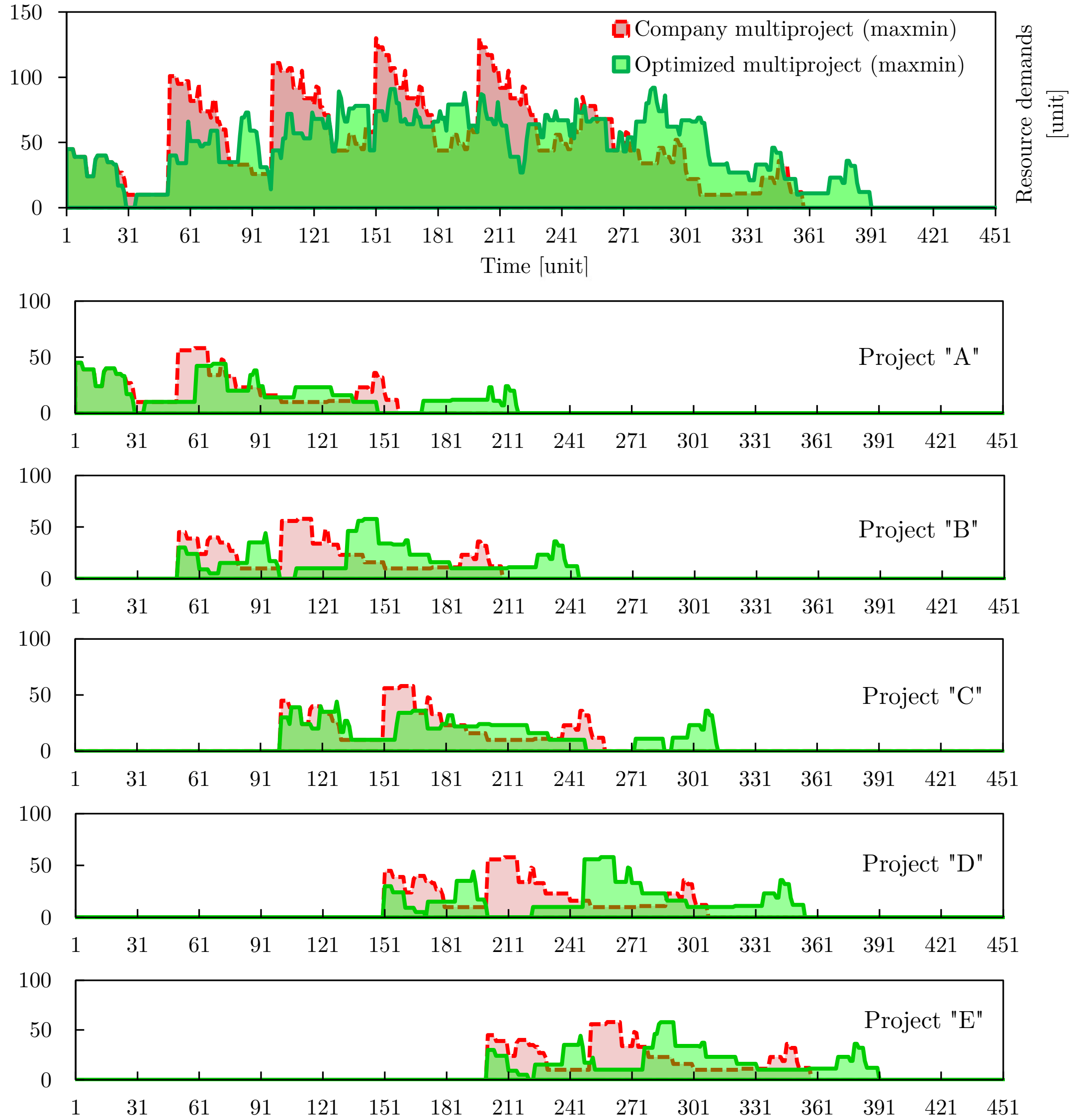


FIGURE 50: Aggregated resource profiles before and after optimization of the maximin company multiproject plan

TABLE 18: Optimized solutions for the different flexible multiproject structures

Structure	$SST_{maximal}$					$SST_{maximin}$					$SST_{minimax}$					$SST_{minimal}$				
Task of subproject	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E
T1	0	50	100	150	200	0	70	105	171	223	0	50	100	150	200	0	50	100	150	200
T2	14	64	114	164	214	14	84	119	185	237	14	64	114	164	214	14	64	114	164	214
T3	14	64	114	164	214	14	84	119	185	237	14	64	114	164	214	14	64	114	164	214
T4	14	64	114	164	214	14	84	119	185	237	14	64	114	164	214	14	64	114	164	214
T5	14	64	114	164	214	0	50	100	150	200	14	64	114	164	214	0	50	110	158	209
T6	29	79	129	179	229	15	84	119	185	237	29	79	129	179	229	15	65	125	173	224
T7	37	87	137	187	237	14	84	119	185	237	56	106	156	207	253	23	73	133	181	232
T8	37	87	137	187	237	23	92	127	193	245	66	116	166	216	256	23	73	133	181	232
T9	39	89	139	189	239	34	107	131	222	251	14	64	114	164	214	13	64	113	163	211
T10	64	114	164	214	264	85	132	180	247	281	39	89	139	189	239	38	89	138	188	236
T11	86	136	186	236	286	59	137	157	247	283	61	111	161	211	261	61	110	161	208	249
T12	93	114	164	214	264	59	132	181	247	276	39	89	139	189	239	38	89	138	188	236
T13	108	157	207	257	307	80	158	196	268	304	54	104	154	204	254	53	104	153	203	251
T14	120	169	219	269	319	59	132	156	247	276	39	89	139	189	239	38	89	138	188	236
T15	64	114	164	214	264	0	50	100	150	200	54	104	154	204	254	53	104	153	203	251
T16	86	136	186	236	286	107	154	202	269	303	54	104	154	204	254	53	104	153	203	251
T17	86	136	186	236	286	107	154	202	269	303	54	104	154	204	254	53	104	153	203	251
T18	135	184	234	284	334	107	170	208	280	316	82	132	182	232	282	82	131	182	229	270
T19	64	114	164	214	264	59	132	156	247	276	122	172	222	272	322	122	171	222	269	310
T20	72	122	172	222	272	67	140	164	255	284	122	172	222	272	322	122	171	222	269	310
T21	107	157	207	257	307	87	160	178	268	304	82	132	182	232	282	82	131	182	229	270
T22	175	224	274	324	374	169	210	272	320	356	122	172	222	272	322	122	171	222	269	310
T23	189	238	288	338	388	183	224	290	334	370	136	186	236	286	336	136	185	236	283	324
T24	189	238	288	338	388	201	224	298	334	370	136	186	236	286	336	136	185	236	283	324
T25	189	240	288	338	388	0	50	100	150	200	136	186	236	286	336	0	54	100	271	254
T26	199	250	298	348	398	207	230	304	340	376	146	196	246	296	346	142	191	242	289	330
T27	201	252	300	350	400	209	232	306	342	378	148	198	248	298	348	144	193	244	291	332
T28	201	252	300	350	400	209	232	306	342	378	148	198	248	298	348	144	193	244	291	332
T29	206	257	305	355	405	0	50	100	150	200	153	203	253	303	353	0	50	100	150	200
T30	210	261	309	359	409	211	234	308	344	380	157	207	257	307	357	146	195	246	293	334

Appendix G

Electronic supplementary materials

All supplementary materials and resources related to the dissertation can be found online on GitHub.

1. Project indicators github repository:
   https://github.com/novakge/project-indicators
2. Project database parsers github repository:
   https://github.com/novakge/project-parsers
3. Scripts and data for figures and analysis
   R scripts, data and analysis
   MATLAB scripts, data and analysis
4. Excel calculations and data
5. Excel simulation framework

Bibliography

Aarseth, Wenche, Rolstadås, Asbjørn, and Andersen, Bjorn (2014). "Managing organizational challenges in global projects". In: *International Journal of Managing Projects in Business*.

Abad, Zahra Shakeri Hossein, Sadi, Mahsa Hasani, and Ramsin, Raman (Nov. 2010). "Towards tool support for situational engineering of agile methodologies". In: *2010 Asia Pacific Software Engineering Conference*. IEEE, pp. 326–335. DOI: 10.1109/APSEC.2010.45. URL: https://doi.org/10.1109/APSEC.2010.45.

Aitken, Alexander C (1936). "IV.—On least squares and linear combination of observations". In: *Proceedings of the Royal Society of Edinburgh* 55, pp. 42–48.

Andres, Hayward P and Zmud, Robert W (2002). "A contingency approach to software project coordination". In: *Journal of Management Information Systems* 18.3, pp. 41–70.

Archibald, Russell D (2003). *Managing high-technology programs and projects*. John Wiley & Sons.

Arefazar, Yasaman, Nazari, Ahad, Hafezi, Mohammad Reza, and Maghool, Sayyed Amir Hossain (2022). "Prioritizing agile project management strategies as a change management tool in construction projects". In: *International Journal of Construction Management* 22.4, pp. 678–689.

Aritua, Bernard, Smith, Nigel J, and Bower, Denise (2009). "Construction client multi-projects–A complex adaptive systems perspective". In: *International Journal of Project Management* 27.1, pp. 72–79.

Artto, Karlos, Kujala, Jaakko, Dietrich, Perttu, and Martinsuo, Miia (2008). "What is project strategy?" In: *International Journal of Project Management* 26.1, pp. 4–12.

Atkinson, Roger (1999). "Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria". In: *International journal of project management* 17.6, pp. 337–342.

Baccarini, David (1996). "The concept of project complexity—a review". In: *International journal of project management* 14.4, pp. 201–204.

Bakalova, Zornitza, Daneva, Maya, Herrmann, Andrea, and Wieringa, Roel (2011). "Agile requirements prioritization: What happens in practice and what is described in literature". In: *International Working Conference on Requirements Engineering: Foundation for Software Quality*. Springer, pp. 181–195.

Baptiste, Philippe and Pape, Claude Le (2000). "Constraint propagation and decomposition techniques for highly disjunctive and highly cumulative project scheduling problems". In: *Constraints* 5.1, pp. 119–139.

Batselier, Jordy and Vanhoucke, Mario (2015). "Construction and evaluation framework for a real-life project database". In: *International Journal of Project Management* 33.3, pp. 697–710.

Bergmann, Thomas and Karwowski, Waldemar (2018). "Agile project management and project success: A literature review". In: *International Conference on Applied Human Factors and Ergonomics*. Springer, pp. 405–414.

Bernardes, Ednilson Santos and Hanna, Mark D (2009). "A theoretical review of flexibility, agility and responsiveness in the operations management literature: Toward a conceptual definition of customer responsiveness". In: *International Journal of Operations & Production Management*.

Bianchi, Mattia, Marzi, Giacomo, Dabic, Marina, et al. (2018). "Call for Papers/Special Issue: Agile beyond software-In search of flexibility in a wide range of innovation projects and industries". In:

Boctor, Fayez Fouad (1993). "Heuristics for scheduling projects with resource restrictions and several resource-duration modes". In: *The international journal of production research* 31.11, pp. 2547–2558. DOI: `10.1080/00207549308956882`.

Boehm, Barry and Turner, Richard (2003). "People factors in software management: lessons from comparing agile and plan-driven methods". In: *Crosstalk-The Journal of Defense Software Engineering,(Dec 2003)*.

Boehm, Barry and Turner, Richard (2005). "Management challenges to implementing agile processes in traditional development organizations". In: *IEEE software* 22.5, pp. 30–39.

Bose, Indranil (2008). "Lessons learned from distributed agile software projects: A case-based analysis". In: *Communications of the Association for Information Systems* 23.1, p. 34.

Browning, Tyson R and Yassine, Ali A (2010a). "A random generator of resource-constrained multi-project network problems". In: *Journal of Scheduling* 13.2, pp. 143–161. DOI: `10.1007/s10951-009-0131-y`.

Browning, Tyson R and Yassine, Ali A (2010b). "Resource-constrained multi-project scheduling: Priority rule performance revisited". In: *International Journal of Production Economics* 126.2, pp. 212–228.

Brucker, Peter, Drexl, Andreas, Möhring, Rolf, Neumann, Klaus, and Pesch, Erwin (1999). "Resource-constrained project scheduling: Notation, classification, models, and methods". In: *European journal of operational research* 112.1, pp. 3–41. DOI: `10.1016/S0377-2217(98)00204-5`.

Čapek, R., Šůcha, P., and Hanzálek, Z. (2012). "Production scheduling with alternative process plans". In: *European Journal of Operational Research* 217.2, pp. 300–311. ISSN: 0377-2217. DOI: `https://doi.org/10.1016/j.ejor.2011.09.018`.

Carlier, Jacques and Néron, Emmanuel (2003). "On linear lower bounds for the resource constrained project scheduling problem". In: *European Journal of Operational Research* 149.2, pp. 314–324.

Charvat, Jason (2003). "Project management methodologies: selecting, implementing, and supporting methodologies and processes for projects". In:

Chen, Chun-Hsien, Ling, Shih Fu, and Chen, Wei (2003). "Project scheduling for collaborative product development using DSM". In: *International Journal of Project Management* 21.4, pp. 291–299.

Ciric, Danijela, Lalic, Bojan, Gracanin, Danijela, Palcic, Iztok, and Zivlak, Nikola (2018). "Agile project management in new product development and innovation processes: challenges and benefits beyond software domain". In: *2018 IEEE International Symposium on Innovation and Entrepreneurship (TEMS-ISIE)*. IEEE, pp. 1–9.

Ciric, Danijela, Lalic, Bojan, Gracanin, Danijela, Tasic, Nemanja, Delic, Milan, and Medic, Nenad (2019). "Agile vs. Traditional approach in project management: Strategies, challenges and reasons to introduce agile". In: *Procedia Manufacturing* 39, pp. 1407–1414.

Ciriello, Raffaele Fabio, Glud, Jeppe Aagaard, and Hansen-Schwartz, Kevin Helge (2022). "Becoming agile together: Customer influence on agile adoption within commissioned software teams". In: *Information & Management* 59.4, p. 103645. DOI: 10.1016/j.im.2022.103645.

Cleland, David I (1994). "Project Management Strategic Design and Implementation (1999), s 483. 92 Dodgson". In: *Mark & Rothwell Roy The Handbook of Industrial Innovation*, p. 331.

Cleland, David I (2007). *Project management: strategic design and implementation*. McGraw-Hill Education.

Coelho, José and Vanhoucke, Mario (2020). "Going to the core of hard resource-constrained project scheduling instances". In: *Computers & Operations Research* 121, p. 104976. DOI: 10.1016/j.cor.2020.104976.

Collyer, Simon and Warren, Clive MJ (2009). "Project management approaches for dynamic environments". In: *International Journal of Project Management* 27.4, pp. 355–364.

Conboy, Kieran (2010). "Project failure en masse: a study of loose budgetary control in ISD projects". In: *European Journal of Information Systems* 19.3, pp. 273–287.

Conforto, Edivandro C and Amaral, Daniel C (2016). "Agile project management and stage-gate model—A hybrid framework for technology-based companies". In: *Journal of Engineering and Technology Management* 40, pp. 1–14.

Conforto, Edivandro C, Salum, Fabian, and Amaral, Daniel C (2014). "Can agile project management be adopted by industries other than software development?" In: *Project Management Journal* 45.3, pp. 21–34.

Continental Annual Report (2021). https://annualreport.continental.com/2021/en/service/docs/annual-report-2021-data.pdf. Accessed: 2022-07-01.

Cooke-Davies, Terry (2002). "The "real" success factors on projects". In: *International journal of project management* 20.3, pp. 185–190.

Cooper, Robert G, Edgett, Scott J, and Kleinschmidt, Elko J (2000). "New problems, new solutions: making portfolio management more effective". In: *Research-Technology Management* 43.2, pp. 18–33.

Cooper, Robert G, Edgett, Scott J, and Kleinschmidt, Elko J (2001). "Portfolio management for new products". In:

Corsten, Hans and Corsten, Hilde (2000). *Projektmanagement: Einführung*. Oldenbourg Verlag.

Creemers, Stefan, Reyck, Bert De, and Leus, Roel (2015). "Project planning with alternative technologies in uncertain environments". In: *European Journal of Operational Research* 242.2, pp. 465–476. ISSN: 0377-2217. DOI: `10.1016/j.ejor.2014.11.014`.

Danilovic, Mike and Browning, Tyson R (2007). "Managing complex product development projects with design structure matrices and domain mapping matrices". In: *International journal of project management* 25.3, pp. 300–314.

Danilovic, Mike and Sandkull, Bengt (2002). "Managing Complexity and Uncertainty in a Multiproject Environment". In: *International Research Network on Organizing By Projects (IRNOP V), Rotterdam, Netherlands, 28-31 May, 2002*. Erasmus University.

Davis, Edward W (1975). "Project network summary measures constrained-resource scheduling". In: *AIIE Transactions* 7.2, pp. 132–142. DOI: `10.1080/05695557508974995`.

Debels, Dieter and Vanhoucke, Mario (2007). "A decomposition-based genetic algorithm for the resource-constrained project-scheduling problem". In: *Operations Research* 55.3, pp. 457–469. DOI: `10.1287/opre.1060.0358`.

DeCarlo, Doug (2004). "Extreme Project Management: Using Leadership". In: *Principles and Tools to Deliver Value in the Face of Volatility. Jossey-Bass, San Francisco*.

Delisle, Julie (2020). "Working time in multi-project settings: How project workers manage work overload". In: *International Journal of Project Management* 38.7, pp. 419–428.

Demeulemeester, Erik L, Herroelen, Willy S, and Elmaghraby, Salah E (1996). "Optimal procedures for the discrete time/cost trade-off problem in project networks". In: *European Journal of Operational Research* 88.1, pp. 50–68. DOI: `10.1016/0377-2217(94)00181-2`.

Demeulemeester, Erik L, Vanhoucke, Mario, and Herroelen, Willy (2003). "RanGen: A random network generator for activity-on-the-node networks". In: *Journal of Scheduling* 6.1, pp. 17–38. DOI: `10.1023/A:1022283403119`.

Denizer, Cevdet, Kaufmann, Daniel, and Kraay, Aart (2013). "Good countries or good projects? Macro and micro correlates of World Bank project performance". In: *Journal of Development Economics* 105, pp. 288–302. ISSN: 0304-3878. DOI: `10.1016/j.jdeveco.2013.06.003`.

Di Muro, Paolo, Lecoeuvre, Laurence, and Turner, Rodney (2021). "Ambidextrous strategy and execution in entrepreneurial project-oriented organizations: The case of Pagani supercars". In: *International Journal of Project Management* 39.1, pp. 45–58.

Dike, Sheldon H (1964). "Project scheduling with resource constraints". In: *IEEE Transactions on Engineering Management* 4, pp. 155–157.

Dingsøyr, Torgeir and Moe, Nils Brede (2014). "Towards principles of large-scale agile development". In: *International Conference on Agile Software Development*. Springer, pp. 1–8.

Dingsøyr, Torgeir, Nerur, Sridhar, Balijepally, VenuGopal, and Moe, Nils Brede (2012). *A decade of agile methodologies: Towards explaining agile software development*. DOI: 10.1016/j.jss.2012.02.033.

Dooley, Lawrence, Lupton, Gary, and O'Sullivan, David (2005). "Multiple project management: a modern competitive necessity". In: *Journal of Manufacturing Technology Management*.

Drexl, Andreas, Nissen, Ruediger, Patterson, James H, and Salewski, Frank (2000). "Progen/πx–An instance generator for resource-constrained project scheduling problems with partially renewable resources and further extensions". In: *European Journal of Operational Research* 125.1, pp. 59–72.

Duimering, P Robert, Ran, Bing, Derbentseva, Natalia, and Poile, Christopher (2006). "The effects of ambiguity on project task structure in new product development". In: *Knowledge and Process Management* 13.4, pp. 239–251.

Dumitriu, Florin, Meșniță, Gabriela, and Radu, Laura-Diana (2019). "Challenges and Solutions of Applying Large-Scale Agile at Organizational Level." In: *Informatica Economica* 23.3.

Dybå, Tore and Dingsøyr, Torgeir (2008). "Empirical studies of agile software development: A systematic review". In: *Information and software technology* 50.9-10, pp. 833–859.

Dybå, Tore, Dingsøyr, Torgeir, and Moe, Nils Brede (2014). "Agile project management". In: *Software project management in a changing world*. Springer, pp. 277–300.

Eisner, Howard (1962). "A generalized network approach to the planning and scheduling of a research project". In: *Operations Research* 10.1, pp. 115–125.

Elonen, Suvi and Artto, Karlos (2003). "Problems in managing internal development projects in multi-project environments". In: *International journal of project management* 21.6, pp. 395–402.

Engwall, Mats and Jerbrant, Anna (2003). "The resource allocation syndrome: the prime challenge of multi-project management?" In: *International journal of project management* 21.6, pp. 403–409.

Eppinger, Steven D, Whitney, Daniel E, Smith, Robert P, and Gebala, David A (1989). "Organizing the tasks in complex design projects". In: *Workshop on Computer-Aided Cooperative Product Development*. Springer, pp. 229–252.

Eskandari, Hamidreza, Mahmoodi, Ehsan, Fallah, Hamed, and Geiger, Christopher D (2011). "Performance analysis of comercial simulation-based optimization packages: OptQuest and Witness Optimizer". In: *Proceedings of the 2011 Winter Simulation Conference (WSC)*. IEEE, pp. 2358–2368.

Evaristo, J Roberto and Scudder, Richard (2000). "Geographically distributed project teams: a dimensional analysis". In: *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*. IEEE, 11–pp.

Evaristo, J Roberto, Scudder, Richard, Desouza, Kevin C, and Sato, Osam (2004). "A dimensional analysis of geographically distributed project teams: a case study". In: *Journal of Engineering and technology Management* 21.3, pp. 175–189.

Evaristo, Roberto and Van Fenema, Paul C (1999). "A typology of project management: emergence and evolution of new forms". In: *International journal of project management* 17.5, pp. 275–281.

Fernandez, Daniel J and Fernandez, John D (2008). "Agile project management—agilism versus traditional approaches". In: *Journal of Computer Information Systems* 49.2, pp. 10–17. DOI: `10.1080/08874417.2009.11646044`.

Forsberg, Kevin and Mooz, Harold (1991). "The relationship of system engineering to the project cycle". In: *INCOSE international symposium*. Vol. 1. 1. Wiley Online Library, pp. 57–65.

Fowler, Martin, Highsmith, Jim, et al. (2001). "The agile manifesto". In: *Software development* 9.8, pp. 28–35.

Franco-Duran, Diana M and Garza, Jesús M de la (2019). "Review of Resource-Constrained Scheduling Algorithms". In: *Journal of Construction Engineering and Management* 145.11, p. 03119006. DOI: `10.1061/(ASCE)CO.1943-7862.0001698`.

Fricke, Scott E and Shenbar, AJ (2000). "Managing multiple engineering projects in a manufacturing support environment". In: *IEEE Transactions on engineering management* 47.2, pp. 258–268.

Gareis, Roland (1991). "Management by projects: the management strategy of the 'new'project-oriented company". In: *International Journal of Project Management* 9.2, pp. 71–76.

Gareis, Roland (2000). "Managing the project start". In: *Turner JR and SJ Simister (editors)*.

Gini, Corrado (1936). "On the measure of concentration with special reference to income and statistics". In: *Colorado College Publication, General Series* 208.1, pp. 73–79.

Godenhjelm, Sebastian, Lundin, Rolf A, and Sjöblom, Stefan (2015). "Projectification in the public sector–the case of the European Union". In: *International Journal of Managing Projects in Business*.

Görög, Mihály (2003). *A projektvezetés mestersége*. Aula.

Govil, Nikhil and Sharma, Ashish (2021). "Information extraction on requirement prioritization approaches in agile software development processes". In: *2021 5th*

International Conference on Computing Methodologies and Communication (ICCMC). IEEE, pp. 1097–1100. DOI: 10.1109/ICCMC51019.2021.9418285.

Gustavsson, Tina Karrbom (2016). "Organizing to avoid project overload: The use and risks of narrowing strategies in multi-project practice". In: *International Journal of Project Management* 34.1, pp. 94–101.

Hans, Erwin W, Herroelen, Willy, Leus, Roel, and Wullink, Gerhard (2007). "A hierarchical approach to multi-project planning under uncertainty". In: *Omega* 35.5, pp. 563–577.

Hansen, Anne-Sofie, Svejvig, Per, and Hansen, Lars K (2022). "Revisiting Shenhar and Dvir's Diamond Model: Do We Need an Upgrade?" In: *Research on Project, Programme and Portfolio Management*. Springer, pp. 175–190.

Hartmann, Sönke and Briskorn, Dirk (2010). "A survey of variants and extensions of the resource-constrained project scheduling problem". In: *European Journal of operational research* 207.1, pp. 1–14.

Hartmann, Sönke and Briskorn, Dirk (2021). "An Updated Survey of Variants and Extensions of the Resource-Constrained Project Scheduling Problem". In: *European Journal of Operational Research*.

Hauder, Viktoria A., Beham, Andreas, Raggl, Sebastian, Parragh, Sophie N., and Affenzeller, Michael (2020). "Resource-constrained multi-project scheduling with activity and time flexibility". In: *Computers & Industrial Engineering* 150, p. 106857. ISSN: 0360-8352. DOI: https://doi.org/10.1016/j.cie.2020.106857.

Hazır, Öncü and Ulusoy, Gündüz (2020). "A classification and review of approaches and methods for modeling uncertainty in projects". In: *International Journal of Production Economics* 223, p. 107522.

Hidalgo, Enric Senabre (2019a). "Adapting the scrum framework for agile project management in science: case study of a distributed research initiative". In: *Heliyon* 5.3, e01447.

Hidalgo, Enric Senabre (2019b). "Adapting the scrum framework for agile project management in science: case study of a distributed research initiative". In: *Heliyon* 5.3, e01447. ISSN: 2405-8440. DOI: 10.1016/j.heliyon.2019.e01447.

Highsmith, James A and Highsmith, Jim (2002). *Agile software development ecosystems*. Addison-Wesley Professional.

Highsmith, Jim (2009). *Agile project management: creating innovative products*. Pearson education.

Hobbs, Brian and Petit, Yvan (2017). "Agile methods on large projects in large organizations". In: *Project Management Journal* 48.3, pp. 3–19.

Hoda, Rashina and Murugesan, Latha K (2016). "Multi-level agile project management challenges: A self-organizing team perspective". In: *Journal of Systems and Software* 117, pp. 245–257.

Homberger, Jörg (2007). "A multi-agent system for the decentralized resource-constrained multi-project scheduling problem". In: *International Transactions in Operational Research* 14.6, pp. 565–589.

Issa, S and Tu, Y (2020). "A survey in the resource-constrained project and multi-project scheduling problems". In: *Journal of Project Management* 5.2, pp. 117–138.

Jacomy, Mathieu, Venturini, Tommaso, Heymann, Sebastien, and Bastian, Mathieu (June 2014). "ForceAtlas2, a Continuous Graph Layout Algorithm for Handy Network Visualization Designed for the Gephi Software". In: *Plos One* 9.6, pp. 1–12. DOI: `10.1371/journal.pone.0098679`.

Jafferali, Mohammed, Venkateshwaran, Jayendran, and Son, Young-Jun (2005). "Performance comparison of search-based simulation optimisation algorithms for operations scheduling". In: *International Journal of Simulation and Process Modelling* 1.1-2, pp. 58–71.

Jerbrant, Anna (2013). "Organising project-based companies: Management, control and execution of project-based industrial operations". In: *International Journal of Managing Projects in Business*.

Johnson, Jim (2020). "Chaos 2020: Beyond Infinity". In: *Standish Group*.

Jonas, Daniel (2010). "Empowering project portfolio managers: How management involvement impacts project portfolio management performance". In: *International journal of project management* 28.8, pp. 818–831.

Karlesky, Michael and Vander Voord, Mark (2008). "Agile project management". In: *ESC* 247.267, p. 4.

Kaufmann, Carsten, Kock, Alexander, and Gemünden, Hans Georg (2020). "Emerging strategy recognition in agile portfolios". In: *International Journal of Project Management* 38.7, pp. 429–440.

Kellenbrink, Carolin and Helber, Stefan (2015). "Scheduling resource-constrained projects with a flexible project structure". In: *European Journal of Operational Research* 246.2, pp. 379–391. ISSN: 0377-2217. DOI: `https://doi.org/10.1016/j.ejor.2015.05.003`.

Kelley Jr, James E (1961). "Critical-path planning and scheduling: Mathematical basis". In: *Operations research* 9.3, pp. 296–320.

Kelley Jr, James E and Walker, Morgan R (1959). "Critical-path planning and scheduling". In: *Papers presented at the December 1-3, 1959, eastern joint IRE-AIEE-ACM computer conference*, pp. 160–173.

Kerzner, Harold (2017). *Project management: a systems approach to planning, scheduling, and controlling*. John Wiley & Sons.

Kleijnen, Jack PC and Wan, Jie (2007). "Optimization of simulated systems: OptQuest and alternatives". In: *Simulation Modelling Practice and Theory* 15.3, pp. 354–362.

Knight, Kenneth (1976). "Matrix organization: a review". In: *Journal of management studies* 13.2, pp. 111–130.

Koenker, R (2005). "Quantile regression Cambridge University Press New York". In:

Koenker, Roger and Hallock, Kevin F (2001). "Quantile regression". In: *Journal of economic perspectives* 15.4, pp. 143–156.

Kolisch, Rainer, Sprecher, Arno, and Drexl, Andreas (1995). “Characterization and generation of a general class of resource-constrained project scheduling problems”. In: *Management Science* 41.10, pp. 1693–1703. DOI: `10.1287/mnsc.41.10.1693`.

Kosztyán, Zs T and Kiss, Judit (2010). “Stochastic network planning method”. In: *Advanced techniques in computing sciences and software engineering*. Springer, pp. 263–268.

Kosztyán, Zsolt T (2012). “Challenges of the project planning methods in the 21st century”. In: *Problems of Management in the 21st Century* 5, pp. 46–60.

Kosztyán, Zsolt T (2015). “Exact algorithm for matrix-based project planning problems”. In: *Expert Systems with Applications* 42.9, pp. 4460–4473.

Kosztyán, Zsolt T (2020). “An Exact Algorithm for the Flexible Multilevel Project Scheduling Problem”. In: *Expert Systems with Applications*, p. 113485.

Kosztyán, Zsolt T, Pribojszki-Németh, Anikó, and Szalkai, István (2019). “Hybrid multimode resource-constrained maintenance project scheduling problem”. In: *Operations Research Perspectives* 6, p. 100129.

Kosztyán, Zsolt T and Szalkai, István (2020). “Multimode resource-constrained project scheduling in flexible projects”. In: *Journal of Global Optimization* 76.1, pp. 211–241.

Kosztyán, Zsolt Tibor, Kiss, Judit, et al. (2010). “PEM–a New Matrix Method for Supporting the Logic Planning of Software Development Projects”. In: *DSM 2010: Proceedings of the 12th International DSM Conference, Cambridge, UK, 22.-23.07. 2010*, pp. 97–110.

Kosztyán, Zsolt Tibor and Kiss, Judit (2011). “Matrix-based project planning methods”. In: *Problems of Management in the 21st Century* 1, p. 67.

Krebs, Jochen (2008). *Agile portfolio management*. Microsoft Press.

Kuprenas, John A (2003). “Implementation and performance of a matrix organization structure”. In: *International Journal of Project Management* 21.1, pp. 51–62.

Kurtulus, IBRAHIMS and Davis, EW (1982). “Multi-project scheduling: Categorization of heuristic rules performance”. In: *Management Science* 28.2, pp. 161–172.

Labro, Eva and Vanhoucke, Mario (2008). “Diversity in resource consumption patterns and robustness of costing systems to errors”. In: *Management Science* 54.10, pp. 1715–1730.

Laguna, Manuel (2011). “OptQuest”. In: *Optimization of Complex Systems*.

Laslo, Zohar and Goldberg, Albert I (2008). “Resource allocation under uncertainty in a multi-project matrix environment: Is organizational conflict inevitable?” In: *International journal of project management* 26.8, pp. 773–788.

Lee, Jihyun and Hur, Sung Jin (2010). “Agile Approach to Manage Projects in Ubiquitous Multi-Project Environment”. In: *2010 Proceedings of the 5th International Conference on Ubiquitous Information Technologies and Applications*. IEEE, pp. 1–5.

Leffingwell, Dean (2010). *Agile software requirements: lean requirements practices for teams, programs, and the enterprise*. Addison-Wesley Professional.

Lenstra, Jan Karel and Rinnooy Kan, AHG (1978). “Complexity of scheduling under precedence constraints”. In: *Operations Research* 26.1, pp. 22–35.

Leybourne, Stephen A (2007). “The changing bias of project management research: A consideration of the literatures and an application of extant theory”. In: *Project Management Journal* 38.1, pp. 61–73.

Liberatore, Matthew J and Pollack-Johnson, Bruce (2003). “Factors influencing the usage and selection of project management software”. In: *IEEE transactions on Engineering Management* 50.2, pp. 164–174.

Litke, HD (2007). *Projektmanagement Methoden, Techniken, Verhaltensweisen. Hanser*.

Liu, DN, Xu, Z, and Li, FF (2019). “Distributed resource constrained multi-project scheduling problem with cooperative-game based negotiation mechanism”. In: *Syst. Eng.-Theory Pract.* 39.6, pp. 1507–1516.

Loiro, Carina, Castro, Hélio, Ávila, Paulo, Cruz-Cunha, Maria Manuela, Putnik, Goran D, and Ferreira, Luís (2019). “Agile project management: A communicational workflow proposal”. In: *Procedia Computer Science* 164, pp. 485–490.

Lova, Antonio, Maroto, Concepción, and Tormos, Pilar (2000). “A multicriteria heuristic method to improve resource allocation in multiproject scheduling”. In: *European journal of operational research* 127.2, pp. 408–424.

Lundin, Rolf A and Söderholm, Anders (1995). “A theory of the temporary organization”. In: *Scandinavian Journal of management* 11.4, pp. 437–455.

MaCurdy, Thomas E and Pencavel, John H (1986). “Testing between competing models of wage and employment determination in unionized markets”. In: *Journal of Political Economy* 94.3, Part 2, S3–S39.

Marchenko, Artem and Abrahamsson, Pekka (2008). “Scrum in a multiproject environment: An ethnographically-inspired case study on the adoption challenges”. In: *Agile 2008 Conference*. IEEE, pp. 15–26.

Maroto, Concepción, Tormos, Pilar, and Lova, Antonio (1999). “The evolution of software quality in project scheduling”. In: *Project scheduling*. Springer, pp. 239–259.

Mastor, Anthony A (1970). “An experimental investigation and comparative evaluation of production line balancing techniques”. In: *Management Science* 16.11, pp. 728–746. DOI: 10.1287/mnsc.16.11.728.

Mathworks, MATLAB (2021). “MATLAB 2021a”. In: *The MathWorks: Natick, MA, USA*.

Mattia, Bianchi, GIacomo, Marzi, and Guerini, Massimiliano (2020). “Agile, Stage-Gate and their combination: Exploring how they relate to performance in software development”. In:

McLain, David (2009). “Quantifying project characteristics related to uncertainty”. In: *Project Management Journal* 40.4, pp. 60–73.

Minogue, P. (Sept. 2011). “"Gantt-Like" DSMs”. In: *DSM 2011: Proceedings of the 13th International DSM Conference*, pp. 259–271. URL: https://www.designsociety.org/publication/30838/%22Gantt-Like%22+DSMs.

Miranda Mota, Caroline Maria de, Almeida, Adiel Teixeira de, and Alencar, Luciana Hazin (2009). "A multiple criteria decision model for assigning priorities to activities in project management". In: *International Journal of Project Management* 27.2, pp. 175–181.

Munns, Andrew K and Bjeirmi, Bassam F (1996). "The role of project management in achieving project success". In: *International journal of project management* 14.2, pp. 81–87.

Olsen, Richard Paul (1971). "Can project management be defined?" In: Project Management Institute.

Owen, Robert, Koskela, Lauri, Henrich, Guilherme, and Codinhoto, Ricardo (2006). "Is agile project management applicable to construction?" In: IGLC.

Özkan, Deniz and Mishra, Alok (2019). "Agile Project Management Tools: A Brief Comprative View". In: *Cybernetics and Information Technologies* 19.4, pp. 17–25. DOI: 10.2478/cait-2019-0033.

Papadakis, Emmanouil and Tsironis, Loukas (2018). "Hybrid methods and practices associated with agile methods, method tailoring and delivery of projects in a non-software context". In: *Procedia computer science* 138, pp. 739–746.

Papadopoulos, Georgios (2015). "Moving from traditional to agile software development methodologies also on large, distributed projects." In: *Procedia-Social and Behavioral Sciences* 175, pp. 455–463.

Papke-Shields, Karen E, Beise, Catherine, and Quan, Jing (2010). "Do project managers practice what they preach, and does it matter to project success?" In: *International journal of project management* 28.7, pp. 650–662.

Patanakul, Peerasit and Milosevic, Dragan (2009). "The effectiveness in managing a group of multiple projects: Factors of influence and measurement criteria". In: *International Journal of Project Management* 27.3, pp. 216–233.

Patterson, James H (1976). "Project scheduling: The effects of problem structure on heuristic performance". In: *Naval Research Logistics Quarterly* 23.1, pp. 95–123. DOI: 10.1002/nav.3800230110.

Payne, John H (1995). "Management of multiple simultaneous projects: a state-of-the-art review". In: *International journal of project management* 13.3, pp. 163–168.

Pellegrinelli, Sergio (1997). "Programme management: organising project-based change". In: *International Journal of Project Management* 15.3, pp. 141–149.

Pellegrinelli, Sergio, Murray-Webster, Ruth, and Turner, Neil (2015). "Facilitating organizational ambidexterity through the complementary use of projects and programs". In: *International Journal of Project Management* 33.1, pp. 153–164.

Pellerin, Robert and Perrier, Nathalie (2019). "A review of methods, techniques and tools for project planning and control". In: *International Journal of Production Research* 57.7, pp. 2160–2178.

Pellerin, Robert, Perrier, Nathalie, and Berthaut, François (2020). "A survey of hybrid metaheuristics for the resource-constrained project scheduling problem". In: *European Journal of Operational Research* 280.2, pp. 395–416.

Peteghem, Vincent Van and Vanhoucke, Mario (2014). "An experimental investigation of metaheuristics for the multi-mode resource-constrained project scheduling problem on new dataset instances". In: *European Journal of Operational Research* 235.1, pp. 62–72. ISSN: 0377-2217. DOI: 10.1016/j.ejor.2013.10.012.

Petit, Yvan and Hobbs, Brian (2010). "Project portfolios in dynamic environments: Sources of uncertainty and sensing mechanisms". In: *Project Management Journal* 41.4, pp. 46–58.

Pinto, Jeffrey K (2013). *Project management: achieving competitive advantage*. s 57. Pearson Boston, MA.

Pinto, Jeffrey K and Slevin, Dennis P (1987). "Critical factors in successful project implementation". In: *IEEE transactions on engineering management*.1, pp. 22–27.

Platje, Adri, Seidel, Harald, and Wadman, Sipke (1994). "Project and portfolio planning cycle: project-based management for the multiproject challenge". In: *International Journal of Project Management* 12.2, pp. 100–106.

PMI (2021). *A Guide to the Project Management Body of Knowledge (PMBOK® guide) – Seventh Edition*. Project Management Institute (PMI).

Pollack, Julien, Helm, Jane, and Adler, Daniel (2018). "What is the Iron Triangle, and how has it changed?" In: *International journal of managing projects in business*.

Pritsker, A Alan B, Waiters, Lawrence J, and Wolfe, Philip M (1969). "Multiproject scheduling with limited resources: A zero-one programming approach". In: *Management science* 16.1, pp. 93–108.

Racheva, Zornitza, Daneva, Maya, and Buglione, Luigi (2008). "Supporting the dynamic reprioritization of requirements in agile development of software products". In: *2008 Second International Workshop on Software Product Management*. IEEE, pp. 49–58.

Radujković, Mladen and Sjekavica, Mariela (2017). "Project management success factors". In: *Procedia engineering* 196, pp. 607–615.

Rajegopal, Shan, McGuin, Philip, and Waller, James (2007). *Project portfolio management: Leading the corporate vision*. Springer.

Rautiainen, Kristian, Schantz, Joachim von, and Vähäniitty, Jarno (2011). "Supporting scaling agile with portfolio management: Case paf. com". In: *2011 44th Hawaii International Conference on System Sciences*. IEEE, pp. 1–10.

Ren, Ran, Zhang, Jiansong, and Jiang, Yi (2021). "New Automated Activity-on-Node Calculation Grading Method for Construction Management Education Innovation". In: *Journal of Civil Engineering Education* 147.3, p. 04021004. DOI: 10.1061/(ASCE)EI.2643-9115.0000043.

Roy, B (1962). "Cheminement et connexité dans les graphes". In: *Applications aux problémes d'ordonnancement. METRA: Série Spéciale*.1.

Sajad, Muhammad, Sadiq, Muhammad, Naveed, Khawar, and Iqbal, Muhammad Shahid (2016). "Software Project Management: Tools assessment, Comparison and suggestions for future development". In: *International Journal of Computer Science and Network Security (IJCSNS)* 16.1, p. 31.

Salameh, Hanadi (2014). "What, when, why, and how? A comparison between agile project management and traditional project management methods". In: *International Journal of Business and Management Review* 2.5, pp. 52–74.

Sánchez, Mariam Gómez, Lalla-Ruiz, Eduardo, Gil, Alejandro Fernández, Castro, Carlos, and Voß, Stefan (2022). "Resource-Constrained Multi-Project Scheduling Problem: A Survey". In: *European Journal of Operational Research*.

Schwaber, Ken and Beedle, Mike (2002). *Agile software development with scrum. Series in agile software development*. Vol. 1. Prentice Hall Upper Saddle River.

Schwindt, Christoph (1995). "ProGen/max: A new problem generator for different resource-constrained project scheduling problems with minimal and maximal time lags". In:

Schwindt, Christoph, Zimmermann, Jürgen, et al. (2015). *Handbook on project management and scheduling vol. 1*. Springer.

Serrador, Pedro and Pinto, Jeffrey K (2015). "Does Agile work?—A quantitative analysis of agile project success". In: *International journal of project management* 33.5, pp. 1040–1051.

Servranckx, Tom and Vanhoucke, Mario (2019a). "A tabu search procedure for the resource-constrained project scheduling problem with alternative subgraphs". In: *European Journal of Operational Research* 273.3, pp. 841–860.

Servranckx, Tom and Vanhoucke, Mario (2019b). "Strategies for project scheduling with alternative subgraphs under uncertainty: similar and dissimilar sets of schedules". In: *European Journal of Operational Research* 279.1, pp. 38–53. DOI: 10.1016/j.ejor.2019.05.023.

Shenhar, Aaron J and Dvir, Dov (2007). *Reinventing project management: the diamond approach to successful growth and innovation*. Harvard Business Review Press.

Shenhar, Aaron J, Dvir, Dov, Lechler, Thomas, and Poli, Michael (2002). "One size does not fit all: True for projects, true for frameworks". In: *Proceedings of PMI research conference*. Project Management Institute, pp. 14–17.

Sohi, Afshin Jalali, Hertogh, Marcel, Bosch-Rekveldt, Marian, and Blom, Rianne (2016). "Does lean & agile project management help coping with project complexity?" In: *Procedia-social and behavioral sciences* 226, pp. 252–259.

Som de Cerff, Wim, Vegte, John van de, Boers, Reinout, Brandsma, Theo, Haij, Marijn de, Moosel, Wim van, Noteboom, Jan Willem, Pagani, Giuliano Andrea, and Schrier, Gerard van der (2018). "Agile development in meteorological R&D: achieving a minimum viable product in a scrum work setting". In: *Bulletin of the American Meteorological Society* 99.12, pp. 2507–2518.

Song, Wen, Xi, Hui, Kang, Donghun, and Zhang, Jie (2018). "An agent-based simulation system for multi-project scheduling under uncertainty". In: *Simulation Modelling Practice and Theory* 86, pp. 187–203.

Sprecher, A and Kolisch, R (1996). "PSPLIB—a project scheduling problem library". In: *European Journal of Operational Research* 96, pp. 205–216. DOI: 10.1016/S0377-2217(96)00170-1.

Sprecher, Arno (1994). "Generation of Instances by ProGen". In: *Resource-Constrained Project Scheduling*. Springer, pp. 70–90. DOI: 10.1007/978-3-642-48397-4_6. URL: https://doi.org/10.1007/978-3-642-48397-4_6.

Spuhler, RW and Biagini, RG (1990). "The role and weaknesses of top management in internal projects". In: *Handbook of management by projects. Vienna: Manzsche Verlag*.

Špundak, Mario (2014). "Mixed agile/traditional project management methodology–reality or illusion?" In: *Procedia-Social and Behavioral Sciences* 119, pp. 939–948.

Srivastava, Abhishek, Mehrotra, Deepti, Kapur, PK, and Aggarwal, Anu G (2021). "Measuring and Evaluating Best Practices in Agile Testing Environment Using AHP". In: *Advances in Interdisciplinary Research in Engineering and Business Management*. Springer, pp. 475–495. DOI: 10.1007/978-981-16-0037-1_37. URL: https://doi.org/10.1007/978-981-16-0037-1_37.

Stapleton, Jennifer (1997). *DSDM, dynamic systems development method: the method in practice*. Cambridge University Press.

Stare, Aljaž (2014). "Agile Project Management in Product Development Projects". In: *Procedia - Social and Behavioral Sciences* 119, pp. 295–304. ISSN: 1877-0428. DOI: 10.1016/j.sbspro.2014.03.034.

Stettina, Christoph J and Smit, Mark NW (2016). "Team portfolio scrum: an action research on multitasking in multi-project scrum teams". In: *International Conference on Agile Software Development*. Springer, Cham, pp. 79–91.

Stettina, Christoph Johann and Hörz, Jeannette (2015). "Agile portfolio management: An empirical perspective on the practice in use". In: *International Journal of Project Management* 33.1, pp. 140–152.

Steward, D (1981). "The design structure matrix: A method for managing the deisgn of complex systems". In: *IEEE Transactions on Engineering Management* 28.1981, pp.

Strode, Diane E (2016). "A dependency taxonomy for agile software development projects". In: *Information Systems Frontiers* 18.1, pp. 23–46.

Strode, Diane E, Hope, Beverley, Huff, Sid, and Link, Sebastian (2011). "Coordination effectiveness in an agile software development context". In:

Sweetman, Roger and Conboy, Kieran (2013). "Exploring the tensions between software project portfolio management and agile methods: A research in progress paper". In: *International Conference on Lean Enterprise Software and Systems*. Springer, pp. 210–217.

Sweetman, Roger and Conboy, Kieran (2018). "Portfolios of agile projects: A complex adaptive systems' agent perspective". In: *Project Management Journal* 49.6, pp. 18–38.

Sydow, Jörg, Lindkvist, Lars, and DeFillippi, Robert (2004). *Project-based organizations, embeddedness and repositories of knowledge*.

Szűcs, István (2000). "Projektmenedzsment a vezetés szolgálatában". In: *Vezetéstudomány-Management and Business Journal* 31.1, pp. 56–62.

Tang, Dunbing, Zhu, Renmiao, Tang, Jicheng, Xu, Ronghua, and He, Rui (2010). "Product design knowledge management based on design structure matrix". In: *Advanced Engineering Informatics* 24.2, pp. 159–166.

Tao, Sha and Dong, Zhijie Sasha (2018). "Multi-mode resource-constrained project scheduling problem with alternative project structures". In: *Computers & Industrial Engineering* 125, pp. 333–347. ISSN: 0360-8352. DOI: `https://doi.org/10.1016/j.cie.2018.08.027`.

Tavares, L Valadares (1999). *Advanced models for project management*. Vol. 16. Springer Science & Business Media. DOI: `10.1007/978-1-4419-8626-9`. URL: `https://doi.org/10.1007/978-1-4419-8626-9`.

Thesing, Theo, Feldmann, Carsten, and Burchardt, Martin (2021). "Agile versus waterfall project management: decision model for selecting the appropriate approach to a project". In: *Procedia Computer Science* 181, pp. 746–756.

Thiry, Michel and Deguire, Manon (2007). "Recent developments in project-based organisations". In: *International journal of project management* 25.7, pp. 649–658.

Thulasiraman, Krishnaiyan and Swamy, Madisetti NS (2011). *Graphs: theory and algorithms*. John Wiley & Sons.

Traag, V. A., Waltman, L., and Eck, N. J. van (Mar. 26, 2019). "From Louvain to Leiden: guaranteeing well-connected communities". In: *Scientific Reports* 9.1, p. 5233. ISSN: 2045-2322. DOI: `10.1038/s41598-019-41695-z`.

Turek, Michał and Werewka, Jan (2016). "Multi-project Scrum methodology for projects using software product lines". In: *Information Systems Architecture and Technology: Proceedings of 36th International Conference on Information Systems Architecture and Technology–ISAT 2015–Part III*. Springer, pp. 189–199.

Turner, J Rodney (1993). *Handbook of project-based management: Leading strategic change in organizations*. McGraw-Hill Book Company, London.

Turner, J Rodney (2009). *Handbook of project-based management: Leading strategic change in organizations*. McGraw-Hill Education.

Turner, J Rodney and Müller, Ralf (2003). "On the nature of the project as a temporary organization". In: *International journal of project management* 21.1, pp. 1–8.

Turner, Rodney J, Huemann, Martina, Anbari, Frank T, and Bredillet, Christophe N (2010). *Perspectives on projects*. Routledge.

Uludag, Ömer, Kleehaus, Martin, Caprano, Christoph, and Matthes, Florian (2018). "Identifying and structuring challenges in large-scale agile development based on a structured literature review". In: *2018 IEEE 22nd International Enterprise Distributed Object Computing Conference (EDOC)*. IEEE, pp. 191–197.

Vähäniitty, Jarno et al. (2012). "Towards agile product and portfolio management". In:

Van Eynde, Rob and Vanhoucke, Mario (2020). "Resource-constrained multi-project scheduling: benchmark datasets and decoupled scheduling". In: *Journal of Scheduling* 23.3, pp. 301–325.

Van Wyngaard, C Jurie, Pretorius, Jan-Harm C, and Pretorius, Leon (2012). "Theory of the triple constraint—A conceptual review". In: *2012 IEEE International Conference on Industrial Engineering and Engineering Management*. IEEE, pp. 1991–1997.

Vanhoucke, Mario (2010a). "A scatter search heuristic for maximising the net present value of a resource-constrained project with fixed activity cash flows". In: *International Journal of Production Research* 48.7, pp. 1983–2001. DOI: 10.1080/00207540802010781.

Vanhoucke, Mario (2010b). "Using activity sensitivity and network topology information to monitor project time performance". In: *Omega* 38.5, pp. 359–370. DOI: 10.1016/j.omega.2009.10.001.

Vanhoucke, Mario (2018). "Planning projects with scarce resources: Yesterday, today and tomorrow's research challenges". In: *Frontiers of Engineering Management* 5.2, pp. 133–149.

Vanhoucke, Mario and Coelho, José (2018). "A tool to test and validate algorithms for the resource-constrained project scheduling problem". In: *Computers & Industrial Engineering* 118, pp. 251–265. DOI: 10.1016/j.cie.2018.02.001.

Vanhoucke, Mario, Coelho, José, and Batselier, Jordy (2016). "An overview of project data for integrated project management and control". In: *Journal of Modern Project Management* 3.3, pp. 6–21.

Vanhoucke, Mario, Coelho, José, Debels, Dieter, Maenhout, Broos, and Tavares, Luís V (2008). "An evaluation of the adequacy of project network generators with systematically sampled networks". In: *European Journal of Operational Research* 187.2, pp. 511–524. DOI: doi.org/10.1016/j.ejor.2007.03.032.

Vanhoucke, Mario, Demeulemeester, Erik L, and Herroelen, Willy (2001). "On maximizing the net present value of a project under renewable resource constraints". In: *Management Science* 47.8, pp. 1113–1121. DOI: 10.1287/mnsc.47.8.1113.10226.

Vázquez, E Pérez, Calvo, M Posada, and Ordóñez, P Martín (2015). "Learning process on priority rules to solve the RCMPSP". In: *Journal of Intelligent Manufacturing* 26.1, pp. 123–138. DOI: 10.1007/s10845-013-0767-5.

Vuorinen, Lauri and Martinsuo, Miia (2018). "Program integration in multi-project change programs: agency in integration practice". In: *International Journal of Project Management* 36.4, pp. 583–599.

Wauters, Tony, Kinable, Joris, Smet, Pieter, Vancroonenburg, Wim, Vanden Berghe, Greet, and Verstichel, Jannes (2016). "The multi-mode resource-constrained multi-project scheduling problem". In: *Journal of Scheduling* 19.3, pp. 271–283. DOI: 10.1007/s10951-014-0402-0.

Westland, Jason (2009). *Project management guidebook*. Method123.

Wiest, Jerome D (1981). "Precedence diagramming method: Some unusual characteristics and their implications for project managers". In: *Journal of Operations management* 1.3, pp. 121–130.

Williams, Laurie (2010). "Agile Software Development Methodologies and Practices". In: *Advances in Computers*. Ed. by Marvin V. Zelkowitz. Vol. 80. Advances in Computers. Elsevier, pp. 1–44. DOI: https://doi.org/10.1016/S0065-2458(10)80001-4. URL: https://doi.org/10.1016/S0065-2458(10)80001-4.

Williams, Paul, Ashill, Nicholas J, Naumann, Earl, and Jackson, Eric (2015). "Relationship quality and satisfaction: Customer-perceived success factors for on-time projects". In: *International Journal of Project Management* 33.8, pp. 1836–1850.

Williams, Terry (2005). "Assessing and moving on from the dominant project management discourse in the light of project overruns". In: *IEEE Transactions on engineering management* 52.4, pp. 497–508.

Wińska, Ewelina and Dąbrowski, Włodzimierz (2020). "Software development artifacts in large agile organizations: a comparison of scaling agile methods". In: *Data-Centric Business and Applications*. Springer, pp. 101–116.

Wohlin, Claes, Runeson, Per, Höst, Martin, Ohlsson, Magnus C, Regnell, Björn, and Wesslén, Anders (2012). *Experimentation in software engineering*. Springer Science & Business Media.

World Bank (2012). *The little data book on financial inclusion 2012*. World Bank Publications. ISBN: 9780821395097. DOI: 10.1596/978-0-8213-9509-7. URL: https://doi.org/10.1596/978-0-8213-9509-7.

Wysocki, Robert K (2011). *Effective project management: traditional, agile, extreme*. John Wiley & Sons.

Wysocki, Robert K (2019). *Effective Project Management: Traditional, Agile, Extreme, Hybrid*.

Yasaman, Arefazar, Nazari, Ahad, Hafezi, Mohammad Reza, and Maghool, Sayyed Amir Hossain (2022). "Prioritizing agile project management strategies as a change management tool in construction projects". In: *International Journal of Construction Management* 22.4, pp. 678–689. DOI: 10.1080/15623599.2019.1644757.

Yin, Robert K (2009). *Case study research: Design and methods*. Vol. 5. sage.

Zhu, Guidong, Bard, Jonathan F, and Yu, Gang (2005). "Disruption management for resource-constrained project scheduling". In: *Journal of the Operational Research Society* 56.4, pp. 365–381.

Zika-Viktorsson, Annika, Sundström, Per, and Engwall, Mats (2006). "Project overload: An exploratory study of work and management in multi-project settings". In: *International Journal of Project Management* 24.5, pp. 385–394.

Zwikael, Ofer (2009). "Critical planning processes in construction projects". In: *Construction innovation*.