THESIS SUMMARY

Bridging theory and practice: simulation-based scheduling performance evaluations for Application Lifecycle Management

Author:
Róbert JAKAB

Supervisor:
Prof. Dr. habil. Zsolt Tibor KOSZTYÁN

UNIVERSITY OF PANNONIA

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

August 2024

Contents

1 Introduction

Over the past decades, software has become an essential enabler for science and the economy. The evolution of software application development and maintenance handling has become a vital domain both in academia and in business practice. In the SW development management from the one-time linear development approach, the focus moved to agile, flexible content handling with frequent SW upgrade approaches. Several vendors are providing tools and toolsets supporting lifecycle development for such a concept of software; however, the related academic literature is still scarce in the area of clear ALM definition, applicable methodologies, and methods.

1.1 The importance of Application Lifecycle Management

Application Lifecycle Management (ALM) has emerged as a crucial focal point for enterprises across numerous industries (IT, automotive, aviation, and so on) aiming to provide a solution for managing application software through complete lifecycle, from their inception till their retirement. It was visible in the last few decades an unparalleled transformation towards a software-centric economy, businesses of all sizes are leveraging software applications to innovate, streamline processes, and give value to customers. Consequently, the economic impact of software creation and maintenance has expanded dramatically (Mishra and Alzoubi, 2023; Al-Saqqa et al., 2020; van den Ende and van Marrewijk, 2014).

Thus the failure of software projects, including applications also, can be economically disastrous. Delays, budget overruns, and sub-optimal software quality are not only costly but also erode market competitiveness. The proper management of the ALM is essential for mitigating these risks and maximizing the economic efficiency of software development challenges, as the ALM governs the entire software application journey and plays a pivotal role in ensuring that investments in software translate into sustainable economic returns. Therefore, applying efficiency in application management, such as proper scheduling methodology in an ALM environment, is crucial.

Although, as there is no clear and commonly agreed definition for ALM existing in the academic literature, so vendors were taking the chance to alter it to their benefit. As a result, it is also hindering the next level of scientific analyses and developments. Lacking this mutual understanding, companies are deemed to use available project management methodologies like traditional- (TPM), agile- (APM), and hybrid project management (HPM), which might be fitting partially only for their purposes.

1.2 Research questions

Therefore, the aim of the dissertation was threefold:

- ✓ Research ALM scientific literature for
  - [+] definition and scope identification,
  - [+] enabling definition determination for methodological research,
- ✓ To confirm the applicability of Matrix representation for scheduling investigation, including:
  - [+] simulation (artificial) environment setup,
  - [+] TPM, APM and HPM feasibility check,
  - [+] TPM, APM and HPM scheduling efficiency analysis.
- ✓ To examine the effects of risk factors on the IT project's structure for scheduling.

As validation, a relevant ALM case study with scheduling performance evaluation to be conducted as well.

Considering the goals above, the current study seeks to answer the following research questions:

RQ1: How can a planning model be identified based on available scientific literature definitions that represents the Application Lifecycle Management (ALM) scheduling problem?

RQ2: Do the existing project management scheduling methodologies (TPM, APM, HPM) produce feasible solutions in the ALM environment, and how are they performing?

RQ3: What are the risk factors for the scheduling problem in the ALM environment, and how are they influencing the feasibility and scheduling performance?

2 Related studies and research assumptions

This chapter provides a concise overview of the essential topics that are required to comprehend the specific methodological study of ALM upcoming in this research. Familiarity with these concepts is essential for understanding and analyzing the methodology and findings of the research.

2.1 Application Lifecycle Management

An application is a specific type of software designed to perform a particular function or set of related functions for end-users. Applications are user-facing and serve specific purposes. In academic contexts, "application" is a subset of software, specifically referring to programs developed to address user needs in various domains. Hereafter, this broader meaning of application is understood for the ALM context.

The performance of the SW projects historically is worse compared to the traditional projects, for example, construction, and the phenomenon was already noticed in the nineties in the Chaos report (StandishGroup, 2020). Several SW development-specific factors are influencing this under-performance, which are not present in other product development-related projects, like complex design processes, the importance of clear requirements availability, specific collaboration methods and tools, and so on identified in detail by the study.

The general approach for product lifecycle management was realized with the PLM (Product Lifecycle Management), and in the beginning, software was just a subpart of it, however, with growing SW complexity, the handling of it called for specialization, first called software development lifecycle (SDLC) management, later application lifecycle management (ALM). Despite this unique IT area's apparent importance, the research community usually leaves it to the PLM and ALM tool vendors, who are ultimately promoting the convergence (Deuter, Otte, et al., 2019; Deuter and Rizzo, 2016; Rao and Palaniappan, 2020). There are still connections, though, for some of the PLM and ALM intersections for the project and program management elements, interlinks between HW and SW products, change management, collaboration, and reports (Deuter and Rizzo, 2016). This also shows that the tremendous number of smart devices, for example, in ICT or even automotive, are challenging the ALM. The case study in this dissertation shows how the ALM is realized in an automotive supplier company and what challenges exist in the day-to-day business.

The domain of Application Lifecycle Management is claimed by many scholars (Chappell et al., 2010; Corallo et al., 2020; Rossberg, 2019; Rossman, 2010) to be a comprehensive software engineering approach that encompasses the entire lifespan of a software application from its initial concept, through development and deployment, to its ultimate retirement. During this lifecycle several aspects are present, however, what makes ALM unique compared to a traditional project management or service approach, is that in the Development perspective after the main phase (first green line) there are potentially additional non-expected tasks and even subprojects appearing (smaller green lines) in a scattered way, as you can see in Figure 1. below. This kind of approach is unique for application development, that during the Operations phase, there are several planned or unplanned, recurring changes happening.

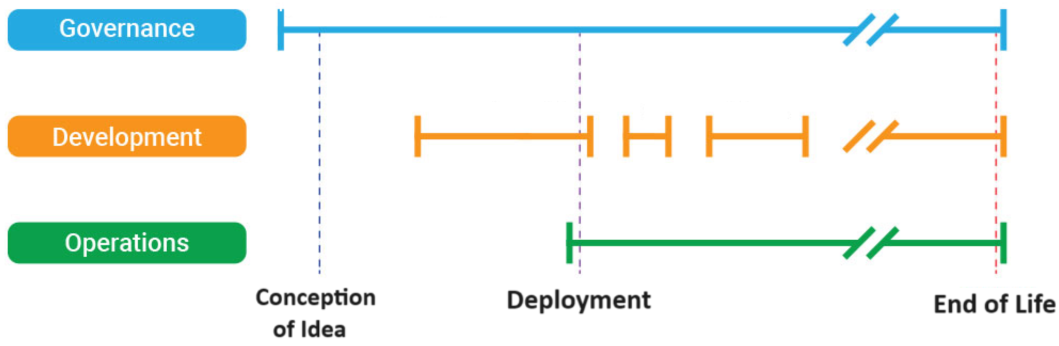


FIGURE 1: ALM process by Chappell et al. (2010)

Due to the fact that the changes are more prevalent in a SW development, the classical project management process models, like the waterfall development model that expects a clear scope definition already at the beginning of the project with no or minimal changes, are challenged by such kind of SW development activities. Thus from a methodology point of view, more flexible approaches are desirable to resolve the situation, like the agile-(SGI, 2019) or partial traditional and agile combinations, so-called hybrid project management approaches are expected in such an ALM environment, however, there is a tremendous gap in the scientific area how to identify this problem context, and also then how to handle exactly such situations efficiently.

In summary, it is visible that the ALM literature in academia is still scarce, and a thorough examination is necessary to find a proper scope definition. Such a review is possible with a systematic literature review.

2.2 Project Management Approaches

For the scheduling performance examination of this research, different project management approaches' behaviors were compared in the simulation. The three selected approach are reflecting the most commonly used methodologies in the software development area, and they are providing assumably significantly different results also, which is why they were selected here. Need to notice, that further project management approaches exists for software development, however, those are not considered here in this context.

In the *traditional project management* (TPM) approach (for example, a construction project or software development project that follows a waterfall process model), the question is how much the realization of the predefined requirements realization will cost. Therefore, while the scope is given and has to be completed, the time, cost, and quality are convertible if necessary. This approach allows more than one completion mode (technologies that require different time/cost/resource demands) (Creemers, 2015)

In the *agile project management* (APM) approach, the question is how many of the features can be included within the given budget and time interval (for example in a sprint). While within a sprint the content is fixed, so new tasks are not allowed, between the sprints in the planning period rearrangements and new tasks can appear. The overall goal is for all the approaches to realize the scope to the highest possible degree. (Rasnacis and Berzisa, 2017; SGI, 2019)

In the *hybrid project management* (HPM) the traditional and agile methods are combined, enabling new activities can appear and be involved anytime, and also capable of handling multiple projects at once. (Reiff and Schlegel, 2022)

TABLE 1: Comparison of various traditional and flexible project management approaches

Approaches	Project Structure	New Tasks	Multiple modes
Traditional (TPMa)	Fixed	Not Allowed	Handled
Agile (APMa)	Flexible	Not Allowed	Not Handled
Hybrid (HPMa)	Flexible	Allowed	Handled

The nature of agile and hybrid projects, such as involving customers in the development process, ensuring strong executive support, and providing the ability to cope with emergent requirements, requires adaptive and flexible thinking for project management as well. In the agile project management approach, the completion of the project is more flexible, and the project structure can adapt to the changing customer requirements. (Dingsøyr et al., 2012).

The feasibility and efficiency comparison of the above-described management approaches in the ALM context has not yet been evaluated, thus creating such a comparison would greatly contribute to the literature and also to the practical implications point of view to business stakeholders.

2.3 Resource-constrained project scheduling problem

The use of mathematical modeling of economic processes has significantly contributed to the development of methodological tools in recent history. Since the 1950s, the resource-constrained project scheduling problem (RCPSP) has been extensively studied in the field of project planning. This classical problem involves scheduling a set of activities, taking into account both precedence and resource constraints, to optimize an objective function such as minimizing the overall project duration or overall costs. Over the years, numerous researchers have devised exact and heuristic solutions for this problem (see Moukrim et al. (2015), Kreter et al. (2018), Tritschler et al. (2017), Abdolshah (2014), Demeulemeester and Herroelen (2006)), and they have also explored various approaches and extensions. In their work, Hartmann and Briskorn (2021) offer a comprehensive overview and classification of the most significant extensions of the RCPSP.

However, for the specific case of ALM, there is not yet available any quantitative methodological analysis and evaluation for the scheduling performance. Thus examining the application of such scheduling feasibility and performance evaluation is reasonable as the first step for this dissertation, which can open up the research for even a potential ALM-specific extension for the RCPSP.

The ALM-related risk evaluation can be particularly important and exciting for academics, business stakeholders, and field experts, as risk realization is usually connected to negative effects during the lifecycle. Therefore the more prepared the better-handled rule is valid, which means recognizing and mitigating the risks in the early phase potentially can have fewer or smaller effects on the plan. As application lifecycle management is a specific field even in the IT area, there are several factors whose relevance need to be examined, and also, the ALM-specific risks need identification. This research provides a summary of the comparison between project-specific and ALM-specific findings. The main points are identified and described, which were found during the ALM area examination in the literature review in Table 4 in Paragraph 4.

2.4 Research assumptions

By assessing the research questions after following up a thorough examination of the literature, it became feasible to develop the related research assumptions. There are three research assumptions:

RA1: A planning model can be identified based on the unified ALM definitions from the scientific literature.

RA2: The TPM, APM, HPM project management approaches using the matrix-based planning method can be extended to solve the scheduling problem, and result in feasible solutions with different results in the ALM environment. A simulation framework can be constructed to handle flexible dependencies and non-planned tasks.

RA3: There are existing project-related risk factors that can be extended for ALM scheduling problems, however, due to the differences between project and ALM scope, ALM-specific risks appear also, which can have an effect on resources, cost, and timing, and can influence the feasibility and scheduling performance.

After the research assumptions are shown in the next chapter, the related works are presented.

3 Results

In this chapter after the presentation of the research question, I present the research completed related to them, with the decision about the research assumptions.

RQ1: *How can a planning model be identified based on available scientific literature definitions that represents the Application Lifecycle Management (ALM) scheduling problem?*

The research addresses the gap in the definition of ALM by proceeding with a rigorous systematic literature review of the available ALM-specific scientific literature. The PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) method was used to support the objective of the research. (Page et al., 2021) Including findings for the keyword search "ALM definition" in the English language from the Google Scholar search engine. A critical analysis then proceeded to explore the definitions of existing application lifecycle management (ALM). This study synthesizes an analytic framework from a total of 3230 sources, which contain explicit information. Since ALM is also strongly vendor-driven and relies heavily on non-referenced sources, therefore this study includes findings limited to peer-reviewed sources such as journal articles, conference proceedings, books and book chapters, PhD dissertations.

The screening overview for the yearly distribution can be seen in Figure 2. The figure demonstrates well that the ALM area overall is currently a small, developing area in the academic literature, the dominant number of conference papers an articles show that intense discussions are ongoing within the field.

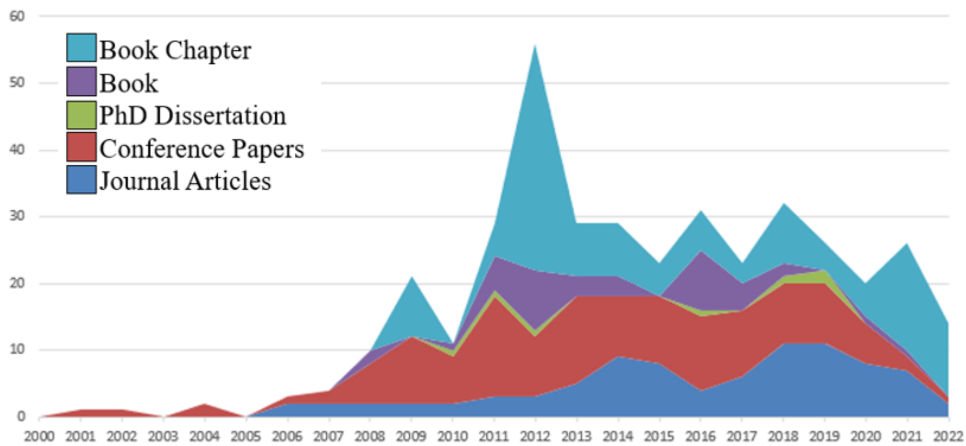


FIGURE 2: Articles, Conference papers, Dissertations, Books and Chapters yearly distribution over the years after the Screening

After analyzing the sources from a quality perspective, it was apparent, that there are visible quality differences among the sources, thus two main categories were created. The *Top Academic* entries, which are the upper half of the scientific journals according to the Scimago ranking (Q1 & Q2), and for Conferences based on Qualis ranking top tier (A1 & A2 & B1). The rest, still peer-reviewed quality sources are belonging to the *Extended Academic* ranking in this overview, including articles from Journals Q3 & Q4, conference papers, PhD dissertations, book and book chapters.

Table 2 shows that after the filtering and full reading the sources contain only very limited information about the ALM definition explicitly, a total 76 relevant sources explicit ALM definition. This is quite a low number of appearances, and in several cases, scholars are highlighting the connection with the business-related strong connections and vendor-specific ALM definition changes as part of the evolution of their toolsets.

The exciting finding is that while in the Top Academic sources, the most common appearance was for Definition E), which refers to ALM *as a paradigm that contains governance, development, operation/ maintenance*, that enhances the level of abstraction in ALM.

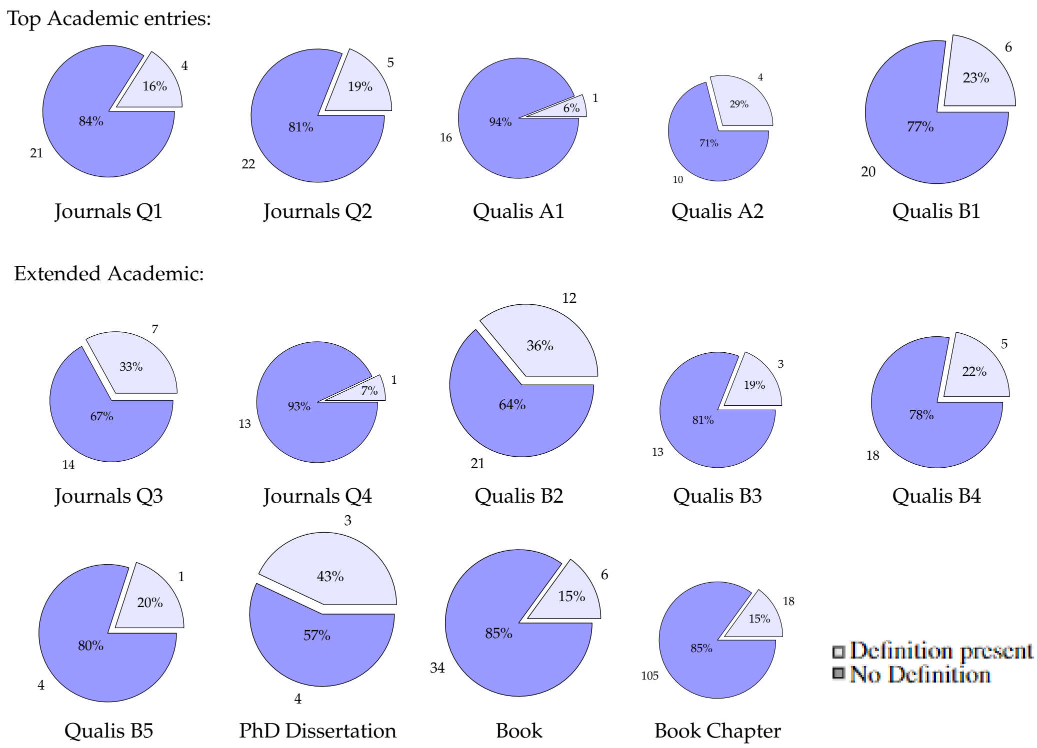


TABLE 2: Definition availability in Top and Extended Academic ranking sources (with percentage and total numbers) - Own edit

In the Extended Academic ranking sources Definition C) was the most prevalent, referring to *ALM as a framework for the coordination of activities* (including requirements, modeling, development, build and testing) *and artifact management* (enforcement of the processes for interconnecting activities; management of relationships and links between the development artifacts; and reporting on the progress of the development) during the SW lifecycle. See Figure 3 for the distribution of the found definitions.

The ALM definition findings after the systematic literature research are summarized in the Table 3 which includes the

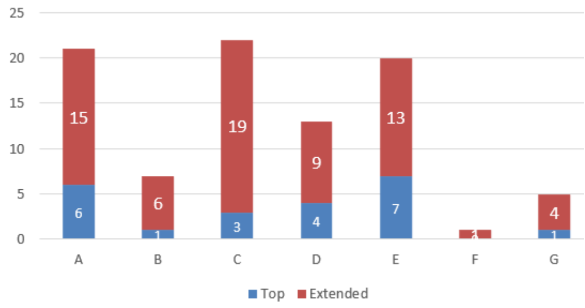


FIGURE 3: Summary of various definition scoped present in included entries

identified ALM definition scope below with the key highlight and the left side and the main sources with references on the right side of the tabular.

Scope: ALM is...	Definition and referenced authors
A) a process for SW PLM/SDLC	Product life-cycle Management (PLM) and its equivalent in software, namely application life-cycle management (ALM), are the overall business process that governs a product or service from its inception to the end of its life in order to achieve the best possible value for the business of the enterprise and its customers and partners. PLM/ALM combines processes, people, and tools for the effective engineering of products —from their inception until the end of service. It involves tacit knowledge of experts and explicit knowledge, codified in procedures, process, and tools. PLM/ALM stretches from know-how to know-what and know-why. (Lachainer, 2011), (Ebert, 2013), (Gatrell, 2016)
B) SW development AND maintenance.	Application Lifecycle Management (ALM), a widely-used lifecycle for software development and maintenance. (J. Rossberg, 2012), (Ramler, 2012)
C) Artefact management tool for SDLC.	ALM “has emerged to indicate the coordination of activities and the management of artefacts (e.g., requirements, source code, test cases) during the software product?s lifecycle” (Kääriäinen and Välimäki, 2009), Gatrell (2016) The coordination of development lifecycle activities, including requirements, modeling, development, build and testing, through the following: 1. enforcement of the processes that interconnect these activities; 2. management of relationships and links between the development artefacts used or generated by these activities; and 3. reporting on the progress of the development effort as a whole. ALM is often seen as a framework that aims at synchronizing all the lifecycle activities instead of focusing on any specific lifecycle activity” (Schwaber 2006)
D) an SDLC extended with phases after development.	ALM is the product lifecycle management of computer programs that is a wider approach than the SDLC, which is limited to the phases of the typical software development stages. In contrast, ALM defines stages after the development lifecycle as well. (OGC ITIL, 2002),(Arya, 2011), (Chappell, 2011)
E) a paradigm: governance, development, operation/maintenance.	Application Lifecycle Management (ALM) is a recent paradigm for integrating and managing the various activities related to the governance, development and maintenance of software products. ALM is a combination of three functions: governance, development and operations, and three milestones: (start of) ideation, deployment and end-of life. (Chappell, 2008) (Rossberg, 2014)
F) ALM is a service for after development part only	application management refers to the lifecycle-oriented control of the problem resolution process for operational application systems excluding any fundamental application development services. (Kueper, 2011)
G) ALM for quality assurance	Establishing a standardized development-to-release workflow, often referred to as the ALM process, is particularly critical for organizations in their efforts to meet tough IT compliance mandates. (Tracy, 2006)

TABLE 3: Summary table of ALM definitions and their scopes

Additionally, after the systematic literature review, a critical review proceeded to look for the establishment of a unified definition. This approach provides a combined and refined synthesis of definitions created to provide a base for methodological research.

In the following the elements of this unified ALM definition are described with the keywords in the head of the line:

1. *Scope of ALM.* ALM is a holistic approach to managing software applications throughout their entire lifecycle, from inception to retirement. It is realized by integrating and managing various activities and workproducts related to 3 functions, namely, governance, development, and operations, including maintenance. Governance is an overarching management activity throughout the whole lifetime of an ALM; however, its importance is greater upstream due to its influencing factors. Development is mostly related to classical SW development projects, which involve main RD-related work. Operations and maintenance are rather similar to service; however, because additional development tasks might occur at different sizes in this phase next to the bug fixing, these tasks may be unique.

2. *Phases of ALM.* There are 3 primary milestones for ALM: ideation, deployment, and end-of-life. There are 7 phases, which include requirement gathering, design, development, testing, deployment, maintenance, and decommissioning.

3. *Key Components.* The core components of ALM support the lifecycle through processes and tools such as version control, issue tracking, continuous integration, and deployment automation. These components play a crucial role in scheduling and resource allocation.

4. *Scheduling Challenges.* There are specific challenges associated with scheduling in ALM. These include the following: resource allocation, as activity realization is also mostly linked to finite resources; task sequencing, as the scheduling of activities in the development and maintenance phases might need to be handled differently; time estimation for resources based on scheduling methodologies might be difficult and not straightforward, and optimizing resource utilization and scheduling, as currently existing methodologies are not yet proven to be optimal for ALM specifics.

5. *ALM development methodologies.* The ALM is tightly integrated with the software development process. For scheduling within ALM, flexible SW development methodologies such as Agile, partially or fully applied as a hybrid approach, should be considered. However, there are currently no specific ALM-related methodologies or frameworks proven to be optimal.

6. *Flexibility and Adaptability.* ALM scheduling methodologies should be flexible and adaptable to accommodate changing requirements, unexpected issues, and evolving project priorities. The process of handling and managing changes occurs not only during development but also during the operation maintenance phase. These additional change requests can extend from the task level up to even smaller subproject levels.

7. *Measurement and Metrics.* Measuring and tracking key performance indicators (KPIs) related to scheduling in ALM is also crucial due to the flexible structure handling and necessary contracted values to be contacted. These metrics include project duration, resource utilization, and task completion rates.

So as a summary, the ALM definition unified contains the extended definition of time, cost, resources, and quality. The time perspective is covered by the scope and phases, stating ALM is from inception to retirement of the software application, which is divided into 7 phases: requirements gathering, design, development, testing, deployment, maintenance, and decommissioning. The resources are the renewable and non-renewable resources necessary to implement the activities in the scope defined. which allocation is handled by the key components along the ALM development methodologies in a flexible scheduling. The quality parameters are determined by the Measurement and Metrics for observing the readiness and maturity levels.

The research question **RQ1** was focused on available ALM definitions and their characteristics and on how a common definition can provide a strong basis for future research. The extracted definitions and their summaries are revealed and shown, reflecting a total of 7 different interpretations. The definitions cover the current academic understanding of the application life cycle management scopes and contents, and the unified ALM definition with its details provides a base for future methodological research as well. As a result, **RA1** is accepted.

RQ2: *Do the existing project management scheduling methodologies (TPM, APM, HPM) produce feasible solutions in the ALM environment, and how are they performing?*

For the research of this question, the Matrix-based representation method was used for the formerly presented project management approaches (TPM, APM, HPM) with the help of a Monte Carlo simulation. This made it possible to have a measurable output for the comparable analysis. The details are described in this section.

Agent-based implementations for project management approaches

For TPM, APM, and HPM, respectively, an agent is programmed to execute the schedule problem solving. Respective, they are called Traditional Project Management agents (TPMa); for the Agile approach, the agent is APMa, and for the Hybrid approach HPMa, see Figure 4 for comparison of various agent's behavior for scheduling problems.

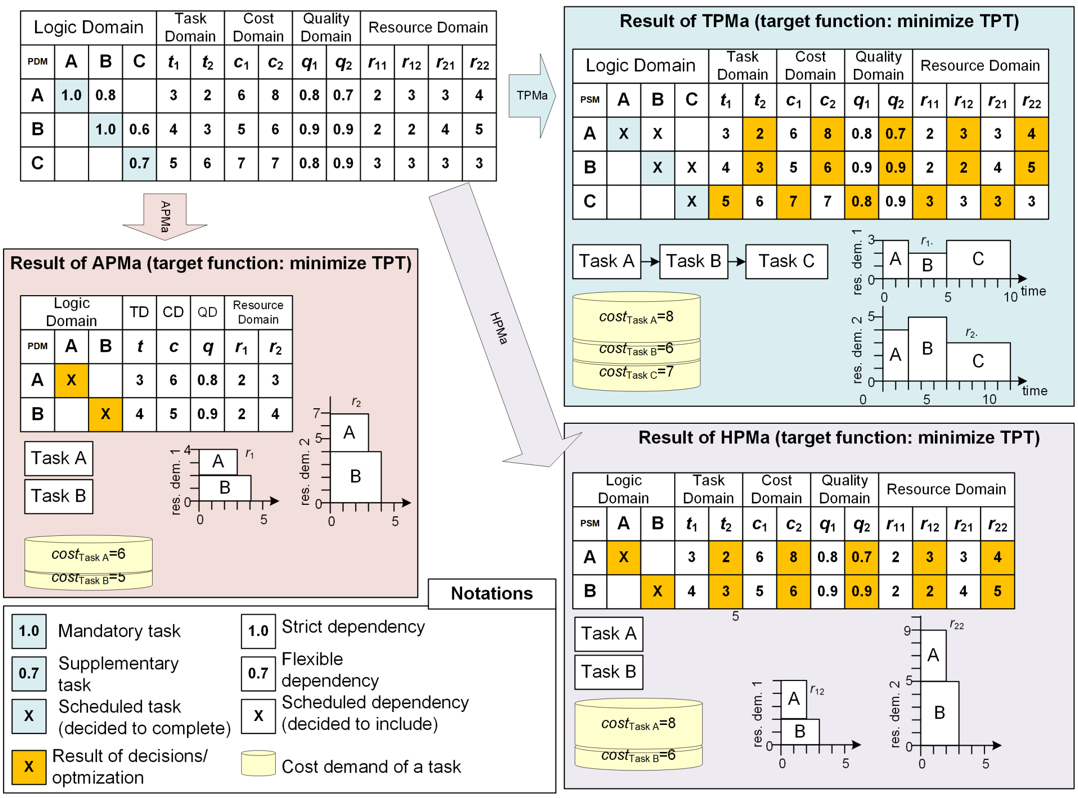


FIGURE 4: Comparison of project management approaches and their agent-based implementations when the target function is the minimal total project time. (t_j, c_j, q_j represent time/cost demands/quality parameters, respectively, of completion mode j, r_{ij} is the resource i of completion mode j.)

Working in a structure called project domain matrix (PDM) which has 3 mandatory domains, namely, logic domain (LD), time domain (TD), and cost domain (CD), and two supplementary domains, namely, quality domain (QD) and resource domain (RD) see Figure 4 for example (Kosztyán, 2022).

Simulation framework

Determining the feasibility of the scheduling problems with the various agents (TPMa, APMa, HPMa) a simulation was created that focused on the solution of the problem. In case the problem was feasible, means the appearing risks were not hindering the solution, it "survived" this challenging, risky environment. Therefore, the Monte Carlo simulation (MCS) was selected, one of the most frequently applied risk management methods. This is a useful technique to simulate project risks and uncertainties. In MCS, risk effects, such as delays, cost overruns, and overwork, can be simulated by changing the time/cost/resource demands of the tasks (Kwak and Ingall, 2007). In MCS, task demands follow theoretical or empirical distributions. By combining MCS with matrix-based techniques, the interdependencies of the risks can also be modeled. In the case of flexible project structures, the project can be restructured (Kosztyán and Szalkai, 2018, 2020), and this extension is crucial for handling flexibility, such as in agile and hybrid projects. During the simulation, the feasibility is evaluated as the project survives the risks, yields a feasible project structure, or fails and becomes infeasible.

Datasources

Selecting adequate project plans from a project database was problematic because none of the known project generators (such as ProGen (Kolisch and Sprecher, 1997), RanGen I (Demeulemeester, Vanhoucke, et al., 2003), and II (Vanhoucke et al., 2008)) nor open project data sources (such as MMLIB (Peteghem and Vanhoucke, 2014) and PSPLIB (Kolisch and Sprecher, 1997)) distinguish mandatory and supplementary tasks or consider strict and flexible dependencies. Therefore, there are no score values linked to task completion or task dependencies. Nevertheless, without considering flexible dependencies and priorities of task completion, the flexible project plans cannot be modeled because lower-priority (supplementary) tasks cannot be postponed, and the project plan cannot be restructured. Since there is still no real project database that contains an empirical distribution of the priorities or the flexible dependencies, the selection of tasks/dependencies and priorities followed a uniform distribution. Another challenge was that currently there not yet existing ALM-related databases, thus as ALM is strongly related to IT projects, such databases were used for this simulation. Three different datasets were used, which are the following:

Dataset A: contains selected data from the project databases. PSPLIB (j30 dataset) and MMLIB (MMLIB50 dataset). Database selection was performed based on the specified criteria, including the number of activities and serial/parallel indicators, of which the values best fit the projects in the IT sector.

Dataset B: In addition to the selected instances from existing standard datasets, project instance generators have been considered as another source of project data. The widely accepted generator ProGen (Kolisch and Sprecher, 1997) was selected.

Dataset C: Consists of empirical project data from the database presented by Batselier and Vanhoucke, 2015 from IT area.

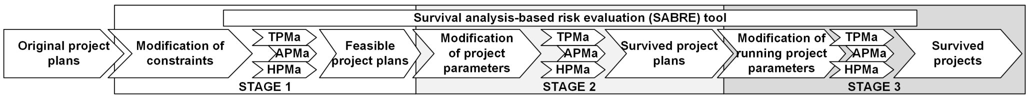
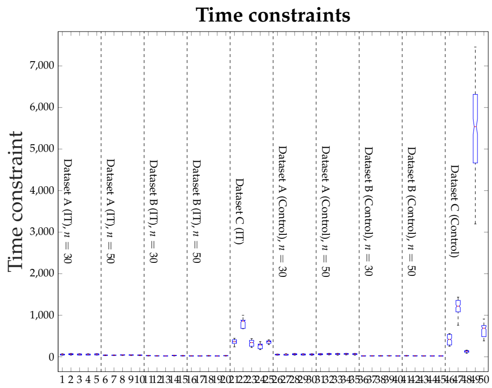


FIGURE 5: The proposed simulation framework

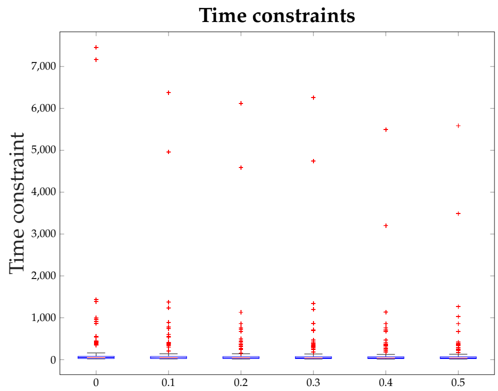
Figure 5 shows the proposed simulation framework. In this simulation, the influence of risk effects, such as the modification of constraints (see stage one) and overruns of cost and time (stage two and stage three) are mitigated by project management agents. The properties of the surviving projects are handled by different kinds of project management agents.

Descriptive statistics

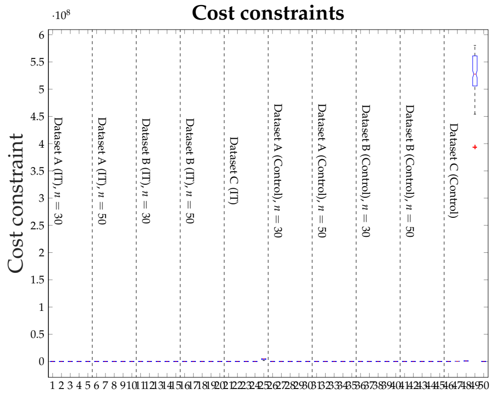
The simulation results in Figure 6 show the descriptive statistics of 48,000 scheduling problems, which are based on a set of 50 project structures. The project structures of 1-25 consisted of generated and real IT projects, and the control groups (26-50) followed construction project structures. Since 0-50% of task completions and dependencies between tasks are considered flexible, the constraints were calculated individually for each scheduling problem. Figure 6 shows the time, cost, quality, score, and resource constraints by project structures and by flexibility parameters. Constraints are specified at $\frac{1}{3}$ and $\frac{2}{3}$ of the theoretical range of project demands. These constraints were the same for all PMas; therefore, they can be compared. However, the specification of constraints fits the possibilities of the project plans. Therefore, it can be ascertained that the real projects from Dataset C have more time and cost demands (see project structures 21-25 and 46-50 in Figure 6(a,c)). In that case, the quality demands are also higher (see project structures 21-25 and 46-50 in Figure 6(e)). On the other hand, the generated projects (from Dataset B) have the highest resource demands (see project structures 15-20, 35-20 in 6(g)). From the MANOVA cluster, only one project structure (49) is shown to exhibit a relevant difference in constraints (compare 6(a) and 6(c) and 6(k)).



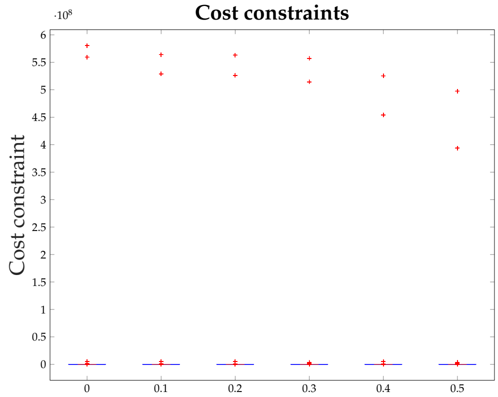
(a) $p = 0$



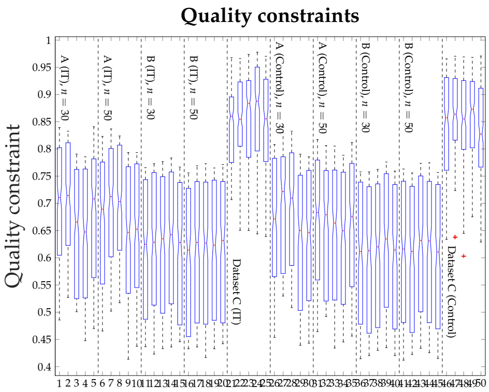
(b) $p = 0.0070$



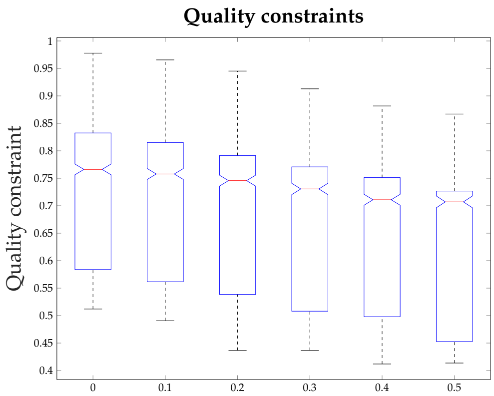
(c) $p = 0$



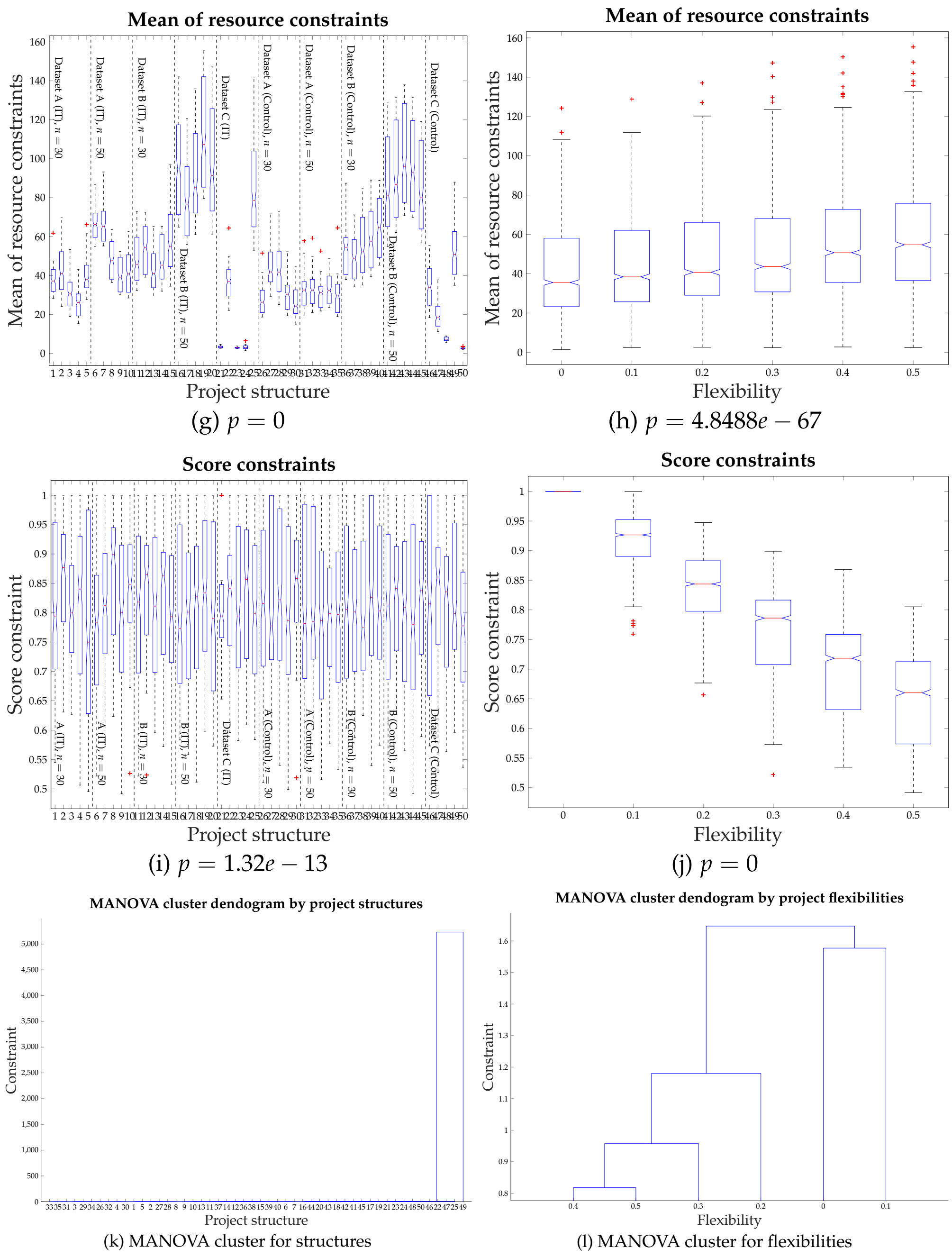
(d) $p = 0.9358$



(e) $p = 0$



(f) $p = 2.8632e - 159$



(g) $p = 0$

(h) $p = 4.8488e - 67$

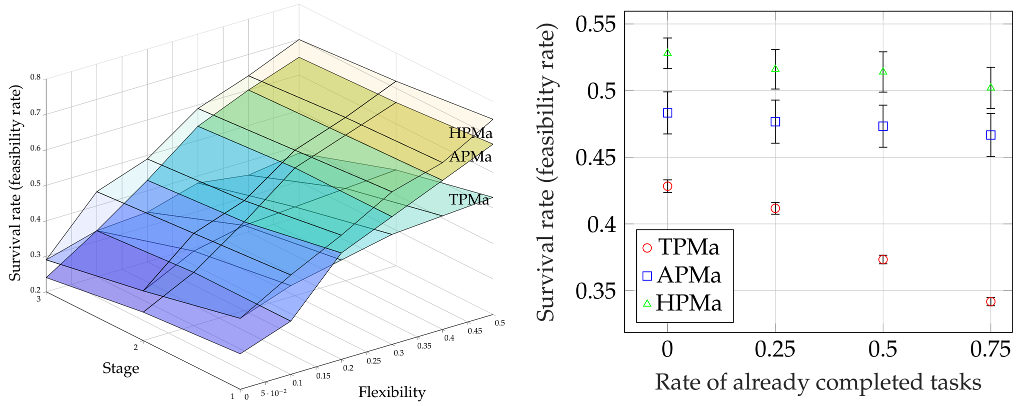
(i) $p = 1.32e - 13$

(j) $p = 0$

(k) MANOVA cluster for structures

(l) MANOVA cluster for flexibilities

FIGURE 6: Results of (M)ANOVA for constraints, project structures and flexibility



(a) Flexibility vs. feasibility

(b) Survivals in the implementation phase (Stage 3)

FIGURE 7: Feasibility rate of project management agents by flexibility

Feasibility versus flexibility

Figure 7(a) shows the feasibility rates of project management agents by stages and flexibility. The survival rate gives the ratio of feasible project scheduling problems in the given stage managed by TPMa, APMa or HPMa. Stage by stage, increasingly fewer projects survive the changes in constraints (Stage 1), the changes in demands and structures in the planning phase (Stage 2), and in the tracking phase (Stage 3). Especially in Stage 3 (see Figure 7(b)), the TPMa is more sensitive to the changes in demands, while the flexible approaches are generally less sensitive (see Figure 7(b)), even if the flexibility ratio is high (see Figure 7(a)).

In line with Figure 6(d,f,h,j), Figure 7(a) shows that generally, the increase in flexibility increases the rate of feasibility for all approaches. However, this opportunity can be exploited primarily by agile and hybrid approaches. In addition, in cases of lower flexibility ($< 20\%$), the TPMa manages more feasible projects than does APMa (see Figure 7(a)).

The interesting result is that HPMa made better use of the opportunities offered by flexibility. HPMa makes more feasible projects than the agile approach.

As a summary, the research question **RQ2** was focused on the feasibility of the project management approaches in for the ALM scope. The simulations showed the feasibility and sensitivity of the solutions based on multiple flexibility factors and different target functions. As a result **RA2** is accepted.

RQ3: *What are the risk factors for the scheduling problem in the ALM environment, and how are they influencing the feasibility and scheduling performance?*

Risk is characterized as the absence of assurance regarding the outcome, which can either be a positive change or a negative threat. Effective risk management involves the process of recognizing and regulating any hazards that could hinder an organization's ability to meet its business goals (Government Commerce, 2007). Managing risks are standard task in project management already, the intent behind Risk Management is to identify, evaluate, analyze, assess, and mitigate potential product issues defined in ISO/IEC 31000 (Barafort et al., 2019) also. Risk Management is a total product life cycle process. Risk is normally perceived as something to be avoided because of its association with threats, and as previously introduced, the ALM environment is more extended compared to the project scope, thus it provides additional space for potential risk factors to appear. Unfortunately, the risk factors for ALM are scarcely researched yet, the literature mainly contains narrowed-down ALM scopes. In the following, those ALM environment-related risks are presented, which are identified from the structural and scheduling point of view from the available academic literature. This means that the several general and introduction related, for example ALM organizational point of view risks are not considered here, even though there are significant risk factors also identified for ALM organization adaptation (Akgun et al., 2020; Tüzün et al., 2019), and later on related to operation (Cheng, 2010).

Risk management approaches are also different for Agile, which is often used in ALM environment also, as the intention of Agile ideology with the iterative loops is to "fail early" and still be able to react to the appearing issues. Buganová and Šimíčková (2019) creates an analysis to compare traditional and agile risk management and highlights the advantages and disadvantages on both sides. She points out that organizations use projects to manage changes for developing and deploying new products. In today's competitive environment, only those who can manage the risks and realize the project more efficiently will succeed.

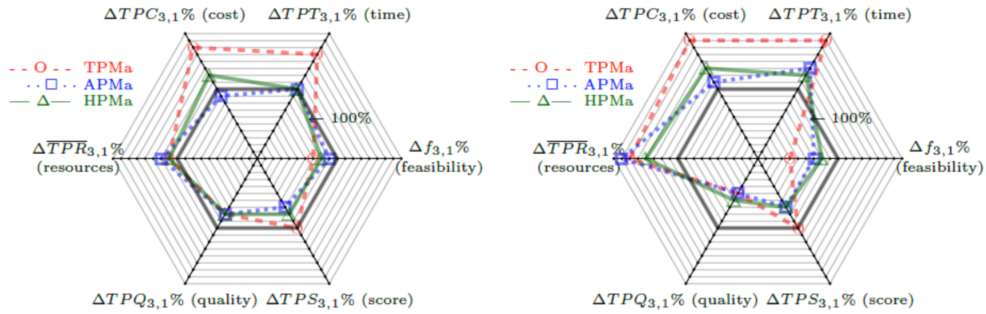
Due to the above-discussed differences in ALM and PM scope also the risk scopes require additional analysis. Project- and SW-wide risks need an extension in theory for the ALM scope also. Academic research for this field is very limited, a risk collection and assessment tool is proposed by Choetkiertikul and Sunetnanta (2012), mostly focusing on distributed SW development-related risks. However, mostly the general Life Cycle Management area risk management (Castaneda et al., 2020; Hummer et al., 2019; Niemann and Pisla, 2018; Sonnemann et al., 2015) or the Software Development Life Cycle is researched (Roy, 1962; Sahu et al., 2014). So in the following, as a restriction it will be treated the relevant risks such as project risks, which should be proper and acceptable for the ALM model. The limitation can be resolved with a further study of the ALM scope in the future.

Table 4 collects the factors from the literature related to the project management

Risk Factor	Presence in Project Management	Presence in ALM	Primary in ALM
Scope Creep	Komal et al. (2020) and Madhuri et al. (2018) Ajmal et al. (2022)	Aiello and Sachs (2016) and Rossberg (2019)	No
Change in Requirements	Kossmann (2016) and Venkatesh and Balani (2016)	Chanda et al. (2013)	No
Budget Overruns	Albtoush and Doh (2019) and Jackson (2002)	Banjanin and Strahonja (2018) and Ebert (2013)	No
Schedule Delays	Majerowicz and Shinn (2016) and Park (2021)	Aiello and Sachs (2016) and Tudenhöfner (2011)	No
Resource Constraints	Issa and Tu (2020) and Mishra (2020)	Rossberg (2019) and Rossman (2010)	No
Feasibility of problem	Issa and Tu (2020) and Rahman et al. (2021) Beek et al. (2024)	Aiello and Sachs (2016)	No
Quality Issues	Komal et al. (2020), Shafqat et al. (2022), and Wawak et al. (2020)	Akgun et al. (2020) and Otibine et al. (2017)	No
Lack of Traceability	No	Akgun et al. (2020) and Corallo et al. (2020)	Yes
Version Control Issues	No	Kääriäinen and Välimäki (2008) and Pirklbauer et al. (2009)	Yes

TABLE 4: Risk factors appearing in Project Management and Application Lifecycle Management environments

and Application Lifecycle Management risk factors, and showed which academics were investigating on the topics. Below providing a brief insight also how they are related to general project approaches, to SW projects and to ALM. Since the focus of this dissertation is the methodological approach, the main emphasis thus is on the ALM-specific non-planned activities elaboration and its effects.



(a) Uncorrelated risk effects

(b) Correlated case

FIGURE 8: Performance of risk mitigation of project management approaches

Performance of risk mitigation

From the simulation Figure 8 shows the performance of risk mitigation of the explored project management approaches. The ideal risk mitigation strategy maintains all project plans as feasible.

The TPMa keeps all tasks, and therefore, the score is maximum in all cases, but the price of this strategy is to "lose" more project plans than other strategies. Moreover, considering only feasible project plans, TPMa shows the most significant tendency to delays and overbudget situations. If risk factors are moderately correlated the TPMa demands a substantial amount of additional resources.

The APMa shows a very different picture. Interestingly, the agile technique is the only approach that reduces project costs despite the risk factors. The price of this strategy, however, is that it attains the largest decrease in quality and scope.

It is also interesting that when risk factors are moderately correlated, because of the forced parallelization, the demand for resources is increased to the greatest extent in this strategy. HPMa keeps most project plans feasible, and this approach creates balance within the multimode methods and the restructuring techniques, which means that this strategy can well mitigate the risk effects in order to keep the project plans within the constraints. In the meantime, it retains more of the scope than agile techniques.

When risk factors are correlated with each other, they greatly enhance each other's risk effects. These effects of interdependencies between risk factors occur particularly in the case of using TPMa. TPMa is very sensitive to the changes in the time, cost, and resource demands and their interdependencies, which is in line with the experience gained so far in software projects. The agile techniques can better mitigate the risk effects; however, if risk factors are correlated with each other, because of the forced parallelization, this technique is also sensitive to the resources.

Figure 9 shows the effect of project management agents, all explored structural properties, such as project structure and flexibility low-level risk factors, such as

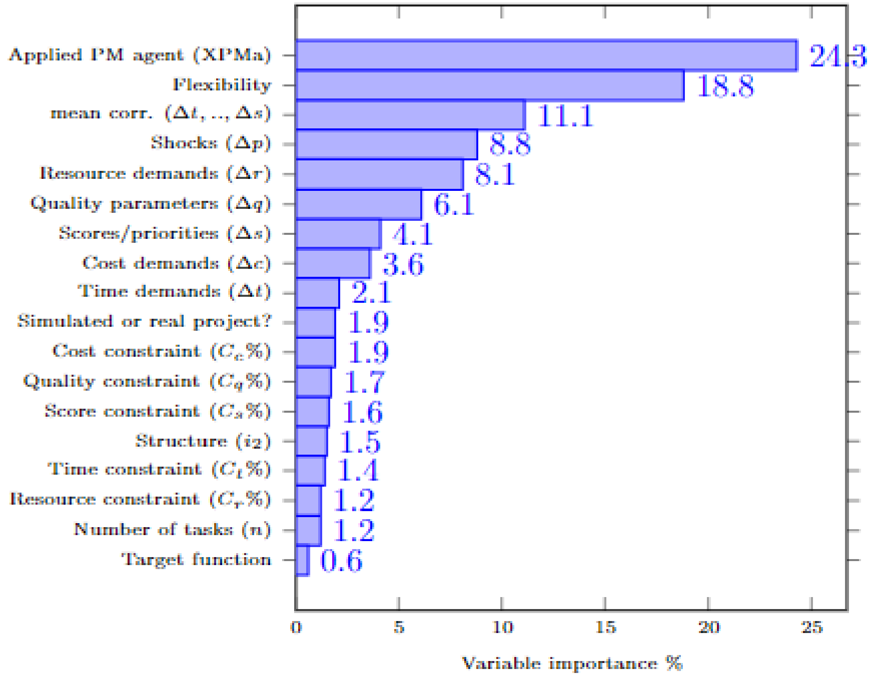


FIGURE 9: Variable importance for feasibility

changes in costs (Δc), duration (Δt), resource demands (Δr), et cetera; and high-level risks, that means when TPT, TPC, TPQ, or TPS values violate the corresponding constraint, that are assessed through the constraints (C_x%). According to the result, the low-level root causes and structural parameters have a greater direct impact on feasibility. The most important variable for maintaining the project feasibility is the selected project management agent (XPMa, 24.3%). In addition, the second most important variable is the flexibility rate (18.8%). The correlation between risk factors is more important (11.1%) than the risk factors themselves. According to the results, TPMa is the most sensitive to the shocks where only a few (10%) of task demands are changed, but these changes are (even 10 times) higher.

The risk factors ($\Delta r, .., \Delta t$) are more important than the constraints as the result of an agreement ($C_t\%, .., C_r\%$). This observation proves that after the contract phase, there are more challenges for the project manager to ensure that the project plan remains feasible. The more challenging task is resource allocation, both in traditional and flexible project management approaches.

As a summary, the research question **RQ3** was focused on the risk factors in the ALM area and how the risk factors are influencing the feasibility and the scheduling performance. The ALM-specific risk factors from the literature review are analyzed and then presented here with their description. Risk factors possible to involve in the simulation were more in detail evaluated and explained the correlations. As a result, **RA3** is accepted.

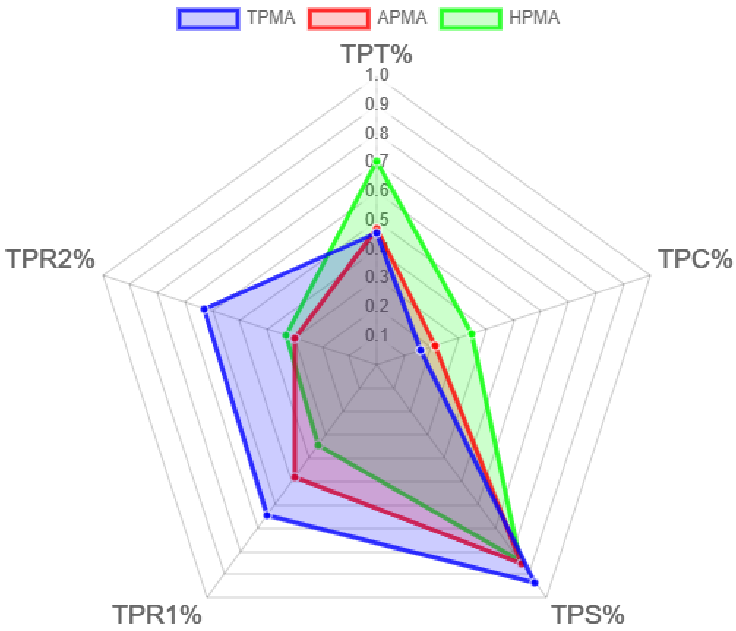


FIGURE 10: Radar chart for the performance of the agents for ALM

Validation

To robust the simulation results, a validation was conducted within an ALM environment for a real-life problem. The case study was carried out at a global automotive supplier established in 1871, a leading company that specializes in manufacturing brake systems, interior electronics, automotive safety, powertrain and chassis components, tires, and various other automotive parts. The organization operates in 58 countries, with a total sales of €33.8 billion and an employee count of approximately 190,000. In this case, the focus was on electronic brake systems' software applications, where the company is a top-tier supplier and competes with well-known companies that showcase the organizational structure and key data. The challenge for ALM handling appeared at the beginning of the decade, and the company has answered with an agile changeover from the structural and methodological point of view, which partially supports their targets, therefore investigating further how to support the challenging continuous SW update requests. The ALM naming convention is not explicitly used (yet); however, the problem definition is fitting for this case very well. As can be seen in the radiograph in Figure 10 the simulation results and the case study results are agree and support each other.

Agent/Function	**Time**	**Cost**	**Score**	**Resource 1**	**Resource 2**
TPMA	Worst	Worst	Best	Best	Best
APMA	Second	Second	Second	Second	Worst
HPMA	Best	Best	Worst	Worst	Second

TABLE 5: Summary table for the performance of agents for ALM

The summary visible in Table 5 refers to the setup of the case study results, which were presented and discussed with the company management and experts for the organization improvement projects. The results confirmed the planned changeovers for a hybrid type organization changeover instead of the full-agile organization introduction. Their main decision factors were related to time and cost, which is why the HPMA was favorable for them. The performance-related score factor, even though it seems to perform the worst, as the customer also accepted the agile methodology introduction, the content management was commonly agreed upon and accepted as not fulfilling all content. The company, though, expects some improvement in the resource efficiency on the training curve.

4 Research theses

Considering the research questions and assumptions with the corresponding results, three research theses with subtheses were formulated as follows:

RT1: Based on the unified ALM definition -including time, cost, resource and quality aspects also- the matrix-based planning model can be applied for scheduling purposes in the ALM environment, as it fulfills the flexible requirements relevant to the planned and non-planned activities.

RT2: The ALM scheduling problem can be set up as an extended project management scheduling problem and the existing project scheduling methodologies (TPM, APM, HPM) provide feasible solutions in the ALM environment with different performance levels:

RT2.1: TPM approach manages more feasible projects in case the flexibility is lower, and it provides all cases the highest score on customer satisfaction due to execution of all defined tasks, and fewer resources per time unit, however also the highest project budgets with longest projects due to the worst project scheduling performance.

RT2.2: APM approach manages significantly more feasible project than TPM when the flexibility is higher, and in general the shortest projects with the lowest budget if other target function is selected, however requires more resources per time unit than TPM.

RT2.3: HPM approach provides the most feasible projects in case higher flexibility is present, and shows the best performance for targets to reach (cost, time, quality), and this secures the best total project value.

RT3: A total of 9 risk factors extended from the project scope to the ALM environment are confirmed related to scope, time, cost, resources, and quality; also ALM specific risk factors focusing on the scope change, specific to ALM scheduling were identified:

RT3.1: The 3 most influential risk factors in the ALM environment are the following: the applied project management approach, the degree of structural flexibility, and the correlation between the risk factors.

RT3.2: In the ALM environment, the low-level risks (changes in cost, time, resources) have a higher impact than the high-level risks (fulfillment of constraints by the target function).

RT3.3: TPM is the most sensitive to the shock effects, with only small changes (10%) of task demands can result even 10 times higher modifications in the duration, resources, and costs.

These theses are in accordance to the research questions and research assumptions. In the next chapter the overall summary and conclusions are presented.

5 Summary and Conclusion

The rapid rise of technology requires a deep understanding and efficient management of software programs or applications, which are essential to modern business operations in industries such as info-communication, automotive, healthcare, aerospace, and many other areas. An unparalleled shift toward a software-based economy could be seen in recent decades. Companies of all sizes use software to innovate, optimize workflows, and offer value to customers. Software creation and maintenance now have a greater economic impact. Software project failures, though, can hurt the economy. Delays, budget overruns, and poor software quality cost money and reduce market competitiveness. Minimizing these risks and maximizing software development project economics requires effective handling of applications throughout their whole lifecycle. Application Lifecycle Management (ALM) offers a framework for such a solution as it manages the entire software application process from inception via development and maintenance of the application till its retirement. It can ensure long-term economic returns for software investments. This is why it is highly important to research this area and provide academic solutions for the business challenges listed above.

The contribution was threefold in this dissertation. On the first count, there is a contribution to the ALM literature by providing a synthesized ALM definition supporting future methodological research as it is based on a thorough systematic literature review for the definition and modeling of ALM based on peer-reviewed quality academic sources. This step was necessary as ALM is a relatively new and yet less researched area in the scientific literature, with mostly vendor-driven information available in the area. Therefore, a rigorous systematic literature review was conducted, including as wide a range of sources as possible, with the presumption of keeping the quality by selecting peer-reviewed sources. It was a keyword search for explicit ALM definition, and after identifying the relevant sources, a critical review was performed to gain the content. It was revealed that overall, seven types of ALM definitions are occurring among them. The most frequently occurring definitions highlight that ALM is strongly related to artifact management during the application management, also that it is a process similar to and based on the PLM but specific for SW development, and thirdly, that ALM is a paradigm, a holistic consisting of governance, development and operation/maintenance elements. Based on these relevant sources and field experience, I have proposed a unified ALM definition that joins the understanding of different aspects like scope, phases, key components, scheduling methodologies, flexibility, and metrics. Considering the fact, that such a widespread summary description was not yet available previously in the literature, this definition can serve as a base for future investigations by any scholars to proceed with methodological research by understanding better the scope and attributes of ALM.

On the second count, quantitative research proceeded for applicability and sensitivity checks of recent PM methodologies, such as traditional, agile, and hybrid, to see how efficiently they provide solutions for ALM scheduling problems. The matrix-based scheduling algorithm, which is applicable for projects, was extended with a flexible schedule handling option in the form of non-planned task handling. The project management execution types were then represented as agents, respectively. For Traditional Project Management (TPM), a Traditional Project Management agent (TPMa) was created, similarly for Agile PM and APMa, and for Hybrid PM and HPMa. In the environment, the scheduling performance is evaluated, and how the algorithms are performing is described. In addition to the performance evaluation, a risk evaluation has proceeded concerning the extended scope of the ALM compared to the classical project scope understandings from the academic literature.

On the third count, a present-day case study is executed in an ALM environment at an automotive supplier company that is facing application development challenges, and after the modeling and evaluations, recommendations are provided to their management about the results and potential changes for improvements. The case study is an important reinforcement of the previously theoretical and simulation environment-proven methodologies that were tested in a real-life problem. The environment and problem definition involved several professionals from the execution level up to management levels in several rounds to ensure the representativity. The simulation with the case-study data showed results according to the expectations based on the theoretical concept. The company appreciated the academic support for confirming an efficient way of working determination in their business area.

As an overall summary, the targeted goal of the dissertation is fulfilled to extend the ALM scientific literature with several value-added results, which also appeared in several conferences, proceedings, and in the form of article publications. Practitioners involved in the process were also highlighting the positive effect by asking and answering questions outside of their daily routine, helping them to rethink the way of their work, and even supporting it with proven academic data. The ALM area though far from being complete, is rather the start of a new journey for potential researchers based on the provided results.

5.1 Contribution to literature

First, it needs to be noted that the ALM-related academic literature is still scarce and expects growth from several perspectives because it started up as a mostly vendor-driven area, and not even a clear or unanimously accepted ALM definition exists. This is due to the vendor's purpose to form the ALM according to their business interest and also to the fact of the quick development of the concept itself. Business-related authors and professionals are sharing and contributing to the general knowledge base of the ALM; however, the scientific community currently has limited time and effort invested in the area.

1. The dissertation's first parts focused on the literature review, in a broader sense, to get to know the ALM more in detail and more focused on finding existing ALM definitions so that, as a next step, a unified concept can be created to support further methodological research by the academic community, which is under-researched today. The cross-sectional systematic literature review method was used to provide the base for the existing definitions in a broad scope of scholarly literature. Then, a critical review proceeded to analyze and create a unified ALM definition intended to integrate the scopes and attributes. So the first significant contribution was the created systematic literature review on the ALM definition. This can also be used as a base for a longitudinal or a meta-research, for example, for SIMILAR method (Kosztyán, Csizmadia, et al., 2021) for further extending the ALM literature. The additionally proposed ALM definition can be a base for further research by academics, opening up new horizons for methodological research, as the problem already exists in the business, as revealed by the case also. Also, in an article, the publication under review *he Evolution of Definition in Application Lifecycle Management – A Systematic Literature Review Article with a Critical Analysis*.

2. A matrix-based method was developed and proposed to examine the feasibility of IT projects with existing project management approaches (TPM, APM, HPM) programmed as agents. Similar feasibility-related comparisons did not exist before in the academic literature based on such complex simulations using real-life data as input. Therefore, the second main contribution is coming from here, published in the article *Survive IT! Survival analysis of IT project planning approaches*.

3. A case study proceeded with an automotive supplier company ALM-related challenging situation evaluation and using the previously demonstrated matrix-based method extended to the existing ALM environment. The case study involved several experts and managers in a leading automotive supplier that had not yet recorded such a complex HW-SW-related approach in the literature beforehand. The

Application Lifecycle Management scheduling problem was recognized and realized after the interviews and internal investigations were followed up on by leading managers. The quantified data and scheduling problem analysis with several approaches (TPM, APM, HPM) was revealing deeper context and potential further organization development for the company towards higher efficiency.

5.2 Practical implications

The systematic literature review can also provide valuable insights to professionals from a practical perspective, as no similar review has existed before. If the objective is to gather information on the fundamental concept and the extent of coverage for experts, this can be achieved through the article and theoretical assessment of the dissertation.

The proposed method compares traditional, agile, and hybrid project management approaches from the viewpoint of different stakeholders. It proposes a meta-network analysis method, which has not been applied in software development projects to date, and has also extended it for the ALM environment. The analysis showed that all methods not only have advantages but also have disadvantages. Most of them are in line with experience, but other methods need a deeper analysis. First, similar to experience, traditional project management approaches produced the most infeasible project plans. This result completely matches the Chaos Report's results (SGI, 2019), where waterfall projects, which follow traditional project management approaches, provided three times more failed projects. However, this study also demonstrated that a benefit would occur only if at least 20% of tasks and dependencies were flexible. The lesson could be learned that when this requirement cannot be satisfied, the agile project management approach can produce more failed (infeasible) projects.

In terms of scheduling, the traditional project management approach and the implemented TPMa operate only in terms of multimode task completion. This approach assumes that tasks can be completed in different kinds of ways. In contrast, agile techniques assume a flexible project structure, where dependencies between tasks can be flexible and lower-priority tasks can be postponed until the next project, but usually, only one completion mode is specified. The results showed that in the case of a flexible project environment, where the flexibility rate is high, this approach can genuinely produce more feasibility, and in this way, it can make remarkably more projects capable of success than traditional approaches. However, this advantage dissipates when the technology requires strict dependencies.

Due to project flexibility, another impressive result is that an agile project management approach usually obtains the shortest and least expensive projects, even though specifying a single implementation mode. However, this strategy's expense is less content and lower quality. For this reason, it is essential to involve customers for whom the scope of activities to be excluded from the project should be defined.

At the same time, it is also a vast challenge for developers to manage many parallel activities simultaneously.

The hybrid project management approach can take advantage of both flexibility and the choice of completion modes for scheduling; therefore, it provides the best schedules and those that are most feasible, and after the risk analysis, those with the most survived project plans, but these values are best only for the target functions.

The hybrid and, especially, the APMas are better in the flexible project environment. In this case, more feasible and better (for example , shorter, less expensive) projects can be specified. Nevertheless, the project structure, such as the size and the parallelization (i_2), are less important factors for survival. Agile and traditional project management approaches can usually better mitigate the effects of risk factors, while the hybrid approach helps to ensure the most surviving projects.

Hybrid techniques allow both multiple modes and flexible structures, and therefore, it is assumed that this is the supreme technique of project management. This assumption is reinforced by the fact that this technique provides the highest ratio of feasible solutions and the best scheduling performance when only the target function is considered. Based on the proposed database, HPMa provides the most feasible solutions; therefore, a software development project is more likely to survive the risk effects if a project plan is managed by a hybrid project management approach.

As an outlook, related to the automotive Case Study, available data can become information and knowledge for organizational setup and scheduling for this specific industry. The future for smart actuators and the challenge of SW becoming a product (SWaaP) are already leading to the Application Lifecycle Management world step-by-step. Industry demands input from academia related to processes, schedule optimization, and more.

5.3 Research Summary Table

See the below Table contains the summarized Research Questions, Assumptions and Theses for a better overview combined together.

Item	Statement
RQ1:	*How can a planning model be identified based on available scientific literature definitions that represents the Application Lifecycle Management (ALM) scheduling problem?*
RA1:	A planning model can be identified based on the unified ALM definitions from the scientific literature.
RT1:	Based on the unified ALM definition -including time, cost, resource and quality aspects also- the matrix-based planning model can be applied for scheduling purposes in the ALM environment, as it fulfills the flexible requirements relevant to the planned and non-planned activities.
RQ2:	*Do the existing project management scheduling methodologies (TPM, APM, HPM) produce feasible solutions in the ALM environment, and how are they performing?*
RA2:	The TPM, APM, HPM project management approaches using the matrix-based planning method can be extended to solve the scheduling problem, and result in feasible solutions with different results in the ALM environment. A simulation framework can be constructed to handle flexible dependencies and non-planned tasks.
RT2:	The ALM scheduling problem can be set up as an extended project management scheduling problem and the existing project scheduling methodologies (TPM, APM, HPM) provide feasible solutions in the ALM environment with different performance levels:
RT2.1:	TPM approach manages more feasible projects in case the flexibility is lower, and it provides all cases the highest score on customer satisfaction due to execution of all defined tasks, and fewer resources per time unit, however also the highest project budgets with longest projects due to the worst project scheduling performance.
RT2.2:	APM approach manages significantly more feasible project than TPM when the flexibility is higher, and in general the shortest projects with the lowest budget if other target function is selected, however requires more resources per time unit than TPM.
RT2.3:	HPM approach provides the most feasible projects in case higher flexibility is present, and shows the best performance for targets to reach (cost, time, quality), and this secures the best total project value.
RQ3:	*What are the risk factors for the scheduling problem in the ALM environment, and how are they influencing the feasibility and scheduling performance?*
RA3:	There are existing project-related risk factors that can be extended for ALM scheduling problems, however, due to the differences between project and ALM scope, ALM-specific risks appear also, which can have an effect on resources, cost, and timing, and can influence the feasibility and scheduling performance.
RT3:	A total of 9 risk factors extended from the project scope to the ALM environment are confirmed related to scope, time, cost, resources, and quality; also ALM specific risk factors focusing on the scope change, specific to ALM scheduling were identified:
RT3.1:	The 3 most influential risk factors in the ALM environment are the following: the applied project management approach, the degree of structural flexibility, and the correlation between the risk factors.
RT3.2:	In the ALM environment, the low-level risks (changes in cost, time, resources) have a higher impact than the high-level risks (fulfillment of constraints by the target function).
RT3.3:	TPM is the most sensitive to the shock effects, with only small changes (10%) of task demands can result even 10 times higher modifications in the duration, resources, and costs.

TABLE 6: Summary for Research Questions, Assumptions and Theses

6 The author's publications related to the topic

International Journal Articles

Kosztyán, Z. T., Novák, G., **Jakab, R.**, Szalkai, I., & Hegedűs, C. (2022). A matrix-based flexible project-planning library and indicators. In: *Expert Systems With Applications*. DOI: doi.org/10.1016/j.eswa.2022.119472

Kosztyán, Z. T., **Jakab, R.**, Novák, G., & Hegedűs, C. (2020). Survive IT! Survival analysis of IT project planning approaches. In: *Operations Research Perspectives*, 7, 100170. DOI: doi.org/10.1016/j.orp.2020.100170

Jakab, R., Novák, G. (2018). Project management approaches in application management services. In: *Chapters from the Academic Aspect of Project Management-Research and Teaching Methodologies Volume II., pp. 152-171. (ISBN: 9786150042190).*

Under review

Jakab, R., Kosztyán, Z. T. (2024). The Evolution of Definition in Application Lifecycle Management – A Systematic Literature Review Article with a Critical Analysis. Under review in: *European Journal of Information Technology* .

Proceedings

Kosztyán, Z. T., Novák, G., **Jakab, R.**, & Hegedűs, C. (2022). A Matrix-based Flexible Multi-level Project Planning Library and Indicators. In: *Proceedings of the 24th International DSM Conference (DSM 2022)*, Eindhoven, The Netherlands, October, 11-13, 2022 (pp. 48-57). DOI: doi.org/10.35199/dsm2022.06

Conferences

Jakab, R. (2023). Defining the way of Application Lifecycle Management. Abstract. PMUni International Conference on Project Management - PMUni 2023, Vienna, Austria.

Jakab, R., & Novák, G. (2023). Defining the way of Application Lifecycle Management. Abstract. OGIK-ISBIS 2023 Conference Proceedings, pp. 41., Pécsi Tudományegyetem, Pécs, Hungary.

Novák, G., & **Jakab, R.** (2021). Multi-level project planning and simulation using different delay cost profiles. Abstract. 15th International Conference on Economics and Business, Hungarian University of Transylvania, Miercurea Ciuc, Romania.

Jakab, R., & Novák, G. (2019). Application Lifecycle Management: evolution and revolution. Abstract. OGIK-ISBIS 2019 Conference Proceedings, pp. 51-52., Milton Friedman University, Budapest, Hungary. (Winning the Conference Award Best Presentation)

Novák, G., & **Jakab, R.** (2019). Multi-level Project Planning and Simulation using Earliness/Tardiness Compensation Profiles. Abstract. OGIK-ISBIS 2019 Conference Proceedings, pp. 44-45., Milton Friedman University, Budapest, Hungary.

Novák, G., & **Jakab, R.** (2018). A parser for standard datasets in project scheduling and simulation. Abstract. In: Bacsárdi, L., Bencsik, G., Pödör Z. OGIK-ISBIS 2018 Conference Proceedings, University of Sopron, pp. 20-21., Sopron, Hungary ISBN: (9786158109802)

Jakab, R., Novák, G. & (2018). Simulation and modeling of flexible projects and Application Management. Abstract. In: Bacsárdi, L., Bencsik, G., Pödör Z. OGIK-ISBIS 2018 Conference Proceedings, University of Sopron, pp. 47-48., Sopron, Hungary ISBN: (9786158109802)

References

Abdolshah, Mohammad (2014). "A review of resource-constrained project scheduling problems (RCPSP) approaches and solutions". In: *International Transaction Journal of Engineering, Management, & Applied Sciences & Technologies* 5.4, pp. 253–286.

Aiello, Bob and Sachs, Leslie (2016). *Agile application lifecycle management: Using DevOps to drive process improvement*. Addison-Wesley Professional.

Ajmal, Mian M, Khan, Mehmood, Gunasekaran, Angappa, and Helo, Petri T (2022). "Managing project scope creep in construction industry". In: *Engineering, Construction and Architectural Management* 29.7, pp. 2786–2809.

Akgun, Zuleyha, Yilmaz, Murat, and Clarke, Paul (2020). "Assessing application lifecycle management (ALM) potentials from an industrial perspective". In: *Systems, Software and Services Process Improvement: 27th European Conference, EuroSPI 2020, Düsseldorf, Germany, September 9–11, 2020, Proceedings 27*. Springer, pp. 326–338.

Albtoush, AM Faten and Doh, SI (2019). "A Review on causes of cost overrun in the construction projects". In: *Int. J. New Innov. Eng. Technol* 12.3, pp. 015–022.

Banjanin, Goran and Strahonja, Vjeran (2018). "Factors of Risk Reduction in Agile and Lean Enabled Governance of IT Project Portfolios". In.

Barafort, Béatrix, Mesquida, Antoni-Lluís, and Mas, Antónia (2019). "ISO 31000-based integrated risk management process assessment model for IT organizations". In: *Journal of Software: Evolution and Process* 31.1, e1984.

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. ISSN: 0263-7863. DOI: https://doi.org/10.1016/j.ijproman.2014.09.004. URL: http://www.sciencedirect.com/science/article/pii/S0263786314001410.

Beek, T van der, Souravlias, D, Essen, JT van, Pruyn, J, and Aardal, K (2024). "Hybrid differential evolution algorithm for the resource constrained project scheduling problem with a flexible project structure and consumption and production of resources". In: *European Journal of Operational Research* 313.1, pp. 92–111.

Buganová, Katarína and Šimíčková, Jana (2019). "Risk management in traditional and agile project management". In: *Transportation Research Procedia* 40. TRANSCOM 2019 13th International Scientific Conference on Sustainable, Modern and Safe Transport, pp. 986–993. ISSN: 2352-1465. DOI: https://doi.org/10.1016/j.trpro.2019.07.138. URL: https://www.sciencedirect.com/science/article/pii/S2352146519303060.

Castaneda, R Riascos, Ostrosi, E, Majić, T, Stjepandić, J, and Sagot, J-C (2020). "A method to explore product risk in product lifecycle management of configured products". In: *Proceedings of the design society: DESIGN conference*. Vol. 1. Cambridge University Press, pp. 687–696.

Chanda, Sandeep, Foggon, Damien, Chanda, Sandeep, and Foggon, Damien (2013). "Application lifecycle management". In: *Beginning ASP. NET 4.5 Databases*, pp. 235–249.

Chappell, David et al. (2010). "What is application lifecycle management?" In: *Chappell & Associates*, p. 6.

Cheng, Yu Chin (2010). "Rapid Application Lifecycle Management: a new Approach with Tool Support." In.

Choetkiertikul, Morakot and Sunetnanta, Thanwadee (2012). “A risk assessment tool using a CMMI Quantitative Approach”. In: *International Journal of Engineering and Technology* 4.4, p. 352.

Corallo, Angelo, Latino, Maria Elena, Menegoli, Marta, and Pontrandolfo, Pierpaolo (2020). “A systematic literature review to explore traceability and lifecycle relationship”. In: *International Journal of Production Research* 58.15, pp. 4789–4807.

Creemers, Stefan (July 2015). “Minimizing the expected makespan of a project with stochastic activity durations under resource constraints”. In: *Journal of Scheduling* 18.3, pp. 263–273. ISSN: 1099-1425. DOI: 10.1007/s10951-015-0421-5. URL: https://doi.org/10.1007/s10951-015-0421-5.

Demeulemeester, Erik, Vanhoucke, Mario, and Herroelen, Willy (Jan. 2003). “RanGen: A Random Network Generator for Activity-on-the-Node Networks”. In: *Journal of Scheduling* 6.1, pp. 17–38. ISSN: 1099-1425. DOI: 10.1023/A:1022283403119. URL: https://doi.org/10.1023/A:1022283403119.

Demeulemeester, Erik Leuven and Herroelen, Willy S (2006). *Project scheduling: a research handbook*. Vol. 49. Springer Science & Business Media.

Deuter, Andreas, Otte, Andreas, Ebert, Marcel, and Possel-Dölken, Frank (2019). “Developing the Requirements of a PLM/ALM Integration: An Industrial Case Study”. In: *Product Lifecycle Management (Volume 4): The Case Studies*, pp. 125–143.

Deuter, Andreas and Rizzo, Stefano (2016). “A critical view on PLM/ALM convergence in practice and research”. In: *Procedia Technology* 26, pp. 405–412.

Dingsøyr, Torgeir, Nerur, Sridhar, Balijepally, VenuGopal, and Moe, Nils Brede (2012). “A decade of agile methodologies: Towards explaining agile software development”. In: *Journal of Systems and Software* 85.6. Special Issue: Agile Development, pp. 1213–1221. ISSN: 0164-1212. DOI: https://doi.org/10.1016/j.jss.2012.02.033. URL: http://www.sciencedirect.com/science/article/pii/S0164121212000532.

Ebert, Christof (2013). “Improving engineering efficiency with PLM/ALM”. In: *Software & Systems Modeling* 12.3, pp. 443–449.

Government Commerce, OGC-Office of (2007). *The official introduction to the ITIL service lifecycle*. The Stationery Office.

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*.

Hummer, Waldemar, Muthusamy, Vinod, Rausch, Thomas, Dube, Parijat, El Maghraoui, Kaoutar, Murthi, Anupama, and Oum, Punleuk (2019). “Modelops: Cloud-based lifecycle management for reliable and trusted ai”. In: *2019 IEEE International Conference on Cloud Engineering (IC2E)*. IEEE, pp. 113–120.

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

Jackson, Simon (2002). “Project cost overruns and risk management”. In: *Proceedings of Association of Researchers in Construction Management 18th Annual ARCOM Conference, Newcastle, Northumber University, UK*. Vol. 1, pp. 1–10.

Kääriäinen, Jukka and Välimäki, Antti (2008). “Impact of Application Lifecycle Management—A Case Study”. In: *Enterprise Interoperability III*. Springer, pp. 55–67.

Kolisch, Rainer and Sprecher, Arno (1997). “{PSPLIB} - A project scheduling problem library: {OR} Software - {ORSEP} Operations Research Software Exchange Program”. In: *European Journal of Operational Research* 96.1, pp. 205–216. ISSN: 0377-2217. DOI: http://dx.doi.org/10.1016/S0377-2217(96)00170-1. URL: http://www.sciencedirect.com/science/article/pii/S0377221796001701.

Komal, Bakhtawar, Janjua, Uzair Iqbal, Anwar, Fozia, Madni, Tahir Mustafa, Cheema, Muhammad Faisal, Malik, Muhammad Noman, and Shahid, Ahmad Raza (2020). "The impact of scope creep on project success: An empirical investigation". In: *IEEE Access* 8, pp. 125755–125775.

Kossmann, Mario (2016). *Requirements management: How to ensure you achieve what you need from your projects*. Routledge.

Kosztyán, Zsolt T (2022). "MFPP: Matrix-based flexible project planning". In: *SoftwareX* 17, p. 100973.

Kosztyán, Zsolt T, Csizmadia, Tibor, and Katona, Attila I (2021). "SIMILAR–Systematic iterative multilayer literature review method". In: *Journal of Informetrics* 15.1, p. 101111.

Kosztyán, Zsolt T and Szalkai, István (2018). "Hybrid time-quality-cost trade-off problems". In: *Operations Research Perspectives* 5, pp. 306–318. ISSN: 2214-7160. DOI: `https://doi.org/10.1016/j.orp.2018.09.003`. URL: `http://www.sciencedirect.com/science/article/pii/S2214716018300812`.

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.

Kreter, Stefan, Schutt, Andreas, Stuckey, Peter J, and Zimmermann, Jürgen (2018). "Mixed-integer linear programming and constraint programming formulations for solving resource availability cost problems". In: *European Journal of Operational Research* 266.2, pp. 472–486.

Kwak, Young Hoon and Ingall, Lisa (2007). "Exploring Monte Carlo simulation applications for project management". In: *Risk Management* 9.1, pp. 44–57.

Madhuri, K Lakshmi, Suma, V, and Mokashi, Uma Mohan (2018). "A triangular perception of scope creep influencing the project success". In: *International Journal of Business Information Systems* 27.1, pp. 69–85.

Majerowicz, Walt and Shinn, Stephen A (2016). "Schedule matters: Understanding the relationship between schedule delays and costs on overruns". In: *2016 IEEE Aerospace Conference*. IEEE, pp. 1–8.

Mishra, Alok and Alzoubi, Yehia Ibrahim (2023). "Structured software development versus agile software development: a comparative analysis". In: *International Journal of System Assurance Engineering and Management*, pp. 1–19.

Mishra, Anjay Kumar (2020). "Implication of theory of constraints in project management". In: *International Journal of Advanced Trends in Engineering and Technology* 5.1, pp. 1–13.

Moukrim, Aziz, Quilliot, Alain, and Toussaint, Hélène (2015). "An effective branch-and-price algorithm for the preemptive resource constrained project scheduling problem based on minimal interval order enumeration". In: *European Journal of Operational Research* 244.2, pp. 360–368.

Niemann, Jörg and Pisla, Adrian (2018). "Sustainable potentials and risks assess in automation and robotization using the life cycle management index tool—LY-MIT". In: *Sustainability* 10.12, p. 4638.

Otibine, Tobias, Mbuguah, Samuel, Kilwake, Humphrey, and Tsinale, Harriet (Nov. 2017). "Application Lifecycle Management Activities For Quality Assurance In Software Development". In: *International Journal of Trend in Research and Development* 4, pp. 2394–9333.

Page, Matthew J et al. (2021). "The PRISMA 2020 statement: an updated guideline for reporting systematic reviews". In: *International journal of surgery* 88, p. 105906.

Park, Jung Eun (2021). “Schedule delays of major projects: what should we do about it?” In: *Transport Reviews* 41.6, pp. 814–832.

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: https://doi.org/10.1016/j.ejor.2013.10.012. URL: http://www.sciencedirect.com/science/article/pii/S0377221713008357.

Pirklbauer, Guenter, Ramler, Rudolf, and Zeilinger, Rene (2009). “An integration-oriented model for application lifecycle management”. In: *International Conference on Enterprise Information Systems*. Vol. 1. SCITEPRESS, pp. 399–402.

Rahman, Md Humyun Fuad, Chakrabortty, Ripon K, and Ryan, Michael J (2021). “Managing uncertainty and disruptions in resource constrained project scheduling problems: a real-time reactive approach”. In: *IEEE Access* 9, pp. 45562–45586.

Rao, Purnima and Palaniappan, Kandhasami (2020). “Continuous Engineering Through ALM-PLM Integration”. In: *Product Lifecycle Management Enabling Smart X: 17th IFIP WG 5.1 International Conference, PLM 2020, Rapperswil, Switzerland, July 5–8, 2020, Revised Selected Papers 17*. Springer, pp. 798–811.

Rasnacis, Arturs and Berzisa, Solvita (2017). “Method for Adaptation and Implementation of Agile Project Management Methodology”. In: *Procedia Computer Science* 104. ICTE 2016, Riga Technical University, Latvia, pp. 43–50. ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2017.01.055. URL: https://www.sciencedirect.com/science/article/pii/S187705091730056X.

Reiff, Janine and Schlegel, Dennis (2022). “Hybrid project management–a systematic literature review”. In: *International journal of information systems and project management* 10.2, pp. 45–63.

Rossberg, Joachim (2019). “Introduction to Application Life Cycle Management”. In: *Agile Project Management with Azure DevOps*. Springer, pp. 1–36.

Rossman, Bruce (2010). *Application Lifecycle Management-Activities, Methodologies, Disciplines, Tools, Benefits, ALM Tools and Products*. Emereo Pty Ltd.

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

Sahu, Kavita, Shree, R, and Kumar, R (2014). “Risk management perspective in SDLC”. In: *International Journal of Advanced Research in Computer Science and Software Engineering* 4.3.

Al-Saqqa, Samar, Sawalha, Samer, and AbdelNabi, Hiba (2020). “Agile software development: Methodologies and trends.” In: *International Journal of Interactive Mobile Technologies* 14.11.

SGI (2019). *CHAOS Manifesto*. Standish Group International.

Shafqat, A, Oehmen, Josef, and Welo, Torgeir (2022). “Planning unplanned design iterations using risk management and learning strategies”. In: *Journal of Engineering Design* 33.2, pp. 120–143.

Sonnemann, Guido, Gemechu, Eskinder Demisse, Remmen, Arne, Frydendal, Jeppe, and Jensen, Allan Astrup (2015). “Life cycle management: Implementing sustainability in business practice”. In: *Life cycle management*, pp. 7–21.

StandishGroup (2020). “The CHAOS report”. In: URL: http://www.standishgroup.com.

Tritschler, Martin, Naber, Anulark, and Kolisch, Rainer (2017). “A hybrid metaheuristic for resource-constrained project scheduling with flexible resource profiles”. In: *European Journal of Operational Research* 262.1, pp. 262–273.

Tudenhöfner, Eduard (2011). *Integration of Performance Management into the Application Lifecycle*. diplom. de.

Tüzün, Eray, Tekinerdogan, Bedir, Macit, Yagup, and İnce, Kürşat (2019). "Adopting integrated application lifecycle management within a large-scale software company: An action research approach". In: *Journal of Systems and Software* 149, pp. 63–82.

van den Ende, Leonore and van Marrewijk, Alfons (2014). "The ritualization of transitions in the project life cycle: A study of transition rituals in construction projects". In: *International Journal of Project Management* 32.7, pp. 1134–1145. ISSN: 0263-7863. DOI: `https://doi.org/10.1016/j.ijproman.2014.02.007`. URL: `https://www.sciencedirect.com/science/article/pii/S0263786314000295`.

Vanhoucke, Mario, Coelho, JosÃ©, Debels, Dieter, Maenhout, Broos, and Tavares, LuÃs 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. ISSN: 0377-2217. DOI: `https://doi.org/10.1016/j.ejor.2007.03.032`. URL: `http://www.sciencedirect.com/science/article/pii/S037722170700375X`.

Venkatesh, Bharti and Balani, Lalit (2016). "Requirement management a key to successful project management for software systems". In: *Voice of Research* 5.1.

Wawak, Sławomir, Ljevo, Žanesa, and Vukomanović, Mladen (2020). "Understanding the key quality factors in construction projects—A systematic literature review". In: *Sustainability* 12.24, p. 10376.