Exploring Indicators for Multiple Modes in Resource-Constrained Project Scheduling

Gergely Lajos Novák[1], Zsolt Tibor Kosztyán[2]

[1]Continental Automotive Hungary Ltd.
[2]University of Pannonia

Abstract: Project scheduling is a critical aspect across various industries that is still extensively studied. Unlike project plans for the resource-constrained project scheduling problem, those in the multi-mode extension (MRCPSP) remain relatively unexplored regarding project characteristics and their implications. The present study explores artificial project plans featuring multiple modes to address this gap. The lower and upper bounds are determined for the adapted project-related indicator groups using metaheuristic optimization as a mode selection strategy. Our approach contributes by offering insights into the multi-mode context and characteristics influencing project scheduling and managerial decisions.

Keywords: Project scheduling, multi-mode, project databases, project indicators, MRCPSP

1 Introduction

Project scheduling has gained a vital role in various industries, with the resource-constrained project scheduling problem (RCPSP) being extensively studied in project planning literature since the 1950s. Traditionally, this problem involves scheduling a set of activities subject to precedence and resource constraints to optimize objectives such as minimizing project duration. However, existing approaches often overlook the inherent flexibility in scenarios where tasks can be executed using alternative methodologies or technologies, each mode representing a combination of different time and resource requirements for the task. While several extensions of the RCPSP have been explored, summarized by Hartmann and Briskorn (2022) and Habibi et al. (2018), one of the most significant variant became the multi-mode RCPSP (MRCPSP) (Elmaghraby, 1977). Due to the high practical relevance, it has brought a significant scholarly and industrial focus since its introduction. Despite the growing importance of MRCPSP, the literature primarily focus on efficient algorithm development, neglecting the exploration of project characteristics, especially regarding project indicator and the implications of the trade-offs provided by multiple modes. In this context, it is essential to have the knowledge in advance on the impact of demands and the degree of flexibility to be able to balance between the different constraints of the projects. Our research is thus guided by the following set of research questions (RQs):

RQ$_1$ How is it possible to determine the lower and upper bounds of project-related indicators using exact or heuristic methods considering multiple execution modes?

RQ$_2$ Which project-related indicators demonstrate the strongest relationship with project duration and resource utilization?

To answer the research questions, we explore artificial project plans incorporating multiple modes by solving the mode assignment subproblem. We determine indicator values for both the lower and upper bounds, capturing worst- and best-case scenarios for time- and renewable-resource demands. Our objective is to close the identified research gap by introducing a framework to analyze multi-mode projects. This framework can extend the understanding of project characteristics and provides insights into project time- and resource characteristics within multi-mode project settings. To demonstrate our approach, a case study was also carried out for a real problem. The current paper is organized as follows. In the following Section 2, we review the relevant literature and provide background of the problem. Section 3 details our approach to mode assignment with the data sources and methodologies used and presents a practical example. Section 4 shows the computational results and the case-study, Section 5 discusses them and finally, Section 6 concludes the paper, gives the limitations the current study, and directions for future research.

2 Literature review

The MRCPSP serves as a fundamental extension of the resource-constrained project scheduling problem (RCPSP), which has been extensively studied in the literature. Although the MRCPSP is more practical than its single-mode counterpart, it is also much more challenging. Researchers have developed both exact (Patterson et al., 1989; Słowiński, 1980; Talbot, 1982; Sprecher et al., 1997; Sprecher and Drexl, 1998; Speranza and Vercellis, 1993; Hartmann and Sprecher, 1996; Zhu et al., 2006), heuristic (Lova et al., 2006; Knotts et al., 2000; Boctor, 1996b; Drexl and Gruenewald, 1993; Özdamar and Ulusoy, 1994; Kolisch and Drexl, 1997; Talbot, 1982; Noori and Taghizadeh, 2018; Gerhards et al., 2017), and metaheuristic (Józefowska et al., 2001; Boctor, 1996a; Bouleimen and Lecocq, 2003; Nonobe and Ibaraki, 2002; Zhang et al., 2006; Ranjbar et al., 2009; Słowiński et al., 1994; Lova et al., 2009; Alcaraz et al., 2003; Hartmann, 2001; Mori and Tseng, 1997; Özdamar and Ulusoy, 1994; Fernandes Muritiba et al., 2018) solutions for solving the MRCPSP. Both the RCPSP and its generalized MRCPSP are classified as NP-hard problems (Blazewicz et al., 1983; Kolisch, 1995). For an overview of the various exact, heuristic, and metaheuristic methods proposed to solve MRCPSP, we refer to Weglarz et al. (2011) and Noori and Taghizadeh (2018). Peteghem and Vanhoucke (2014) present a detailed review and analysis of heuristic methods and Changchun et al. (2018) lists further articles.

The basic RCPSP's object is to minimize the project duration. The objective of the MRCPSP is extended by finding a mode in addition to the start time for each activity such that the makespan is minimized and the schedule is feasible concerning the precedence and resource constraints. The MRCPSP encompasses two primary challenges: firstly, determining a single mode for each activity, and secondly, scheduling these activities considering constraints and further possible attributes. In this paper, we primarily investigate the mode selection sub-problem and only give a general overview as a background of the related scheduling sub-problem.

Exact methods often fail to find optimal solutions for large problem sizes in a reasonable time. In the discrete time/resource trade-off problem (DTRTP), the duration of an activity varies based on the allocation of renewable resources. The enumeration of all feasible mode combinations is practically impossible due to the huge number of possible combinations. In contrast, metaheuristic algorithms can obtain near-optimal solutions quickly. For single-mode problems, there are several ways to calculate the lower bounds (LB) of project duration (Boctor, 1990). In contrast, the lower bounds for the MRCPSP are typically derived from the critical path method (CPM) (Kelley Jr and Walker, 1959), selecting the mode with the shortest duration for each activity (Sprecher et al., 1997).

The key literature is summarized by selecting a recent study connected to the main features connected to this paper. Such papers are Noori and Taghizadeh (2018), considering genetic algorithm of the mode-selection sub-problem. Van Eynde and Vanhoucke (2020) consider recent time- and resource-related indicators for RCMPSP, also relevant to our study. Hartmann and Briskorn (2022) give an overview of various alternative objectives for RCPSP, and while Stürck and Gerhards (2018) deals with the classical lower bounds for the MRCPSP, we consider indicators as objectives for lower and upper bounds of the mode selection subproblem. Beşikci et al. (2015) considers MRCMPSP with multiple modes, while our study contributes to the single-project case. Lova et al. (2009) includes non-renewable resources which are not considered in our study. Kosztyán and Szalkai (2020) also uses matrix-based representation for MRCPSP and considers various trade-off problems, while we study the discrete time-resource trade-off problem (DTRTP).

It can be stated that various methods were proposed in the literature to tackle the MRCPSP by also solving the mode assignment problem to convert multi-mode problem into a fixed, single-mode RCPSP. While significant attention has been given to developing efficient algorithms for MRCPSP, there remains a notable gap in the literature regarding the exploration of project characteristics, particularly in terms of project indicators and the implications of trade-offs inherent in multiple modes. By addressing this gap, our study aims to provide valuable insights into these aspects of MRCPSP, enhancing our understanding of the problem and potentially leading to more effective solutions.

To characterize projects for structural complexity, duration, slacks, and resource demands, project indicators are essential. This information can fundamentally influence how the scheduling or resource allocation algorithm performs. Two main groups of indicators emerge from the literature. The first group, referred to as structural indicators, encompasses metrics related to project complexity (see, e.g., Sprecher, 1994; Davis, 1975; Mastor, 1970), which consider the topological features of the project. The second group, known as demand indicators, includes metrics related to various domains, such as time (Patterson 1976) and renewable resources (Patterson, 1976; Kolisch et al., 1995; Van Eynde and Vanhoucke, 2020). Quality and cost demands are represented by only a few indicators in this group.

3 Methods

3.1 Data sources and selected projects

The Boctor (Boctor, 1996b) multi-mode dataset and its subsets with 50 and 100 activities were used for our investigation. Each subset supports one, two or four renewable resources and is used in various studies. The instances are available for download from Boctor (2004). To process the data in a matrix-based format, we employed the parser tool of Kosztyán et al. (2023).

3.2 Matrix-based representation

The calculation of project indicators and the proposed metaheuristic optimization for mode selection rely on a matrix-based model. This representation is based on the DSM (Steward, 1981) utilizing DMM (Danilovic and Browning, 2007) to extend the model for projects (Kosztyán, 2020) called the project domain matrix (PDM). It specifies two mandatory domains: the logic domain (LD) and the time domain (TD). Additionally, it supports four supplementary domains: cost domain (CD), quality domain (QD), nonrenewable-resources domain (ND), and renewable-resources domain (RD), each represented as submatrices within the model structure. The model can represent all the mentioned attributes, nevertheless, the available data and the existing indicators primarily focuses on single projects and renewable resources, thereby we excluded the nonrenewable resources, cost, and quality domains from the analysis as shown in Figure 1.

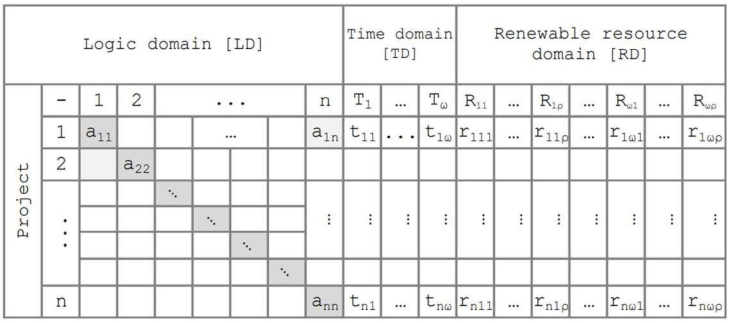


Figure 1. Matrix representation of project plans

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.

LD is the mandatory domain containing the matrix representation of the logic structure of the project's, where **LD** ∈ {0, 1} $n \times n$, for each $i \leq n$, $[\mathbf{LD}]_{ii} = 1$, and for each $i \neq j$, we have $(A_i, A_j) \in A$ if and only if $[\mathbf{LD}]_{i,j} = 1$ (otherwise $[\mathbf{LD}]_{i,j} = 0$). **TD** is an n by w matrix domain (submatrix) of positive real values for task durations, where n is the number of tasks and w is the number of completion modes. In the case of $w = 1$, the project plan is called single-modal, while in our case of $w > 1$, the project plan is called a multimodal project plan. **RD** is an n by $w \cdot r$ domain for renewable resources of each task, where r is the number of renewable resources. 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, $l_{ij} = [\mathbf{LD}]_{i,j} = 0$ if $i > j$). Let us denote **PDM** = [**LD**,**TD**,**RD**] as the matrix representation of a project plan. **PDM** is an $n \cdot w \cdot (1 + r)$, where n is the number of tasks, w is the number of completion modes, and r is the number of resources.

3.3 Calculating bounds

The sharp bounds for the maximum critical project length (total project time) can be calculated as follows. Let $\mathbf{TD}_{\min}$ and $\mathbf{TD}_{\max}$ denote the minimal and maximal time demands, respectively, where $\mathbf{TD}_{i\min} = t_{i\,\min} = \min_{\omega}, t_{i,\omega} \in [\mathbf{TD}_i]$, $\mathbf{TD}_{imax} = t_{i\,\max} = \max_{\omega}, t_{i,\omega} \in [\mathbf{TD}_i]$ for $i = 1, \ldots, n$; $\omega = 1, \ldots, w$. We find the mode with the lowest and largest task duration for each task. The vectors $\mathbf{v}_{\min}$ and $\mathbf{v}_{\max}$ represent the indices of modes corresponding to the minimal and maximal values respectively, for each task: $\mathbf{v}_{\min} = [\omega_{\min,1}, \omega_{\min,2}, \ldots, \omega_{\min,n}]$, similarly, $\mathbf{v}_{\max} = [\omega_{\max,1}, \omega_{\max,2}, \ldots, \omega_{\max,n}]$. It worths mentioning, that the critical project length stands out as the exception, as for other indicators, there's no straightforward method available to calculate the optimal lower and upper bounds.

3.4 Studied project-related indicators

Indicators describe specific properties of projects, which allows academics and practitioners to describe, classify and compare project plans, or even map solutions to procedures (Guo et al., 2021) and to generate new benchmarks (Van Eynde et al., 2024). As for our case, the mode selection does not have an impact on the structural indicators, the complexity and topological indicators are only used to characterize project plans. The order strength (OS) (Mastor, 1970) and the serial-parallel (SP) indicators (Tavares, 1999; Vanhoucke et al., 2008) are selected for this purpose. Furthermore, they can predict the complexity of solving the RCPSP (Coelho and Vanhoucke, 2020). The rest of the indicators characterize the project demands. There are time-related indicators, such as the maximum critical path length (MAXCPL), the mean and variance of activity durations (XDUR and VA-DUR, respectively), the percent of activities with positive total slack (PCTSLACK), the average total slack per activity (XSLACK), the total and average slack ratios (TOTSLACK-R and XSLACK-R, respectively), the percent of activities with positive free slack (PCTFREESLK) and the average free slack per activity (XFREESLK) (Patterson, 1976). There are also renewable resource-related indicators, such as the resource factor (RF) (Kolisch et al., 1995) (i.e., the density of the resource domain RD), the percent of activities that require the given resource type (PCTR) (Patterson, 1976), the resource use (RU) of the activities (Demeulemeester et al., 2003), the average demand from each resource type, resource constrainedness (RC) (Patterson, 1976), resource strength (RS) (Kolisch et al., 1995), and the following four indicators used by (Patterson, 1976), which consider the precedence relations of the activities to describe resource needs, the utilization of each type of resource, the constrainedness of the resources, and obstruction and underutilization of the resources. The indicators considering the strictness of the renewable resources, such as RC, RF, RS have a significant impact on the hardness of the scheduling problem as well. The Gini coefficient (Van Eynde and Vanhoucke, 2020) measures the inequality of renewable resource demands. We refer to Kosztyán and

Novák (2024) for a summary of all relevant indicators and their operationalizations, adapted to the matrix-based representation, which was also integrated for our framework.

3.5 Applied meta-heuristic algorithm

For most indicators, the process involves optimizing each project-related indicator by minimizing or maximizing their values as objective functions. Enumerating all mode combinations is practically infeasible due to the huge number of possible combinations. However, meta-heuristic algorithms could offer an approach to obtain near-optimal solutions and thus a genetic algorithm (GA) is proposed to solve the underlying problem. The genetic algorithm iteratively generates and refines potential solutions $\mathbf{v}'_{\min}$ ($\mathbf{v}'_{\max}$), represented as individuals within a population, to minimize (maximize) the target function. The selected modes are extracted from the submatrices of the original **PDM** so the indicators can be calculated for a single-mode case. As an example, the lower bound for **TD** becomes an $n \cdot 1$ column vector $\mathbf{TD}' = [\mathbf{TD}]_{i,\omega'\min,i}$, $i = 1, \ldots, n$ with the optimized mode selection. Similarly, **RD** becomes for all ρ an n by $1 \cdot r$ submatrix, $\mathbf{RD}' = [\mathbf{RD}]_{i,w(\rho-1)+\omega'\min,i}$, $i = 1, \ldots, n$, $\rho = 1, \ldots, r$, $\omega = 1, \ldots, w$ with optimized modes selected for each renewable resource. Finally, the **PDM** becomes $\mathbf{PDM}' = [\mathbf{LD}, \mathbf{TD}', \mathbf{RD}']$ for which the indicator values can be calculated.

The initial population is randomly generated within the constraints, then each iteration, or generation, evaluates the fitness of individuals based on the calculated indicator value. Individuals with higher fitness are more likely to reproduce through crossover and mutation. Offsprings replace less fit individuals, evolving the population over generations. The best individuals in the final generation represent approximate solutions to the optimization problem. The MATLAB (Mathworks, 2021) Global Optimization toolbox was used to implement the genetic algorithm. The population size was set to 100, which employs an elite count parameter, ensuring 5% of individuals advance to the next iteration. Additionally, for the non-elite individuals, the crossover fraction is set at 80%, with the remaining 20% set for mutation. The selected operators are scattered crossover, which combines genetic material from two parents randomly, and Gaussian mutation, which benefits combinatorial problems by injecting minor, random alterations into solution candidates, aiding the exploration of adjacent solution spaces. The mode indices were constrained between 1 and *w*, representing the lower and upper bound vectors for the algorithm, respectively. A fixed random seed ensured reproducibility across runs. The stopping criterion was either reaching a generation limit of 200 generations or achieving an average relative change in the best fitness function value over generations less than or equal to the function tolerance of 1E-8. For validation purposes, we tested various crossover and mutation functions for speed and accuracy.

3.6 Sum of ranking differences

To compare the various indicator bounds, we employed the sum of ranking differences (SRD) method (Héberger, 2010). SRD is a general, non-parametric statistical measure commonly used to assess the similarity between ranked observations. The test allows the comparison of different solutions through a reference by first performing a rank transformation on the input, and then calculating and comparing the distances between the solutions and the reference. A lower (higher) SRD value indicates a smaller (higher) difference in rankings between the indicators. We also used Spearman's rank correlation coefficient to validate exact (sharp bounds) with approximate indicator bounds. To calculate and visualize the SRD, we used the rSRD package in R (Héberger, 2010).

3.7 Practical example

An example based on a real problem is shown in Figure 2, where a selection of specific test environments was desired by an automotive supplier for their electronic brake systems project. The product can be tested with model- (MIL), software- (SIL), hardware- (HIL) and vehicle-in-the-loop (VIL), analogous to four execution modes associated with different time and (renewable-)resource, where substitution is possible. All test tasks need to be executed by considering precedence relations of the logic plan represented as a design structure matrix (DSM). Data have only been aggregated and linear scaling was applied to keep validity.

There are three resource types: developers, test engineers, and test drivers. Not all environments can be operated by all resource types, e.g., developers can work with MIL and SIL, but not with HIL and VIL. Test engineers are familiar with SIL and HIL but lack a license for vehicle and not familiar with MIL and SIL. Test drivers can test on HIL, and they are the only ones who are allowed to drive the vehicle (VIL). These constraints are coded in the original model by zeroed demands (non-executable modes), either due to incompatibility of the resource type with the task, with the test environment, or both. Furthermore, all inefficient modes were removed, if the duration is not shorter and resource demands are not less than the other modes of the same activity (Sprecher et al., 1997). Similarly, redundant modes having identical demands could be removed. Resource constraints were three unit for developers, two for test engineers and one for test drivers.

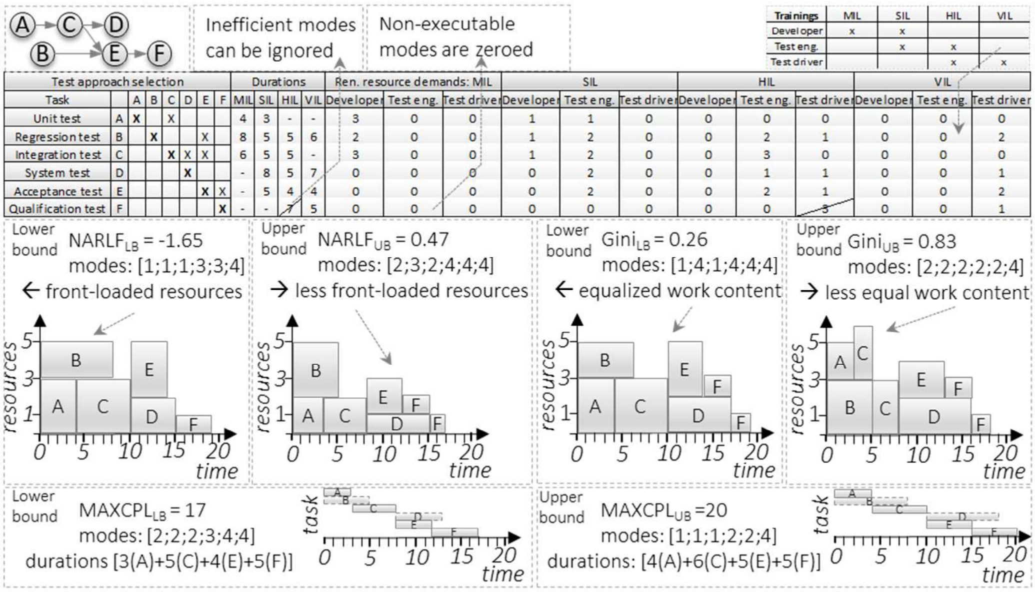


Figure 2. Various indicators are explained on a real example where test environments are selected as execution modes

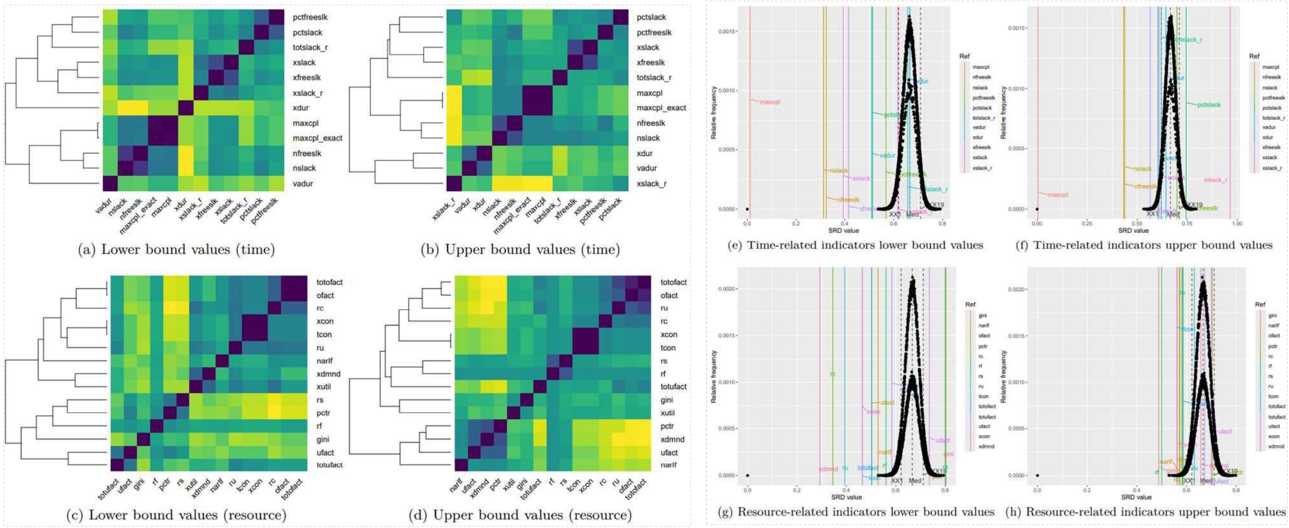
Figure 2 also illustrates selected indicators, optimized by the mode selection, to obtain lower- and upper bounds of their values. The negative value of normalized average resource loading factor (NARLF) indicates a front-loaded resource distribution throughout the test plan, when minimized for lower bound and a positive value for the maximized upper bound. Gini indicator shows a rather equal distribution of work content (i.e., duration multiplied by resource demand) on the lower bound and an inequally distributed upper bound (closer to 1). The critical project length (MAXCPL) range can be calculated both in an exact and optimized way. As expected, for the (lower) upper bounds, tasks with (shorter) longer duration were selected by the corresponding modes without considering resource demands. By solving the optimization problem, the company can identify the mode selection for each task considering the desired indicator as objectives and the consequences.

The basic idea behind ranking indicators is explained briefly for the resource-related upper bound. Normalization of indicator values is necessary as they can have different ranges. The upper bound (UB_{Norm}) values are then sorted in ascending order and assigned with ranks *1,2,…,n,* of $Rank_{UB}$. Ties, e.g., on the 3rd and 4th position, can be resolved using average ranks as (3+4)/2=3.5. The absolute values of the differences ($|Diff_{UB}|$) need to be calculated between these rankings ($Rank_{UB}$) and the arbitrary benchmark indicator's rank, $UTIL_{UB}$. The SRD values can be calculated by the sum of these differences (distances) for all project instances (only one in this example), similarly, for lower bounds and for the time-related indicators. This way, not only the SRD values order the indicators (the closer is the SRD value to zero, i.e., the closer is the ranking to the benchmark, the better is the indicator), but also groups of indicators can be recognized. The close proximity of SRD values suggests close similarity of the indicators, whereas large distance between SRD values realizes dissimilarity. Validation of the SRD method can be carried out using simulated random vectors for comparison similar to permutation tests which is explained by Héberger (2010).

4 Results

Due to the limitation of the size of this paper, only selected results are shown. First, the results of the clustering done for the projects' demand-related indicator groups are presented. The heatmap of the lower bound values for the time-related indicator's group in Figure 3(a) shows a clustering, where slack-related indicators (PCTFREESLK, PCTSLACK, TOTSLACK_R, XSLACK, XFREESLK, XSLACK_R), are connected to the average task duration (X-DUR). Slack indicators representing the number of slacks (NFREESLK, NSLACK) are placed in the group of variance in task duration (VA-DUR) with both the approximate and the exact longest critical path lengths (MAXCPL and $MAXCPL_{exact}$, respectively). Regarding upper bounds, the average task duration (X-DUR) and variance in task duration (VA-DUR) now belong to the same cluster, while the critical paths and the number of slacks share the same cluster, like the lower bound case.

From the resource perspective, on the lower bound at Figure 3(c), there are some indicators close within groups, such as the different obstruction and underutilization factors (TOTOFACT, OFACT, UFACT, TOTUFACT). Indicators considering constraints, such as XCON, TCON, RC, TOTOFACT, OFACT are close, joined by the resource use (RU) and normalized average resource factor (NARLF). Some resource-demand related indicators (XDMND, XUTIL) are also closely connected. Finally, a separate group emerges from RS, Gini, PCTR, UFACT and TOTUFACT indicators. On the upper bound of resource values, the groups show some differences in Figure 3(d). Here, NARLF, UFACT, XDMND, PCTR are related also with Gini and XUTIL. The next group is RS, RF, RC, TCON and XCON. OFACT, TOTOFACT, TOTUFACT, and RU are as well close to each other.



(a) Lower bound values (time) (b) Upper bound values (time) (e) Time-related indicators lower bound values (f) Time-related indicators upper bound values

(c) Lower bound values (resource) (d) Upper bound values (resource) (g) Resource-related indicators lower bound values (h) Resource-related indicators upper bound values

Figure 3. Heatmaps (left) and SRD rankings (right) of the time- and resource-related indicator groups

As benchmark of the SRD analysis, we defined the $MAXCPL_{exact}$ for the time-related group, and XUTIL for the resource-related indicator group instead of aggregated data (e.g., mean) to rank the values of the other indicators based on these references. If an SRD value (similarity metric) overlaps with the Gaussian curve, it is not distinguishable from random ranking. The 5% (XX1), Median (Med) and 95% (XX19) values are also indicated. Vertical axis shows relative frequencies for random rankings. Scaled SRD values are shown.

On Figure 3(e-g), the indicators marked by the vertical line that are closer to the left of the horizontal axis, has a better rank compared to the reference. For the time related indicators in Figure 3(e), it is visible that the slack group with the order of highest similarity to lower are: NSLACK, NFREESLK, XSLACK, XFREESLK, PCTSLACK. These indicators have the closest relationship with $MAXCPL_{exact}$. The variance in duration (VA-DUR) also contributes, while the others remain indifferent. For the upper bound of time-related values of Figure 3(f), only the NSLACK and NFREESLK are close to the selected reference $MAXCPL_{exact}$. Interestingly, XSLACK_R is on the right side of the Gaussian curve, which means that it is dissimilar to the reference exhibiting a reversed SRD ranking.

From the resource aspect, on the lower bound, (Figure 3(g)) XDMND, RC, RU, XCON, TCON, TOTOFACT, OFACT, RS, NARLF, RF and TOTUFACT is the relevant order of similarity from highest to lowest, compared to XUTIL as reference. Many indicators remain indifferent, while Gini, UFACT and RS are considered dissimilar. On the resource upper bound (Figure 3(h)), Gini, RF, OFACT, NARLF, XCON, RS, RC, TOTOFACT stands out, as others are indifferent. PCTR and UFACT are the least similar indicators. To compare the approximate $MAXCPL_{LB}$ and the exact, calculated $MAXCPL^{exact}_{LB}$, the correlation was also checked, finding a significant Kendall's tau coefficient of $\tau = 0.9916$, and a *p*-value of $p < .0001$ between the results. Similarly, $MAXCPL_{UB}$ and $MAXCPL^{exact}_{UB}$ shows a $\tau = 0.9968$ with a *p*-value of $p < .0001$ which demonstrates a reliable approximation of the indicator values through the optimization of the selected genetic algorithm.

4.1 Case study

To demonstrate our approach and to validate the findings, we selected a single case study of an international automotive R&D company developing software for electronic brake systems. The case satisfied the conditions for our method. To improve projects, high-management plans to extend their existing agile approach with emerging techniques, such as pair- and mob programming, only briefly introduced here (for details, see, e.g., Ståhl and Mårtensson, 2021). Pair programming involves two people working together on one workstation, enhancing collaboration in real-time. Mob programming extends this to a bigger team that works on the same task, promoting collective problem-solving.

Based on the data collected from company experts, and shown in Figure 5, the project template contains 3 subprojects with optional features (clusters are marked with dashed line) which provides the possibility to analyze a total of $2^3=8$ project variants.

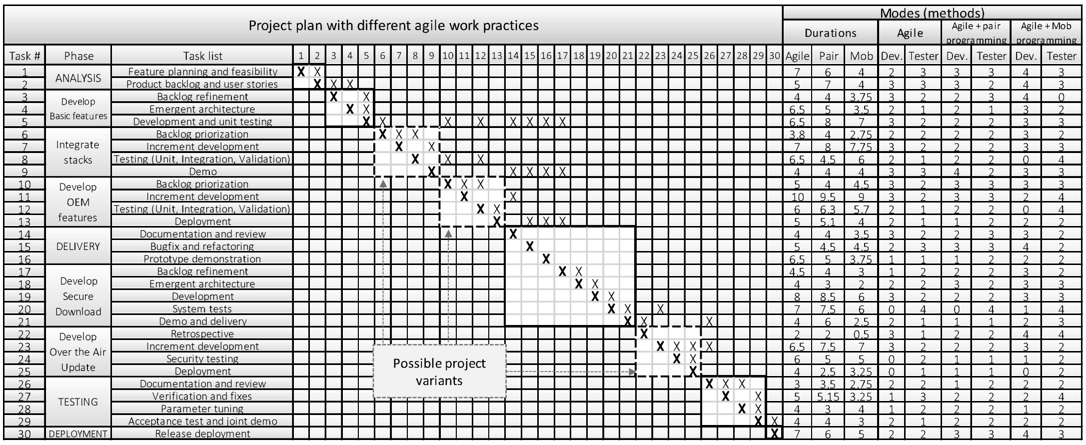


Figure 5. Different Agile work practices in a software development project

The tasks with their requirements for each 3 modes and logical dependencies are specified in Figure 5. Resource constraints for the 2 renewable resource types were 7 developers and 5 testers. The assumption is, that compared to the basic agile execution, the addition of pair programming techniques brings advantages for programming, testing, and reviews, and adds overhead to planning and deployment. Mob programming tends to be more effective for planning, reviews, and problem-solving tasks, but less efficient in development.

The case study aims to extend RQ_2 with an additional sub-question: what combination of agile technique(s) results in the lowest and highest constrainedness of resources in the project? The SRD rankings were calculated for all projects shown in Figure 6(a-d). Due to the small sample size, clustering was not feasible, but some patterns are visible in the distance matrices (lower values indicate closer proximity) shown in Figure 6(e-h).

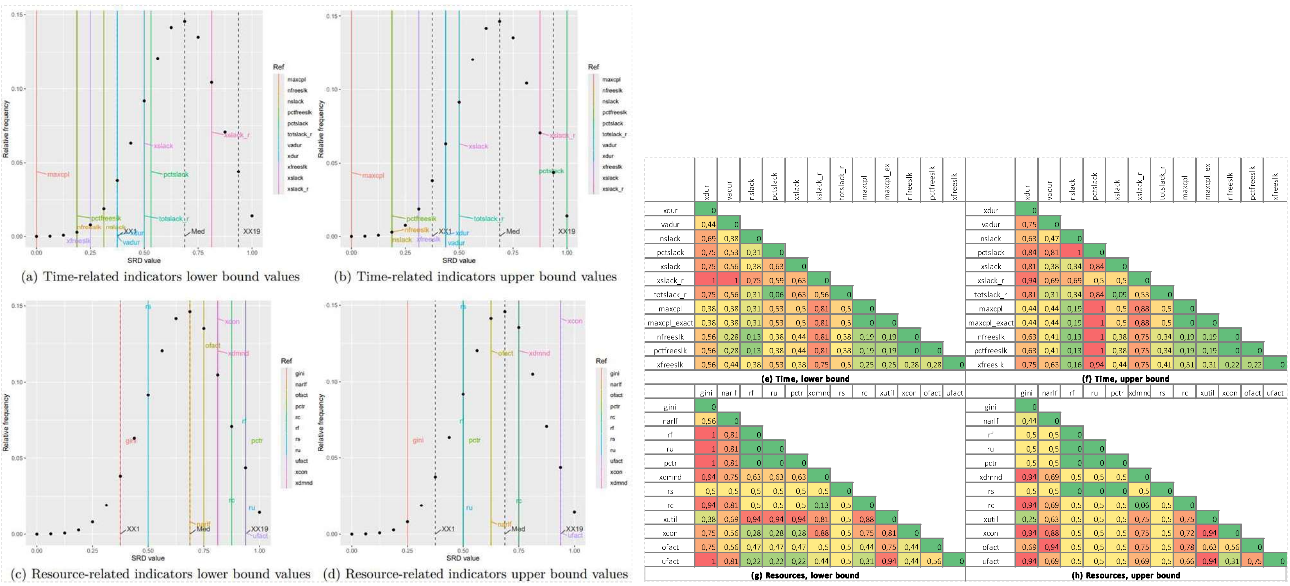


Figure 6. Indicator rankings (left) and distance matrices (right) of the software development case study

For time, PCTSLACK is the most dissimilar for the upper bound and cannot be distinguished at the lower bound from the reference. Task duration measures are closer to reference, but not conclusive for the upper bound. Some slack indicators, such as XSLACK, XSLACK-R, TOTSLACK-R also cannot be distinguished from random samples for any bounds. Interestingly, Gini stands out as the most similar measure, while UFACT and XCON as the most dissimilar measure for resource utilization (UTIL) benchmark. RF, RU, PCTR are dissimilar measures on the lower bound.

For both of time-related bounds, a group of slack-related indicators emerges. From a resource point of view, Gini, RS and NARLF are the closest for the lower bound. RU, RF, PCTR are generally close to each other. Finally, the selected modes, i.e., the agile practices selected for each task could be identified for the lower- and upper bound of the resource constrainedness (RC), using the modes solution vector. For the lower bound, the goal was to have less constrained resources, which resulted in a mixture of techniques, mob- and pair programming selected mostly for areas of expected excellence. On the upper bound, both mob and pair programming appeared overused, highly constraining the resources.

5 Discussion

The clusters formed by the indicators can be better understood and explained by examining their specific purpose and details. For instance, the resource indicators that are considering constraints, are grouped. The case study has revealed the importance of setting specific resource thresholds for each project, from a practical perspective. When the same thresholds were applied for the different project variants (e.g., because the same project team was considered during planning), these indicators exhibited a higher similarity with each other. Another group was emerging of similar resource indicators that are based on an earliest schedule. Similarly, slack indicators tend to form a group depending on their calculation method (ratio or count). These groupings can be considered in project planning, e.g., by assigning weights. In terms of ranking, the critical path length of the project shows similarities with and heavily influenced by slack values for both lower and upper bound. These results are also in line with the findings of our case study. The indicators regarding task durations were not found as conclusive as expected. Based on the results, emphasizing slack in planning-related decisions regarding total project duration seems crucial. Regarding resources, the lower bound of average resource utilization is particularly influenced by factors like average demand and resource constraints, while the Gini index (inequalities in the work content) and the resource factor are crucial for both bound considerations in project planning. The case study emphasized the dissimilarity of the under-utilization factor for both bounds.

6 Summary and conclusion

In this research, we investigated the impact of the decisions made in the mode selection process, related to the characteristics of the indicators considering alternative execution modes. This study gives insight into how the different activity execution modes impact the project characteristics represented by project indicators. It aids managerial decisions for project planning considering time and resources. We answered RQ_1 by showing that the mode selection procedure of the applied genetic algorithm has provided good approximations for the project-related indicator bounds that are calculated by the exact method. RQ_2 is answered with the result that the slack-related indicators show the strongest relationship with the project duration, i.e., the length of the longest critical path, which should be the focus of project planning decisions, especially for efficient buffer management to prevent project delays. The applied SRD method can consistently capture more complex, non-linear relationships as well. A recently applied resource indicator, the Gini coefficient turned out as an important measure for considering average resource utilization of project plans for all results. Our case study of an automotive company extended the findings of the artificial project plans and demonstrated the mode selection process to improve an agile software development project. Understanding the characteristics of projects provides a foundation for making informed decisions based on predictable and consistent indicators.

Our study suggests several promising areas for further research. These include the development and application of additional indicators as well as exploring a real project database (not yet available) that supports multiple modes. We also propose investigating the impact of mode assignment on indicators using stochastic frameworks and sensitivity analysis. Furthermore, there is a potential to explore the feasibility and performance of (multi-objective) scheduling using various indicators as objective functions.

References

Alcaraz J, Maroto C, Ruiz R (2003) Solving the multi-mode resource-constrained project scheduling problem with genetic algorithms. Journal of the Operational Research Society 54(6):614–626, ISSN 1476-9360, URL http://dx.doi.org/10.1057/palgrave.jors.2601563.

Beşikci, U., Bilge, Ü., & Ulusoy, G. (2015). Multi-mode resource constrained multi-project scheduling and resource portfolio problem. European Journal of Operational Research, 240(1), 22-31. URL https://doi.org/10.1016/j.ejor.2014.06.025.

Blazewicz J, Lenstra J, Kan A (1983) Scheduling subject to resource constraints: classification and complexity. Discrete Applied Mathematics 5(1):11–24, ISSN 0166-218X, URL https://doi.org/10.1016/0166-218X(83)90012-4

Boctor F (1996a) An adaption of the simulated annealing for solving resource-constrained project scheduling problems. International Journal of Production Research 34:2335–2351.

Boctor FF (1990) Some efficient multi-heuristic procedures for resource-constrained project scheduling. European Journal of Operational Research 49(1):3–13, ISSN 0377-2217, URL https://doi.org/10.1016/0377-2217(90)90116-S

Boctor FF (1996b) A new and efficient heuristic for scheduling projects with resource restrictions and multiple execution modes. European Journal of Operational Research 90(2):349–361, ISSN 0377-2217, URL http://dx.doi.org/https://doi.org/10.1016/0377-2217(95)00359-2.

Boctor FF (2004) Other benchmarks. Available at: http://www.om-db.wi.tum.de/psplib/ dataob.html, accessed 03, May 2024.

Bouleimen K, Lecocq H (2003) A new efficient simulated annealing algorithm for the resource-constrained project scheduling problem and its multiple mode version. European Journal of Operational Research 149(2):268–281, ISSN 0377-2217, URL http://dx.doi.org/https:// doi.org/10.1016/S0377-2217(02)00761-0, sequencing and Scheduling.

Coelho, J., Vanhoucke, M. (2020). Going to the core of hard resource-constrained project scheduling instances. Computers & Operations Research, 121, 104976. URL https://doi.org/10.1016/j.cor.2020.104976

Chakrabortty RK, Abbasi A, Ryan MJ (2020) Multi-mode resource-constrained project scheduling using modified variable neighborhood search heuristic. International Transactions in Operational Research 27(1):138–167, URL http://dx.doi.org/https://doi.org/10.1111/itor. 12644.

Changchun L, Xi X, Canrong Z, Qiang W, Li Z (2018) A column generation based distributed scheduling algorithm for multi-mode resource constrained project scheduling problem. Computers & Industrial Engineering 125:258–278, ISSN 0360-8352, URL https://doi.org/10.1016/j.cie.2018.08.036.

Danilovic M, Browning TR (2007) Managing complex product development projects with design structure matrices and domain mapping matrices. International journal of project management 25(3):300–314, URL http://dx.doi.org/doi.org/10.1016/j.ijproman.2006.11.003.

Davis EW (1975) Project network summary measures constrained-resource scheduling. AIIE Transactions 7(2):132–142, URL http://dx.doi.org/10.1080/05695557508974995.

Demeulemeester EL, Vanhoucke M, Herroelen W (2003) Rangen: A random network generator for activity-on-the-node networks. Journal of Scheduling 6(1):17–38, URL http://dx.doi.org/ 10.1023/A:1022283403119.

Drexl A, Gruenewald J (1993) Nonpreemptive multi-mode resource-constrained project scheduling. IIE Transactions 25(5):74–81, URL http://dx.doi.org/10.1080/07408179308964317.

Elmaghraby S (1977) Activity Networks: Project Planning and Control by Network Models. A Wiley-Interscience publication (Wiley), ISBN 9780471238614, URL https://books.google.com/books?id=8nZRAAAAMAAJ.

Fernandes Muritiba AE, Rodrigues CD, Araùjo da Costa F (2018) A path-relinking algorithm for the multi-mode resource-constrained project scheduling problem. Computers & Operations Research 92:145–154, ISSN 0305-0548, URL http://dx.doi.org/https://doi.org/ 10.1016/j.cor.2018.01.001.

Gerhards P, Stürck C, Fink A (2017) An adaptive large neighbourhood search as a matheuristic for the multi-mode resource-constrained project scheduling problem. European Journal of Industrial Engineering 11(6):774–791, URL http://dx.doi.org/10.1504/EJIE.2017.089101.

Guo, W., Vanhoucke, M., Coelho, J., & Luo, J. (2021). Automatic detection of the best performing priority rule for the resource-constrained project scheduling problem. Expert systems with applications. URL https://doi.org/10.1016/j.eswa.2020.114116.

Habibi F, Barzinpour F, Sadjadi S (2018) Resource-constrained project scheduling problem: review of past and recent developments. Journal of project management 3(2):55–88, URL http: //dx.doi.org/10.5267/j.jpm.2018.1.005.

Hartmann S (2001) Project scheduling with multiple modes: A genetic algorithm. Annals of Operations Research 102(1):111–135, ISSN 1572-9338, URL http://dx.doi.org/10.1023/A: 1010902015091.

Hartmann S, Briskorn D (2022) An updated survey of variants and extensions of the resource-constrained project scheduling problem. European Journal of Operational Research 297(1):1– 14, URL http://dx.doi.org/https://doi.org/10.1016/j.ejor.2021. 05.004.

Hartmann S, Drexl A (1998) Project scheduling with multiple modes: A comparison of exact algorithms. Networks 32(4):283–297, URL http://dx.doi.org/https://doi.org/10.1002/ (SICI)1097-0037(199812)32:43.0.CO;2-I.

Hartmann S, Sprecher A (1996) A note on hierarchical models for multi-project planning and scheduling. European Journal of Operational Research 94(2):377–383, ISSN 0377-2217, URL http://dx.doi.org/https://doi.org/10.1016/0377-2217(95)00158-1.

Héberger K (2010) Sum of ranking differences compares methods or models fairly. TrAC Trends in Analytical Chemistry 29(1):101–109, ISSN 0165-9936, URL http://dx.doi.org/https: //doi.org/10.1016/j.trac.2009.09.009.

Józefowska J, Mika M, Różycki R, Waligóra G, Weglarz J (2001) Simulated annealing for multi-mode resource-constrained project scheduling. Annals of Operations Research 102(1):137–155, ISSN 1572-9338, http://dx.doi.org/10.1023/A:1010954031930.

Kelley Jr JE, Walker MR (1959) Critical-path planning and scheduling. Papers presented at the December 1-3, 1959, eastern joint IRE-AIEE-ACM computer conference, 160–173, URL http: //dx.doi.org/https://doi.org/10.1145/1460299.1460318.

Knotts G, Dror M, Hartman BC (2000) Agent-based project scheduling. IIE Transactions 32(5):387– 401, ISSN 1573-9724, URL http://dx.doi.org/10.1023/A:1007666324223.

Kolisch R (1995) Project Scheduling under Resource Constraints—Efficient Heuristics for Several Problem Cases (Heidelberg, Germany: Physica-Verlag).

Kolisch R, Drexl A (1997) Local search for nonpreemptive multi-mode resource-constrained project scheduling. IIE Transactions 29(11):987–999, URL http://dx.doi.org/10.1080/ 07408179708966417.

Kolisch R, Sprecher A, Drexl A (1995) Characterization and generation of a general class of resource-constrained project scheduling problems. Management Science 41(10):1693–1703, URL http: //dx.doi.org/10.1287/mnsc.41.10.1693.

Kosztyán ZT, Novák G, Jakab R, Szalkai I, Hegedűs C (2023) A matrix-based flexible project-planning library and indicators. Expert Systems with Applications 216:119472, URL http: //dx.doi.org/10.1016/j.eswa.2022.119472.

Kosztyán ZT (2020) An exact algorithm for the flexible multilevel project scheduling problem. Expert Systems with Applications 158:113485, ISSN 0957-4174, URL https://doi.org/10.1016/j.eswa.2020.113485.

Lova A, Tormos P, Barber F (2006) Multi-mode resource constrained project scheduling: Scheduling schemes, priority rules and mode selection rules. Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial 10(30):69–86.

Lova A, Tormos P, Cervantes M, Barber F (2009) An efficient hybrid genetic algorithm for scheduling projects with resource constraints and multiple execution modes. International Journal of Production Economics 117(2):302–316, ISSN 0925-5273, URL http://dx.doi.org/https: //doi.org/10.1016/j.ijpe.2008.11.002.

Mastor AA (1970) An experimental investigation and comparative evaluation of production line balancing techniques. Management Science 16(11):728–746, URL http://dx.doi.org/10. 1287/mnsc.16.11.728.

Mathworks M (2021) Matlab 2021a. The MathWorks: Natick, MA, USA.

Mori M, Tseng CC (1997) A genetic algorithm for multi-mode resource constrained project scheduling problem. European Journal of Operational Research 100(1):134–141, ISSN 0377-2217, URL https://doi.org/10.1016/S0377-2217(96)00180-4.

Nonobe K, Ibaraki T (2002) Formulation and Tabu Search Algorithm for the Resource Constrained Project Scheduling Problem, 557–588 (Boston, MA: Springer US), ISBN 978-1-4615-1507-4, URL http://dx.doi.org/10.1007/978-1-4615-1507-4_25.

Noori S, Taghizadeh K (2018a) Multi-mode resource constrained project scheduling problem: a survey of variants, extensions, and methods. International Journal of Industrial Engineering & Production Research 29(3):293–320, URL http://dx.doi.org/10.22068/ijiepr.29.3.293.

Patterson JH (1976) Project scheduling: The effects of problem structure on heuristic performance. Naval Research Logistics Quarterly 23(1):95–123, URL http://dx.doi.org/10.1002/nav. 3800230110.

Patterson JH, Słowiński R, Talbot F, Weglarz J (1989) An algorithm for a general class of precedence and resource constrained scheduling problems. Advances in project scheduling, 3–28 (Elsevier).

Peteghem VV, Vanhoucke M (2010) A genetic algorithm for the preemptive and non-preemptive multi-mode resource-constrained project scheduling problem. European Journal of Operational Research 201(2):409–418, ISSN 0377-2217, URL http://dx.doi.org/https://doi.org/10. 1016/j.ejor.2009.03.034.

Peteghem VV, Vanhoucke M (2014) An experimental investigation of metaheuristics for the multi-mode resource-constrained project scheduling problem on new dataset instances. European Journal of Operational Research 235(1):62 – 72, ISSN 0377-2217, URL http://dx.doi.org/ 10.1016/j.ejor.2013.10.012.

Speranza M, Vercellis C (1993) Hierarchical models for multi-project planning and scheduling. European Journal of Operational Research 64(2):312–325, ISSN 0377-2217, URL http://dx.doi.org/https://doi.org/10.1016/0377-2217(93)90185-P

Sprecher A (1994) Generation of Instances by ProGen. Resource-Constrained Project Scheduling, 70–90 (Springer), URL http://dx.doi.org/10.1007/978-3-642-48397-4_6.

Sprecher A, Drexl A (1998a) Multi-mode resource-constrained project scheduling by a simple, general and powerful sequencing algorithm. European Journal of Operational Research 107(2):431–450, ISSN 0377-2217, URL http://dx.doi.org/https://doi.org/10.1016/S0377-2217(97)00348-2.

Sprecher A, Drexl A (1998b) Multi-mode resource-constrained project scheduling by a simple, general and powerful sequencing algorithm supported by the deutsche forschungsgemeinschaft. European Journal of Operational Research 107(2):431–450, ISSN 0377-2217, URL http://dx.doi.org/https://doi.org/10.1016/S0377-2217(97)00348-2.

Sprecher A, Hartmann S, Drexl A (1997) An exact algorithm for project scheduling with multiple modes. Operations-Research-Spektrum 19(3):195–203, ISSN 1436-6304, URL http://dx.doi. org/10.1007/BF01545587.

Ståhl, D., & Mårtensson, T. (2021). Mob programming: From avant-garde experimentation to established practice. Journal of Systems and Software, 180, 111017. URL https://doi.org/10.1016/j.jss.2021.111017.

Steward D (1981) The design structure matrix: A method for managing the design of complex systems. IEEE Transactions on Engineering Management 28(1981):pp.

Stürck, C., & Gerhards, P. (2018). Providing lower bounds for the multi-mode resource-constrained project scheduling problem. In Operations Research Proceedings 2016, (pp. 551-557). Springer. URL http://dx.doi.org/10.1007/978-3-319-55702-1_73.

Słowiński R (1980) Two approaches to problems of resource allocation among project activities — a comparative study. Journal of the Operational Research Society 31(8):711–723, URL http://dx.doi.org/10.1057/jors.1980.134.

Słowiński R, Soniewicki B, Weglarz J (1994) Dss for multiobjective project scheduling. European Journal of Operational Research 79(2):220–229, ISSN 0377-2217, URL https://doi.org/10.1016/0377-2217(94)90353-0.

Talbot FB (1982) Resource-constrained project scheduling with time-resource tradeoffs: The non-preemptive case. Management Science 28(10):1197–1210, URL http://dx.doi.org/10.1287/ mnsc.28.10.1197.

Tavares LV (1999) Advanced models for project management, volume 16 (Springer Science & Business Media), URL http://dx.doi.org/10.1007/978-1-4419-8626-9.

Van Eynde R, Vanhoucke M (2020) Resource-constrained multi-project scheduling: benchmark datasets and decoupled scheduling. Journal of Scheduling 23(3):301–325, URL http://dx. doi.org/10.1007/s10951-020-00651-w.

Van Eynde, R., Vanhoucke, M. & Coelho, J. On the summary measures for the resource-constrained project scheduling problem. Ann Oper Res 337, 593–625 (2024). URL https://doi.org/10.1007/s10479-023-05470-8

Vanhoucke M, Coelho J, Debels D, Maenhout B, Tavares LV (2008) An evaluation of the adequacy of project network generators with systematically sampled networks. European Journal of Operational Research 187(2):511–524, URL http://dx.doi.org/doi.org/10.1016/j.ejor. 2007.03.032.

Weglarz J, Józefowska J, Mika M, Waligóra G (2011) Project scheduling with finite or infinite number of activity processing modes – a survey. European Journal of Operational Research 208(3):177–205, ISSN 0377-2217, URL http://dx.doi.org/https://doi.org/10.1016/j. ejor.2010.03.037.

Zhang H, Tam CM, Li H (2006) Multimode project scheduling based on particle swarm optimization. Computer-Aided Civil and Infrastructure Engineering 21(2):93–103, URL http://dx.doi. org/https://doi.org/10.1111/j.1467-8667.2005.00420.x.

Özdamar L, Ulusoy G (1994) A local constraint based analysis approach to project scheduling under general resource constraints. European Journal of Operational Research 79(2):287–298, ISSN 0377-2217, URL http://dx.doi.org/https://doi.org/10.1016/0377-2217(94)90359-X.

Contact: Gergely L. Novák, Continental Automotive Hungary Ltd., Házgyári street 6-8., H-8200, Veszprém, Hungary, +36-88-540-304, gergely.novak@continental.com, www.continental.com