SoftwareX 17 (2022) 100973



Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx



Original software publication

MFPP: Matrix-based flexible project planning



Zsolt T. Kosztyán

University of Pannonia, Department of Quantitative Methods, Hungary

ARTICLE INFO

Article history:
Received 2 March 2021
Received in revised form 4 January 2022
Accepted 6 January 2022

Keywords:
Flexible project planning
Agile
Extreme
Hybrid project management
Scheduling
Risk analysis

ABSTRACT

Although flexible project approaches, such as agile, hybrid, and extreme project planning methods, are very popular in software project planning, they still lack algorithmic and (especially) software support. The proposed matrix-based flexible project planning (MFPP) software tool fills this gap. This software tool is freely available and implements recent work on flexible project planning/scheduling and risk analysis.

© 2022 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Code metadata

Current code version	v 1.0.2
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-21-00050
MATLAB central link	https://www.mathworks.com/matlabcentral/profile/authors/14230519
Legal code license	GNU General Public License v3.0
Code versioning system used	git
Software coding languages, tools, and services used	MATLAB
Compilation requirements, operating environments & dependencies	MATLAB r2015a and later releases, Global Optimization Toolbox
If available, link to developer documentation/manual	
Support email for questions	kzst@gtk.uni-pannon.hu

1. Motivation and significance

While project planning and scheduling algorithms have been extensively studied in the past six decades [1,2], these algorithms, except for those in a few papers (such as the examples in Refs. [3,4]), assume fixed dependencies between tasks and neglect the task completion priority. Flexible methods such as agile, extreme, and hybrid project management approaches enable the consideration of the (relative) importance or (relative) priorities with regard to completing tasks; thus, the sets of mandatory and supplementary tasks can be specified. In addition, flexible dependencies allow project managers to implement serial and parallel completion of pairs of tasks [3]. Since every decision to include/exclude supplementary tasks and every decision of parallel/serial completion increase the number of possible project structures by a factor of two, there is usually no chance to determine all possible project structures to select an adequate project plan. Nevertheless, each possible project execution can have a different scope, duration, and task demands. In addition, according to a target function, project managers must select the best project structure without determining and ranking all project structures. Furthermore, considering flexibility can reduce project demands, and how better (i.e., shorter, lower budgeted) project plans can be identified is important to determine. Finally, the use of flexible project management approaches, which consider flexible technology, mitigates risks compared to traditional approaches. The motivation to specify the new software tool for project managers is twofold. First, a freely available software tool is proposed to handle flexible technologies, such as flexible dependencies and task completions, to support flexible project management approaches. Second, a simulation environment is offered to help decision makers select adequate project management approaches based on their scheduling and risk mitigation performance. To the best of our knowledge, this is the first freely available software application tool that models both traditional and flexible project management approaches. This is also the first tool that can compare and compete with both traditional

E-mail address: kzst@gtk.uni-pannon.hu.

https://doi.org/10.1016/j.softx.2022.100973
2352-7110/© 2022 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Table 1
Comparison of various traditional and flexible approaches.
Source: [6].

Approaches	Features		
	Project structure	New tasks	Multiple modes
Traditional (TPMa)	Fixed	Not allowed	Handled
Agile (APMa)	Flexible	Not allowed	Not handled
Extreme (XPMa)	Flexible	Allowed	Not handled
Hybrid (HPMa)	Flexible	Allowed	Handled

and flexible project management approaches in terms of their scheduling and risk mitigation performance.

In terms of scheduling, traditional time–cost trade-off problems [1,2] support traditional project management approach (TPMa) and are usually not, or only slightly, considered in agile project management approaches (APMa) [5]). In addition, other flexible approaches, such as the extreme project management approach (XPMa), allow for new, unplanned tasks in response to changes in customers' desires. The last explored approach is the HPMa, which has a flexible structure but can apply traditional trade-off methods and/or multimode task completion (or alternative technology) [5,6].

Table 1 compares the various project management approaches in terms of scheduling. The first feature of the comparison is the project structure, where a fixed project structure means that the structure of the project plan cannot be changed during project completion: a new project plan must be specified to complete remaining tasks. A flexible structure means that the structure of the project can be reorganized considering the priorities of task completions based on customer preferences. The second comparison feature is the inclusion of new tasks: if it is allowed, new tasks can be included during the completion of the (sub)project; otherwise, new tasks can be considered only in the next (sub)project (i.e., sprint). The last comparison feature is how multiple completion modes (or in other words alternative technology) are handled. If multiple completion modes (technology) are not involved, then only a single completion mode can be defined for each task, while if multiple modes are allowed, the project manager (in this case, the agent) can choose the appropriate technology for completing the task.

Flexible approaches, such as agile, extreme and hybrid project management, allow for flexible dependencies between tasks. Because of such flexible technologies, different completion modes, such as serial and parallel completion, can be utilized, and the realization of a given project depends on its constraints [5]. Neither the TPMa nor APMa allow new tasks to be added to a running project: TPMa and APMa postpone these tasks to the next project or sprint. By contrast, the extreme and hybrid approaches can include new tasks if they are within budget. Both agile and extreme approaches consider single completion modes, that is, one possible technology at a time; while traditional and hybrid approaches implement predefined technologies [7–9].

Both empirical [10] surveys and simulation results [6,11] have shown that flexible project scheduling can provide more feasible project plans with better scheduling performance than those output by traditional approaches [6]. Scheduling approaches (as agents) can imitate decision makers; hence, the risk mitigation properties of project management approaches can also be investigated [11].

The first goal of the proposed software tool (MFPP) is to implement traditional agents, as in TPMa, and specify flexible agents, as in APMa, XPMa, and HPMa, to solve project scheduling problems to compare their performance in scheduling and risk mitigation.

Various models are available for project stakeholders; their usefulness depends on the alignment of the information and tools that they provide with the objectives of the user [12]. Network-based project risk management techniques can simulate and evaluate changes in parameters (i.e., task duration, cost and resource demands) of a single project [13]; however, these models cannot precisely represent changes in customer needs. New customer demands may indicate new tasks and/or new (sub)projects, which may affect the project parameters (such as time/cost/resource demands) and change the project structure. *Thus, matrix-based methods are used in this study to model all these changes in parameters and requirements.*

To the best of our knowledge, the use of matrix-based methods for project planning started with [14], who extended the original design structure matrix method of [15] and called it the dependency structure matrix (keeping the DSM acronym). A cell of the dependency structure matrix indicates whether there is a relation between two project tasks that are assigned to the pertinent row and column. Although this binary DSM method is used for scheduling [16,17], resource allocation [18] and risk management [19], it is not suitable for modeling projects where both parallel and sequential implementation of two tasks is feasible. The numerical DSM (NDSM), where the degree of (i.e., uncertain or flexible) dependency between two tasks [20,21] can be expressed as a numerical value, addresses this problem. In the case of flexible dependencies in a flexible project, tasks can often be completed in both sequential and parallel orders. This situation can be modeled by another matrix-based project planning method called the stochastic network planning method (SNPM), which was introduced by Ref. [3]. This acronym indicates that one model can represent several possible network plans (i.e., project structures). The project expert matrix (PEM) is an extension of SNPM [3]. In this case, we also include the probability, relative importance or priority values on the diagonal of the PEM. Here, the numbers (i.e., score value of task completion) on the diagonal can indicate the importance, (relative) priority, or probability of task completion.

In addition to task dependencies, other issues must be addressed in project planning and management. Time, cost and resource demands and constraints can be represented as new domains (i.e., submatrices), and their relations to the tasks and one another can be handled with a domain mapping matrix [22]. The DMM makes it possible to consider several properties of the tasks in an optimization problem. [3] created a project domain matrix (PDM), which is a domain mapping extension of the project expert matrix (PEM). The PDM is similar to a specialized DMM, and the PEM is an extended NDSM. The PEM can contain numerical values on the diagonal, so it can represent both uncertain dependencies and uncertain task realizations.

The second goal of the proposed software tool (MFPP) is to specify a unified matrix-based model that can handle both traditional and flexible project plans.

A project plan is modeled by a PDM, which has 3 mandatory domains, namely, logic domain (LD), time domain (TD) and cost domain (CD) [3], and two supplementary domains, namely, quality domain (QD) and resource domain (RD) [6] (see Table A.3). The proposed matrix-based flexible project planning (MFPP) tool implements a genetic algorithm-based version of the method in Refs. [3,5,6]. Since all agents must decide which tasks and dependencies must be included in the project and which completion mode to implement, the result contains neither flexible dependencies nor supplementary tasks or different completion modes.

In addition to minimizing project duration, several other target functions can be specified. In this MFPP tool, the following target functions and their combinations are considered:

TPT → min**:** minimize the project duration (total project time, TPT)

TPC → min**:** minimize the project cost (total project cost, TPC)

TPQ → max**:** maximize the project quality (total project quality, TPQ)

TPS → max**:** maximize the project score/scope (total project score/scope, TPS)

These results are also represented in a matrix (PSM) that has 4 mandatory domains, including the LD, TD, CD and scheduling domain (SD), where the scheduled start time (SS) is presented, and 2 supplementary domains, namely, the QD and RD. The TD, CD, and QD are still vectors.

This paper proposes a new software application for project managers to schedule both traditional and flexible project plans. In addition, it compares the project management approaches by their scheduling performances and risk mitigation for the decision-makers to choose the best project management approach. Furthermore, in the case of analyzing project libraries, scholars can analyze project management approaches and their resilience to risks.

The MFPP proposed project management and risk analysis tool:

- models traditional and flexible projects via a unified matrix-based model;
- models traditional, agile, extreme and hybrid project management approaches (see Table 1) as project management agents;
- handles various target functions using genetic algorithms;
- handles various trade-offs and multimodal resource allocation problems, which can be solved for both traditional and flexible project planning approaches;
- considers different kinds of risk effects such as scheduling uncertainties, shock effects, and structural changes.
- compares agents by their scheduling performances and risk mitigation.

2. Software description

The key functions are shown in a *DEMO.m* file. Section 2.2 follows the procedure from problem generation to scheduling to risk analysis.

2.1. Software architecture

The MFPP software is written in MATLAB. The scheduling components also require the Global Optimization Toolbox. The functions work in MATLAB r2015a or later releases.

Fig. 1 summarizes the processing pipeline of the MFPP software. In the first step, a PDM is generated or imported from a project database. The logic domain of the PDM is an upper triangular matrix because we assume that the tasks in the project are topologically ordered (there is no backward dependency). If the project plan contains cycles or iterations, we use Ref. [3] to resolve the cycles. The next stage (block 2) specifies the constraints and target function. The following stage (block 3) is the scheduling stage, where three kinds of approaches/agents can be used to specify the scheduled project plan represented by the PSM. The last block in the scheduling part is to check the feasibility and evaluate the scheduling performance. If there is no risk analysis, the project schedule is finished. Outputs are PSM matrices by solvers (i.e., agents), which contain logic plans, task demands, and scheduled starts of tasks.

The changing constraints and changing demands are separated. All risk effects are optional. Both the original project plan (PDM) and the constraints can be modified in the risk analysis process. In addition, the demands and constraints can be modified throughout the entire scheduling process, and they can change the scheduled projects modeled by PSM matrices.

2.2. Software functionalities

This section shows how to generate (see Section 2.2.1), schedule (see Section 2.2.2) and analyze (see Section 2.2.3) a flexible project plan.

2.2.1. Generate a PDM

The first function shows how to generate a PDM.

Usage:

$PDM = generatepdmq_scale(N, ff, cf, mTD, mCD, mRD, w, nR, nW)$

Output:

- PDM=[LD,TD,CD,QD,RD]: Project domain matrix, where the PDM is an $(N + nW)$ by $(N + nW)$ upper triangular matrix in the logic domain, N is the number of planned tasks, and nW is the maximum number of unplanned tasks.
- TD is an $(N+nW)$ by w matrix containing the task duration, where w is the number of completion modes.
- CD is an $(N + nW)$ by w matrix of the cost demands.
- QD is an $(N + nW)$ by w matrix of the quality parameters.
- RD is an $(N + nW)$ by $w \cdot nR$ matrix containing the resource demands, where nR is the number of resources.

Input:

- N=number of tasks
- ff=flexibility factor (between 0 to 1)
- cf=connectivity factor (1, …) ∼ number of parallel paths
- mT=max value of TD (must be positive)
- mC=max value of CD (must be positive)
- mR=max value of RD (must be positive)
- w=number of completion modes (1,2,..)
- nW=number of possible extra tasks (0,1,2,..)
- *scale*=structural parameter (1.0-2.0)

Since the QD is optional, if the QD is not required, the correct usage is:

$PDM = generatepdm_scale(N, ff, cf, mTD, mCD, mRD, w, nR, nW)$.

If there are no resources, $nR = 0$.

Example code:

$PDM = generatepdmq_scale(30, 0.05, 1, 20, 30, 20, 2, 2, 2, 1.8)$

In this case, the number of tasks is 30, but 2 unplanned tasks are possible. There is only one critical path ($cf = 1$), and the number of modes and number of resources are both 2. The scale parameter influences the number of connections among the tasks (see Ref. [5]).

2.2.2. Project schedule

There are three kinds of genetic algorithm-based project schedule approaches: TPMa, APMa, and HPMa. The implementation of the HPMa is shown, but TPMa and APMa have identical functionalities.

Output:

PSM: an N by M matrix containing the calculated project plan. $M = N + 2 + nR + 1$ if the quality parameter is not specified, and $M = N + 3 + w + nR + 1$ if quality parameters are specified. The PSM contains the logic domain, time and cost demands, and (depending on the problem selection that it contains) quality and/or resource demands. Finally, it contains the scheduled starts of the tasks: PSM=[DSM, TD, CD, QD, RD, SST].

- DSM is an N by N upper triangular binary matrix in the calculated logic domain.

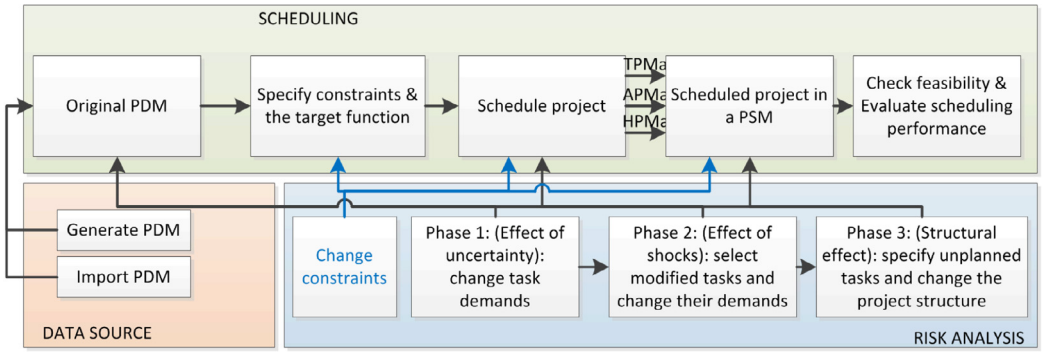


Fig. 1. Pipeline of the scheduling and risk analysis process.

Logic Domain				TD	CD	QD	Resource Domain		SD
PSM	A	B	C	t	c	q	r_1	r_2	SS
A	X	X		2	8	0.7	3	4	0
B		X	X	3	6	0.9	2	5	2
C			X	5	7	0.8	3	3	5

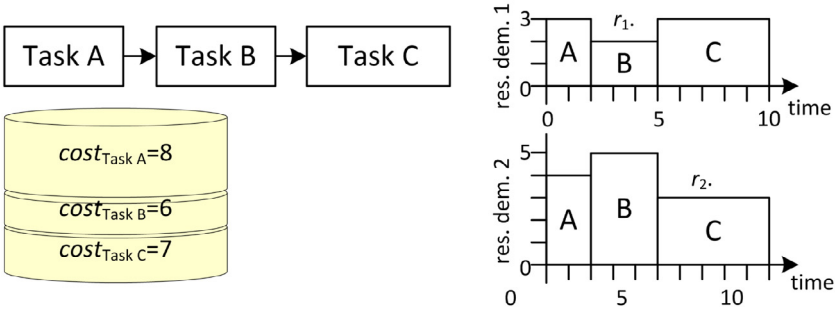


Fig. A.2. The result of TPMa.

Logic Domain			TD	CD	QD	Resource Domain		SD
PDM	A	B	t	c	q	r_1	r_2	SS
A	X		3	6	0.8	2	3	0
B		X	4	5	0.9	2	4	0

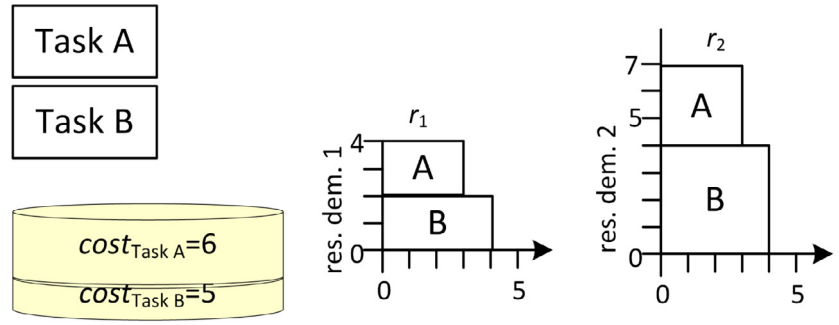


Fig. A.3. The result of APMa.

Table A.3
An example of a flexible project plan represented by a PDM matrix.

Logic Domain				**Task Domain**		**Cost Domain**		**Quality Domain**		**Resource Domain**			
PDM	**A**	**B**	**C**	t_1	t_2	c_1	c_2	q_1	q_2	r_{11}	r_{12}	r_{21}	r_{22}
A	1.0	0.8		3	2	6	8	0.8	0.7	2	3	3	4
B		1.0	0.6	4	3	5	6	0.9	0.9	2	2	4	5
C			0.7	5	6	7	7	0.8	0.9	3	3	3	3

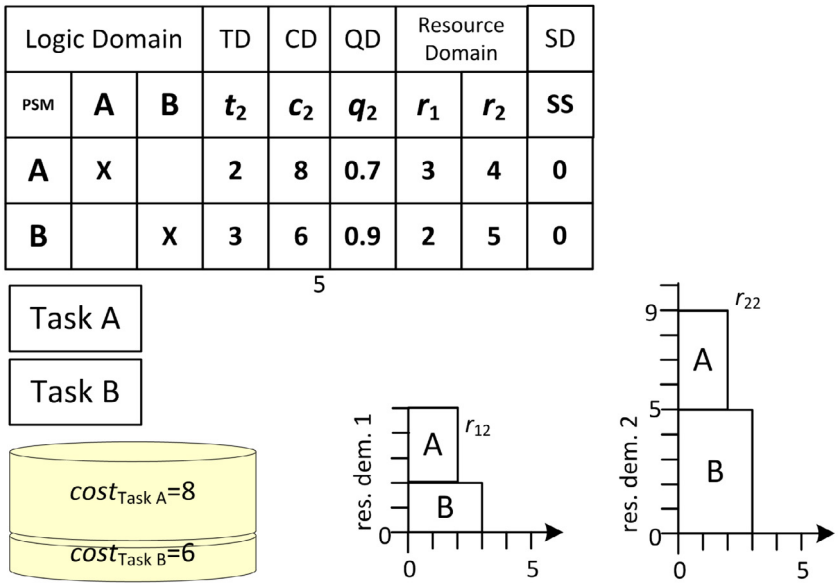
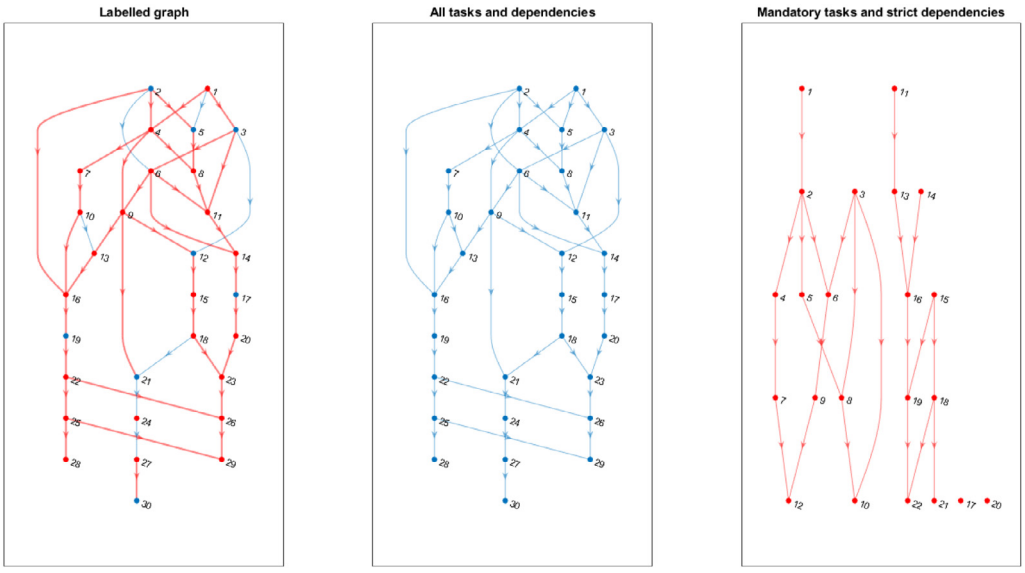


Fig. A.4. The result of HPMa.

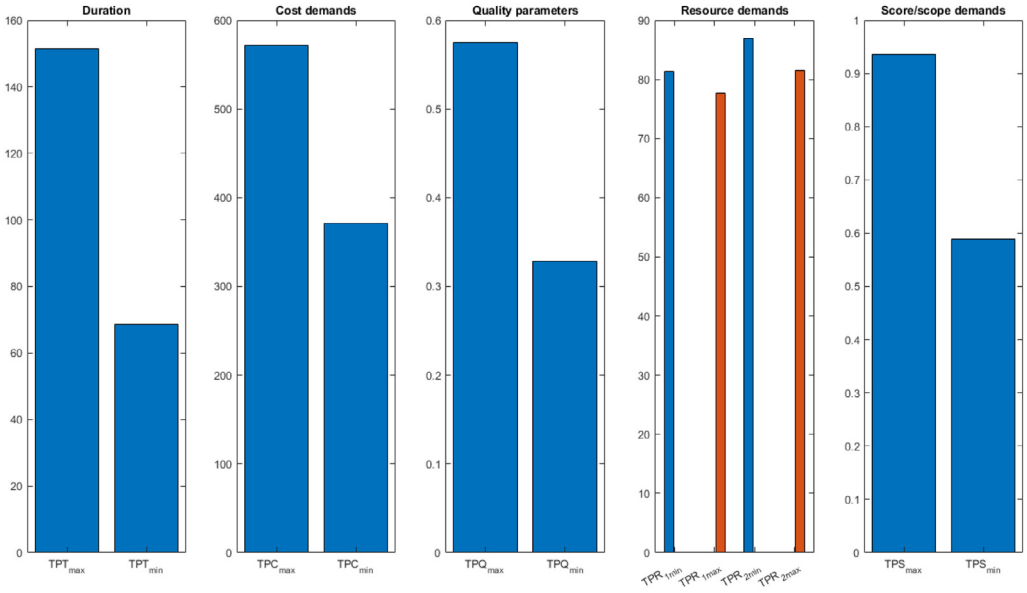
- The TD is an N by 1 column vector of task durations. The CD is an N by 1 column vector of cost demands depending on the problem selection.
- The QD is an N by 1 column vector of quality demands. (Optional)
- The RD is an N by nR matrix of resource demands depending on whether resource allocation is performed.
- The SST is an N by 1 column vector that contains the scheduled start times of the tasks if resource allocation is applied, and the EST is an N by 1 column vector of the earliest start times of the tasks if resource demands are not considered.

Inputs:

- PDM=[PEM,TD,CD,QD,RD], where PEM is an $N \times N$ upper triangular matrix in the logic domain,
  - TD is an N by w matrix of task durations,
  - CD is an N by w matrix of cost demands (depending on the problem selection),
  - QD is an N by w matrix of quality parameters, and
  - RD is an N by $w \cdot nR$ matrix of resource demands.
- Constraints: a $3 + nR + 1$-element row vector of [Ct,Cc,Cq, CR,Cs] values if the quality parameters and quality constraint are specified, where
  - Ct is the time constraint (max constraint)
  - Cc is the cost constraint (max constraint)
  - Cq is the quality constraint (min constraint, optional)
  - CR are the resource constraints (max constraints)
  - Cs is the score constraint (min constraint)
- Select: 1-22 scalars for problem selection
  1. APSP: Agile project scheduling problem
  2. APSPq: Agile project scheduling problem with quality parameters
  3. HDTCTP: Hybrid discrete time–cost trade-off problem
  4. HCTCTP: Hybrid continuous time–cost trade-off problem
  5. HDTQCTP: Hybrid discrete time–quality–cost trade-off problem
  6. HCTQCTP: Hybrid continuous time–quality–cost trade-off problem
  7. RC-APSP: Resource-constrained agile project scheduling problem
  8. PO-APSP: Pareto-optimal resource-allocated agile project scheduling problem
  9. RC-HDTCTP: Resource-constrained hybrid discrete time–cost trade-off problem
  10. RC-HCTCTP: Resource-constrained hybrid continuous time–cost trade-off problem
  11. PO-HDTCTP: Pareto-optimal resource-allocated hybrid discrete time–cost trade-off problem
  12. PO-HCTCTP: Pareto-optimal resource-allocated hybrid continuous time–cost trade-off problem
  13. RC-APSPq: Resource-constrained agile project scheduling problem with quality parameters
  14. PO-APSPq: Pareto-optimal resource-allocated agile project scheduling problem with quality parameters
  15. RC-HDTQCTP: Resource-constrained hybrid discrete time–quality–cost trade-off problem
  16. PO-HDTQCTP: Pareto-optimal resource-allocated hybrid discrete time–quality–cost trade-off problem
  17. RC-HCTQCTP: Resource-constrained hybrid continuous time–quality–cost trade-off problem
  18. PO-HCTQCTP: Pareto-optimal resource-allocated hybrid continuous time–quality–cost trade-off problem



(a) Logic network.



(b) Minimal/maximal project demands.

Fig. B.5. Flexible logic network and minimal/maximal project demands.

19. H-PO-DMRCPSP: Hybrid, discrete version of the Pareto-optimal multimodal resource-constrained project scheduling problem without quality parameters
20. H-PO-DMRCPSP: Hybrid, continuous version of the Pareto-optimal multimodal resource-constrained project scheduling problem without quality parameters
21. H-PO-DMRCPSP-q: Hybrid, discrete version of the Pareto-optimal multimodal resource-constrained project scheduling problem with quality parameters
22. H-PO-DMRCPSP-q: Hybrid, continuous version of the Pareto-optimal multimodal resource-constrained project scheduling problem with quality parameters

- typetfcn: Type of target function: 0=maxTPQ, 1=minTPT, 2=minTPC, 3=maxTPS, 4=minUF (level resources), otherwise composite
- w: Number of modes

Usage:

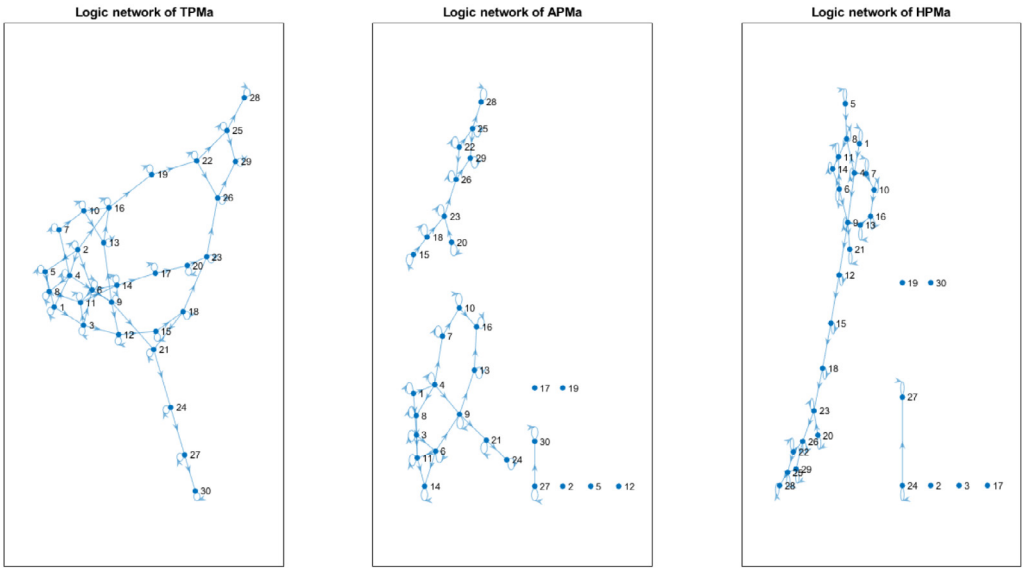
PSM=mhpma(PDM,CONS,Select,typefcn,w)

Example:

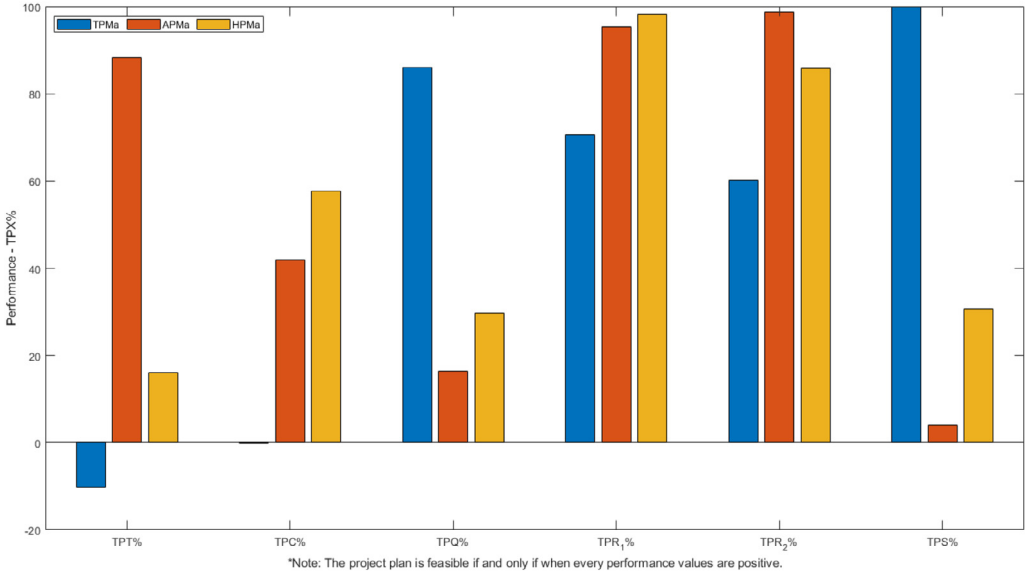
```
PDM=generatepdmq_scale(30,0.05,1,20,30,20,2,2,2,1.8) ;
w=2;
const=[percentt(PDM,w,.9), percentc(PDM,w,.9), …
percentq(PDM,w,.7), percentqr(PDM,w,1)', percents(PDM,.7)];
Select=15;
typefcn=1;
PSM=mhpma(PDM,const,Select,typefcn,w)
```

2.2.3. Risk analysis

The risk analysis simulates changes in the demands, structures, and constraints. These changes can update the entire PDM matrices and constraints. Any updates that come from either the



(a) Logic network for the PMas.



(b) Scheduling performances of the PMas.

Fig. B.6. Logic networks and scheduling performances of the project management agents (PMas).

control and revision of the task executions or generating changes are allowed. The paper considers only the generated changes to perform the risk analysis.

Three kinds of risk effects can be specified and analyzed. The *phase*1 and *phase*1*beta* functions specify the effects of uncertainties on task demands. *phase*1 specifies random modifications for the time/cost/resource demands and quality parameters, while *phase*1*beta* specifies that changes follow a beta distribution.

Output:

PDMout: a PDM with an identical structure as the input PDM

Inputs:

- PDM=[LD,TD,CD,QD,RD]: Project Domain Matrix
- a=the lower bound of demands
- b=the upper bound of demands

Usage: PDMout=phase1(PDM,a,b)

Example:

PDM=generatepdmq(30,0.05,0,20,30,20,2,2,2);

a=-.1; *%New values will be greater than or equal to 90% of the original values*

b=0.3; *%New values will be less than or equal to 130% of the original values*

PDM1=phase1(PDM,a,b)

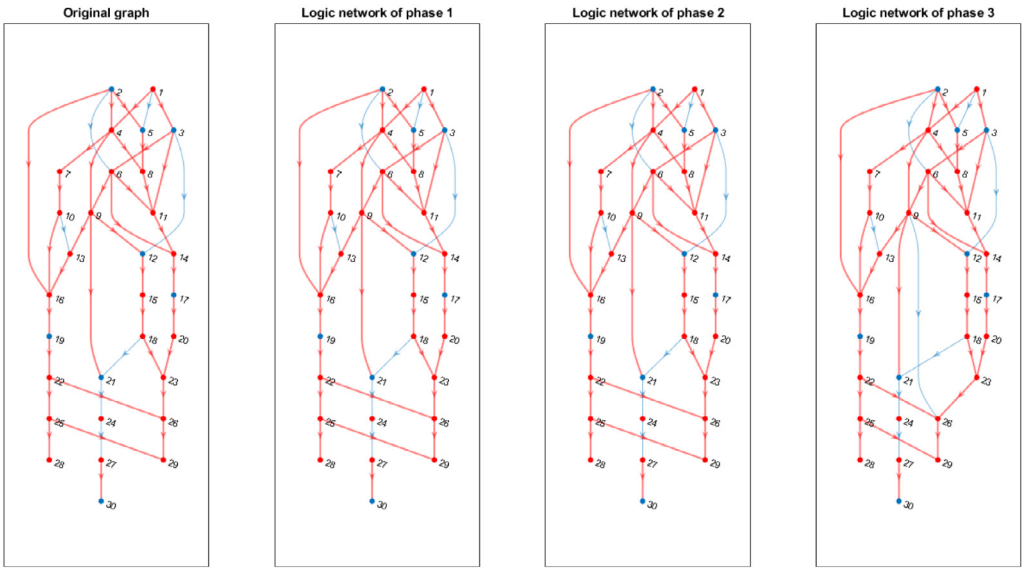
Function *phase*2 simulates the effects of extreme changes (shock effects) in several types of demand. Two Monte Carlo simulation steps are applied to select (1) and change (2) the task demands.

Example:

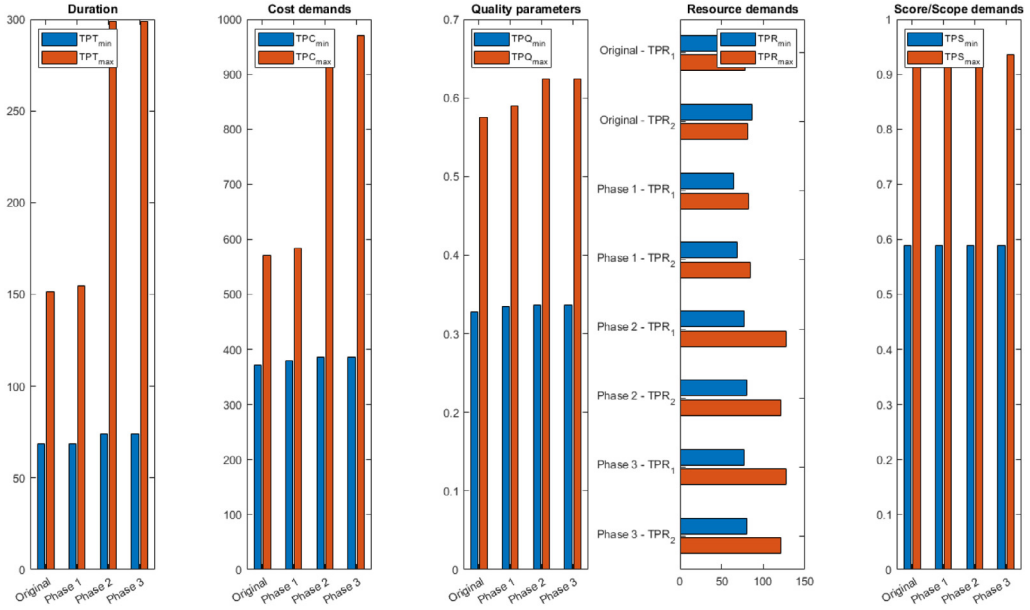
p=.05; *%Probability* factor for task selection

s=2.0; *%Scale factor: the ratio of the modification*

PDM2=phase2(PDM1,p,s)



(a) Changes in logic networks.



(b) Changes in the minimal/maximal demands.

Fig. B.7. Effects of changing project structures and project demands.

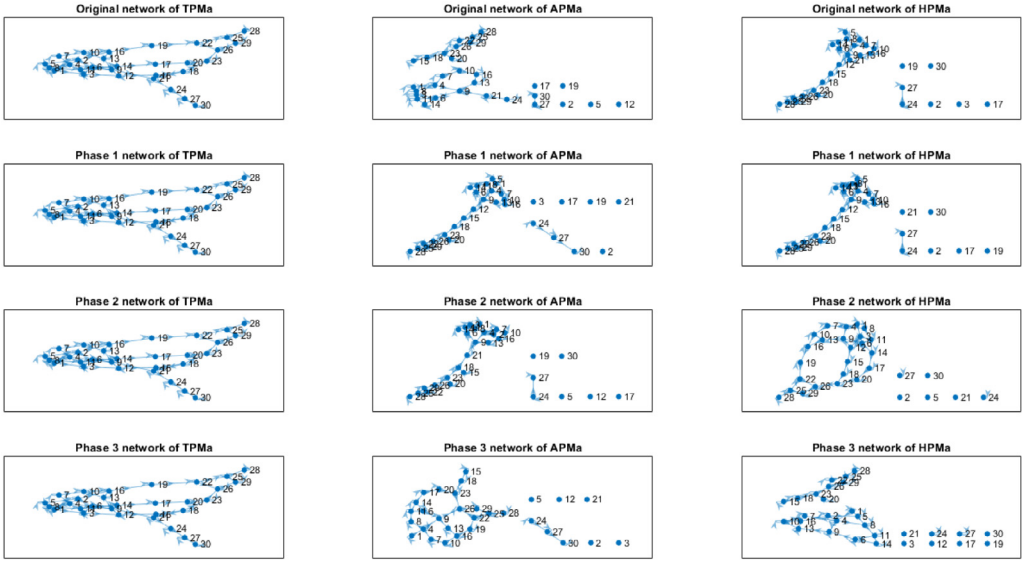
The last phase changes the structures, specifies unplanned tasks and changes the priorities.

Example:

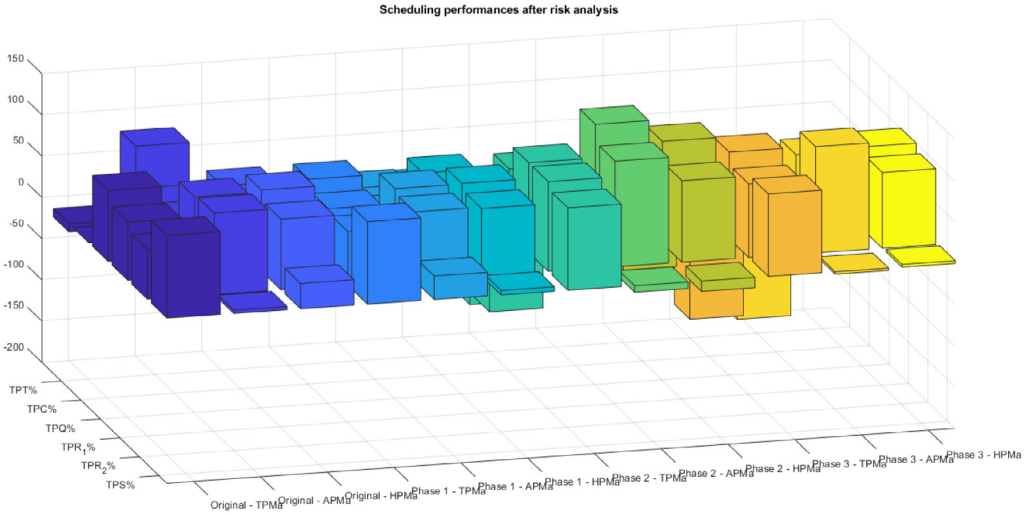
P=0.01 *%Probability of structural change*
S=0.50 *%Increase/decrease ratio for priorities*
PDM3=phase3(PDM2,P,S)

3. Impact

A substantial proportion of projects (more than a quarter of IT projects) are never completed [10]. However, many projects that can eventually be completed are often unsuccessful [10]. Many of them exceed the costs and time frames of the clients before the projects start, and this value has sadly hardly changed in the last 20 years [10]. We spend billions of dollars every year on failed projects that will never work. Research has shown that project failures are underpinned by the lack of applying project planning methodologies and management and organizational problems [23]. However, in our view, the role of project planning methods should not be underestimated [11]. In the past few decades, several new, flexible project management approaches have emerged [24], but their methodological elaborations remain incomplete. As these methods are increasingly used in IT, R&D projects and new product development projects, we increasingly see their limitations. In addition, these projects run in parallel and compete for resources with one another. Nevertheless, as long as no planning and scheduling methodology comparably examines the individual project planning procedures, management cannot choose which approach to use. In this software,



(a) Modified project structures.



(b) Risk mitigation performances.

Fig. B.8. Modifications to the project structures and scheduling performances. (*Colors have only a distinctive role.*)

we propose different kinds of agents (solvers) that simulate different kinds of project management approaches. They can be compared in terms of their scheduling and risk mitigation performance for both traditional and flexible projects. There are three groups of potential users of the proposed software application. (1) First, project managers can schedule both fixed and flexible project plans to manage flexible project plans. (2) Second, decision-makers, such as managers, multiproject managers, and project portfolio managers, can select the most adequate project management approach to minimize project demands and maximize quality according to customers' claims. (3) Third, scholars can compare project management approaches simulated by agents and find their advantages and shortcomings on a set of projects.

4. Conclusions

The proposed MFPP software creates a matrix-based model for both traditional and flexible project management approaches. The genetic algorithm approach allows us to solve several project scheduling problems, such as continuous and discrete versions of trade-off problems and multimodal resource-constrained project scheduling problems. In addition, the PMas can be compared according to their scheduling and risk mitigation performances.

One recently posed practical and theoretical question is whether flexible approaches extend to non-IT projects. Studies and practical experience on this topic remain controversial [11, 25,26]. The MFPP tool can help decision makers determine which approaches are best for their requirements.

In our future work, the standard project databases (see Refs. [27,28]) will be converted to matrix-based models to compare these approaches in several projects.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Project TKP2020-NKA-10 has been implemented with support provided by the National Research Development and Innovation Fund of Hungary, financed under the 2020-4.1.1-TKP2020 Thematic Excellence Program 2020-National Challenges sub-program funding scheme and by the Research Centre at the Faculty of Business and Economics, Hungary (No. PE-GTK-GSKK A095000000-1) of the University of Pannonia (Veszprém, Hungary).

Appendix A. An empirical example of the agent comparison

Table A.3 shows a comparison of project management agents where the target function, in this case, the minimal project duration, is common.

Table A.3 shows an example of a mini project plan with three tasks, two completion modes and two kinds of (renewable) resources. The diagonal of the LD indicates that there are two mandatory tasks, task **A** and task **B**, where the relative priority is 1.0, and there is one supplementary task (task **C**) that has a relative priority of 0.7. Therefore, task **C** can be postponed to another project, if necessary. Since the dependency score between task **A** and task **B** is less than 1.0, this dependency is flexible, which means, task **A** and task **B** can be completed either in parallel or in sequence. Another flexible dependency exists between task **B** and task **C**, where the dependency score is 0.6, which means if task **C** is selected for completion, task **B** and task **C** can be completed either in parallel or in sequence. Nevertheless, if task **C** is excluded or postponed, both its dependency and demands are also excluded from the project.

If there is more than one completion mode, both the traditional and hybrid project management agents can chose an appropriate technology. These technologies (i.e., completion modes) can have different durations, cost demands, quality parameters and resource demands. Different kinds of technology can be selected for different kinds of task completion. The TPMa cannot distinguish fixed and flexible dependencies or mandatory and supplementary tasks. If a task or dependency is described in the project, it has to be considered within the whole completion process. Nevertheless, TPMa can choose the technology with the shortest duration for each task (see Fig. A.2).

The result was also coded by a DMM matrix, which is called the project structure matrix (PSM). This matrix already contains a schedule domain of scheduled (in this case earliest) start time. The PSM does not contain flexible dependencies or supplementary tasks because agents decide which tasks and dependencies have to be included or excluded from the project. TPMa saves all supplementary tasks and all flexible dependencies that have a score greater than 0.5.

In contrast to the TPMa, APMa does not distinguish technologies (completion modes). Therefore, only a single completion mode must be specified for all task completion. If there is more than one completion mode, before running APMa, adequate completion modes have to be selected. By default, APMa chooses the first (see Fig. A.3).

Fig. A.3 shows that task **C** is postponed and that task **A** and task **B** are completed in parallel to minimize the project duration, but the first completion mode is always selected.

The HPMa can also select the completion mode and decide which tasks or dependencies should be included or excluded from the project (see Fig. A.4).

The result of HPMa is similar to the result of APMa; however, HPMa always chooses the completion mode with the shortest duration.

Appendix B. Illustrative examples

An illustrative example with a typical pipeline (demonstrated in Fig. 1) is provided. *DEMO.m* leads us through the generation of a PDM for scheduling and risk analysis.

In this example, the number of (planned) tasks is 30 ($N = 30$), the rate of flexible tasks and dependencies is 30% ($ff = 0.3$), and the number of parallel paths (parallel subprojects) is 3 ($cf = 3$). The maximum values of task, cost and resource demands are 20, 30, and 20, respectively ($mTD = 20, mCD = 30, mRD = 20$). The numbers of completion modes and resources are 2 ($w = 2, nR = 2$), and there are no unplanned tasks ($nW = 0$). The scale parameter specifies the rate of dependencies between tasks and is set to 1.6 ($scale = 1.6$) in this example.

Based on the generated project plan, Fig. B.5 shows a flexible project plan (Fig. B.5(a)) and the corresponding minimal/maximal task demands (Fig. B.5(b)). In Fig. B.5(a), the labeled graph represents the supplementary (labeled blue) and mandatory (labeled red) tasks and the flexible (labeled blue) and strict (labeled red) dependencies between tasks. Fig. B.5(b) shows the minimal and maximal demands of the projects.

For example, the minimal/maximal demands for the project duration (TPT) can be calculated as follows:

TPTmin=percentt(PDM,w,0); TPTmax=percentt(PDM,w,1);

Fig. B.6(a) shows the results of the logic networks of project management agents and scheduling performances (see Fig. B.6(b)). The scheduling performance of project demand is positive if the project plan is feasible. This ratio is one (100%) if the (scheduled) project (time/cost/resource) demands reach the minimal project demands, and the scheduled project (scope, quality) values reach the minimal project (scope/quality) values; however, this ratio is 0 (0%) if the scheduled (time/cost/resource) project demands and scheduled (scope/quality) values reach the constraint levels (see in detail in Ref. [6,11]).

Flexible approaches (i.e., the APMa, HPMa) allow for parallelized paths and postpone or ignore low-priority tasks (see Fig. B.6(a)). Therefore, the scheduling performance regarding project scope (TPS%) in flexible approaches may be lower than that in the case of the TPMa; however, these approaches can produce more feasible solutions and decrease the cost and time demands of the project. Thus, they can increase the scheduling performance of the approaches in terms of the time (TPT%) and cost (TPC%) demands (see Fig. B.6(b)).

In the case of risk analysis, the inherent uncertainty in the estimation of the demands (phase 1), shock effects (phase 2) and structural effects (phase 3) can be investigated (see Fig. B.7).

Fig. B.8 shows the modifications made to the project structures **a** and scheduling performances. Due to the risk effects (phase 1 - phase 3), the risk mitigation performances of the PMas can be compared in Fig. B.8(b).

The proposed software application has gone through the normal testing phases. For all MATLAB functions, a test environment is specified, and each function is tested via at least 10 different parameters, at least one of which is included as a comment before starting the program.

References

[1] Brucker P, Drexl A, Mohring R, Neumann K, Pesch E. Resource-constrained project scheduling: Notation, classification, models, and methods. Eur J Oper Res 1999;112(1):3–41. http://dx.doi.org/10.1016/S0377-2217(98)00204-5.

[2] Franco-Duran DM, d. l. Garza JM. Review of resource-constrained scheduling algorithms. J Constr Eng Manag 2019;145(11):03119006. http://dx.doi.org/10.1061/(ASCE)CO.1943-7862.0001698.

[3] Kosztyán ZT. Exact algorithm for matrix-based project planning problems. Expert Syst Appl 2015;42(9):4460–73. http://dx.doi.org/10.1016/j.eswa.2015.01.066.

[4] Servranckx T, Vanhoucke M. Strategies for project scheduling with alternative subgraphs under uncertainty: similar and dissimilar sets of355schedules. European J Oper Res 2019;279(1):38–53. http://dx.doi.org/10.1016/j.ejor.2019.05.023.
[5] Kosztyán ZT, Szalkai I. Hybrid time-quality-cost trade-off problems. Oper Res Perspect 2018;5:306–18. http://dx.doi.org/10.1016/j.orp.2018.09.003.
[6] Kosztyán ZT, Szalkai I. Multimode resource-constrained project scheduling in flexible projects. J Global Optim 2020;76(1):211–41. http://dx.doi.org/10.1007/s10898-019-00832-8.
[7] Rahimian V, Ramsin R. Second international conference on research challenges in information science. In: IEEE 2008. 2008, p. 337–42. http://dx.doi.org/10.1109/RCIS.2008.4632123.
[8] Tyagi MK, Munisamy S, Reddy LSS. Traditional and hybrid software project tracking technique formulation: state space approach with initial state uncertainty. CSI Trans ICT 2014;2(2):141–51. http://dx.doi.org/10.1007/s40012-014-0037-5.
[9] Kuhrmann M, Diebold P, Münch J, Tell P, Trektere K, Mc Caffery F, et al. Hybrid software development approaches in practice: a European perspective. IEEE Softw 2018;36(4):20–31. http://dx.doi.org/10.1109/MS.2018.110161245.
[10] SGI. Chaos manifesto. Standish Group International; 2020, https://www.standishgroup.com/news/45.
[11] Kosztyán ZT, Novák G, Jakab R, Hegedűs C. Survive IT! survival analysis of IT project planning approaches. Oper Res Perspect 2020;7:1–51. http://dx.doi.org/10.1016/j.orp.2020.100170.
[12] Browning TR. On the alignment of the purposes and views of process models in project management. J Oper Manage 2010;28(4):316–32. http://dx.doi.org/10.1016/j.jom.2009.11.007.
[13] Sami Ur Rehman M, Thaheem MJ, Nasir AR, Khan KIA. Project schedule risk management through building information modelling. Int J Comput Math 2020;1–11. http://dx.doi.org/10.1080/15623599.2020.1728606.
[14] Eppinger SD, Whitney DE, Smith RP, Gebala DA. A model-based method for organizing tasks in product development. Res Eng Des 1994;6(1):1–13. http://dx.doi.org/10.1007/BF01588087.
[15] V. Steward D. The design structure system: A method for managing the design of complex systems. IEEE Trans Eng Manage 1981;28(3):71–4. http://dx.doi.org/10.1109/TEM.1981.6448589.
[16] Chen CH, Ling SF, Chen W. Project scheduling for collaborative product development using DSM. Int J Proj Manag 2003;21(4):291–9. http://dx.doi.org/10.1016/S0263-7863(02)00023-6.
[17] Minogue P, Minogue P. Gantt-like dsms. InDSM 2011: Proceedings of the 13th international DSM conference. 2011, https://www.designsociety.org/publication/30838/.
[18] Browning TR. Managing complex project process models with a process architecture framework. Int J Proj Manag 2014;32(2):229–41. http://dx.doi.org/10.1016/j.ijproman.2013.05.008.
[19] Fang C, Marle F. A simulation-based risk network model for decision support in project risk management. Decis Support Syst 2012;52(3):635–44.
[20] Yassine A, Falkenburg D, Chelst K. Engineering design management: an information structure approach. Int J Prod Res 1999;37(13):2957–75. http://dx.doi.org/10.1080/002075499190374.
[21] Tang D, Zhu R, Tang J, Xu R, He R. Product design knowledge management based on design structure matrix. Adv Eng Inf 2010;24(2):159–66. http://dx.doi.org/10.1016/j.aei.2009.08.005.
[22] Danilovic M, Browning TR. Managing complex product development projects with design structure matrices and domain mapping matrices. Int J Proj Manag 2007;25(3):300–14. http://dx.doi.org/10.1016/j.ijproman.2006.11.003.
[23] Cooke-Davies T. The real success factors on projects. Int J Proj Manag 2002;20(3):185–90. http://dx.doi.org/10.1016/S0263-7863(01)00067-9.
[24] Wysocki RK. Effective project management: Traditional, agile, extreme, hybrid. John Wiley & Sons; 2019, http://dx.doi.org/10.1002/9781119562757.
[25] McAvoy J, Butler T. A failure to learn in a software development team: the unsuccessful introduction of an agile method. In: Information systems development. Springer; 2009, p. 1–13. http://dx.doi.org/10.1007/978-0-387-68772-8_1.
[26] Conforto EC, Salum F, Amaral DC, Da Silva SL, De Almeida LFM. Can agile project management be adopted by industries other than software development? Proj Manag J 2014;45(3):21–34. http://dx.doi.org/10.1002/pmj.21410.
[27] Kolisch R. Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation. European J Oper Res 1996;90(2):320–33. http://dx.doi.org/10.1016/0377-2217(95)00357-6.
[28] Vanhoucke M. Measuring the efficiency of project control using fictitious and empirical project data. Int J Proj Manag 2012;30(2):252–63. http://dx.doi.org/10.1016/j.ijproman.2011.05.006.