A survey in the resource-constrained project and multi-project scheduling problems

Samer Ben Issa[a] and Yiliu Tu[a*]

[a]Schulich School of Engineering, Department of Mechanical and Manufacturing Engineering, University of Calgary, Canada

CHRONICLE	ABSTRACT
	Resource-Constrained Project and Multi-Project Scheduling Problems (RCPSPs and RCMPSPs) have been essential topics of study over the last three decades. Both problems consist of activities that must be scheduled subject to precedence and resource constraints. This paper surveys studies of RCPSPs and RCMPSPs under consideration of four categories of project activities, simply recorded as categories A, B, C, and D. Category A refers to activities can be performed using fixed resources along The Y-axis over fixed durations along The X-axis, and cannot be interrupted. Category B applies to activities that can be performed using the same type of resource in category A but can be interrupted. Category C refers to activities that can be performed using flexible resources over flexible durations and cannot be interrupted. Category D refers to activities can be performed using flexible resources over flexible durations and can be interrupted. Many algorithms have been developed to solve the RCPSPs and RCMPSPs when activities are classified individually under category A, B, or C. However, in practice, welding, cutting or assembly activities in a manufacturing projects for an oil cargo can be under a new category so-called D. The project manager can speed up or slow down these activities by allocating or removing more resources, and these activities can be interrupted or can be resumed at any time. From the perspective of activity categories, we intend to review the literature on RCPSPs and RCMPSPs and to obtain the new research directions for solving the problems.

1. Introduction

In this section, Table 1 shows the entire notations used in the study.

Table 1
The entire notations used in this study.

A-O-A	Activity-On-Arc
A-O-N	Activity-On-Node
B&B	Branch and Bound
BCO	Bee Colony Optimization
BFI	Backward-Forward Improvement.
BPGA	Bi-Population Genetic Algorithm.

* Corresponding author.
E-mail address: paultu@ucalgary.ca (Y. Tu)

CA	Combinatorial Auction
Category A	The activity can be executed using constant duration and resource when interruptions are not allowed
Category B	The activity can be executed using constant duration and resource when interruptions are allowed.
Category C	The activity can be executed using flexible duration and resource when interruptions are not allowed.
Category D	The activity can be executed using flexible duration and resource when interruptions are allowed.
COA	Consolidated Optimization Algorithm
CPM	Critical Path Method.
DE	Differential Evaluation
EAs	Evolutionary Algorithms.
F-RCPSP	Flexible-Resource Constrained Project Scheduling Problem.
F-RCMPSP	Flexible-Resource Constrained Multi-Project Scheduling Problem.
GA	Genetic algorithm
GA-SA Hybrid	A hybrid heuristic based on genetic algorithm and simulated annealing
LFT	Latest Finish Time
LPF	Longest Path Following
MAXTWK	Maximum Total Work
MILP	Mixed-Integer Linear Program.
MIN-LST	Minimum Latest Start Time.
MINWCS	Minimum Worst-Case Slack
MMLIB	Multi-Mode Library.
MOA	Multi-Operator Algorithm
MODE	Multi-Operator Differential Evaluation
MOGA	Multi-Operator Genetic Algorithm
MPM	Multi-Project Model
MRCPSP	Multi-mode Resource-constrained Project Scheduling Problem.
MRCPSP/R	Multi-mode Resource-Constrained Project Scheduling Problem with Renewable Resources.
MRCPSP-RR	Multi-mode Resource-Constrained Project Scheduling Problem with Renewable Resource.
MTS	Most Total Successors
MWR	Most Work-content Remaining
MSA	Modified simulated annealing
MUF	Maximum Utilization Factor
NC	Network complexity
OKP	One-of-a-Kind Production
PERT	Program Evaluation and Reviewer Technique.
P-F-RCPSP	Preemptive-Flexible-Resource-Constrained Project Scheduling Problem.
P-F-RCMPSP	Preemptive-Flexible-Resource-Constrained Multi-Project Scheduling Problem.
P-MRCPSP	Preemptive-Multi-mode Resource-constrained Project Scheduling Problem.
PR	Priority rules
P-RCPSP	Preemptive-Resource Constrained Project Scheduling Problem.
P-RCMPSP	Preemptive-Resource Constrained Multi-Project Scheduling Problem.
ProGen/πx	Project Generator
PSP	Project Scheduling Problem
PSPLIB	Project Scheduling Problem Library.
PSPWC	Project Scheduling Problem Work-Content
P-MRCPSP-MC	Preemptive Multi-mode Resource-Constrained Project Scheduling Problem with permitted Mode Change
RCMPSP	Resource-Constrained Multi-Project Scheduling Problem.
RCMPSP(ABD)	Resource-Constrained Multi-Project Scheduling Problem when project activities under A, B, and D category.
RCPSP	Resource-Constrained Project Scheduling Problem.
RCPSP(ABD)	Resource-Constrained project scheduling problem when project activities under A, B, and D category.

RCPSP-FWP	Resource-Constrained Project Scheduling Problem with Flexible Work Profile
RDP	Resource Dedication Problem
RPP	Resource Portfolio Projects
RUCPSP	Resource-Un-Constrained Project Scheduling Problem.
SA	Simulated Annealing
SASP	Shorter Activity from Shorter Project
SFM-CWBM	Shortest Feasible Mode with Conditional Wait for the Better Mode
SFM-CWFM	Shortest Feasible Mode with Conditional Wait for the Fastest Mode
SSGS	Serial Schedule Generation Scheme.
TMS	Total Makespan
TPD	Total Project Delay
TS	Tabu Search.
TWK-LST	Maximum Total Work and earliest Late Start Time
URCPSP	Uncertain Resource-Constrained Project Scheduling Problem

A project is generally understood as a set of activities that have a precedence relationship or constraint and achieves specified objectives. These goals include constructing a building, developing a drug, building a ship, or fulfilling a customer order. The project's lifecycle has two crucial phases, the definition phase, which comes first, followed by the scheduling phase. Both are displayed in Fig 1. In the first phase, the project is executed with unlimited resources and a low level of uncertainties. Moreover, the project's objectives and specifications and organization must be articulated. Besides, the project network diagram and the estimated start and finish times for each activity are the definition phase's outputs, which in turn become the inputs of the scheduling phase.
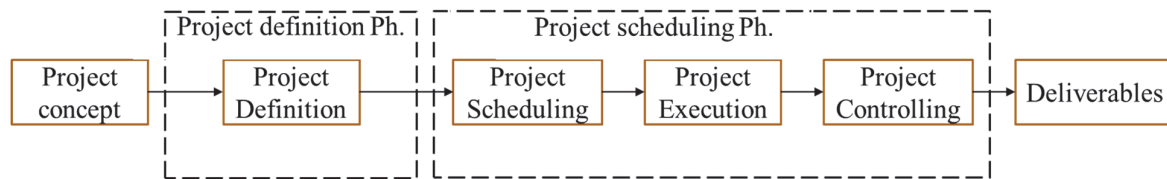

Fig. 1. The project definition and scheduling phase

In the second phase, in cases where the project is imposed under precedence relationships, the problem is called the Resource-Un-Constrained Project Scheduling Problem (RUCPSP). Here, a schedule is constructed to identify the project makespan. Various resources, such as workers, tools, and money, can be used to execute the project activities. Minimizing project maksepan is also a common objective during the scheduling phase. To represent the network activities and their relationships, in everyday use, there are two types of network diagrams: Activity-On-Node (A-O-N) and Activity-On-Arrow (A-O-A). Both of which are assumed to be acyclic. Project activities are labeled from A_0 to $A_{(n+1)}$. A_0 is the early activity without predecessor (source), and $A_{(n+1)}$ is the last activity without a successor (sink). In the A-O-N diagram, activities are represented by the nodes, whereas in the A-O-A diagram, the arcs represent activities (Kelly & Walker 1959; Malcolm et al. 1959).

The Gannt chart, which represents the project schedule, uses the Critical Path Method (CPM) and Program Evaluation and Review Technique (PERT), two well-known scheduling techniques. The durations of the project activities are assumed to be deterministic in CPM and probabilistic in PERT. However, in cases where the activities are executed using limited resources over the project time horizon, the problem is called the Resource-Constrained Project Scheduling Problem (RCPSP). Shortage and inefficient use of resources delay the execution of the activities even when all the preceding activities are completed. The K resources are labeled $k = 1,...,K$. The availability of R_k is assumed to be constant for the activity. Each activity, labeled i, requires r_{ik} units of resource K for each time unit. Where activities A_0 or $A_{(n+1)}$ do not exist, then dummy activities of zero duration and zero resource requirements are added. Five primary methods for modeling the resources in the projects are used (Gordon and Tulip 1997). First, aggregation determines the total number of resources required across the project makespan for the unit time (the most frequently used unit time

is the day). In the schedule, each column represents a project's unit time, and each network activity must be examined to ensure the activity uses resources. At the bottom of the schedule, the contents are represented in various ways as, for example, a histogram, bar chart, or table.

Second, cumulation – as it is so-called – provides a running cumulation of the resource required during the activity's execution, and the input to this process is the output from calculating the aggregation. The result from the input gives the project manager the total resource required to complete the project. Third, the allocation is used to provide a feasible schedule for completing the project under resource constraint. Seen as a problem, allocation employs the serial and parallel methods, and both have to be taken into consideration. At the same time, the two time-limited and resource-limited constraints have to be factored in. Fourth, the smoothing method generates a feasible schedule within the time constraints using a consistent level of resource requirements. And fifth, the leveling method removes the peaks and valleys from the feasible schedule. It works to improve the output from one of the four previous methods.

Scarce resources and precedence relationships between project activities make the process of scheduling problematic. To address this problem, since the 1990s, Icmile et al. (1993), Elmaghraby (1995), Ozdamar and Ulusoy (1995), Herroellen et al. (1998), Burker et al. (1999), and Kolisch and Hartmann (2006) have focused on developing algorithms to allocate resources to activities. Put another way, an unexpected delay in any activities leads to differing project schedules compared to the original schedule.

The following assumptions have shaped the project scheduling problem (PSP). First, when there are unlimited resources for executing project activities, and these activities have to be restricted only through precedence relationships. The problem, in this case, is so-called Resource-Un-Constrained Project or Multi-Project Scheduling Problem (RUCPSP or RUCMPSP). The RUCPSP and RUCMPSP assume activities under category A (i.e., activities are determined in advance and can be executed using fixed resources over fixed duration), as depicted below in Figure 2, and PERT or CPM techniques are used to resolve the problem. Two, conversely, when the precedence relationships and limited resources constraint the activities, then the problem is so-called Resource-Constrained Project or Multi-Project Scheduling Problems (RCPSPs or RCMPSPs). In the context of these assumptions, both the RCPSPs and RCMPSPs have been attacked by the existing literature. Such a schedule will be feasible when the interruptions are not allowed and when both the precedence and resource constraints are met. Alternatively, the schedule can be better when the project makespan is minimized. The RCPSPs and RCMPSPs are extended to different types when the non-preemption restriction is relaxed, in which case the project activities must be located under category B or C.

When project activities have to be carried out using fixed resources over fixed durations, then the activities should be classified under category "A" as depicted in Fig. 2.
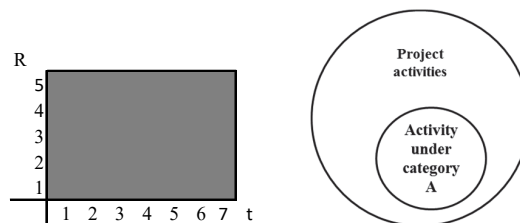

Fig. 2. An example of Activity classified under category A

When project activities have to be executed using fixed resources over fixed durations and can be interrupted, then the activities should be classified under category "B" as depicted in Fig. 3.
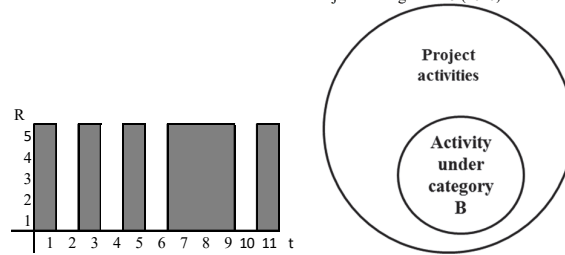
Fig. 3. An example of activity classified under category B

Whereas, when project activities have to be executed using flexible resources over flexible durations and cannot be interrupted, then the activities should be classified under category "C" as depicted in Fig. 4.
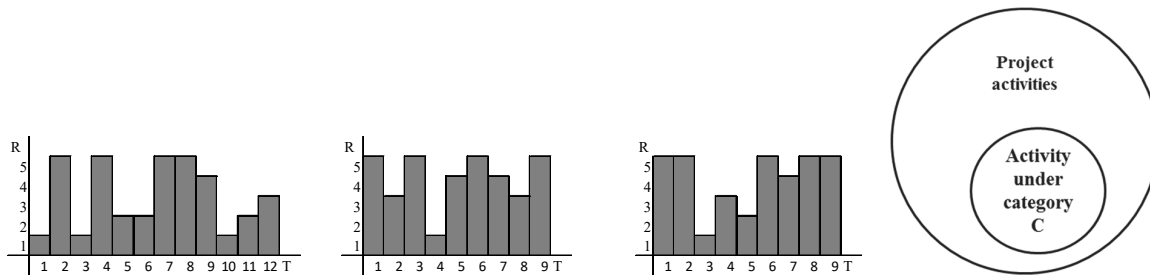

Fig. 4. An example of activity classified under category C

In most project scheduling software packages, the activity assumptions and any expected splitting are manually entered by the user before creating the schedules. From the methodology point of view, the current approaches, which include exact methods, heuristics, or meta-heuristics, are used to solve the RCPSPs and RCMPSPs, as illustrated in Fig. 5:

- Exact methods. These include dynamic programming, zero-one programming, and branch and bound (B&B). These algorithms only solve small problems and cannot find the optimal solutions in reasonable computation times. Thus, these algorithms do not work for large and complex projects. Many research papers have employed exact methods to deal with RCPSPs (Talbot 1982, Patterson et al. 1989, Hartmann and Sprecher 1996, Zhu et al. 2006, and Issa and Tu 2017).
- Heuristics. These include rule-based priority scheduling. The priority rules are faster than the exact method in finding solutions, and this makes them exceedingly practical, but the solutions found can be near-optimal. A significant number of heuristics have been used to find feasible solutions for the large-sized problems (Boctor 1993; Boctor 1996b; Kolisch and Drexl 1997; Knott et al. 2000). Boctor (1993), for example, tested 21 heuristic scheduling rules to get the best schedule.
- Meta-heuristics. These also include various algorithms to find the best feasible schedule for the RCPSPs. For example, Mori and Tseng (1997), Hartmann (2001), and Al-caraz et al. (2003) used Genetic Algorithms (GA), and Simulated Annealing (SA) was used by Slowinski et al. (1994), Boctor (1996a), and Bouleimen and Lecocq (2003). However, Nonobe and Ibaraki (2001) used Tabu Search (TS) to find a rational solution.
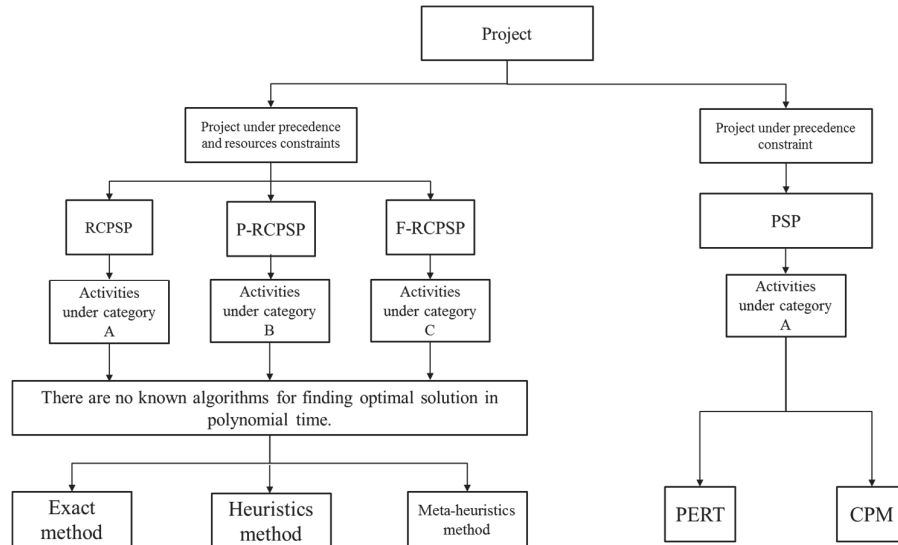
Fig. 5. The classifications of the project and multi-project scheduling problems

In theory, many algorithms have been used to solve the RCPSPs and RCMPSPs problems when project activities are under only one category A, B, or C. Accordingly, a new method or heuristic needs to be developed to solve the RCPSPs and RCMPSPs, one that considers project activities under four categories, the fourth specifically being category D, as depicted in Fig. 6. Activities under Category D need to be considered to give project managers more flexibility and efficiency in planning and scheduling a project. Categories A, B, and C are special cases of category D when the constraints imposed on them are relaxed. Therefore, the method or heuristic for solving the RCPSPs and RCMPSPs under category D can be applicable when project activities are under A, B, and C categories.
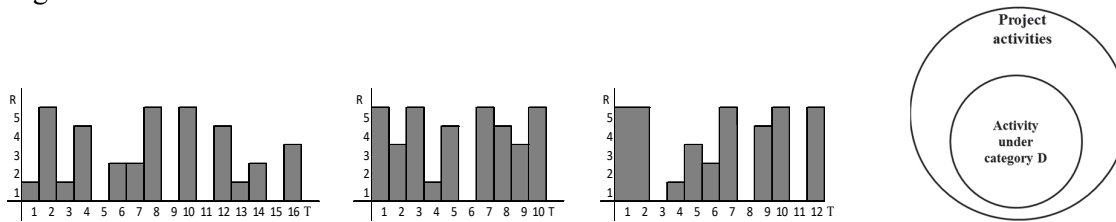

Fig. 6. An activity classified under category D

Table 2 below summarizes the essential activity assumptions implemented in different types of RCPSP and RCMPSP. In the RCPSP and RCMPSP, project activities are classified under category A. When the preemptive is added to them, a new extension problem is created, which is called the Preemptive Resource-Constrained Project and Multi-Project Scheduling Problems (P-RCPSPs and P-RCMPSPs). For P-RCPSPs and P-RCMPSPs, project activities are classified under category B.

In the Flexible Resource-Constrained Project and Multi-Project Scheduling Problems (F-RCPSP and F-RCMPSP), project activities are classified under category C. For the Flexible-Preemptive RCPSP and RCMPSPs (F-P-RCPSPs and F-P-RCMPSP) (i.e., project activities and resources can be preempted), project activities are classified under category D.

Table 2
The activity assumptions implemented in different types of RCPSPs and RCMPSPs.

General	Activity Categories	Fixed Duration	Fixed Resource	Flexible Resources	Interruption	State
RCPSP– RCMPSP	A	✓	✓	×	×	Used
P- (RCPSP– RCMPSP)	B	✓	✓	×	✓	Used
F- (RCPSP- RCMPSP)	C	×	×	✓	×	Used
F- P- (RCPSP- RCMPSP)	D	×	×	✓	✓	New

Equally important, activity assumptions in most project scheduling packages consider project activities under category A. However, categories B and C are not considered, but if users take them into account, they manually convert them to category A before creating a feasible resource schedule. The execution of activities, such as welding, cutting, and assembly, can be accelerated by increasing the resources allocated to them. Most, if not all, past research has classified these activities under categories B or C. Therefore, and worth highlighting, researchers have offered different algorithms as a way to find the best solution for each of the three types of classification, each taken individually. However, the problem is: researchers have classified the project activities individually under categories A, B, or C (e.g., Peteghem and Vanhoucke 2010; Kellenbrink and Helber 2015). However, in practice, in most engineering projects, if not all, some of these activities can be classified under category D rather than category B or C to give the manager more flexibility in scheduling and planning projects. The F-RCPSP can be enforced in the P-F-RCPSP as a particular case by allowing the activity to be preempted. Put another way, category C can be covered by category D (i.e., relaxing the non-preemptive constraint). As a result, from a preemptive point of view, we need to develop a new algorithm deals with the three types of activity assumptions A, B, and D simultaneously. We identify the extensions of RCPSP and RCMPSP as $(\text{RCPSP}_{(ABD)} - \text{RCMPSP}_{(ABD)})$.

The fundamental RCPSP and RCMPSP deal with scheduling activities in order to minimize the feasible completion time, subject to Finish-to-Start relations having a time-lag of zero, unless otherwise indicated. Here, resource constraints are considered unless otherwise indicated, and activity pre-emptions are not allowed. During the last few decades, resources have been ranked as an essential feature of any project, and the RCPSP has become a standard problem in project scheduling. The objectives and constraints, which are imposed on activities, have determined that the RCPSP is an NP-hard optimization problem. Blazewicz et al. (1983) have attracted many researchers; for overviews, Brucker et al. (1999), Kolisch and Hartmann (2006), and Ozdamar and Ulusoy (1995). Brucker et al. (1999) provide a notation to classify RCPSP to $\alpha / \beta / \gamma$ where α denotes resources characteristic, β illustrates project activities, and γ indicates the problem objectively. A number of researchers have developed a project scheduling problem using the standard RCPSP as a starting point, such as, Icmeli et al. (1993), Elmaghraby (1995), Ozdamar and Ulusoy (1995), Herroelen et al. (1998), Kolisch and Hartmann (1999), Brucker et al. (1999), Hartmann and Kolisch (2000), Kolish and Padman (2001), Brucker (2002), and Kolisch and Hartmann (2006). In this paper, we first investigated RCPSPs, RCMPSPs and their extensions. Then we presented a new perspective on these issues based on activity classifications A, B, C, and D.

2.   Literature review

The limited-resources and precedence relationships constraints, which are imposed on projects, are studied in the literature through two approaches: First, scheduling a single project, which is referred to as RCPSP or Multi-Mode Resource-Constrained Project Scheduling Problem (MRCPSP). Second, scheduling multiple projects, which is referred to as RCMPSP or as Multi-Mode Resource-Constrained Multi-Project Scheduling Problem (MRCMPSP).

2.1. Research on Resource-Constrained Single Project Scheduling Problem (RCPSP):

The term Resource-Constrained Project Scheduling Problem (RCPSP) has been used first in 1967 by Johanson. Many papers have been published for solving the RCPSP, which have been generated based on the problem type in real-life applications. After years of research on the RCPSP, extensions started to attract researchers' attention. For example, Hartmann and Briskorn (2010) address an overview of the RCPSP extensions according to the structure of the problem. Additionally, Weglarz et al. (2011) summarize the RCPSP extensions based on a unified framework of project scheduling model including resources, activity, objectives, and schedules. In this section of the survey, we investigate the most important research papers that used an A-O-N presentation network,

and then we classified the research papers based on activity categories used in their case studies. The RCPSP is an NP-hard, i.e., there are no known algorithms for finding the optimal solution in polynomial time. Peteghem and Vanhoucke (2010) introduce a Genetic Algorithm (GA) (i.e., meta-heuristic method) for solving Multi Mode-Resource Constrained Project Scheduling Problem (MRCPSP) and Preemptive Multi Mode-Resource Constrained Project Scheduling Problems (P-MRCPSPs). The MRCPSP is a generalized version of RCPSP, where each activity can be executed in one out of a set of modes, with a specific activity duration and resource requirements. The problem is subject to precedence relationships and limited renewable and non-renewable resource constraints. The objective of the MRCPSP is to minimize the project makespan by finding a mode and a start time for each activity. Peteghem and vanhoucke illustrate the preemptive extension of the MRCPS problems, which allows the project activities to be under category B (i.e., the assumption enabled project activities to be interrupted at any time instance and to be restarted at no additional cost). In their paper, they have extended the procedures for solving the RCPSP based on a bi-population approach, that was proposed by Debels and Vanhoucke (2005), to solve the MRCPSP and P-MRCPSP using the Bi-Population Genetic Algorithm (BPGA). Two different populations with the same size have been used for the BPGA; the first population contains right-justified schedules, and the second one contains left-justified schedules. Both populations have the same size. The iterative forward-backward introduced by Li and Willis (1992) is applied to build the left and right-justified schedules, which are stored in the left and right population, and the process is repeated until the stop criterium is met. The algorithm has been tested using the dataset in PSPLIB (Kolisch and Sprecher, 1996) and the dataset of Boctor (1993) to compare with other existing procedures from the literature.

Fundeling and Trautmann (2010) have considered a Project Scheduling Problem (PSP) in which the activities are characterized by work-content (PSPWC). That is, the resources allocated to activity usually may vary over time subject to some restrictions. The authors deal with the following project scheduling problem. First, the project resource is specified as the work content resource. Second, for each activity, the work-content in terms of resource-time units is pre-determined. Third, the work-content can be varied between an upper and a lower bound and must be an integer. Fourth, the relations among activities are constrained. Finally, all the resources are renewable and have a time-invariant capacity. Minimizing project makespan is the objective, subject to a decision for how much of the work-content is processed per period. This means that the project activities are classified under category C. They have presented priority rules (PR) based on a scheduling-generation scheme for large problem instances. These rules have shown a markedly better performance comparing with the exact approach, which requires a long computational time even for the small problem instances. The Longest Path Following (LPF), the Most Total Successors (MTS), and the Most Work-Content Remaining (MWR) are priority rules used when several activities are eligible for scheduling, and one needs to be selected. They develop a serial schedule-generation scheme; in each iteration, they determine a resource-usage profile for activities, which need to be scheduled by assigning a resource usage to each period. There are no test instances available which capture all features of the problem. Therefore, they generated new problem instances by modifying PSPLIB instances. The proposed priority-rules solve problems with up to 200 activities. Ranjbar and Kianfar (2010) propose a procedure for the Resource-Constrained Project Scheduling Problem with Flexible Resource Profile (RCPSP-FWP) to find all the feasible work content for each activity. The RCPSP-FWP is a different version of the RCPSP where activity duration and resource usage are known constants. They use a GA with a new crossover operator to schedule activities and minimize project makespan. The RCPSPFWP was proposed by Kolisch (2006) in the field of pharmaceutical R&D projects where, for the activities, the total work content – how much work has to be performed – is given. Put another way, the duration of and resource usage of the activities for each time-period of execution are unknown. The activities are restricted by the following five constraints: no preemptive is allowed (i.e., activities are classified under category C); each activity in processing has resource usage within a resource-dependent interval; work content for each activity equals the summation of resource units used for all periods of execution; resources used for each activity have to be constant

over a resource-dependent period; and each work profile for each activity is considered as an activity mode. Ranjbar and Kianfar validate their procedure by applying it to 30j and 60j non-dummy activities. They generate these activities drawing on a PSPLIB that uses one renewable resource.

Bianco and Caramia (2013) propose a new formulation for RCPSP subject to finish-to-start constraints, pre-emption is not allowed, resources are limited, and minimum makespan is the objective. Pritsker et al. (1969), Kaplan (1988), Alvarez-Valdes and Tamarti (1993), Mingozzi et al. (1997), and Klein (2000) have proposed Several mathematical formulations to solve the problem. In this paper, the new formulation is an attempt to generalize the mathematical model proposed by Klein (2000). It introduces a decision variable related to the amount of each activity processed at each time period. The formulation exploits three variables: the first and second variable, which are associated with the start and finish times of activity, and the third variable, which is associated with the amount of an activity in progress in a given time. They discretize the time horizon into T unit-width time period $[0,1)$, $[1,2)$, …, $[T-1, T)$ and consider the percentage of an activity executed until the time period t is x_{it}. A binary variable is $s_{it} = 1$ if an activity has started in a time period or $= 0$ otherwise. A binary variable is $f_{it} = 1$ if an activity has finished in a time period or $= 0$ otherwise. Bianco and Caramia tested the formulation on the RCPSP instances for the PSPLIB with 60, 90, and 120 activities with four resources. The formulation can produce better results in competing for other approaches. Project activities in this paper are considered under category A.

Colak et al. (2013) consider the Multi-Mode Resource-Constrained Project Scheduling Problem with Renewable Resources (MRCPSP-RR), where each activity can be executed in one of the possible modes. The Minimum Latest Start Time (Min-LST), the Shortest Feasible Mode with Conditional Wait for the Fastest Mode (SFM-CWFM), and the Shortest Feasible Mode with Conditional Wait for the Better Mode (SFM-CWBM) are heuristics, which not been used in MRCPSP-RR before for activity selection. The priority rules for executions are designed to take into account the criticality of activity when deciding on the execution modes. In their paper, they propose the use of Serial Schedule Generation Scheme (SSGS). To improve the solution quality, they employ the backward-forward improvement (BFI) that is known as a double justification in the literature, i.e., remove any unnecessary space in the Gantt chart. A new proposed approach called ACE-SP (Agarwal, Colak and Erenguc-single Pass) and an adaptive meta-heuristic have been proposed in conjunction with each of the two heuristics, Minimum Latest Start Time (LST) and Maximum Remaining Work (RWK), to improve the solution quality. Two benchmark problem sets are used in this paper: first one is created by Boctor (1993) and the second one is part of the PSPLIB (Kolish and Specher 1993). All the activities are considered under category A. Baumann and Trautmann (2013) formulate the Flexible Resource-Constrained Project Scheduling Problem (FRCPSP) as a Mixed-Integer Linear Program (MILP). In the classical project scheduling formulation, activities have been executed using fixed resources over a fixed duration. In practice, however, project managers were flexible; they changed the activity's resource usage during this time period (i.e., activities classified under **category C**), and this allowed for more efficient usage of the work content. Baumann and Trautmann take into consideration the activities subject to finish-to-start precedence relationships, the requirements as a prescribed of work-content, and variable amounts required for the work-content. They apply their proposed formulation to 480 problem instances, which Funeling and Trautmann (2010) introduced, also drawing on a PSPLIB with 10 activities. This proposed model solves 400 out of 480 problem instances.

Naber and Kolisch (2014) address how to minimize the project makespan in the FRCPSP by determining start time, resource usage, and duration of each preemptive activity (i.e., activities classified under category C). They propose four discrete-time model formulations and investigate the model efficiency in terms of problem size, solution quality, and runtime. Both the resource usages and the durations of the activities are unknowns, and they must be satisfied under the three following constraints:

The resource usage must be within specific lower and upper ranges; the total amount for each required resource must at least satisfy the resource available; a minimum number of consecutive periods must have a constant resource usage. Naber and Kolisch get the test instances from set A derived from the PSPLIB-RCPSP instances and from set B of 10, 20, and 40 activities, which are used as benchmark problems with no modifications.

Peteghem and Vanhoucke (2014) present an overview of the existing meta-heuristic for solving MRCPSP. The MRCPSP aims to find a mode and a start time for each activity to schedule the project within the minimal makespan, subject to precedence and resource constraints. The research paper considers only renewable resources, and the problem has been referred to as MRCPSP/R. Several exact methods, heuristics, and meta-heuristics solution procedures for the MRCPSP have been proposed in the literature. For example, Sprecher et al. (1997), Sprecher and Drexl (1998), and Zhu et al. (2006) presented the exact method; Talbot (1982), Drexl and Grunewald (1993), and Kolisch and Drexl (1997) addressed the heuristics method; Kolisch and Hartmann (1999), Weglarz et al. (2011) presented overview papers of the available exact, heuristic and meta-heuristic approach to the MRCPSP. However, heuristics cannot be used for solving large-size projects, since they are unable to find the optimal schedule in reasonable computation time. There are three objectives of the paper: The First objective is presenting an overview of the classification and the available meta-heuristics based on four classification criteria as follows: the meta-heuristics strategy, the schedule representation, the mode representation, and the schedule generation scheme. The second objective is proposing a new benchmark dataset to deal with the shortcoming of the PSPLIB and Boctor dataset. Peteghem and Vanhoucke suggest that the main advantages of the Boctor 100 dataset are: a large number of activities for each project instance, only renewable resources are taken into account, the renewable resources are almost not restricted, and the projects are mainly serial. The third objective is making a good comparison between the different current meta-heuristic solution procedures and evaluates the impact of the network and resource parameters of the project on the efficiency of the procedures. All the activities in this paper are under category A.

Cheng et al. (2015) illustrate the difference between the preemption and activity splitting in the RCPSP as following: a preempted activity is an activity eligible to be processed but is not being processed (i.e., by choices). Alternatively, a split activity is an activity that can be processed in uncontinuing time periods (i.e., due to insufficient resources), and it must be resumed at the next eligible processing time period until the activity is completed. Therefore, when an ongoing activity is paused because of resource unavailability and resumed later it leads to a small financial or time impact. Whereas interrupting an ongoing activity by switching to another activity leads to a high penalty such as setup time lost or reconfiguration complicated equipment setting. The branch-and-bound algorithm is employed to deal with three different problem settings under renewable and non-renewable resources constraints as follows: P1 represents the RCPSPs without activity splitting, P2 represents the RCPSPs with non-preemptive activity splitting, and P3 represents the P-RCPSP. The main modification of the algorithm is the use of the priority rule-based simple heuristics to find a better initial solution. The proposing of priority rule-based heuristics shows a significant advantage in computational time. The problem has been considered as an extension of RCPSP in which they examined a general case of calendarization by allowing time-varying resource-constrained and multiple processing modes for each activity. The benchmark problem instances have been generated by modifying the PSPLIB because it does not consider time-varying resource profile, resource vacation, or activity ready times and due dates. The problem instances have 12 or 16 activities, tow renewable resources, tow non-renewable resources, and three alternative processing modes. Many tested instances cannot find the optimal solution within the one-hour CPU limit time. In this paper, project activities considered under Category B.

Ma et al. (2016) address an Uncertain Resource-Constrained Project Scheduling Problem (UR-CPSP). The start and finish times and resource usage in most literature about the RCPSP are given

in advance for each activity. In the URCPSP, the historical data for project activities are not available, the activity durations are estimated by experts, and the objective is to minimize the project makespan. The execution of a project in an uncertain context can be considered as a dynamic decision process. The expected finish time, based on the expected value of activity durations, is more than 4% larger in projects containing 20 activities (Stork, 2000), and the value, sometimes, is more than 10% when the number of activities in projects arises to 120 (Ballestin, 2007). Thus, an uncertainty-theory-based project scheduling model has been proposed to deal with the estimations. The actual activity duration can be obtained after finishing its entire processes. The Genetic Algorithm has been developed to deal with the case. Each activity in the model is assumed to be executed by one mode, and preemption is not allowed. The problem instances are all derived from the PSPLIB (Kolisch and Sprecher, 1997). 30J, 60J, 90J, and 120J are problem instances, which have been considered in this paper. The uncertain activity durations are generated by $(d_i - \sqrt{d_i}, d_i, d_i + \sqrt{d_i})$. This implies the activities are classified under category A. Issa and Tu (2017) develop the Branch and Bound (B&B) heuristic to solve the RCPSP. They use the splitting activity as a way to cut down the project makespan. A new criterion called Moment of Resources Required (MORR) around the X-Y axes has been developed to measure and find the best plan among many alternatives of the resources distributed on the activities. In their model, the activities are classified under category B. The findings show that this approach gives project managers more flexibility to handle the scarce resources. Naber (2017) proposes a mixed-integer linear programming model for solving the RCPSP with flexible resource profile (FRCPSP) in continuous time (i.e., the activities are classified under category C). Under the resource allocation to be vary over an activity duration, neither the resource usage in each time period nor the duration of each activity is known and needs to be simultaneously determined with the schedule of each activity. The FRCPSP determines a schedule of activities and their resources under the following constraints: 1) Each activity, once started, must continue without interruption until its completion; 2) Finish-to-start relationships with zero time-lag between two activities; 3) The total amount of each resource allocated to each activity must at least satisfy its requirement; 4) Minimum block length (i.e., a fixed resource usage be allocated for a certain period; 5) minimum and maximum resource usage must be determined. The continuous-time and the discrete-time models for the RCPSP achieve the same project makespan, whereas it is not the case for the FRCPSP due to the dependency between the duration and the resource usage. According to Naber computational experience, the continues-time model is more challenging to solve than its discrete-time model. Due to extended run time, Naber performs only the small instances sets of 10 and 20 activities, which runs on the test set B of Fundeling and Trautmann (2010). The proposed continues-time model practically prescribes the project schedulers with shorter makespan than the discrete-time model. Elsayed et al. (2017) present a Consolidated Optimization algorithm (COA), which has more one optimization algorithm, each of which uses two multi-operator algorithms (MOAs) to solve the RCPSP. Heuristics perform better than exact methods for most RCPSPs. It is challenging to find an exact method or heuristic that performs well for a range of problem instances with varying complexities. However, meta-heuristics have demonstrated better performance than a heuristics-based approach, and they cannot guarantee an optimal solution. That is, Elsayed et al. proposed method is to use a high-level heuristic to select a heuristic in the low-level. The COA, which has more than one optimization algorithm to improve the quality solutions by rising the convergence rate, utilized the strength of two Evolutionary Algorithms (EAs) such as the Genetic Algorithm (GA) and the Differential Evaluation (DE). To speed up the rate of convergence, Elsayed et al. used a general population evolved by two Multi-Operator Algorithms (MOA) called Multi-Operator Genetic Algorithm (MOGA) and Multi-Operator Differential Evaluation algorithm (MODE), sequentially. The MOGA deals with the integer-based solution whereas the MODE deals with real-valued solutions. The probability of applying the MOA is based on their effectiveness in identifying the right solution and emphasizing the best- performing operator. A set of problem instances with J30, J60, and J120 activities, from the PSPLIB (Kolisch and Sprecher, 1996), have been used to evaluate the proposed approach. The best performances, in terms of quality of solutions, for the J30 and J60 problem instances have been obtained using the proposed model

compared with the state-of-the-art algorithms and was very competitive for J120. The activities in this paper are under category A.

Oztemel and Selam (2017) use a new meta-heuristic to select an effective single-mode for MRCPSP. The Bee Colony Optimization (BCO) approach has been used to complete the project on time. Project scheduling is a complex task having a significant effect on the project makespan, and it requires a significant amount of effort to prioritize project activities. In the literature, several papers have been published and addressed different algorithms and heuristics for planning and scheduling projects. For example, Li and Zhang (2013) and Xiao et al. (2013) used Ant Colony Optimization (ACO). Pacini et al. (2014) and Salem and Hassine (2015) applied Swarm Intelligence. Afshar et al. (2013), Barrios et al. (2011), and Peteghem and Vanhoucke (2010) demonstrated a Genetic Algorithm (GA). Buddhakulsomsiri and Kim (2007) created a priority rule-based heuristic. Some other studies employ computer programs such as Microsoft Project and Primavera. However, most of the studies employ algorithms, heuristics, and computer programs, which are not designed to concentrate enough on changing requirement, and they are not sensitive to sudden changes in resources that have to be utilized. Besides sensitivity, the software tools do not seem to facilitate the variation of solution procedures and cannot use alternative solutions in the different stage of project. They instead require a long time to solve the problem.

In plastic injection manufacturing, each molding process is assumed to be a project in a make-to-order environment. Plastic injection manufacturing requires a unique process for designing each mold: each mold needs a set of activities to be processed within a limited time-frame, and this specific time and the appropriates cost have to be satisfied. This is the reason for defining and solving mold manufacturing problems using the project scheduling methodology. The manufacturing systems of the mold industry can be considered as make-to-order projects where planning is critical because of uncertainty in product specification. In these dynamic and complex processes, it is significant to track each mold on its own for the customer. The Oztemel and Selam proposed BCO for scheduling plastic injection mold manufacturing by considering the following assumptions: each activity in the mold can be handled by a single resource, the activity duration must be deterministic, all resources are renewable, the activities are considered under category A, and the precedence relations among activities is finish-to-start with time lag = zero. The experimental projects originated from the plastic injection molding industry, and a group of projects with J10, J20, J30, J50, and J80 has been tested by the BCO. The proposed algorithm was able to generate suitable schedules and calculate the shortest completion time.

Afshar-Nadjafi (2018) extends the MRCPSP to the Preemptive Multi-mode Resource-Constrained Project Scheduling Problem with permitted Mode Change (P-MRCPSP-MC) after preemption. This model is not considered in previous studies. The standard MRCPSP involves selecting an execution mode for each activity and determining the start and finish times such that the precedence and resource constraints are met to minimize the project makespan. The Fixed work content is given for each of the project activities instead of a fixed duration and known resource requirements. Renewable and non-renewable resource types have been used in the problem. The accomplishing time of an activity can be interrupted at discrete time instances and restarted later with the same or different modes. The execution mode, which can be changed after being preempted, has not yet been studied. The objective of the P-MRCPSP-MC is to find a feasible schedule to minimize the project makespan. However, changeable execution modes and a preemption plan for each activity must be determined. The paper presents an efficient meta-heuristic solution procedure based on the Simulated Annealing (SA) approach, and binary value 0-1 on the decision variables are defined to specify whether an activity is in progress in period t or not. To increase the quality of proposed SA, an efficient dynamic heuristic is implemented to construct a schedule. An initial solution is created by setting project activities on the activity list, based on the Latest Finish Time (LFT). A set of 480 problems was generated by the project generator (ProGen/πx) developed by Drexl et al. (2000) to validate the proposed SA. The changeability leads to an average makespan improvement, and the

SA algorithm could efficiently solve the project scheduling problem. The activities are considered under category B. Tao et al. (2018) propose an extension of MRCPSP when the project network can be selected according to specific rules. The traditional MRCPSP, where it assumes each activity must be implemented with fixed precedence relations, is a particular case of the problem proposed in this paper, that is, the project does not have a fixed network diagram for its execution. In real-world applications, project structure is variant and how to choose a project structure is a significant decision for the scheduling problem. The AND-OR network, which was created by Tao and Dong (2017), is used to present the alternative project structures, that is, the activities are required to be selected based on the definition of AND/OR nodes. Besides the alternative project structure, three other important features are taken into account. First, various modes are available to accomplish the activity with different durations and resources, but only one mode must be chosen. Second, renewable and non-renewable resources are limited. Third, two objectives, project makespan, and total cost are considered. Kellenbrink and Hellber (2015) address the same problem, but they only focus on minimizing the project makespan subject to limited resources. The proposed problem can be decomposed into two sub-problems.The first subproblem is an activity selection problem. The second sub-problem is a bi-objective MRCPSP problem to identify which mode is selected for each activity and when to start to minimize the makespan and total cost. A tow-layer meta-heuristic has been developed with a combination of adapted Tabu search (TS) and NSGA-II to explore solutions efficiently. The TS algorithm is used to solve the first sub-problem, and the NSGA-II is used for the second sub-problem. A ten instances group of different problem scales is selected using the MRCPSP in PSPLIB. The computational experimental results demonstrate the efficiency of the proposed two-layer meta-heuristic. The project activities in this paper are under Category A.

Vanhoucke and Coelho (2018) present an overview of state-of-the-art algorithms for RCPSP and MRCPSP. The paper aims at demonstrating that most algorithms are still not able to solve instances much more significant in size than the ones presented between (1995-2017). Most algorithms, also, cannot solve problems with a different network and/or resource structure than usually used in the academic literature. Therefore, new algorithms will be needed to solve other problem instances, and new comparisons and a benchmark dataset will be necessary to simulate the development of such algorithms. It is worth mentioning that researchers are still spending their efforts on solving project instances of 30-activities, but they should drive their search to solve subsets from the PSPLIB and MMLIB that cannot be solved by state-of-the-art algorithms.

The paper does not present novel algorithms to solve unsolved problems but instead provides a tool to quickly analyze results for different sets of problems. The main goal of the paper is to provide a way to present best solutions obtained from the best performing procedure in the literature and to set up a system for uploading solutions for an alternative project data such as the PSPLIB and MMLIB uploading system. A new solution can be uploaded but not replaced on the existing system, whereas the new proposed system aims at helping researchers to upload and download solutions for the most used dataset in the literature. The restrictions of the paper were to review, define and discuss project data that can be used to solve the RCPSP and MRCPSP. The datasets contain network and resource data under strict assumptions as follows: 1) In a project network, a successor activity can immediately start once all the predecessor activities are finished, that is, they assume time lag = 0. Hence, there are no extensions to start-to-start, finish-to-finish, or start-to-finish, nor to the precedence of maximal time-lags being considered. 2) The resources are either restricted to renewable with limited availability for RCPSP and renewable and non-renewable for the MRCPSP. Hence, no other types of resources, (e.g., doubly-constrained resources), are taken into consideration. 3) The RCPSP has only one mode per activity whereas the MRCPSP has multiple modes. 4) No activity preemption is allowed. 5) Each activity is assumed to have a fixed duration mode and can therefore not be replaced by its corresponding work content. 6) All solutions in this paper have aimed at minimizing project makespan. 7) Project activities are considered under category A.

The main advantage of the new tool can be summarized as follows: 1) The tool offers easy access to solutions, a straightforward process of adding new data to be shared, and easy-to-maintain results. 2) The tool offers a simple selection of instances with specific characteristics; that is, it allows the researchers to focus on the most complicated problem instances. 3) By using the new dataset, researchers will focus on the areas for improvement to solve different project instances, rather than on testing algorithms on benchmark data, which aim to solve bigger project instances.

Most engineering projects cannot be efficiently scheduled the project by using the current algorithm because these algorithms assume that the activities are classified under categories A, B or C individually. To give project managers more flexibility to plan and schedule engineering projects efficiently, a need exists to develop a new algorithm in order to cope with all the activity assumptions (i.e., A, B, C, and D) and to create a resource-feasible schedule.

2.2. Research on Resource-Constrained Multi-project scheduling problems (RCMPSPs)

The other challenge in project management is to schedule multiple projects that are carried out simultaneously in a dynamic environment. Besides this, precedence relationships and limited resources available are two constraints which must be satisfied. Managing many projects using limited resources to achieve high production efficiency is more complicated than running a single project. In practice, many companies, particularly OKP companies, need to manage several projects simultaneously. For multi-projects, it is critical to share a common resource-pool during the implementation of activities and to optimize the allocation of these resources for achieving a minimum total makespan and hence, the best project scheduling. Furthermore, in OKP companies, due to high customization and production uncertainties, project rescheduling is frequently required. In this research, we will present two approaches to solve multiple projects scheduling problems. These include a single project approach and multi-project approach. The single project approach employs dummy activities and precedence arcs to combine multi-projects to a single mega-project and then use a current single project scheduling method to solve the problem. Alternatively, the multi-project approach maintains the separate critical path for each of the projects and develops methods to solve multi-critical paths project scheduling problems. Under these two approaches, a large number of papers have been published, and we review those milestone papers as in the following sub-sections.

Tsubakitani and Deckro (1990) develop a heuristics-based scheduling model by using actual firm data. The priority rules, Shorter Activity from Shorter Project (SASP) and Maximum Total Work (MAXTWK) (proposed by Kurtulus and Davis 1982) have been utilized to select the model's heuristic decision rules for the RCMPSP. The amount of free slack is calculated for each activity in the multi-project scheme, and an update feature is provided to allow project manager to re-schedule activities as more projects arrive, (i.e., the heuristic decision rule resolves the conflicts and allocates resources to activities). The following assumptions have been made in the development of the Multi-Project Model (MPM): the start and finish times and precedence relationships are deterministic, activity splitting is not allowed (i.e., activities are classified under **category A)**, and resources required to execute each activity are known and must be constant. When resource conflicts occur, following the priority rules, the scheduling model selects which activities start first. The scheduling model developed in this paper can improve the planning and control of multi-projects in a dynamic setting. The PMP was developed based on a project structure present in the housing industry. Thus, complete data for other housing industry firms were not available to provide a test on data. Two examples from the literature were tested to provide a comparison.

Lova et al. (2000) develop a multi-criteria heuristic approach which consists of several algorithms based on the improvement of feasible multi-project schedules. The problem is solved using heuristics based on priority rules. One of the most used methods to solve RCPSP is heuristics. This is because they are fast in terms of computational efforts, provide good results even for large size projects, and they are easy to integrate into project management tools (Kolisch, 1996). From work

in literature, 84% of the companies work with multiple projects, and heuristics based on priority rules with the RCMPSP perform better than the ones based on the single RCPSP. Project scheduling techniques need more flexibility for the practitioner. Put another way, companies frequently manage multi-tasks, and the project scheduling software packages need to be adapted accordingly. The multi-criteria algorithms had two phases. Phase 1 consists of an iterative forward-backward pass and uses to search and find a feasible project schedule. The forward-pass is applied to improve the idle resource. The backward-pass is applied to improve project efficiency, in-process inventory, and resource-leveling. From phase 1, a rational schedule for multi-project obtained. Phase 2 consists of several scheduling algorithms to further improve the schedule generated from phase 1. The multi-project cases that are used in this study are subjected to finish-start precedence constraints with time lag = 0. Splitting activity is not allowed (i.e., activities are under category A).

Moreover, each activity has a single mode, and the activities in each project are from 30 to 60 activities. The cases of the multi-projects have been solved with two priority rules, viz. Maximum Total Work (MAXTWK) and Minimum Latest Finish Time (MINLFT). In general, with the Project Scheduler 6 software, the developed algorithm obtains the most significant improvements.

Chen et al. (2009) propose a hybrid heuristic approach based on genetic algorithm and simulated annealing (GA-SA Hybrid) to schedule multiple projects with multiple resource constraints. The objective function is to minimize the largest finish time. They compared the model with the modified simulated annealing (MSA). The MSA is more potent than generic GA and SA. In general, using these methods in multi-project scheduling are rare. The GS-SA Hybrid model is applicable to most kinds of optimization problems which depend on the following five assumptions: resources are positive integers; preemptive activities are not allowed (i.e., all activities categorized under category A); precedence relations among activities should be identified; the priority of the activities should be determined; the activities of two different projects cannot be linked.

Four meta-heuristics, GA, SA, MSA, and GA-SA Hybrid, are given and compared using a unique system of coding for validation purposes. Moreover, each of the four heuristics is iterated with 10,000 iterations. The resource allocation when conflict occurs has to align with the priority of the activities. Three test projects and three real projects are tested as a way to validate the GA-SA Hybrid model. The results indicate that this proposed model generates the second-best feasible schedule for executing the three test projects and the best feasible schedule for executing the three real projects.

Browning and Yassine (2010) apply 20 priority rules (PR) to 12,320 test problems in project portfolios, where each portfolio consists of three projects. However, in practice, project managers have to deal with up to four projects simultaneously. The goal is to prioritize activities so as to optimize an objective function, and even a small improvement can be beneficial. Past research did not provide managers with clear guidance concerning which PR to use in various situations. The PRs are essential in practice because most project managers do not build formal activity network models, the prerequisites for applying the meta-heuristics. Although many project management textbooks discuss which PRs to use, they do not provide conclusive guidance for managers.

Most research papers on precedence relations and resource constraints deal with a single project. In practice, managers cannot choose appropriate PRs to use in the project to generate the best schedule. Browning and Yassine have demonstrated the superiority of the new measures and analyzed the performance of 20 PRs. Some of them which were developed for the RCMPSPs and others for RCPSPs. The goal of their research was to guide in using PRs for various project objectives and situations, such as cutting down the duration of projects when the activities cannot be preemptive. For project situations, four essential features of the RCMPSP have been identified: Objective function, network complexity, resource distribution, and resource contention. According to these four features, 12,320 project portfolios have been generated. Browning and Yassine have developed a decision table to guide project managers in choosing the best priority rules, and their analysis has

confirmed the superiority both of the Minimum Worst-Case Slack (MINWCS) and the Maximum Total Work and earliest Late Start Time (TWK-LST) when the activities are under category A.

Besikci et al. (2013) present a new heuristic solution called Combinatorial Auction (CA) for solving RCMPSP where the resources cannot be shared among projects, that is, where each project has its limited resources. In the literature, the approaches for modeling RCMPSP have allowed resource sharing between projects, but for these approaches resource dedication has not been considered. Besikci et al. describe the RCMPSP as the Resource Dedication Problem (RDP) and define it as the optimal dedication of resource capacities applied to different projects initially ready to start. Despite this, they did not consider that the projects involved finish-to-start with zero-time lag. Neither did they deal with the uncertainties nor the non-preemptive activities (i.e., activities under category A). Nevertheless, they took into account renewable and non-renewable resources having limited capacities. Past studies have depended on the underlying assumption that the entire renewable resource capacity is available for each project. However, this assumption does not work in cases where projects are distributed geographically, or where sharing resources is not preferred, or where the characteristics of the project do not allow resource sharing. In these cases, the RCMPSP, now entirely different, is difficult to manage. Two different problems have been addressed in this paper: scheduling activities with multi-mode under renewable and non-renewable resources; and the dedication of the resource capacities to one set of projects. The measure of performance minimizes the total weighted tardiness in overall projects. The new mathematical model consists of the Genetic Algorithm (GA), which is based on the preference concept, and the Lagrange relaxation-based heuristic. Six different projects from the J20 and J30 sets in PSPLIB are created to test this model. Two levels of Network complexity (NC) and three levels of maximum utilization factor (MUF) are selected, and a full factorial design with 10 problems in each combination is created. Satisfactory results for the RDP have been attained using the proposed model according to the solution time and quality.

Amol Singh (2014) presents a new hybrid algorithm to handle the RCMPSP that integrates the project criticality index with the activity priority. As a way to isolate the relevant problem, Singh engages the literature and selects three observations: First, most methods for scheduling activities are applicable in single project scheduling activities; second, scheduling resources for multiple projects is more complicated than for single projects; third, most of the literature has considered single resource for generating the schedule in multi-project. Accordingly, Singh argues that there is a need for more research on the scheduling of multiple projects and on developing a more efficient algorithm concerning computation time and quality solution. The author also shows there is a scope of research on multi-project scheduling that considers a variety of resources. In Singh's paper, five projects are placed in the Case-Study section, and four criteria are used to prioritize these projects: urgency, risk, growth, and net present value. These criteria control the project objectives. The activities are considered under category A. In addition, the benefits of traditional optimizing techniques cannot be utilized for generating the schedule of multiple projects. Therefore, researchers have developed multiple heuristics and meta-heuristics for RCMPSPs. However, there is still a need for developing a more efficient algorithm for computation time and quality solutions, one that generates multiple project schedules for complex problems. In this study, an efficient algorithm is further developed for generating schedules, and a variety of resources is considered for each activity during schedule development. The algorithm is demonstrated to be preferable to exist heuristics-based on priority rules, and its effectiveness is validated using the computational results.

Besikci et al. (2015) propose a meta-heuristic that consists of two approaches to solve the MRCMP-SPs, a two-phase GA and a so-called monolithic GA. Operative in this study is three assumptions: the due date of multiple projects must be identified; the sharing-resources between projects is not allowed; the total budget must be distributed across resources. Thus, the general resource capacities is a decision, needs to be determined, and is made according to a given total budget. This study investigates three procedures. First, determine the capacities of the resources. Second, establish the number of resources to be dedicated to each project. Third, identify the efficient solution of the

MRCMPSP results. Besikci et al. refer to the Resource Portfolio Projects (RPP) as determination of the general resource capacities subject to a total budget and the solution of the MRCMPSP subject to these capacities under RDP policy. The contribution of the study is a new mathematical model and a proposed solution approach for RPP in MRCMPS. In this study, five assumptions have been taken into account: The projects need to be ready to start, the uncertainties are not considered, the activity preemption is not allowed (i.e., activities are under category A), the activities have a set of different execution mode, and Renewable and non-renewable resources have limited capacities. They assume projects are not subject to precedence relations between themselves but coupled with the general resource capacity decision. The RPP solution approach is tested by combing six projects from J20 and J30 sets of PSPLIB, and two different performance measures characterize are considered: network complexity and maximum utilization factor. The problem sets have four resources, two being renewable and two non-renewable. Four different modes are generated with different durations and resources usages. The projects' due dates are calculated as makespan of the unconstrained case using CPM. The critical part of the issue is the determination of the general resource capacities according to the requirements of the projects from the general budget. The two-phase GA model gives best results comparing with the monolithic GA and exact solution approach.

Geiger (2017) proposes a new local search approach for solving MRCMPSP based on Variable Neighborhood Search (VNS) and Iterated Local Search (ILS) presented by Hansen and Mladenovic (2001) and Lourenco et al. (2003) respectively. In the MRCMPSP, a set of several projects, each of which is independent of the others and has to be scheduled simultaneously. The combination of the multi-project setting and the resource-constrained variants assume that at least one resource is commonly shared among the projects. The projects, each of which comprises a set of activities, need to be scheduled into a single overall plan. While precedence constraints among the activities of a single sub-project exist, there are no precedence relations between projects, activities are not allowed to be preemptive (i.e., activities categorized under category A), resources are assumed to be used or shared among the projects, and multi-execution modes for each activity are provided. In this paper, the quality of the scheduled obtained is evaluated by two objective functions: first, the Total Project Delay (TPD), and second, the Total Mmakespan (TMS). The two research goals are formulated as follow: Fast construction of the feasible first solution; fast convergence towards a high-quality solution; and manage and implement the communication between multiple threads in parallel. The solution approach has been tested on the novel datasets of the MISTA 2013 challenge. The MISTA 2013 brought three sets of instances: A, B, and X. The datasets combine several single projects taken from the J10, J20, and J30-instances. Moreover, additional experiments were conducted on a single project from the MMLIB, which are more intensely studied, and extensive computational results are available that allow a thorough comparison.

Table 3 represents a summary for each research paper mentioned previously and includes the names of the authors, the year of publication, the type of problem-whether RCPSP or RCMPSP, the method used, the dataset, and the activity assumptions. The research papers in Table 3 classified the project activities under category A, B, or C. Based on Vanhoucke (2012), an activity with work content = 12 man-day can be executed in 3 days using 4 units of resource, in 4 days using 3 units of resource, in 6 days using 2 units of resources, in 2 days using 6 units of resource, in 1 day using 12 units of resource, or in 12 days using 1 unit of resource and that can be occurred when the preemptions are not allowed. However, in practice, some project activities can be interrupted and can be used flexible resources for the execution (i.e., classified under category D). The execution can be in several possible ways. For example, the activity with work content = 12 man-day can be executed as follows: (1,12), (12,1), (3,4), (4,3), (2,6), (6,2) and that can occur when the preemptions are not allowed. Or can be executed as follows: [(2,3), (3,1), and (1,3)], [(5,1), (1,3), and (2,2)], or [(4,1), (2,2), (2,1), (1,1), and (1,1)] and that can be occurred when the preemptions are allowed.

Table 3

The summary of the research papers mentioned previously

	Author	Year	Type of the prob.	Method	Dataset	A	B	C	D
1	Peteghem and Vanhoucke	2010	MRCPSP and P-MRCPSP	Meta-heuristic BPGA	PSPLIB Boctor		√		
2	Fundeling and Trautmann	2010	PSPWC	Heuristic PR	Modified PSPLIB				√
3	Ranjbar and Kianfar	2010	RCPSP-FWP	GA	PSPLIB				√
4	Bianco and Caramia	2013	RCPSP	Exact method	PSPLIB	√			
5	Colak et al.	2013	MRCPSP-RR	Heuristic ACE-SP and meta-heuristic	PSPLIB Boctor	√			
6	Baumann and Trautmann	2013	FRCPSP	MILP	PSPLIB				√
7	Naber and Kolisch	2014	FRCPSP	MILP	PSPLIB				√
8	Peteghem and Vanhoucke	2014	An overview of MRCPSP	Existing Meta-heuristic	PSPLIB Boctor	√			
9	Cheng et al.	2015	(P1-P2-P3) RCPSP	Exact (B&B) meth. Heuristics-based PR	Modified PSPLIB		√		
10	Ma et al.	2016	URCPSP	Meta-heuristic GA	Modified PSPLIB	√			
11	Issa and Tu	2017	RCPSP	Exact-method B&B	Own		√		
12	Naber	2017	FRCPSP	MILP	Modified PSPLIB				√
13	Elsayed	2017	RCPSP	COA	PSPLIB	√			
14	Oztemel and Selam	2017	MRCPSP	Meta-heuristic BCO	Own	√			
15	Afshar-Nadjafi	2018	P-MRCPSP-MC	Meta-heuristic SA	ProGen/πx		√		
16	Tao et al.	2018	MRCPSP-APS	Meta-heuristic TS	PSPLIB	√			
17	Vanhoucke and Coelho	2018	RCPSP-MRCPSP	-	New Datasets	√			
18	Tsubakitani and Deckro	1990	RCMPSP	Heuristic-MPM	Own	√			
19	Lova et al.	2000	RCMPSP	Multi-criteria Heuristic	Own	√			
20	Chen et al.	2009	MRCMPSP	GA-SA Hybrid	Own	√			
21	Browning and Yassine	2010	MRCMPSP	Heuristic-PRs	Own	√			
22	Besikci et al.	2013	MRCMPS with RDP	Heuristic-CA	PSPLIB	√			
23	Amol Singh	2014	MRCMPSP	Hybrid-Alg.	Own	√			
24	Besikci et al.	2015	MRCMPSP	Meta-heuristic GA	Modified PSPLIB	√			
25	Geiger	2017	MRCMPSP	Heuristic VNS&ILS	MISTA 2013 and MMLIB	√			

Welding activities, painting activities, and assembly activities in OKP companies must be classified under category D rather than category B or C in the previous algorithms. An effort to tighten the gap between the project scheduling literature and the needs of the project manager should be investigated by the researchers. Much effort has been made to generate feasible schedules when the activities are classified individually under A, B, or C categories. However, the project manager, in his/her worksite, confronts activities that can be categorized under category D in addition to A and B simultaneously, that is, efficient algorithms to deal with these activity assumptions are required to generate more economically feasible schedules to solve the complex problems. Table 4 displays the needs of the project manager to achieve the best project schedules, which correspond to activity assumptions in reality.

Table 4

The needs of the project manager in achieving the best project schedules

	Author	Year	Type of prob.	M	Dataset	A, B, and D
1	*************	2019	RCPSP & RCMPSP	Heuristics	Modified PSPLIB	√

In summary, project scheduling requires significant amounts of effort to prioritize activities that will assure the best performance in terms of the completion time. Past studies have indicated that scheduling projects under limited resources are the main problem that researchers have encountered. Equally important, understanding the nature of the project activities (i.e., whether the activities can be executed using fixed or flexible resources over durations and can be preempted or not) is the more crucial point in order to obtain the best feasible schedule. The researchers must consider the ABCD activity classifications in their algorithms to improve resource utilization and project makespan, as depicted in figure 8. The activities in the RCPSPs and RCMPSPs often individually classify under category A, B, or C, but not simultaneously under category A, B, and D.
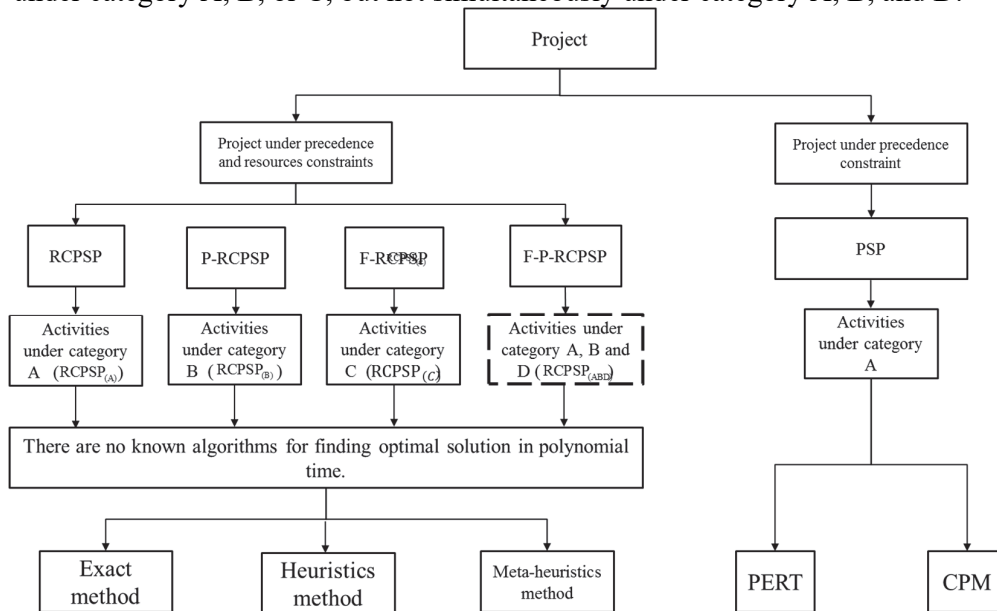

Fig. 8. Our vision for the new activity assumptions in the RCPSP and RCMPSP.

3. Conclusion

We have surveyed and summarized the achievements in the current literature for solving the RCPSP and RCMPSPs from an activity assumptions perspective. We have defined and classified project activities under four categories, A, B, C, and D. In our survey we have noted that the present approaches for solving the RCPSP and RCMPSP through classifying project activities under category D had not been considered previously. In practice, an engineering project has often included activities under more than one category. Our survey has revealed that a research gap in the RCPSP and RCMPSP persists and needs to be addressed. The methods or heuristics for solving the RCPSP and RCMPSP in OKP under activity categories A, B, and D used simultaneously need to be developed in order to more practically and efficiently plan and schedule engineering projects.

References

Alvarez-Valde´s R, Tamarit JM (1993) Project scheduling polyhedron: dimension, facets and lifting theorems. European Journal of Operational Research 67(2):204–220

Afshar-Nadjafi, B. (2018). A solution procedure for preemptive multi-mode project scheduling problem with mode changeability to resumption. *Applied computing and informatics*, *14*(2), 192-201.

Afshar-Nadjafi, B., Rahimi, A., & Karimi, H. (2013). A genetic algorithm for mode identity and the resource constrained project scheduling problem. *Scientia Iranica*, *20*(3), 824-831.

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.

Ballestín, F. (2007). When it is worthwhile to work with the stochastic RCPSP? *Journal of Scheduling*, *10*(3), 153-166.

Bianco, L., & Caramia, M. (2013). A new formulation for the project scheduling problem under limited resources. *Flexible Services and Manufacturing Journal*, *25*(1-2), 6-24.

Blazewicz, J., Lenstra, J. K., & Kan, A. R. (1983). Scheduling subject to resource constraints: classification and complexity. *Discrete applied mathematics*, *5*(1), 11-24.

Boctor, F. F. (1993). Heuristics for scheduling projects with resource restrictions and several resource-duration modes. *The international journal of production research*, *31*(11), 2547-2558.

Boctor, F. F. (1996). Resource-constrained project scheduling by simulated annealing. *International Journal of Production Research*, *34*(8), 2335-2351.

Brucker, P. (2002). Scheduling and constraint propagation. *Discrete Applied Mathematics*, *123*(1-3), 227-256.

Brucker, P., Drexl, A., Möhring, R., Neumann, K., & Pesch, E. (1999). Resource-constrained project scheduling: Notation, classification, models, and methods. *European journal of operational research*, *112*(1), 3-41.

Buddhakulsomsiri, J., & Kim, D. S. (2007). Priority rule-based heuristic for multi-mode resource-constrained project scheduling problems with resource vacations and activity splitting. *European Journal of Operational Research*, *178*(2), 374-390.

Boctor, F. F. (1996). A new and efficient heuristic for scheduling projects with resource restrictions and multiple execution modes. *European Journal of Operational Research*, *90*(2), 349-361.

Bouleimen, K. L. E. I. N., & Lecocq, H. O. U. S. N. I. (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.

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.

Browning, T. R., & Yassine, A. A. (2010). Resource-constrained multi-project scheduling: Priority rule performance revisited. *International Journal of Production Economics*, *126*(2), 212-228.

Beşikci, U., Bilge, Ü., & Ulusoy, G. (2013). Resource dedication problem in a multi-project environment. *Flexible Services and Manufacturing Journal*, *25*(1-2), 206-229.

Browning, T. R., & Yassine, A. A. (2010). Resource-constrained multi-project scheduling: Priority rule performance revisited. *International Journal of Production Economics*, *126*(2), 212-228.

Barrios, A., Ballestín, F., & Valls, V. (2011). A double genetic algorithm for the MRCPSP/max. *Computers & Operations Research*, *38*(1), 33-43.

Colak, S., Agarwal, A., & Erenguc, S. (2013). Multi-mode resource-constrained project-scheduling problem with renewable resources: new solution approaches. *Journal of Business & Economics Research (Online)*, *11*(11), 455.

Cheng, J., Fowler, J., Kempf, K., & Mason, S. (2015). Multi-mode resource-constrained project scheduling problems with non-preemptive activity splitting. *Computers & Operations Research*, *53*, 275-287.

Chen, P. H., & Shahandashti, S. M. (2009). Hybrid of genetic algorithm and simulated annealing for multiple project scheduling with multiple resource constraints. *Automation in Construction*, *18*(4), 434-443.

Debels, D., & Vanhoucke, M. (2005, October). The electromagnetism meta-heuristic applied to the resource-constrained project scheduling problem. In *International Conference on Artificial Evolution (Evolution Artificielle)* (pp. 259-270). Springer, Berlin, Heidelberg.

Drexl, A., & Gruenewald, J. (1993). Nonpreemptive multi-mode resource-constrained project scheduling. *IIE transactions*, *25*(5), 74-81.

Drexl, A., Nissen, R., Patterson, J. H., & Salewski, F. (2000). ProGen/πx–An instance generator for resource-constrained project scheduling problems with partially renewable resources and further extensions. *European Journal of Operational Research*, *125*(1), 59-72.

Elsayed, S., Sarker, R., Ray, T., & Coello, C. C. (2017). Consolidated optimization algorithm for resource-constrained project scheduling problems. *Information Sciences*, *418*, 346-362.

Elmaghraby, S. E. (1995). Activity nets: A guided tour through some recent developments. *European journal of operational research*, *82*(3), 383-408.

Fündeling, C. U., & Trautmann, N. (2010). A priority-rule method for project scheduling with work-content constraints. *European Journal of Operational Research*, *203*(3), 568-574.

Geiger, M. J. (2017). A multi-threaded local search algorithm and computer implementation for the multi-mode, resource-constrained multi-project scheduling problem. *European Journal of Operational Research*, *256*(3), 729-741.

Gordon, J., & Tulip, A. (1997). Resource scheduling. *International Journal of Project Management*, *15*(6), 359-370.

Herroelen, W., De Reyck, B., & Demeulemeester, E. (1998). Resource-constrained project scheduling: a survey of recent developments. *Computers & Operations Research*, *25*(4), 279-302.

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.

Hartmann, S. (2001). Project scheduling with multiple modes: a genetic algorithm. *Annals of Operations Research*, *102*(1-4), 111-135.

Hartmann, S., & Kolisch, R. (2000). Experimental evaluation of state-of-the-art heuristics for the resource-constrained project scheduling problem. *European Journal of Operational Research*, *127*(2), 394-407.

Hartmann, S., & Briskorn, D. (2010). A survey of variants and extensions of the resource-constrained project scheduling problem. *European Journal of operational research*, *207*(1), 1-14.

Hansen, P., & Mladenović, N. (2001). Variable neighborhood search: Principles and applications. *European journal of operational research*, *130*(3), 449-467.

Icmeli, O., Selcuk Erenguc, S., & Zappe, C. J. (1993). Project scheduling problems: a survey. *International Journal of Operations & Production Management*, *13*(11), 80-91.

Issa, S., & Tu, Y. (2017). Integrated multi-resource planning and scheduling in engineering project. *Journal of Project Management*, *2*(1), 11-26.

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

Kolisch, R., & Hartmann, S. (2006). Experimental investigation of heuristics for resource-constrained project scheduling: An update. *European journal of operational research*, *174*(1), 23-37.

Kolisch, R. (1996). Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation. *European Journal of Operational Research*, *90*(2), 320-333.

Kolisch, R., & Drexl, A. (1997). Local search for nonpreemptive multi-mode resource-constrained project scheduling. *IIE transactions*, *29*(11), 987-999.

Knotts, G., Dror, M., & Hartman, B. C. (2000). Agent-based project scheduling. *Iie Transactions*, *32*(5), 387-401.

Kolisch, R., & Hartmann, S. (1999). Heuristic algorithms for the resource-constrained project scheduling problem: Classification and computational analysis. In *Project scheduling* (pp. 147-178). Springer, Boston, MA.

Kolisch, R., & Padman, R. (2001). An integrated survey of deterministic project scheduling. *Omega*, *29*(3), 249-272.

Klein, R. (2000). Project scheduling with time-varying resource constraints. *International Journal of Production Research*, *38*(16), 3937-3952.

Kellenbrink, C., & Helber, S. (2015). Scheduling resource-constrained projects with a flexible project structure. *European Journal of Operational Research*, *246*(2), 379-391.

Kurtulus, I. B. R. A. H. I. M. S., & Davis, E. W. (1982). Multi-project scheduling: Categorization of heuristic rules performance. *Management Science*, *28*(2), 161-172.

Li, K. Y., & Willis, R. J. (1992). An iterative scheduling technique for resource-constrained project scheduling. *European Journal of Operational Research*, *56*(3), 370-379.

Lova, A., Maroto, C., & Tormos, P. (2000). A multicriteria heuristic method to improve resource allocation in multiproject scheduling. *European Journal of Operational Research*, *127*(2), 408-424.

Lourenço, H. R., Martin, O. C., & Stützle, T. (2003). Iterated local search. In *Handbook of metaheuristics* (pp. 320-353). Springer, Boston, MA.

Li, H., & Zhang, H. (2013). Ant colony optimization-based multi-mode scheduling under renewable and nonrenewable resource constraints. *Automation in construction*, *35*, 431-438.

Malcolm, D. G., Roseboom, J. H., Clark, C. E., & Fazar, W. (1959). Application of a technique for research and development program evaluation. *Operations research*, *7*(5), 646-669.

Mori, M., & Tseng, C. C. (1997). A genetic algorithm for multi-mode resource constrained project scheduling problem. *European Journal of Operational Research*, *100*(1), 134-141.

Mingozzi A, Maniezzo V, Ricciardelli S, Bianco L (1997) An exact algorithm for the resourceconstrained project scheduling problem based on a new mathematical formulation. Manag Sci 44(5):714–729

Ma, W., Che, Y., Huang, H., & Ke, H. (2016). Resource-constrained project scheduling problem with uncertain durations and renewable resources. *International journal of machine learning and cybernetics*, *7*(4), 613-621.

Nonobe, K., & Ibaraki, T. (2002). Formulation and tabu search algorithm for the resource constrained project scheduling problem. In *Essays and surveys in metaheuristics* (pp. 557-588). Springer, Boston, MA.

138

Oztemel, E., & Selam, A. A. (2017). Bees Algorithm for multi-mode, resource-constrained project scheduling in molding industry. *Computers & Industrial Engineering*, *112*, 187-196.

Özdamar, L., & Ulusoy, G. (1995). A survey on the resource-constrained project scheduling problem. *IIE transactions*, *27*(5), 574-586.

Pacini, E., Mateos, C., & Garino, C. G. (2014). Distributed job scheduling based on Swarm Intelligence: A survey. *Computers & Electrical Engineering*, *40*(1), 252-269.

Patterson, J. H., Słowiński, R., Talbot, F. B., & Węglarz, J. (1989). An algorithm for a general class of precedence and resource constrained scheduling problems. In *Advances in project scheduling* (pp. 3-28). Elsevier.

Pritsker, A. A. B., Waiters, L. J., & Wolfe, P. M. (1969). Multiproject scheduling with limited resources: A zero-one programming approach. *Management science*, *16*(1), 93-108.

Salem, H., & Hassine, A. B. (2015). Meeting scheduling based on swarm intelligence. *Procedia Computer Science*, *60*, 1081-1091.

Słowiński, R., Soniewicki, B., & Węglarz, J. (1994). DSS for multiobjective project scheduling. *European Journal of Operational Research*, *79*(2), 220-229.

Sprecher, A., & Drexl, A. (1998). Multi-mode resource-constrained project scheduling by a simple, general and powerful sequencing algorithm. *European Journal of Operational Research*, *107*(2), 431-450.

Sprecher, A., Hartmann, S., & Drexl, A. (1997). An exact algorithm for project scheduling with multiple modes. *Operations-Research-Spektrum*, *19*(3), 195-203.

Stork, F. R. E. D. E. R. I. K. (2000). Branch-and-bound algorithms for stochastic resource-constrained project scheduling. *Technical rep*, 702-2000.

Talbot, F. B. (1982). Resource-constrained project scheduling with time-resource tradeoffs: The nonpreemptive case. *Management science*, *28*(10), 1197-1210.

Tao, S., & Dong, Z. S. (2017). Scheduling resource-constrained project problem with alternative activity chains. *Computers & Industrial Engineering*, *114*, 288-296.

Tao, S., & Dong, Z. S. (2018). Multi-mode resource-constrained project scheduling problem with alternative project structures. *Computers & Industrial Engineering*, *125*, 333-347.

Tsubakitani, S., & Deckro, R. F. (1990). A heuristic for multi-project scheduling with limited resources in the housing industry. *European Journal of Operational Research*, *49*(1), 80-91.

Vanhoucke, M. (2012). *Project management with dynamic scheduling*. Springer Berlin Heidelberg.

Van Peteghem, V., & 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.

Van Peteghem, V., & Vanhoucke, M. (2014). An experimental investigation of metaheuristics for the multi-mode resource-constrained project scheduling problem on new dataset instances. *European Journal of Operational Research*, *235*(1), 62-72.

Vanhoucke, M., & Coelho, J. (2018). A tool to test and validate algorithms for the resource-constrained project scheduling problem. *Computers & Industrial Engineering*, *118*, 251-265.

Van Peteghem, V., & 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.

Xiao, J., Ao, X. T., & Tang, Y. (2013). Solving software project scheduling problems with ant colony optimization. *Computers & Operations Research*, *40*(1), 33-46.

Zhu, G., Bard, J. F., & Yu, G. (2006). A branch-and-cut procedure for the multimode resource-constrained project-scheduling problem. *INFORMS Journal on Computing*, *18*(3), 377-390.