IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 26 April 2023, accepted 9 June 2023, date of publication 27 June 2023, date of current version 30 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3289822

RESEARCH ARTICLE

Resource-Constrained Multi-Project Reactive Scheduling Problem With New Project Arrival

WULIANG PENG, DONGMIN YU, AND JIALI LIN
School of Economic and Management, Yantai University, Yantai 264005, China

Corresponding author: Jiali Lin (linjl1997@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 71671117 and in part by the Shandong provincial Natural Science Foundation under Grant ZR2022MG051.

ABSTRACT Reactive project scheduling is an important branch of project scheduling under uncertainty. This paper proposes a resource-constrained multi-project reactive scheduling problem (RCMPRSP) with new project arrival to minimize the adjustment cost of the baseline schedule while achieving the deterministic multi-project scheduling goal. Although the RCMPRSP aims to minimize the adjustment cost, it adds a constraint to ensure that the reactive schedule is one of the schedules with shortest possible make-span after new project arrival. Therefore, the make-span and cost can both be optimized during when the multi-project schedule must be reactive scheduled due to new project arrival. Therefore, this scheduling problem has two stages. In the first stage, the shortest make-span is obtained by fully rescheduling the multi-project after new project arrival. If the new project has a higher priority, the new project will be prioritized for implementation. Otherwise, the new project will be delayed. In the second stage, the multi-project is rescheduled to minimize the adjustment cost of the baseline schedule after new project arrival. The exact solution is implemented on IBM ILOG CPLEX Optimization Studio. It is found in the experiments that the CPLEX execution efficiency is acceptable for activity numbers below 80. The results of computational experiments demonstrate that the proposed method has distinct advantages over existing methods.

INDEX TERMS Project scheduling, reactive scheduling, resource-constrained multi-project scheduling, new project arrival, uncertainty.

I. INTRODUCTION

The resource-constrained multi-project scheduling problem (RCMPSP) combines multiple independent projects into a large project and all projects are scheduled uniformly based on common goals and same available resources of all the projects by the multi-project manager [1], [2], [3]. The traditional RCMPSP is a deterministic project scheduling problem that assumes that all project parameters are known during the planning phase and will not change during the execution of multi-project [4], [5]. However, this assumption is often in direct conflict with the reality that all the parameters that define a project may change during the implementation phase [6]. Consequently, many researchers in the past few decades have devoted themselves to establishing models or developing algorithms for project scheduling problems under uncertainty [4]. For review papers on project scheduling under uncertainty, we refer to Herroelen and Leus [5] and Hazır and Ulusoy [4].

Reactive project scheduling refers to modifying or re-optimizing the baseline schedule when it becomes no longer feasible or optimal after the unanticipated disturbance events occur during project execution [4], [5], [7], [8]. The disturbance event commonly occurs when activity durations, resource requirements, and availabilities vary during the implementation of projects. Another disturbance event is the new project arrival. For example, in mass customization production, the emergence of a new customer will lead to a new project, which should be inserted into the running multiple projects and disrupts the schedules of the running multiple projects. Other than the variation of activity duration or resources, the new project arrival will change the network structure of the multi-project, causing the baseline schedule formulated during the planning phase to

The associate editor coordinating the review of this manuscript and approving it for publication was Nagarajan Raghavan.

become no longer optimal [9]. In this case, reactive measures are necessary [7], [8]. In many situations, project managers require the reactive schedule to be optimal in the new project execution environment [10]. Given this prerequisite, they hope that the deviation between the reactive schedule and the baseline schedule is as small as possible because a large deviation from the baseline schedule may lead to significant adjustments in personnel, equipment, and materials [8]. These adjustments result in undesirable side-effects, such as having to change agreements with subcontractors, accumulating inventory costs, and coping with employee malcontent [8].

The reactive single-project scheduling problem has been well studied by researchers. However, there is little research on reactive multi-project scheduling, and to our knowledge, there are only two studies on reactive multi-project scheduling problems [11], [12]. In these two reactive multi-project papers, the scheduling situation is similar to that of a single-project when a disruption interruption event occurs. All of them use tardiness penalty and adjustment cost as scheduling objectives, construct and solve the problem model with precedence constraints and resource constraints, but do not consider the new project arrival. Therefore, the majority of research is directed at reactive single-project scheduling [4], [5], [8], [13], [14], [15]. However, the new project arrival is a special situation, in which many activities should be inserted into the multi-project schedule. To the best of our knowledge, literature on reactive multi-project scheduling that considers new project arrival has not been reported so far, although new project arrival occurs frequently during multi-project execution. Furthermore, the existing reactive multi-project scheduling methods only focus on the adjustment cost of the baseline schedule and cannot guarantee that the reactive schedule is still optimal in the new project execution environment [8], [11], [13], [16], [17].

New project arrival increases the number of activities in the multi-project, each of which has resource requirements. Since the network of the multi-project is changed, new project arrival causes more serious disruption than the variation of activity duration, resource requirement, or resource availability. The new project can be started before all the running projects in the multi-project are finished. However, the new multi-project schedule is no longer an optimal schedule since the make-span of the new multi-project is prolonged and the resources may be fully utilized. The original optimal baseline schedule typically becomes suboptimal in the new project execution environment when the new project started directly [9]. In many cases, project managers prioritize minimizing the sum of tardiness penalty of each project over minimizing the adjustment cost of the baseline schedule [4]. Consequently, this paper proposes a resource-constrained multi-project reactive scheduling problem (RCMPRSP) with new project arrival, where the adjustment cost of the baseline schedule is minimized on the premise of minimizing the sum of tardiness penalty of each project.

The structure of the paper is as follows. In Section II, we briefly review the literature on multi-project scheduling under uncertainty and reactive project scheduling. The RCMPRSP with new project arrival proposed in this paper is described in Section III. In Section IV, the mathematical models of the multi-project full rescheduling (FR) problem and the proposed multi-project reactive scheduling problem are established and illustrated. The solution to the proposed problem is described in Section V. In Section VI, the experimental scheme and experimental results and analysis are discussed. We provide overall conclusions in the final section.

II. LITERATURE REVIEW

A. MULTI-PROJECT SCHEDULING UNDER UNCERTAINTY

Reactive scheduling, proactive scheduling, stochastic scheduling, and fuzzy scheduling are four standard project scheduling methods under uncertainty [4], [5]. The definitions and characteristics of these four scheduling methods can be found in Herroelen and Leus [5] and Hazır and Ulusoy [4]. The current research on project scheduling under uncertainty focuses on single-project problems [4], [5], [8], [13], [14], [15], [17]. For reviews of single-project scheduling under uncertainty, we refer to Herroelen and Leus [5] and Hazır and Ulusoy [4].

However, the literature on multi-project scheduling under uncertainty is scarce [4]. To the best of our knowledge, there is only one study on purely reactive multi-project scheduling by Wang et al. [12], where a dual population genetic algorithm was proposed to solve the reactive multi-project scheduling problem with the goal of minimizing the deviation between the new scheduling scheme and the baseline schedule. Likewise, the literature on fuzzy multi-project scheduling and proactive multi-project scheduling is scarce. Hu et al. [18] proposed an outer-inner fuzzy cellular automata algorithm for dynamic uncertainty multi-project scheduling problem; Wang et al. [11] designed a genetic simulated annealing algorithm to address the proposed proactive-reactive multi-project scheduling problem; Afruzi et al. [19] solved the proposed proactive multi-project scheduling problem with a scenario-relaxation algorithm.

Compared with these three multi-project scheduling methods under uncertainty, more literature exists on stochastic multi-project scheduling. In the field of stochastic multi-project scheduling, Wang et al. [20], Chen et al. [9], and Liu and Xu [21] designed a strategy approximation method, a heuristic hybrid method, and a heuristic algorithm, respectively, with multiple priority rules to solve their proposed problems. Wang et al. [22] and Satic et al. [23] explored the performance of several priority rules and several scheduling policies, respectively. Both Wang et al. [22] and Satic et al. [23] established a Markov decision process for the stochastic RCMPSP. However, only Chen et al. [9] and Satic et al. [23] considered the new project arrival, a common disruption event during multi-project execution. Burdett and Kozan et al. [24] conducted a similar study in the field of railway scheduling, where they considered the competition for railway infrastructure between new services and existing services.

From the above existing research, in terms of coping with the uncertainty of activity duration and resource, there is little difference between solving multi-project problems and solving single-project problems for the above four project scheduling methods under uncertainty a major reason for the lack of research on uncertain multi-project scheduling [4], [5]. Unlike the uncertainty of activity duration and resource, new project arrival usually only occurs during multi-project execution and hardly occurs in single-project scheduling problems [9], [23]. Consequently, in recent years, the study of uncertain multi-project scheduling problems with new project arrival has become an important research topic. Although similar research has been reported in railway scheduling field [24], literature on reactive multi-project scheduling that considers new project arrival has not been reported so far.

B. REACTIVE PROJECT SCHEDULING

Reactive project scheduling refers to the process of adjusting the baseline schedule to minimize the breakage of the baseline schedule caused by unanticipated disruption events during project execution [4], [5], [7], [8]. With the scheduling objective as the classification criteria, reactive project scheduling methods can be divided into two categories [4], [5]. One is the full rescheduling (FR) method with a deterministic project scheduling goal as the objective [5], [25], [26], [27], and the other is the general reactive scheduling method to minimize the adjustment cost of the baseline schedule [4], [7], [8], [13], [15]. The latter is the most common in the reactive project scheduling literature [7], [8], [13], [14], [15], [17], [25], [26]. There are few studies on FR [10], [27] because the FR method closely resembles the deterministic project scheduling method [5], [27].

Generally, the goal of reactive project scheduling is to minimize the deviation between the reactive schedule and the baseline schedule [4], [5], [8], [13], [14], [15], [17], usually measured by the adjustment cost of the baseline schedule [8], [11], [13], [16], [17]. Most reactive project scheduling problems attempt to minimize the adjustment cost of the baseline schedule [8], [11], [13], [16], [17]. Many studies regard the cost of changing the activity starting time caused by reactive scheduling as an important part of the adjustment cost of the baseline schedule [15], [16], [28], [29], [30]. The cost of changing the activity starting time, is usually expressed by the total weighted deviation of the activity starting time between the reactive schedule and the baseline schedule, $\sum_{i\in N} w_i \cdot |S_i - s_i|$, where w_i is the cost incurred when decreasing or increasing the starting time of an activity $i \in N$ by one time unit, and s_i and S_i represent the baseline starting time and the reactive starting time of an activity $i \in N$, respectively [16], [29], [30]. Van de Vonder et al. [15], Deblaere et al. [28], Lambrechts et al. [29], Zheng et al. [30] and Ning et al. [16] modeled the reactive project scheduling problem to minimize the total weighted deviation of the activity starting time. Similarly, the reactive project scheduling problem proposed by Chakrabortty et al. [7], [13], [14] attempts to minimize the total weighted deviation of the activity finishing time. The objectives of Suwa and Morita [17], Deblaere et al. [8] and Kuster et al. [26] were minimizing the sum of activity delay time, minimizing the sum of mode switching cost and the cost of delaying the activity starting time, and minimizing a weighted sum of overall process tardiness, activity execution costs and the number of schedule modifications, respectively. All of these reactive project scheduling problems are single-objective problems. Except for these single-objective problems, there are very few studies which proposed dual-objective reactive project scheduling problems, such as Elloumi et al. [25] and Zaman et al. [31]. Both of these studies proposed a dual-objective reactive scheduling problem for MRCPSP. The first goal of these two studies is to minimize the project makespan. The second goal of Elloumi et al. [25] is to maximize schedule stability, and the second goal of Zaman et al. [31] is to minimize the repair cost for the disrupted resources.

In terms of constraints, most reactive project scheduling problems only consider precedence constraints and resource constraints [11], [15], [16], [25], [26], [29], [30], [31]. Based on these two constraints, several reactive project scheduling problems limit the reactive starting time of some activities [7], [8], [13], [14], [17]. There are two common restrictions on the activity starting time: one requires that no activity can be started before its baseline starting time [8], [17], and the other limits the activity starting time to the recovery time window [7], [13]. By connecting the objectives of the existing reactive project scheduling problems with their constraints, it can be found that the existing reactive project scheduling method ignores the optimal achievement of deterministic project scheduling goals when adjusting the baseline schedule. Consequently, it cannot guarantee that the reactive schedule is still optimal in the new project execution environment.

Reactive project scheduling problems can be classified under the resource-constrained project scheduling problem (RCPSP), which is NP-hard [14]. Developing algorithms for solving such problems is an important research direction in the field of reactive project scheduling [4]. For reactive single-project scheduling problems, many heuristic algorithms [8], [15], [25], [28], [30], [31], [32] and meta-heuristic algorithms [14], [16], [33], [34] have been developed. However, the literature on exact algorithms for the reactive single-project scheduling problem is scarce. To the best of our knowledge, only Deblaere et al. [8] proposed and evaluated a number of dedicated exact reactive scheduling procedures for repairing a disrupted schedule. For heuristic algorithms, Van de Vonder et al. [15] described new heuristic reactive project scheduling procedures; Deblaere et al. [28] proposed three integer programming-based heuristics and one constructive procedure for resource allocation; Deblaere et al. [8] proposed a tabu search heuristic; Elloumi et al. [25] proposed an Evolutionary Algorithm and a new reactive multi-objective heuristic; Zheng et al. [30] developed three heuristic algorithms, including TS, VNS, and VNTS; Zhao et al. [32] developed a heuristic algorithm based on priority rules; Zaman et al. [31] employed a new E-HA. For meta-heuristic algorithms, Ge and Xu [35] proposed a software project staffing model considering dynamic elements of staff productivity with a Genetic Algorithm(GA) and Hill Climbing (HC) based optimizer; Ning et al. [16] proposed two hybrid algorithms, Tabu-SA and VNTS; Zhang et al. [34] proposed a novel generic rescheduling strategy based on adaptive ant colony optimization algorithm; Chakrabortty et al. [14] developed an enhanced iterated greedy approach; Rahman et al. [33] developed an advanced metaheuristic based approaches called IGFBIS and IGFBID.

Although sufficient research has been conducted on the models and algorithms of reactive project scheduling problems, the majority of research is directed at reactive single-project scheduling [5], [6], [15], [36], [37]. The literature on reactive multi-project scheduling, however, is scarce. To the best of our knowledge, there are only two studies on reactive multi-project scheduling problems [11], [12]. Wang et al. [11] developed a genetic simulated annealing algorithm to solve the proposed reactive multi-project scheduling problem to minimize the total cost of delaying activity starting time. A reactive multi-project scheduling problem to minimize the sum of the duration deviation and cost deviation, along with a dual-population genetic algorithm, was proposed by Wang et al. [12]. Moreover, to the best of our knowledge, literature on developing exact algorithms for reactive multi-project scheduling problems has not been reported so far.

Many reactive project scheduling problems assume that the activity duration or resource availability follows a certain probability distribution, which are usually classified as stochastic project scheduling problems. In classic reactive project scheduling problems, the duration and resources are deterministic before or after the interruptions. The problem presented in this work is a variant of the classic reactive project scheduling problems. Comparison of the most representative studies on the classic reactive project scheduling problems and this work are listed in Table 1.

The existing reactive project scheduling problems only consider the adjustment cost of the baseline schedule, ignoring the optimal achievement of deterministic project scheduling goals. Consequently, they cannot achieve the reactive schedule minimizing the tardiness penalty of the multi-project after a new project arrives. Furthermore, the existing research on reactive project scheduling focuses on single-project problems, and the literature on reactive multi-project scheduling is scarce. The RCMPRSP proposed in this paper aims to cope with the arrival of new projects, a common disruption event during multi-project execution to minimize the adjustment cost of the baseline schedule and minimizing the sum of tardiness penalty of each project. Accordingly, the proposed RCMPRSP with new project arrival has obvious practical and theoretical research significance.

III. NOMENCLATURE AND PROBLEM DESCRIPTION

A. NOMENCLATURE

Sets

Q	Set of existing projects before new project arrival.
J_p	Set of all activities of project p.
$P_{p,j}$	Set of all immediate predecessors of activity p_j.
K	Set of all renewable resource types.
A_t	Set of all activities being executed at time instant t.
$\tilde{Q}$	Set of existing projects after new project arrival.
U^q	Set of all activities that have not started to execute at time instant q.

Parameters

p	Serial number of projects.
D_p	Completion deadline of project p.
M_p	Actual completion date of project p.
c_p	Tardiness penalty of project p per unit time.
p_j	The j^{th} activity in project p.
$d_{p,j}$	Duration of activity p_j.
$\omega_{p,j}$	Cost incurred when decreasing or increasing the starting time of activity p_j by one time unit.
k	Serial number of the renewable resource type.
R_k	Per-period availability of the renewable resource type k, $k \in K$.
$r_{p,j,k}$	Per-period requirement of activity p_j for the renewable resource type k, $k \in K$.
T	Upper bound of the multi-project makespan before new project arrival.
S^b	Baseline schedule.
$s^b_{p,j}$	Baseline starting time of activity $p_j, p \in Q, j \in J_p$.
q	Time instant of the multi-project reactive scheduling after new project arrival.
δ^q	Optimal value of the sum of tardiness penalty of each project (including newly arrived projects) which can be achieved in the new project execution environment at time instant q.
S^q	Reactive schedule determined at time instant q.
$s^q_{p,j}$	Reactive starting time of activity p_j, $p \in \tilde{Q}, j \in J_p$.
$f^q_{p,j}$	Reactive finishing time of activity p_j, $p \in \tilde{Q}, j \in J_p$.

TABLE 1. Comparisons of representative reactive project scheduling problems.

References	Multi-project	Objective function			Source of uncertainty		Solution method
		Earliness reward or Tardiness penalty	Adjustment cost	(Multi-)Project duration	Activity duration or Resource availability	New project arrival	
Chakrabortty et al. [13]			*		*		Exact algorithm
Chakrabortty et al. [14]		*			*		Meta heuristic
Chakrabortty et al. [7]			*	*	*		Exact algorithm
Deblaere et al. [8]			*		*		Exact algorithm
Elloumi et al. [25]			*	*	*		Meta heuristic
Elloumi et al. [10]				*			Heuristic algorithm
Kuster et al. [26]			*				Heuristic algorithm
Van de Vonder et al. [15]			*		*		Heuristic algorithm
Wang et al. [11]	*	*	*		*		Meta heuristic
Wang et al. [12]	*	*	*		*		Meta heuristic
Zhao et al. [32]			*				Meta heuristic
Zaman et al. [31]			*	*	*		Meta heuristic
This work	*	*	*			*	Exact algorithms

B. PROBLEM DESCRIPTION

RCMPSP involves several parallel projects and a shared resource pool containing several types of renewable resources with limited availability. There is no precedence relationship between projects and no precedence relationship between activities from different projects. Consequently, competition for limited shared resources is the only link between these projects [1], [2], [3]. Furthermore, it is assumed that all projects are scheduled uniformly in a centralized environment. In the study of classic reactive project scheduling problems, it is usually assumed that the durations of activities are deterministic before and after unexpected disturbances. This work follows this assumption that the completed activities will be executed according to the baseline schedule without deviation before the new project arrival. This assumption is reasonable for projects with high levels of certainty, such as manufacturing projects.

The set of existing projects during the planning phase is represented by Q. This set contains $|Q|$ independent projects, and all projects are numbered from 1 to $|Q|$. The set of all activities of project p is denoted as J_p, and all activities of project p are numbered from 1 to $|J_p|$. All activities have only one execution mode. The project tardiness penalty shall be paid when M_p is later than D_p. All activities are non-preemptive (i.e., they cannot be interrupted when in progress). Non-renewable resources are not considered in this paper; only renewable resources are considered.

The traditional RCMPSP assumes that all project parameters are known in advance and fixed. Its goal is to determine a baseline schedule S^b during the planning phase, satisfying the precedence relations and resource constraints while minimizing the sum of tardiness penalty of each project [38]. The baseline schedule $S^b = (s^b_{1,1}, s^b_{1,2}, \ldots, s^b_{1,|J_1|}; s^b_{2,1}, s^b_{2,2}, \ldots, s^b_{2,|J_2|}; \ldots; s^b_{|Q|,1}, s^b_{|Q|,2}, \ldots, s^b_{|Q|,|J_{|Q|}|})$, where $s^b_{p,j}, p \in Q, j \in J_p$ represents the baseline starting time of activity p_j.

q is the time instant of the multi-project reactive scheduling after new project arrival, namely the time instant of adjusting the baseline schedule, referred to as the adjustment time instant. The set of existing projects after new project arrival is represented by $\tilde{Q}$, containing $|\tilde{Q}|$ independent projects. The reactive schedule determined at time instant q is denoted as S^q. $S^q = (s^q_{1,1}, s^q_{1,2}, \ldots, s^q_{1,|J_1|}; s^q_{2,1}, s^q_{2,2}, -, s^q_{2,|J_2|}; \ldots; s^q_{|\tilde{Q}|,1}, s^q_{|\tilde{Q}|,2}, \ldots, s^q_{|\tilde{Q}|,|J_{|\tilde{Q}|}|})$, where $s^q_{p,j}, p \in \tilde{Q}, j \in J_p$, represents the reactive starting time of activity p_j. U^q is the set of all activities that have not started to execute at time instant q. The reactive starting time of activity $p_j \notin U^q$ should be the same as its baseline starting time, that is, $s^q_{p,j} = s^b_{p,j}$, because activity $p_j \notin U^q$ has started to execute at time instant q under the assumption that all activities are non-preemptive. However, the reactive starting time of activity $p_j \in U^q$ must be determined in real-time according to changes in the project execution environment because activity $p_j \in U^q$ has not started to execute at time instant q.

Under the assumption that other project parameters remain unchanged, the problem to be solved in this paper is how to adjust the baseline schedule to obtain a new scheduling scheme after new project arrival. The new scheduling scheme can guarantee that (1) the precedence constraints and resource constraints are satisfied, (2) the minimum of the sum of tardiness penalty of each project that can be achieved after new project arrival is achieved, and (3) based on (1) and (2), the deviation from the baseline schedule is minimized as much as possible. Of these, (2) has not received attention in the existing research on RCMPRSP.

IV. MATHEMATICAL MODELS

A. MODEL OF MULTI-PROJECT FR PROBLEM

When the baseline schedule cannot be implemented due to unanticipated events, the project manager may reschedule the activities that have not been started or completed according to the original goals of the baseline schedule, commonly called the tardiness penalty of projects. The goal of the multi-project FR problem in this paper is to formulate an FR schedule satisfying the precedence relations and resource constraints while minimizing the sum of tardiness penalty of each project (including newly arrived projects) after the unanticipated disruption events occur during multi-project execution. The mathematical model of the multi-project FR problem, denoted by M1 can be written as follows:

$$M1: Min\ TP = \sum\nolimits_{p=1}^{\left|\tilde{Q}\right|} c_p \cdot \max\{M_p - D_p, 0\} \quad (1)$$

$$s.t.\ M_p = \max_{j \in J_p}\{s_{p,j}^q + d_{p,j}\}, p \in \tilde{Q} \quad (2)$$

$$s_{p,j}^q = s_{p,j}^b, p_j \notin U^q \quad (3)$$

$$s_{p,j}^q \geq q, p_j \in U^q \quad (4)$$

$$s_{p,z}^q + d_{p,z} = f_{p,z}^q, p \in \tilde{Q}, z \in J_p \quad (5)$$

$$f_{p,z}^q \leq s_{p,j}^q, p_z \in P_{p,j}, p \in \tilde{Q}, j \in J_p \quad (6)$$

$$\sum\nolimits_{p_j \in A_t} r_{p,j,k} \leq R_k, k \in K, t = 1, 2, \ldots, T \quad (7)$$

$$s_{p,j}^q\ is\ non-negative\ integer, p \in \tilde{Q}, j \in J_p \quad (8)$$

The objective (Eq. (1)) is to minimize the sum of tardiness penalty of each project (including newly arrived projects),denoted by TP. Eq. (2) is the calculation formula of M_p indicating that the actual completion date of project p equals the maximum of the actual finishing time of activities of project p. Regardless of the status of the other projects, it is sufficient to find the maximum of the actual finishing time of activities of project p. Each activity in each project releases the assigned resources as soon as completed. In other words, completed projects release resources to subsequent projects, driving them forward and then generating a complete reactive schedule. Constraint (3) indicates that the starting time of activity $p_j \notin U^q$ should be the same as its baseline starting time because activity $p_j \notin U^q$ has started to execute at time instant q under the assumption that all activities are non-preemptive. Constraint (4) ensures that the starting time of activity $p_j \in U^q$ shall not be earlier than time instant q because activity $p_j \in U^q$ has not started to execute at time instant q. Constraint (5) represents duration constraints on the activity, where the activity start time plus the activity duration equals the activity end time, with no extra time for the activity to be interrupted from start to finish, ensuring that the activity is not preempted. Constraint (6) represents precedence constraints between the activities within each project—all predecessors must finish before an activity can start. Constraint (7) represents renewable resource constraints of the multi-project before or after the new project arrives—the usage of any type of renewable resource at any time instant cannot exceed its availability. Constraint (8) identifies the value range of the decision variable $s_{p,j}^q$.

The mathematical model of the traditional RCMPSP can be obtained by setting the value of q in the model represented by Eqs. (1)-(8) to 0 and replacing $\tilde{Q}$ with Q. The goal of the traditional RCMPSP is to determine a baseline schedule S^b during the planning phase that satisfies the precedence relations and resource constraints while minimizing the sum of tardiness penalty of each project.

Therefore, the model represented by Eqs. (1)-(8) has two purposes: (1) it can be used to determine a baseline schedule and (2) it can be used to formulate an FR schedule to obtain the optimal value of the deterministic multi-project scheduling goal (i.e., minimizing the sum of tardiness penalty of each project) achievable after new project arrival.

Obviously, the impact of the new project arrival on the original multi-project depends on the priority of the new project. If the new project has a higher priority, the model M1 will prioritize the new project to ensure the overall optimization of the new multi-project. Otherwise, the new project will be delayed. Of course, the priority of project activities is determined by the tardiness penalty per unit time, such as objective function 1.

B. MODEL OF PROPOSED MULTI-PROJECT REACTIVE SCHEDULING PROBLEM

When the baseline schedule cannot be executed due to unexpected events, a more common practice is to adjust the baseline schedule with the lowest cost, which is the most commonly used goal of reactive project scheduling. However, in most cases, the project manager needs to make a better schedule to minimize both the tardiness penalty of the multi-project and the adjustment cost of the baseline schedule in reactive scheduling.Similar to the existing RCMPRSPs, the RCMPRSP proposed in this paper also aims to minimize the adjustment cost of the baseline schedule. The cost of reactive scheduling is directly caused by the changes of personnel, equipment or materials when adjusting the baseline schedule. Each activity is different from others, and thus it is too complicated to consider the particular cost of each activity when adjusting activities. Therefore, the adjustment cost is simplified by a parameter $\omega_{p,j}$, which is the adjustment cost per unit time for activity j of project p. $\omega_{p,j}$ represents the all possible adjustment cost, including the changes of personnel, equipment and so on. The value of $\omega_{p,j}$ depends on the specific situation in actual production. The value of $\omega_{p,j}$ varies greatly in different environments and is reasonably taken by the project manager after judgment and analysis. For example, if the equipment used in the activity is more complex to start up, then the adjustment of the activity will be more costly, and in this case, a higher value of $\omega_{p,j}$ should be taken for the activity. Unlike the existing RCMPRSPs, the proposed RCMPRSP prioritizes achieving minimized tardiness penalty of multiple projects,and takes the minimized tardiness penalty as a constraint to further minimize the adjust cost of the baseline schedule. In other words, the proposed RCMPRSP minimizes the adjustment cost of the baseline schedule on the premise that the sum of tardiness penalty of each project (including newly arrived projects) has been minimized. The mathematical model of the proposed RCMPRSP, denoted by M2, can be written as follows:

$$M2: Min\ AC = \sum\nolimits_{p=1}^{|Q|} \sum\nolimits_{j=1}^{|J_p|} \omega_{p,j} \cdot \left| s_{p,j}^{q} - s_{p,j}^{b} \right| \quad (9)$$

$$s.t.\ \sum\nolimits_{p=1}^{|\tilde{Q}|} c_p \cdot \max\{M_p - D_p, 0\} = TP \quad (10)$$

$$Constraints\ (2) - (8)\ of\ M1$$

The objective (Eq. (9)) is to minimize the total weighted deviation of the starting time of activities between the reactive schedule and the baseline schedule (i.e., minimizing the adjustment cost of the baseline schedule), denoted by AC. Compared to the existing research, constraint (10) is a distinct improvement, which requires the tardiness penalty to be the lowest while minimizing the adjustment cost of the baseline schedule. Constraint (10) ensures that the reactive schedule S^q is one of optimal scheduling schemes in the new project execution environment by requiring that the sum of tardiness penalty of each project (including newly arrived projects) of the reactive schedule S^q must be equal to TP, the optimal value of the sum of tardiness penalty of each project after new project arrival.

Multiple iterations of reactive scheduling may be required during multi-project execution because new project arrival may occur multiple times. In this paper, each iteration of reactive scheduling is based on the result of the last reactive scheduling (except for the first time). Accordingly, the scheduling scheme formulated by the last reactive scheduling is regarded as the baseline schedule of each iteration of reactive scheduling (except for the first time).

C. BI-OBJECTIVE MULTI-PROJECT REACTIVE SCHEDULING PROBLEM

The model of the multi-project FR problem minimizes the tardiness penalty of the multi-project, and the model of the proposed multi-project reactive scheduling problem further optimizes the adjustment cost of the baseline schedule on the premise of the result of the multi-project FR problem. However, this multi-project reactive scheduling problem can only achieve a unique solution. Based on the above research, a bi-objective model can be constructed to balance tardiness penalty and adjustment cost comprehensively, so as to obtain the Pareto front of reactive scheduling and provide more powerful decision support for project managers.

Based on Section IV-A and Section IV-B, we can easily construct a bi-objective model. The two optimization objectives correspond to M1 and M2, respectively, and the constraints are the same as the constraints (2) - (8) of M1. The model, denoted by M3, can be formulated as:

$$\begin{aligned} M3 &: MinAC \\ &\quad MinTP \\ &\quad s.t. Constraints\ (2) - (8)\ of\ M1 \end{aligned}$$

V. SOLUTION APPROACH

A. SOLUTION OF THE SINGLE-OBJECTIVE VERSION OF THE PROPOSED PROBLEM

A solution to the proposed RCMPRSP is designed according to the characteristics of the problem. The main feature of this solution is to divide the problem-solving process into two stages. In the first stage, problem model 4.1 is called, indicating that the multi-project is fully rescheduled to obtain the optimal value of the deterministic multi-project scheduling goal (i.e., minimizing the sum of tardiness penalty of each project) achievable in the new multi-project execution environment. In the second stage, problem model 4.2 is called. Consequently, a new optimal scheduling scheme is formulated as the reactive schedule by taking the optimal achievement of the deterministic multi-project scheduling goal as the constraint and minimizing the adjustment cost of the baseline schedule as the objective. The solution procedure is as follows:

Step 1. During the planning phase, an optimal scheduling scheme (i.e., achieving the minimum sum of tardiness penalty of each project) satisfying precedence relations and resource constraints is given as the baseline schedule S^b.

Step 2. During the implementation phase, after the arrival of new projects, U^q can be obtained by judging which activities have begun to execute at time instant q. It is required that the starting time of activity $p_j \notin U^q$ should be the same as its baseline starting time and the starting time of activity $p_j \in U^q$ shall not be earlier than time instant q.

Step 3. Problem model 4.1 is called to fully reschedule the multi-project. In this step, an FR schedule is formulated that satisfies the precedence relations and resource constraints while minimizing the sum of tardiness penalty of each project (including newly arrived projects). The purpose of this step is to obtain the optimal value of the deterministic multi-project scheduling goal (i.e., minimizing the sum of tardiness penalty of each project) achievable in the new multi-project execution environment, according to the formulated FR schedule.

Step 4. Problem model 4.2 is called to schedule the multi-project with the reactive scheduling method proposed in this paper. By taking the optimal achievement of the deterministic multi-project scheduling goal as an important constraint and minimizing the adjustment cost of the baseline schedule as the goal, the baseline schedule S^b is adjusted to formulate a reactive schedule S^q. The purpose of this step is to determine a reactive schedule S^q, which is the one with the smallest deviation from the baseline schedule S^b among all the scheduling schemes that have achieved the deterministic multi-project scheduling goal after the arrival of new projects.

Step 5. If the arrival of new projects occurs again during multi-project execution, Steps 2, 3, and 4 are executed successively until the multi-project is completed. Each time the multi-project is reactively scheduled or fully rescheduled, the reactive schedule formulated during the last iteration should be used as the baseline schedule.

B. SOLUTION OF THE BI-OBJECTIVE PROBLEM BASED ON ε-CONSTRAINT METHOD

The ε-constraint method is one of the well-known methods for solving multi-objective problems, which can produce an accurate Pareto front. In this paper, the ε-constraint method is employed to solve the bi-objective problem presented in Section IV-C. The procedures are as follows:

Step 1. The adjustment cost minimization of the benchmark scheduling scheme is selected as the main objective function. The minimization of the delay penalty cost is set as a sub-objective function. Initiate a candidate set of Pareto solutions Cds $= \varnothing$.

Step 2. Solve M1 and achieve the value of the objective function TP as TP^{min}, estimate the largest possible value of TP as TP^{max}, and then set a step, denoted by g for increasing the value of ε according to the tardiness penalty per unit time.

Step 3. Add an extra constraint that requires the second objective TP equal to ε, minimize the unique objective AC under all the constraints. The achieved value of AC along with current ε are appended to the candidate set Cds, i.e., Cds: $=$ Cds$\cup\{(\varepsilon, AC)\}$.

Step 4. $\varepsilon := \varepsilon + g$. If $\varepsilon < TP^{max}$, return Step 3, else go to Step 5.

Step 5. All non-dominated solutions in Cds are identified to form the Pareto front of the bi-objective problem.

It is noted that for the two objectives of M3, the main goal and the secondary goal can be interchanged. In other words, as using the inverse sequence, the tardiness penalty cost becomes the main objective, and the adjustment cost becomes the secondary objective function. The computational result is the same as that of the original sequence.

VI. COMPUTATIONAL EXPERIMENT

A. EXPERIMENTAL SCHEME

The resource-constrained multi-project full rescheduling method (FR), the existing resource-constrained multi-project reactive scheduling method (OR), and the proposed resource-constrained multi-project reactive scheduling method (NR) are tested and compared in this section. The differences between the three methods are depicted in Table 2.

In this paper, all scheduling problems are programmed with IBM ILOG optimization programming language OPL and solved with CP Optimizer, the constraint programming optimization engine of CPLEX Optimization Studio V12.8.0. The run configuration is as follows: Intel(R) Core(TM) i5-9500 CPU @ 3.00GHz (6 CPUs), 3.0GHz processor; 8192MB RAM internal storage; Windows 10 operating system.

Browning and Yassine [39] comprehensively considered the various characteristic parameters of multiple projects and constructed a RCMPSP dataset containing 12,320 problems (http://sbuweb.tcu.edu/tbrowning/RCMPSPinstances.htm). Each problem contains 3 projects, each with 20 activities and 4 types of renewable resources. There are 20 file replications in this dataset, and each replication contains 8 Excel files, each with 77 problems on separate worksheets and a summary worksheet at the front. Each of the 8 Excel files differs in complexity settings (4 levels: "HHH", "HHL", "HLL", or "LLL") and MAUF variance (2 levels: 0 or 0.25). One hundred sixty instances are randomly selected from the 160 Excel files in the dataset and only one instance is chosen from one Excel file.

The setting of the disruption scenario is as follows: the newly arrived project 4 is inserted into the multi-project composed of projects 1, 2, and 3 at time instant q and the various characteristic parameters of project 4 are the same as the counterpart of project 1. Some project parameters involved in this paper are not included in this RCMPSP dataset. The settings of these project parameters are presented in Table 3.

Two evaluation indicators are selected to compare the solution quality of FR, NR, and OR. The first is the value of the deterministic multi-project scheduling objective, which is the sum of tardiness penalty of each project (including newly arrived projects) TP, TP $= \sum_{p=1}^{|\tilde{Q}|} c_p \cdot \max\{M_p - D_p, 0\}$. The second is the adjustment cost of the baseline schedule AC, the total weighted deviation of the starting time of activities between the reactive schedule and the baseline schedule, where AC $= \sum_{p=1}^{|Q|} \sum_{j=1}^{|J_p|} \omega_{p,j} \cdot \left|s_{p,j}^{q} - s_{p,j}^{b}\right|$.

B. EXPERIMENTAL RESULTS AND ANALYSIS

One hundred sixty randomly selected instances are divided into 20 groups according to the file replication to which each instance belongs. Accordingly, the groups are named Rep1, Rep2, …, Rep20. Each group contains 8 instances from 8 Excel files in the file replication to which these 8 instances belong. By testing 160 randomly selected instances, comparison results of the solution quality and computational time (C-T) of FR, NR, and OR can be obtained, as presented in Tables 4, 5, 6 and 7. Due to limited space, Table 4 only lists the average value (AVG) of the optimization performance evaluation indicators for each group of instances under the above three reactive scheduling methods. Tables 5, 6 and 7 present the test results of the 8 instances in Rep1, Rep11 and Rep20, respectively. Penalty in Tables 5, 6 and 7 represents the sum of tardiness penalty of each project of the baseline schedule S^b.

Tables 4, 5, 6, and 7 indicate that the computational time of NR is usually longer than FR. That is because FR only needs to find a reactive schedule achieving the deterministic multi-project scheduling goal after the arrival of new projects but NR aims to determine a reactive schedule with the smallest deviation from the baseline schedule among all the scheduling schemes achieving the deterministic multi-project scheduling goal after the arrival of new projects. In contrast, the computational time of OR is always the shortest among the three reactive schedules. There are two reasons. First, when other project parameters remain unchanged, the baseline schedule is still feasible after new project arrival while it becomes no longer optimal. Second, in contrast, the existing reactive scheduling method only seeks to minimize

TABLE 2. Differences between three resource-constrained multi-project reactive scheduling methods.

Scheduling method	Scheduling objective	Constraint conditions
Full rescheduling (FR) model M1	Same as deterministic multi-project scheduling objectives (e.g., minimizing the sum of tardiness penalty of each project)	Precedence constraints; Resource constraints; Constraints on the activity starting time
The existing reactive scheduling (OR)	Minimizing the adjustment cost of the baseline schedule (e.g., minimizing the total weighted deviation of the starting time of activities between the reactive schedule and the baseline schedule)	Same as the constraints of FR
The proposed reactive scheduling (NR) model M2	Same as the objective of OR	Precedence constraints; Resource constraints; Constraints on the activity starting time; A new constraint that ensures the optimal achievement of the deterministic multi-project scheduling objective
The ε-constraint method model M3	Same as deterministic multi-project scheduling objectives; Minimizing the adjustment cost of the baseline schedule	Precedence constraints; Resource constraints; Constraints on the activity starting time

TABLE 3. Settings of some project parameters.

Project parameter	Value
$D_p (p = 1, 2, 3)$	Critical path length of project p
q	Half of the multi-project makespan of S^b
$D_p (p = 4)$	q + the critical path length of project p
$\omega_{p,j}$ $(p \in Q, j \in J_p)$	1

TABLE 4. Comparison results of solution quality and computational time of each group of instances under different reactive scheduling methods.

Group	FR			OR			NR		
	TP (AVG)	AC (AVG)	C-T (AVG)/s	TP (AVG)	AC (AVG)	C-T (AVG)/s	TP (AVG)	AC (AVG)	C-T (AVG)/s
Rep1	3263	69	2.33	5950	0	0.15	3263	54	99.96
Rep2	4475	115	5.88	8275	0	0.13	4475	93	536.69
Rep3	2850	79	4.18	5738	0	0.14	2850	59	33.98
Rep4	3800	70	1.96	6700	0	0.13	3800	62	465.10
Rep5	2900	76	2.10	5888	0	0.16	2900	66	391.07
Rep6	2538	52	1.14	4675	0	0.18	2538	41	51.85
Rep7	4100	104	3.75	7363	0	0.17	4100	93	421.13
Rep8	5938	139	2.92	10213	0	0.16	5938	116	178.63
Rep9	4450	61	8.47	7063	0	0.13	4450	47	242.03
Rep10	1563	65	1.17	3938	0	0.15	1563	48	33.45
Rep11	1763	67	1.42	4300	0	0.20	1763	49	2.39
Rep12	6125	123	4.14	9488	0	0.18	6125	94	122.89
Rep13	4438	135	7.54	8325	0	0.14	4438	118	233.46
Rep14	5275	119	2.81	8575	0	0.12	5275	101	14.24
Rep15	8613	158	4.95	11250	0	0.13	8613	134	46.19
Rep16	1350	36	1.62	3575	0	0.13	1350	27	39.87
Rep17	1925	46	1.65	4475	0	0.15	1925	38	8.83
Rep18	3175	53	1.79	5200	0	0.13	3175	46	120.00
Rep19	4363	76	2.89	7150	0	0.13	4363	60	5.18
Rep20	2513	75	1.83	4875	0	0.14	2513	59	6.17

the adjustment cost of the baseline schedule under the feasibility constraints of the scheduling scheme. Therefore, when formulating a reactive schedule with the existing reactive scheduling method, it is only necessary to schedule the starting time for each activity of newly arrived projects based on the baseline schedule. There is no need to adjust the start time

TABLE 5. Comparison results of solution quality and computational time of 8 instances from Rep1 under different reactive scheduling methods.

Instance	Penalty	q	FR			OR			NR		
			TP	AC	C-T/s	TP	AC	C-T/s	TP	AC	C-T/s
01-1-65	2000	15	3800	137	2.49	6500	0	0.15	3800	76	3.51
01-2-41	1000	12	2700	15	2.68	3400	0	0.12	2700	4	0.19
01-3-71	2800	16	5400	65	5.41	7600	0	0.14	5400	48	741.58
01-4-40	500	12	1700	30	1.05	3800	0	0.20	1700	27	2.23
01-5-7	0	22	0	14	0.12	3300	0	0.14	0	12	0.17
01-6-50	3900	34	6400	182	1.59	12300	0	0.14	6400	179	33.09
01-7-27	2000	34	4000	71	4.17	5900	0	0.15	4000	55	17.79
01-8-38	900	17	2100	33	1.09	4800	0	0.12	2100	29	1.11

TABLE 6. Comparison results of solution quality and computational time of 8 instances from Rep11 under different reactive scheduling methods.

Instance	Penalty	q	FR			OR			NR		
			TP	AC	C-T/s	TP	AC	C-T/s	TP	AC	C-T/s
11-1-48	800	13	1300	32	0.57	2300	0	0.09	1300	19	0.49
11-2-52	1400	16	1900	104	0.12	4400	0	0.15	1900	84	0.86
11-3-21	500	16	1600	61	0.87	3800	0	0.19	1600	56	0.98
11-4-30	600	11	1400	19	0.81	2700	0	0.59	1400	14	0.67
11-5-22	200	16	600	47	0.36	4400	0	0.10	600	28	2.34
11-6-8	1700	28	3600	65	1.39	5900	0	0.19	3600	65	8.18
11-7-5	0	15	100	36	0.19	2400	0	0.12	100	29	0.82
11-8-13	1600	36	3600	170	7.03	8500	0	0.14	3600	99	4.77

TABLE 7. Comparison results of solution quality and computational time of 8 instances from Rep20 under different reactive scheduling methods.

Instance	Penalty	q	FR			OR			NR		
			TP	AC	C-T/s	TP	AC	C-T/s	TP	AC	C-T/s
20-1-3	900	19	1800	11	0.14	3000	0	0.14	1800	4	0.13
20-2-76	3300	19	4400	166	8.83	7200	0	0.14	4400	118	18.62
20-3-25	0	9	200	30	0.21	2100	0	0.14	200	20	3.18
20-4-57	2200	17	3000	96	0.40	5500	0	0.16	3000	69	10.36
20-5-23	0	9	400	24	0.14	1800	0	0.14	400	20	0.76
20-6-1	0	16	0	2	0.14	300	0	0.14	0	1	0.14
20-7-71	7400	44	9400	234	3.93	15800	0	0.10	9400	202	9.72
20-8-7	0	23	900	37	0.88	3300	0	0.13	900	35	6.42

of each activity of the original projects. The above reasons can also explain why the AC value of the OR schedule is always zero.

The results are not unique to this dataset, since: 1) FR minimizes the sum of tardiness penalty of each project (TP), without limiting the adjustment cost of the baseline scheduling (AC). If multiple computational experiments are performed for the same instance, TPs are the same minimum and ACs may be different values; 2) OR minimized the adjustment cost of the baseline scheduling (AC), without limiting the sum of tardiness penalty of each project (TP). If multiple computational experiments are performed for the same instance, ACs are the same minimum and TPs may be different values; 3) NR minimizes the adjustment cost of the baseline scheduling (AC), which requires the sum of tardiness penalties of projects (TP) to be the lowest. If multiple computational experiments are performed for the same instance, ACs are the same minimum while TPs are also a constant minimum.

From the characteristics of the FR problem, the sum of tardiness penalty of each project (including newly arrived projects) of the FR schedule is the optimal value that can be achieved after the arrival of new projects. Tables 4, 5, 6, and 7 show that the TP value of the NR schedule is always the smallest among the three reactive schedules, which is equal to the TP value of the FR schedule. Furthermore, the AC value of the NR schedule is less than the AC value of the FR schedule. In other words, the NR method performs as well as the FR method in achieving the goal of deterministic multi-project scheduling. However, in terms of minimizing the adjustment cost of the baseline schedule, the performance of the NR method outperforms that of the FR method. Accordingly, in terms of solution quality, the NR method outperforms the FR method overall.

Tables 4, 5, 6, and 7 show that the TP value of the NR schedule is far smaller than that of the OR schedule, and the gap between the two is significant. Tables 4, 5, 6, and 7 also indicate that the AC value of the NR schedule is larger than that of the OR schedule. However, the difference between them is minimal. In other words, in terms of minimizing the adjustment cost of the baseline schedule, the performance of the NR method is similar to that of the OR method, and the latter slightly outperforms the former. However, in achieving deterministic multi-project scheduling objectives, the performance of the NR method significantly outperforms the OR method, and the performance of the latter is very poor. Consequently, a conclusion can be drawn that the NR method significantly outperforms the OR method in terms of solution quality.

In summary, the NR method considers the adjustment cost of the baseline schedule and optimally achieves deterministic multi-project scheduling objectives, with distinct advantages over existing methods. On the one hand, in terms of minimizing the adjustment cost of the baseline schedule, the NR method significantly outperforms the FR method. On the other hand, the NR method significantly outperforms the OR method in terms of achieving deterministic multi-project scheduling objectives. Although the computational time of NR is longer than that of FR and OR, it is acceptable for the instances used in this paper. Each instance contains 3 projects, each with 20 activities and 4 types of renewable resources.

C. AN EXAMPLE OF SOLVING THE PRESENTED BI-OBJECTIVE PROBLEM

We randomly select 16-6-30, one of the 160 instances used in the previous section, as the example of this section. This example contains three projects. All the activity durations, resource requirements, and precedence relationships of the new project are the same as those of the first project. According to the setting of Section VI-A, the new project arrives at half the make-span of the baseline schedule. Other parameters are also set as those in Section VI-A. Then, the bi-objective RCMPRSP with new project arrival for the example is solved by the method described in Section V-B. The TP^{min} is valued at 3400 by solving the model M1, and TP^{max} is estimated at 6800 (2×3400). Since the tardiness penalty per unit time is 100, the increasing step of ε is set to 100, i.e., $g = 100$. The achieved candidate set is shown in Table 8, in which the dominated solutions after $\varepsilon > 5300$ are not listed. Remove all dominated solutions from the candidate set, and then the Pareto front of the problem is shown in Figure 1.

TABLE 8. The solutions in the candidate set.

	TP	AC	No-dominated
1	3400	70	Y
2	3500	70	N
3	3600	69	Y
4	3700	36	Y
5	3800	32	Y
6	3900	31	Y
7	4000	27	Y
8	4100	22	Y
9	4200	22	N
10	4300	22	N
11	4400	21	Y
12	4500	19	Y
13	4600	17	Y
14	4700	17	N
15	4800	8	Y
16	4900	7	Y
17	5000	5	Y
18	5100	0	Y
19	5200	0	N
20	5300	0	N

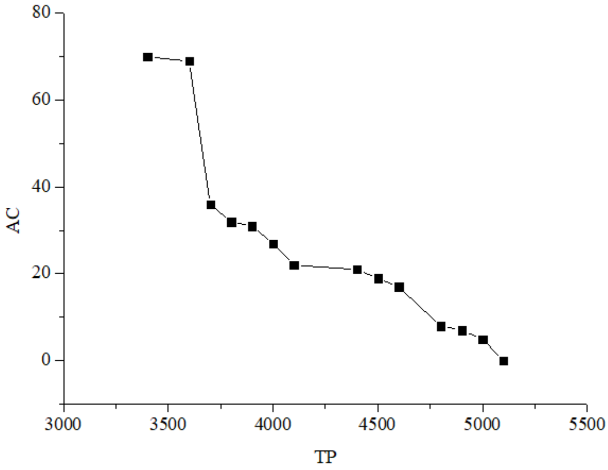

FIGURE 1. The Pareto front of the example.

In this paper, the tardiness penalty essentially represents the weighted durations of multiple projects. Therefore, the presented bi-objective problem is a duration/cost tradeoff problem. It can be seen from Figure 1 that the bi-objective problem proposed in this paper can well balance the multi-project duration and tardiness penalty after a new project arrives. Based on the presented bi-objective problem model, project managers can select an appropriate reactive schedule from the Pareto front according to the actual situation of the multi-project. Therefore, the presented bi-objective problem is a meaningful supplement to the current single objective reactive scheduling problems. It has the ability to provide project managers with higher decision-making ability when a new project arrives in the process of multi-project implementation.

VII. CONCLUSION

Existing reactive multi-project scheduling problems only consider minimizing the adjustment cost of the baseline schedule, ignoring the optimal achievement of minimizing the sum of tardiness penalty of each project. Consequently, they cannot achieve the reactive schedule minimizing the tardiness penalty of the multi-project after a new project arrives. In addressing the shortcomings of the existing reactive multi-project scheduling methods, this paper conducts an in-depth study on the resource-constrained multi-project reactive scheduling method. It proposes an RCMPRSP with a constraint to ensure that the deterministic multi-project scheduling goal is achieved to cope with a typical multi-project disruption event, new project arrival.

The main contributions of this paper are as follows. First, the model of the existing RCMPRSPs is improved. The proposed RCMPRSP with new project arrival aims at minimizing the adjustment cost of the baseline schedule and adds a constraint based on those of existing reactive multi-project scheduling problems to ensure that the reactive schedule is one of the optimal scheduling schemes in the new project execution environment. Second, a two-stage reactive multi-project scheduling method is proposed. In the first stage, the optimal value of the deterministic multi-project scheduling goal achievable in the new multi-project execution environment can be obtained by fully rescheduling the multi-project after the arrival of new projects. In the second stage, a new optimal scheduling scheme is formulated to minimize the adjustment cost of the baseline schedule that achieves the new optimal value of the deterministic multi-project scheduling objective.

Based on the benchmark instances, the proposed reactive scheduling method and the existing two reactive scheduling methods, including the FR method, are sufficiently compared and tested. The results demonstrate that the proposed reactive scheduling method considers the adjustment cost of the baseline schedule and optimally achieves the deterministic multi-project scheduling objective, with distinct advantages over existing methods. Since RCMPSP is an extension of reactive project scheduling problems, it is obviously also an NP-hard problem. The authors have previously tested this problem through the smallest single-mode project in project scheduling problem library (PSPLIB) and found the solution in this work cannot solve most of the multi-project instances with 120 activities in an acceptable time. It is found in the experiments of this work that some of the multi-project instances with 80 activities cannot be solved by the solution based on the IBM CPLEX. Therefore, it is necessary to develop efficient heuristic algorithms or meta-heuristic algorithms for the proposed RCMPRSP in the future.

REFERENCES

[1] E. Pérez, M. Posada, and A. Lorenzana, "Taking advantage of solving the resource constrained multi-project scheduling problems using multi-modal genetic algorithms," *Soft Comput.*, vol. 20, no. 5, pp. 1879–1896, May 2016.

[2] R. Van Eynde and M. Vanhoucke, "Resource-constrained multi-project scheduling: Benchmark datasets and decoupled scheduling," *J. Scheduling*, vol. 23, no. 3, pp. 301–325, Jun. 2020.

[3] T. Wauters, J. Kinable, P. Smet, W. Vancroonenburg, G. Van den Berghe, and J. Verstichel, "The multi-mode resource-constrained multi-project scheduling problem," *J. Scheduling*, vol. 19, no. 3, pp. 271–283, Jun. 2016.

[4] Ö. Hazır and G. Ulusoy, "A classification and review of approaches and methods for modeling uncertainty in projects," *Int. J. Prod. Econ.*, vol. 223, May 2020, Art. no. 107522.

[5] W. Herroelen and R. Leus, "Project scheduling under uncertainty: Survey and research potentials," *Eur. J. Oper. Res.*, vol. 165, no. 2, pp. 289–306, Sep. 2005.

[6] G. Zhu, J. F. Bard, and G. Yu, "Disruption management for resource-constrained project scheduling," *J. Oper. Res. Soc.*, vol. 56, no. 4, pp. 365–381, 2005.

[7] R. K. Chakrabortty, R. A. Sarker, and D. L. Essam, "Single mode resource constrained project scheduling with unreliable resources," *Oper. Res.*, vol. 20, no. 3, pp. 1369–1403, Sep. 2020.

[8] F. Deblaere, E. Demeulemeester, and W. Herroelen, "Reactive scheduling in the multi-mode RCPSP," *Comput. Oper. Res.*, vol. 38, no. 1, pp. 63–74, Jan. 2011.

[9] H. Chen, G. Ding, J. Zhang, and S. Qin, "Research on priority rules for the stochastic resource constrained multi-project scheduling problem with new project arrival," *Comput. Ind. Eng.*, vol. 137, Nov. 2019, Art. no. 106060.

[10] S. Elloumi, T. Loukil, and P. Fortemps, "Reactive heuristics for disrupted multi-mode resource-constrained project scheduling problem," *Expert Syst. Appl.*, vol. 167, Apr. 2021, Art. no. 114132.

[11] W. Wang, X. Ge, L. Li, and J. Su, "Proactive and reactive multi-project scheduling in uncertain environment," *IEEE Access*, vol. 7, pp. 88986–88997, 2019.

[12] W. Wang, J. Su, J. Xu, and X. Ge, "Reactive strategies in the multiproject scheduling with multifactor disruptions," *Math. Problems Eng.*, vol. 2020, pp. 1–11, May 2020.

[13] R. K. Chakrabortty, R. A. Sarker, and D. L. Essam, "Multi-mode resource constrained project scheduling under resource disruptions," *Comput. Chem. Eng.*, vol. 88, pp. 13–29, May 2016.

[14] R. K. Chakrabortty, H. F. Rahman, K. M. A. Haque, S. K. Paul, and M. J. Ryan, "An event-based reactive scheduling approach for the resource constrained project scheduling problem with unreliable resources," *Comput. Ind. Eng.*, vol. 151, Jan. 2021, Art. no. 106981.

[15] S. Van De Vonder, F. Ballestín, E. Demeulemeester, and W. Herroelen, "Heuristic procedures for reactive project scheduling," *Comput. Ind. Eng.*, vol. 52, no. 1, pp. 11–28, Feb. 2007.

[16] M. Ning, Z. He, N. Wang, and R. Liu, "Metaheuristic algorithms for proactive and reactive project scheduling to minimize contractor's cash flow gap under random activity duration," *IEEE Access*, vol. 6, pp. 30547–30558, 2018.

[17] H. Suwa and D. Morita, "Reactive project scheduling method to enhance project progress under uncertainty," *J. Adv. Mech. Design, Syst., Manuf.*, vol. 10, no. 3, pp. 1–15, 2016.

[18] W. Hu, H. Wang, C. Peng, H. Wang, H. Liang, and B. Du, "An outer–inner fuzzy cellular automata algorithm for dynamic uncertainty multi-project scheduling problem," *Soft Comput.*, vol. 19, no. 8, pp. 2111–2132, Aug. 2015.

[19] E. N. Afruzi, A. Aghaie, and A. A. Najafi, "Robust optimization for the resource-constrained multi-project scheduling problem with uncertain activity durations," *Scientia Iranica*, vol. 27, pp. 361–376, Jan. 2020.

[20] X. Wang, Q. Chen, N. Mao, X. Chen, and Z. Li, "Proactive approach for stochastic RCMPSP based on multi-priority rule combinations," *Int. J. Prod. Res.*, vol. 53, no. 4, pp. 1098–1110, Feb. 2015.

[21] D. Liu and Z. Xu, "A multi-PR heuristic for distributed multi-project scheduling with uncertain duration," *IEEE Access*, vol. 8, pp. 227780–227792, 2020.

[22] Y. Wang, Z. He, L.-P. Kerkhove, and M. Vanhoucke, "On the performance of priority rules for the stochastic resource constrained multi-project scheduling problem," *Comput. Ind. Eng.*, vol. 114, pp. 223–234, Dec. 2017.

[23] U. Satic, P. Jacko, and C. Kirkbride, "Performance evaluation of scheduling policies for the dynamic and stochastic resource-constrained multi-project scheduling problem," *Int. J. Prod. Res.*, vol. 60, no. 4, pp. 1411–1423, Feb. 2022.

[24] R. L. Burdett and E. Kozan, "Techniques for inserting additional trains into existing timetables," *Transp. Res. B, Methodol.*, vol. 43, nos. 8–9, pp. 821–836, Sep. 2009.

[25] S. Elloumi, P. Fortemps, and T. Loukil, "Multi-objective algorithms to multi-mode resource-constrained projects under mode change disruption," *Comput. Ind. Eng.*, vol. 106, pp. 161–173, Apr. 2017.

[26] J. Kuster, D. Jannach, and G. Friedrich, "Applying local rescheduling in response to schedule disruptions," *Ann. Oper. Res.*, vol. 180, no. 1, pp. 265–282, Nov. 2010.

[27] K.-K. Yang, "Effects of erroneous estimation of activity durations on scheduling and dispatching a single project," *Decis. Sci.*, vol. 27, no. 2, pp. 255–290, Jun. 1996.

[28] F. Deblaere, E. Demeulemeester, W. Herroelen, and S. Van De Vonder, "Robust resource allocation decisions in resource-constrained projects," *Decis. Sci.*, vol. 38, no. 1, pp. 5–37, Feb. 2007.

[29] O. Lambrechts, E. Demeulemeester, and W. Herroelen, "Proactive and reactive strategies for resource-constrained project scheduling with uncertain resource availabilities," *J. Scheduling*, vol. 11, no. 2, pp. 121–136, Apr. 2008.

[30] W. Zheng, Z. He, N. Wang, and T. Jia, "Proactive and reactive resource-constrained max-NPV project scheduling with random activity duration," *J. Oper. Res. Soc.*, vol. 69, no. 1, pp. 115–126, Jan. 2018.

[31] F. Zaman, S. M. Elsayed, R. Saker, and D. Essam, "Resource constrained project scheduling with dynamic disruption recovery," *IEEE Access*, vol. 8, pp. 144866–144879, 2020.

[32] Y. Zhao, N. Cui, and W. Tian, "A two-stage approach for the critical chain project rescheduling," *Ann. Oper. Res.*, vol. 285, nos. 1–2, pp. 67–95, Feb. 2020.

[33] Md. H. F. Rahman, R. K. Chakrabortty, and M. J. Ryan, "Managing uncertainty and disruptions in resource constrained project scheduling problems: A real-time reactive approach," *IEEE Access*, vol. 9, pp. 45562–45586, 2021.

[34] W. Zhang, Y. Yang, X. Liu, C. Zhang, X. J. Li, R. B. Xu, F. T. Wang, and M. A. Babar, "Decision support for project rescheduling to reduce software development delays based on ant colony optimization," *Int. J. Comput. Intell. Syst.*, vol. 11, pp. 894–910, Jan. 2018.

[35] Y. Ge and B. Xu, "Dynamic staffing and rescheduling in software project management: A hybrid approach," *PLoS ONE*, vol. 11, no. 6, Jun. 2016, Art. no. e0157104.

[36] R. Larsen and M. Pranzo, "A framework for dynamic rescheduling problems," *Int. J. Prod. Res.*, vol. 57, no. 1, pp. 16–33, Jan. 2019.

[37] S. Van De Vonder, E. Demeulemeester, and W. Herroelen, "A classification of predictive-reactive project scheduling procedures," *J. Scheduling*, vol. 10, no. 3, pp. 195–207, Jun. 2007.

[38] U. Beşikci, Ü. Bilge, and G. Ulusoy, "Multi-mode resource constrained multi-project scheduling and resource portfolio problem," *Eur. J. Oper. Res.*, vol. 240, no. 1, pp. 22–31, Jan. 2015.

[39] T. R. Browning and A. A. Yassine, "A random generator of resource-constrained multi-project network problems," *J. Scheduling*, vol. 13, no. 2, pp. 143–161, Apr. 2010.

WULIANG PENG received the Ph.D. degree in industrial engineering from Northeastern University, in 2008. He is currently a Full Professor with Yantai University. He has published articles in various journals, such as the *International Journal of Project Management*, the *International Journal of Production Economics*, and the *International Journal of Production Research*. He is a reviewer of several international journals.

DONGMIN YU received the B.S. degree in engineering from the Ningbo University of Engineering, Ningbo, China, in 2017. He is currently pursuing the master's degree in business management with Yantai University. His research interests include project management and technology economics and management.

JIALI LIN received the bachelor's degree in accounting from Lanzhou University, in 2019, and the master's degree in business management from Yantai University, in 2022. She has published two articles in *Annals of Operations Research* and *International Journal of Computing Science and Mathematics*.

• • •