IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 10 October 2023, accepted 13 December 2023, date of publication 25 December 2023,
date of current version 18 January 2024.

Digital Object Identifier 10.1109/ACCESS.2023.3346789

RESEARCH ARTICLE

Managing the Balance Between Project Value and Net Present Value Using Reinforcement Learning

CLAUDIO SZWARCFITER[1] AND YALE T. HERER[2]
[1]Faculty of Industrial Engineering and Technology Management, Holon Institute of Technology (HIT), Holon 5810201, Israel
[2]Faculty of Data and Decision Sciences, Technion—Israel Institute of Technology, Haifa 3200003, Israel

Corresponding author: Claudio Szwarcfiter (szwarcfiterc@hit.ac.il)

This work was supported in part by the European Institute of Innovation and Technology (EIT) Food, the Innovation Community on Food of EIT, a Body of the European Union (EU) under the Horizon 2020, the EU Framework Program for Research and Innovation under Project 19147; and in part by the Israel Science Foundation (ISF) under Grant 2550/21.

ABSTRACT Project managers make decisions weighing financial returns (net present value, NPV) and value creation expected by stakeholders. Often, plans maximizing NPV neglect stakeholder benefits while those focused strictly on value creation may reduce financial viability. This paper puts forth a new stochastic optimization model handling this compromise using a mixed integer program solved with reinforcement learning. The model incorporates uncertain activity durations and considers positive and negative cash flows. Our Monte Carlo control method with ϵ-greedy policies and timed start actions for activities facilitates the simultaneous maximization of NPV and project value. The resulting efficient frontier delineates various project plans, demonstrating the trade-off between maximizing NPV and project value, providing decision makers with visual analysis to select plans that fit organizational needs. Computational experiments demonstrate superior performance over a mathematical solver limited by the problem's complexity and a metaheuristic lacking guided online learning. The results help senior management select satisfactory plans that balance financial returns with stakeholder preferences. The methodology contributes a novel tool for quantitatively incorporating value creation alongside financial objectives in project planning.

INDEX TERMS Integer programming, project management, project scheduling, reinforcement learning, simulation.

I. INTRODUCTION

Solutions to the maximization of project net present value (max-NPV) problem are a sought-after commodity today. Decision makers need to evaluate different project alternatives, make go/no go decisions, and decide which projects will be part of their project portfolio [1]. It is common knowledge, however, that the evaluation of a project should not be based solely on financial considerations; a project can be unsuccessful by NPV criteria and yet deliver the expected value[1] to customers and other stakeholders. For example, [2] describes a construction company planning a major industrial safety campaign in response to its poor safety record and high insurance premiums. The project aims to reduce insurance costs by $250,000 annually and improve the company's ranking in an industry safety review from the 90th percentile to the 10th percentile. The project has a negative NPV of −$350,000, which, taken alone, may mean a no-go decision. Nevertheless, if the company's board also considers the value criterion — the improved industry ranking and that it outweighs the financial loss — it could decide to proceed.[2] Thus, project value is increasingly becoming a vital factor in Project Management [2].[3]

Project value can be defined as a combination of attributes that depend on the stakeholders' preferences ''such as

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

[1]Project value and benefit are used interchangeably in the literature. In this paper we use the term project value.

[2]On the flip side, project value alone may be an insufficient criterion for project evaluation, since the negative NPV may be prohibitive for the Board.

[3]Throughout this paper we understand value according to the definition found in [3]: ''Value is what the customer says it is, considers important, and is willing to pay for.''

features, functions, reliability, size, speed, availability, design aesthetics, etc.'' [4]. This paper adopts the framework used in [5], where the attributes are formulated as an objective function that reflects the value according to the customers' and stakeholders' preferences. In Section IV, we present an example of how the project value is calculated.

Research has tended to focus on the max-NPV problem and project value as separate research tracks rather than considering them together. We feel strongly that the consideration of both goals in tandem presents decision makers with a more thorough evaluation of a project when reviewing project alternatives. In this paper, we introduce a new formulation of the maximization problem that includes both NPV and project value, model a multimode setting that allows the consideration of different project plan alternatives, develop algorithms to solve the problem, and consider the tradeoff between realizing both objectives.

Other key components of our formulation are stochastic activity durations and the use of a robust form of NPV. Uncertainties are common in real-life projects and often result in budget and schedule overruns. According to one report that analyzed over 50,000 projects in 1,000 organizations, more than half (56%) of the projects went over the planned budget and almost two-thirds (60%) of the projects fell behind schedule [6]. More recent findings by the Project Management Institute indicate global figures of 38% and 45% for project budget and time overruns, respectively [7]. By focusing on stable solutions, we think that we provide a more relevant tool for decision makers than is available today.

The literature dealing with this paper's topic can be divided into two main branches: the max-NPV problem and project value management.

A. THE MAX-NPV PROBLEM

There is a considerable amount of research on the max-NPV problem. An early study of the deterministic problem was carried out by [8], where the objective function was linearized by approximation using the first terms in the Taylor expansion. Since then, more research on the max-NPV problem has accumulated. A review of past literature can be found in [9]; we focus on more current research. The problem has been extended to include resource constraints, in the resource-constrained project scheduling problem with discounted cash flows (RCP-SPDC). This is also an extension of the resource constrained project scheduling problem (RCPSP), which was proven to be NP-hard [10]. Gu et al. [11] offered an exact solution approach for the RCPSPDC limited to small projects and a Lagrangian relaxation with a decomposition method for large problems. Leyman and Vanhoucke [12] solved the RCPSPDC by constructing sets of activities and moving them together. Later they extended their work to include capital constraints and different cash outflow models [13]. Klimek [14] examined projects with payment milestones and different scheduling techniques such as activity right-shift, backward scheduling and left-right justification are compared. In [15], the RCPSPDC is solved by combining a genetic algorithm and an immune algorithm. The authors employ different crossover, mutation, and immunization operators and select the best one at each stage. In a similar research, the authors enhance the combined genetic and an immune algorithm with a variable insertion-based local search, a forward-backward improvement, a restart mechanism and an activity move rule to delay the activities with negative cash flow [16].

The multimode version of the RCPSPDC is an extension of the original problem. Projects with up to 30 activities and three modes are solved optimally with a network flow model in [17]. The scheduling technique in [12], mentioned above, is extended in [18] to include multimode projects and different payment models for cash inflows. Zhang et al. [19] balanced the NPV of the contractor and client in a bi-objective optimization problem.

Another extension of the original deterministic max-NPV problem is the stochastic max-NPV problem (denoted as SNPV by [20]) where the activity durations and cash flows are random variables ([20] present a detailed review of early literature on this topic). Creemers et al. [21] maximized the expected value of the NPV (eNPV) with variable activity durations, the risk of activity failure and different paths or modules to mitigate this risk, ignoring resource constraints. In a similar vein, [1] considered a general project failure risk that decreases with project progress, and activity-specific risks; earlier activity completion on the one hand eliminates its risk of failure, improving the eNPV, but on the other hand may also accelerate costs, which worsens the eNPV. Weather condition modeling was incorporated into stochastic durations by [22], where the decision variables are gates when resources are made available for specific activities.

Creemers [23] found globally optimal solutions for the SNPV problem where activity durations are phase-type distributed, cashflows are deterministic, and no resource constraints are considered. The author subsequently applied the results to finding the optimal sequence of stages in multistage sequential projects with stochastic stage durations, also obtaining exact, closed-form expressions for the moments of the NPV and using a three-parameter lognormal distribution to approximate the NPV distributions accurately [24]. He showed that the problem is equivalent to the least cost fault detection problem (LCFDP; this was also proven by [25]). Hermans and Leus [26] offered a new efficient algorithm and showed that in Markovian PERT networks, where activities are exponentially distributed and there are no resource constraints, the optimal preemptive solution solves the non-preemptive case as well. Two known proactive scheduling time buffering methods and two reactive scheduling models were employed by [27] to investigate the max-eNPV problem with stochastic activity durations. Time-buffer allocation was also proposed in [28], who added the expected penalty cost as a measure of solution robustness. Rezaei et al. [29] considered uncertainty in activity duration

and cash flow and two objectives: maximization of eNPV and minimization of NPV risk. Their model ignores resource constraints.

B. PROJECT VALUE MANAGEMENT

In [30] we reviewed the growing body of literature on project value management; here we cite the main references. A qualitative approach was taken by some researchers who study value in projects. For example, these researchers developed a framework to evaluate and formulate value [31]. They also examined the influence of value management on project success [32], [33], [34], suggested a scale to determine target values [35], explained how value is created [36], investigated value management in the dismantling of infrastructure projects [37], and contrasted the value of projects done offshore with those done domestically [38].

Another research direction that focuses on value is quantitative. It involves measuring the progress of product development based on the value added to customers [39], computing the value contribution to the project by staff member skills [40], quantifying value according to the attributes that matter to stakeholders [41], and developing a framework to plan and monitor cost, schedule, risk, and technical performance based on these attributes [4]. Some researchers in this quantitative field also use Quality Function Deployment (QFD) for project management. QFD is a well-known tool that captures the voice of the customer and converts it into engineering requirements [42]. It measures the value or performance of a product in multiple dimensions. We apply QFD to determine project value by using value parameters in the activity modes. Section IV shows an example of how we translate the voice of the customer into product value parameters and calculate the value of a specific project. Other recent papers apply QFD to project management [43], [44], [45].

A third research direction in project value is a new branch of research integrating project scope with product scope, which is the outlook we adopt in this paper. This research branch is characterized by expanding the idea of activity mode to cover not only cost and duration but also value parameters. Mode selection will, therefore, affect project value. In [46], a cost-effective design strategy is investigated, aiming to maximize the effectiveness-to-cost ratio and integrating decisions on project schedules, resource allocations, and product performance. Balouka et al. [5], on the other hand, extended the deterministic multimode resource-constrained project scheduling problem to include project value. Project management is combined with systems engineering in [47], who synchronize each activity mode with selected architectural components. Shtub et al. [48] and [49] described the use of simulation-based training in the integration of project and product scopes.

Table 1 summarizes the main features and characteristics of this and existing studies. It also highlights the gaps in the literature that this paper addresses. The most prominent lacuna is that none of the previous papers considers both NPV and value as objectives in the same model. The present study, in contrast, aims to find the efficient frontier between these two alternative goals. Another gap is that most papers that involve NPV assume single mode projects, whereas this paper deals with multimode projects, which allow generating alternate project plans that offer a range of value outcomes to stakeholders. Moreover, the present paper incorporates risk into the NPV calculation by using stochastic activity durations, while most papers that use NPV adopt deterministic models. Furthermore, we develop a quantitative optimization model for project value management, which is rare in the literature, as most papers focusing on project value are qualitative or descriptive. Finally, this paper employs reinforcement learning (RL) as a solution method, which is a novel and powerful approach for project management problems. To the best of our knowledge, the only previous paper that uses RL for project value management is our own [29], and this is the first paper that applies RL to project management problems related to NPV.

The aim of this paper is to model the tradeoff between project value and NPV in a multimode setting, where the selection of an activity mode will impact cost, duration, resource usage, and value, thus combining project scope (the tasks to be completed) with product scope (the characteristics and capabilities of the product and the resulting value) [50]. We consider stochastic activity durations to model realistic uncertain environments and introduce a new measurement of robustness in the NPV decision variable. Both objectives are introduced in a mixed integer program, and the evaluation of the objective function can be used to plot the efficient frontier (see Section IV for an example). To solve the proposed problem, we offer an innovative reinforcement learning (RL) based algorithm.

We have organized the rest of this paper in the following way. Section II presents the mathematical formulation. In Section III, our RL based solution is explained. We describe an example in Section IV and the experimental setting in Section V. Section VI presents our results, which are discussed in Section VII. Some conclusions are drawn in the final section.

II. THE PROPOSED MIXED INTEGER PROGRAM (MIP) FORMULATION

We formulate the problem as a mixed integer program (MIP). To model resource allocations, we employ a flow-based formulation adopted in many recent project scheduling works, e.g., [51], [52], [53], [54], [55], [56], [57], [58]. This formulation is especially suitable for stochastic models, where the activity start or finish times may vary according to the realized durations. The multimode setting, where each activity mode represents an alternative with its own time, cost, resource, and value parameters, is essential for the generation of different solutions on the efficient frontier of the project value/NPV curve. In a single-mode problem no change in

TABLE 1. A summary of the main features and characteristics of this and existing studies.

	Objective		Execution modes		Risk consideration		Model type		Solution method		
Reference	NPV	Value	Single mode	Multimode	Deterministic	Stochastic	Quantitative	Qualitative	Exact	Heuristic	RL
[8]	×		×		×		×		×		
[11]	×		×		×		×		×	×	
[12]–[16]	×		×		×		×			×	
[17]	×			×	×		×		×		
[18],[19]	×			×	×		×			×	
[1],[21],											
[23]–[26]	×		×			×	×		×		
[22],[27]–											
[29]	×		×			×	×			×	
[30]		×		×		×	×		×		×
[31]–[38],											
[48]		×						×			
[39],[4]		×	×			×	×				
[40]		×	×		×		×		×		
[41]		×				×	×				
[46]		×		×		×	×				
[5]		×		×	×		×		×	×	
[47]		×		×	×		×		×		
[49]		×		×		×		×			
This study	×	×		×		×	×		×		×

the project value is possible, and an efficient frontier cannot be constructed. The model seeks to maximize the robust project NPV and the project value. We tackle the chance constraints using a scenario approach (SA), introduced in [59] and applied in recent project scheduling papers [60], [61], [62]. The idea is to take S samples or scenarios of the realization of the random variables in the constraints—in our case, the activity durations—and substitute the deterministic scenario constraints for the stochastic chance constraints. Table 2 lists the mathematical model's sets, parameters, and decision variables.

Let us consider a project with J activities. Each activity j can be executed in one of M_j modes and is preceded by a set of immediate predecessors $\mathcal{P}(j)$. Each activity j executed in mode m in scenario $s \in \{1, \cdots, S\}$ has a duration d_{jms}. There are K different renewable resources, each with unit cost c_k per period. Activity j executed in mode m needs r_{jm}^k units of resource k, which has a total availability of $\mathcal{R}^k$. Apart from the duration-dependent resource costs, there is a fixed cash inflow or outflow c_{jm} associated with activity j executed in mode m, composed of fixed costs and payments received. Without loss of generality, we assume that payments are received or made at the end of each activity. The literature contains two main approaches to avoid gaps between activities and to prevent an activity with negative cash flow from being indefinitely postponed: 1) using a deadline [12] and 2) assuming a sufficiently large payoff at the end of the project that offsets the gains from postponing activities that affect project completion [21]. In this paper we adopt the latter approach.

For problems that seek to minimize project duration, a common robustness measure is the timely project completion probability (employed, for example, in [63]). We adopt this concept and define, in our problem, decision variable $rNPV$, the robust NPV, as the project NPV delivered with a probability of at least γ. This way, instead of applying the robustness measure to a given schedule, we search directly for a schedule with the desired robustness.

We set parameter NPV^{UP} as an upper bound for $rNPV$. Parameter $\widehat{r}$ is the discount rate, and EF_{js} and LF_{js} are the earliest and latest finish times for activity j in scenario s, respectively. $T_{\max}$ is an upper bound for the project duration.

Binary decision variable δ_{jm} indicates (value 1) if activity j is carried out in mode m (as presented in [64]) and decision variable $t_{js} \in \{EF_{js}, \ldots, LF_{js}\}$ denotes, for scenario s, the finish time of activity j, $j = 0, \ldots, J+1$, where activities 0 and $J+1$ are dummy activities (milestones) with a single mode, no duration, and no resources, and represent the start and end of the project, respectively. τ_s is a binary decision variable indicating (value 1) whether the scenario NPV is greater than $rNPV$. Decision variable β_{js} is the discount factor for activity j in scenario s and parameter β^{UP} is an upper bound for the discount factor. Binary decision variable z_{ij} indicates (value 1) if activity j starts after activity i finishes. The amount of resource k transferred from activity i to activity j is modeled by the flow variable ϕ_{ij}^k.

The project has V different value attributes. As noted in the Introduction, these attributes depend on the stakeholders' preferences (see there for examples of attributes). Let V_{jmv} be the parameter that represents the value of attribute v for activity j performed in mode m. Let V'_{jv} be the decision variable that denotes the value of attribute v for activity j performed in its chosen mode. We use a project-specific function $F_v(V'_{1v}, \cdots, V'_{Jv})$ that computes the project value for each attribute v based on the individual attributes V'_{jv} and a project-specific function $V''\left(F_1\left(V'_{11}, \cdots, V'_{J1}\right), \cdots, F_V\left(V'_{1V}, \cdots, V'_{JV}\right)\right)$ that determines the project value based on the values for each attribute (we introduced these value functions, decision variables and parameters in [50]). Parameters w_1 and w_2 represent the objective function weights for $rNPV$ and project value, respectively. By solving the MIP for different

TABLE 2. Sets, parameters, and decision variables for the mathematical model.

Sets	
$\mathcal{P}(j)$	Immediate predecessors of activity j
Parameters	
J	Number of project activities
M_j	Number of modes for activity j
S	Number of scenarios
d_{jms}	Duration of activity j executed in mode m in scenario s
K	Number of different renewable resources
c_k	Resource unit cost per period
r^k_{jm}	Units of resource k needed by activity j executed in mode m
$\mathcal{R}^k$	Total availability of resource k
c_{jm}	Fixed cash inflow or outflow associated with activity j executed in mode m
γ	Probability threshold set by the decision makers for the project to yield the robust NPV
NPV^{UP}	Upper bound for the robust NPV
$\widehat{r}$	Discount rate
EF_{js}	Earliest finish time for activity j in scenario s
LF_{js}	Latest finish time for activity j in scenario s
$T_{\max}$	Upper bound for the project duration
β^{UP}	Upper bound for the discount factor
V	Number of different project value attributes
V_{jmv}	Value of attribute v for activity j performed in mode m
w_1	Objective function weight for the robust NPV
w_2	Objective function weight for the project value
Decision variables	
$rNPV$	Robust NPV; project NPV delivered with a probability of at least γ
δ_{jm}	Binary decision variable indicating (value 1) if activity j is carried out in mode m
t_{js}	Finish time of activity j in scenario s
τ_s	Binary decision variable indicating (value 1) whether the scenario NPV is greater than $rNPV$
β_{js}	Discount factor for activity j in scenario s
z_{ij}	Binary decision variable indicating (value 1) if activity j starts after activity i finishes
ϕ^k_{ij}	Flow variable specifying the amount of resource k transferred from activity i to activity j
V'_{jv}	Value of attribute v for activity j performed in its chosen mode
t^p_{js}	Binary variables equal to 0 for all $p < t_{js}$ and 1 for all $p \geq t_{js}$
y_{jms}	Variables replacing the products $\beta_{js}\delta_{jm}$

weights w_1 and w_2, the efficient frontier between $rNPV$ and the project value can be determined.

We also employ additional variables for linearizing two constraints. Binary variables t^p_{js} are equal to 0 for all $p < t_{js}$ and 1 for all $p \geq t_{js}, p = 0, \cdots, T_{\max}$. Variables y_{jms} replace the products $\beta_{js} \cdot \delta_{jm}$. We now present the model, followed by an explanation of the objective function and constraints.

$$\text{Max } \left(w_1 \cdot rNPV + w_2 \cdot V'' \left(F_1\left(V'_{11}, \cdots, V'_{J1}\right), \cdots, F_V\left(V'_{1V}, \cdots, V'_{JV}\right)\right)\right), \tag{1}$$

subject to:

$$\sum_{j=0}^{J+1} \sum_{m=1}^{M_j} y_{jms} \left(c_{jm} + \sum_{k=1}^{K} c_k \cdot r^k_{jm} \cdot d_{jms}\right) + NPV^{UP}(1-\tau_s) \geq rNPV, \quad \forall s = 1, \cdots, S, \tag{2}$$

$$y_{jms} \leq \beta^{UP} \cdot \delta_{jm}, \quad \forall j = 0, \cdots, J+1, \forall m = 1, \cdots, M_j, \forall s = 1, \cdots, S, \tag{3}$$

$$y_{jms} \leq \beta_{js}, \quad \forall j = 0, \cdots, J+1, \forall m = 1, \cdots, M_j, \forall s = 1, \cdots, S, \tag{4}$$

$$y_{jms} \geq \beta_{js} - \left(1 - \delta_{jm}\right)\beta^{UP}, \quad \forall j = 0, \cdots, J+1, \forall m = 1, \cdots, M_j, \forall s = 1, \cdots, S, \tag{5}$$

$$y_{jms} \geq 0, \quad \forall j = 0, \cdots, J+1, \forall m = 1, \cdots, M_j, \forall s = 1, \cdots, S, \tag{6}$$

$$\beta_{js} = \sum_{p=1}^{T_{\max}} \left(1 + \widehat{r}\right)^{-p} \left(t^p_{js} - t^{p-1}_{js}\right), \quad \forall j = 0, \cdots, J+1, \forall s = 1, \cdots, S, \tag{7}$$

$$\sum_{p=1}^{T_{\max}} p \left(t^p_{js} - t^{p-1}_{js}\right) = t_{js}, \quad \forall j = 0, \cdots, J+1, \forall s = 1, \cdots, S, \tag{8}$$

$$\sum_{p=1}^{T_{\max}} \left(t^p_{js} - t^{p-1}_{js}\right) = 1, \quad \forall j = 1, \cdots, J+1, \forall s = 1, \cdots, S, \tag{9}$$

$$t^p_{is} \geq t^p_{js}, \quad \forall i \in \mathcal{P}(j), \forall j = 1, \cdots, J+1, \forall p = 0, \cdots, T_{\max}, \forall s = 1, \cdots, S, \tag{10}$$

$$t^p_{js} = 0, \quad \forall j = 1, \cdots, J+1, \forall p = 0, \cdots, EF_j - 1, \forall s = 1, \cdots, S, \tag{11}$$

$$t^p_{js} = 1, \quad \forall j = 1, \cdots, J+1, \forall p = LF_j + 1, \cdots, T_{\max}, \forall s = 1, \cdots, S, \tag{12}$$

$$t^p_{0,s} = 1, \quad \forall p = 0, \cdots, T_{\max}, \forall s = 1, \cdots, S, \tag{13}$$

$$\sum_{s=1}^{S} \tau_s \geq \gamma \cdot S, \tag{14}$$

$$z_{ij} + z_{ji} \leq 1, \quad \forall i = 0, \cdots, J, \forall j = 1, \cdots, J+1, \forall i < j, \tag{15}$$

$$z_{ij} + z_{jh} - z_{ih} \leq 1, \quad \forall i, j, h = 0, \cdots, J+1, \forall i \neq j \neq h, \tag{16}$$

$$z_{ij} = 1, \quad \forall i \in \mathcal{P}(j), \forall j = 1, \cdots, J+1, \tag{17}$$

$$t_{js} - \sum_{m=1}^{M_j} \delta_{jm} \cdot d_{jms} - M \cdot z_{ij} \geq t_{is} - M, \quad \forall i, j = 0, \cdots, J+1, \forall i \neq j, \forall s = 1, \cdots, S, \tag{18}$$

$$EF_{js} \leq t_{js} \leq LF_{js}, \quad \forall j = 0, \cdots, J+1, \forall s = 1, \cdots, S, \tag{19}$$

$$\phi_{ij} - \min\left(\tilde{r}^k_{im}, \tilde{r}^k_{jm'}\right) z_{ij} - \left(1 - \delta_{im}\right)\left(\tilde{r}^{\max,k}_{ij} - \min\left(\tilde{r}^k_{im}, \tilde{r}^k_{jm'}\right)\right) - \left(1 - \delta_{jm'}\right)\left(\tilde{r}^{\max,k}_{ij} - \min\left(\tilde{r}^k_{im}, \tilde{r}^k_{jm'}\right)\right) \leq 0,$$

where $\tilde{r}^{\max,k}_{ij} = \max\left(\max_{m=1,\cdots,M_i} \tilde{r}^k_{im}, \max_{m'=1,\cdots,M_j} \tilde{r}^k_{jm'}\right)$,

$$\text{and} \quad \tilde{r}^k_{jm} = \begin{cases} r^k_{jm} & \text{if} \quad 0 < j < J+1 \\ \mathcal{R}^k & \text{if} \quad j = 0 \quad \text{or} \quad j = J+1, \end{cases}$$
$$\forall i = 0, \cdots, J, \forall j = 1, \cdots, J+1, \forall i \neq j, \forall k = 1, \cdots, K,$$
$$\forall m = 1, \cdots, M_i, \forall m' = 1, \cdots, M_j, \tag{20}$$
$$\sum_{m=1}^{M_j} \delta_{jm} = 1, \quad \forall j = 0, \cdots, J+1, \tag{21}$$
$$\sum_{j \in \{1, \cdots, J+1\} \setminus \{i\}} \phi^k_{ij} = \sum_{m=1}^{M_i} \tilde{r}^k_{im} \cdot \delta_{im}, \quad \forall i = 0, \cdots, J,$$
$$\forall k = 1, \cdots, K, \tag{22}$$
$$\sum_{i \in \{0, \cdots, J\} \setminus \{j\}} \phi^k_{ij} = \sum_{m=1}^{M_j} \tilde{r}^k_{jm} \cdot \delta_{jm}, \quad \forall j = 1, \cdots, J+1,$$
$$\forall k = 1, \cdots, K, \tag{23}$$
$$0 \leq \phi^k_{ij} \leq \min \left(\max_{m=1, \cdots, M_i} \tilde{r}^k_{im}, \max_{m=1, \cdots, M_j} \tilde{r}^k_{jm} \right),$$
$$\forall i = 0, \cdots, J, \forall j = 1, \cdots, J+1, \forall i \neq j,$$
$$\forall k = 1, \cdots, K, \tag{24}$$
$$V'_{jv} = \sum_{m=1}^{M_j} \delta_{jm} \cdot V_{jmv}, \quad \forall v = 1, \cdots, V,$$
$$\forall j = 1, \cdots, J. \tag{25}$$

The objective function (1) aims to maximize a weighted sum of the project's *rNPV* and value. The weighted-sum approach is commonly used in multi-objective optimization in general [65] and is applied in a number of project scheduling papers (for example, [28]; [66]). Constraints (2) indicate whether a scenario's NPV is greater than the project's *rNPV*. Initially, these constraints would be nonlinear because the positive or negative cash flow associated with each activity mode, $c_{jm} + \sum_k c_k \cdot r_{jkm} \cdot d_{jms}$, would have to be multiplied by the discount factor variable and the indicator variable, $\beta_{js} \cdot \delta_{jm}$, indicating that the cash flow would have to be discounted according to the finish time and realized only for the selected mode. To avoid this nonlinearity, we use variables y_{jms} in constraints (2). Constraints (3)–(6) guarantee that $y_{jms} = \beta_{js} \cdot \delta_{jm}$.

We use a discrete discount factor as in [67], which has the form $\beta_{js} = \left(1 + \widehat{r}\right)^{-t_{js}}$. Constraints (7) linearize this exponential function. Constraints (8) link the binary variables t^p_{js} with t_{js} and constraints (9) make sure that an activity only finishes once. Constraints (10) further bound t^p_{js}, since a predecessor will always assume the value of 1 before its successor. Likewise, constraints (11) and (12) fix the value of t^p_{js} for finish times before the early finish and after the late finish, respectively, and constraints (13) fix the value for the initial dummy activity. Constraint (14) counts the fraction of scenarios that yield the *rNPV* and force this fraction to remain above the predetermined threshold.

The following constraints were introduced by us in a prior conference paper [64]. Constraints (15) and (16) avoid cycles of 2 and 3 or greater, respectively, thus guaranteeing that the network is acyclic [51], [68]. Constraints (17) enforce the precedence constraints. Constraints (18) link the continuous activity finish time variables with the binary sequencing variables. Constraints (19) give upper and lower bounds for the activity finish times. Constraints (20), from [51], connect the continuous resource flow variables with the binary sequencing variables and the binary mode variables. Constraints (21) force the selection of only one mode per activity. Outflow constraints (22) ensure that all activities, except for $J+1$, send their resources to other activities. Inflow constraints (23) ensure that all activities, except for activity 0, receive their resources from other activities. Constraints (24) bound the flow variables with the maximum resource consumption modes. Finally, constraints (25), which we introduced in [50], determine the value attributes according to the selected modes.

With the linearization of the constraints described above, if the project's value function is linear, the MIP is a mixed integer linear program (MILP) and can be solved with a commercial solver. We use this method as a benchmark in the computational experiments (Section V).

We previously presented a scenario-based MIP model for the multimode RCPSP (MRCPSP) with the objective of minimizing the duration in [64]. In this paper, we extend that model by incorporating the following innovations: 1) The new objective function that jointly maximizes the robust NPV and the project value; 2) Additional constraints that capture the NPV and value aspects of the project; 3) The extra variables and constraints for linearizing the non-linear terms in the model.

III. THE REINFORCEMENT LEARNING SOLUTION

From learning to play backgammon at near the level of the world's best players [69], through landing unmanned aerial vehicles (UAVs) [70], beating the highest ranked players in Jeopardy! [71], and human-level performance in Atari games [72], RL has been successful in applications for uncertain environments. This success is the factor motivating our application of RL to the formulation discussed in Section II. RL-based heuristics have been applied to project scheduling [73], [74], [75], [76], but to the best of our knowledge, [30] is the only study that tackled multimode problems involving chance constraints.

The RL model begins with an agent in state $\mathcal{S}$. It takes action $\mathcal{A}$ and transitions to state $\mathcal{S}'$, earning reward $\mathcal{R}'$. Then it performs action $\mathcal{A}'$, transitioning to state $\mathcal{S}''$, and earning reward $\mathcal{R}''$, and so on. The agent's life trajectory can be expressed as $\mathcal{S}, \mathcal{A}, \mathcal{R}', \mathcal{S}', \mathcal{A}', \mathcal{R}'', \mathcal{S}'', \mathcal{A}'', \mathcal{R}''', \mathcal{S}''', \mathcal{A}'''$, etc. The agent follows a policy $\pi\ (\mathcal{S}, \mathcal{A})$ that indicates which action it should choose at each state. The goal of the RL problem is to learn a policy that maximizes the agent's reward. We also define an action-value function $q\ (\mathcal{S}, \mathcal{A},)$ as the estimated reward for choosing action $\mathcal{A}$ on state $\mathcal{S}$ and then following policy $\pi\ (\mathcal{S}, \mathcal{A})$ [30].

TABLE 3. Additional notation for the RL method.

π	ϵ-greedy policy, decision-making rule
$q(j,m,t)$	Value of choosing mode m and start time t for activity j under ϵ-greedy policy π
$\pi(j,m,t)$	Probability of selecting mode m and start time t for activity j under ϵ-greedy policy π
$\mathcal{R}(j,m,t)$	Reward for selecting mode m and start time t for activity j under ϵ-greedy policy π
ϵ	Probability of random action in ϵ-greedy policy
$\hat{m}_j$ or $\hat{m}$	Selected mode for activity j
$\hat{t}_j$	Selected start time for activity j
η	Parameter specifying the number of possible start times to select from
$d^{ML}_{j\hat{m}}$	Most likely duration of activity j in selected mode $\hat{m}$
$r^k_{j\hat{m}}$	Quantity of resources of type k needed to execute activity j in selected mode $\hat{m}$
$A(j)$	Set of activities scheduled in parallel to activity j
N	Number of times mode m_j and start time t_j are selected for activity j
α	Step-size parameter

Applying the RL model to the formulation presented in Section II, we define a state as project activity j. The agent undertakes an action by choosing a mode $\hat{m}_j$ and start time $\hat{t}_j$ for activity j, and then moves on to the next activity. After selecting modes and start times for all activities $j = 1, \cdots, J$, it can calculate its reward $\mathcal{R}(j,m,t)$. As it receives rewards, it learns the action-value function $q(j,m,t)$ and which policy $\pi(j,m,t)$ to follow.

The RL method that we apply in this paper is Monte Carlo control (MCC), based on [77]. We employed MCC because it fits best the problem at hand, making full use of Monte Carlo simulation to run the project plans, determine the cumulative probability distributions for *rNPV*, and obtain an exact value of the reward for each simulation run, without the need for bootstrapping, i.e., estimating the reward based on another estimate. MCC is a state-of-the-art RL method that has been employed in recent works such as [78], [79], [80], [81], [82], [83], [84], [85], and [86] to solve various problems in different domains. Table 3 summarizes our RL notation, in addition to the notation employed in the quantitative model. Our pseudocode and an explanation of our MCC method follows. The main procedure is shown in Algorithm 1.

Our algorithm starts with the initialization of the action-value list (Algorithm 2). For each activity, the action taken is selecting the mode and the start time. We use η start times, equally spaced from zero to an upper bound, the maximum sequential project duration. We initialize the table with artificially high values, a technique known as optimistic initial values, in order to allow initial exploration of all actions.

The action-value list is then used to calculate the policy (Algorithm 3). To balance exploration and exploitation, we adopt an ϵ-greedy policy, meaning that in the policy list, we ascribe a probability ϵ of taking a random action and a probability $(1-\epsilon)$ of taking a greedy action, i.e., the action with the highest action value.

Next, we take an action based on the policy (Algorithm 4), selecting, for each activity, the mode and start time according to the probabilities in the policy list. Then, by right-shifting the activities, adding to each start time the finish time of the latest-finishing immediate predecessor, we adjust the start times to make them precedence-feasible. This means that if we select a start time of $\hat{t}_j$ for activity j, we right-shift this activity to start at time $\hat{t}_j$ after its immediate predecessor finishes. The finish times are determined using the most likely duration of each activity in its selected mode.

Algorithm 1 Main Procedure for MCC

initialize_action_values$\left(J, \eta, M_j, LS_j, \forall j = 1, \cdots, J\right)$ from Algorithm 2;
while *not stopping criterion*:
  calculate_policy$\left(J, \eta, M_j, q(j,m,t), \forall j = 1, \cdots, J\right)$ from Algorithm 3;
  choose_mode_start$\left(\pi(j,m,t), d^{ML}_{j\hat{m}}, \mathcal{P}(j), \forall j = 1, \cdots, J\right)$ from Algorithm 4;
  calculate_reward$\left(\text{sorted}(\hat{m}_j), \mathcal{P}(j), \eta, d^{ML}_{j\hat{m}}, r^k_{j\hat{m}}, R^k, \forall j = 1, \cdots, J, \forall k = 1, \cdots, K\right)$ from Algorithm 5;
  update_action_values_RL_1 $\left(J, \hat{m}_j, \hat{t}_j, \forall j = 1, \cdots, J\right)$ from Algorithm 6;
  or
  update_action_values_RL_2 $\left(J, \hat{m}_j, \hat{t}_j, \forall j = 1, \cdots, J\right)$ from Algorithm 7;

Algorithm 2 Initialization of the Action-Value List

def *initialize_action_values*$\left(J, \eta, M_j, LS_j, \forall j = 1, \cdots, J\right)$:
  for *activity* $j = 1, \cdots, J$:
    for *mode* $m = 1, \cdots, M_j$:
      for *start time* $t = 0, \frac{LS_j}{\eta-1}, \frac{2LS_j}{\eta-1}, \cdots, LS_j$:
        $q(j,m,t) =$ large number;
  return $q(j,m,t), \forall j = 1, \cdots, J, \forall m = 1, \cdots, M_j, \forall t = 0, \frac{LS_j}{\eta-1}, \frac{2LS_j}{\eta-1}, \cdots, LS_j$

Algorithm 3 Policy Calculation

def *calculate_policy*$\left(J, \eta, M_j, q(j,m,t), \forall j = 1, \cdots, J\right)$:
  for *activity* $j = 1, \cdots, J$:
    $q^* = \max_{m,t} q(j,m,t)$;
    $x =$ number of action values for which $q(j,m,t) = q^*$;
    $\pi(j,m,t) =$
    $$\begin{cases} \frac{1}{x}\left(1 - \frac{\epsilon}{\eta \cdot M_j}\left(\eta \cdot M_j - x\right)\right), \forall m, t \mid q(j,m,t) = q^* \\ \frac{\epsilon}{\eta \cdot M_j}, \forall m, t \mid q(j,m,t) \neq q^*; \end{cases}$$
  return $\pi(j,m,t), \forall j = 1, \cdots, J, \forall m = 1, \cdots, M_j, \forall t = 0, \frac{LS_j}{\eta-1}, \frac{2LS_j}{\eta-1}, \cdots, LS_j$

Algorithm 4 Select Activity Mode and Start Time

def *choose_mode_start*$\left(\pi\ (j,m,t)\ , d_{j\hat{m}}^{ML}, \mathcal{P}\ (j)\ , \forall j=1,\cdots,J\right)$**:**
  for *activity* $j=1,\cdots,J$**:**
    choose $\hat{m},\hat{t}$ according to $\pi\ (j,m,t)$;
    if $\mathcal{P}\ (j)==\varnothing$**:**
      $\hat{t}_j^*=\hat{t}_j$;
    else:
      $\hat{t}_j^*=\hat{t}_j+\max\left(\hat{t}_i^*+d_{i\hat{m}}^{ML}, \forall i\in\mathcal{P}\ (j)\right)$;
  return sorted $\left(\hat{m}_j \mid j\in\{1,\cdots,J\},\right.$
  $\left.\hat{m}_j\in\{1,\cdots,M_j\}, \text{key}=\hat{t}_j^*\right)$

Algorithm 5 Calculating the Reward

def *calculate_reward*$\left(sorted\ \left(\hat{m}_j\right), \mathcal{P}\ (j)\ , \eta, d_{j\hat{m}}^{ML},\right.$
$\left. r_{j\hat{m}}^k, R^k, \forall j=1,\cdots,J, \forall k=1,\cdots,K\right)$**:**
  $t^*_{\text{sorted}(\hat{m}_j)[0]}=0$;
  for *activity mode* $\hat{m}_j$ *in sorted* $\left(\hat{m}_j\right)[1:\ ]$**:**
    $\mathcal{I}=b-a$, where $b=\max\left(t_j^*+d_{j\hat{m}}^{ML}\right), a=$
    $\min\left(t_j^*\mid t_j^*\geq t_i^*+d_{i\hat{m}}^{ML}, \forall i\in\mathcal{P}(j)\right)$;
    $t_j^*=\min\Big(t_j\mid t_j\geq\Big[a, a+\frac{\mathcal{I}}{\eta-1}, a+2\frac{\mathcal{I}}{\eta-1},$
    $\cdots, b\Big]\ [\pi\ (j,m,t)\ .\text{index}]\ \left(\hat{t}_j\right)$ and $r_{j\hat{m}}^k\leq$
    $R^k_{\text{surplus}}, \forall k=1,\cdots,K\Big)$, where
    $R^k_{\text{surplus}}=\min_{\left[t_j, t_j+d_{j\hat{m}}^{ML}\right)}\Big(R^k$
    $-\sum_{i\in A(j)} r_{i\hat{m}}^k\Big)$;
  return $\mathcal{R}\left(j,\hat{m}_j,\hat{t}_j\right)=w_1\cdot rNPV$
  $+w_2\cdot V''\Big(F_1\left(V'_{11},\cdots,V'_{J1}\right),\cdots,F_V\left(V'_{1V},\cdots,\right.$
  $\left.V'_{JV}\right)\Big)\mid \Pr\left[\text{NPV}\geq rNPV\right]\geq\gamma, \forall j=1,\cdots,J$

Thereafter, we sort all activities according to their adjusted start times, obtaining a precedence-feasible activity list with the activities and their selected modes. The construction of this precedence-feasible activity list is the first of two steps of the implementation of the start time selection. The second step (Algorithm 5, described below) is implemented in the calculate_reward function.

Note that the selection of start times to generate an activity list is really a surrogate for selecting different combinations of precedencies between the activities. The range of possible start times between zero and the upper bound provides ample options of early start or postponement of each activity, providing a richer search space with the possibility of better solutions. Furthermore, the adjustment to generating only precedence-feasible activity lists avoids both wasting runtime with infeasible solutions and discarding potentially good solutions.

The next step in the algorithm is to calculate the reward for the actions taken (Algorithm 5). Here, we implement the second step of the start time choice: the insertion of each activity in the baseline schedule. We handle each activity sequentially. First, we determine the interval between the earliest precedence-feasible start and the latest activity finish time of the activities scheduled until this point. This interval is divided into η equal periods and we start the activity according to the index of its start time $\hat{t}_j$ in the policy list. For example, if $\hat{t}_j$ is the third start time in the policy list, we use the third period in the interval, rounding it to the nearest activity finish time. If there are not enough resources, we repeatedly right-shift the activity to the next scheduled activity finish time until there are enough resources. With the schedule in place, we calculate the objective function value. To calculate *rNPV* we simulate the NPV cumulative distribution function (CDF). For example, if the decision-makers desire a 95% probability of delivering the *rNPV*, the baseline schedule is simulated 1000 times, the realized NPVs are sorted in increasing order, and the 50$^{\text{th}}$ element of the NPV list is the *rNPV*. We define the reward as the objective function value. As pointed out in Section II, to calculate the objective function value we define weights w_1 and w_2 for *rNPV* and project value, respectively (equation 1); repeating the algorithm for different weight values gives us different points on the efficient frontier.

In [64], we introduced an RL algorithm for the MRCPSP with the objective of minimizing the project duration. In this paper, we extend that algorithm by incorporating a novel feature: Algorithm 5, which allows the agent to select the start time of each activity from a set of feasible options, rather than always choosing the earliest possible start time. This feature enables the agent to account for the impact of positive and negative cash flows. For instance, when an activity has a negative cash flow, delaying its start time can increase the NPV by reducing the present value of the cash outflow.

The last step in the algorithm is to update the action-value list using the reward. We can choose from two update methods, RL_1 (Algorithm 6) and RL_2 (Algorithm 7). For both, we only update the action values corresponding to the selected modes and start times. RL_1 learns an action value by averaging all the rewards this action (mode and start time) has received each time it was taken. This signifies that new rewards have an increasingly smaller impact the more the actions are taken. The means are calculated incrementally to speed up the process and save memory. RL_2 updates the action values using a formula very similar to the incremental mean from RL_1, but instead of using the decreasing step $\frac{1}{N}$, it uses a constant step α, giving an exponentially large weight to the last action. These methods are explained in [77]; RL_1 and, to a lesser extent, RL_2 appear in the recent MCC papers listed above in this section [78], [79], [80], [81], [82], [83], [84], [85], [86].

We repeat the process, calculating the policy based on the updated action values, selecting the modes and start times based on the policy, calculating the reward, and updating

Algorithm 6 Action-Value Update Using Average Rewards (RL_1)

def *update_action_values_RL*$_1\left(J, \hat{m}_j, \hat{t}_j, \forall j = 1, \cdots, J\right)$**:**
  for *activity* $j = 1, \cdots, J$**:**
    $q\left(j, \hat{m}_j, \hat{t}_j\right) += \frac{1}{N}\left(\mathcal{R}\left(j, \hat{m}_j, \hat{t}_j\right) - q\left(j, \hat{m}_j, \hat{t}_j\right)\right)$;
  return $q\left(j, \hat{m}_j, \hat{t}_j\right), \forall j = 1, \cdots, J$

Algorithm 7 Action-Value Update Using Constant Step (RL_2)

def *update_action_values_RL*$_2\left(J, \hat{m}_j, \hat{t}_j, \forall j = 1, \cdots, J\right)$**:**
  for *activity* $j = 1, \cdots, J$**:**
    $q\left(j, \hat{m}_j, \hat{t}_j\right) += \alpha\left(\mathcal{R}\left(j, \hat{m}_j, \hat{t}_j\right) - q\left(j, \hat{m}_j, \hat{t}_j\right)\right)$;
  return $q\left(j, \hat{m}_j, \hat{t}_j\right), \forall j = 1, \cdots, J$

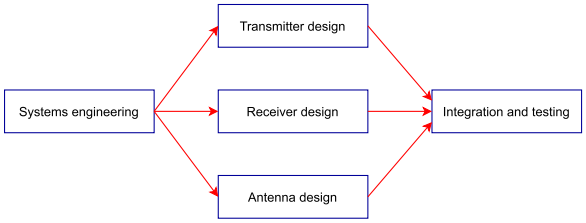

FIGURE 1. Project network diagram.

the action values, until reaching a stopping criterion. The solution takes the form of a baseline schedule consisting of the selected activity modes and start times, the *rNPV* and the project value.

IV. EXAMPLE

We use a radar development example, a simplified version of a real project [30], to demonstrate our problem and RL solution approach. The AON network of the project is shown in Fig. 1 and Table 4 gives the five project activities with two modes each, the durations (O, ML and P) for optimistic, most likely and pessimistic scenarios, the fixed cost (FC) of each activity, the resources per period needed for each activity mode (engineers, E, and technicians, T), the value parameters, and the income received after completing the activity. Three activities have negative cash flows, comprising the fixed and resource costs, and two of them have positive cash flows due to the income.

This is a practical example of how to define and compute value. The value attributes of range, quality, and reliability (R, Q and Re in Table 4) reflect the needs and expectations of the project stakeholders. They are influenced by the value parameters in each activity mode. We use the notation from Section II and have three value attributes, $V = 3$. The radar equation [87] is applied to calculate the range (R), quality (Q) and reliability (Re) of the radar system [46], since they depend on technical parameters such as transmitter power and antenna gain, which vary according to the technological alternatives considered for each mode. The mode selection for the project plan will affect not only the value, but also the cost and NPV, thus integrating both project value components.

We now present the value functions for each attribute, $F_v\left(V'_{1v}, \cdots, V'_{Jv}\right), J = 5, v = 1, 2, 3$. The equation for the radar range (R) is $F_1 = ([TP] \cdot [RS] \cdot [AG])^{0.25}$, where $[TP]$ is transmitter power, $[RS]$ is receiver sensitivity and $[AG]$ is antenna gain, extracted from the activities of "transmitter design" (TD), "receiver design" (RD), and "antenna design" (AD), respectively, in Table 4. Using the notation from Section II, $[TP]$, $[RS]$, and $[AG]$ are decision variables V'_{21}, V'_{31}, and V'_{41}, respectively. When we select one of the two modes, say, for the TD activity, we are also determining which of the parameter values, V_{211} (50 in Table 4) or V_{221} (100 in Table 4), will be assigned to the decision variable V'_{21} ($[TP]$ in this example). This same mechanism applies for decision variables V'_{31} and V'_{41} ($[RS]$ and $[AG]$), allowing us to compute the value of function F_1, the value for the (R) attribute.

The equation for the radar quality (Q) is $F_2 = 100\,[SEQ] \cdot [QT] \cdot [QR] \cdot [QA] \cdot [QI]$, where the factors $[SEQ]$, $[QT]$, $[QR]$, $[QA]$, and $[QI]$ indicate the impact of systems engineering, transmitter, receiver, antenna, and integration on quality, respectively. The equation for the radar reliability (Re) is $F_3 = 100\,[AR] \cdot [IR] \cdot [TR] \cdot [RR]$, where the factors $[AR]$, $[IR]$, $[TR]$, and $[RR]$ denote the reliability of antenna design, integration effort, transmitter, and receiver, respectively. The value of the project $V''\left(F_1\left(V'_{11}, \ldots, V'_{J1}\right), \ldots, F_V\left(V'_{1V}, \ldots, V'_{JV}\right)\right)$ is calculated by a weighted sum of the three value attributes, a technique that is widely used in multi-attribute utility theory [88]: $V'' = \frac{7}{21}F_1 + \frac{8}{21}F_2 + \frac{6}{21}F_3$.

There are a total of 11 engineers and four technicians available and the resource unit costs per period are \$100 for engineers and \$50 for technicians. We want to solve the problem for different weights w_1 and w_2 and find the efficient frontier for an *rNPV* probability of 95%. The result is shown in Fig. 2 with four non-dominated points. We reached similar objective values using RL_1 and RL_2. For convenience, we normalized the project values to be between 0 and 100 as in [5]. Decision makers can conduct a tradeoff analysis and select the solution that best meets stakeholders' needs and requirements.

As explained in Section III, *rNPV* is determined in each iteration by simulating the NPV CDF. For the point (76.62, 40,772) in Fig. 2, the CDF plot is shown in Fig. 3 and the *rNPV* is marked. For any solution that the decision makers select, a baseline schedule can be constructed easily by the process highlighted in Algorithm 5 and explained in Section III. The solution highlighted above produces the Gantt chart shown in Fig. 4. The activity durations are the most likely durations from the three-point estimates, and the selected modes are shown next to the activities.

It is interesting to visualize how our RL agent learns better solutions. Recall from Section III that the agent wants to learn the best actions, i.e., select activity modes and start times

TABLE 4. Summary of data for radar development activity modes.

Activity	Mode	Duration			FC	Resources		Value parameters			Income
		O	ML	P		E	T	R	Q	Re	
SE	Small team	5	7	10	2000	1	0		0.8		
	Large team	3	4	4	4000	3	1		0.99		
TD	Reengineer	3	5	8	5000	2	1	50	0.99	0.9	52,500
	New design	7	9	11	10,000	4	2	100	0.95	0.8	
RD	Reengineer	3	5	9	2000	2	1	30	0.95	0.9	
	New design	8	10	11	15,000	3	1	200	0.8	0.99	
AD	Reengineer	3	7	9	3000	3	2	10	0.8	0.9	
	New design	6	7	9	7000	5	2	30	0.99	0.9	
I&T	In-house	3	4	4	4000	3	3		0.99	0.9	20,000
	Subcontract	2	2	5	6000	1	0		0.9	0.99	

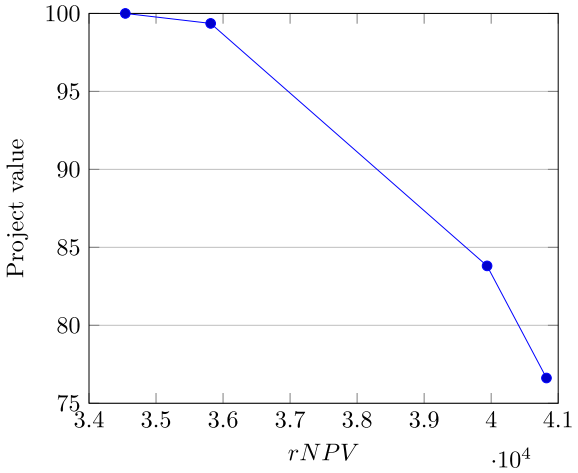

FIGURE 2. Efficient frontier for radar project.

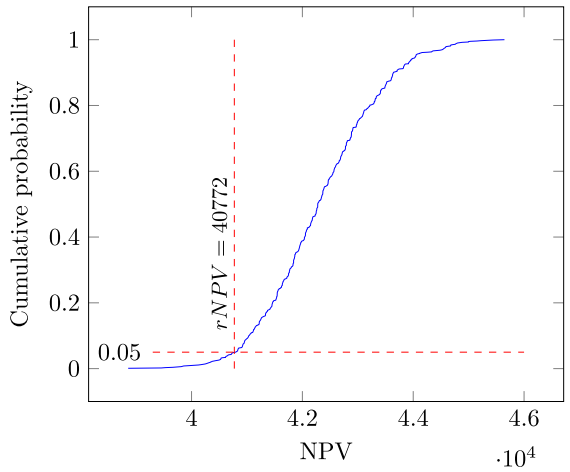

FIGURE 3. NPV cumulative distribution for the solution with *rNPV* = 40,772.

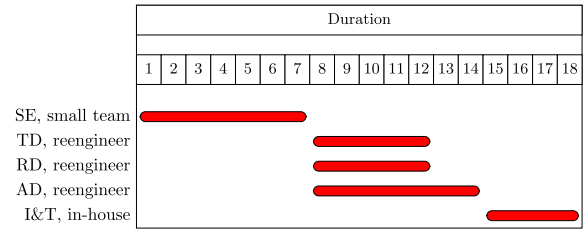

FIGURE 4. Gantt chart for a project with value = 76.62 and *rNPV* = 40,772.

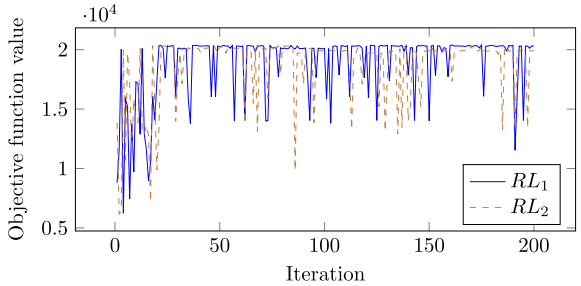

FIGURE 5. Radar example learning curves for RL_1 and RL_2: 95% *rNPV* probability, $w_1 = w_2 = 0.5$.

that will maximize its reward. In our RL model, we defined the reward as the weighted sum of *rNPV* and project value; thus, the agent will learn the action values, generate policies from the action values, and take actions based on the policies, seeking to maximize the objective function.

Fig. 5 exhibits the learning curves for both action-value updating variants, RL_1 and RL_2. We see at the beginning of the curves the effect of the optimistic initial values (Section III): even though a near-maximum objective was found early on, the agent kept searching haphazardly, ''thinking'' that it could receive a better reward by taking other actions, since the action-value list was initialized with artificially high values. The objective eventually stabilized on about 20,000. Since we used ϵ-greedy policies (Section III), sometimes the agent still wanted to explore, so the project delivery never settled completely on the maximum, jumping occasionally to smaller objective values.

V. HYPOTHESES AND EXPERIMENTAL SETTING

In this section, we explain our factorial experiment, summarized in Table 5. The experiments were conducted to prove two hypotheses:

A. HYPOTHESIS H_1

Our RL methods are suitable for solving the formulation presented in Section II, compared to established benchmarks.

B. HYPOTHESIS H_2

The start time selection RL actions can be leveraged for solving the problem with positive and negative cashflows.

TABLE 5. Fractional factorial design.

	RL_1	RL_2	Solver	TS	TS$^{\mathrm{D}}$	RL_1^{ES}	RL_2^{ES}
10+			×	×	×	×	×
20+			×	×	×	×	×
50+				×	×	×	×
100+				×	×	×	×
10^{+}_{-}	×	×	×			×	×
20^{+}_{-}	×	×	×			×	×
50^{+}_{-}	×	×				×	×
100^{+}_{-}	×	×				×	×

To evaluate H_1, we selected two additional methods as benchmarks. First, we solved our MILP from Section II using the Python interface for the commercial solver Gurobi version 9.5, and second, we employed tabu search (TS), applied in several project scheduling studies.[4] In the different TS applications in the literature, the general TS algorithm, described in [91], is tweaked specifically for the problem at hand. We selected a multimode RCPSPDC application [92], closer to our subject matter, thus simplifying the adaptation of TS to our problem.[5] Additionally, we compared the performances of the algorithms for three project sizes, each with three modes per activity: 10 and 20 activities, which are small problems with greater potential of quickly covering a wider search space and obtaining faster solutions, and 50 and 100 activities, closer to real-life projects. To evaluate H_2 we ran our RL algorithm using both methods for updating the action values described in Section III, RL_1 and RL_2, and compared them with RL_1^{ES} and RL_2^{ES}, a simplification of RL_1 and RL_2 where all activities are scheduled as early as possible.

For the 10- and 20-activity projects we used the complete PSPLIB J10 and J20 datasets [93], and for the 50- and 100-activity projects, the complete MMLIB50 and MMLIB100 datasets [94]. These datasets are the standard in the multimode project management literature [95]. We analyzed projects with two types of cash flows: positive cash flows only (+), and cash flows that are both positive and negative ($^{+}_{-}$). In the former, the NPV criterion is a regular scheduling objective, meaning that in a given schedule it is never beneficial to delay an activity if it could be scheduled earlier. In the latter, this observation does not hold [96].

Because of the long runtimes, the solver runs were performed only for 10- and 20-activity projects.[6] The TS application in [92] was developed for a deterministic multimode RCPSPDC problem with positive cash flows, and thus in this paper we use TS as a benchmark only for settings with positive cash flows (see Appendix A for more details about our TS implementation).

We calculated the activity start times for RL_1^{ES} and RL_2^{ES} in the calculate_reward function (Algorithm 5), as $\hat{t}_j^* = \min\{t_j \mid t_j \geq t_i^* + d_{i\hat{m}}^{ML}, \forall i \in \mathcal{P}(j) \text{ and } r_{j\hat{m}}^k \leq R_{\text{surplus}}^k, \forall k = 1, \cdots, K\}$. The start times in TS were also calculated this way. R_1^{ES} and R_2^{ES} were the only RL methods used with the positive cashflow instances. In the positive and negative cashflow instances, they were used for comparison to evaluate the improvement in the objective function obtained by selecting the start times instead of starting as early as possible.

The stopping criterion for all RL methods was 1000 iterations after having visited all states with optimistic initial values. For TS, we used two stopping criteria: the maximum runtime between RL_1^{ES} and RL_2^{ES} for the corresponding instance[7] and double this time (TS$^{\mathrm{D}}$). For the solver, because of the long runtimes, we set the gap between the lower and upper objective bounds to 10% (Gurobi parameter MIPGap = 0.1) and a maximum runtime of 30 minutes (Gurobi parameter timeLimit = 1800). We employed 1000 scenarios for the solver for the 10-activity runs and 100 scenarios for the 20-activity experiments (because of the long runtimes); we used 1000 simulation runs to calculate each RL reward and TS objective function.

To determine the durations of different activity modes, we employed a three-point estimation technique. The dataset's duration was defined as the most likely duration, while the pessimistic duration was set at 2.25 times this value. Similarly, the optimistic duration was determined to be half of the most likely duration. These factors, which can be found in [97], align with the widely recognized observation that activity durations in project management tend to be skewed towards longer durations (refer to [98] for an example). To simulate realized durations for the activities, we utilized a triangular distribution, a commonly used method in project simulation (see the scenario presented in [99]). The resulting durations were then rounded to the nearest integer. The optimistic, most likely and pessimistic durations were used for the triangular distribution lower limit, mode and upper limit parameters, respectively [100].

The objective function was evaluated with weights $w_1 = w_2 = 0.5$. We set γ, the desired probability of the project to yield the $rNPV$, to 0.95. We defined the discount rate $\hat{r} = 0.01$ per period and generated positive activity mode cashflows, randomly drawing from uniform distributions from the interval (0, 10), and positive and negative cashflows from the interval (-100, 100). At the end dummy activity, a final payment of 10 was received in the experiments with positive cash flows; in the positive and negative cash flows, the final payments were 1000, 2000, 5000, and 10,000 for 10, 20, 50, and 100 activities, respectively.

To tune the RL algorithm parameters, we undertook a full factorial experiment based on the F-Race algorithm, following [101]. The inputs for F-Race are a target algorithm

[4] A literature review on TS applications in project scheduling falls outside the scope of this paper. Recent research includes [27], [28], [89], and [90].

[5] We opted for TS because in that publication it produced smaller maximal relative deviations from the best solutions than simulated annealing.

[6] In our tests the solver could not generate a single incumbent solution for a sample of four 50-activity projects after 48 hours of runtime. Even when we tried to run the sample for 100 scenarios instead of 1000, after 6.8 hours the solver was still running the linear relaxation and had not yet started to solve the MIP.

[7] We wished to allow TS at least the same RL runtime.

TABLE 6. Configurations evaluated in the factorial experiment.

RL algorithm	RL configurations evaluated
RL_1, RL_1^{ES}	$\epsilon = 0.1, 0.05, 0.01$; $\eta = 2, 5, 10$ (all 9 combinations)
RL_2, RL_2^{ES}	$\epsilon = 0.1, 0.05, 0.01$; $\eta = 2, 5, 10$; $\alpha = 0.1, 0.05, 0.01$ (all 27 combinations)

(in our case, the RL algorithms), a set of configurations, a set of problem instances, and a performance metric (in our case, the objective function). F-Race internally employs statistical tests to guide its search process, as follows. When a minimum sample of instances is run for all configurations, a rank-based Friedman test is performed. If the test indicates significant performance differences, Wilcoxon signed rank (WSR) pairwise tests are executed and the configurations with inferior performance are gradually eliminated (for more details, refer to [101]). The advantage of using F-Race in the factorial experiment is that we do not need to run all instances for all configurations, only for the ''winners'' at each step.

To run F-Race in our experiments, we randomly shuffled the complete datasets. In most cases, one best configuration (with the highest objective function values) was found after running some instances; in some cases, there were ties. We conducted the Friedman and WSR tests with a significance level of 0.05. In all experiments, after finding the best configuration, we continued running it on the remaining instances; thus, the best configurations were run on the complete datasets. Table 6 details the configurations evaluated.[8]

We used two value attributes ($V = 2$) and defined their relative weights as 0.6 and 0.4. We established an additive project value function F_v for each attribute, forming the linear objective function $0.5rNPV + 0.5\left(0.6\sum_{j=1}^{J} V'_{j1} + 0.4\sum_{j=1}^{J} V'_{j2}\right)$ that could be tackled by Gurobi. The value parameters V_{jmv} were drawn from uniform distributions from the interval (0, 10) for the experiments with positive cash flows and from the interval (0, 100) for positive and negative cash flows.

The algorithms were coded in Python. We ran all experiments on a computer with an Intel(R) Core (TM) i7-7700 CPU 3.60GHz, 8 GB RAM. To analyze the data, we conducted pairwise comparisons of the objective function value generated by each method and used JMP to calculate the p-values (p) for the WSR tests with a significance level of 0.05. Pareto analysis was also used to gain more insight into the results.

VI. RESULTS

This section begins by examining the leading RL algorithm configurations found by the full factorial experiment. We then present the experiment results for instances with positive cash flows, and those with cash inflows and outflows. The files with the datasets used and the results obtained can be accessed in [102].

[8]These parameter values are found in examples in [77] and other RL resources.

TABLE 7. Best configurations. Those for RL_1 and RL_1^{ES} experiments are represented by the tuple $(\epsilon; \eta)$, and for RL_2 and RL_2^{ES} experiments, by the tuple $(\epsilon; \eta; \alpha)$.

	RL_1	RL_2	RL_1^{ES}	RL_2^{ES}
10+	(0.1; 2)	(0.1; 2; 0.1)	(0.1; 2)	(0.1; 2; 0.1)
20+	(0.1; 2)	(0.1; 2; 0.1)	(0.1; 2)	(0.1; 2; 0.1)
50+	(0.1; 2)	(0.1; 2; 0.1)	(0.1; 2)	(0.1; 2; 0.1)
100+	(0.1; 2)	(0.1; 10; 0.1)	(0.1; 2)	(0.1; 2; 0.1)
10^+_-	(0.1; 2, 5, 10)(*)	(0.1; 2; 0.1)	(0.1; 2, 5)(*)	(0.1; 10; 0.1)
20^+_-	(0.1; 2)	(0.1; 2; 0.1)	(0.1; 2)	(0.1; 2; 0.1)
50^+_-	(0.1; 2)	(0.1; 10; 0.1)	(0.1; 2)	(0.1; 10; 0.1)
100^+_-	(0.1; 2)	(0.1; 10; 0.1)	(0.1; 2)	(0.1; 10; 0.1)

(*)Two or more values listed for a parameter (separated by a comma) mean that no significant difference was found between those configurations.

A. PARAMETER TUNING FACTORIAL EXPERIMENT

Table 7 lists the top configurations for all RL algorithms and experiments.

The best value found for the probability of random action ϵ and for step-size parameter α was 0.1. As regards the number of possible start times to select from, η, discounting two ties, the top values were 2, followed by 10. In our shared results [102], we show the F-Race process gradually narrowing down to the best configurations.

B. POSITIVE CASH FLOWS

Strong evidence of the suitability of the RL methods was found. Table 8 presents the results of the pairwise comparison. The average percent difference (%dif) and WSR p-value (p) for each pair of methods is shown.

TS and TS^D generated objectives closest to the solver values in the smaller 10- and 20-activity projects, outperforming RL_1^{ES} and RL_2^{ES}. For the larger 50- and 100-activity projects, however, RL_1^{ES} outperformed the TS algorithms, and RL_2^{ES} only lost to TS^D in 50-activity projects. The average difference between the solver solutions and other methods increased for 20 activities in relation to ten activities.

Note that throughout this subsection and the next one, we considered only solver solutions with a maximum gap between the lower and upper objective bounds of 0.1 rounded up to the nearest tenth. The solutions with larger gaps were inferior; including them, thus, would distort the results. Please refer to Appendix B for the results with gaps larger than 10%.

Table 9 reports the number of times each method generated the highest objective value. The results are in complete agreement with the pairwise comparison shown above. Where the solver found a solution within the time limit, the MILP solution generated more best solutions. Otherwise, TS^D gave better results for ten and 20 activities, RL_1^{ES} outperformed the other methods for 50 and 100 activities, and RL_2^{ES} outperformed TS^D for 100 activities.

C. POSITIVE AND NEGATIVE CASH FLOWS

The tests showed that RL_1 generated the objective values closest to the MILP solver solutions and outperformed the other methods. The MILPs for the 20-activity projects, which had more difficulty in generating feasible solutions in the

TABLE 8. Pairwise comparison between the objective values for projects with positive cash flows only. Data is for the pairwise difference between the row value and the column value.

J		RL_1^{ES}		RL_2^{ES}		TS		TS^D	
		%dif	p	%dif	p	%dif	p	%dif	p
10	Solver	1.76	< 0.0001	2.57	< 0.0001	1.47	< 0.0001	1.40	< 0.0001
	RL_1^{ES}			0.79	< 0.0001	-0.22	< 0.0001	-0.28	< 0.0001
	RL_2^{ES}					-1.00	< 0.0001	-1.06	< 0.0001
	TS^D					0.06	< 0.0001		
20	Solver	7.12	< 0.0001	10.6	< 0.0001	4.98	< 0.0001	4.85	< 0.0001
	RL_1^{ES}			3.46	< 0.0001	-0.97	< 0.0001	-1.45	< 0.0001
	RL_2^{ES}					-4.26	< 0.0001	-4.72	< 0.0001
	TS^D					0.48	< 0.0001		
50	RL_1^{ES}			5.67	< 0.0001	7.23	< 0.0001	2.71	< 0.0001
	RL_2^{ES}					1.49	< 0.0001	-2.79	< 0.0001
	TS^D					4.41	< 0.0001		
100	RL_1^{ES}			1.71	< 0.0001	4.09	< 0.0001	3.81	< 0.0001
	RL_2^{ES}					2.36	< 0.0001	2.09	< 0.0001
	TS^D					0.27	< 0.0001		

TABLE 9. Pareto count of highest objective values (in percentage).

J		Solver	RL_1^{ES}	RL_2^{ES}	TS	TS^D
10	solver solution	82.5	6.0	0	2.1	11.5
10	no solver solution		34.9	6.0	28.5	59.0
20	solver solution	75.6	14.6	0	2.4	9.8
20	no solver solution		27.8	0.6	9.5	71.6
50			68.5	0.2	0	31.3
100			67.0	18.7	5.7	14.3

30-minute runtime, only produced statistically significant comparisons with the RL_2 variants. Table 10 highlights the results of the pairwise comparisons. We omitted the differences $RL_1 - RL_2^{ES}$ and $RL_2 - RL_1^{ES}$ because we are interested in comparing start time selection with early start schedule generation for the same RL methods (Hypothesis H_2). Accordingly, in all cases, the non-early start strategies generated better results than the early start ones.

The count of the number of times each method generated the highest objective value (Table 11) reflects the results obtained above. RL_1 found more best solutions than RL_2 and RL_1^{ES}, and RL_2 outperformed RL_2^{ES}.

Our results lead us to accept Hypotheses H_1 and H_2, validating both the quality of our RL results, particularly RL_1, and the leverage of RL start time selections to increase the objective values.

VII. DISCUSSION

Although all values for the probability of random action ϵ and step-size parameter α inputted into the F-Race algorithm are found in the literature, the best value found for both parameters, 0.1, is consistent with [77], where this configuration is the most common one employed.[9]

If we turn to the number of possible start times from which to select η, in most RL variants and project sizes, two start-time actions, an early start and a late one, were sufficient to generate the highest objective values. Presumably, there is a balance between, on one hand, the improved learning generated by the higher frequency in which each start-time action is taken due to the fewer number of start-time options, and on the other hand, the potential gains accrued by a wider range of start-time alternatives. In most cases, the improved learning offset the finer start-time tuning. In some cases, $\eta = 10$ was found to be superior; understandably, with one exception, this was observed in the larger 50- and 100-activity projects, where the longer project durations could warrant the need for more intermediary start-time actions between the early and late starts.

As hypothesized, our experiments validate the usefulness of RL as a method for analyzing the tradeoff between the project value and its net present value compared to established benchmarks (Hypothesis H_1). Our RL agent captures a signal at each iteration (i.e., the reward) indicating how good the solution is, and immediately acts upon this signal. Therefore, from the beginning, an informed search is launched based on online information. TS, in contrast, is a neighborhood search with a memory mechanism to avoid being trapped in local optima. It does not use information obtained during the search to direct its next steps. Apparently, this works well for smaller projects, where the search space can be thoroughly covered by TS's local search mechanism. For example, for 10-activity projects, the average difference between TS^D and TS was only 0.06% (Table 8) and in 238 instances (almost half the dataset) the difference was null, meaning that the search space was already covered before doubling the runtime. This lack of learning, however, hampers TS's ability to embark on more promising sections of the search space earlier on, and this factor could well explain why RL_1^{ES} outperformed TS for 50- and 100-activity projects, even when TS was given double the time.

As far as TS is concerned, the results point to the likelihood of a deterioration in the quality of its solutions vis-'a-vis RL_1^{ES} as the projects increase in size. This can be seen in the 50- and 100-activity projects if we observe the significant differences between the solutions (Table 8; for 100 activities RL_2^{ES} also outperformed TS^D) and the smaller percentage of instances where the TS algorithms rendered the highest objective values

[9]This configuration is also common in the well-known data science and machine learning resources such as towardsdatascience.com and geeksforgeeks.org.

TABLE 10. Pairwise comparison between the objective values for projects with cash flows that are both positive and negative. Data is for the pairwise difference between the row value and the column value.

J		RL_1		RL_2		RL_1^{ES}		RL_2^{ES}	
		%dif	p	%dif	p	%dif	p	%dif	p
10	Solver	4.34	< 0.0001	5.51	< 0.0001	4.79	< 0.0001	6.20	< 0.0001
	RL_1			1.04	< 0.0001	3.43	< 0.0001		
	RL_2							4.01	< 0.0001
20	Solver	-2.01	0.8256	4.04	0.0164	-1.50	0.6892	3.30	0.0096
	RL_1			4.58	< 0.0001	4.94	< 0.0001		
	RL_2							4.97	< 0.0001
50	RL_1			2.89	< 0.0001	6.63	< 0.0001		
	RL_2							6.40	< 0.0001
100	RL_1			2.40	< 0.0001	7.32	< 0.0001		
	RL_2							7.87	< 0.0001

TABLE 11. Pareto count of highest objective values (in percentage).

J		Solver	RL_1	RL_2	RL_1^{ES}	RL_2^{ES}
10	solver solution	62.7	27.4	5.9	3.6	0.3
10	no solver solution		48.5	12.6	3.6	0.2
20	solver solution	27.6	44.8	0	27.6	0
20	no solver solution		76.9	1.6	19.9	0.2
50			63.7	15.4	18.0	3.0
100			61.9	19.6	16.5	2.0

(Table 9). It would seem that the explanation reported above applies here also: larger search spaces slow down the process of exploring superior search space sections because of TS's lack of exploitation or learning.

A significant difference was identified between RL_1, RL_2 and their early start counterparts, confirming Hypothesis H_2. These values correlate fairly well with [13] and further support the idea of NPV improvement by moving activity start times. We accomplished this in the RL framework by integrating the activity moves into the RL actions. The first step of the start time selection was implemented for the experiments with positive cash flows where the RL actions generated an activity list that was then decoded into a unique early start schedule (this process is known as a serial scheduling scheme; for a detailed review on this topic, see [103]). This worked well with the positive cash flows because, as pointed out in Section V, for a regular scheduling objective it is never beneficial to delay an activity if it could be scheduled earlier. The second step of the start time selection was implemented for the experiments with positive and negative cash flows, where the activity list no longer generated a unique early start schedule, but rather a schedule with start times determined by the start time selection actions.

As expected, the solver generated the best results. Nevertheless, as was noted in the Introduction, the problem is NP-hard and thus the long runtimes prevent the use of this method for larger problems. Even for 10-activity projects, the solver could not find an incumbent solution after the 30-minute limit in 39% of the projects for the experiments with positive cash flows,[10] and in 43% of the projects for the experiments with positive and negative cash flows. For 20-activity projects, these figures grow to 85% and 95%, respectively, even with the reduced number

[10]Note that even for positive cash flows, an early-start schedule is not necessarily feasible because of the resource constraints. This problem is, thus, NP-hard, which explains the long runtimes.

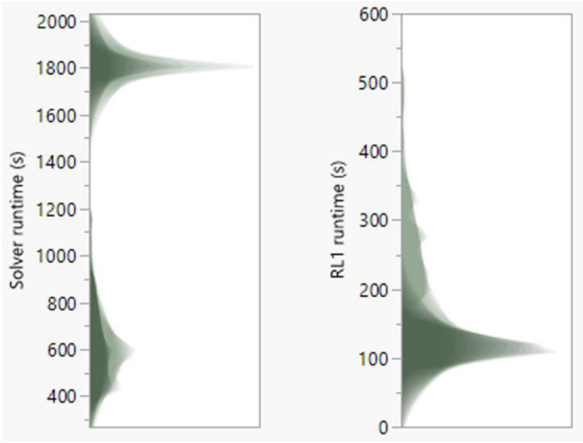

FIGURE 6. Solver and RL_1^{ES} runtime distributions for experiment with positive cash flows and ten activities. The high frequency on 1800s corresponds to the runtime limit.

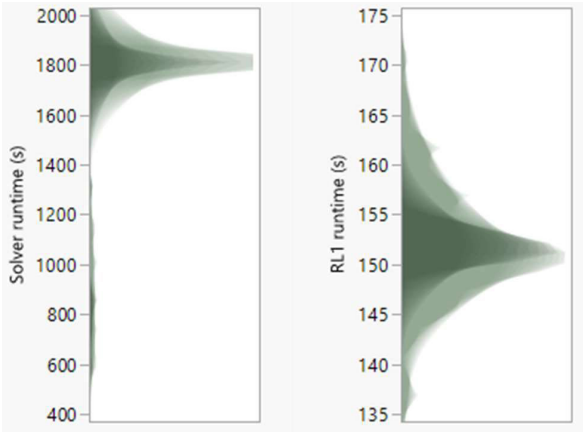

FIGURE 7. Solver and RL_1 runtime distributions for experiment with positive and negative cash flows and ten activities. The high frequency on 1800s corresponds to the runtime limit.

of scenarios considered in the MILPs. Comparing the solver and RL_1 runtime distributions for both experiments (Figs. 6 and 7), MILP solver-based solutions tend to be a less interesting option.

We were surprised to find that RL_1 and RL_1^{ES} outperformed RL_2 and RL_2^{ES} in all our experiments.[11] Since the constant-step action-value update gives larger weight to the last actions

[11]We applied RL_2 trying to maximize a project value function and in some cases RL_2 outperformed RL_1 [30].

TABLE 12. Pairwise comparison between the objective values obtained with RL and TS and those obtained with the solver for positive cashflows for 10-activity projects. WSR tests for the 20-activity projects did not show statistical significance.

RL_1^{ES}		RL_2^{ES}		TS		TS^D	
%dif	p	%dif	p	%dif	p	%dif	p
10.65	< 0.0001	10.98	< 0.0001	10.53	< 0.0001	10.49	< 0.0001

TABLE 13. Pairwise comparison between the objective values obtained with RL and those obtained with the solver for positive and negative cash flows.

J	RL_1		RL_2		RL_1^{ES}		RL_2^{ES}	
	%dif	p	%dif	p	%dif	p	%dif	p
10	36.79	< 0.0001	36.65	< 0.0001	34.04	< 0.0001	31.81	< 0.0001
20	30.31	0.0020	24.94	0.0020	29.79	0.0020	23.85	0.0039

and exponentially less weight to previous ones, we would think that the RL_2 and RL_2^{ES} results could be more promising: the last decisions tend to be better because of the learning and ascribing them more weight could more quickly point to better policies. It would appear that RL_2 and RL_2^{ES} could find acceptable results with fewer iterations than RL_1 and RL_1^{ES}; however, after more iterations, while RL_1 and RL_1^{ES} stabilize the action values by averaging the rewards, RL_2 and RL_2^{ES} over-emphasize the last decisions, good or bad. This short memory causes the forgetting of near-optimal policies that could maximize the objective value. Further research is required to consider potential upgrades to the constant-step methods.

Finally, our findings suggest that analyzing the tradeoff between the project value and its NPV using the RL method can be a valuable tool for project managers. The near-optimal solutions obtained can be used to plot the efficient frontier between project value and *rNPV* and decision makers can conduct a tradeoff analysis to select the project plan that satisfies stakeholders' requirements sufficiently.

VIII. CONCLUSION

This paper has investigated the tradeoff between project value and its NPV in a stochastic multimode setting. We have presented an MIP formulation for the problem using a flow-based model with a project-specific value function and a robust NPV decision variable, and modeled its robustness by means of chance constraints, tackled using a scenario approach. We have employed linearization techniques that allowed us, in the case of linear benefit functions, to produce MILP models that could be solved for small projects by a commercial solver.

We have found a cutting-edge solution for the MIP formulation using RL and illustrated its application with an example. We have designed and conducted a fractional factorial experiment and obtained satisfactory results showing that the RL method is suitable for solving our formulation (Hypothesis H_1) and that the activities' start time selection can be leveraged as RL actions for solving the problem with positive and negative cashflows (Hypothesis H_2). Furthermore, this work has revealed that our RL method is able to tackle large multimode projects with 50 and 100 activities, where the search space is very large.

The usefulness of our contribution lies in finding the efficient frontier between the robust NPV and the project value, enabling the decision makers to make focused tradeoffs between different alternatives of project plans. Since these two factors represent the project scope and the product scope, decision makers are presented with a more thorough evaluation of each project alternative.

While results demonstrate the promise of the proposed approach, scalability to highly complex projects remains untested. Performance on large-scale programs with hundreds of project activities or multi-year durations may expose limitations in the optimization efficiency. Additionally, expanding benchmarking to include comparisons against a wider range of emerging metaheuristic and hyperheuristic algorithms for project scheduling problems could further validate effectiveness. Generalizability also requires examination through real-world project data case studies and evaluation in multiproject environments.

Several meaningful extensions present avenues for advancing the model. Applying state-of-the-art function approximation techniques may enhance scalability for mega-projects. Variations that combine project- and portfolio-level goals or integrate additional objectives such as flexibility metrics could significantly increase applicability to practice.

APPENDIX A
THE TABU SEARCH (TS) ALGORITHM USED IN THIS PAPER

As a benchmark we customized the general TS algorithm found in [91] and adopted the solution representation and neighborhood moves published in [92]. The following adaptations were made for the formulation presented in Section II:

- The objective function (1) from Section II was used instead of the original pure NPV objective. As pointed out in Section V, TS was designed for a regular objective. Since our value function is time-independent, it does not affect the regular objective attribute and thus TS can be applied.
- The TS method was published for deterministic problems. To calculate the objective function in our stochastic problem, we proceeded as in the RL algorithms: we simulated 1000 project runs as shown in Algorithm 5 and explained in Section III, with the early start simplification explained in Section V.
- There were no penalty functions since all our solutions are feasible.

A discussion of TS falls outside the scope of this paper. More details on this topic can be found in [91] and in the references cited in footnote 3.

APPENDIX B
RESULTS FOR MIP GAPS LARGER THAN 10%

As was noted in the Results section, the solver results for large MILP gaps were not considered because their low quality would distort the results. In this subset of instances, the solver was outperformed by all the other methods both for the experiments with positive cashflows (Table 12) and for those with positive and negative cashflows (Table 13).

REFERENCES

[1] W. Zhao, N. G. Hall, and Z. Liu, "Project evaluation and selection with task failures," *Prod. Oper. Manage.*, vol. 29, no. 2, pp. 428–446, Feb. 2020, doi: 10.1111/poms.13107.

[2] O. Zwikael and J. R. Smyrk, *Project Management: A Benefit Realisation Approach*. Cham, Switzerland: Springer, 2019, doi: 10.1007/978-3-030-03174-9.

[3] J. Oehmen, Ed., "The guide to lean enablers for managing engineering programs," Joint MIT-PMI-INCOSE Community Pract. Lean Program Manag., Cambridge, MA, USA, 2012. [Online]. Available: https://dspace.mit.edu/bitstream/handle/1721.1/70495/oehmenetal2012-theguidetoleanenablersformanagingengineeringprograms.pdf?sequence=4

[4] T. R. Browning, "Planning, tracking, and reducing a complex project's value at risk," *Project Manage. J.*, vol. 50, no. 1, pp. 71–85, 2019. [Online]. Available: http://www.itdashboard.gov, doi: 10.1177/8756972818810967.

[5] N. Balouka, I. Cohen, and A. Shtub, "Extending the multimode resource-constrained project scheduling problem by including value considerations," *IEEE Trans. Eng. Manag.*, vol. 63, no. 1, pp. 4–15, Feb. 2016. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7353135, doi: 10.1109/TEM.2015.2497209.

[6] J. Johnson, "CHAOS report 2015 edition," Standish Group Int., Inc., Boston, MA, USA, Tech. Rep., 2015. [Online]. Available: https://www.standishgroup.com/sample_research_files/CHAOSReport2015-Final.pdf

[7] Project Management Institute, "Beyond agility: Flex to the future," Newtown Square, PA, USA, Tech. Rep. EXEC-014-2019 (10/19), 2021. [Online]. Available: https://www.pmi.org/-/media/pmi/documents/public/pdf/learning/thought-leadership/pulse/pmi_pulse_2021.pdf?v=b5c9abc1-e9ff-4ac5-bb0d-010ea8f664da&sc_lang_temp=en

[8] A. H. Russell, "Cash flows in networks," *Manage. Sci.*, vol. 16, no. 5, pp. 357–373, Jan. 1970.

[9] V. Tantisuvanichkul, "Optimizing net present value using priority rule-based scheduling," Ph.D. dissertation, School Mech., Aerosp. Civil Eng., Univ. Manchester, Manchester, U.K., 2014.

[10] J. Blazewicz, J. K. Lenstra, and A. H. G. R. Kan, "Scheduling subject to resource constraints: Classification and complexity," *Discrete Appl. Math.*, vol. 5, no. 1, pp. 11–24, Jan. 1983, doi: 10.1016/0166-218X(83)90012-4.

[11] H. Gu, A. Schutt, P. J. Stuckey, M. G. Wallace, and G. Chu, "Exact and heuristic methods for the resource-constrained net present value problem," in *Handbook on Project Management and Scheduling*, vol. 1. Cham, Switzerland: Springer, Jan. 2015, pp. 299–318, doi: 10.1007/978-3-319-05443-8_14.

[12] P. Leyman and M. Vanhoucke, "A new scheduling technique for the resource-constrained project scheduling problem with discounted cash flows," *Int. J. Prod. Res.*, vol. 53, no. 9, pp. 2771–2786, 2015. [Online]. Available: https://www.tandfonline.com/action/journalInformation?journalCode=tprs20, doi: 10.1080/00207543.2014.980463.

[13] P. Leyman and M. Vanhoucke, "Capital- and resource-constrained project scheduling with net present value optimization," *Eur. J. Oper. Res.*, vol. 256, no. 3, pp. 757–776, Feb. 2017, doi: 10.1016/j.ejor.2016.07.019.

[14] M. Klimek, "Techniques of generating schedules for the problem of financial optimization of multi-stage project," *Appl. Comput. Sci.*, vol. 15, no. 1, p. 18, 2019, doi: 10.23743/acs-2019-02.

[15] M. Asadujjaman, H. F. Rahman, R. K. Chakrabortty, and M. J. Ryan, "Multi-operator immune genetic algorithm for project scheduling with discounted cash flows," *Expert Syst. Appl.*, vol. 195, p. 116589, 2022. https://doi.org/10.1016/j.eswa.2022.116589

[16] M. Asadujjaman, H. F. Rahman, R. K. Chakrabortty, and M. J. Ryan, "An immune genetic algorithm for solving NPV-based resource constrained project scheduling problem," *IEEE Access*, vol. 9, pp. 26177–26195, 2021, doi: 10.1109/ACCESS.2021.3057366.

[17] M. Chen, S. Yan, S.-S. Wang, and C.-L. Liu, "A generalized network flow model for the multi-mode resource-constrained project scheduling problem with discounted cash flows," *Eng. Optim.*, vol. 47, no. 2, pp. 165–183, Feb. 2015. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/0305215X.2013.875167, doi: 10.1080/0305215X.2013.

[18] P. Leyman and M. Vanhoucke, "Payment models and net present value optimization for resource-constrained project scheduling," *Comput. Ind. Eng.*, vol. 91, pp. 139–153, Jan. 2016, doi: 10.1016/j.cie.2015.11.008.

[19] Z.-X. Zhang, W.-N. Chen, H. Jin, and J. Zhang, "A preference biobjective evolutionary algorithm for the payment scheduling negotiation problem," *IEEE Trans. Cybern.*, vol. 51, no. 12, pp. 6105–6118, Dec. 2021, doi: 10.1109/tcyb.2020.2966492.

[20] W. Wiesemann and D. Kuhn, "The stochastic time-constrained net present value problem," in *Handbook on Project Management and Scheduling*, vol. 2. Cham, Switzerland: Springer, 2015, pp. 753–780, doi: 10.1007/978-3-319-05915-0_5.

[21] S. Creemers, B. De Reyck, and R. Leus, "Project planning with alternative technologies in uncertain environments," *Eur. J. Oper. Res.*, vol. 242, no. 2, pp. 465–476, Apr. 2015, doi: 10.1016/j.ejor.2014.11.014.

[22] L. P. Kerkhove and M. Vanhoucke, "Optimised scheduling for weather sensitive offshore construction projects," *Omega*, vol. 66, pp. 58–78, Jan. 2017, doi: 10.1016/j.omega.2016.01.011.

[23] S. Creemers, "Maximizing the expected net present value of a project with phase-type distributed activity durations: An efficient globally optimal solution procedure," *Eur. J. Oper. Res.*, vol. 267, pp. 16–22, May 2018, doi: 10.1016/j.ejor.2017.11.027.

[24] S. Creemers, "Moments and distribution of the net present value of a serial project," *Eur. J. Oper. Res.*, vol. 267, pp. 835–848, Jun. 2018, doi: 10.1016/j.ejor.2017.12.039.

[25] S. Creemers, "Two sequencing problems: Equivalence, optimal solution, and state-of-the-art results," *SSRN Electron. J.*, 2017. [Online]. Available: https://ssrn.com/abstract=3082785https://ssrn.com/abstract=3082785, doi: 10.2139/ssrn.3082785.

[26] B. Hermans and R. Leus, "Scheduling Markovian PERT networks to maximize the net present value: New results," *Oper. Res. Lett.*, vol. 46, pp. 240–244, Mar. 2018, doi: 10.1016/j.orl.2018.01.010.

[27] 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, 2018. [Online]. Available: https://www.tandfonline.com/action/journalInformation?journalCode=tjor20, doi: 10.1057/s41274-017-0198-3.

[28] Y. Liang, N. Cui, T. Wang, and E. Demeulemeester, "Robust resource-constrained max-NPV project scheduling with stochastic activity duration," *OR Spectr.*, vol. 41, no. 1, pp. 219–254, Mar. 2019, doi: 10.1007/s00291-018-0533-3.

[29] F. Rezaei, A. A. Najafi, and R. Ramezanian, "Mean-conditional value at risk model for the stochastic project scheduling problem," *Comput. Ind. Eng.*, vol. 142, Apr. 2020, Art. no. 106356, doi: 10.1016/j.cie.2020.106356.

[30] C. Szwarcfiter, Y. T. Herer, and A. Shtub, "Project scheduling in a lean environment to maximize value and minimize overruns," *J. Scheduling*, vol. 25, no. 2, pp. 177–190, Apr. 2022, doi: 10.1007/s10951-022-00727-9.

[31] Y.-Y. Chih and O. Zwikael, "Project benefit management: A conceptual framework of target benefit formulation," *Int. J. Project Manage.*, vol. 33, no. 2, pp. 352–362, Feb. 2015, doi: 10.1016/j.ijproman.2014.06.002.

[32] A. Badewi, "The impact of project management (PM) and benefits management (BM) practices on project success: Towards developing a project benefits governance framework," *Int. J. Project Manage.*, vol. 34, no. 4, pp. 761–778, 2016. http://www.sciencedirect.com/science/article/pii/S0263786315001027, doi: 10.1016/J.IJPROMAN.2015.05.005.

[33] A. U. Musawir, C. E. M. Serra, O. Zwikael, and I. Ali, "Project governance, benefit management, and project success: Towards a framework for supporting organizational strategy implementation," *Int. J. Project Manage.*, vol. 35, no. 8, pp. 1658–1672, Nov. 2017, doi: 10.1016/j.ijproman.2017.07.007.

[34] C. E. M. Serra and M. Kunc, "Benefits realisation management and its influence on project success and on the execution of business strategies," *Int. J. Project Manage.*, vol. 33, no. 1, pp. 53–66, Jan. 2015, doi: 10.1016/j.ijproman.2014.03.011.

[35] O. Zwikael, Y.-Y. Chih, and J. R. Meredith, "Project benefit management: Setting effective target benefits," *Int. J. Project Manage.*, vol. 36, no. 4, pp. 650–658, 5 2018. https://www.sciencedirect.com/science/article/pii/S0263786317311328, doi: 10.1016/J.IJPROMAN.2018.01.002.

[36] M. Laursen, "Project networks as constellations for value creation," *Project Manage. J.*, vol. 49, no. 2, pp. 56–70, Apr. 2018, doi: 10.1177/875697281804900204.

[37] D. C. Invernizzi, G. Locatelli, M. Grönqvist, and N. J. Brookes, "Applying value management when it seems that there is no value to be managed: The case of nuclear decommissioning," *Int. J. Project Manage.*, vol. 37, no. 5, pp. 668–683, Jan. 2019. https://www.sciencedirect.com/science/article/pii/S0263786318304447, doi: 10.1016/J.IJPROMAN.2019.01.004.

[38] A. Mishra, K. K. Sinha, and S. Thirumalai, "Project quality: The Achilles heel of offshore technology projects?" *IEEE Trans. Eng. Manag.*, vol. 64, no. 3, pp. 272–286, Aug. 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7892006/, doi: 10.1109/TEM.2017.2662021.

[39] T. R. Browning, J. J. Deyst, S. D. Eppinger, and D. E. Whitney, "Adding value in product development by creating information and reducing risk," *IEEE Trans. Eng. Manag.*, vol. 49, no. 4, pp. 443–458, Nov. 2002, doi: 10.1109/.TEM.2002.806710.

[40] H. Y. Chiang and B. M. T. Lin, "A decision model for human resource allocation in project management of software development," *IEEE Access*, vol. 8, pp. 38073–38081, 2020, doi: 10.1109/ACCESS.2020.2975829.

[41] T. R. Browning, "A quantitative framework for managing project value, risk, and opportunity," *IEEE Trans. Eng. Manag.*, vol. 61, no. 4, pp. 583–598, Nov. 2014. [Online]. Available: http://ieeexplore.ieee.org/document/6840973/, doi: 10.1109/TEM.2014.2326986.

[42] H. Dinçer, S. Yüksel, and L. Martínez, "Balanced scorecard-based analysis about European energy investment policies: A hybrid hesitant fuzzy decision-making approach with quality function deployment," *Expert Syst. With Appl.*, vol. 115, pp. 152–171, Jan. 2019. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0957417418305001, doi: 10.1016/j.eswa.2018.07.072.

[43] E. C. Cordeiro, G. F. Barbosa, and L. G. Trabasso, "A customized QFD (quality function deployment) applied to management of automation projects," *Int. J. Adv. Manuf. Technol.*, vol. 87, nos. 5–8, pp. 2427–2436, Nov. 2016. [Online]. Available: http://link.springer.com/10.1007/s00170-016-8626-0

[44] A. Liu, H. Hu, X. Zhang, and D. Lei, "Novel two-phase approach for process optimization of customer collaborative design based on fuzzy-QFD and DSM," *IEEE Trans. Eng. Manag.*, vol. 64, no. 2, pp. 193–207, May 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7839280/, doi: 10.1109/TEM.2017.2651052.

[45] S. M. Lo, H.-P. Shen, and J. C. Chen, "An integrated approach to project management using the Kano model and QFD: An empirical case study," *Total Quality Manag. Bus. Excellence*, vol. 28, nos. 13–14, pp. 1–25, Nov. 2016. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/14783363.2016.1151780, doi: 10.1080/14783363.2016.1151780.

[46] I. Cohen and M. Iluz, "When cost–effective design strategies are not enough: Evidence from an experimental study on the role of redundant goals," *Omega*, vol. 56, pp. 99–111, Oct. 2015, doi: 10.1016/j.omega.2014.09.007.

[47] M. Masin, Y. Dubinsky, M. Iluz, E. Shindin, and A. Shtub, "EMI: Engineering and management integrator," in *Complex Systems Design & Management*. Cham, Switzerland: Springer, 2016, pp. 143–155. [Online]. Available: http://link.springer.com/10.1007/978-3-319-26109-6_11

[48] A. Shtub, M. Iluz, K. Gersing, J. Oehmen, and Y. Dubinsky, "Implementation of lean engineering practices in projects and programs through simulation based training," *PM World J.*, vol. 3, no. 3, pp. 1–13, 2014.

[49] I. Cohen, M. Iluz, and A. Shtub, "A simulation-based approach in support of project management training for systems engineers," *Syst. Eng.*, vol. 17, no. 1, pp. 26–36, Mar. 2014, doi: 10.1002/sys.21248.

[50] C. Szwarcfiter, A. Shtub, and Y. T. Herer, "Maximizing value–modeling and solving lean project management," in *Proc. 17th Int. Conf. Project Manag. Scheduling, Extended Abstract*, Toulouse, France, 2020, pp. 329–332. [Online]. Available: https://pms2020.sciencesconf.org/298313/document

[51] N. Balouka and I. Cohen, "A robust optimization approach for the multi-mode resource-constrained project scheduling problem," *Eur. J. Oper. Res.*, vol. 291, no. 2, pp. 457–470, 6 2021, doi: 10.1016/j.ejor.2019.09.052.

[52] P. Nansheng and M. Qichen, "Resource allocation in robust scheduling," *J. Oper. Res. Soc.*, vol. 74, no. 1, pp. 125–142, 2023, doi: 10.1080/01605682.2022.2029593.

[53] H. Dai, L. Li, R. Mao, X. Liu, and K. Zhou, "A solution-based tabu search algorithm for the resource-constrained project scheduling problem with step deterioration," *Math. Problems Eng.*, vol. 2023, May 2023, Art. no. 8353962, doi: 10.1155/2023/8353962.

[54] Y. Liu, J. Zhou, A. Lim, and Q. Hu, "A tree search heuristic for the resource constrained project scheduling problem with transfer times," *Eur. J. Oper. Res.*, vol. 304, no. 3, pp. 939–951, Feb. 2023, doi: 10.1016/j.ejor.2022.05.014.

[55] F. Zuo, E. Zio, and Y. Yuan, "Risk-response strategy optimization considering limited risk-related resource allocation and scheduling," *J. Construct. Eng. Manage.*, vol. 148, no. 11, pp. 04022123-1–04022123-14, Nov. 2022, doi: 10.1061/(asce)co.1943-7862.0002392.

[56] F. Zuo, E. Zio, and Y. Xu, "Bi-objective optimization of the scheduling of risk-related resources for risk response," *Rel. Eng. Syst. Saf.*, vol. 237, Sep. 2023, Art. no. 109391, doi: 10.1016/.j.ress.2023.109391.

[57] M. Bold and M. Goerigk, "A faster exact method for solving the robust multi-mode resource-constrained project scheduling problem," *Oper. Res. Lett.*, vol. 50, no. 5, pp. 581–587, Sep. 2022, doi: 10.1016/j.orl.2022.08.003.

[58] P. Peykani, J. Gheidar-Kheljani, S. Shahabadi, S. H. Ghodsypour, and M. Nouri, "A two-phase resource-constrained project scheduling approach for design and development of complex product systems," *Oper. Res.*, vol. 23, no. 1, p. 17, Mar. 2023, doi: 10.1007/s12351-023-00750-4.

[59] G. Calafiore and M. C. Campi, "Uncertain convex programs: Randomized solutions and confidence levels," *Math. Program.*, vol. 102, no. 1, pp. 25–46, 2005. [Online]. Available: https://link.springer.com/content/pdf/10.1007%2Fs10107-003-0499-y.pdf, doi: 10.1007/s10107-003-0499-y.

[60] W. J. Gutjahr, "Bi-objective multi-mode project scheduling under risk aversion," *Eur. J. Oper. Res.*, vol. 246, no. 2, pp. 421–434, 2015, doi: 10.1016/j.ejor.2015.05.004.

[61] P. Lamas and E. Demeulemeester, "A purely proactive scheduling procedure for the resource-constrained project scheduling problem with stochastic activity durations," *J. Scheduling*, vol. 19, no. 4, pp. 409–428, 2016. [Online]. Available: https://link.springer.com/content/pdf/10.1007%2Fs10951-015-0423-3.pdf, doi: 10.1007/s10951-015-0423-3.

[62] J. Tian, X. Hao, and M. Gen, "A hybrid multi-objective EDA for robust resource constraint project scheduling with uncertainty," *Comput. Ind. Eng.*, vol. 130, pp. 317–326, Apr. 2019, doi: 10.1016/j.cie.2019.02.039.

[63] W. Tian and E. Demeulemeester, "Railway scheduling reduces the expected project makespan over roadrunner scheduling in a multi-mode project scheduling environment," *Ann. Oper. Res.*, vol. 213, no. 1, pp. 271–291, 2014, doi: 10.1007/.s10479-012-1277-0.

[64] C. Szwarcfiter, A. Shtub, and Y. T. Herer, "Solving the stochastic multimode resource-constrained project scheduling problem," in *Proc. 17th Int. Conf. Project Manag. Scheduling, Extended Abstract*, Toulouse, France, 2020, pp. 325–328. [Online]. Available: https://pms2020.sciencesconf.org/298323/document

[65] K. Deb, "Multi-objective optimization," in *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*, G. K. Edmund and K. Burke, Eds. Boston, MA, USA: Springer, 2014, ch. 15, pp. 403–449. [Online]. Available: http://link.springer.com/10.1007/978-1-4614-6940-7_15

[66] F. Bomsdorf and U. Derigs, "A model, heuristic procedure and decision support system for solving the movie shoot scheduling problem," *OR Spectr.*, vol. 30, no. 4, pp. 751–772, Oct. 2008, doi: 10.1007/s00291-007-0103-6.

[67] R. Etgar, A. Shtub, and L. J. Leblanc, "Scheduling projects to maximize net present value - The case of time-dependent, contingent cash flows," *Eur. J. Oper. Res.*, vol. 96, no. 1, pp. 90–96, 1997, doi: 10.1016/0377-2217(95)00382-7.

[68] C. Artigues, O. Koné, P. Lopez, and M. Mongeau, "Mixed-integer linear programming formulations," in *Handbook on Project Management and Scheduling*, vol. 1. Cham, Switzerland: Springer, 2015, pp. 17–41. [Online]. Available: http://link.springer.com/10.1007/978-3-319-05443-8_2

[69] A. G. Barto, "Reinforcement learning: Connections, surprises, challenges," *AI Mag.*, vol. 40, no. 1, pp. 3–15, 2019. [Online]. Available: https://search.proquest.com/docview/2213790487/fulltextPDF/4D2E04E3AFCF4428PQ/1?accountid=27233

[70] R. Polvara, S. Sharma, J. Wan, A. Manning, and R. Sutton, "Autonomous vehicular landings on the deck of an unmanned surface vehicle using deep reinforcement learning," *Robotica*, vol. 37, pp. 1–16, Apr. 2019. [Online]. Available: https://www.cambridge.org/core/product/identifier/S0263574719000316/type/journal_article, doi: 10.1017/S0263574719000316.

[71] D. Ferrucci, A. Levas, S. Bagchi, D. Gondek, and E. T. Mueller, "Watson: Beyond jeopardy!" *Artif. Intell.*, vols. 199–200, pp. 93–105, Jun. 2013. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0004370212000872, doi: 10.1016/j.artint.2012.06.009.

[72] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015. [Online]. Available: http://www.nature.com/articles/nature14236, doi: 10.1038/nature14236.

[73] P. Jędrzejowicz and E. Ratajczak-Ropel, "Reinforcement learning strategy for solving the MRCPSP by a team of agents," in *Intelligent Decision Technologies*, R. Neves-Silva, L. Jain, and R. Howlett, Eds. Cham, Switzerland: Springer, 2015, pp. 537–548. [Online]. Available: http://link.springer.com/10.1007/978-3-319-19857-6_46, doi: 10.1007/978-3-319-19857-6_46.

[74] T. Wauters, K. Verbeeck, P. De Causmaecker, and G. V. Berghe, "A learning-based optimization approach to multi-project scheduling," *J. Scheduling*, vol. 18, no. 1, pp. 61–74, Feb. 2015. [Online]. Available: http://link.springer.com/10.1007/s10951-014-0401-1, doi: 10.1007/s10951-014-0401-1.

[75] K. M. Sallam, R. K. Chakrabortty, and M. J. Ryan, "A reinforcement learning based multi-method approach for stochastic resource constrained project scheduling problems," *Expert Syst. Appl.*, vol. 169, May 2021, Art. no. 114479, doi: 10.1016/j.eswa.2020.114479.

[76] W. Peng, X. Lin, and H. Li, "Critical chain based proactive-reactive scheduling for resource-constrained project scheduling under uncertainty," *Expert Syst. Appl.*, vol. 214, Mar. 2023, Art. no. 119188, doi: 10.1016/j.eswa.2022.119188.

[77] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, MA, USA: MIT Press, 2018.

[78] H. Dong, Z. Ding, and S. Zhang, *Deep Reinforcement Learning: Fundamentals, Research and Applications*. Singapore: Springer, 2020.

[79] J. Zheng, K. He, J. Zhou, Y. Jin, and C. M. Li, "Combining reinforcement learning with Lin-Kernighan-Helsgaun algorithm for the traveling salesman problem," in *Proc. 35th AAAI Conf. Artif. Intell.*, vol. 14A, 2021, pp. 12445–12452, doi: 10.1609/aaai.v35i14.17476.

[80] J. Subramanian and A. Mahajan, "Renewal Monte Carlo: Renewal theory-based reinforcement learning," *IEEE Trans. Autom. Control*, vol. 65, no. 8, pp. 3663–3670, Nov. 2020, doi: 10.1109/.TAC.2019.2953089.

[81] W. Liu, T. Tang, S. Su, Y. Cao, F. Bao, and J. Gao, "An intelligent train control approach based on the Monte Carlo reinforcement learning algorithm," in *Proc. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 1944–1949, doi: 10.1109/ITSC.2018.8569399.

[82] W. A. Suttle, A. S. Bedi, B. Patel, B. M. Sadler, A. Koppel, and D. Manocha, "Beyond exponentially fast mixing in average-reward reinforcement learning via multi-level Monte Carlo actor-critic," in *Proc. 40th Int. Conf. Mach. Learn.*, Honolulu, HI, USA, 2023, pp. 33240–33267, doi: 10.1016/j.automatica.2021.109693.

[83] J. Liu, "On the convergence of reinforcement learning with Monte Carlo exploring starts," *Automatica*, vol. 129, Jul. 2021, Art. no. 109693, doi: 10.1016/j.automatica.2021.109693.

[84] A. Winnicki and R. Srikant. (Apr. 2023). *On The Convergence Of Policy Iteration-Based Reinforcement Learning With Monte Carlo Policy Evaluation*. [Online]. Available: https://proceedings.mlr.press/v206/winnicki23a.html

[85] C. Wang, S. Yuan, K. Shao, and K. Ross, "On the convergence of the Monte Carlo exploring starts algorithm for reinforcement learning," in *Proc. 10th Int. Conf. Learn. Represent. (ICLR)*, 2022, pp. 1–33.

[86] V. T. Aghaei, A. Ağababaoğlu, B. Bawo, P. Naseradinmousavi, S. Yıldırım, S. Yeşilyurt, and A. Onat, "Energy optimization of wind turbines via a neural control policy based on reinforcement learning Markov chain Monte Carlo algorithm," *Appl. Energy*, vol. 341, Jul. 2023, Art. no. 121108, doi: 10.1016/j.apenergy.2023.121108.

[87] M. I. Skolnik, *Radar Handbook*. New York, NY, USA: McGraw-Hill, 1970. [Online]. Available: https://trid.trb.org/view/49654

[88] R. K. Sarin, "Multi-attribute utility theory," in *Encyclopedia of Operations Research and Management Science*. Boston, MA, USA: Springer, 2013, pp. 1004–1006. [Online]. Available: http://link.springer.com/10.1007/978-1-4419-1153-7_644

[89] M. Ning, Z. He, T. Jia, and N. Wang, "Metaheuristics for multi-mode cash flow balanced project scheduling with stochastic duration of activities," *Autom. Construct.*, vol. 81, pp. 224–233, Sep. 2017, doi: 10.1016/j.autcon.2017.06.011.

[90] T. Servranckx and M. Vanhoucke, "A tabu search procedure for the resource-constrained project scheduling problem with alternative subgraphs," *Eur. J. Oper. Res.*, vol. 273, no. 3, pp. 841–860, Mar. 2019. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0377221718307513, doi: 10.1016/j.ejor.2018.09.005.

[91] M. Gendreau and J.-Y. Potvin, Eds., *Handbook of Metaheuristics*, 3rd ed. Cham, Switzerland: Springer, 2019. [Online]. Available: http://www.springer.com/series/6161

[92] M. Mika, G. Waligóra, and J. Weglarz, "Simulated annealing and tabu search for multi-mode resource-constrained project scheduling with positive discounted cash flows and different payment models," *Eur. J. Oper. Res.*, vol. 164, no. 3, pp. 639–668, Aug. 2005, doi: 10.1016/J.EJOR.2013.10.012.

[93] R. Kolisch and A. Sprecher, "PSPLIB—A project scheduling problem library: OR software—ORSEP operations research software exchange program," *Eur. J. Oper. Res.*, vol. 96, no. 1, pp. 205–216, 1997, doi: 10.1016/j.ejor.2018.09.005.

[94] V. Van Peteghem and M. Vanhoucke, "An experimental investigation of metaheuristics for the multi-mode resource-constrained project scheduling problem on new dataset instances," *Eur. J. Oper. Res.*, vol. 235, no. 1, pp. 62–72, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0377221713008357, doi: 10.1016/J.EJOR.2013.10.012.

[95] M. Vanhoucke and J. Coelho, "A tool to test and validate algorithms for the resource-constrained project scheduling problem," *Comput. Ind. Eng.*, vol. 118, pp. 251–265, Apr. 2018. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0360835218300378, doi: 10.1016/j.cie.2018.02.001.

[96] M. Vanhoucke, *Integrated Project Management Sourcebook*. Cham, Switzerland: Springer, 2016.

[97] S. Van De Vonder, E. Demeulemeester, W. Herroelen, and R. Leus, "The trade-off between stability and makespan in resource-constrained project scheduling," *Int. J. Prod. Res.*, vol. 44, no. 2, pp. 215–236, 2006, doi: 10.1080/00207540500140914.

[98] Z. Ma, E. Demeulemeester, Z. He, and N. Wang, "A computational experiment to explore better robustness measures for project scheduling under two types of uncertain environments," *Comput. Ind. Eng.*, vol. 131, pp. 382–390, May 2019, doi: 10.1016/j.cie.2019.04.014.

[99] M. Iluz, B. Moser, and A. Shtub, "Shared awareness among project team members through role-based simulation during planning—A comparative study," *Proc. Comput. Sci.*, vol. 44, pp. 295–304, Jan. 2015. [Online]. Available: http://www.sciencedirect.com and http://linkinghub.elsevier.com/retrieve/pii/S1877050915002793, doi: 10.1016/j.procs.2015.03.043.

[100] *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*, 6th ed. Newtown Square, PA, USA: Project Management Institute, 2017.

[101] H. H. Hoos, "Automated algorithm configuration and parameter tuning," in *Autonomous Search*, 1st ed., Y. Hamadi, E. Monfroy, and F. Saubion, Eds. Berlin, Germany: Springer, 2012, pp. 37–71, doi: 10.1007/978-3-642-21434-9.

[102] C. Szwarcfiter and Y. Herer, "Modeling and solving the tradeoff between project value and net present value," Mendeley Data, V3, 2022, doi: 10.17632/4fvv36zwsj.3.

[103] R. Kolisch, "Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation," *Eur. J. Oper. Res.*, vol. 90, no. 2, pp. 320–333, Apr. 1996. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/0377221795003576, doi: 10.1016/0377-2217(95)00357-6.

CLAUDIO SZWARCFITER received the B.S. degree in chemical engineering from the Federal University of Rio de Janeiro, Brazil, in 1991, the M.S. degree in industrial engineering from the Pontifical Catholic University of Rio de Janeiro, Brazil, in 1995, and the Ph.D. degree in industrial engineering and management from the Technion—Israel Institute of Technology, in 2021.

From 2020 to 2022, he conducted postdoctoral research with Tel Aviv University, on sustainable logistics and the physical internet. His current research interests include project management and scheduling, sustainable logistics, reinforcement learning and machine learning applications, and simulation. He believes that reinforcement learning-based techniques can be leveraged to build better project plans, balancing time, cost, risk, and benefit.

Dr. Szwarcfiter received the 2021 Project Management Institute (PMI) James R. Snyder International Student Paper of the Year Award; the Second Prize for an Outstanding Paper at the IE & M 2021: 22nd National Industrial Engineering and Management Conference, Israel; and the finalist of the Best Student Paper Award at the PMS 2021: 17th International Conference on Project Management and Scheduling, Toulouse. In 2020, he was awarded the Nahmani Prize at the Faculty of Industrial Engineering and Management, Technion, Israel.

YALE T. HERER received the B.S. degree in 1986, and the M.S. and Ph.D. degrees from the Department of Operations Research and Industrial Engineering, Cornell University, in 1990.

He is currently a Professor with the Faculty of Data and Decision Sciences, Technion—Israel Institute of Technology, where he has been the Vice Dean of the Programs of Study, since 2018. He was with several industrial concerns, both as a Consultant and as an Advisor to project groups. His research interest includes covering production planning and control. More recently, he has focused his research on the area of supply chain management, especially when integrated with transshipments or other responsive operational activities.

Prof. Herer has won various prizes, including the 1996 IIE Transactions Best Paper Award, the 2002 Mitchner Award in Quality Sciences and Quality Management, the 2008 IBM Faculty Award, and the INFORM's 2013 Daniel H. Wagner Prize for Excellence in Operations Research Practice. He has successfully planned and executed four conferences, including the 2010 annual conference for the Manufacturing and Service Operations Management Society (MSOM). He serves as an Associate Editor for *Naval Research Logistics* and an Editorial Staff for *IIE Transactions* and *Operations Research Letters*.