



Stochastic resource-constrained project scheduling problem with time varying weather conditions and an improved estimation of distribution algorithm

Yifan Zhou^{a,*}, Jindan Miao^a, Bin Yan^b, Zhisheng Zhang^a

^a School of Mechanical Engineering, Southeast University, Nanjing 211189, PR China

^b Beijing Goldwind Science & Creation Windpower Equipment Co., Ltd, Beijing 100176, PR China

ARTICLE INFO

Keywords:

SRCPSP
Estimation of distribution algorithm
Ranking and selection
Common random numbers

ABSTRACT

Construction projects with outdoor operations are affected by time-varying weather conditions. However, most existing research on stochastic resource-constrained project scheduling problems (SRCPSPs) considers activity duration as a random variable from a time-independent distribution. To address this issue, this study investigates SRCPSP under time-varying weather conditions; an improved estimation of distribution algorithm (EDA) including a ranking and selection method using common random numbers is proposed for enhancing the performance of project scheduling. The benchmark J120 dataset from PSPLIB and a practical case of windfarm construction are used to validate the improved EDA. For three randomly selected cases from the J120 dataset, the improved EDA can reduce the expected makespan by 17.0, 29.4, and 12.5 days when compared with deterministic scheduling. The corresponding makespan reductions obtained by the original EDA are 10.8, 22.7, and 7.1 days. Similarly, the improved EDA obtains 23% higher expected makespan reduction for the practical case.

1. Introduction

Weather conditions are one of the most important factors affecting outdoor activities; for example, most activities in a windfarm construction project cannot be executed during rainy and windy weather. Therefore, it is practically significant to consider uncertain weather conditions for project scheduling. Weather conditions change with the seasons, and therefore, the durations of activities executed outdoors are time-dependent random variables that have rarely been considered in existing research. To the best of our knowledge, Huang and Ding (2011) considered activity durations to be both random and time dependent. They scheduled a software project without including resource constraints. To bridge this research gap, this study investigates an SRCPSP under time-varying weather conditions using an improved EDA.

According to Herroelen and Leus (2005), three approaches can address uncertainties in the SRCPSP: proactive scheduling, reactive scheduling, and stochastic scheduling. Among these, proactive scheduling aims to obtain a baseline schedule that can absorb variability during project execution; this variability includes the uncertainty of activity durations and resources. Lamas and Demeulemeester (2016) generated a proactive scheduling procedure using a branch-and-cut

method wherein new robustness measures were proposed. Bruni, Beraldi, and Guerriero (2015) developed a chance-constrained-based heuristic algorithm to derive the baseline schedule of an SRCPSP and illustrated a practical case study of student apartment construction. Reactive scheduling is used to repair or reformulate the baseline schedule when the initial schedule is infeasible (Deblaere, Demeulemeester, & Herroelen, 2011). Proactive and reactive scheduling methods are often used together through a combined cost function for the baseline schedule and reactions during execution (Davari & Demeulemeester, 2019). The third approach is stochastic scheduling, where the duration of an activity is regarded as a random variable. The result of stochastic scheduling is no longer a definite plan; instead it is a scheduling policy that decides how to schedule activities dynamically in each possible scenario. Stochastic scheduling was adopted in this research.

Stochastic scheduling can be divided into exact evaluation and simulation-based methods according to the method employed for evaluating the objective function (Rostami, Creemers, & Leus, 2018). The exact evaluation method is largely based on Markov analysis. Creemers (2015) developed backward stochastic dynamic programming to solve SRCPSP with exponential and phase-type distributed activity durations; Markov analysis was used to evaluate the objective function. Owing to

* Corresponding author.

E-mail address: yifan.zhou@seu.edu.cn (Y. Zhou).

<https://doi.org/10.1016/j.cie.2021.107322>

Received 18 February 2020; Received in revised form 9 April 2021; Accepted 12 April 2021

Available online 15 April 2021

0360-8352/© 2021 Elsevier Ltd. All rights reserved.

Nomenclature			
Indices			
g	Index of a generation in the improved EDA	N_{max}	Largest number of scenarios to evaluate an individual during the EDA
i	Index of an individual in the improved EDA	P_{per}	Probability to alter the execution order of an activity during local search
j	Index of an activity	Q	Number of elite individuals
k	Index of the type of renewable resources	$r_{j,k}$	Demand of resource k by activity j
n	Index of a scenario during simulation	R_k	Total amount of resource k
Parameters		S_j	Start time of activity j
C_{end}	Reward for completing a project	T	Planning horizon
C_j^f	Fixed cost of activity j	$u_{y,t}$	Binary indicator of workable weather condition at time t in year y
C_j^v	Variable cost of activity j	$w(t)$	Workable probability of day t
d_j^s	Standard duration of activity j	η	Proportion of elite individuals in the whole population
$d_{j,t}$	Duration of activity j starting at time t	$\Phi_{j,t}$	Probability distribution of the duration of activity j starting from t
$d_{j,t}^E$	Expected duration of activity j starting at time t	$\Phi_{j,t}^{-1}(\cdot)$	Inverse cumulative distribution function of distribution $\Phi_{j,t}$
$d_{j,y,t}^H$	Historical duration of activity j starting at time t in year y	A_t	Set of activities that are in execution at time t
$d_{j,t,n}$	Duration of activity j starting at time t in scenario n	P_j	Set of precedence activities of activity j
D_n	Matrix of activity durations in scenario n	ψ	Set of activities which have been scheduled
E_j	End time of activity j	χ	Eligible set containing unscheduled activities whose predecessors have all been scheduled
IR	Discount rate per unit time in the NPV	Ω	Set of individuals in the EDA
J	Number of non-dummy activities in a project	Decision variables	
K	Number of types of renewable resource	$\lambda = [a_0, a_1, \dots, a_{J+1}]$	Execution order of activities
L_{rate}	Learning rate of the probability model in the improved EDA	$a_m \in \{0, 1, \dots, J+1\}$	Index of m^{th} activity in vector λ
M	Population size of the improved EDA	τ	Start time of a project
n_w	Sample size of the start date in local search	$\pi = [\lambda, \tau]$	Solution of SRCPSP
N	Number of scenarios used to estimate the expected objective function		

the memory bottlenecks, the approach proposed by Creemers (2015) was effective for small and medium problems (up to 62 activities). Creemers, Leus, and Lambrecht (2010) solved an SRCPSP to maximise the net present value (NPV) by adopting Markov analysis. Rostami et al. (2018) proposed an exact evaluation method based on Markov analysis to solve SRCPSP with exponentially distributed activity durations. The above-mentioned literature indicates that the exact evaluation method is appropriate for only small and medium SRCPSPs.

A simulation-based method is effective for processing large-scale problems. Ballestín and Leus (2009) developed a greedy randomised adaptive search procedure to solve the SRCPSP. The numerical study showed that generating 10 scenarios for each solution could help identify considerably better results than applying a larger number of scenarios; this is consistent with the conclusion of Ballestín (2007). Rostami et al. (2018) developed a two-phase metaheuristic algorithm that includes a greedy randomised adaptive search procedure and a genetic algorithm (GA) to search for a high-quality scheduling policy. Further, they adopted a descriptive sampling method to reduce the variance in activity durations in different scenarios.

The EDA is a commonly used heuristic approach in project scheduling and related areas (Jarbouli, Eddaly, & Siarry, 2009; Shi et al., 2019; Tian, Hao, & Gen, 2019). Unlike the classical GA, the EDA builds an explicit distribution model to select individuals during recombination, and this results in a stronger global searching ability. Wang and Fang (2012) adopted the EDA to solve a multimode project scheduling problem with the aim of minimising the makespan. The effectiveness of the EDA was investigated using benchmark cases. Hao, Sun, and Gen (2018) developed a new EDA hybridised with the GA for robust project scheduling. A scenario-based simulation was used to measure the robustness. Fang, Kolisch, Wang, and Mu (2015) applied the EDA to solve SRCPSP, and the Taguchi experimental design method was adopted to identify the best combination of parameters in the EDA. In

addition, the EDA was used and improved in this study to solve SRCPSP.

Most existing studies treated the search of the optimal solution and the evaluation of objective functions as two independent processes when solving SRCPSP via simulation (Bruni, Beraldi, Guerriero, & Pinto, 2011; Chen, Demeulemeester, Bai, & Guo, 2018; Tao, Wu, Sheng, & Wang, 2018). In these studies, different solutions received the same simulation efforts, and unnecessary replications were generated for solutions with low quality. Chen (2016a) found that generating the realisations of objective functions is the most time-consuming part during optimisation via simulation. Therefore, the optimisation efficiency can be enhanced by appropriately allocating the simulation efforts to different solutions. Choi and Kim (2018) developed a robust and efficient R&S algorithm by allocating limited simulation resources. Gao, Shi, Gao, and Xiao (2019) investigated the problem of selecting the optimal subset, and a simulation budget allocation rule was proposed using the large deviation theory. The common random number (CRN) is a simulation technique for variance reduction, which compares different individuals under similar circumstances. Individuals are evaluated using positively correlated inputs (Chen, 2016b). Kesur (2009) optimised traffic signals using GA where the CRN was used to reduce the variance in the simulation. Hedlund and Mollaghasemi (2001) applied the CRN method to GA. A two-step indifference-zone R&S method was employed to determine the number of replications simulated for each solution. A recent study by Nazzal, Mollaghasemi, Hedlund, and Bozorgi (2012) provided a more comprehensive introduction of the methodology of Hedlund and Mollaghasemi (2001).

References that combined heuristic algorithms and simulation-based methods are summarised in Table 1. These references were selected from the project scheduling and related areas. Table 1 shows that the R&S and CRN methods have not been used for project scheduling. Studies that embedded both R&S and CRN into heuristic algorithms are also limited. The two most related papers (Hedlund & Mollaghasemi, 2001; Nazzal

Table 1

References that combine heuristic algorithms with simulation-based methods.

References	Heuristic algorithm	R&S	CRN	Area
Chen et al. (2018)	EDA & GA			Project scheduling
Tao et al. (2018)	Artificial algae algorithm			Project scheduling
Tian et al. (2019)	EDA & GA			Project scheduling
Hao et al. (2018)	EDA & GA			Project scheduling
Fang et al. (2015)	EDA			Project scheduling
Alrabghi and Tiwari (2016)	Simulated annealing			Maintenance optimisation
Kennedy, Walsh, Mastaglio, and Scully (2016)	GA			Maintenance optimisation
Salgado Duarte, Szytko, and del Castillo Serpa (2020)	Particle swarm optimisation & GA			Maintenance optimisation
Buchholz and Thummmler (2005)	GA	✓		Not specified
Kesur (2009)	GA		✓	Vehicular traffic
Hedlund and Mollaghasemi (2001)	GA	✓	✓	Not specified
Nazzal et al. (2012)	GA	✓	✓	Design of production system
This paper	EDA	✓	✓	Project scheduling

et al., 2012) adopted both the R&S and CRN in GA. The two studies used the R&S method to identify the number of replications for a solution that guarantees the probability of correct selection. However, the same number of replications were generated for different solutions in an iteration of the GA, and the unnecessary simulation of inferior solutions was not avoided.

According to the above literature review, the combination of the R&S and CRN methods with heuristic algorithms has not received the necessary attention, especially in project scheduling. Thus, this study develops an improved EDA to solve SRCPSPs where an R&S procedure based on the CRN is utilised to select elite solutions. Through this procedure, promising solutions receive more simulation effort than other low-quality solutions. In the investigated SRCPSP, activity durations are time-dependent random variables, and both resource and precedence constraints are considered. The start date of a project is adopted as an additional decision variable during optimisation because the start date affects the makespan of the project when the distributions of activity durations are time-dependent. Numerical examples modified from benchmark instances in the J120 dataset of the PSPLIB are studied to validate the performance of the improved EDA. A practical case study of windfarm construction is analysed, and the results show that the improved EDA can effectively solve the SRCPSP under time-varying weather conditions.

The major contributions of this paper can be summarised as follows: (1) an R&S method using the CRN is combined into the EDA; (2) Time-dependent and random activity durations are considered in SRCPSP; and (3) a practical case about the construction of a windfarm is analysed.

The rest of this manuscript is organised as follows. Section 2 introduces the formulation and assumptions of the time-dependent SRCPSP. Section 3 proposes an optimisation model of time-dependent SRCPSP. The improved EDA including an R&S procedure is developed in Section 4. Section 5 evaluates the performance of the improved EDA

using numerical examples modified from benchmark instances in the J120 dataset of PSPLIB. Finally, a practical case study of windfarm construction is presented in Section 6.

2. Problem formulation and assumptions

The SRCPSP can be regarded as scheduling a set of activities under precedence and resource constraints considering uncertainties. In this study, only the uncertainty of activity durations induced by weather conditions is considered, whereas other parameters (availability and amount of resources, precedence constraints, and execution mode of activities) are assumed to be determinate. There are $J+2$ activities in the project, i.e. $\{0, 1, \dots, J, J+1\}$. Activities 0 and $J+1$ are dummy activities that represent the start and end of the project, respectively. The duration of the two dummy activities is zero. The precedence constraint between activities j and j' is modelled by an indicator function $I(j', j)$. Activity j' is a predecessor of activity j if $I(j', j) = 1$. Then, $P_j = \{j' | I(j', j) = 1\}$ is the set of precedence activities of activity j . For simplifying the scheduling problem, it is assumed that the execution of an activity cannot be interrupted by another activity. There are K types of renewable resources, and R_k units of resource k are available throughout the project. Activity j needs $r_{j,k}$ ($0 \leq r_{j,k} \leq R_k$) units of resource k . The two dummy activities do not consume any resource, i.e. $r_{0,k} = r_{J+1,k} = 0$ for $\forall k \in \{1, 2, \dots, K\}$. Unlike the deterministic problem, it is impossible to identify the exact start time S_j and end time E_j of activity j caused by the influence of uncertain weather conditions. Therefore, the scheduling result is modelled as a scheduling policy $\pi = [\lambda, \tau]$. Vector $\lambda = [a_0, a_1, \dots, a_{J+1}]$ represents the execution order of activities, where $a_m \in \{0, 1, \dots, J+1\}$ denotes the index of the m^{th} activity. The first and last activities are two dummy activities, i.e. $a_0 = 0$ and $a_{J+1} = J+1$. The variable τ represents the start time of the project. The objective of scheduling is to identify a policy π^* that incurs the shortest expected makespan or the largest expected NPV.

The activity durations in this study are influenced by time-varying weather conditions. Therefore, the distribution of the duration of activity j depends on the start time of the activity given by

$$d_{j,t} \sim \Phi_{j,t}, \forall j = 1, 2, \dots, J; t = 1, 2, \dots, T. \quad (1)$$

Here, $d_{j,t}$ refers to the duration of activity j that starts from t , and $\Phi_{j,t}$ is the corresponding probability distribution. Constant T is the planning horizon.

Algorithm 1 Calculating durations of activity j starting at time t in different years

```

1: Initialise  $D_{j,t} = \emptyset$ 
2: for  $y = 1$  to  $Y$ 
3:   Initialise the accumulative effective working time:  $count = 0$ , the duration of activity  $j$  starting at day  $t$  in year  $y$ :  $d_{j,y,t}^H = 0$ , and the time index  $t' = t$ 
4:   while  $count < d_j^S$  ( $d_j^S$  is the standard duration of activity  $j$ )
5:      $count = count + u_{y,t'}$ ,  $d_{j,y,t}^H = d_{j,y,t}^H + 1$ , and  $t' = t' + 1$ 
6:   end while
7:    $D_{j,t} = D_{j,t} \cup d_{j,y,t}^H$ 
8: end for

```

Probability distribution $\Phi_{j,t}$ is fitted using the historical durations of activity j starting at time t in different years. The detailed steps for calculating the historical durations of activity j are listed in Algorithm 1. The daily historical weather data for Y years are available. In Algorithm 1, $u_{y,t}$ denotes a binary indicator of workable weather conditions. When the weather of day t in year y is suitable for work, $u_{y,t} = 1$; otherwise, $u_{y,t} = 0$. Using the binary variables $u_{y,t}$, the probability that an activity

can be executed on day t is calculated as

$$w(t) = \left(\sum_{y=1}^Y u_{y,t} \right) / Y, \forall t = 1, 2, \dots, 365. \quad (2)$$

After the set of historical durations $D_{j,t} = \{d_{j,y,t}^H | y = 1, \dots, Y\}$ is calculated using Algorithm 1, distribution $\Phi_{j,t}$ can be fitted to durations in $D_{j,t}$.

3. Optimisation model for time dependent SRCPSP

According to the formulation and assumptions of SRCPSP, decision variables of the scheduling problem include a_1, a_2, \dots, a_J , and τ . The start time of a project affects the makespan when the distribution of activity durations is time-dependent. Therefore, the start time τ of a project is a decision variable as well, and this is different from research that considers the time-independent distributions of activity durations.

Two objectives are considered in SRCPSP. The first objective is to minimise the expected makespan of the project given by $E(E_{J+1}(\pi)) - \tau$. The other objective is to maximise the expected NPV, which is the difference between the present values of cash inflows (incomes) and cash outflows (expenditures). The money earned in the future is less than that at present because of inflation and the benefits from investments. Consequently, all cash flows are discounted to the start time of the project to calculate the NPV. Inspired by Kerkhove and Vanhoucke (2017), the NPV of a project is calculated as

$$NPV = \frac{C_{end}}{(1+IR)^{E_{J+1}}} - \sum_{j=1}^J \left[\frac{C_j^f}{(1+IR)^{S_j}} + \frac{C_j^v(E_j - S_j)}{(1+IR)^{E_j}} \right] \quad (3)$$

Here, C_{end} is the reward for completing the project, and this is acquired at the end of the project; IR is the discount rate; and C_j^f and C_j^v are the fixed and variable costs of activity j , respectively. In a practical case study on windfarm construction, the expected makespan and expected NPV are both adopted as objective functions. In the numerical studies of instances modified from the J120 dataset, only the expected makespan is used as the objective function.

The mathematical model of the SRCPSP is formulated as

$$\min E(E_{J+1}(\pi)) - \tau \quad (4)$$

$$\max E(NPV(\pi)) \quad (5)$$

Subject to

$$S_j \geq \max_{j \in P_j} E_j, \forall j \in \{1, 2, \dots, J, J+1\} \quad (6)$$

$$\sum_{j \in A_k} r_{jk} \leq R_k, \forall k \in \{1, 2, \dots, K\}, \forall t \in [1, T] \quad (7)$$

$$S_j \geq \max_{j \in G_j} S_j, \forall j \in \{1, 2, \dots, J, J+1\} \quad (8)$$

Inequality (6) represents the precedence constraints among activities, where the start time of activity j cannot be earlier than the end times of its predecessors. Inequality (7) is a constraint on the resource, in which

$$A_t = \{j | S_j \leq t \leq E_j, j \in \{1, 2, \dots, J\}\} \quad (9)$$

is the set of activities in execution at time t . Indeed, the amount of resource k occupied by the activities in execution cannot be larger than R_k at each time point. Inequality (8) is the constraint on the execution

order of the activities. Set G_j contains activities whose execution order is before activity j in λ .

The expected makespan and NPV cannot be calculated analytically because of the random activity durations, resource constraints, and precedence constraints. Therefore, this study uses the Monte Carlo method to estimate the two objective functions. Here, the evaluation of the expected NPV is demonstrated as an example, and the expected makespan can be estimated in the same way. A realisation of activity durations \mathbf{D}_n is generated according to distributions $\Phi_{j,t}$. Here, \mathbf{D}_n represents a matrix, and element $(\mathbf{D}_n)_{j,t}$ represents the realisation of the duration of activity j starting from t . The subscript n represents the index of realisation in the simulation. The realisation of the activity duration \mathbf{D}_n is called a scenario. The $NPV(\pi, \mathbf{D}_n)$ denotes the NPV given by policy π and scenario \mathbf{D}_n . The expected NPV under policy π can be estimated using N scenarios as

$$E(NPV(\pi)) = \frac{1}{N} \sum_{n=1}^N NPV(\pi, \mathbf{D}_n). \quad (10)$$

The expected makespan can be estimated similarly as $E(E_{J+1}(\pi)) - \tau$. Unfortunately, the calculation of $NPV(\pi, \mathbf{D}_n)$ is not straightforward. Policy π only contains the execution order of activities and the start time of the first dummy activity; the start and end times of other activities are not available. This research uses the schedule generation scheme (SGS) to calculate the start and end times of activities under policy π and scenario \mathbf{D}_n . The SGS selects activities to execute at each decision point and obtains the start time of each activity based on the constraints (6)–(8). Two commonly used SGS methods include the parallel (PSGS) and serial (SSGS) SGS (Kolisch, 1996). According to Kolisch (1996), neither SSGS nor PSGS is dominant. The ranking of the two methods is case dependent. The PSGS has better results when the problem is highly resource constrained. The SSGS is more appropriate for problems with moderate resource constraints. This paper follows the choice of several papers on SRCPSP where the SSGS was used (Ballestín, 2007; Zhao, Ke, & Chen, 2016).

Algorithm 2 The SSGS to identify the start and end times of activities

Input: Policy $\pi = [a_0, a_1, \dots, a_{J+1}, \tau]$ and Scenario \mathbf{D}_n

- 1: Initialise the remaining resources at time t : $R_{k,t} = R_k \forall t = 1, 2, \dots, T$
 - 2: Initialise the start and end times of the first dummy activity $S_0 = E_0 = \tau$ and the set of scheduled activities $\psi = \{0\}$
 - 3: **For** $m = 1$ **to** $J + 1$
 - 4: Next activity to schedule is $j = a_m$
 - 5: Latest end time of the predecessors of activity j : $t^E = \max\{E_j | j' \in P_j\}$
 - 6: Latest start time of activities that are executed before activity j : $t^S = \max\{S_{j'} | m' < m\}$
 - 7: Earliest start time of activity j is $S' = \max\{t^E, t^S\}$
 - 8: **While** $R_{k,t} < r_{jk}, \forall t \in [S', S' + (\mathbf{D}_n)_{j,S'}]$
 - 9: $S' = \min\{E_j | E_j > S', j' \in \psi\}$
 - 10: **end while**
 - 11: $S_j = S', E_j = S_j + (\mathbf{D}_n)_{j,S_j}$ and $\psi = \psi \cup \{a_m\}$
 - 12: Update the remaining resources as $R_{k,t} = R_{k,t} - r_{jk}$ for $t \in [S_j, E_j]$
 - 13: **end**
 - 14: **Return** $\{S_0, S_1, S_2, \dots, S_J, S_{J+1}\}$ and $\{E_0, E_1, E_2, \dots, E_J, E_{J+1}\}$
-

The detailed steps of the SSGS are provided in Algorithm 2, which sequentially chooses activities in vector λ . The start time of activity j is postponed until constraints (6)–(8) are satisfied. After all activities are scheduled, the start and end times of activities under policy π and scenario \mathbf{D}_n are obtained. The corresponding NPV can then be calculated using Eq. (3).

Although only the uncertainty caused by the weather condition is considered in this study, other types of uncertainties can be processed after they are modelled in the simulation process. For example, when the uncertainty caused by recourses is considered, the repair time and time between the breakdown of a resource can be assumed to follow the exponential distribution (Fu, Lau, & Varakantham, 2015). Uncertain logistic delays can be considered using a random lead time that follows an appropriate distribution (Xu, Zhao, & Chen, 2012). Further, the budget can be modelled by a random variable, and a constraint on the budget is added to the project scheduling model (Yang, 2005). Given space limitations, the consideration of other uncertainties is regarded as a future research direction.

4. Improved EDA with an R&S procedure

An improved EDA is shown in Algorithm 3; this EDA can achieve better performance when solving the SRCPSP. An R&S procedure using the CRN is embedded into the EDA to identify elite individuals more efficiently and accurately. The sampling probability model for the next generation is modified from that of Fang et al. (2015). Compared with Fang et al. (2015), the start time of a project is adopted as an additional decision variable, and it is included in the sampling probability model. Furthermore, the method to sample the new population in Fang et al. (2015) is improved by considering resource constraints and activity durations, which are provided in Algorithm 5. In addition, the local search method proposed by Fang et al. (2015) is replaced by a more general version for a broader search scope. The flowchart of the improved EDA in Algorithm 3 is shown in Fig. 1; it explains the relationship between the algorithms and expressions in this study. Algorithm 1 is used to prepare data for the probability distributions of activity durations that are the inputs of the improved EDA. Algorithm 2 is used to calculate the objective function, and it is involved in different steps in the improved EDA. Algorithms 1 and 2 are not shown in Fig. 1.

Algorithm 3: Framework of the improved EDA

Step	Generating initial population: Generate M initial individuals $\Omega_0 : \{\pi_{0,1}, \pi_{0,2}, \dots, \pi_{0,M}\}$ as indicated in Section 4.1. Initialise the probability matrix Φ_0 through Eq. (19). Set the optimal solution $\pi^* = \pi_{0,1}$ and the index of iteration $g = 0$.
Step	Selecting elite individuals: Select Q elite individuals from population Ω_g using the R&S procedure in Algorithm 4. The selected elite set is denoted as $\Omega_g^H : \{\pi_{g,1}^H, \pi_{g,2}^H, \dots, \pi_{g,Q}^H\}$
Step	Improving elite individuals by local search: Generate new individuals from the elite set Ω_g^H using the local search method suggested in Section 4.3. The set of Q new elite individuals is then obtained as $\Omega_g^E : \{\pi_{g,1}^E, \pi_{g,2}^E, \dots, \pi_{g,Q}^E\}$. The solution with the smallest makespan or largest NPV is denoted as π_g^{E*} . If π_g^{E*} surpasses π^* , π_g^{E*} is selected as the new optimal solution π^* .
Step	Updating individual sampling probabilities: Update the probability matrix Φ_{g+1} using the elite set Ω_g^E through Eq. (20).
Step	Sampling new population: Sample M individuals according to Φ_{g+1} through Algorithm 5 to form a new population Ω_{g+1} .
Step	Stopping criterion: If the stopping criterion is not satisfied, the index of the iteration is updated as $g = g + 1$; then, the algorithm goes to Step 2. Otherwise, the algorithm is stopped, and π^* is regarded as the optimal scheduling policy. The total number of generated schedules is used as the stopping criterion to control the computation time because the simulation is the most time-consuming part of this algorithm. Here, performing SSGS under a scenario for a policy π is counted as one schedule.

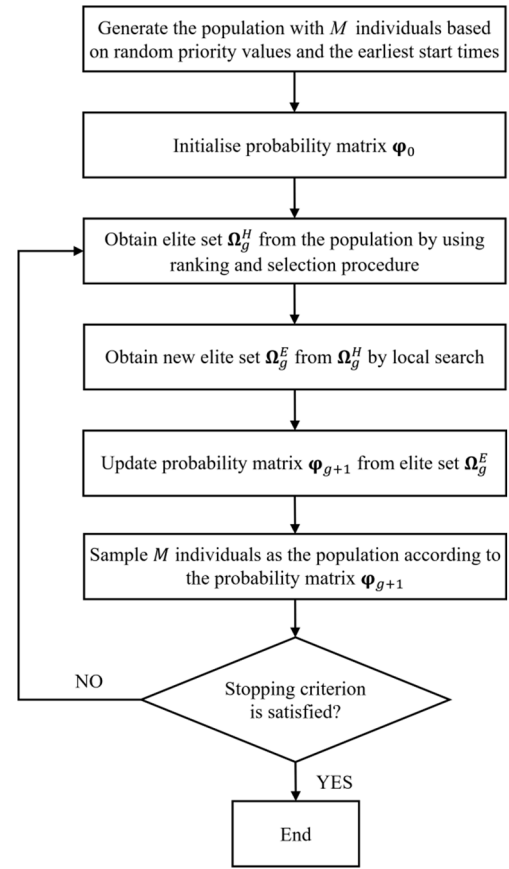


Fig. 1. Flowchart of the improved EDA.

4.1. Generating initial population

In the first step of the EDA, an initial population $\Omega_0 : \{\pi_{0,1}, \pi_{0,2}, \dots, \pi_{0,M}\}$ that includes M feasible solutions is generated. The start time of the project in the initial solutions is set to be the first time point of the planning horizon. Initial execution orders of the activities are created through random priority values and the earliest start times of the activities. The priority value of activity j is a uniformly distributed random variable ρ_j . The earliest start time of activity j is calculated using the expected value of activity durations, which ignores resource constraints as

$$\gamma_j = \max_{j \in P_j} \left(\gamma_j + \left(\bar{D}_{j, \gamma_j} \right) \right), j \in \chi(\psi). \quad (11)$$

Here, $\left(\bar{D}_{j, \gamma_j} \right)$ denotes the average duration of activity j starting from t, ψ denotes a set that contains all scheduled activities, and $\chi(\psi)$ represents the eligible set containing unscheduled activities whose predecessor activities have been scheduled. Activities in $\chi(\psi)$ are selected according to

$$\chi(\psi) = \{j | j \notin \psi \text{ and } P_j \subset \psi\} \quad (12)$$

Starting from $\psi = \{0\}$, activity $j \in \chi(\psi)$ with the smallest value of $\rho_j \times \gamma_j$ is selected to be scheduled next; this set of scheduled activities is updated as $\psi = \psi \cup \{j\}$. After all activities are scheduled, a precedence

feasible individual is generated.

4.2. Selecting elite solutions via the R&S procedure with the CRN

Estimating objective functions through simulation is the most time-consuming part of the optimisation. Instead of generating the same number of scenarios for different solutions, a sequential R&S procedure with the CRN is adopted in this study to identify the elite solutions. Thus, the simulation efforts are allocated to different solutions more appropriately.

The CRN is an effective method to reduce the variance when comparing two or more populations by generating positively correlated samples. In this study, the objective function is estimated by realising activity durations. Therefore, a positive correlation of the objective function corresponding to different solutions can be achieved by positively correlated activity durations. First, a random variable $\xi_{j,n}$ that is uniformly distributed between 0 and 1 is generated for activity j in scenario n for all solutions. The durations of activity j starting from different times are generated based on the same variable $\xi_{j,n}$ as

$$d_{j,t,n} = \Phi_{j,t}^{-1}(\xi_{j,n}). \quad (13)$$

Here, $\Phi_{j,t}^{-1}(\cdot)$ denotes the inverse cumulative distribution function of distribution $\Phi_{j,t}$. Consequently, the durations of the same activity that starts from different time points are positively correlated. Thus, objective functions of different solutions estimated based on these durations are positively correlated. The paired t -test is then used to test the difference between the expected objective functions of different solutions.

The sequential R&S procedure using the CRN is developed as Algorithm 4 to identify elite solutions. In Algorithm 4, $\zeta_n = [\xi_{1,n}, \xi_{2,n}, \dots, \xi_{J,n}]$ is the vector of random variables used to generate the activity durations in scenario n .

Algorithm 4: Identifying elite solutions using R&S procedure

Input: $\Omega_g = \{\pi_{g,1}, \pi_{g,2}, \dots, \pi_{g,M}\}$

- 1: Set the initial set of individuals $\mathcal{J}_0 = \Omega_g$ and the index of the iteration $z = 0$.
- 2: The number of simulated observations of the objective function corresponding to solution i is defined as n_i , and $|\mathcal{J}_z|$ denotes the number of individuals in \mathcal{J}_z .
- 3: Generate additional $3 - n_i$ observations of the objective function for individual $\pi_{g,i}$ in \mathcal{J}_z if $n_i < 3$ with $\zeta_{n_i+1}, \dots, \zeta_3$.
- 4: **while** the number of solutions in set \mathcal{J}_z is larger than Q
- 5: Generate one observation of the objective function with ζ_{n_i+1} for each $\pi_{g,i}$ in \mathcal{J}_z , and $n_i = n_i + 1$
- 6: The solutions in set \mathcal{J}_z are ranked according to the value of objective functions as $\pi_{g,1}, \pi_{g,2}, \dots, \pi_{g,|\mathcal{J}_z|}$. When the expected makespan (NPV) is used as the objective function, the solutions are sorted in an ascending (descending) order.
- 7: if the smallest number of observations for an individual in \mathcal{J}_z exceeds N_{max}
- 8: The R&S procedure is terminated, and the elite solution set is obtained as $\Omega_g^H = \{\pi_{g,1}, \pi_{g,2}, \dots, \pi_{g,Q}\}$.
- 9: **end if**
- 10: **for** $i_1 = Q+1$ to $|\mathcal{J}_z|$
- 11: Initialise the number of solutions that are significantly better than π_{g,i_1} as $count = 0$.
- 12: **for** $i_2 = 1$ to $i_1 - 1$
- 13: Perform the paired t -test using $n = \min(n_{i_1}, n_{i_2})$ pairs of observations of objective functions corresponding to the solutions π_{g,i_1} and π_{g,i_2}
- 14: if the paired t -test indicates that solution π_{g,i_1} is statistically worse than π_{g,i_2} at a significance level α
- 15: $count = count + 1$;
- 16: **end if**
- 17: **end for**
- 18: if $count \geq Q$ Solution π_{g,i_1} is removed from set \mathcal{J}_z
- 19: **end if**
- 20: **end for**
- 21: $\mathcal{J}_{z+1} = \mathcal{J}_z$ and $z = z + 1$
- 22: **end while**
- 23: **Return** the elite solution set: $\Omega_g^H = \mathcal{J}_z$

4.3. Local search method

A local search method is adopted to further enhance the exploitation ability of the EDA, which is realised through the weighted random sampling of the start time τ and random alteration of the execution order of activities. In weighted random sampling, each item is randomly selected according to a probability equal to its relative weight (Efratidis & Spirakis, 2006). In this study, the relative weight of the start time τ is computed according to its working probability as

$$p_\tau^w = w(\tau) / \sum_{\tau=T_{ST}}^{T_{ED}} w(\tau), \tau \in [T_{ST}, T_{ED}] \quad (14)$$

T_{ST} and T_{ED} are the predetermined upper and lower bounds of the start date τ , respectively. First, n_W different start times are selected for a solution $\pi_{g,i}^H$ in the elite set Ω_g^H . Thus, n_W additional solutions are generated according to $\pi_{g,i}^H$. These $n_W + 1$ solutions are evaluated using Algorithm 4, and the optimal individual is selected as a member of the new solution set Ω_g^W .

For a solution in set Ω_g^W , each activity has a probability P_{per} to be selected. The selected activity j is inserted into the position of another randomly selected activity x between RM_j and LM_j . As shown in Fig. 2, RM_j is the rightmost position of the predecessor activities of activity j , and LM_j is the leftmost position of the successor activities of activity j . If the position of activity x is on the left of j , j will be inserted before activity x . Otherwise, j will be placed after activity x . Activities between RM_j and LM_j have the same probability of being selected as activity x . Then, new individuals are evaluated together with Ω_g^W , and Q individuals are selected as the new elite set Ω_g^E .

4.4. Probability model to sample individuals for the next generation

New individuals in the next generation of the EDA are sampled according to the probability model built using elite individuals in Ω_g^E . To build this probability model, a solution $\pi = [\lambda, \tau]$ is transferred into a matrix of binary variables

$$\mathbf{X} = \begin{bmatrix} A_{1,1} & \dots & A_{1,J} & A_{1,J+1} \\ \vdots & \ddots & \vdots & \vdots \\ A_{J,1} & \dots & A_{J,J} & A_{J,J+1} \end{bmatrix} \quad (15)$$

Here,

$$A_{j,j'} = \begin{cases} 1, & j \text{ is scheduled before } j' \text{ in } \pi, \forall j, j' \in \{1, 2, \dots, J\} \\ 0, & \text{else} \end{cases} \quad (16)$$

The first and last dummy activities are not considered here. The last column of matrix \mathbf{X} , i.e. $[A_{1,J+1}, A_{2,J+1}, \dots, A_{J,J+1}]$, is the binary form of the start date τ . Here, $A_{J,J+1}$ is the lowest bit of the binary vector. Thus, solution π is mapped into a binary matrix \mathbf{X} through a function $\mathbf{X} = G(\pi)$. For instance, the solution $\pi = [0, 2, 3, 1, 4, 3]$ with dummy activities 0 and 4 is converted to

$$\mathbf{X} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix} \quad (17)$$

The probability matrix of generation g is defined as

$$\Phi_g = \begin{bmatrix} \Pr(A_{1,1}^g = 1) & \dots & \Pr(A_{1,J}^g = 1) & \Pr(A_{1,J+1}^g = 1) \\ \vdots & \ddots & \vdots & \vdots \\ \Pr(A_{J,1}^g = 1) & \dots & \Pr(A_{J,J}^g = 1) & \Pr(A_{J,J+1}^g = 1) \end{bmatrix} \quad (18)$$

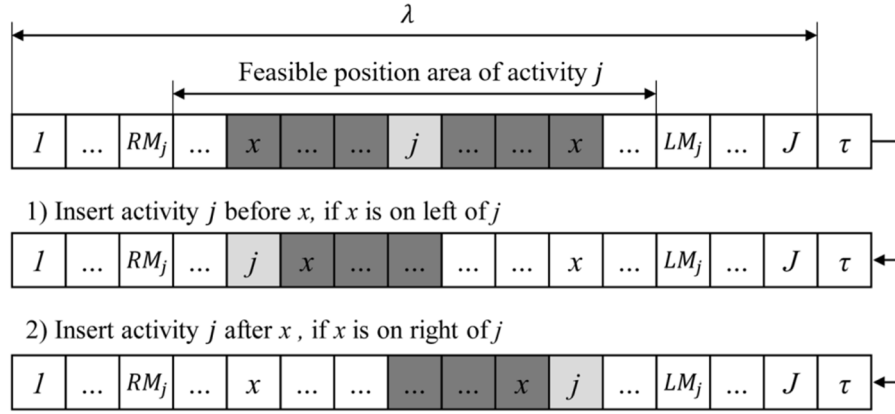


Fig. 2. Alteration of the execution order of activities during local search.

The initial probability matrix Φ_0 is assumed as

$$\Phi_0 = \begin{bmatrix} 0 & 0.5 & \dots & 0.5 & 0 \\ 0.5 & 0 & \dots & \vdots & \vdots \\ \vdots & \vdots & \ddots & 0.5 & 0 \\ 0.5 & \dots & 0.5 & 0 & 1 \end{bmatrix} \quad (19)$$

The probabilistic model Φ_g is updated according to the solutions in elite set Ω_g^E as

$$\Phi_{g+1} = L_{rate} \frac{1}{Q} \sum_{i=1}^Q G(\pi_{g,i}^E) + (1 - L_{rate}) \Phi_g. \quad (20)$$

Here, $L_{rate} \in (0, 1)$ is the learning rate.

4.5. Sampling a new population from the probability model

The probability matrix Φ_{g+1} is used to sample the population of generation $g + 1$. The sampling probability for the j^{th} bit of τ is

$$P_{ST,j} = (\Phi_{g+1})_{j,J+1}, \forall j \in \{1, 2, \dots, J\}. \quad (21)$$

Here, $(\Phi_{g+1})_{j,J+1}$ denotes the element of the matrix Φ_{g+1} in row j and column $J + 1$. Then, a binary random number $A_{j,J+1} \in \{0, 1\}$ is generated according to the probability $\Pr(A_{j,J+1} = 1) = P_{ST,j}$. Finally, a sampled τ is given by

$$\tau = \sum_{j=1}^J A_{j,J+1} \times 2^{J-j}. \quad (22)$$

Unlike the sampling method in Fang et al. (2015), the execution order of activities is obtained according to the duration of activities and resource constraints. The detailed process is listed in Algorithm 5. At each step, an activity j is selected from the set of unscheduled activities χ based on the probability

$$P_{List,j} = \frac{\sum_{j \in \chi} (\Phi_{g+1})_{j,j}}{\sum_{j \in \chi} (\Phi_{g+1})_{j,j}}. \quad (23)$$

The sampling procedure is repeated M times, and M new individuals are

obtained as the new population.

Algorithm 5 Process of sampling the execution order of activities

Input: Probability matrix Φ_g and average activity durations $\bar{\mathbf{D}}$

- 1: Initialise remaining resources at time t : $R_{k,t} = R_k \forall t = 1, 2, \dots, T$
- 2: Initialise start and end times of the first dummy activity $S_0 = E_0 = \tau$, and the set of scheduled activities $\psi = \{0\}$
- 3: **for** $m = 1$ **to** $J + 1$
- 4: Sample an activity j from the unscheduled activities χ according to the probabilities in Eq. (23).
- 5: The earliest start time of activity j is $S' = \max\{E_j | j' \in P_j\}$
- 6: **While** $R_{k,t} < r_{jk}, \forall t \in [S', S' + (\bar{\mathbf{D}})_{j,S'}]$
- 7: $S' = \min\{E_j | E_j > S', j' \in \psi\}$
- 8: **end while**
- 11: $S_j = S', E_j = S_j + (\bar{\mathbf{D}})_{j,S_j}$ and $\psi = \psi \cup \{j\}$
- 12: Update the remaining resources as $R_{k,t} = R_{k,t} - r_{j,k}$ for $t \in [S_j, E_j]$
- 13: **end**
- 14: **Return** A solution $\pi = [a_0, a_1, \dots, a_m, \dots, a_{J+1}, \tau]$ where $S_{a_m} < S_{a_m'}$ if $m < m'$

5. Numerical studies using the J120 dataset from PSPLIB

The J120 dataset from PSPLIB provides benchmark instances of project scheduling that contains 600 RCPSPs with 120 non-dummy activities (Kolisch & Sprecher, 1997). For the numerical study, 60 RCPSPs were randomly selected from the J120 dataset. Then, the 60 RCPSPs were modified to SRCPSPs by assuming random activity durations that depend on time-varying weather conditions. This section evaluates the performance of the improved EDA using modified benchmark project instances. Only the expected makespan was used as the objective function in this section. The algorithms were implemented on a desktop computer with an Intel i9-9900 CPU and 8 GB of memory using MATLAB 2018b.

5.1. Weather condition modelling

Simulated weather conditions were used for the numerical study. The time unit was one day, and the horizon of the weather condition model was one year. The effect of weather conditions on construction operations was described by climate reduction coefficients (CRCs) (Ballesteros-Pérez, Smith, Lloyd-Papworth, & Cooke, 2018). The CRCs refer to the proportion of workable days during the same time in

historical years. The sine wave was used to model the CRC data, which is given by

$$w(t) = 0.5 + 0.2\cos(2\pi(t/365 - 0.5)), t \in \{1, 2, 3, \dots, 365\} \quad (24)$$

Here, $w(t)$ denotes the probability that day t is a workable day. Then, the expected activity duration d_{jt}^E is calculated using Algorithm 6. Further, it was assumed that the activity duration follows a continuous uniform distribution within the interval $[0, 2d_{jt}^E]$.

Algorithm 6 Calculation of expected duration of activity j starting from time t	
1:	Set the accumulate effective working time: $count = 0$, and set the expected duration of activity j starting from $t: d_{jt}^E = 0$
2:	While $count < d_j^E$
3:	$count = count + w(t)$ and $d_{jt}^E = d_{jt}^E + 1$
4:	$t = t + 1$, and set $t = 1$ when $t = 366$
5:	end while

5.2. Overall performance of improved EDA

According to Ballestín (2007), the performance of an algorithm for an RCPSP was measured through the relative deviation between the optimal expected makespan $MS = F(\pi^*)$ and the critical path-based lower bound LB ; this is equated as $(MS - LB)/LB$. To measure the overall performance of the improved EDA on the J120 dataset, the average deviation from the lower bound (Ave.Dev.LB) of the 60 selected instances was used as the indicator.

The performance of the improved EDA depends on the parameters of the algorithm. Inspired by Fang et al. (2015), an orthogonal experiment was conducted to identify an appropriate combination of parameters. Important parameters in the EDA include the population size M , proportion of elite individuals η , sample size of the start date n_W in the local search, probability P_{per} of selecting an activity in the local search, and learning rate L_{rate} . Four factor levels were set for each parameter during the orthogonal experiment. According to the orthogonal table $L_{16}(4^5)$, 16 parameter combinations were tested. For each combination, 30,000 schedules were generated to determine the optimal expected makespan for each problem. The makespan corresponding to the optimal solution for each instance was evaluated using 1,000 additional simulated schedules for a more accurate estimate.

The two additional parameters used by the improved EDA are N_{max} and α . Because of the interaction between parameters, the result of the

orthogonal experiment became unreliable when N_{max} and α were selected via an orthogonal experiment. Consequently, two parameters were determined outside the orthogonal experiment. After several trials, the two parameters were selected as $N_{max} = 50$ and $\alpha = 0.01$. There are more suitable combinations of N_{max} and α ; however, the numerical examples in the rest of the paper show that the improved EDA using $N_{max} = 50$ and $\alpha = 0.01$ already outperform the original EDA. Therefore, this research did not search for a better combination of the two parameters. The performance trends of the factor levels of the improved EDA are illustrated in Fig. 3. Accordingly, the parameters used by the improved EDA were selected as $M = 100$, $\eta = 0.01$, $n_W = 20$, $P_{per} = 0.1$, $L_{rate} = 0.9$, $N_{max} = 50$, and $\alpha = 0.01$. Under these parameters, the Ave.Dev.LB of the 60 selected J120 instances calculated by the improved EDA was 1.409.

5.3. Comparisons between different versions of EDAs

In the existing research on SRCPSP, the fitness function of a solution is often evaluated with a fixed number of simulation scenarios. Ballestín (2007) confirmed that a small number of simulation scenarios are beneficial because more solutions can be found and evaluated. Therefore, in the original EDA without the R&S method, 10 scenarios were generated for each solution. Other parameters of the original EDA were selected using the orthogonal experiment as $M = 150$, $\eta = 0.01$, $n_W = 10$, $P_{per} = 0.05$, and $L_{rate} = 0.9$. The Ave.Dev.LB of the 60 instances was then calculated as 1.451 using 30,000 schedules for each instance. The Ave.Dev.LB obtained by the improved EDA using the same number of schedules is 1.409. A deterministic scheduling approach using the EDA is adopted in this section, where the makespan is calculated directly using the expected activity durations. Deterministic scheduling was repeated 20 times for each instance; the optimal policy identified by deterministic scheduling was evaluated in 1,000 simulated scenarios. The Ave.Dev.LB obtained by deterministic scheduling was 1.501. Therefore, the original and improved EDA reduced the Ave.Dev.LB by 3.33% and 6.09%, respectively, when compared with the deterministic scheduling approach.

Three projects with different characteristics were randomly selected from the J120 dataset to further explore the convergence process of the EDA; they are named projects #1, #2, and #3. Parameters of the three projects are listed in the first row of Table 2. The parameters of the EDAs were selected using an orthogonal experiment, and the results are listed

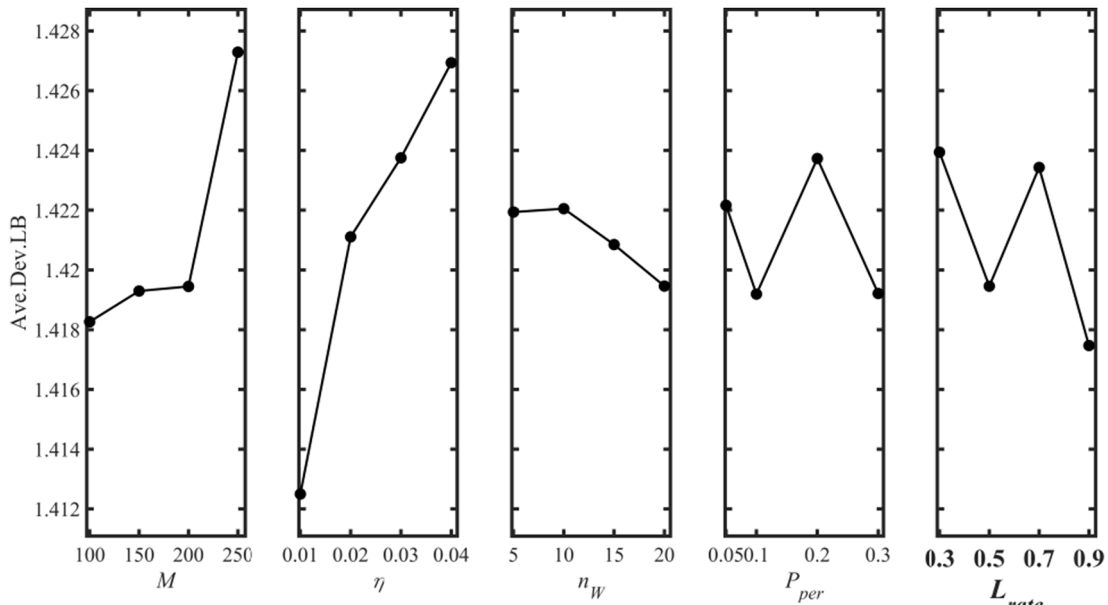
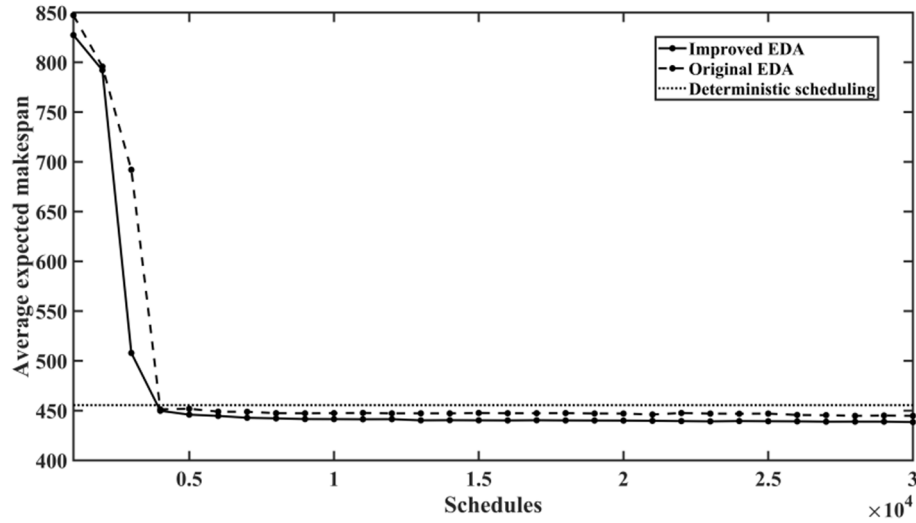
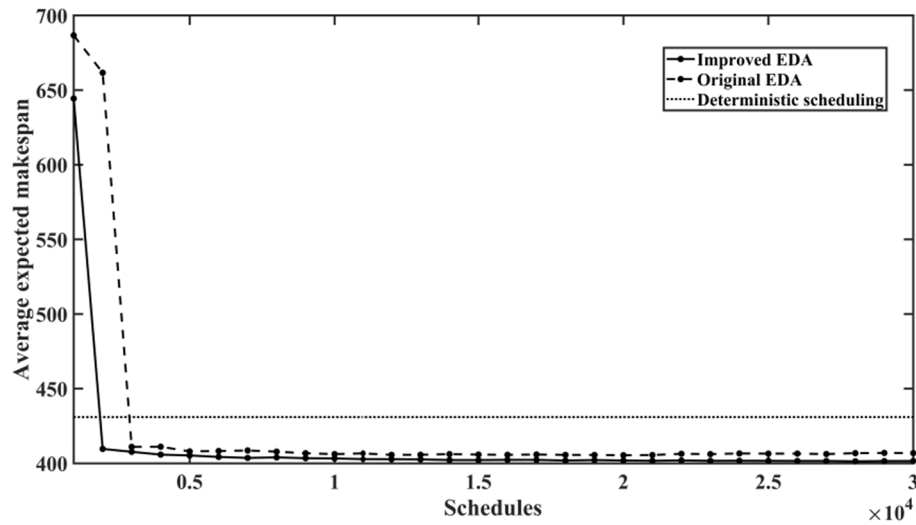


Fig. 3. Trend of factor levels of parameters in the orthogonal experiment.

Table 2

Parameters of the three projects and those used in the EDAs.

	Project#1	Project#2	Project#3
(RF, RS, NC)	(0.25, 0.1, 1.5)	(0.5, 0.3, 1.8)	(1, 0.5, 2.1)
Original EDA ($M, \eta, n_W, P_{per}, L_{rate}$)	(150, 0.01, 10, 0.05, 0.5)	(100, 0.02, 20, 0.05, 0.7)	(150, 0.01, 5, 0.05, 0.5)
Improved EDA ($M, \eta, n_W, P_{per}, L_{rate}$)	(200, 0.01, 5, 0.05, 0.9)	(100, 0.01, 10, 0.05, 0.9)	(150, 0.01, 20, 0.05, 0.9)

**Fig. 4.** Convergence processes of EDAs corresponding to project#1 with $RF = 0.25$, $RS = 0.1$, and $NC = 1.5$.**Fig. 5.** Convergence processes of EDAs corresponding to project#2 with $RF = 0.5$, $RS = 0.3$, and $NC = 1.8$.

in the second and third rows of Table 2. The convergence processes of projects #1, #2, and #3 are plotted in Figs. 4–6. Each convergence curve in the figures is the average result of 20 replications of the EDA. The convergence processes in Figs. 4–6 show that the improved EDA can obtain a shorter makespan compared to the original EDA during the entire convergence process. The makespan obtained by deterministic scheduling is also provided in the figures for comparison. Compared with deterministic scheduling, the makespan reduction of stochastic scheduling is calculated in Table 3. The results show that the improved EDA can obtain a more significant makespan reduction than the original EDA. The elapsed time and optimal makespan obtained at different stages of the EDA are compared in Table 4. Table 4 indicates that the computing time of the improved EDA is longer than that of the original

EDA under the same number of schedules because of the additional operators in the R&S procedure. However, the improved EDA can still obtain a better solution by using less computing time. The improved EDA can obtain an expected makespan 441.5 days in 287 s when scheduling project #1. In contrast, the original EDA required 587 s to achieve an average makespan 444.6 days. Therefore, the improved EDA is more efficient and can obtain a shorter expected makespan.

5.4. Analysis of optimisation process

An important contribution of this study is embedding the R&S procedure into the EDA, and therefore, it is necessary to investigate the performance of the R&S procedure for identifying elite solutions. To this

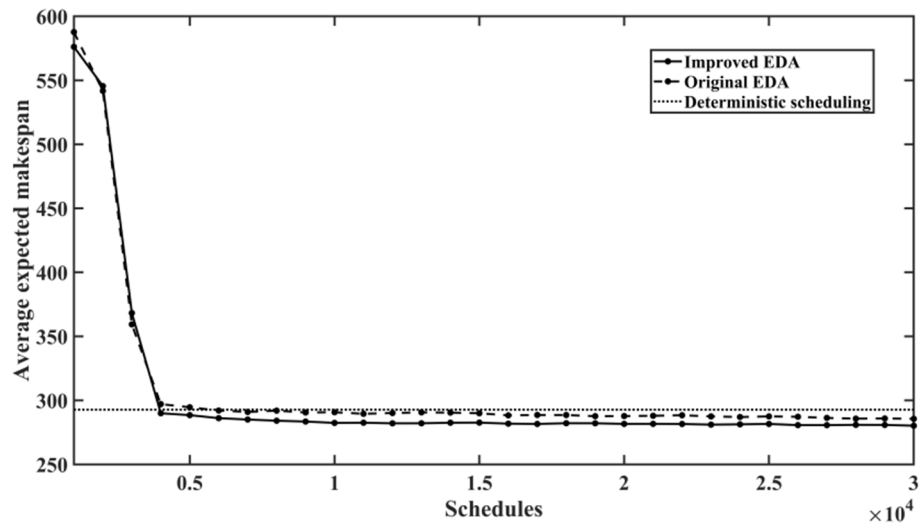


Fig. 6. Convergence processes of EDAs corresponding to project#3 with RF = 1, RS = 0.5, and NC = 2.1.

Table 3
Makespan reduction compared with deterministic scheduling.

Methods	Project#1	Project#2	Project#3
Improved EDA	17.0	29.4	12.5
Original EDA	10.8	24.1	7.1

Table 4
Elapsed time and optimal makespan obtained at different stages of EDAs.

Project	Schedules	Original EDA		Improved EDA	
		Makespan	CPU time (s)	Makespan	CPU time (s)
#1	10,000	448.8	184	441.5	287
	20,000	447.3	371	440.0	528
	30,000	444.6	587	438.5	765
#2	10,000	409.5	180	403.3	248
	20,000	404.7	372	401.9	443
	30,000	406.5	569	401.5	641
#3	10,000	290.8	167	282.4	224
	20,000	288.0	372	281.6	408
	30,000	286.7	575	280.2	602

end, 100 policies of project#1 visited by the original EDA were randomly selected. First, 10 simulated schedules were generated for each solution, and two elite solutions were selected according to the mean values of the simulated makespans. Subsequently, the R&S procedure was adopted to identify two elite solutions from the 100 solutions, and the R&S procedure terminated when the total number of simulated schedules was greater than 1,000. The two methods were repeated 50 times to process the same 100 policies. Each obtained elite solution was evaluated using 1,000 additional simulated schedules to obtain more accurate estimates of the makespan. The makespans of the elite solutions identified by the two methods are compared in Fig. 7; this figure shows that the elite solutions obtained by the R&S procedure incur significantly shorter makespans. Further, the variance among the results of different replications of the R&S method are significantly smaller. Further, 41 out of 50 replications of the R&S procedure consumed less than 1,000 simulated schedules to identify two elite solutions. Thus, the R&S procedure can select elite solutions more effectively.

The R&S procedure is more effective because some inferior solutions are removed at an early stage. The histogram of the number of simulated schedules for a solution in a randomly selected convergence process of

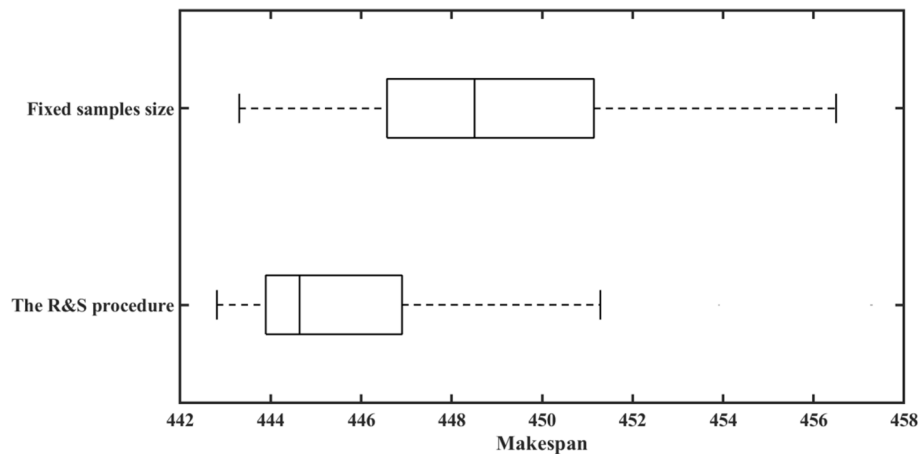


Fig. 7. Boxplot of the makespans of the elite solutions.

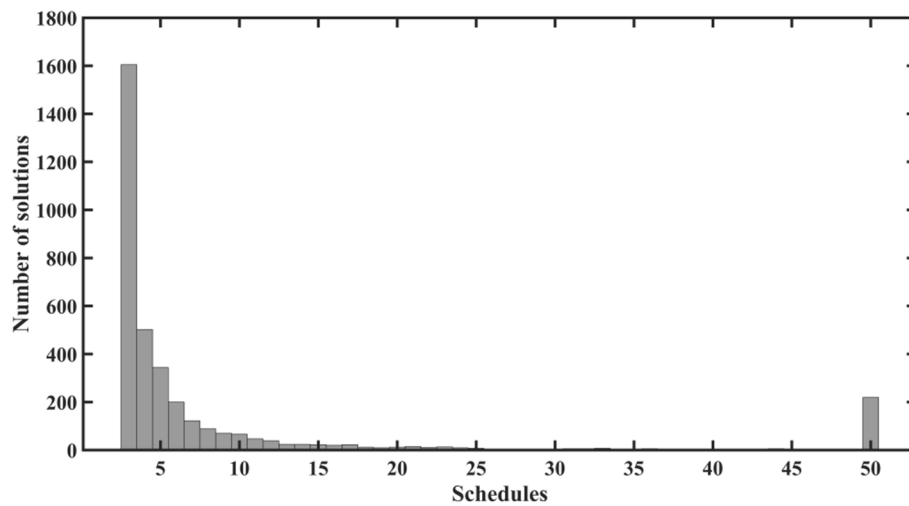


Fig. 8. Histogram about the number of schedules simulated for a solution by the improved EDA.

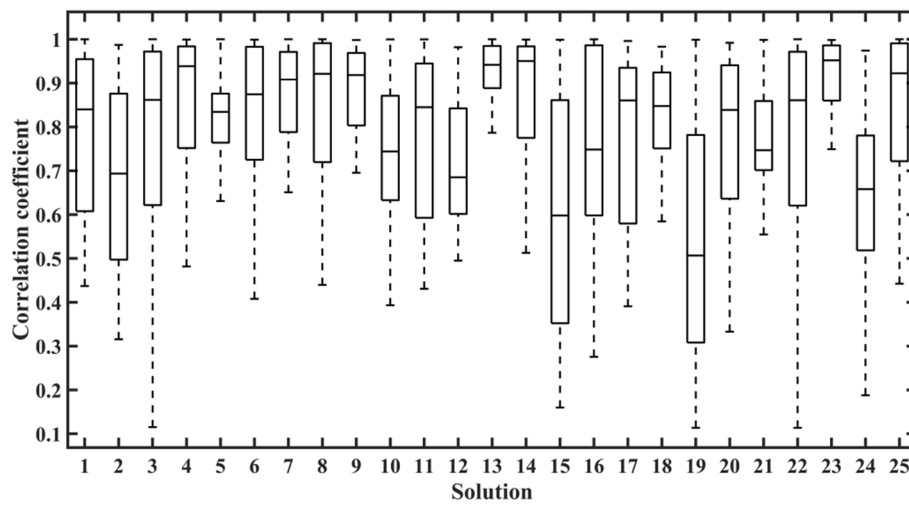


Fig. 9. Boxplots of the correlation coefficients of makespans in the improved EDA.

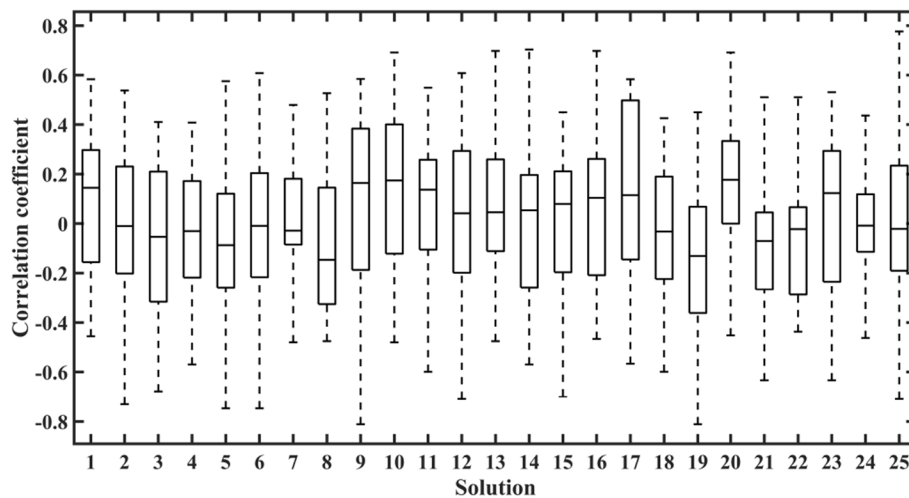


Fig. 10. Boxplots of correlation coefficients of makespans in the original EDA.

Table 5

Effect of project characteristics on scheduling result.

Characteristics	NC			RF				RS				
	1.5	1.8	2.1	0.25	0.5	0.75	1	0.1	0.2	0.3	0.4	0.5
Ave.Dev.LB	1.392	1.372	1.451	1.178	1.283	1.471	1.611	2.107	1.760	1.353	1.081	0.832

the improved EDA for project#1 is plotted in Fig. 8; it shows that most solutions are removed from the R&S procedure after less than five simulated schedules. Thus, extensive simulation of low-quality solutions is avoided, and the simulation efforts are allocated more appropriately.

The method for generating positively correlated makespans of different solutions via the CRN is validated. Simulated makespans generated in the two versions of the EDA when scheduling project#1 were analysed; the simulated makespans of 25 randomly selected solutions were collected. The boxplots of the correlation coefficients corresponding to the two algorithms are shown in Figs. 9 and 10, respectively. Each boxplot in the two figures represents the distribution of the correlation coefficients between the makespan of a solution and those of the other 24 solutions. Fig. 9 shows that the simulated makespans of most solutions are positively correlated when the CRN is used. Fig. 10 indicates that the simulated makespan is almost statistically independent without the CRN. Therefore, the simulation-based method using CRN can derive positively correlated makespans that can reduce the variance of the difference between the average makespan of different solutions.

5.5. Sensitivity analysis of results obtained by improved EDA

Instances in the J120 dataset were generated according to three parameters: network complexity ($NC \in \{1.5, 1.8, 2.1\}$), resources factor ($RF \in \{0.25, 0.5, 0.75, 1\}$), and resource strength ($RS \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$). NC is defined as the average number of predecessors per activity; RF refers to the average fraction of resource types required by a non-dummy activity, RS reflects the relationship between resource requirements and supply, and a small value of RS implies a scarce resource supply.

The results for some selected project instances are listed in Table 5. Ave.Dev.LB increases with RF and decreases with RS . The change in Ave.Dev.LB is nonmonotonic when NC increases. These findings are similar to those reported by Fang et al. (2015). A large RF value and a small RS value indicate more severe resource constraints on the SRCPSP. Therefore, the difference between the obtained makespan and lower bound that does not consider the resource constraints is more obvious. In contrast, a large NC value reveals that precedence constraints on the activities are strict. The precedence constraints affect both the obtained makespan and lower bound of the makespan. Consequently, Ave.Dev.LB does not change monotonically when NC increases.

Table 6

Steps to construct a wind turbine.

Index	Step Name	Resource	Index	Step Name	Resource
1	Road construction	1	8	Foundation backfilling	8
2	Foundation material unloading		9	Tower unloading	9
3	Foundation excavation	2	10	Wind turbine shell unloading	9
4	Pile driving	5	11	Impeller unloading	9
5	Steel bar binding	6	12	Generator unloading	9
6	Foundation grouting	7	13	Blade unloading	9
7	Foundation maintenance		14	Hoisting	9, 3, 4

6. Case study about windfarm construction

A windfarm construction project was studied to demonstrate the performance of the proposed method in real-world case studies; the case study also reveals management implications of the proposed method in practical project scheduling problems. The data and parameters used in this case study are not provided in the text because of confidentiality policies. Detailed information about the project has been not been provided for the same reason.

6.1. Windfarm construction project

There are ten wind turbines on the windfarm, and 14 steps are required to construct a wind turbine. Therefore, the construction project contains a total of 140 activities. The 14 steps required to construct a wind turbine are listed in Table 6; the required resources for each step are also demonstrated. A total of 9 renewable resources with limited amounts were considered; each resource was represented by a number. Renewable resources include engineering machines and technicians required for different construction activities.

A payment is received from the owner of the windfarm after the project is completed. There is a fixed cost for an activity in the project, which is independent of the duration of the activity. The fixed cost is spent preparing for the execution of the activity; for example, the transportation of materials, equipment, and technicians. Further, the variable cost of an activity is proportional to the duration of the activity, and it includes the salary of technicians and the rent of equipment. The NPV is used as another objective function besides the makespan, and this is calculated using the project payment, fixed activity cost, and variable activity cost.

These activities are sensitive to weather conditions. For example, there is a safety threshold for the wind speed for hoisting, and most activities cannot be performed in rain or snow. Therefore, the duration of activities is calculated considering weather conditions. This case study uses historical weather data from the China Meteorological Data Service Centre, which contains daily mean values of wind speed and perception over 40 years (from 1979 to 2018). After standard durations of activities were provided by relevant experts, the activity durations were re-calculated using Algorithm 1. The beta distribution was used to model activity durations, and the three-point estimation method in Hajdu (2013) was used to identify the parameters. The expected value $\mu(j, t)$ and the two parameters of the beta distribution (i.e. $a(j, t)$ and $b(j, t)$) for the duration of activity j starting from t were calculated as

$$\mu(j, t) = (V_O(j, t) + V_P(j, t) + 4 \times V_M(j, t))/6 \quad (25)$$

$$a(j, t) = \frac{(\mu(j, t) - V_O(j, t)) \times (2 \times V_M(j, t) - V_O(j, t) - V_P(j, t))}{(V_M(j, t) - \mu(j, t)) \times (V_P(j, t) - V_O(j, t))}, \quad (26)$$

and

$$b(j, t) = a(j, t) \times [V_P(j, t) - \mu(j, t)] / [\mu(j, t) - V_O(j, t)] \quad (27)$$

Here, $V_O(j, t)$, $V_P(j, t)$, and $V_M(j, t)$ represent the minimum value, maximum value, and mode of the activity durations in set $D_{j,t}$, respectively. After normalisation, the distribution of the duration of activity j starting from time t is given by

$$[d_{j,t} - V_P(j, t)] / [V_O(j, t) - V_P(j, t)] \sim \text{beta}(a(j, t), b(j, t)) \quad (28)$$

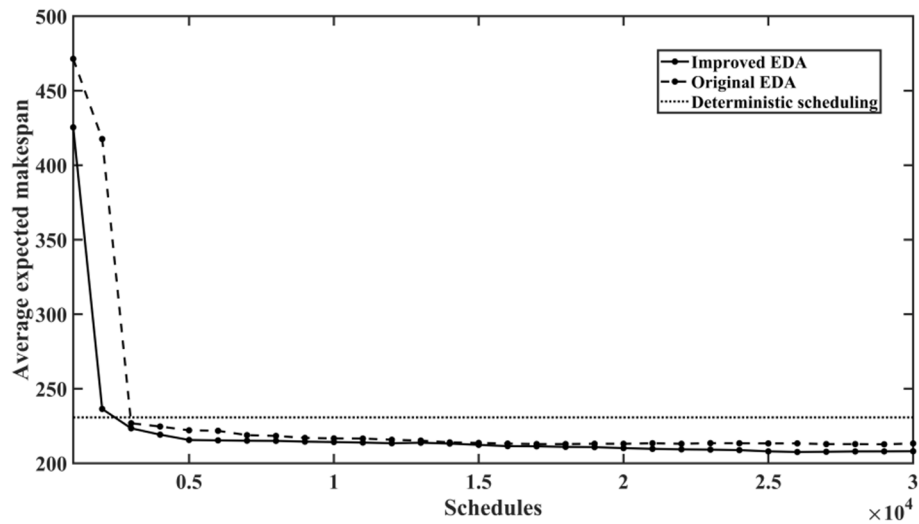


Fig. 11. Convergence processes of EDAs to minimise the expected makespan.

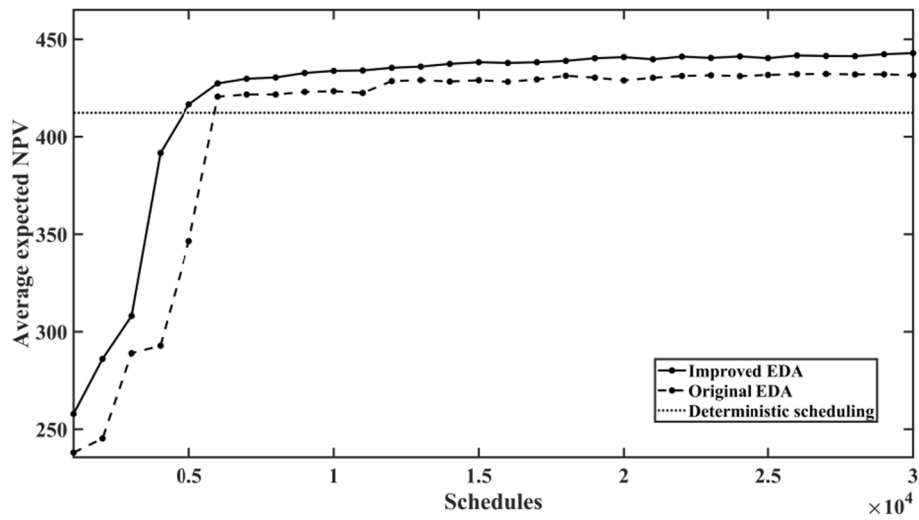


Fig. 12. Convergence processes of EDAs to maximise the expected NPV.

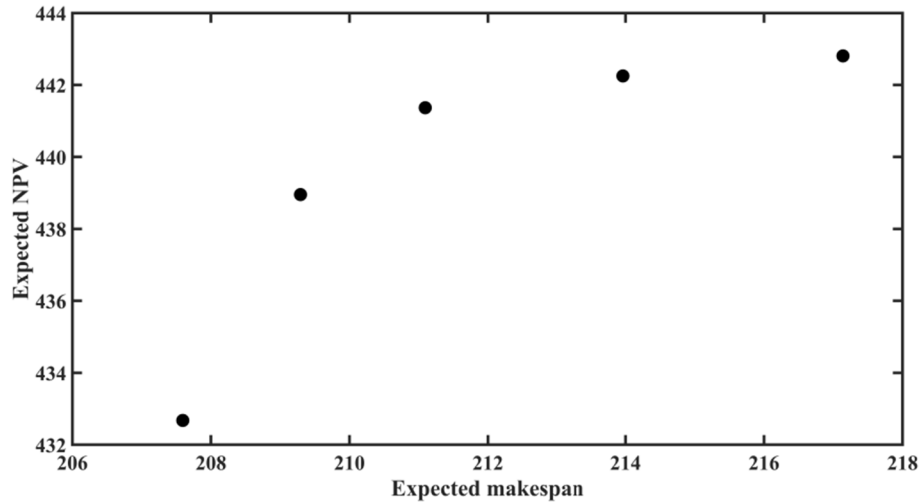


Fig. 13. Pareto solutions for the SRCPSP about windfarm construction.

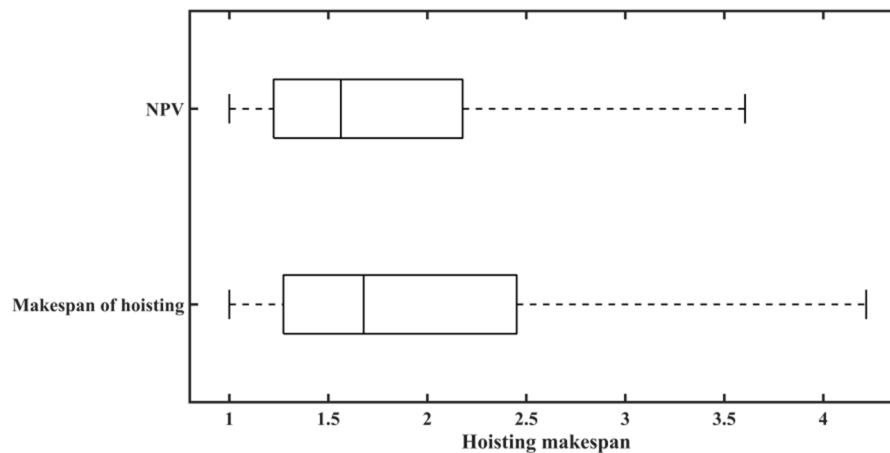


Fig. 14. Distributions of the makespan of hoisting under different objective functions.

6.2. Scheduling results and comparisons

When the project was scheduled according to the objective function of the expected makespan, the parameters of the original and improved EDA were selected using the orthogonal experiment as ($M = 100$, $\eta = 0.01$, $n_W = 20$, $P_{per} = 0.05$, $L_{rate} = 0.9$) and ($M = 100$, $\eta = 0.01$, $n_W = 5$, $P_{per} = 0.2$, $L_{rate} = 0.9$), respectively. The convergence processes of the improved EDA and original EDA are plotted in Fig. 11, the results of the deterministic scheduling are marked in the figure. Compared with the result of deterministic scheduling, the original and improved EDA reduced the expected makespan by 17.5 and 22.7 days, respectively.

When the expected NPV was considered an objective function, the parameters of the original and improved EDA were selected as ($M = 200$, $\eta = 0.02$, $n_W = 15$, $P_{per} = 0.1$, $L_{rate} = 0.9$) and ($M = 150$, $\eta = 0.01$, $n_W = 10$, $P_{per} = 0.2$, $L_{rate} = 0.9$), respectively. The comparison result of the three methods is shown in Fig. 12. The results indicated that the original and improved EDA increased the NPV by 19.24 and 30.60, respectively, based on the result of deterministic scheduling. Consequently, the improved EDA achieves a shorter expected makespan and a larger expected NPV in a practical case study.

When both the expected makespan and NPV are considered during scheduling, the SRCPSP becomes a bio-objective optimisation problem. This research converts the bio-objective optimisation problem to several single-objective optimisation problems that use weighted combinations of NPV and makespan as the objective function. The solutions of the single-objective optimisation problems are non-dominated solutions of the bio-objective optimisation problem. The weights of NPV and makespan were selected as (0, 1), (1, 2), (1, 1), (1, 0.5), and (1, 0). Five corresponding Pareto solutions are illustrated in Fig. 13.

A short makespan can increase the NPV because the project payment is not received until it is completed. Further, the NPV is influenced by the cost of the activities. The duration of an activity with a large variable cost should be reduced to increase NPV. For example, the hoisting of turbines requires a crane and other devices. The rents of these devices are expensive. Therefore, a short duration of hoisting is desired to reduce the variable cost and increase the NPV. Two solutions under the objective functions of the expected project makespan and NPV were selected. According to each of the two solutions, 1,000 simulated schedules were generated. Distributions of the hoisting duration under the two different objective functions are plotted in Fig. 14. The average durations of the hoisting under the objective function of makespan and NPV were 2.14 and 1.88, respectively. The p-value of the t -test was 2.86×10^{-52} , which indicates that the average duration of hoisting under the objective function of the NPV is significantly shorter.

7. Conclusions

This study investigated an SRCPSP considering weather conditions. The execution order of the activities and the start date of the project were optimised, which allowed decision makers to schedule activities under time-varying weather conditions. Both the expected makespan and the expected NPV were considered as objective functions to reduce the average project makespan and shorten the average durations of activities with large variable costs. An improved EDA was developed to solve the SRCPSP, which includes a recursive R&S method to allocate simulation efforts to different solutions appropriately. The CRN was introduced into the generation of scenarios to further enhance the performance of the scheduling algorithm. The improved EDA was used to schedule 60 benchmark projects selected and modified from the J120 dataset in the PSPLIB. The results showed that the improved EDA can obtain a shorter expected makespan than the original EDA. Furthermore, the improved EDA converged to the optimal solution more quickly. The improved EDA was used to process a practical scheduling problem for windfarm construction, where both the expected makespan and NPV were adopted as the objective function. The results of the practical case study confirm the superiority of the improved EDA.

In addition to weather conditions, other uncertain factors can be processed by the improved EDA after these factors are considered in the simulation process. The proposed algorithm can be applied to similar domains. For example, in some practical job-shop scheduling problems, the processing time of an operation is a random variable because of the uncertain conditions of machines and operators. The improved EDA can be used to solve job-shop scheduling problems after appropriate modification. Another possible direction for future research is enhancing the efficiency of the improved EDA by further reducing simulation efforts. Subsequently, SRCPSPs of a larger scale can be solved. Finally, SGS methods that are more timesaving and resource efficient than the SSGS can be developed to enhance the performance of the project scheduling algorithm.

CRediT authorship contribution statement

Yifan Zhou: Conceptualization, Methodology, Writing - review & editing, Software. **Jindan Miao:** Writing - original draft, Software, Visualization. **Bin Yan:** Investigation, Resources. **Zhisheng Zhang:** Writing - review & editing.

Declaration of Competing Interest

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

Acknowledgements

The research work is supported by the National Natural Science Foundation of China (Grant Nos. 71671041).

References

- Alrabghi, A., & Tiwari, A. (2016). A novel approach for modelling complex maintenance systems using discrete event simulation. *Reliability Engineering & System Safety*, 154, 160–170.
- Ballesteros-Pérez, P., Smith, S. T., Lloyd-Papworth, J. G., & Cooke, P. (2018). Incorporating the effect of weather in construction scheduling and management with sine wave curves: Application in the United Kingdom. *Construction Management and Economics*, 36(12), 666–682.
- Ballestín, F. (2007). When it is worthwhile to work with the stochastic RCPSP? *Journal of Scheduling*, 10(3), 153–166.
- Ballestín, F., & Leus, R. (2009). Resource-constrained project scheduling for timely project completion with stochastic activity durations. *Production and Operations Management*, 18(4), 459–474.
- , 2015Bruni, M.E., Beraldi, P., Guerriero, F. (2015). The Stochastic Resource-Constrained Project Scheduling Problem. In: *Handbook on Project Management and Scheduling Vol. 2* (pp. 811–835).
- Bruni, M. E., Beraldi, P., Guerriero, F., & Pinto, E. (2011). A heuristic approach for resource constrained project scheduling with uncertain activity durations. *Computers and Operations Research*, 38(9), 1305–1318.
- Buchholz, P., & Thummler, A. (2005). Enhancing evolutionary algorithms with statistical selection procedures for simulation optimization. In: *Proceedings of the Winter Simulation Conference*, 2005. (pp. 11 pp.).
- Chen, E. J. (2016a). Sample sizes and stopping rules. In *Crafts of Simulation Programming* (pp. 27–49). World Scientific.
- Chen, E. J. (2016b). Using common random numbers with selection procedures. In *Crafts of simulation programming* (pp. 219–236). World Scientific.
- Chen, Z., Demeulemeester, E., Bai, S., & Guo, Y. (2018). Efficient priority rules for the stochastic resource-constrained project scheduling problem. *European Journal of Operational Research*, 270(3), 957–967.
- Choi, S. H., & Kim, T. G. (2018). Efficient ranking and selection for stochastic simulation model based on hypothesis test. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(9), 1555–1565.
- Creemers, S. (2015). Minimizing the expected makespan of a project with stochastic activity durations under resource constraints. *Journal of Scheduling*, 18(3), 263–273.
- Creemers, S., Leus, R., & Lambrecht, M. (2010). Scheduling Markovian PERT networks to maximize the net present value. *Operations Research Letters*, 38(1), 51–56.
- Davari, M., & Demeulemeester, E. (2019). The proactive and reactive resource-constrained project scheduling problem. *Journal of Scheduling*, 22(2), 211–237.
- Deblaere, F., Demeulemeester, E., & Herroelen, W. (2011). Reactive scheduling in the multi-mode RCPSP. *Computers and Operations Research*, 38(1), 63–74.
- Efraimidis, P. S., & Spirakis, P. G. (2006). Weighted random sampling with a reservoir. *Information Processing Letters*, 97(5), 181–185.
- Fang, C., Kolisch, R., Wang, L., & Mu, C. (2015). An estimation of distribution algorithm and new computational results for the stochastic resource-constrained project scheduling problem. *Flexible Services and Manufacturing Journal*, 27(4), 585–605.
- Fu, N., Lau, H. C., & Varakantham, P. (2015). Robust execution strategies for project scheduling with unreliable resources and stochastic durations. *Journal of Scheduling*, 18(6), 607–622.
- Gao, F., Shi, Z., Gao, S., & Xiao, H. (2019). Efficient simulation budget allocation for subset selection using regression metamodels. *Automatica*, 106(September), 192–200.
- Hajdu, M. (2013). Effects of the application of activity calendars on the distribution of project duration in PERT networks. *Automation in Construction*, 35, 397–404.
- Hao, X., Sun, L., & Gen, M. (2018). Solving robust resource constrained scheduling problem by multi-objective optimization method based on hybridization of EDA and GA. *Procedia Manufacturing*, 17, 1187–1194.
- Hedlund, H.E., & Mollaghasemi, M. (2001). A genetic algorithm and an indifference-zone ranking and selection framework for simulation optimization. In: *Proceeding of the 2001 Winter Simulation Conference* (Cat. No.01CH37304) (Vol. 1, pp. 417–421 vol.411).
- Herroelen, W., & Leus, R. (2005). Project scheduling under uncertainty: Survey and research potentials. *European Journal of Operational Research*, 165(2), 289–306.
- Huang, W., & Ding, L. (2011). Project-scheduling problem with random time-dependent activity duration times. *IEEE Transactions on Engineering Management*, 58(2), 377–387.
- Jarboui, B., Eddaly, M., & Siarry, P. (2009). An estimation of distribution algorithm for minimizing the total flowtime in permutation flowshop scheduling problems. *Computers and Operations Research*, 36(9), 2638–2646.
- Kennedy, K., Walsh, P., Mastaglio, T. W., & Scully, T. (2016). Genetic optimisation for a stochastic model for opportunistic maintenance planning of offshore wind farms. In: *2016 4th International Symposium on Environmental Friendly Energies and Applications (EFEA)* (pp. 1–6).
- Kerkhove, L. P., & Vanhoucke, M. (2017). Optimised scheduling for weather sensitive offshore construction projects. *Omega (United Kingdom)*, 66, 58–78.
- Kesur, K. B. (2009). Advances in genetic algorithm optimization of traffic signals. *Journal of Transportation Engineering*, 135(4), 160–173.
- Kolisch, R. (1996). Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation. *European Journal of Operational Research*, 90(2), 320–333.
- Kolisch, R., & Sprecher, A. (1997). PSPLIB - A project scheduling problem library. *European Journal of Operational Research*, 96(1), 205–216.
- Lamas, P., & Demeulemeester, E. (2016). A purely proactive scheduling procedure for the resource-constrained project scheduling problem with stochastic activity durations. *Journal of Scheduling*, 19(4), 409–428.
- Nazzal, D., Mollaghasemi, M., Hedlund, H., & Bozorgi, A. (2012). Using genetic algorithms and an indifference-zone ranking and selection procedure under common random numbers for simulation optimisation. *Journal of Simulation*, 6(1), 56–66.
- Rostami, S., Creemers, S., & Leus, R. (2018). New strategies for stochastic resource-constrained project scheduling. *Journal of Scheduling*, 21(3), 349–365.
- Salgado Duarte, Y., Szpytko, J., & del Castillo Serpa, A. M. (2020). Monte Carlo simulation model to coordinate the preventive maintenance scheduling of generating units in isolated distributed Power Systems. *Electric Power Systems Research*, 182, Article 106237.
- Shi, W., Chen, W. N., Lin, Y., Gu, T., Kwong, S., & Zhang, J. (2019). An adaptive estimation of distribution algorithm for multipolicy insurance investment planning. *IEEE Transactions on Evolutionary Computation*, 23(1), 1–14.
- Tao, S., Wu, C., Sheng, Z., & Wang, X. (2018). Stochastic project scheduling with hierarchical alternatives. *Applied Mathematical Modelling*, 58, 181–202.
- Tian, J., Hao, X., & Gen, M. (2019). A hybrid multi-objective EDA for robust resource constraint project scheduling with uncertainty. *Computers and Industrial Engineering*, 130(8), 317–326.
- Wang, L., & Fang, C. (2012). An effective estimation of distribution algorithm for the multi-mode resource-constrained project scheduling problem. *Computers and Operations Research*, 39(2), 449–460.
- Xu, X., Zhao, Y., & Chen, C.-Y. (2012). Project-driven supply chains: Integrating safety-stock and crashing decisions for recurrent projects. *Annals of Operations Research*, 241(1–2), 225–247.
- Yang, I. T. (2005). Impact of budget uncertainty on project time-cost tradeoff. *IEEE Transactions on Engineering Management*, 52(2), 167–174.
- Zhao, C., Ke, H., & Chen, Z. (2016). Uncertain resource-constrained project scheduling problem with net present value criterion. *Journal of Uncertainty Analysis and Applications*, 4(1), 12.