CrossMark

ORIGINAL ARTICLE

Resource-constrained project scheduling problem with uncertain durations and renewable resources

Weimin Ma[1] · Yangyang Che[1] · Hu Huang[1] · Hua Ke[1]

Abstract Resource-constrained project scheduling problem is to make a schedule for minimization of the makespan subject to precedence and resource constraints. In this paper, we consider an uncertain resource-constrained project scheduling problem (URCPSP) in which the activity durations, with no historical data generally, are estimated by experts. In order to deal with these estimations, an uncertainty-theory-based project scheduling model is proposed. Furthermore, a genetic algorithm integrating a 99-method based uncertain simulation is designed to search the quasi-optimal schedule. Numerical examples are also provided to illustrate the effectiveness of the model and the algorithm.

Keywords Resource-constrained project scheduling · Uncertain duration · Genetic algorithm · Uncertainty theory · Renewable resource

1 Introduction

The resource-constrained project scheduling problem (RCPSP) has been a widely studied optimization problem since it was initiated. The problem considers the minimization of the makespan while schedules the project subject to precedence and resource constraints [1]. As a job-shop generalization, RCPSP is NP-hard which was proved by Blazewicz et al. [2]. Most literature about RCPSP has based on the deterministic assumption that complete information of resource usage and deterministic activity duration are given in advance in which a feasible baseline schedule, i.e., a list of activity starting times and a definite makespan, can be determined as a result. During project execution, however, this simplification may lead to poor performance since project activities are usually subject to considerable disruptions such as unavailable resources, delayed delivery of materials, absent workers etc. [3]. Stochastic resource-constrained project scheduling problem (SRCPSP) has been developed to deal with the variability in activity durations [4–10]. In these researches, the execution of a project is viewed as a dynamic decision process in which activity durations are represented as random variables. Aiming at the minimization of the expected makespan, different scheduling policies are proposed to schedule the activities with random durations; these policies define which activities to start at each decision point (the start of project and the completion times of activities).

Considering the uniqueness of each project in real world, it is not uncommon that its activities are seldom or even never have been executed before. As a result, the activity durations cannot be depicted by probability distributions for the lack of historical data. In this case, belief degrees given by experienced project managers or experts can be employed to estimate distributions of the durations. Surveys have shown that these human beings' estimations generally possessed much wider range of values than the real ones [11]. Therefore, these indeterminacies cannot be treated as fuzziness [12, 13], probability [14, 15], roughness [16], ambiguity [17] or entropy [18]. Instead, uncertainty theory, initiated by Liu [19, 20], can be a useful tool to deal with these belief degrees. So far, the new theory has been successfully applied in various problems: option pricing problem [21, 22], production control problem [23],

✉ Hu Huang
huanghu0213@163.com

[1] School of Economics and Management, Tongji University, Shanghai 200092, China

inventory problem [24], assignment problem [25], supply chain pricing problem [26] etc. In project scheduling field, Zhang and Chen [27] built an uncertain model to minimize the expected project makespan under chance constraint of the total cost. Ji and Yao [28] considered an uncertain project scheduling problem to make a schedule for allocating the loans so that the total cost and the completion time of project are balanced. Ding and Zhu [29] considered two types of models for uncertain project scheduling problems to satisfy different management requirements. Based on genetic algorithm, Ke [30] proposed a hybrid approach to solve a project time-cost trade-off problem with uncertain measure, and also further studied this trade-off problem in uncertainty and randomness coexisting environment [23, 31].

However, no research deal with the RCPSP with uncertain durations and renewable resource constraints, this paper will fill the gap and explore this uncertain resource-constrained project scheduling problem. Specially, the activity durations whose distributions are estimated by professional managers or experts are characterized as uncertain variables. Furthermore, an uncertain serial project generation scheme (US-SGS) is introduced to produce a schedule from an activity list. An uncertain simulation based on 99-method is employed to approximate the uncertain functions, since the calculated makespan based on the expected value or medium number of each activity duration is usually less than the real one [6]. Afterwards, we design a GA-based algorithm to search for the quasi-optimal schedule.

The reminder of this paper organized as follows: In Sect. 2, formulations of the URCPSP are detailed. An uncertain expected value model is established and transformed into a crisp form in some special cases in Sect. 3. Furthermore, Sect. 4 proposes a hybrid algorithm integrating 99-method and genetic algorithm. Section 5 presents numerical examples to illustrate the effectiveness of the model and the algorithm. Relevant conclusions and some prospective extensions are discussed in Sect. 6. Specially, some basic concepts in uncertainty theory are introduced in appendix.

2 Problem description

The URCPSP considered in this paper is described as an activity-on-the-node network $G(N, A)$, where N is the set of activities and A is the set of pairs of activities between which a zero-lag finish-start precedence relationship exists. Dummy activities $i = 1$ and $i = N + 2$ respectively mark the beginning and the end of the project. The duration of each activity $\tilde{d}_i$ is assumed to be an uncertain variable

whose distribution is given by experts' or managers' belief degrees, and the resource requirement of resource k is indicated by r_{ik}. For renewable resource k, the availability R_k is a constant, which is constrained throughout the project. Each activity is executed in exclusive one mode and preemption is not allowed in this paper.

The execution of a project in uncertain context can be regarded as a dynamic decision process. At each decision time (the start of the project and the completion times of activities), the manager makes decisions about which activities to start subject to the precedences and resource constraints. Therefore, the solution of URCPSP is an activity order list which depicts the priority of each activity. Due to the uncertainty, the actual activity duration can only be attained after its entire processes completed, and meanwhile, the decision maker can merely apply partial information which is available before or at the time he makes the decision. So the schedule generation scheme (SGS) used in deterministic condition may not always be feasible. In this paper, we develop a new procedure denoted as US-SGS to decode the activity list λ, which works as the Serial SGS, but adds the side constraint

$$s_i(\lambda) \le \max_{\lambda^j < \lambda^i} \left(s_j(\lambda) \right) \tag{1}$$

where λ^i is the position of activity i in activity list λ and s_i is the starting time of activity i.

The US-SGS works as follows:

0: Given a network $G(N, A)$. λ:= activity list; λ_i:=activity in position i; d:= duration; A:= underway activity set; U:= unexecuted activity set; S:= starting time; F:= finish time; T:= Time point; num: number of the activities; K:=the number of renewable resources; r_{ik}: demand of activity i for resource type k; R_k:= available amount of resource i.

1: $A = \varnothing$, $U = [1, 2, \ldots, num]$, $F = [inf, inf, \ldots, inf]$, $S = [0, 0, \ldots, 0]$

2: $im = 1$, $T = 0$

3:   **While** is empty $(U) = 0$

4: $S_{\lambda_{im}} = \max_{i < im} S_{\lambda_i} \vee \max_{(j, \lambda_{im}) \in A} F_j$

5:   **if** $\sum_{i \in A}(r_{ik}) + r_{imk} \le R_k$, $k = 1, 2 \ldots K$ & $S_{\lambda_{im}} \le T$

6: $F(\lambda_{im}) = T + d(\lambda_{im})$

7:   Put λ_{im} into A

8: $im = im + 1$

9:   **else**

10: $T = \min_{i \in A}(F(i))$

11:   Take out i from A and U if: $F(i)_{i \in A} \le T$

12:   **end**

13: **end**

14: Return S and F

3 The expected value model

The URCPSP aims at minimizing the expected total project duration (makespan), equivalent to the start time of activity $N + 2$, denoted as s_{n+2}, since it is a dummy end. Therefore the makespan F_t is:

$$F_t = s_{n+2}$$

The mathematical model of the URCPSP can be established as follows:

$$\begin{cases} \min \ \mathrm{E}[F_t] \\ \text{subject to:} \\ s_i + d_i \le s_j, \ \forall(i,j) \in A \\ \sum_{i \in P_t} r_{ik} \le R_k, \ k = 1, 2 \ldots K, \ P_t = \{i \mid s_i \le t < s_i + d_i\} \end{cases}$$

In the above model, the objective function is to minimize expected makespan. The first constraint ensures that the earliest start time of activity j is forbidden to be smaller than the finish time of its predecessor activity i. The second constraint guarantees, for each time point t and resource type k, the resource demand should be below the resource limitation. Note that P_t represents the set of underway activities at time t, r_{ik} is the demand of activity i for resource type k, and R_k is the given available amount of resource k.

Because of the uncertain durations, the finish time of each activity is changeable and the makespan is an uncertain variable as well. Given the activity list λ, the finish time of each activity can be attained as follows:

$$\tilde{f}_i(\lambda) = s_i + \tilde{d}_i$$

The critical problem is to determine the starting time of activity i. If resource constraint is eliminated, the s_i, which is an uncertain variable as well, can be computed by

$$s_i(\lambda) = \max_{(j,i) \in A} \left(s_j(\lambda) + \widetilde{d}_j \right) \tag{2}$$

where λ is an activity list depicting the priority of each activity.

When resource constraint is considered, however, this formula is no longer valid. Since activity may be feasible in terms of precedence relation when all of its predecessors have been finished, while simultaneously infeasible for the lack of resource. In other words, one activity has predecessors in both precedence relation logic and resource logic. Besides, a side constraint is added to decode the activity list. To address this problem, similar with Artigues and Roubellat [32], we define an extended precedence relation set as $G(N, A^*)$ by adding extra resource flow network into the original activity-on-the-

node network $G(N, A)$. The resource flow network exists if there is a resource flow between two activities without precedence relations. Then the starting time of activity i, the function of activity list λ and uncertain durations, can be attained by

$$s_i(\lambda) = \max_{\lambda^j < \lambda^i} \left(s_j(\lambda) \right) \vee \max_{(j,i) \in A^*} \left(s_j(\lambda) + \widetilde{d}_j \right). \tag{3}$$

where λ^i is the position of activity i in activity list λ.

Given an activity list λ, one can attain that $s_i(\lambda)$ is monotone increasing function of $\widetilde{d}_j$ where $\lambda^j < \lambda^i$ or $(j, i) \in A^*$. Therefore, referring to Lemma 3, we have

Theorem 1 Suppose that the duration of activity i has regular distribution Φ_i and inverse uncertainty distribution $\Phi_i^{-1}(\alpha)$. Given the activity list λ, $s_i(\lambda)$ has an inverse uncertainty distribution $\Psi_i^{-1}(\lambda, \alpha) \ \alpha \in (0, 1]$, we can get

$$\Psi_i^{-1}(\lambda, \alpha) = \max_{\lambda^j < \lambda^i} \left(\Psi_j^{-1}(\lambda, \alpha) \right) \vee \max_{(j,i) \in A^*} \left(\Psi_j^{-1}(\lambda, \alpha) + \Phi_j^{-1}(\alpha) \right)$$

With Theorem 1, we can attain that the α-value of the makespan can be calculated by $\boldsymbol{d} = (\Phi_{d_1}^{-1}(\alpha), \Phi_{d_2}^{-1}(\alpha), \ldots, \Phi_{d_{n+2}}^{-1}(\alpha))$. Therefore, referring to Lemma 4, we have

Theorem 2 The expected makespan, depicting by the starting time of dummy end activity, can be attained as follows:

$$E[F_t] = \int_0^1 \Psi_{n+2}^{-1}(\lambda, \alpha) d\alpha$$

The equivalent form of the uncertain model can be transformed as follows:

$$\begin{cases} \min \ \int_0^1 \Psi_{n+2}^{-1}(\lambda, \alpha) d\alpha \\ \text{subject to:} \\ s_i + d_i \le s_j, \ \forall(i,j) \in A \\ \sum_{i \in P_t} r_{ik} \le R_k, \ k = 1, 2 \ldots K, \ P_t = \{i \mid s_i \le t < s_i + d_i\} \end{cases}$$

With the crisp-form of the expected model, 99-method can be applied to make approximation of the expected makespan.

$$\begin{cases} \min \ \sum_{m=1}^{99} \Psi_{n+2}^{-1}(\lambda, m/100)/99 \\ \text{subject to:} \\ s_i + d_i \le s_j, \ \forall(i,j) \in A \\ \sum_{i \in P_t} r_{ik} \le R_k, \ k = 1, 2 \ldots K, \ P_t = \{i \mid s_i \le t < s_i + d_i\} \end{cases}$$

616

Int. J. Mach. Learn. & Cyber. (2016) 7:613–621

4 GA-based algorithms

In RCPSP with uncertain durations, the expected duration or medium number is usually employed to replace the random or uncertain numbers and then the deterministic model is applied. However, the optimal solution attained from the deterministic model may perform very poor in real context. Meanwhile, the expected value of calculated makespan, based on the expected value of each activity duration, is often less than the real one. Stork [6] presented that the expected makespan is more than 4 % larger than the deterministic makespan with instances containing 20 activities. Moreover, coherently, Ballestin [7] revealed that this variation may extend to more than 10 % when the number of activities increases to 120.

Even just for precedence constrained jobs without any resource restrictions, the computation of the expected length of a critical path is NP-complete, if each activity processing time distribution has two (discrete) values. Similar to the researches in stochastic cases, uncertain simulation is essential for approximation of the expected makespan.

In order to solve URCPSP, a heuristic algorithm integrated by 99-method and GA is introduced in this section. The chromosome is denoted by the activity list λ in which activities are arranged in different orders on the premise that the precedence relationship is satisfied. Then, with a specific schedule generation scheme, a feasible solution can be gained. Besides, the most essential part of the algorithm is the process of decoding in accordance with different schedule generation schemes, which will be highlighted in this part.

The structure of the hybrid algorithm is as follows:

Step 1: Initialization

The first step is to randomly generate initial population with size *Popsize* as parent population. Individual creations rely on a list of chromosome representation: an activity order list (λ) in which activities are arranged randomly on the premise that the precedence relationship is satisfied.

Step 2: Crossover

Crossover is the main operation to produce the new chromosomes, simultaneously expanding the population. In this paper, we implement one point crossover. The crossover points are randomly generated as well as the randomly chosen mother and father chromosomes. Taking the position denoted by crossover point as divide, the left positions inherit exactly the same activity order from the mother. Subsequently, the remaining right positions are filled according to the father by left-to-right scan on the premise that the precedence relationship is satisfied, skipping the activities already taken from the mother.

Step 3: Mutation

Mutation is significant for the purpose of population diversity. For the activity order list, mutation begins with
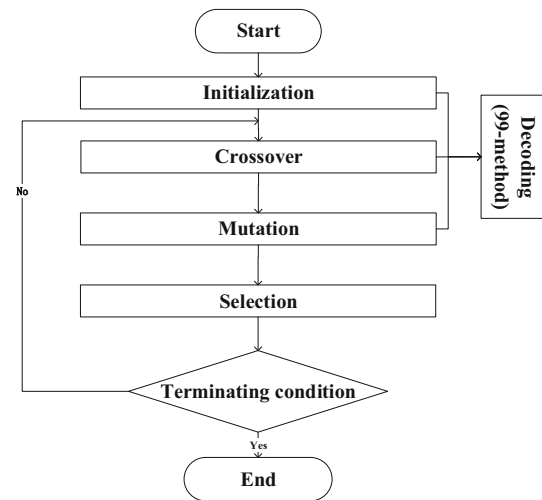

Fig. 1 Outline of the hybrid GA algorithm

randomly selecting the activity which will be mutated. Mutation applied to the activity order list is built to keep list precedence feasible. It determines the predecessor and successor of the selected activity first, taking their positions as bound, and then randomly chooses and inserts in a position in the range to generate a feasible child chromosome.

Step 4: Decoding

In this step, we employ the US-SGS to decode the activity list λ and approximate the fitness of each chromosome by uncertain simulation based on 99-method, initiated by [20] and successfully applied in project scheduling by [27].

The uncertain simulation based on 99-method for $E[F_t]$ is presented in Algorithm 1.

Algorithm 1 (Uncertain simulation for $E[F_t]$)

1: Given an activity list λ;

2: Generate $d_1^m, d_2^m, \ldots, d_n^m$ according to the inverse uncertainty distributions of activities' durations $\Phi_{d_1}^{-1}$, $\Phi_{d_2}^{-1}, \ldots, \Phi_{d_n}^{-1}$, and denote $\boldsymbol{d}^m = (\Phi_{d_1}^{-1}(m/100), \Phi_{d_2}^{-1}(m/100), \ldots, \Phi_{d_n}^{-1}(m/100))$, $m = 1, 2, \ldots, 99$, respectively;

3: Decode the activity list λ by using US-GSG for each $\boldsymbol{d}^m$, $m = 1, 2, \ldots, 99$. Then the start time $\boldsymbol{s}^m$ and finish time $\boldsymbol{f}^m$ can be attained;

4: $f_t^m = s_n^m$, $m = 1, 2, \ldots, 99$;

5: $E[F_t] = \sum\limits_{m=1}^{99} f_t^m / 99$.

Step 5: Selection

With the purpose of keeping the population size consistent, new parent population requires to be selected from the combined population. This paper employs the genetic roulette wheel selection, in which individuals with better fitness value are awarded with larger probability to be selected in the roulette wheel. Additionally, the top-5

Table 1 The information of the test instances

Activity	Duration	Expected duration	R_1	R_2	R_3	R_4	Successors
1	0	0	0	0	0	0	2, 3, 4
2	$\mathscr{Z}(5,8,10)$	7.75	4	0	0	0	6, 11, 15
3	$\mathscr{L}(3,5)$	4	10	0	0	0	7, 8, 13
4	$\mathscr{Z}(4,6,7)$	5.75	0	0	0	3	5, 9, 10
5	$\mathscr{L}(2,4)$	3	3	0	0	0	20
6	$\mathscr{Z}(6,8,9)$	7.75	0	0	0	8	30
7	$\mathscr{Z}(3,5,6)$	4.75	4	0	0	0	27
8	$\mathscr{Z}(7,9,13)$	9.5	0	1	0	0	12, 19, 27
9	$\mathscr{L}(1,3)$	2	6	0	0	0	14
10	$\mathscr{Z}(6,7,9)$	7.25	0	0	0	1	16, 25
11	$\mathscr{Z}(5,9,11)$	8.5	0	5	0	0	20, 26
12	$\mathscr{L}(7,9)$	8	0	7	0	0	14
13	$\mathscr{Z}(4,6,7)$	5.75	4	0	0	0	17, 18
14	$\mathscr{Z}(2,3,6)$	3.5	0	8	0	0	17
15	$\mathscr{Z}(6,10,12)$	9.50	3	0	0	0	25
16	$\mathscr{Z}(7,11,12)$	10.25	0	0	0	5	21, 22
17	$\mathscr{Z}(3,6,7)$	5.5	0	0	0	8	22
18	$\mathscr{Z}(3,5,6)$	4.75	0	0	0	7	20, 22
19	$\mathscr{Z}(1,3,4)$	2.75	0	1	0	0	24, 29
20	$\mathscr{Z}(5,7,8)$	6.75	0	10	0	0	23, 25
21	$\mathscr{Z}(1,2,4)$	2.25	0	0	0	6	28
22	$\mathscr{Z}(5,7,8)$	6.75	2	0	0	0	23
23	$\mathscr{Z}(2,3,5)$	3.25	3	0	0	0	24
24	$\mathscr{Z}(11,13,14)$	12.75	0	9	0	0	30
25	$\mathscr{Z}(1,3,4)$	2.75	4	0	0	0	30
26	$\mathscr{Z}(5,7,8)$	6.75	0	0	4	0	31
27	$\mathscr{Z}(6,8,11)$	8.25	0	0	0	7	28
28	$\mathscr{Z}(1,3,4)$	2.75	0	8	0	0	31
29	$\mathscr{Z}(5,7,8)$	6.75	0	7	0	0	32
30	$\mathscr{Z}(10,12,13)$	11.75	0	7	0	0	32
31	$\mathscr{Z}(10,12,13)$	11.75	0	0	2	0	32
32	0	0	0	0	0	0	

1, 32 are dummy activity, the limits of the four resource are (12, 13, 4, 12)

best-to-now chromosomes are specifically preserved in the population. As a consequence, selection process guarantees the high quality of population, and meanwhile avoids the local optimum.

The framework of the algorithm is presented in Fig. 1.

5 Numerical examples

In this section, we consider a project scheduling problem containing 30 activities (J30) and 4 renewable resources. Specially, the durations are estimated by experts and characterized as uncertain variables. Interested readers can consult [11] for more details about how to use experts' belief degrees to estimate uncertain distributions.

Information of the numerical instance including durations, resource utilizations and successor activities is presented in Table 1, and correspondingly, the structure of the project is provided in Fig. 2.

To minimize the expected makespan, the following model can be applied.

$$
\begin{cases}
\min \ \mathrm{E}[F_{32}] \\
\text{subject to:} \\
s_i + d_i \leq s_j, \ \forall (i,j) \in A \\
\sum_{i \in P_t} r_{ik} \leq R_k, \ k = 1, 2 \ldots 4, \quad P_t = \{i \mid s_i \leq t < s_i + d_i\}
\end{cases}
$$

The algorithm is coded by MATLAB and runs on a PC with core i3 with 2.3GHz clock speed and 2G RAM. The

Fig. 2 The activity-on-the-node network of the project
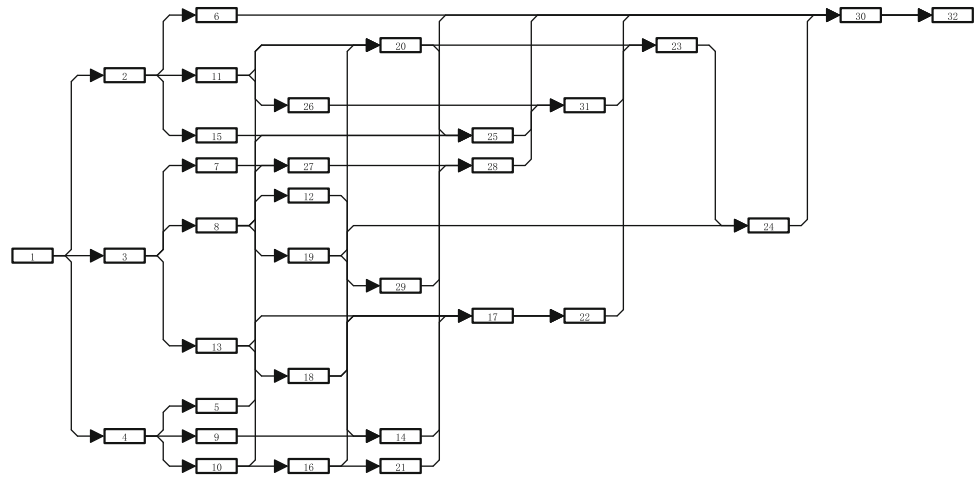
Table 2 The top 10 quasi-optimal schedules

Seq	Popsize	p_c	p_m	Quasi-optimal schedules	Expected makespan
1	20	0.8	0.4	1,3,4,2,8,10,13,7,18,11,12,5,16,9,27,14,15,19,20,26,17,21,22,29,6,25,23,28,31,24,30,32	65.9300
2	30	0.7	0.4	1,3,4,8,7,2,13,10,9,18,11,16,19,12,27,26,5,14,21,15,20,17,22,6,28,31,29,23,24,25,30,32	66.1130
3	20	0.8	0.3	1,4,3,8,2,13,5,10,18,7,11,16,12,9,27,26,14,21,19,15,20,17,22,28,29,31,25,23,24,6,30,32	66.1130
4	20	0.8	0.4	1,3,4,13,8,2,10,18,7,11,16,12,27,9,26,14,5,21,15,19,20,17,22,28,31,6,29,23,24,25,30,32	66.1130
5	30	0.7	0.4	1,4,3,2,8,7,10,13,18,11,19,12,5,16,9,27,14,15,20,17,22,29,21,26,23,28,31,6,24,25,30,32	66.1440
6	30	0.7	0.4	1,3,4,2,8,10,13,7,18,9,5,16,12,11,27,14,15,20,19,17,22,29,25,21,26,23,28,31,24,6,30,32	66.1440
7	30	0.6	0.3	1,4,3,7,2,13,8,10,5,11,18,16,19,12,9,27,14,15,20,17,22,21,29,26,23,25,28,31,24,6,30,32	66.1500
8	20	0.8	0.3	1,3,4,8,2,13,10,11,7,18,12,16,27,9,14,5,15,19,20,17,22,25,26,29,21,23,28,31,6,24,30,32	66.1500
9	30	0.6	0.2	1,4,3,13,2,8,10,18,7,11,16,9,12,5,27,14,15,21,20,26,17,19,22,28,6,31,29,23,25,24,30,32	66.2120
10	30	0.6	0.2	1,3,4,2,13,8,5,10,7,18,11,16,12,27,9,14,15,21,20,17,26,19,22,28,31,29,23,6,24,25,30,32	66.2120

Table 3 The effectiveness of the GA

Gen	Popsize	p_c	p_m	The best	The worst	Average	Error (%)
1000	30	0.6	0.2	66.212	68.019	66.7959	2.73
1000	30	0.6	0.3	66.150	69.415	67.5808	4.94
1000	30	0.7	0.4	66.113	69.160	66.5831	4.61
1000	20	0.7	0.2	66.445	68.458	67.2897	3.03
1000	20	0.8	0.3	66.113	67.890	66.9114	2.69
1000	20	0.8	0.4	65.925	67.793	66.7275	2.83

$$\text{Error } (\%) = \frac{The\ worst - The\ best}{The\ best} * 100\%$$

quasi-optimal schedules attained from experiments are presented in Table 2, resulting from 10 times' run of the hybrid intelligent algorithm with 1000 cycles for the expected models under different parameter settings.

Similar to [33, 34], the quality of solution can be obtained via the deviation of the worst solution from the optimal solution attained in the 10 times' run and the results are presented in Table 3. The results reveal that the

deviation denoted by percent error does not exceed 5%, which implies the effectiveness of the algorithm integrating uncertain simulations.

Additionally, some other computational experiments are presented. Our test instances are all come from the benchmark library PSPLIB http://www.om-db.wi.tum.de/psplib/data.html initiated by [35] in which different characteristics of the deterministic RCPSP are presented. Since

Table 4 The results of the computational experiments

Instances	Gen	Popsize	p_c	p_m	The best	The worst	Average	Error (%)
J_{301_1}	1000	30	0.6	0.2	46.929	49.242	48.197	4.93
J_{601_1}	1000	30	0.6	0.2	79.348	84.843	82.303	6.93
J_{901_1}	1000	30	0.6	0.2	91.091	97.465	94.559	7.00
J_{1201_1}	1000	30	0.6	0.2	141.130	153.062	147.420	8.53

$$\text{Error } (\%) = \frac{The\ worst - The\ best}{The\ best} * 100\%$$

the uncertain simulation is the time-consuming task, this paper only considers four instances with 30, 60, 90 and 120 activities under 4 renewable resource constraints. The uncertain distributions of the activity processing times are generated by $\mathscr{L}(d_i - \sqrt{d_i}, d_i, d_i + \sqrt{d_i})$ with the deterministic d_i as expectation. After running each instance 10 times, the results are presented in Table 4. Referring to Table 4, we can find that the deviation denoted by percent error does not exceed 10%.

6 Conclusion

In this paper, an uncertain model was proposed to deal with uncertain resource constrained project scheduling problem (URCPSP), in which the durations, with no historical data generally, were described as uncertain variables. Furthermore, an uncertain serial schedule generation scheme (US-SGS) was introduced to process the activities with uncertain durations. Besides, a GA-based hybrid algorithm integrating uncertain simulation was designed to search the quasi-optimal schedule. Numerical examples were provided to illustrate the effectiveness of model and the algorithm as well.

Meanwhile, there are some limitations in this paper. First of all, this paper involves merely one type of indeterminacy, while the real world project might be complicated that randomness (activity duration estimated by historical data) and uncertainty (activity duration estimated by experts' experienced data) might co-exist. Future research can extend the RCPSP to twofold indeterminacy and correspondingly uncertain random variable can be applied. Secondly, this paper assumed that each activity was executed with one mode, multi-mode RCPSP in uncertain environment can be taken into consideration in further development.

Appendix

In this section, some concepts and theorems of uncertainty theory are introduced to lay the foundation for the URCPSP modeling. Uncertainty theory is a branch of axiomatic mathematics for subjective uncertainty modeling, which has been well developed and applied in a wide variety of real problems.

Let Γ be a nonempty set, $\mathscr{L}$ a σ-algebra over Γ, and each element Λ in $\mathscr{L}$ is called an event. Uncertain measure is defined as a function from $\mathscr{L}$ to [0, 1]. In detail, the concept of uncertain measure is pioneered by [19] and redefined by [20]. Uncertain measure $\mathscr{M}$ is a set function defined over the following four axioms.

Axiom 1 (Normality Axiom) $\mathscr{M}\{\Gamma\} = 1$.

Axiom 2 (Duality Axiom) $\mathscr{M}\{\Lambda\} + \mathscr{M}\{\Lambda^c\} = 1$ for any event Λ.

Axiom 3 (Subadditivity Axiom) For every countable sequence of events $\{\Lambda_i\}$, we have:

$$\mathscr{M}\left\{\bigcup_{i=1}^{\infty} \Lambda_i\right\} \leq \sum_{i=1}^{\infty} M\{\Lambda_i\}.$$

Axiom 4 (Product Measure Axiom) Let $(\Gamma_k, \mathscr{L}_k, \mathscr{M}_k)$ be uncertainty spaces for $k = 1, 2, \ldots$, the product uncertain measure $\mathscr{M}$ is an uncertain measure satisfying

$$\mathscr{M}\left\{\prod_{k=1}^{\infty} A_k\right\} = \bigwedge_{k=1}^{\infty} \mathscr{M}_k\{A_k\}$$

where A_k are arbitrarily chosen events from $\mathscr{L}_k$ for $k = 1, 2, \ldots$, respectively.

Definition 1 [19] An uncertain variable is a measurable function ξ from an uncertainty space $(\Gamma, \mathscr{L}, \mathscr{M})$ to the set of real numbers, i.e., for any Borel set B of real numbers, the set

$$\{\xi \in B\} = \{\gamma \in \Gamma \mid \xi(\gamma) \in B\}$$

is an event.

The uncertainty distribution is indispensable to establish the practical uncertain optimization problems model.

620

Int. J. Mach. Learn. & Cyber. (2016) 7:613–621

Definition 2 [19] The uncertainty distribution Φ of an uncertain variable ξ is defined by

$$\Phi(x) = \mathcal{M}\{\xi \leq x\}$$

for any real number x.

An uncertainty distribution Φ is confirmed to be regular if its inverse function $\Phi^{-1}(\alpha)$ exists uniquely for each $\alpha \in [0, 1]$.

Definition 3 [19] Let ξ be an uncertain variable. The expected value of ξ is defined by

$$E[\xi] = \int_0^{+\infty} \mathcal{M}\{\xi \geq r\}\mathrm{d}r - \int_{-\infty}^0 \mathcal{M}\{\xi \leq r\}\mathrm{d}r$$

provided that at least one of the above two integrals is finite.

Lemma 1 [20] Let ξ be an uncertain variable with uncertainty distribution Φ. If the expected value exists, then

$$E[\xi] = \int_0^{+\infty} (1 - \Phi(x))\mathrm{d}x - \int_{-\infty}^0 \Phi(x)\mathrm{d}x.$$

Lemma 2 ([20]) Let ξ be an uncertain variable with regular uncertainty distribution Φ. If the expected value exists, then

$$E[\xi] = \int_0^1 \Phi^{-1}(\alpha)d\alpha.$$

Example 1 The expected value of the zigzag uncertain variable $\xi = \mathcal{Z}(a, b, c)$ is:

$$E[\xi] = \int_0^{0.5} ((1 - 2\alpha)a + 2\alpha b)\mathrm{d}\alpha + \int_{0.5}^1 ((2 - 2\alpha)b \\ + (1 - 2\alpha)c)\mathrm{d}\alpha = \frac{a + 2b + c}{4}. \tag{4}$$

Lemma 3 [20] Let $\xi_1, \xi_2, \ldots, \xi_n$ be independent uncertain variables with regular uncertainty distributions $\Phi_1, \Phi_2, \ldots, \Phi_n$, respectively. A function $f(x_1, x_2, \ldots, x_n)$ which is strictly increasing with respect to $x_1, x_2, \ldots, x_m$ and strictly decreasing with respect to $x_{m+1}, x_{m+2}, \ldots, x_n$. Then the $\xi = f(\xi_1, \xi_2, \ldots, \xi_n)$ is an uncertain variable with inverse uncertainty distribution

$$\Psi^{-1}(\alpha) = f(\Phi_1^{-1}(\alpha), \ldots, \Phi_m^{-1}(\alpha), \Phi_{m+1}^{-1}(1 - \alpha), \ldots, \\ \times \Phi_n^{-1}(1 - \alpha)). \tag{5}$$

Lemma 4 [36] Let $\xi_i(i = 1, 2, ..., n)$ be independent uncertain variables with regular uncertainty distributions $\Phi_1, \Phi_2, \ldots, \Phi_n$ respectively. If a function $f(x_1, x_2, \ldots, x_n)$ is strictly increasing with respect to $x_1, x_2, \ldots, x_m$ and strictly decreasing with respect to $x_{m+1}, x_{m+2}, \ldots, x_n$, respectively, then the expected value of uncertain variables $\xi = f(\xi_1, \xi_2, \ldots, \xi_n)$ has an expected value as follows:

$$E[\xi] = \int_0^1 f(\Phi_1^{-1}(\alpha), \ldots, \Phi_m^{-1}(\alpha), \Phi_{m+1}^{-1}(1 - \alpha), \Phi_n^{-1}(1 - \alpha))\mathrm{d}\alpha \tag{6}$$

provided that the expected value $E[\xi]$ exists.

References

1. Herroelen W, De Reyck B, Demeulemeester E (1998) Resource-constrained project scheduling: a survey of recent developments. Comput Oper Res 25(4):279–302
2. Blazewicz J, Lenstra JK, Kan A (1983) Scheduling subject to resource constraints: classification and complexity. Discret Appl Math 5(1):11–24
3. Herroelen W, Leus R (2005) Project scheduling under uncertainty: survey and research potentials. Eur J Oper Res 165(2): 289–306
4. Igelmund G, Radermacher FJ (1983) Preselective strategies for the optimization of stochastic project networks under resource constraints. Networks 13(1):1–28
5. Tsai YW , Gemmill DD (1998) Using tabu search to schedule activities of stochastic resource-constrained projects. Eur J Oper Res 111(1):129–141
6. Stork F (2000) Branch-and-bound algorithms for stochastic resource-constrained project scheduling. Technical Rep 702–2000
7. Ballestín F (2007) When it is worthwhile to work with the stochastic rcpsp? J Sched 10(3):153–166
8. Ballestin F, Leus R (2009) Resource-constrained project scheduling for timely project completion with stochastic activity durations. Prod Oper Manag 18(4):459–474
9. Ashtiani B, Leus R, Aryanezhad MB (2011) New competitive results for the stochastic resource-constrained project scheduling problem: exploring the benefits of pre-processing. J Sched 14(2):157–171
10. Creemers S (2015) Minimizing the expected makespan of a project with stochastic activity durations under resource constraints. J Sched 1–11
11. Liu B (2015) Uncertainty theory. Springer-Verlag, Berlin
12. Ke H, Liu B (2010) Fuzzy project scheduling problem and its hybrid intelligent algorithm. Appl Math Model 34(2):301–308
13. Wang X, Xing H, Li Y, Hua Q, Dong C, Pedrycz W (2015) A study on relationship between generalization abilities and fuzziness of base classifiers in ensemble learning. Fuzzy Syst IEEE Trans 23(5):1638–1654. doi:10.1109/TFUZZ.2014.2371479
14. Ke H, Liu B (2005) Project scheduling problem with stochastic activity duration times. Appl Math Comput 168(1):342–353
15. Wang XZ, He YL, Wang DD (2014) Non-naive bayesian classifiers for classification problems with continuous attributes. Cybern IEEE Trans 44(1):21–39
16. Sun B, Ma W (2015) Rough approximation of a preference relation by multi-decision dominance for a multi-agent conflict analysis problem. Inf Sci 315:39–53
17. Wang XZ, Dong LC, Yan JH (2012) Maximum ambiguity-based sample selection in fuzzy decision tree induction. Knowl Data Eng IEEE Trans 24(8):1491–1505
18. Wang XZ, Dong CR (2009) Improving generalization of fuzzy if-then rules by maximizing fuzzy entropy. Fuzzy Syst IEEE Trans 17(3):556–567
19. Liu B (2007) Uncertainty Theory. Springer-Verlag, Berlin
20. Liu B (2010) Uncertainty Theory. Springer-Verlag, Berlin
21. Chen X (2011) American option pricing formula for uncertain financial market. Int J Oper Res 8(2):32–37

22. Chen X (2012) Variation analysis of uncertain stationary independent increment processes. Eur J Oper Res 222(2):312–316
23. Ke H., Su T., Ni Y (2014) Uncertain random multilevel programming with application to production control problem. Soft Comput 1–8. doi:10.1007/s00500-014-1361-2
24. Qin Z, Kar S (2013) Single-period inventory problem under uncertain environment. Appl Math Comput 219(18):9630–9638
25. Zhang B, Peng J (2013) Uncertain programming model for uncertain optimal assignment problem. Appl Math Model 37(9):6458–6468
26. Huang H, Ke H (2014) Pricing decision problem for substitutable products based on uncertainty theory. J Intell Manuf. doi:10.1007/s10845-014-0991-7
27. Zhang X, Chen X (2012) A new uncertain programming model for project scheduling problem. Inf Int Interdiscip J 15(10): 3901–3910
28. Ji X, Yao K (2014) Uncertain project scheduling problem with resource constraints. J Intell Manuf 1–6. doi:10.1007/s10845-014-0980-x
29. Ding C, Zhu Y (2014) Two empirical uncertain models for project scheduling problem. J Oper Res Soc. doi:10.1057/jors.2014.115
30. Ke H (2014) A genetic algorithm-based optimizing approach for project time-cost trade-off with uncertain measure. J Uncertain Anal Appl 2(8)
31. Ke H, Liu H, Tian G (2015) An uncertain random programming model for project scheduling problem. Int J Intell Syst 30(1):66–79
32. Artigues C, Roubellat F (2000) A polynomial activity insertion algorithm in a multi-resource schedule with cumulative constraints and multiple modes. Eur J Oper Res 127(2):297–316
33. Ke H, Liu B (2007) Project scheduling problem with mixed uncertainty of randomness and fuzziness. Eur J Oper Res 183(1):135–147
34. Zhu J, Li X, Shen W (2011) Effective genetic algorithm for resource-constrained project scheduling with limited preemptions. Int J Mach Learn Cybern 2(2):55–65
35. Kolisch R, Sprecher A (1997) Psplib-a project scheduling problem library: Or software-orsep operations research software exchange program. Eur J Oper Res 96(1):205–216
36. Liu Y, Ha M (2010) Expected value of function of uncertain variables. J Uncert Syst 4(3):181–186