International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 10 Issue 6 ǁ 2022 ǁ PP. 99-107

Resource-constrained project-scheduling optimization with overlap for complex products and systems

Jing YU, Jingjing ZHAO, Xinyu LI
(School of Management, Tianjin University of Technology, Tianjin 300384)

Abstract
Activity overlap is commonly adopted as an effective measure for reducing the duration of complex products and systems. The resource-constrained project-scheduling problem with overlap is an extension of the classical problem. This paper studies a resource-constrained project-scheduling problem with overlap for complex products and systems. First, rework on downstream activity resulting from overlap is assumed under uncertainty, and a binomial probability model is built. Second, an optimized model is built with the objective of minimizing the duration, and a parallel schedule generation scheme is designed using a decoding operator to complete a genetic algorithm. Finally, the algorithm is tested on a data set. The duration of the project is analyzed with changes in the overlap and rework parameters. The results show that the greater the number of overlapping activities, the greater the reduction in the duration of the project. In addition, with increasing downstream activity overlap or a decreasing rework probability, the reduction effect on the duration increases. The value of this research is its application of resource-constrained project-scheduling optimization with overlap to the complex products and systems. The results of this research can serve as a guide for project managers in their decision-making around complex engineering projects.

Keywords *Project scheduling; genetic algorithm; overlap; rework; binomial probability model*

Date of Submission: 20-05-2022 Date of acceptance: 03-06-2022

I. Introduction

Research on new technology, methods and materials is applied in the process of complex products and systems. This process usually faces technical difficulties, innovations and other challenges, accompanied by a variety of risks and uncertainties, and the duration of the project is subsequently extended. The duration determines the speed at which an enterprise can respond to competitors and technological progress. Decreasing the project duration to the extent possible is a primary consideration in establishing a project plan and the process of project execution (Liet al., 2018).

The overlap execution mode is an effective approach for decreasing the project duration (Chu et al., 2019). Many activities contain technical precedence constraints, meaning that these activities cannot be executed concurrently, but overlap execution can decrease the project duration to a certain extent. For several overlapping activities, the downstream activity begins before the upstream activity is finished. When the upstream activity is completed, the downstream activity likely must be corrected based on the upstream work. This situation results in additional rework time and extends the execution time(Zhang et al.,2018). In general, the reduction in project duration due to overlap is greater than the duration of additional rework resulting from activity corrections. Thus, although the decrease in project duration due to overlapping activities occurs at the expense of rework costs, the overlap is still worthwhile(Lin et al.,2009).

The classical resource-constrained project-scheduling problem (RCPSP) focuses on obtaining a baseline schedule that satisfies the priority relationships and resource constraints by optimizing a goal (duration, cost, resource, etc.) under a deterministic environment and complete information(Hartmann & Briskorn, 2010). The classical RCPSP has been proven as a serious NP-hard problem (Blazewicz et al.,1983). When no overlap occurs between activities (i.e., the overlap time is zero), the RCPSP with overlap becomes the classic RCPSP. Thus, the RCPSP with overlap can be viewed as an extension of the classical RCPSP. Consequently, the RCPSP with overlap is also a serious NP-hard problem.

Use of the design structure matrix (DSM) is a common method for building an overlap model. Cho & Eppinger (2005) estimated the duration and cost model for complex product development projects based on DSM simulation. Li & Xu (2012) established a simulation model for a complex product development process that considered the conflicts of random overlap and resources. Maheswari & Varghese (2005) used the DSM time factor matrix to establish an overlap quantization model and estimated the project duration. Bai &Wan (2008) considered the quantitative effects of overlapping activities on the project duration, which was calculated based on the time factor matrix. Yang & Huang (2011) used the information output and input time factors to describe sequence overlap and feedback overlap, and a genetic algorithm was used to minimize the duration of the project. Zhang & Yan (2013) used the DSM to describe the overlap and proposed the use of the mixed particle swarm algorithm to optimize the duration of the project. In addition to the use of the DSM as a tool, the optimization model is a problem with overlap. Gerk & Qassim (2008) considered three acceleration tools, i.e., crashing, overlapping and substitution. A mixed integer nonlinear model was used to determine the optimal allocation of the three tools. Lin et al. (2009)noted that overlap can reduce the project duration at the expense of rework and communication. Lin et al. (2010)proposed an analysis model that weighed the positive and negative effects caused by overlap and determined the optimal communication strategy and overlap level. Wang & Lin (2009) established an overlap process model to analyze process structures based on the project duration. The process structures indicated the activity execution orderand the degree of overlap between activities.

Recently, scholars have begun to research project-scheduling issues that overlap. Berthaut et al. (2011a) introduced an overlap model to RCPSPs in which the activity overlap mode and the activity execution pattern were analyzed, and a nonlinear 0-1 integer programming model was established. Berthaut et al. (2011b) introduced rework and communication costs into the overlap model, and an overlap model considering the trade-off between time and cost was established. A nonlinear 0-1 integer programming model was transformed into a linear 0-1 integer programming model. Greze et al. (2014) established a model for the RCPSP with overlap, and a heuristic algorithm was proposed to solve this problem with the conclusion that use of the maximum benefit as a goal can more effectively avoid losses through a series of analyses. Bozejko et al. (2014) used an overlap mode to achieve the minimum cost of a target in road construction scheduling. Koyuncu et al. (2015) used the particle swarm optimization algorithm to solve the project-scheduling problem with overlap. Yu et al. (2015) integrated overlapping activities into the RCPSP. The overlap mode of the activities was divided into natural and forced categories, and the DSM was used to research the impact of time and resources on overlap. To minimize the duration of the project, the model of the RCPSP with overlap was solved using a genetic algorithm. Chu et al.(2019) study a resource-constrained project scheduling problem with multiple overlapping modes, which is NP-hard. To obtain high-quality near-optimal solutions, they formulate it as a dynamic program (DP), and develop a rollout policy based approximate dynamic programming (ADP) algorithm to obtain near-optimal policy efficiently.

From the above research and analysis, we observe that the research on overlapping project scheduling has gradually attracted the attention of scholars, but the current research has certain limitations. First, an in-depth study of overlap description and modeling is lacking, and an overlap analysis of the impact of project scheduling is not sufficient. For example, in the overlap description model, most studies assume that a simple linear relationship exists between overlap time and rework time. However, during the actual execution of a project, the rework time on downstream activity due to overlap is uncertain. Therefore, it is necessary to establish a probability distribution model of rework time. Second, the scheduling model constraints include not only priority constraints and resource constraints but also the overlap, downstream and rework constraints of downstream activities. Because of these constraints, the complexity of the problem is increased . In recent years, some scholars have studied only the 0-1 integer programming model and the time/cost trade-off model and have proposed certain in efficient heuristic algorithms. In addition, in the experimental design, the existing research lacks an analysis of the impact and problem parameters on the algorithm based on a large number of instances.

This paper studies the problem of project scheduling with overlap. The main contributions are listed as follows: (1) The impact of overlap on project scheduling is analyzed in depth, and a detailed description of overlap with uncertainty is presented. (2)An overlap optimization scheduling model is established, and a genetic algorithm is designed to solve the model.(3) Based on a certain number of test instances, the validity of the model and the algorithm is verified. The effects of overlap and rework parameters on the duration are analyzed. Here, the current project scheduling needs of engineering enterprises are fully considered. Depending on the cooperation between the research team and the engineering enterprises, we apply the research results to the practical complex engineering scheduling problems, and the improvement and expansion of the theoretical results can be achieved. Therefore, our research results will improve the management efficiency of complex engineering projects and provide decision-making basis and theoretical guidance for project managers.

The remainder of the paper is organized as follows. Section 2 establishes a bivariate distribution probability model for the rework time of downstream activities and constructs a resource optimization project-scheduling optimization model with overlap. Section 3 presents a simulation-based genetic algorithm and uses an overlap parallelism generation mechanism as the decoding method. Section 4 solves the designed computational experiments and analyzes the performance of the algorithm. The conclusions of this paper are summarized in the final section.

II. Problem statement

2.1 A binomial probability model for rework time

In an engineering project, scheduling is conducted on a periodic basis (i.e., hour, day, week), and resource availabilities and allocations are estimated per period. As a result, we assume that all types of activities (normal, overlap, and rework) consume the same resources per unit of time.

For the scheduling problem with overlap, most studies consider a simple linear relationship between overlap time and rework time. However, during practical execution of a project, rework and rework time are affected by the amount of information released upstream to the downstream activities at the information release points. The amount of information that an activity can release at an informational point is affected by many uncertainties, e.g., the overall progress of the project and the progress of upstream activities. Therefore, the rework time in downstream activity might be uncertain. Based on the above analysis, it is necessary to establish a probability distribution model for rework time.

In this section, we assume that Δ denotes the overlap time and σ denotes the rework time on the downstream activity such that the overlap time can be divided into Δ periods, which equals Δ times in independent tests. In each test, the overlap can cause rework or no rework for the downstream activity. Therefore, the amount of rework in the test follows a binomial distribution. The probability that rework occurs is represented by β. We find that the rework time σ follows the binomial distribution $C_0^{\sigma}\beta^{\sigma}(1-\beta)^{\Delta-\sigma}$ ($\sigma = 0,1,2,...,\Delta$), denoted by $\sigma \sim B(\Delta,\beta)$.

For example, assuming that the overlap time is 5 units and that the downstream activity rework probability is 0.4,the rework time has the following six possibilities: 0, 1, 2, 3, 4, 5. The probability distribution is presented in Fig. 1.

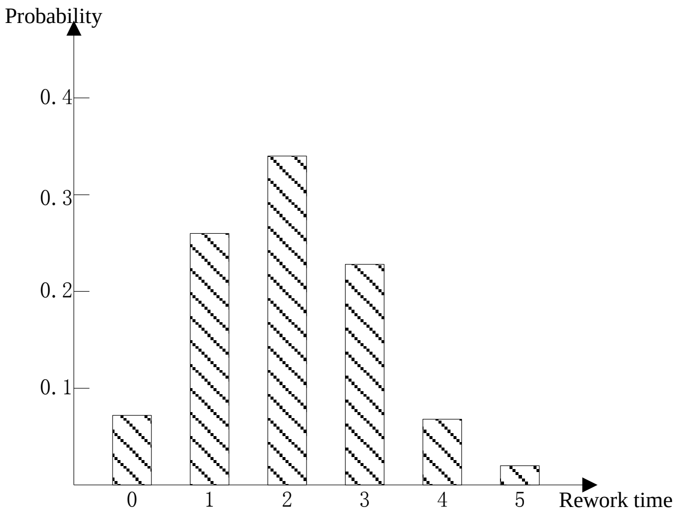

Fig.1 Probability distribution ($\Delta = 5$, $\beta = 0.4$)

2.2 Project-scheduling optimization model

In the model, the project is represented by an AoN (activity-on-node) network and is not interrupted once the activity begins. The rework time on downstream activity is expressed using the binomial distribution probability model. Because the rework time is uncertain, the objective function is the expected value of the minimum duration of the project. The corresponding project-scheduling optimization model is given as follows:

$$\text{Min } E(ST_{N+1}) \tag{1}$$

$$\text{s.t. } ST_j \geq ST_i + d_i, \quad (i,j) \in A_n \tag{2}$$

$$ST_j \geq ST_i + d_i - o_{ij}, \quad (i,j) \in A_o \tag{3}$$

$$ST_j + d_j > ST_i + d_i, \quad (i,j) \in A_o \tag{4}$$

$$\sum_{i \in S_t} r_{ik} \leq R_k, \forall k, t \tag{5}$$

In objective (1), ST_{N+1} is the completion time of the fictitious activity N+1, which denotes the completion time of the entire project. Constraint (2) represents the precedence constraints when activities are not overlapped. If activities are overlapped, constraint (3) states that the downstream activity must start after the upstream activity releases information. Constraint (4) ensures that downstream activities cannot be completed until the upstream activities are completed such that downstream activities can receive the final information published by the upstream activities. Constraint (5) defines the resource constraints.

III. Genetic algorithm based on simulation

The genetic algorithm is a common heuristic algorithm based on biological genetic and evolutionary theory that searches for the best individual (Debels et al., 2007; Hartmann, 1998, 2000). A simulation optimization algorithm based on the genetic algorithm is proposed, and a special decoding procedure is designed to obtain the schedule. For each individual, the fitness value is calculated via simulation, which is the expected value of the duration. The framework of the genetic algorithm is shown in Algorithm 1.

Algorithm1.Framework of the genetic algorithm

```
Encoding();
Generate_initial_population();
Decoding();
Calculate_fitness();
While not terminating condition
        Selection_Population();
                Crossover_Population();
        Mutation_Population();
                    Decoding();
        Calculate_fitness();
End While
```

3.1 Individual representation

In this section, the Encoding() procedure uses the activity sequence to encode the individual. A sequence of activities is considered as a prioritized arrangement of activities. For instance, a network is given by a diagram *G*=(*V*,*A*), where *V*={1,2,3,4,5,6}is the set of vertices, and *A*={(1,3),(3,5),(2,4),(4,6)} is the set of arcs. The individuals can be defined as I_1=(1,3,2,5,4,6), I_2=(2,4,6,1,3,5), as shown in Fig. 2.

1	3	2	5	4	6	I_1
2	4	6	1	3	5	I_2

Fig.2 Representation of individuals

3.2 Initial population generation

If we assume that the initial population consists of $\overline{n}$ individuals, then $\overline{n}$ activity lists are generated by the biased random sampling method based on the regret value. The priority rule in the sample algorithm uses the maximum number of successor activities. The priority modulus of activity j is denoted by $v(j)$, and the regret value is $\rho_j = \max_{i \in D} v(i) - v(j)$ when activity j is not scheduled. Therefore, we can calculate the probability of activity j by $\psi(j) = (\rho_j + 1)^\alpha \big/ \sum_{i \in E} (\rho_i + 1)^\alpha$. In this work, the adjustment modulus α ($\alpha > 0$) can be used to change the magnitude of the preference, and E is the set of alternative activities.

3.3 Parallel schedule generation scheme with overlap

In this section, the parallel schedule generation scheme with overlap (PSGS_Overlapping()) is proposed to decode individuals based on the characteristics of overlap. PSGS_Overlapping() utilizes time as the stage variable. In Algorithm 2, AL denotes the activity list, F denotes the set of finished activities, S denotes the set of executing activities, D denotes the set of active activities and *Current* denotes the decision points. At the beginning of the procedure, the decision point is determined by the start time of D. The activities that satisfy the resource constraints in *D* are known as the alternative activities, denoted by *E*. The candidate activity set is rearranged according to the activity list, and the activity to start is selected until *E* is empty. Finally, the active activity set and the start time of the activity are updated. The above process is repeated until all of the activities are sorted.

Algorithm 2.Pseudocode of PSGS_Overlapping()

Initialization:
Generate the list of activities AL;
$F = \Phi$; //Set of finished activities
$S = \{0\}$; //Set of executing activities
$D = Succeeding\{0\}$; //Set of active activities
$Current = 0$; //Decision point
While ($|F| < N+1|$)
For $l = 1$ to $length(D)$
$i \leftarrow D(l)$;
$Current \leftarrow \min\{ST_i\}$;
End For
For $l = 1$ to $length(S)$
$j \leftarrow S(l)$;
If $FT_j \le Current$
$F \leftarrow F \cup \{j\}$; //Activities finished at the decision point
End If
End For
$S \leftarrow S - F$;
//Delete the finished activities from the executing activities
At the decision point $Current$, calculate the set of alternative activities E;
For $l = 1$ to $length(D)$
$k \leftarrow D(l)$; //Each activity k in D
If $ST_k = Current$ & Resource satisfaction
$E \leftarrow E \cup \{k\}$;
End If
End For
While $E \neq \Phi$
Rearrange the set of alternative activities E according to the activity list;
$h \leftarrow E_m(1)$;
$ST_h \leftarrow Current$;
If $(g,h) \in A_o$ & $RT_g = Current$
$FT_h \leftarrow Current + d_h + \sum rw_h$;
Else
$FT_h \leftarrow Current + d_h$;
End If
Update the remaining amount of resources;
Update E;

Update $D \leftarrow D - \{h\} + Suceeding\{h\}$;
End While
For $l = 1$ to $length(D)$
$g \leftarrow D(l)$;

Set of all immediate predecessors of activity g are recorded as $Preceding(g)\ \forall f \in Preceding(g)$
If the activities f and g are not overlappable

$$ST_g = \max\{FT_f\};$$

Else

$$ST_g = \max\{FT_f - o_{fg}\};$$

End If
End For
End While

3.4 Genetic operators

(1) Selection operator. The individuals are selected using a simple ranking method. In this work, the expected duration of the project for each activity list is calculated as the fitness value. According to the order of the fitness values, activity lists are selected as a new population.

(2) Two-point crossover operator. Two individuals are selected in the population by the crossover probability p_c as parents, denoted by F and M. Two integers k_1, k_2 are randomly selected ($1 \le k_1 < k_2 \le n$).The two offspring individuals are generated from the parents, denoted by S and D. In the activity sequence of D, the position $i = 1, \ldots k_1$ is taken from individual M, and the position $i = k_1 + 1, \ldots, k_2$ is taken from individual F. The activities that have been selected from M are not considered again.

As shown in Fig. 3, for individuals $I_F = (1,3,2,5,4,6)$ and $I_M = (2,4,6,1,3,5)$, the positions $k_1 = 1$, $k_2 = 3$ are given. Thus, $I_S = (1,2,4,3,5,6)$, $I_D = (2,1,3,4,6,5)$ are obtained by the crossover procedure.

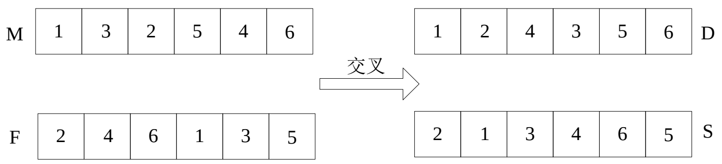

Fig. 3 Example of a two-point crossover operator

(3) Mutation operator. An individual is selected from the population with a mutation probability of p_m .A genotype is selected and inserted into the activity list at random. The process ensures that the resulting list of activities is feasible in terms of precedence. As shown in Fig. 4, for a given individual $I_F = (1,3,2,5,4,6)$,the genotype 5 is randomly inserted into the list, and the resulting individual $I' = (1,3,2,4,6,5)$ is generated.

Fig. 4 Example of a mutation operator

3.5 Calculation of fitness values

The pseudo code of the method for calculating the fitness value is shown in Algorithm 3, where $\overline{f}$ indicates the project duration obtained by PSGS_Overlapping() for activity list AL, f indicates the sum of all of the obtained project durations after REF simulations, and f^* indicates the expected value of the project duration. As more simulations are performed, the deviation between the simulations and theory decreases.

Algorithm 3.Pseudocode of Calculate_fitness()

Given activity list AL and the times of simulation REF ;

$\overline{f} \leftarrow 0$;

$f \leftarrow 0$;

For $rep = 0$ to REF

$\overline{f} \leftarrow$ PSGS_Overlapping(AL) ;

$f \leftarrow f + \overline{f}$;

End For

$f^{*} \leftarrow f / REF$;

IV. Experimental Analysis

4.1 Experimental design

First, we construct test instances with different parameters using the RanGen2 software (Vanhoucke et al., 2008; Kolish &Sprecher, 1996). A serial/parallel index (I2) is used to describe the topology of the network. In the network, a larger value of I2 corresponds to more serial relationships, and a smaller value of I2 corresponds to more parallel relationships. Resource constrainedness (RC) is used to measure the scarcity of resources. The smaller the value of RC, the more scarce the resources. Resource usage (RU) is used to reflect the degree of resources in demand, where a larger value of RU indicates more types of demand. The overlap and rework parameters are defined as follows: (1) The percentage of overlapping activities in the total pairs of activities is denoted by *Percentage*; this parameter can control the number of overlapping pairs in the project.(2)The overlap parameters α can control the length of overlap time between each pair of overlapping activities. (3) The rework parameters β indicate the probability that overlap will cause rework on the downstream activity in each time period, which can influence the duration of the rework time.

The value of each parameter is shown in Table 1. Each set of parameters randomly generates 10 examples, generating a total of $3\times2\times3\times3\times3\times3\times10 = 4860$ examples. The data set is denoted as Set (60). This algorithm was coded in Matlab R2013b, and the computational experiment was executed on an Intel Core i3 3.40-GHz computer under a Windows7 64-bitoperating system.

Table 1 Design of parameters

Parameter	Value
Number of activities N	60
Types of resource K	4
Serial/parallel index $I2$	0.3 ; 0.5； 0.7
Resource usage RU	2； 3
Resource constrainedness RC	0.3； 0.5； 0.7
Percentage	20%; 50%; 80%
α	0.3； 0.5； 0.7
β	0.3； 0.5； 0.7

4.2 Experimental results

To analyze the influence of different parameters on the project duration, the average objective function value (Ave) of the problem is used. Table 2 lists all calculation results of Ave under the combinations of parameters.

Table 2 Results for Ave with different parameter combinations

Data Set	α	*Percentage*	β		
			0.7	0.5	0.3
Set（60）	0.7	20%	201.80	199.12	197.30
		50%	196.90	190.76	184.73
		80%	191.84	182.94	174.83
	0.5	20%	202.30	200.42	198.52

	50%	197.79	193.18	188.08
	80%	194.34	186.41	178.64
0.3	20%	203.80	202.48	201.37
	50%	201.17	198.19	195.23
	80%	198.79	194.15	189.60

From Table 2, we draw the following conclusions: (1) As *Percentage* increases, Ave is significantly reduced, showing that the changes in the number of overlapping pairs have a great impact on the project duration, and the increase in overlapping pairs can significantly reduce the project duration.②As α increases, Ave decreases, indicating that the increase in overlap time can reduce the project duration.③As β increases, Ave increases, and the probability of rework time for the downstream activity increases the project duration.

V. Conclusion

In this paper, a genetic algorithm based on simulation is proposed to solve the scheduling optimization model. A large number of test instances with strong random characteristics are generated using the RanGen2 software, and computational experiments are conducted. The experimental results show the proportion of overlapping pairs to the total activity *Percentage*, the overlap parameter α , and the probability of the rework on downstream activity β.The effect of these three parameters on the project duration is also analyzed. We show that increasing *Percentage* and α and decreasing β can shorten the project duration.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (No. 72071092) and the Tianjin Intelligent Manufacturing Special Fund Project (No. 20201195).

References

[1] Bai S, Wan X. Project schedule cycle based on design structure matrix[J]. System Engineering Theory & Practice, 2008, 28(11): 51-54.

[2] Berthaut F, Greze L, Pellerin R, Perrier N, Hajji A. Optimal resource-constrained project scheduling with overlapping modes [EB/OL]. https://www.cirrelt.ca/DocumentsTravail/CIRRELT-2011-09.pdf, 2011a.

[3] Berthaut F, Pellerin R, Perrier N, Hajji A. Time cost tradeoffs in resource-constrained project scheduling with overlapping modes[EB/OL]. https://www.cirrelt.ca/DocumentsTravail/CIRRELT-2011-10.pdf,2011b.

[4] Blazewicz J, Lenstra J K, Rinnoy K. Scheduling projects subject to resource constraints: Classification and complexity[J]. Discrete Applied Mathematics, 1983, 5(1): 11-24.

[5] Bozejko W, Hejducki Z, Uchronski M et al. Solving resource-constrained construction scheduling problems with overlaps by metaheuristic[J]. Journal of civil engineering and management, 2014, 20(5): 649-659.

[6] Cho S H, Eppinger S D. A simulation-based process model for managing complex design projects [J]. IEEE Transactions on Engineering Management, 2005, 52(3): 316-328.

[7] Chu Z, Xu Z, Li H. New heuristics for the RCPSP with multiple overlapping modes[J]. Computer and Industrial Engineering, 2019,131:146-156.

[8] Chu,Z, Xu Z, Xie F. Experimental Evaluation of Overlapping Strategy for the Multimode Resource-Constrained Project Scheduling Problem[J]. Arabian journal for science and engineering, 2019, 44(3):2503-2517.

[9] Debels D, Vanhoucke M. A decomposition-based genetic algorithm for the resource-constrained project-scheduling problem[J]. Operations Research, 2007, 55(3): 457-459.

[10] Li C, Chu M. Is it always advantageous to add-on item recommendation service with a contingent free shipping policy in platform retailing?[J] Electronic Commerce Research and Applications, 2019, 37: 1-11.

[11] Gerk J E Y, Qassim R Y. Project Acceleration via Activity crashing, overlapping and substitution[J]. IEEE Transactions on Engineering Management, 2008, 55(4): 590-601.

[12] Greze L, Pellerin R, Leclaire P, Perrier N. A heuristic method for resource-constrained project scheduling with activity overlapping[J]. Journal of intelligent manufacturing, 2014, 25(4): 797-811.

[13] Hartmann S, Briskorn D. A survey of variants and extensions of the resource-constrained project scheduling problem [J]. European Journal of Operational Research, 2010, 207(1): 1-14.

[14] Zhao W. Optimal consumption with reference-dependent preferences in on-the-job search and savings[J]. Journal of Industrial and Management Optimization, 2017, 13(1): 503-527.

[15] Hartmann S. A competitive genetic algorithm for resource-constrained project scheduling[J]. Naval Research Logistics, 1998, 45(7): 733-750.

[16] Hartmann S. A self-adapting genetic algorithm for project scheduling under resource-constraints[J]. Naval Research Logistics, 2002, 49(5): 433-448.

[17] Kolish R, Sprecher A. PSPLIB-A project scheduling problem library[J]. European Journal of Operational Research, 1996, 96(1): 205-216.

[18] Koyuncu E, Erol R. PSO based approach for scheduling NPD projects including overlapping process[J]. Computers & Industrial Engineering, 2015, 85: 316-327.

[19] Krishnan V. Managing the simultaneous execution of coupled phases in concurrent product development[J]. IEEE Transactions on Engineering Management, 1996, 43(2):210-217.

[20] Yu J, Zhou C, Wang Y, et al. Incentive contract design for supply chain enterprise's pollution abatement with carbon tax[J]. Mathematical Problems in Engineering, 2021, Article ID 5556796: 1-14.

[21] Lin J, Chai K H, Brombacher A C, Wong Y S. Optimal overlapping and functional interaction in product development[J]. European Journal of Operational Research, 2009, 196(3): 1158-1169.

[22] Chen P, Zhao R, Yan Y, et al. Promoting end-of-season product through online channel in an uncertain market[J]. European Journal of Operational Research, 2021, 295(3): 935-948.

[23] Zhou C, Leng M, Liu Z, et al. The impact of recommender systems and pricing strategies on brand competition and consumer search[J]. Electronic Commerce Research and Applications, 2022, 53: 1-15.

[24] Li B, Xu Z, Wei WC.Bi-Objective Scheduling Optimization for Discrete Time/Cost Trade-Off in Projects[J].Sustainability. 2018, 10(8), 2802.4(7): 1412-1418.

[25] Zhou C, Zhao R. Optimal consumer search with prospect utility in hybrid uncertain environment[J]. Journal of Uncertainty Analysis and Applications, 2015, 3(6): 1-20.

[26] Lin J, Qian Y, Cui W, Miao Z. Overlapping and communication policies in product development[J]. European Journal of Operational Research, 2010, 201(3): 737-750.

[27] Maheswari J U, Varghese K. Project scheduling using dependency structure matrix[J]. International Journal of Project Management, 2005, 23(3): 223-230.

[28] Vanhoucke M, Coelho J, Debels D, Maenhout B, Tavares V L. An evaluation of the adequacy of project network generators with systematically sampled networks[J]. European Journal of Operational Research, 2008, 187(2): 511-524.

[29] Liu Z, Zhou C, Chen H, et al. Impact of cost uncertainty on supply chain competition under different confidence levels[J]. International Transactions in Operational Research, 2021, 28(3): 1465-1504.

[30] Wang J T, Lin Y I. An overlapping process model to assess schedule risk for new product development[J]. Computer and Industrial Engineering, 2009, 57(2): 460-474.

[31] Zhou C, Ma N, Cui X, et al. The impact of online referral on brand market strategies with consumer search and spillover effect[J]. Soft Computing, 2020, 24(4): 2551-2565.

[32] Yu J, Xu Z, Li H. Modeling and solving the resource-constrained project scheduling problem with activities overlapping[J]. Systems Engineering Theory & Practice, 2015, 35(5): 1236-1245.

[33] Li Y, Deng Q. Environmental governance strategies in a two-echelon supply chain with tax and subsidy interactions[J]. Annals of Operations Research, 2020, 290(1-2): 439-462.

[34] Zhou C, Xu G, Liu Z. Incentive contract design for internet referral services: Cost per click vs cost per sale[J]. Kybernetes, 2020, 49(2): 601-626.

[35] Zhang XL, Tan YJ, Yang ZW. Rework Quantification and Influence of Rework on Duration and Cost of Equipment Development Task[J]. Sustainability.2018,10(10),3590.