International Journal of Engineering

Journal Homepage: www.ije.ir

TECHNICAL NOTE

Resource Constrained Project Scheduling with Material Ordering: Two Hybridized Meta-Heuristic Approaches

N. Zoraghi [a], A. A. Najafi [b], S. T. A. Niaki* [c]

[a] *Faculty of Industrial and Mechanical Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran*
[b] *Department of Industrial Engineering, K.N. Toosi University of Technology, Tehran, Iran*
[c] *Department of Industrial Engineering, Sharif University of Technology, Tehran, Iran*

PAPER INFO

Paper history:
Received 08April 2014
Received in revised form 20August 2014
Accepted 18September 2014

Keywords:
Project Scheduling
Material Ordering
Resource Constraint
Hybrid Meta-Heuristic Algorithm
Taguchi Design

ABSTRACT

Resource constrained project scheduling problem (RCPSP) is mainly investigated with the objective of either minimizing project makespan or maximizing project net present value. However, when material planning plays a key role in a project, the existing models cannot help determining material ordering plans to minimize material costs. In this paper, the RCPSP incorporated with the material ordering problem is first formulated into a NP-hard optimization model. Then, two hybridized meta-heuristic algorithms are proposed to solve the integrated problem. In addition, statistical methods are employed to tune the parameters of both algorithms. Finally, computational results for a set of test problems taken from the project scheduling problem library (PSPLIB) are presented.

***doi**: 10.5829/idosi.ije.2015.28.06c.10*

1. INTRODUCTION

Project scheduling plays a major part in project management. Indeed, the scheduling process amounts to deciding when the project activities will start and how they will use the available resources. Traditionally, project scheduling problem (PSP) and material ordering (MO) are considered independently, i.e., at first the resource profile of a project is obtained by finding a schedule for that project, and then the material ordering approaches are called to find an ordering plan. Trade-offs between cost elements such as material ordering and holding, which are usually too complicated to be analytically derived, are not taken into account in this strategy. Therefore, it is not optimal in terms of project total cost.

Aquilano and Smith-Daniels [1] and Smith-Daniels and Aquilano [2] firstly integrated the PSP and MO. Later, Smith-Daniels and Smith-Daniels [3], Dodin and Elimam [4], and Sajadieh et al. [5] considered the integrated model. Recently, Najafi et al. [6] introduced a new mixed integer programming for PSMO model with no renewable resource based on the research of Sajadieh [5]. They proposed a hybrid meta-heuristic algorithm to solve the problem. The algorithm consists of two outside and inside searches where in one hand the outside search is a simulated annealing to determine the project schedule. On the other hand, the inside search, which is a genetic algorithm, recognizes the demand profiles of all nonrenewable resources produced from the outside search.

In general, project scheduling aims to determine the demand for renewable resources and materials, which corresponds to activity duration as well. Thus, the order size can be recognized to adhere to the requirement at present. Hence, a schedule composed of available resource profiles and durations of all the activities, meantime, an ordering plan, which in turn contains order timing and order size, can be achieved as well. This shows that project scheduling and material ordering is actually performed simultaneously. While no resources of renewable type have been involved in the available project scheduling with material ordering (PSMO) models of literature so far, in this paper, we

*Corresponding Author's Email: Niaki@Sharif.edu (S.T.A. Niaki)

lease cite this article as: N. Zoraghi, A. A. Najafi, S. T. A. Niaki, Resource Constrained Project Scheduling with Material Ordering: Two Hybridized Meta-Heuristic Approaches, International Journal of Engineering (IJE), TRANSACTIONS C: Aspects Vol. 28, No. 6, (June 2015) 896-902

extend the PSMO model to include problems in which the resources are constrained. The resource constrained PSMO integrates renewable resource constrained project scheduling (RCPS) and material ordering (MO) which has been investigated infrequently in the literature. Although this consideration certainly causes more complexities to the model, a closer to reality situation is investigated. Furthermore, two powerful hybridized meta-heuristics with two novel optimization loops are proposed to solve the problem.

2. PROBLEM FORMULATION

The objective considered in this paper involves scheduling of project activities such that not only the makespan of the project does not exceed a given due date, but also the total material holding and ordering cost is minimized, while the resource constraints and the precedence relation of activities are satisfied. Each activity is assumed to be carried out without interruption, zero-lag finish-to-start precedence constraints are imposed on the sequencing of the activities, and that the network is of the activity on node type with no loop. Furthermore, activities have one mode of executions. The renewable resource and non-renewable resource (material) usage over an activity is taken to be uniform. No material is available at the starting point of the project.Therefore,at the beginning of a typical period the model must determine the order quantity of the material. The lead-time is assumed inappreciable and the capacity of the warehouse is considered unlimited. However, it is assumed that any order quantity replenishment is instantaneous. Activity $i \in (0.1.....n+1)$ has duration d_i and uses r_{il} units of renewable resource $l \in (1.....L)$ and u_{if} units of nonrenewable resource (material) $f \in (1.....F)$ per period. For each renewable resource type l, the availability R_l is constant throughout the project horizon. A_f and H_f are the ordering and holding costs of material f, respectively, and DD stands forthe deadline of the project. S_i is the starting time of activity i, where ES_i and LS_i are earliest starting time and latest starting time of activity i, respectively. Meanwhile $M^{\times}$ denote a relatively large positive number. To formulate the problem, the decision variables are defined as:

X_{it} : A binary variable where it is one if activity i is started in per period t and zero otherwise

λ_{ft} : A binary variable where it is one if material type $f \in (1.....F)$ is ordered in per period t and zero otherwise

Q_{ft} : The ordered quantity of material type f in period t

I_{ft} : The inventory level of material type f in period t

Then, the problem at hand is formulated as follows:

$$Min\ Z = \sum_{f=1}^{F} \sum_{t=1}^{DD-1} H_f \times I_{ft} + \sum_{f=1}^{F} \sum_{t=0}^{DD} A_f \times \lambda_{ft} \tag{1}$$

Subject to:

$$S_i - S_j \geq d_j \ \ ; \ \forall j \in P(i) \ , \ i = 1,2,...,n \tag{2}$$

$$\sum_{i=1}^{n} \sum_{w=Max(t-d_i+1,ES_i)}^{Min(t,LS_i)} r_{il} X_{iw} \leq R_l \ ; t=0,1,2,...,DD, l=1,2,...,L \tag{3}$$

$$I_{ft} = I_{f(t-1)} + Q_{ft} - \sum_{i=1}^{n} \sum_{w=max\{t-d_i+1,ES_i\}}^{min\{t,LS_i\}} u_{if} \times X_{iw} \quad for \ \ f = 1,2,...,F \ \ , \ \ t = 1,2,...,DD \tag{4}$$

$$I_{f0} = 0; \ f = 1.2.....F \tag{5}$$

$$\sum_{t=ES_i}^{LS_i} X_{it} = 1 \ \ ; \ \ i = 1,2....,n \tag{6}$$

$$\sum_{t=ES_n}^{LS_n} t \times X_{nt} \leq DD \ \ ; \ \ i = 1,2,...,n \tag{7}$$

$$S_i = \sum_{t=ES_i}^{LS_i} t \times X_{it} \ \ ; \ \ i = 1,2,...,n \tag{8}$$

$$Q_{ft} \leq \lambda_{ft} \times M^{\times} ; \ f = 1.2.....F \ ; \ t = 1.2.....DD \tag{9}$$

$$X_{it} \in \{0.1\}; \ i = 1.....n \ ; \ t = ES_i .....LS_i \tag{10}$$

$$\lambda_{ft} \in \{0.1\}; f = 1.2.....F; \ t = 1.2.....DD \tag{11}$$

$$Q_{ft} \geq 0 \ ; \ f = 1.2.....F \ ; \ t = 1.2.....DD \tag{12}$$

$$I_{ft} \geq 0 \ ; \ f = 1.2.....F \ ; \ t = 1.2.....DD \tag{13}$$

The objective function (1) minimizes the total material holding and ordering costs. Constraint (2) enforces the precedence relations between activities. Inequality (3) enforces sufficiency to the provided resource units to implement the schedule. Constraints (4) and (5) are balance equations to monitor the inventory level of the resources over the project duration. Equation (6) states that every activity must be started only once. Constraint (7) ensures that the project ends by the latest allowable completion time. Equation (8) states how the starting time of activities is defined. SinceInequality (9) states the ordering time of any material $Q_{ft} > 0$, then $\lambda_{ft} = 1$ can be used to prevent it to become unbounded. Sets of constraints (10)-(13) show the domain of the variables.

According to Blazewicz et al. [7], the *RCPSP* is an NP-hard problem. Consequently, the resource constrained project scheduling integrated with material ordering (*RCPSMO*) is also NP-hard. Therefore, in the next section two hybrid meta-heuristics including genetic algorithm (*GA*) and simulated annealing (*SA*) are proposed to solve it.

3. HYBRID META-HEURISTICS

The problem at hand consists of two parts; project scheduling (PS) and material ordering (MO). Here, two different meta-heuristic algorithms are first proposed for PS part and one meta-heuristic for MO part. Then, these algorithms are combined to solve the integrated problem. The hybrid algorithms include an approach to find schedules of activities and another approach to find the best ordering plan for materials whose demand profiles have been determined by the schedule. To search the PS, a genetic algorithm (GA) and a simulated annealing (SA) are designed and for ordering materials, a genetic algorithm (GA) is developed. The details of these methods are defined in the following sections.

3. 1. Revealing the Project Schedule The structure of a solution in both algorithms is similar. An initial *n* dimensional solution vectors $\boldsymbol{X}$ are randomly generated on the interval [0,1] based on the random key representation introduced by Kolisch and Hartmann [8], where the position of each place in the vector $\boldsymbol{X}$ corresponds to the non-dummy activity $X_i ; i \in (1,2,\ldots,n)$. Then, the activity sequence representation method is used to make an activity sequence list (*AL*) from the vector $\boldsymbol{X}$ [8]. The precedence-feasible activity list contains ALs that are feasible. To transform an activity list into a feasible project scheduling $S_i \in (S_1, S_2, \ldots, S_n)$, the serial schedule generation scheme (*SGS*) investigated by Kolisch and Hartmann [8] is employed in this research to obtain precedence and resource feasible schedule for activity lists. For example, a small project with 6 non dummy activities and two renewable and non-renewable resources is shown in Figure 1.

To create an initial solution for the network above (a project schedule), the previously mentioned steps are employed. Figure 2 depicts these steps.

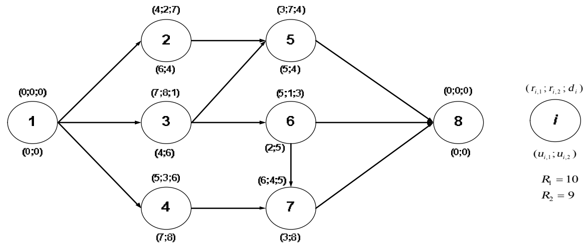


Figure 1. Activity network with eight activities

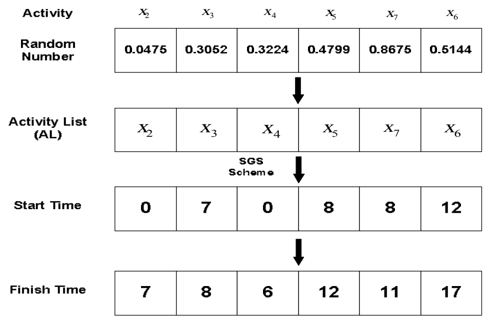


Figure 2. Create a project schedule as an initial solution

When the starting and finishing time of the activities are revealed as a project schedule, resource usage for both renewable and nonrenewable resources are calculated. In order to evaluate the objective function of each solution, the order quantities of the materials (Q_{ft}) are needed. For this purpose, a genetic algorithm as an inside searcher is applied to determine the best ordering policy. In each period, X_i and Q_{ft} are produced and the procedure iterates until the algorithm is unable to find a better solution. As a result, the schedule with the best ordering policy is obtained.

3. 1. 1. Simulated Annealing Simulated annealing *(SA)* basically starts with a generation of an initial solution (one point). In the proposed hybrid *SA* of this research, the algorithm starts with generation of several initial solutions (multiple points) in an attempt to find a better solution. For this purpose, solution $\boldsymbol{X}$ is randomly generated on the interval [0,1] according to steps mentioned above. In order to reduce the probability of getting trapped in local optima, *SA* accepts moves to inferior neighboring solution under the control of randomized scheme on X_i. Two neighborhood search structures (*NS*) are employed in this research. In the first type of *NS*, one cell of a current vector solution is chosen randomly and is exchanged with the value of a randomly generated number between [0,1]. The second mechanism deletes a value of a cell selected randomly from a position and inserts it in a new randomly selected position. Note that, the proposed SA algorithm uses the above two *NS* with an equal probability of 0.5. The cooling scheme is one of the most important parameter in *SA* algorithms. In this research, a linear cooling scheme is applied to decrease the temperature. To stop SA, we use a fixed number of iterations. The *NS*s are illustrated in Figure 3.

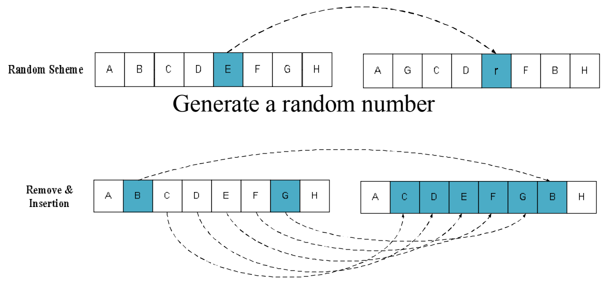
3. 1. 2. Genetic Algorithm In the developed *GA* of this research, the chromosomes of a population are generated using uniformly distributed random numbers in the interval [0,1]. The coded chromosome is deciphered to the vector $S_i \in (S_1, S_2, \ldots, S_n)$. Prior to the crossover operation, pairs of individuals (parents) must be first selected. The tournament selection strategy is used for this action. For combining the parents, the arithmetic crossover operator is employed on $\boldsymbol{X}$ to create feasible offspring. To demonstrate this type of crossover operation, consider two individuals selected for a crossover operation. Then, each gene of the two created children is obtained as:

$$\begin{cases} CH^1 = P^1(\alpha_t) + P^2(1-\alpha_t) \\ CH^2 = P^1(1-\alpha_t) + P^2(\alpha_t) \end{cases} \tag{14}$$

Note that this type of crossover operator assures feasible solution. The mutation operator of the *GA* works similar to the first neighborhood structure mentioned in the *SA* algorithm. The process of generating new chromosome and searching for better solutions continues until a fixed number of generations are made.

3. 2. Determining the Material Ordering Until recently, determining the time-varying material ordering lot sizing problem has been solved using the well known dynamic programming, Wagner and Whitin, and Silver-Meal algorithms. In general, finding an optimal solution or even sub-optimal solutions is not an easy task when the size of problem is relatively large. Consequently, meta-heuristic algorithms especially genetic algorithm is used to solve the intractable time-varying material ordering lot sizing problems in a reasonable time in this research. This *GA* is combined with both outer *SA* and *GA* to incorporate two hybrid algorithms. In order to determine the objective function, order quantities of the materials in each period must be depicted. In inside *GA*, a binary representation based on the order quantities of the materials in each period is employed to generate a chromosome (Shittu [9]).

Each individual chromosome considered as a matrix, $\boldsymbol{Q}$, which represents the order quantity (Q_{ft}) of material type f in period t. Obviously, the length of a matrix is equal to time periods $t = 1, \ldots, DD$. To create the chromosome, each row of matrix $\boldsymbol{Q}$ is generated randomly based on '0' and '1'. Thus, the chromosome representation is illustrated in Figure 4.



Generate a random number

Removal and insertion

Figure 3. The schematic neighborhood structures

$$Q_{ft} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}_{Material \times Time}$$

Figure 4. Chromosome representation

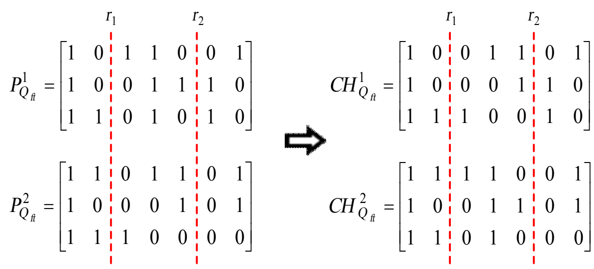


Figure 5. Two point crossover operator

In this figure, '0' indicates no order in the corresponding period and '1' shows an order in the corresponding period for its demand, where the demands of all subsequent periods are '0's. Meanwhile we assume that the first gene in all rows of a chromosome is '1'. Afterward, the coded chromosome must be transformed to the order quantity of material in each period based on the value of the gene in the demand profile. It should be noted that the total order quantity of each row in the chromosome (materials) must be equal to the total demand profile of corresponding material type f during the project. Thereafter, the initial population is generated randomly according to this chromosome structure. In order to evaluate a chromosome, the fitness value obtained by the objective function given in (1) is calculated. Then, the roulette wheel approach is utilized to select parents. Next, the two-point crossover operation is performed on the selected parents with probability of P_{cr} to produce two new feasible individuals (children). With these crossovers, the value of each gene in the offspring coincides with the value of this gene in one of the parents.

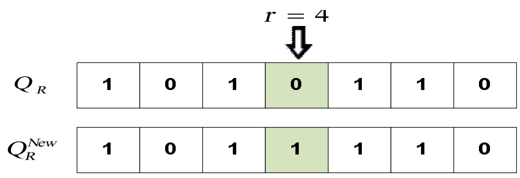


Figure 6. The mutation representation

To do this, two points of the parent are randomly selected $r_1 , r_2 \in \{2,3,\ldots,DD-1\}$ where $r_1 < r_2$. The parent, defined by them, is exchanged to generate two off-springs according to Figure 5.

Likewise, for increasing diversification of chromosome, mutation operator is introduced with the probability of P_{mut}. For this purpose, two integer random numbers are generated in the intervals $[1, F]$ to select a material (R) and $[2, DD]$ to mutate the genes of the child (r) respectively. Consequently, if the value of gene (r) is equal to '0', it is changed into '1', otherwise, it remains "0". Figure 6 depicts this process. The best individual of the last generation is the best ordering policy of the *GA* for a given activity schedule. This process continues until a certain number of iterations is performed.

4. EXPERIMENTAL RESULTS

4. 1. Comparison with an Exact Solver

A set of 360 instances including 10, 20 and 30 non-dummy activities with three renewable and nonrenewable resources are generated by *PROGEN* generator software. The instances are solved by the well-known *CPLEX* software as well by the hybrid algorithms of this research. During *CPLEX* execution, the time is limited to 900 seconds, i.e. if *CPLEX* cannot find the optimal solution in 900 seconds, the algorithm is stopped. The proposed hybrid algorithms are first compared with the *CPLEX* software in number of instances in which the algorithms or *CPLEX* could find a solution (#S). Thereafter, the relative deviation (*GAP*) from the optimal solution extracted from *CPLEX* is used as a criterion for comparison. *GAP* is defined as:

$$GAP = \frac{\text{Algorithm}_{\text{Result}} - CPLEX_{Result}}{CPLEX_{Result}} \qquad (15)$$

Table 1 illustrates the outcomes of the comparison. As observed, the *CPLEX* solver is unable to find the global optimum solution when the problem dimension increases. This confirms the NP-hardness of the problem. Moreover, the average *GAP* reveals that, when the number of activities, resources, and materials increases, both the proposed SA-GA and GA-GA are able to find solutions that are close to the one obtained by the *CPLEX* solver. It can be inferred that the *GAP* index increases gradually when the number of activities and resources (renewable and material) increases. As seen, the average *GAP* given by GA-GA is smaller than the one obtained by SA-GA where there are more activities and resources. This indicates that GA-GA finds better and closer solutions to the optimal one. In terms of maximum *GAP*, similar outcomes are observed.

4. 2. Performance Analysis

To evaluate the performance of each algorithm, 600 large-size problems are generated by the *PROGEN* software. The problems vary in the number of activities (*N*) within 30 to 120, in the number of renewable resources (*R*) including 1 to 3, and in the number of non-renewable resources (*NR*) ranging from 1 to 3. The duration of activity varies between [1, 10]. Moreover, to convert the problems to *RCPSMO* instances, holding and ordering costs are randomly generated from the interval [10; 200] and (500; 5000], respectively. To assess the efficiency of both algorithms in large-size problems, two criteria are applied as:

$$\%\, Improve = 100 \times \left(\frac{Fitness_{BI} - Fitness_{BA}}{Fitness_{BI}} \right) \qquad (16)$$

$$(CV)^2 = \frac{\sum_{i=1}^{n}\left(Fitness_i - \sum_{i=1}^{n} Fitness_i \Big/ n \right)^2}{n-1} \Bigg/ \left(\sum_{i=1}^{n} Fitness_i \Big/ n \right)^2 \qquad (17)$$

In the first criterion, the best solution of the initial implementation ($fitness_{BI}$) is compared with the best solution found by the algorithm ($fitness_{BA}$) in a limited available computational time. The algorithm convergence variation is the second criterion, based on which the small values are preferred since they are indicative of better convergences after normalization of the objective functions. In this criterion, *n* shows the number of replications for each problem. Results in Table 2 indicate that the proposed GA-GA has better improvement percentages compared to the ones of SA-GA. According to the results in this table, when the problem size and the number of materials are increased, the average and minimum percentage improvement of both algorithms increase. It is witnessed that the average improvement resulted from GA-GA, 24.92%, is a bit greater than that of SA-GA, 22.42%. In addition, the minimum improvement of SA-GA, 4.12%, is smaller than GA-GA. It means that the proposed GA-GA outperforms SA-GA.

TABLE 1. The results obtained to compare the algorithms

N	R	NR	#S			Average of (GAP)		Maximum of (GAP)	
			SA-GA	GA-GA	CPLEX	SA-GA	GA-GA	SA-GA	GA-GA
10	1	1	30	30	30	0.05	0.04	0.09	0.08
	1	2	30	30	28	0.06	0.05	0.10	0.10
	2	2	30	30	24	0.07	0.07	0.14	0.13
	2	3	30	30	20	0.09	0.09	0.16	0.15
20	1	1	30	30	22	0.13	0.13	0.26	0.23
	1	2	30	30	19	0.15	0.14	0.28	0.26
	2	2	30	30	16	0.17	0.15	0.33	0.30
	2	3	30	30	12	0.19	0.18	0.36	0.32
30	1	1	30	30	15	0.23	0.21	0.41	0.36
	1	2	30	30	13	0.26	0.24	0.45	0.39
	2	2	30	30	7	0.28	0.26	0.49	0.42
	2	3	30	30	3	0.30	0.27	0.58	0.48

TABLE 2. The computational results of the large-sized problems

N	R	NR	No. of Instance	Time Limit (s)	Average Improvement (%)		Min Improvement (%)		$(CV)^2$	
					GA-GA	SA-GA	GA-GA	SA-GA	GA-GA	SA-GA
30	3	1	50	90	16.97	12.91	3.21	3.18	0.116	0.121
	2	2	50	90	19.70	17.41	7.09	7.09	0.042	0.044
	1	3	50	90	34.67	23.30	10.46	8.78	0.029	0.033
60	3	1	50	120	16.94	15.89	3.88	3.08	0.042	0.048
	2	2	50	120	20.95	20.83	4.19	3.86	0.028	0.030
	1	3	50	120	27.97	21.11	6.15	4.91	0.020	0.023
90	3	1	50	150	24.02	23.88	3.60	2.91	0.037	0.043
	2	2	50	150	26.78	25.46	4.68	3.54	0.030	0.031
	1	3	50	150	30.77	28.11	5.58	4.54	0.021	0.025
120	3	1	50	180	23.90	25.85	2.12	1.42	0.040	0.044
	2	2	50	180	27.57	26.72	3.14	2.93	0.016	0.020
	1	3	50	180	28.74	27.58	3.81	3.17	0.011	0.014
Average					24.92	22.42	4.83	4.12	----	----

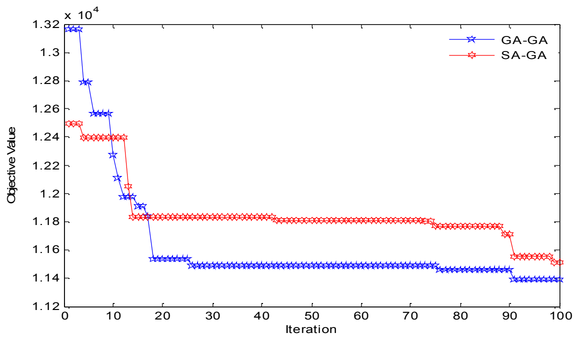


Figure 7. Convergence of objective value

Based on $(CV)^2$ outcomes, when the problem size (both activities and materials) increases, the convergence variation of both algorithm is gradually reduced. As seen, the convergence of hybrid *GA* in all instances is better than the hybrid *SA* algorithm. The convergence of each algorithm are illustrate in Figure 7.

5. CONCLUSIONS

In this research, a class of project-scheduling problems called resource-constrained project scheduling integrated with material ordering was considered. While no resources of renewable type has been used in the available PSMO models of literature so far, in this paper, we extended the PSMO model to include problems in which the resources are renewable and constrained. The problem was first formulated into a mixed integer-programming model. Since the model was strongly NP-hard, two hybrid meta-heuristic approaches of hybrid genetic algorithm (GA-GA) and hybrid simulated annealing (SA-GA) were developed to solve it. Both algorithms consist of an outside and an inside search engine. The outside search engine is either a simulated annealing, or a genetic algorithm to determine the project schedule. The inside search engine that is a genetic algorithm recognizes the demand profiles of all materials produced from the outside search engine. To evaluate and compare the performances of the two algorithms, various test problems of different sizes were generated by the *PROGEN* software, where they were modified to fit the model. Comparison of results of the two meta-heuristics revealed better performances of GA-GA compared to SA-GA.

6. REFERENCES

1. Aquilano, N.J., Smith-Daniels, D.E. "A formal set of algorithms for project scheduling with critical path method material requirements planning", ***Journal of Operations Management***, Vol. 1, (1980), 57-67.
2. Smith-Daniels, D.E., Aquilano, N.J. "Constrained resource project scheduling subject to material constraints", ***Journal of Operations Management***, Vol. 4, (1984), 369-388.
3. Smith-Daniels, D.E., Smith-Daniels, V.L. "Maximizing the net present value of a project subject to material and capital constraint", ***Journal of Operations Management***, Vol. 7, (1987), 33-44.
4. Dodin, B., Elimam, A.A. "Integrated project scheduling and material planning with variable activity duration and rewards", ***IIE Transactions***, Vol. 33, (2001), 1005-1018.
5. Sajadieh, M.S., Shadrokh, S., Hassanzadeh, F. "Concurrent project scheduling and material planning: A genetic algorithm approach", ***ScientiaIranicaTransactions E: Industrial Engineering,***Vol. 16, (2009), 91-99.
6. Najafi, A.A., Zoraghi, N., Azimi, F. "Scheduling a project to minimize costs of material requirements", ***World Academy of Science, Engineering and Technology***, Vol. 78, (2011), 134-137.
7. Blazewicz, J., Lenstra, J., Kan, A.R. "Scheduling subject to resource constraints: Classification and complexity", ***Discrete Applied Mathematics***, Vol. 5, (1983), 11–24.
8. Kolisch, R., Hartmann, S. "Heuristic algorithms for the resource-constrained project scheduling problem: classification and computational analysis", In:Weglarz J (ed) Project scheduling recent models, algorithms and applications. Kluwer Academic, Boston (1999) 147–178.
9. Shittu, E. "Applying genetic algorithms to the deterministic time-varying fixed quantity lot sizing problem", Master's Thesis. The American University in Cairo, Egypt (2003).

Resource Constrained Project Scheduling with Material Ordering: Two Hybridized Meta-Heuristic Approaches

TECHNICAL NOTE

N. Zoraghi [a], A. A. Najafi [b], S. T. A. Niaki [c]

[a]*Faculty of Industrial and Mechanical Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran*
[b]*Department of Industrial Engineering, K.N. Toosi University of Technology, Tehran, Iran*
[c]*Department of Industrial Engineering, Sharif University of Technology, Tehran, Iran*

PAPER INFO

Paper history:
Received 08 April 2014
Received in revised form 20 August 2014
Accepted 18 September 2014

Keywords:
Project Scheduling
Material Ordering
Resource Constraint
Hybrid Meta-Heuristic Algorithm
Taguchi Design

چکیده

مسئله زمان‌بندی پروژه با محدودیت منابع (RCPSP) اساساً به بررسی یکی از اهداف کمینه سازی زمان اتمام پروژه یا بیشینه سازی ارزش خالص فعلی پروژه می‌پردازد. بنابراین، مدل‌های موجود زمانی که برنامه ریزی مواد نقشی کلیدی در یک پروژه را ایفا می‌کنند، قادر به تعیین برنامه سفارش‌دهی مواد برای کمینه‌سازی هزینه‌ها نیستند.در این مقاله، ابتدا مسئله (RCPSP) با مسئله سفارش‌دهی مواد در قالب یک مدل بهینه‌سازی NP-Hard ترکیب می‌شود. سپس، دو الگوریتم فراابتکاری ترکیبی برای حل مسئله ادغام شده پیشنهاد می‌شود. علاوه براین، روش‌های آماری نیز برای تنظیم پارامترهای هر دو الگوریتم به کارگرفته می‌شوند. در نهایت، نتایج محاسباتی برای مجموعه‌ای از مسائل نمونه، برگرفته از کتاب‌خانه مسائل زمان‌بندی پروژه (PSPLIB)ارائه می‌شوند.

doi*: 10.5829/idosi.ije.2015.28.06c.10*