Management and Production Engineering Review
Volume 11 • Number 2 • June 2020 • pp. 88–98
DOI: 10.24425/mper.2020.133731

A HYBRID ALGORITHM BASED ON NON-DOMINATED SORTING ANT COLONY AND GENETIC ALGORITHMS FOR SOLVING MULTI-OBJECTIVE MULTI-MODE PROJECT SCHEDULING PROBLEMS UNDER RESOURCE CONSTRAINTS

Jafar Bagherinejad[1], Fariborz Jolai[2], Raheleh Abdollahnejad[1], Mahnaz Shoeib[1]

[1] *Department of Industrial Engineering, Faculty of Engineering, Alzahra University, Tehran, Iran*
[2] *Department of Industrial Engineering, Faculty of Engineering, Tehran University, Tehran, Iran*

Corresponding author:
Jafar Bagherinejad
Department of Industrial Engineering
Faculty of Engineering
Alzahra University
Tehran, Iran
phone: +982188041469
e-mail: jbagheri@Alzahra.ac.ir

Received: 18 July 2019
Accepted: 31 May 2020

Abstract
A project scheduling problem investigates a set of activities that have to be scheduled due to precedence priority and resource constraints in order to optimize project-related objective functions. This paper focuses on the multi-mode project scheduling problem concerning resource constraints (MRCPSP). Resource allocation and leveling, renewable and non-renewable resources, and time-cost trade-off are some essential characteristics which are considered in the proposed multi-objective scheduling problem. In this paper, a novel hybrid algorithm is proposed based on non-dominated sorting ant colony optimization and genetic algorithm (NSACO-GA). It uses the genetic algorithm as a local search strategy in order to improve the efficiency of the ant colony algorithm. The test problems are generated based on the project scheduling problem library (PSPLIB) to compare the efficiency of the proposed algorithm with the non-dominated sorting genetic algorithm (NSGA-II). The numerical result verifies the efficiency of the proposed hybrid algorithm in comparison to the NSGA-II algorithm.

Keywords
Multi-objectives project scheduling, Ant colony algorithm, Time-cost trade-off, Resources leveling.

Introduction

Although the resource constraint is one of the main characteristics of real project scheduling problems, the basic form of project scheduling problems has not considered resource availability constraints. Problems without recourse limitation are called non-resource constrained project scheduling problems. The problems are known as resource-constrained project scheduling problems (RCPSP) when the capacity of resources is limited [1–4]. Elmaghraby [5] developed the RCPSP in which each activity can be performed in the multiple modes. The multi-mode resource-constrained project scheduling problem is known as MRCPSP. The RCPSP has a broad application such as scheduling of manufacturing systems, construction industry, and software development [6]. The project scheduling problem attempts to optimize a set of objective functions concerning the various precedence constraints and resource constraints [7]. The previous studies classified scheduling problems based on some categories such as a framework of a project scheduling model including resources, activities, objectives, and schedules [8]. During the last decades, exact (small size) and approximate methods (heuristics and metaheuristics) have been used to solve MRCPSP [8]. Some of these methods are: branch and bound algorithm [9–11], branch and cut algorithm [12], local constraint heuristic algorithm [13], stochastic scheduling [14], genetic algorithm (GA) [15–19], tabu search algorithm [20], ant colony optimization (ACO) algorithm [21], particle swarm optimization (PSO) algorithm [22], and hybrids metaheuristics [23, 24].

MRCPSP could be developed to the multi-objective problems (MOMRCPS). In project scheduling problem studies, some have focused on time and cost parameters simultaneously [25–30]. These problems are known as time-cost trade-off problems (TCTP). The first TCTP solution approach was based on exact methods [31]. Some of these exact mathematical programming methods are network flow computations, linear programming, integer programming, mixed integer linear programming, and network decomposition [8]. Exact methods are applicable only for small size projects. Using exact methods quickly becomes impossible as the size of the problem increases. So, researches have attempted to develop heuristic and exact approaches to solve large size problems. Ant colony algorithm has been applied as a TCTP solution methodology [27, 32–34]. Sonmez and Bettemir [29] developed a hybrid genetic-simulated annealing algorithm to solve discrete TCTP. Some studies have considered two objective functions (including time and cost) in TCTP. Tavana et al. [35] considered quality as the third objective function. The resource leveling problem (RLP) is one of the most important topics in project management. This problem determines the best assignment of resources by minimizing the fluctuation of required resources over time. Researchers proposed different solution methodologies to solve the RLP based on the minimum squares optimization, minimum moment, and entropy maximization [36]. Nudtasomboon and Randhawa [37] considered multiple objectives (time, cost, and resource leveling) in the resource-constrained project scheduling problem. Hariga and Alsayegh [38] developed a multi-resource RLP in order to minimize costs caused by resource consumption fluctuations and splitting non-critical activities. Considering interruption property was the main novelty of their proposed model. Geng et al. [39] presented a directed ant colony algorithm (DACO) to solve non-linear resource leveling problems. Ponz-Tienda et al. [36] presented an adaptive GA to slove RLP. The proposed algorithm used Weibull distribution in order to evaluate the global optimum as a termination condition.

Most project scheduling models consider only one aspect of the problem, and few studies consider different problems simultaneously. Ghoddousi et al. [35] studied the simultaneous incorporation of MRCPSP, discrete TCTP, and RLP. Recently, parallel and hybrid metaheuristics have been proposed to solve the project scheduling problems. In this paper, a new hybrid genetic-ant colony algorithm has been developed for multi-objective multi-mode resource-constrained project scheduling problem (MOMRCPSP). Based on the literature review, only a few studies [21, 40] have presented an ant colony algorithm to solve MRCPS. In this paper, ACO presented by Li and Zhang [21] has been used to develop a multi-objective algorithm. A non-dominated sorting version of ACO (NSACO) is applied to find Pareto-optimal solutions of the problem. To the best of our knowledge, this study is the first application of the hybrid NSACO-GA optimization methodology in MOMRCPS. Finally, the efficiency of the developed algorithm has been investigated in comparison to the well-known NSGA-II algorithm. The numerical results verify the efficiency of the proposed algorithm based on diversification and convergence criteria in multi-objective project scheduling problems.

The remainder of this paper is organized as follows. In Sec. 2, the problem is described and formulated. The solution methods are presented in Sec. 3. Numerical examples appear in Sec. 4. Comparison metrics are provided in Sec. 5. Results are discussed in Sec. 6. Finally, Sec. 7 concludes the paper and suggests directions for future research.

Model description

In the construction industry, contractors usually manage and execute multiple activities by considering precedence relationships among the activities, in which each activity can be performed in one of several modes. So, construction planning is a challenging activity in the management and execution of construction projects. Hereby, the resource-constrained project scheduling model presented by Ghoddousi et al. [17] is considered. To present this model, some assumptions are given:

Consider a project which includes J activities. The structure of the project is represented by a so-called activity-on-node (AON) graph G (V, E), where V and E indicate a set of activities and precedence between activities, respectively. Let define the set of project activities by $J^+ = \{0, 1, ..., J+1\}$ which node 0 and $J+1$ are dummy start and terminal nodes of the project, respectively. According to the assumptions of MRCPSP, each activity j can be performed in only one of M_j possible modes. Activity j in mode $m \in M_j$ possesses a duration d_{jm} and a cost c_{jm}. Activity j executed in mode m requires r_{jmk} units of renewable resource type k and nr_{jml} units of non-renewable resource type l. Moreover, R_k and NR_l indicates the total availability limit of the renewable resource k and non-renewable resource l, respectively. Suppose c_i, c_p and T_c are indirect costs per period in the project makespan, penalty cost in each period of delay and project deadline. It is assumed that once the execution of activity j is started in mode m, it has to be completed in mode m without interruption. Let $r_k^{(t)}$ be the consumption of resource k in period t, the average consumption of resource k is defined as (1)

$$\overline{r}_k = \frac{1}{T}\sum_{t=1}^{T} r_k^{(t)}. \tag{1}$$

The variables of the model are as follows:

$$x_{jm} = \begin{cases} 1 & \text{if the activity } j \text{ is performed in mode } m, \\ 0 & \text{otherwise,} \end{cases}$$

$$y_J = \begin{cases} 1 & f_J > T_c, \\ 0 & f_J \leq T_c. \end{cases}$$

In which f_j be the finish time of activity j. Obviously, the project duration is equal to the finish time of end activity (f_J).

The proposed multi-objective mathematical model of the MOMRCPS is as follows [17]:

$$\text{Min} \quad f_J, \tag{2}$$

$$\text{Min} \quad \sum_j \sum_{m\in M_j} x_{jm}c_{jm} + f_J c_i + y_J c_p (f_J - T_c), \tag{3}$$

$$\text{Min} \quad \sum_{k=1}^{K}\sum_{t=1}^{T}\left(r_k^{(t)} - \overline{r}_k\right)^2, \tag{4}$$

s.t.

$$\sum_{m\in M_j} x_{jm} = 1 \qquad j \in V, \tag{5}$$

$$f_j - \sum_{m\in M_j} x_{jm}d_{jm} \geq f_i \qquad \forall (i.j) \in E, \tag{6}$$

$$\sum_{j\in A_t}\sum_{m\in M_j} x_{jm}r_{jmk} \leq R_k \qquad \forall k \,.\, A_t = \{j \,|\, f_j - d_j < t \leq f_j\}, \tag{7}$$

$$\sum_{j=1}^{J}\sum_{m\in M_i} x_{jm}nr_{jml} \leq NR_i \qquad \forall l. \tag{8}$$

The objective function (2) minimizes the total project duration. The objective function (3) minimizes the project costs including costs of delay, direct and indirect costs of the project. The objective function (4) minimizes the fluctuations in resource consumptions in a way that the differences between the amount of required resource and mean consumption of that resource get minimized. Constraint (5) enforces that each activity is allowed to be done only in one mode. Constraint (6) defines the precedence relationships between each pair of activities (i, j) which belongs to the set E. Constraints (7) and (8) guarantee that the total consumed amounts of non-renewable and renewable resources should not exceed the maximum available amount of these resources.

Solution methods

I) The proposed hybrid NSACO-GA for MOMRCPSP

The ant colony optimization algorithm is a metaheuristic solution method presented by Dorigo [41] in 1992. It states that a set of artificial ants collaborate in order to achieve an optimal solution in discrete optimization problems. The better solution arises from an interactive collaboration between existing ants [42]. Each ant releases a chemical substance called pheromone along its path. All members of the ant colony feel these released pheromones and follow the path with the most amount of released pheromone [42]. This algorithm consists of three main processes: generating initial solutions by ants, updating the amount of pheromone at each iteration, and auxiliary operations [42]. In order to build a new set of solutions, the ants in each colony evaluate the neighboring ants and determine the next movement direction. This decision is based on the pheromone of each path and heuristic information. The pheromone updating is a process of changing the quantity of pheromone in each path. The quantity of pheromone in each path could be increased due to the pheromone releasing by ants. It also decreases because of pheromone evaporation. Auxiliary operations are a set of processes done for implementing operations which could not be performed by each ant, lonely. Applying a local optimization method or general data collection method for decision making about other pheromones releasing- which directs the solutions toward a locally optimal solution- are among the auxiliary operations.

In this study, similar to Li and Zhang [21], two pheromone levels, two types of the corresponding probability, and heuristic information have been considered in the proposed MRCPSP. GA has been applied to improve the efficiency of the local search. A holistic schematic view of the proposed algorithm is shown in Fig. 1. The proposed hybrid algorithm is similar to the hybrid GA-ACO proposed by Akpinar et al. [43] in the assembly line balance problem. The letters l and c used in Fig. 1 refers to the number of iteration of GA and ACO algorithms, respectively. As it is obvious, the newly built solution by ants and the best-known solutions up to each iteration is considered as the initial solution of GA. Then, the mutation and crossover operations are used to create new solutions from GA population. New generated population evaluated using a non-dominant sorting method, and a predefined number of them are selected for the next iteration of GA. When the termination condition of GA is satisfied, ACO updates the quantity of pheromone at each path using the best efficient solutions obtained by GA. Also, the best solutions are updated in each iteration. The process continues until maximum iterations or some other stopping criteria are met. Then the solutions created using NSACO-GA are decoded as feasible scheduling solutions using the serial schedule generation scheme.

Based on the study done by Li and Zhang [21], each ant faces with two decisions during each step: (1) activity sequence (assigning i to activity j (i, j)), then (2) mode assignment for activity execution (i, j, k). The terms τ_{ij} and τ_{ijk} represent the pheromone related to these decisions. Similarly, η_{ij} and η_{ijk} state two related heuristic information with probability p_{ij}, p_{ijk} which is define as follows:

$$\eta_{ij} = 1/(LS_j - ES_j + 1), \qquad \forall j \in E_i, \tag{9}$$

$$\eta_{ijk} = 1/t_{jk}c_{jk}, \qquad \forall k \in M_{ij}. \tag{10}$$

Consider ES_j and LS_j as the earliest and latest start time of activity j. The difference between LS_j and ES_j determines the total float of activity j. Based on η_{ij}, the activity with lower total float has higher priority in comparison to the others. The relation between pheromone and heuristic information could be defined as follows:

$$p_{ij} = \begin{cases} \dfrac{[\tau_{ij}]^{\alpha}[\eta_{ij}]^{\beta}}{\sum\limits_{h}[\tau_{ih}]^{\alpha}[\eta_{ih}]^{\beta}} & \text{if} \quad h \in E_i, \\ 0 & ow, \end{cases} \tag{11}$$

$$p_{ijk} = \begin{cases} \dfrac{[\tau_{ijk}]^{\alpha}[\eta_{ijk}]^{\beta}}{\sum\limits_{m}[\tau_{ijm}]^{\alpha}[\eta_{ijm}]^{\beta}} & \text{if} \quad m \in M_{ij}, \\ 0 & ow, \end{cases} \tag{12}$$

where E_i refers to the set of activities which are allowed to be done at the sequence i in the activity

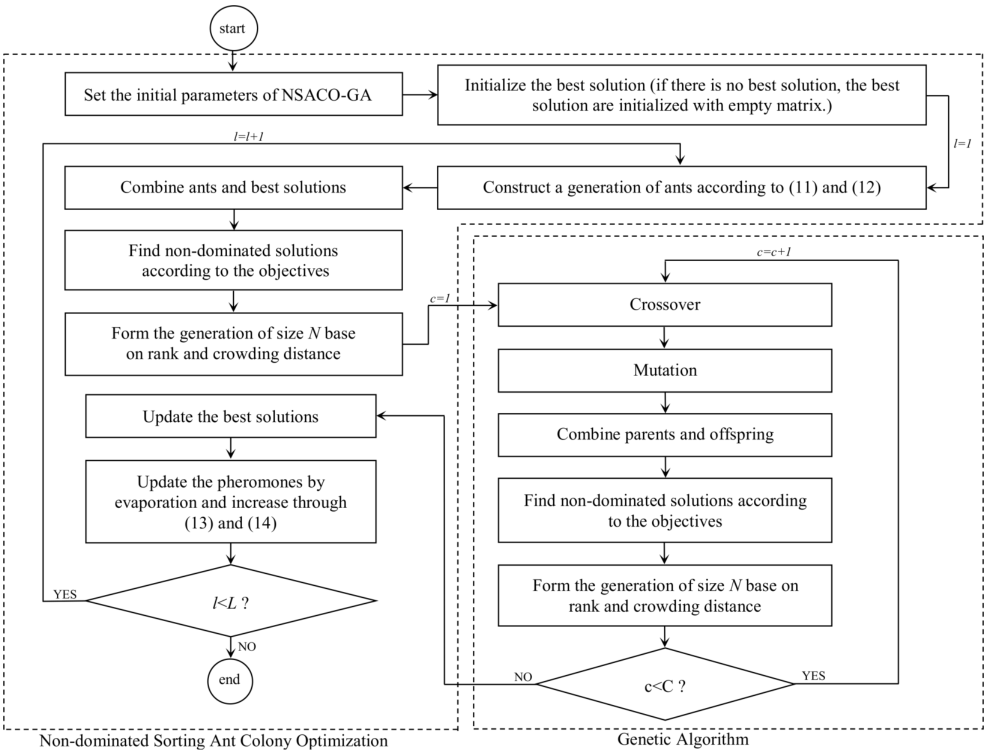

Fig. 1. Flowchart of the proposed algorithm.

list due to the precedence relations, also M_{ij} indicates all possible modes for activity j which is done in sequence i. α and β state the relative importance of pheromone and heuristic information for selecting the next activity to be done. Figure 3 illustrates the structure of two chromosomes of the project which is presented in Fig. 2. In each chromosome, the first row shows the sequence of activities and the second one shows the associated execution modes. The feasibility of created solutions by each ant gets checked regarding the nonrenewable resources constraints. The infeasible solutions are then repaired based on the mode improvement method presented by Peteghem and Vanhoucke [15]. For example, the solution shown in Fig. 3a is an infeasible solution because it needs 15 units of nonrenewable resources that exceed the maximum available amount of nonrenewable resources (14). In Fig. 3b, the infeasible solution changes to a feasible solution (as mentioned above).

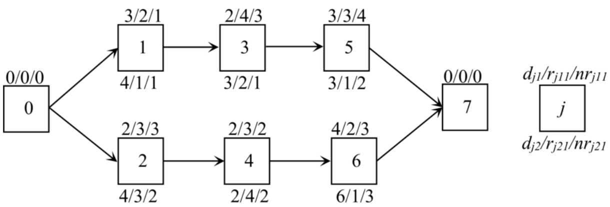

Fig. 2. Activity network of the project instance.

1	2	4	6	3	5
2	1	1	2	1	1

(a) Infeasible schedule

1	2	4	6	3	5
2	2	1	2	2	1

(b) Feasible schedule

Fig. 3. The structure of two chromosomes.

II) The genetic algorithm

A novel algorithm that delivers the benefits of GA and ACO is proposed to solve MOMRCPS. We consider GA developed by Ghoddousi et al. [17]. Suppose N as the number of initial solutions, p_c as crossover probability and p_m as mutation probability. In the proposed algorithm, the single point crossover has been applied. The mutation has been done on both the activity list and the set of assigned modes based on Ghoddousi et al. [17]. In each iteration, solutions are sorted and non-dominated solutions are more likely to pass the next generations.

The non-dominant solutions which outcomes the previous generation, updates as follows:

$$\tau_{ij} = (1-\rho) \times \tau_{ij} + \Delta\tau_{ij}, \tag{13}$$

$$\tau_{ijk} = (1-\rho) \times \tau_{ijk} + \Delta, \tag{14}$$

where ρ is the volatile coefficient of phenomena which indicates the speed of phenomena volatilization. Also, $\Delta\tau_{ij}$ and $\Delta\tau_{ijk}$ represent the amount of released pheromone in (i,j) and (i,j,k), respectively. $\Delta\tau_{ij}^r$ and $\Delta\tau_{ijk}^r$ indicate the pheromone amount released at each step by ant r in (i,j) and (i,j,k), respectively (15)–(18)

$$\Delta\tau_{ij} = \sum_{r=1}^{e} \Delta\tau_{ij}^r, \tag{15}$$

$$\Delta\tau_{ij}^r = \begin{cases} \dfrac{Q}{T^r + C^r + R^r} & (i.j) \in A^r, \\ 0 & ow, \end{cases} \tag{16}$$

$$\Delta\tau_{ijk} = \sum_{r=1}^{e} \Delta\tau_{ijk}^r, \tag{17}$$

$$\Delta\tau_{ijk}^r = \begin{cases} \dfrac{Q}{T^r + C^r + R^r} & (i.j.k) \in M^r, \\ 0 & ow. \end{cases} \tag{18}$$

In which Q is a constant value and e indicates the number of non-dominant solutions. The time, cost, and resource fluctuations for ant r are illustrated by T^r, C^r, and R^r. Moreover, A^r and M^r represent the list of activities and modes that belong to the solution type r.

Numerical examples

In this section, the efficiency of the metaheuristics has been compared on the basis of a computational experiment performed on three series of standard test problems constructed by the ProGen project generator. We considered five problems from each size (small, medium, and large). Table 1 summarizes the features of the project with 10, 20, and 30 number of non-dummy activities. Each non-dummy activity could be performed in three different modes while each mode considers two renewable and two non-renewable resources. Also, project costs including the costs of performing activities under different performing modes, indirect costs, and penalty costs are presented.

The proposed algorithms have been coded in MATLAB 2012. The algorithm parameters are tuned based on experimental analysis. The suitable value has been presented in Table 2.

Table 1
Description of the test problems.

Problem number	Project file name at (PSPLIB)	Number of activities	Size of the problem	Number of arcs	Number of renewable resources	Number of nonrenewable resources	Number of modes
1	j1024_1.mm	10	Small	18	2	2	3
2	j1040_8.mm	10	Small	18	2	2	3
3	j1059_2.mm	10	Small	18	2	2	3
4	j1062_5.mm	10	Small	18	2	2	3
5	j1064_7.mm	10	Small	18	2	2	3
6	j309_6.mm	30	Medium	58	2	2	3
7	j3020_1.mm	30	Medium	58	2	2	3
8	j3025_4.mm	30	Medium	58	2	2	3
9	j3033_2.mm	30	Medium	58	2	2	3
10	j3057_10.mm	30	Medium	58	2	2	3
11	j5017_3.mm	50	Large	193	2	2	3
12	j5031_4.mm	50	Large	193	2	2	3
13	j5066_3.mm	50	Large	193	2	2	3
14	j5094_5.mm	50	Large	193	2	2	3
15	j50108_2.mm	50	Large	193	2	2	3

Table 2
The parameters of algorithms.

	Test problems				Test problems		
NSGA-II	Small	Medium	Large	**NSACO-GA**	Small	Medium	Large
Population size	150	300	400	Number of ants	100	200	250
Maximum iteration	100	200	300	Maximum iteration	25	30	30
Crossover rate	0.7	0.7	0.7	Initial pheromone level	$10Q/J(\bar{t}+\bar{c}+\bar{r})$	$10Q/J(\bar{t}+\bar{c}+\bar{r})$	$10Q/J(\bar{t}+\bar{c}+\bar{r})$
Mutation rate	0.5	0.5	0.5	Maximum iteration of GA	10	15	20

$Q = 10$, $\rho = 0.05$, $\alpha = 1$, $\beta = 0.01$

Comparison metrics

The efficiency of the proposed algorithms has been compared with each other in terms of the accuracy and diversification criteria. These criteria have been presented by Tavana et al. [35] for evaluating the multi-objective optimization methods. These criteria are described as follows:

NNS: The number of non-dominant solutions found by each algorithm (*NSS*) is a convenient criterion for measuring the performance of the algorithms.

Error rate (*ER*): To calculate this criterion, the non-dominant solutions found by both algorithms get merged. Then, the non-dominated solutions are considered as the reference set (*RS*). In this case, *RS* could be defined as the set of best solutions. Note that non-dominant solutions of each algorithm which do not belong to *RS* are known as *ER*. Whatever the *ER* be less, the algorithm would be more efficient.

Generational distance (*GD*): The generational distance could be defined as the difference between *RS* and non-dominant solutions obtained from each algorithm. This criterion is calculated as follows:

$$d_i = \min_{p \in RS} \left\{ \sqrt{\sum_{f=1}^{F} \left(Z_f^i - Z_f^p \right)^2} \right\}, \tag{19}$$

$$GD = \sum_{i=1}^{n} d_i / n. \tag{20}$$

In which F refers to the number of objective functions and n is the number of non-dominant solutions obtained by each algorithm. The algorithm with the lowest value of *GD* is the most efficient one.

Spacing metric (*SM*): This criterion assesses how the uniform is the dispersion of the set of non-dominant solutions for each algorithm. The algorithm with a lower value of *SM* is more efficient. This criterion is calculated as follow:

$$SM = \sqrt{\frac{\sum_{i=1}^{n} (GD - d_i)^2}{n-1}}. \tag{21}$$

Diversification metric (*DM*): This criterion is defined to evaluate how the non-dominant solutions are distributed. This criterion is calculated as follow:

$$DM = \sqrt{\sum_{i=1}^{n} \mathrm{Max}(x_i - y_i)}. \qquad (22)$$

In which $(x_i - y_i)$ is the Euclidean distance between the efficient solutions x_i and y_i. The higher *DM* value indicates the better performance of the algorithm.

Results

In this section, the efficiency of the proposed hybrid algorithm has been assessed using the evaluation criteria given in Table 3. As it is obvious, the numbers of efficient solutions produced by NSGA-II, in some cases, are more than the numbers of efficient solutions produced by the proposed NSACO-GA algorithm. However, most of the solutions produced by this method are dominated by the solutions produced by the proposed hybrid algorithm. In all the test problems, the *ER* for the proposed NSACO-GA is less than that of the NSGA-II algorithm. It could be concluded that the proposed hybrid algorithm is more convergent for the *RS* set. Moreover, the proposed hybrid algorithm sets a lower *GD* metric value. Comparing the *GD* values of both algorithms reveals that the solutions obtained by the NSACO-GA algorithm are closer to the *RS* set. The numerical results of the three criteria *NNS*, *ER*, and *GD* show a higher accuracy of the proposed NSACO-GA in comparison to the NSGA-II. The spacing metric (*SM*) and diversification metric (*DM*) values validate that the proposed hybrid algorithm is more efficient in comparison to the NSGA-II. Figure 4 represents the set of non-dominant solutions which is produced by both NSACO-GA and NSGA-II algorithms in three random examples. It is clear that the non-dominant solutions produced by the proposed hybrid algorithm are denser than those produced by the NSGA-II. The diagram proposed by Nabipoor Afruzi et al. [44] has been applied to CPU time analysis.

Figure 5 and Fig. 6 show the trend of non-dominant efficient solutions which is obtained by both algorithms for three mentioned examples in the given computational time. The trend of the *GD* values for both algorithms is also given. Figure 5 reveals that the numbers of non-dominant efficient solutions which are obtained by the proposed hybrid algorithm are more than those of the NSGA-II. In the same computational time, the *GD* value of the proposed algorithm is lower than the *GD* value of NSGA-II which is showed in Fig. 6. These output results verify that the proposed hybrid algorithm produces more efficient solutions and has a higher powerful search ability at the same time. Based on the experimental results, the performance of the proposed hybrid algorithm is significantly improved and its time complexity is fairly equal compared to the GA and ACO.

Table 3
The computational results for the accuracy and diversity metrics.

Problem number	Accuracy metrics						Diversity metrics			
	NNS		*ER*		*GD*		*SM*		*DM*	
	NSGA-II	NSACO-GA	NSGA-II	NSACO-GA	NSGA-II	NSACO-GA	NSGA-II	NSACO-GA	NSGA-II	NSACO-GA
1	34	30	0.941	0	559.068	0	24.758	0	755.15	692.059
2	13	17	0.692	0.059	320.319	0.586	46.756	0.147	367.72	397.434
3	28	39	0.714	0	344.936	0	18.119	0	704.93	831.568
4	17	28	0.882	0	736.778	0	184.19	0	459.10	416.867
5	26	38	0.615	0.053	346.74	8.617	69.348	1.417	611.23	695.685
6	30	36	0.889	0	892.125	0	98.591	0	1214.9	1136.7
7	14	30	0.929	0	3209	0	670.27	0	839.20	1011.3
8	22	33	0.818	0.061	1640.2	25.766	274.58	4.555	986.73	892.79
9	28	30	0.75	0	371.539	0	42.024	0	1506.8	595.95
10	27	45	0.741	0.133	1055.8	66.058	10.173	7.291	1145.9	1433.8
11	38	47	0.723	0.132	1494.2	147.725	38.357	24.286	2286.1	2081.6
12	23	24	0.75	0.174	2299.4	284.266	479.46	60.606	1859	964.7
13	66	62	0.712	0.242	1301.2	236.353	161.39	30.262	2081.1	2087.8
14	35	32	0.714	0	959.129	0	60.158	0	1632.5	970.89
15	32	52	0.594	0.192	629.411	289.902	56.54	87.774	1472	1621.9

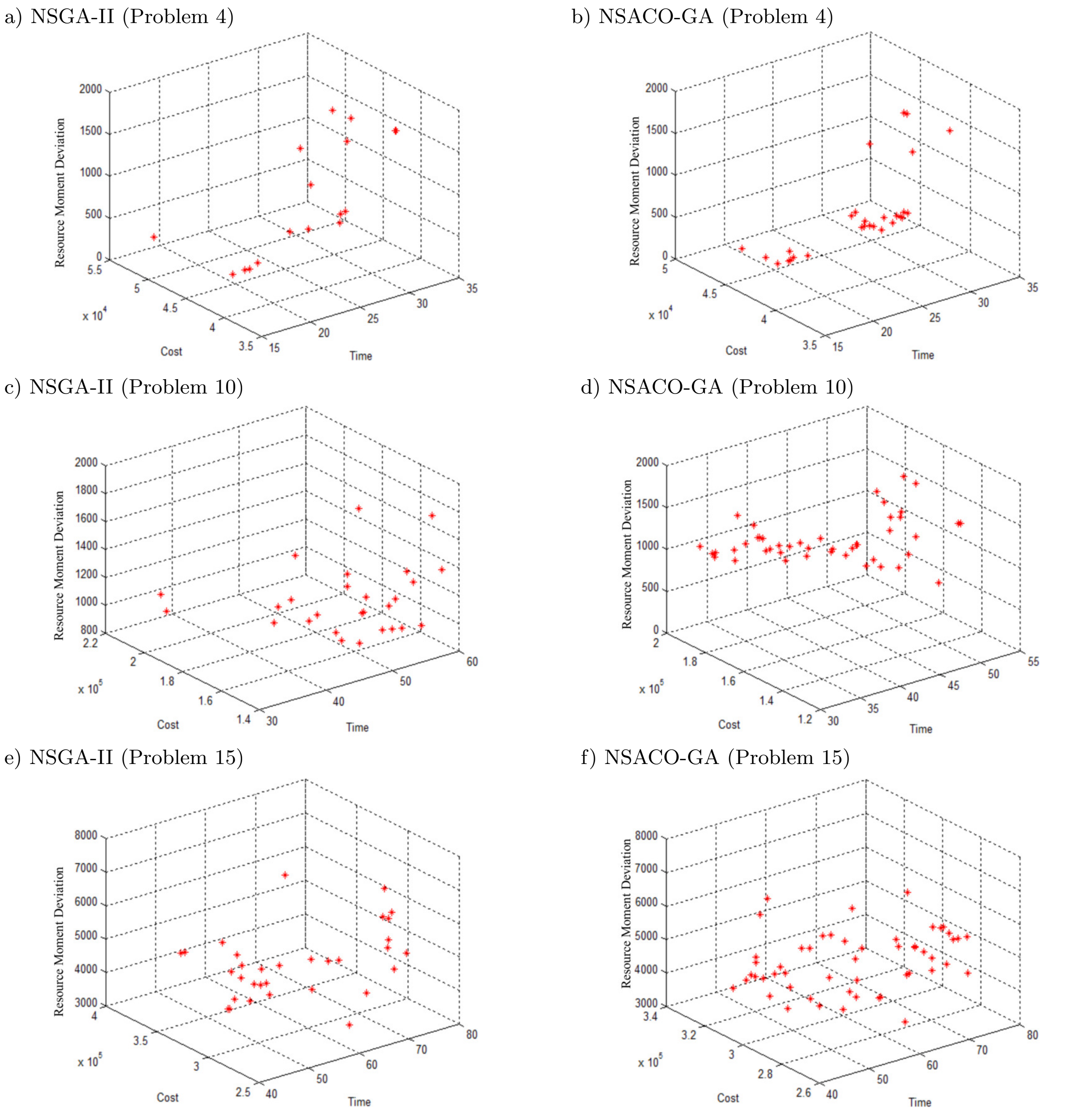

Fig. 4. The generated non-dominated solutions for each methods.

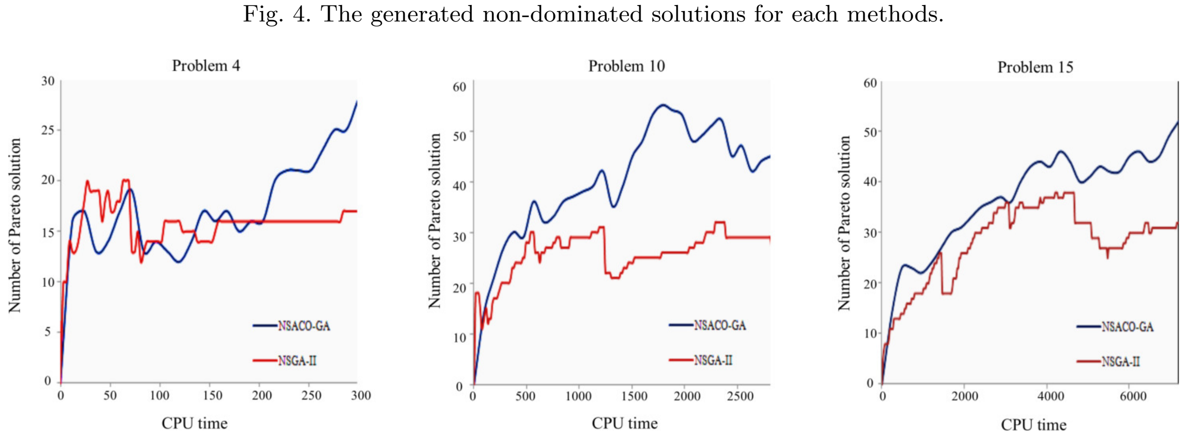

Fig. 5. A trend of non-dominated solutions for each method.

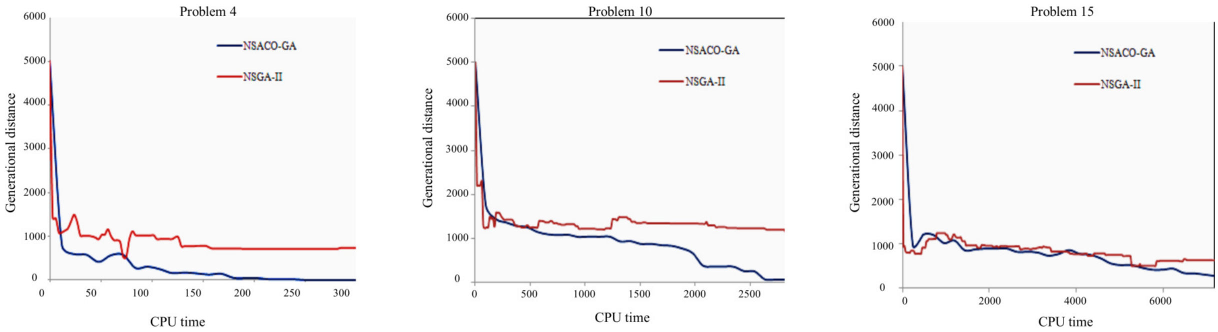

Fig. 6. A trend of *GD* for each method.

Conclusion

This paper studied the multi-objective multi-mode resource-constrained project scheduling problem (MOMRCPSP) with renewable and non-renewable resources to achieve a better balance between time, cost, and resource fluctuations. A hybrid non-dominated sorting genetic-ant colony optimization algorithm has been developed to solve the MOMRCPSP. Since the mentioned model is a multi-objective, the optimal solutions are expressed as Pareto solutions. Test problems (in three groups of small, medium, and large scale) have been solved by the hybrid NSACO-GA and the NSGA-II presented by Ghoddousi et al. [17].

The set of criteria presented by Tavana et al. [35] has been applied as the set of principles for comparing the mentioned algorithms. The set of efficient solutions which are produced by proposed algorithms has been compared with the reference set in response to the convergence and distribution criteria. The numerical results verify the higher efficiency of the proposed hybrid algorithm compared to the NSGA-II algorithm. The proposed hybrid algorithm has been developed to solve a project scheduling problem, but it is obvious that the other algorithms can be used to solve this problem. Some future studies may focus on developing the studied MOMRCPS considering activity interruptions and activity setup times. Also, some studies may focus on developing new strategies which enhance the efficiency of the proposed algorithm such as applying a more efficient parameter tuning strategy or proposing a more desirable initial solution generating procedure. The proposed model has been studied in a specific environment which could be extended to the project scheduling problems under uncertainty.

References

[1] Đumić M., Šišejković D., Čorić R., Jakobović D., *Evolving priority rules for resource-constrained project scheduling problem with genetic programming*, Future Generation Computer Systems, 86, 211–221, 2018.

[2] Elsayed S., Sarker R., Ray T., Coello C.C., *Consolidated optimization algorithm for resource-constrained project scheduling problems*, Information Sciences, 418–419, 346–362, 2017.

[3] Zhi C., Demeulemeester E, Sijun B., Yuntao G., *Efficient priority rules for the stochastic resource-constrained project scheduling problem*, European Journal of Operational Research, 270, 3, 957–967, 2018.

[4] Creemers S. *The preemptive stochastic resource constrained project scheduling problem*, European Journal of Operational Research, 277, 1, 238–247, 2019.

[5] Elmaghraby S.E., *Activity networks: Project planning and control by network models*, Wiley, New York, 1977.

[6] Chand S., Singh H., Ray T., *Evolving heuristics for the resource-constrained project scheduling problem with dynamic resource disruptions*, Swarm and Evolutionary Computation, 44, 897–912, 2019.

[7] Singh G., Ernst A.T., *Resource constraint scheduling with a fractional shared resource*, Operations Research Letters, 39, 5, 363–368, 2011.

[8] Weglarz J., Jozefowska J., Mika M., Waligora G., *Project scheduling with finite or infinite number of activity processing modes – A survey*, European Journal of Operational Research, 208, 3, 177–205, 2011.

[9] Sprecher A., Hartmann S., Drexl A., *An exact algorithm for project scheduling with multiple modes*, OR Spektrum, 19, 3, 195–230, 1997.

[10] Hartmann S., Drexl A., *Project scheduling with multiple modes: a comparison of exact algorithms*, Networks: An International Journal, 32, 4, 283–297, 1998.

[11] Sprecher A., Drexl A., *Multi-mode resource-constrained project scheduling by a simple, general*

and powerful sequencing algorithm, European Journal of Operational Research, 107, 2, 431–450, 1998.

[12] Zhu G., Bard J.F., Yu G., *A branch-and-cut procedure for the multi-mode resource-constrained project scheduling problem*, Informs Journal on Computing, 18, 3, 377–390, 2006.

[13] Ozdamar L., Ulusoy G., *A local constraint based analysis approach to project scheduling under general resource constraints*, European Journal of Operational Research, 79, 2, 287–298, 1994.

[14] Drexl A., Grunewald J., *Nonpreemptive multi-mode resource-constrained project scheduling*, IIE Transactions, 25, 5, 74–81, 1993.

[15] Peteghem V.V., Vanhoucke M., *A genetic algorithm for the preemptive and non-preemptive multi-mode resource-constrained project scheduling problem*, European Journal of Operational Research, 201, 2, 409–418, 2010.

[16] Lova A., Tormos P., Cervantes M., Barber F., *An efficient hybrid genetic algorithm for scheduling projects with resource constraints and multiple execution modes*, International Journal of Production Economics, 117, 2, 302–316, 2009.

[17] Ghoddousi P., Eshtehardian E., Jooybanpour Sh., Javanmardi A., *Multi-mode resource-constrained discrete time-cost-resource optimization in project scheduling using non-dominated sorting genetic algorithm*, Automation in Construction, 30, 216–227, 2013.

[18] Kellenbrink C., Helber S., *Scheduling resource-constrained projects with a flexible project structure*, European Journal of Operational Research, 246, 2, 379–391, 2015.

[19] Baradaran S., Fatemi Ghomi S.M.T., Ranjbar M., Hashemin S.S., *Multi-mode renewable resource constrained allocation in PERT networks*, Applied Soft Computing, 12, 1, 82–90, 2012.

[20] Mika M., Waligóra G., Węglarz J., *Tabu search for multi-mode resource-constrained project scheduling with schedule-dependent setup times*, European Journal of Operational Research, 187, 3, 1238–1250, 2008.

[21] Li H., Zhang H., *Ant colony optimization – based multi-mode scheduling under renewable and nonrenewable resource constraints*, Automation in Construction, 35, 431–438, 2013.

[22] Chen R.M., *Particle swarm optimization with justification and designed mechanisms for resource-constrained project scheduling problem*, Expert Systems with Applications, 38, 6, 7102–7111, 2011.

[23] Pellerin R., Perrier N., Berthaut F.A., *Survey of hybrid metaheuristics for the resource-constrained project scheduling problem*, European Journal of Operational Research, 280, 2, 395–416, 2020.

[24] Tao S., Dong Z.S., *Multi-mode resource-constrained project scheduling problem with alternative project structures*, Computers and Industrial Engineering, 125, 333–347, 2018.

[25] Ke H., Liu B., *Project scheduling problem with stochastic activity duration times*, Applied Mathematics and Computation, 168, 1, 342–353, 2005.

[26] Eshtehardian E., Afshar A., Abbasnia R., *Fuzzy-based MOGA approach to stochastic time-cost trade-off problem*, Automation in Construction, 18, 5, 692–701, 2009.

[27] Kalhor E., Khanzadi M., Eshtehardian E., Afshar A., *Stochastic time-cost optimization using non-dominated archiving ant colony approach*, Automation in Construction, 20, 8, 1193–1203, 2011.

[28] Chen S.P., Tsai M.J., *Time-cost trade-off analysis of project networks in fuzzy environments*, European Journal of Operational Research, 212, 2, 386–397, 2011.

[29] Sonmez R., Bettemir O.H., *A hybrid genetic algorithm for the discrete time-cost trade-off problem*, Expert Systems with Applications, 39, 13, 11428–11434, 2012.

[30] El-kholy A.M., *Time-cost trade-off analysis considering funding variability and time uncertainty*, Alexandria Engineering Journal, 52, 1, 113–121, 2013.

[31] Fen C.W., Liu L., Burns S.A., *Using genetic algorithm to solve construction time-cost trade-off problems*, Journal of Computing in Civil Engineering, 11, 3, 184–189, 1997.

[32] Zheng D., Ng S., Kumaraswamy M., *Applying a genetic algorithm-based multi-objective approach for time-cost optimization*, Journal of Construction Engineering and Management, 130, 2, 168–176, 2004.

[33] Ng S., Zhang Y., *Optimizing construction time and cost using ant colony optimization approach*, Journal of Construction Engineering and Management, 134, 9, 721–728, 2008.

[34] Afshar A., Ziaraty A., Kaveh A., Sharifi F., *Non-dominated archiving multi-colony ant algorithm in time-cost trade-off optimization*, Journal of Construction Engineering and Management, 135, 7, 668–674, 2009.

[35] Tavana M., Abtahi A.R., Khalili-Damghani K., *A new multi-objective multi-mode model for solving preemptive time-cost-quality trade-off project scheduling problems*, Expert Systems with Applications, 41, 4, 1830–1846, 2014.

[36] Ponz-Tienda J.L., Yepes V., Pellicer E., Moreno-Flores J., *The Resource Leveling Problem with multiple resources using an adaptive genetic algorithm*, Automation in Construction, 29, 161–172, 2013.

[37] Nudtasomboon N., Randhawa S.U., *Resource-constrained project scheduling with renewable and non-renewable resources and time-resource trade-offs*, Computers and Industrial Engineering, 32, 1, 227–242, 1997.

[38] Alsayegh H., Hariga M., *Hybrid meta-heuristic methods for the multi-resource leveling problem with activity splitting*, Automation in Construction, 27, 89–98, 2012.

[39] Geng J.O., Weng L.P., Liu S.H., *An improved ant colony optimization algorithm for nonlinear resource-leveling problems*, Computers and Mathematics with Applications, 61, 8, 2300–2305, 2011.

[40] Chiang C.W., Huang Y.Q., Wang W.Y., *Ant colony optimization with parameter adaptation for multi-mode resource-constrained project scheduling*, Journal of Intelligent and Fuzzy Systems, 19, 4, 345–358, 2008.

[41] Dorigo M., *Optimization, learning and natural algorithms*, Ph.D.Thesis Politecnico di Milano, Italy, 1992.

[42] Dorigo M., Birattari M., Stutzle T., *Ant colony optimization*, IEEE Computational Intelligence Magazine, 1, 4, 28–39, 2006.

[43] Akpınar A., Bayhan G.M., Baykasoglu A., *Hybridizing ant colony optimization via genetic algorithm for mixed-model assembly line balancing problem with sequence dependent setup times between tasks*, Applied Soft Computing, 13, 1, 574–589, 2013.

[44] Nabipoor Afruzi E., Roghanian E., Najafi A.A., Mazinani M., *A multi-mode resource-constrained discrete time-cost tradeoff problem solving using an adjusted fuzzy dominance genetic algorithm*, Scientia Iranica, 20, 3, 931–944, 2013.