Asian Journal of Civil Engineering
https://doi.org/10.1007/s42107-018-0048-x

ORIGINAL PAPER

Optimization model for construction project resource leveling using a novel modified symbiotic organisms search

Doddy Prayogo[1] · Min-Yuan Cheng[2] · Foek Tjong Wong[1] · Daniel Tjandra[1] · Duc-Hoc Tran[3]

Received: 10 January 2018 / Accepted: 19 April 2018

Abstract
In the construction industry, determining project schedules has become one of the most critical subjects among project managers. These schedules oftentimes result in significant resource fluctuations that are costly and impractical for the construction company. Thus, construction managers are required to adjust the resource profile through a resource leveling process. In this paper, a novel optimization model is presented for resource leveling, called the "modified symbiotic organisms search" (MSOS). MSOS is developed based on the standard symbiotic organisms search, but with an improvement in the parasitism phase to better tackle complex optimization problems. A case study is employed to investigate the performance of the proposed optimization model in coping with the resource leveling problem. The experimental results show that the proposed model can find a better quality solution in comparison with existing optimization models.

Keywords Construction management · Resource leveling · Optimization · Metaheuristic · Symbiotic organisms search

Introduction

The capability of a construction company to handle resources is essential in order to survive and thrive in today's market (Karaa and Nasr 1986; Wu and An 2012).

✉ Doddy Prayogo
prayogo@petra.ac.id

Min-Yuan Cheng
myc@mail.ntust.edu.tw

Foek Tjong Wong
wftjong@petra.ac.id

Daniel Tjandra
danieltj@petra.ac.id

Duc-Hoc Tran
tdhoc@hcmut.edu.vn

1 Department of Civil Engineering, Petra Christian University, Jalan Siwalankerto 121-131, Surabaya 60236, Indonesia

2 Department of Civil and Construction Engineering, National Taiwan University of Science and Technology, #43, Sec. 4, Keelung Rd., Taipei 106, Taiwan, ROC

3 Department of Construction Engineering and Management, Ho Chi Minh City University of Technology, 268 Ly Thuong Kiet St., Dist. 10, Ho Chi Minh City, Vietnam

Any resource mismanagement could lead to a rise in operational expenses as well as scheduling and financial issues. If extra resources are needed on a construction site, then it could delay the projected completion of the project. When there is such a delay, the owner could sustain a financial loss because the facility is not yet available (Georgy 2008). Furthermore, delays also lead to disagreements between parties, increases in overhead costs, loss in reputation, and, ultimately, total project failure (Arditi and Pattanakitchamroon 2006; Assaf and Al-Hejji 2006). This is why implementing proper resource management in the planning stage of the project is an important task.

Generally, construction resources comprise equipment, materials, manpower, experience, and money. The key to a successful project is effective management of each resource (Georgy 2008). Nonetheless, construction schedules, as the result of network scheduling methods, tend to lead to inefficient, impractical, and expensive resource fluctuations (El-Rayes and Jun 2009). Therefore, construction managers must routinely perform some adjustments to the schedule to decrease the possibility of interrupting fluctuations during the project.

Construction companies are often bothered by these resource fluctuations due to the associated costs of hiring

Published online: 24 April 2018

and laying-off workers, even if it is for a short time (Martinez and Loannou 1993). If resources are not efficiently managed, they may result in the project being incomplete on the specified deadline. Moreover, construction companies need to maintain idle resources even when demand is low. These facts could lead to a profit loss for a construction company.

Resource leveling is a method used to smooth resource fluctuations during a project and one that has received increasing attention by many researchers (Christodoulou et al. 2009; El-Rayes and Jun 2009). The goal of resource leveling is to reduce peak demand and fluctuations in the resource usage pattern. The method looks to reduce variation in the resource profile by altering the non-important activities in their respective places and, thereby, maintaining the project's schedule. There are several methods available such as mathematical, heuristics, and metaheuristics that can solve a resource leveling problem for a construction project.

In the beginning, resource leveling problems were addressed with mathematical methods as they offered the best solutions to a problem. Over time, these techniques were no longer practical, especially for larger projects as resource leveling became a type of combinatorial problem as the increasing number of variables resulted in an infeasible problem-solving method. Therefore, mathematical techniques are no longer ideal for real-life projects. Several studies have used heuristic methods to address the resource leveling issue (Harris 1990). Although resource leveling heuristics are simple and can be used widely with commercial project management software (e.g., Microsoft project), project managers are often left unsatisfied. This is because heuristic methods work on the premise of pre-designated rules and their performance is dependent upon a certain kind of problem and the implemented rules. There is no guarantee for the best solution (Hegazy 1999).

With the limitations of both the heuristic and mathematics methods, researchers have shown an increased interest in adopting metaheuristic methods for solving resource leveling problems (Geng et al. 2011; Leu et al. 2000; Kaveh et al. 2016). Metaheuristic algorithms, inspired by natural phenomena, have been implemented successfully in various construction project problems to address optimization issues (Cheng et al. 2015; Kaveh 2017). These methods use iterative calculations to preserve the randomly initiated population to the best solution. Although metaheuristic algorithms, such as genetic algorithm (GA) (Goldberg 1989), particle swarm optimization (PSO) (Kennedy and Eberhart 1995), and differential evolution (DE) (Storn and Price 1997), play an integral part in the optimization field, they still have their drawbacks. For instance, the biggest challenges for metaheuristic algorithms is the poor exploitation and premature convergence to deal with intricate optimization issues (Geng et al. 2011). Some researchers have boosted the performance of algorithms by using a hybrid approach (Cheng, et al. 2016b; Cheng and Prayogo 2017; Kaveh and Nasrollahi 2013; Kaveh and Ilchi Ghazaan 2018; Prayogo et al. 2018).

Symbiotic organisms search (SOS), which is a population-based searching algorithm, has garnered interest over the past few years across the science and engineering fields (Cheng and Prayogo 2014). SOS is deemed both efficient and effective for continuous global optimization, using three symbiosis-inspired operators to ensure the population had the best global solution. SOS has been used successfully many times over, surpassing other algorithms (Tran et al. 2016; Cheng et al. 2014; Prayogo et al. 2017). This paper presents a new modification of the standard SOS to achieve an acceptable solution to the resource leveling problems in construction projects, the so-called "modified symbiotic organisms search" (MSOS). In MSOS, a new and effective parasitism mechanism is employed to generate a better searching ability by integrating the neighborhood search mechanism to the previous parasitism formula. For validating the performance of MSOS, an actual case study of a construction resource leveling problem is adopted.

The rest of the article is organized as follows: "Literature review" reviews literature related to the establishment of the new optimization model; in "Proposed modified symbiotic organisms search (MSOS) algorithm" and "Practical implementation of MSOS for resource leveling", the overall scenario of the newly proposed optimization model is presented in detail; "Experimental results" demonstrates a numerical experiment and results comparison for the proposed model; finally, "Conclusions" presents the conclusions and a discussion.

Literature review

Formulation of objective function for resource leveling problem

The objective behind finding a solution to the resource leveling problem was to decrease the demand in peak resources and the daily consumption of that resource within the project timeframe and with the assertion that there was an unlimited amount of resources available. This study considers resource leveling as an optimization problem. Here, the following objective function is minimized as follows (Hegazy 1999):

$$f = \sum_{i=1}^{T} (y_i - y_u)^2, \tag{1}$$

where T is the project duration, y_i represents the resource requirements of all activities carried out at time unit i, and y_u is a uniform resource level provided by:

$$y_u = \frac{\sum_{i=1}^{T} y_i}{T}. \tag{2}$$

Son and Skibniewski (1999) further proposed a new formulation of the objective function as follows:

$$f = \sum_{i=1}^{T} y_i^2 - 2y_u \sum_{i=1}^{T} y_i + y_u^2, \tag{3}$$

where y_u and $\sum_{i=1}^{T} y_i$ are constant because the rates of resource and activity duration for each activity are fixed. Accordingly, the following equation expresses the objective function:

$$f = \sum_{i=1}^{T} y_i^2. \tag{4}$$

Generally, as shown in Fig. 1 and drawing upon relevant literature (Hegazy 1999), Eq. (4) equals the minimum moment of the resource histogram around the time axis. Additionally, it is necessary to modify the objective function of the resource leveling problem as the optimization process can produce different scheduling solutions. Furthermore, the objective function values might be identical but the resource fluctuations might differ. Therefore, it is needed to take into consideration the deviations between the peak of resource demand and resource consumption in consecutive time periods (Easa 1989) to determine the most optimal resource profile. This research will later introduce a modified objective function for the resource leveling optimization model.

Metaheuristic applications on resource leveling problem

When it comes to construction management, the problems with resource scheduling are looked at intensely because of their importance. Resource leveling, within resource scheduling, is one of the biggest problems because it is so complex (Hegazy 1999). Mathematical models have been used to try solving the problems (Easa 1989) but are unable to address complex, large-scale problems that people encounter in real life. Heuristic rules were later applied to address the convoluted nature of resource levels (Martinez and Loannou 1993). However, it appeared that this method only worked to solve certain problems. Heuristic rules do not always guarantee a solution, which is a real problem for some practitioners. Therefore, these facts motivate researchers to find other modern techniques, e.g.,

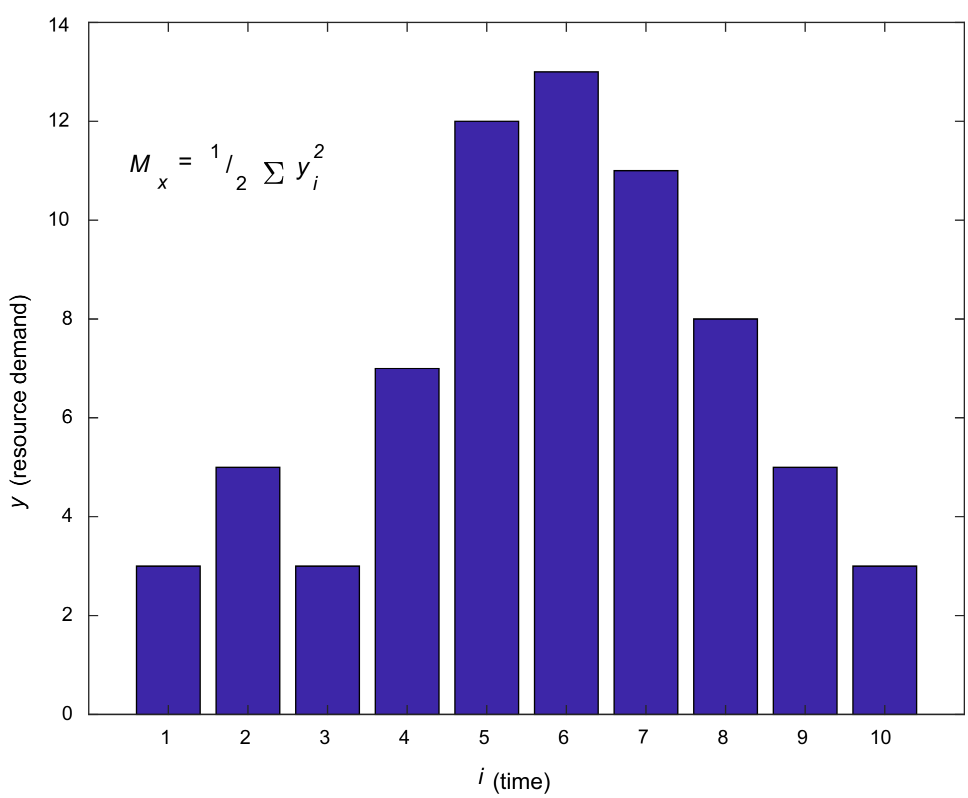

Fig. 1 Moment of resource histogram around the time axis

metaheuristic algorithms, to search for solutions to particular problems (Geng et al. 2011; Leu et al. 2000).

Many studies have been proposed in the past decades to investigate the performance of metaheuristic optimization algorithms in solving different resource leveling problems in construction projects. In an early attempt, Hegazy (1999) proposed a GA to solve a resource leveling model based on minimum moment of resource histogram. Son and Skibniewski (1999) developed a simulated annealing (SA) hybrid model for finding the best solution of the given resource leveling problems. To minimize the total of absolute deviations between each and average resource usage, Leu et al. (2000) employed a GA and further introduces a prototype of decision support system for resource leveling in addition to the optimization model. Khanzadi et al. (2016) applied newly developed algorithms—colliding bodies optimization (CBO) and charged system search (CSS)—for both simultaneously solving the resource allocation and resource leveling problems. Recently, Cheng et al. (2016a) applied a promising symbiotic organisms search (SOS) algorithm to solve multiple-resources leveling problems in multiple construction projects. Although several metaheuristic algorithms have been successfully applied in solving past resource leveling problems to some degree, there is still a need for improvement in terms of the quality and efficiency of the solution, particularly as the problem becomes more complex.

Symbiotic organisms search (SOS) algorithm

SOS is a population-based metaheuristic algorithm proposed by Cheng and Prayogo (Cheng and Prayogo 2014) and designed for continuous optimization. In the SOS algorithm, the three phases (inspired by symbiotic interactions) are performed to lead a population (ecosystem) of candidate solutions (organisms) toward the global optima region in the search space. Mutualism, commensalism, and parasitism are the three symbiotic interactions that SOS uses to modify the candidate solutions. It is expected that the simulation of symbiotic interactions through successive generations improves the fitness value of the organism.

Each phase consists of two operators, called the "interaction operator" and "selection operator". Basically, the interaction operator in each phase is based on the linear combination of two or more different solution/organism vectors. The interaction operator plays a key role for updating the solutions. Meanwhile, the selection operator is employed as a mechanism to preserve the best possible solutions to the next generation. Each phase yields one or more "offspring" vectors that compete with the "parent" vector in the selection process. SOS employs greedy selection, which considers better fitness value as the single criterion in the selection process. Thus, if the offspring vector can yield a lower objective function value than its parent, then the offspring vector supersedes the parent vector (see Fig. 2).

The formulas for mutualism, commensalism, and parasitism phases are explained below:

Mutualism phase

- Interaction operator:

$$x_{i\text{new}} = x_i + \text{rand}(0,1) \times \left[x_{\text{best}} - \left(\frac{x_i + x_{ii}}{2}\right) \times (1 + \text{round}(\text{rand}(0,1)))\right] \tag{5}$$

$$x_{ii\text{new}} = x_{ii} + \text{rand}(0,1) \times \left[x_{\text{best}} - \left(\frac{x_i + x_{ii}}{2}\right) \times (1 + \text{round}(\text{rand}(0,1)))\right]. \tag{6}$$

- Selection operator:

$$x_i = \begin{cases} x_i & f(x_i) \le f(x_{i\,\text{new}}) \\ x_{i\,\text{new}} & \text{otherwise} \end{cases} \tag{7}$$

$$x_{ii} = \begin{cases} x_{ii} & f(x_{ii}) \le f(x_{ii\,\text{new}}) \\ x_{ii\,\text{new}} & \text{otherwise} \end{cases}, \tag{8}$$

where x_i is the i-th organism vector of the ecosystem, x_{ii} is the ii-th organism vector of the ecosystem where $ii \neq i$, x_{best} represents the best organism in the current generation, $x_{i\ \text{new}}$ and $x_{ii\ \text{new}}$ represent candidate solutions for x_i and x_{ii} after their interaction, respectively, $f(x_i)$ is the fitness value of x_i, $f(x_{ii})$ is the fitness value of x_{ii}, $f(x_{i\ \text{new}})$ is the fitness value of $x_{i\ \text{new}}$, and $f(x_{ii\ \text{new}})$ is the fitness value of $x_{ii\ \text{new}}$.

Commensalism phase

- Interaction operator:

$$x_{i\text{new}} = x_i + \text{rand}(-1,1) \times (x_{\text{best}} - x_{ii}) \tag{9}$$

- Selection operator:

$$x_i = \begin{cases} x_i & f(x_i) \le f(x_{i\,\text{new}}) \\ x_{i\,\text{new}} & \text{otherwise} \end{cases}, \tag{10}$$

where x_i is the i-th organism vector of the ecosystem, x_{ii} is the ii-th organism vector of the ecosystem where $ii \neq i$, x_{best} represents the best organism in the current generation, $x_{i\ \text{new}}$ represents candidate solutions for x_i

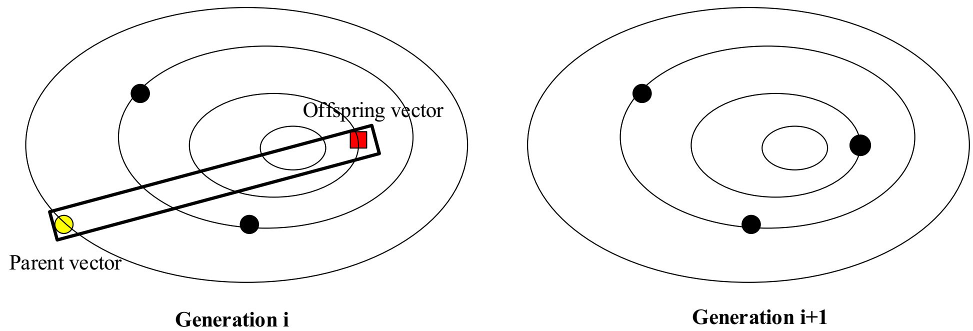

Fig. 2 Greedy criterion-based selection

after the interaction, $f(x_i)$ is the fitness value of x_i, and $f(x_{i\ \text{new}})$ is the fitness value of $x_{i\ \text{new}}$.

Parasitism phase

– Interaction operator:

$$x_{\text{parasite},j} = \begin{cases} x_{i,j} & \text{if rand } (0,1) \le \text{rand } (0,1) \\ \text{LB} + \text{rand}(0,1) \times (\text{UB} - \text{LB}) & \text{otherwise} \end{cases}. \tag{11}$$

– Selection operator:

$$x_{ii} = \begin{cases} x_{ii} & f(x_{ii}) \le f(x_{\text{parasite}}) \\ x_{\text{parasite}} & \text{otherwise} \end{cases}, \tag{12}$$

where x_i is the i-th organism vector of the ecosystem, x_{ii} is the ii-th organism vector of the ecosystem where $ii \neq i$, x_{parasite} is the artificial parasite organism created to compete with the host organism x_{ii}, $x_{i\ \text{new}}$ represents candidate solutions for x_i after the interaction, $f(x_{ii})$ is the fitness value of x_{ii}, $f(x_{\text{parasite}})$ is the fitness value of x_{parasite}, and UB and LB are the upper and lower bounds of the problem, respectively.

When the stopping criteria have been met, the optimization process is complete. The user can dictate what this stopping criterion is, which is usually noted as the maximum number of iterations (maxIter). When complete, the user will see the best possible solution to the problem.

Proposed modified symbiotic organisms search (MSOS) algorithm

In this section, the proposed MSOS algorithm is described in detail. It is noticed that our algorithm is developed based on standard SOS (Cheng and Prayogo 2014) with a modified formulation of the parasitism phase. The local search has proven successful in improving the results of meta-heuristic algorithms (Yu et al. 2017). In those studies, the local search improvement phase used several neighborhood operators to enhance the solution's quality to yield a better objective function value.

Modifications on parasitism phase

In this phase, x_{parasite} are created from x_i. In this situation, x_i^s will be chosen between x_i and x_{ii}. In this new parasitism phase, Eq. (14) was proposed to enhance the local search ability. The selection operator is now updated according to the proposed adaptive crowding concept.

- Interaction operator:
  If rand $(0,1) <$ rand $(0,1)$

$$x_{\text{parasite},j} = \begin{cases} x_{i,j} & \text{if rand } (0,1) \le \text{rand } (0,1) \\ \text{LB} + \text{rand } (0,1) \times (\text{UB} - \text{LB}) & \text{otherwise} \end{cases}, \tag{13}$$

else

$$x_{\text{parasite},j} = \begin{cases} x_{i,j} & \text{if rand}(0,1) \le \text{rand}(0,1) \\ x_{i,j} + (F \times (\text{rand}n(0,1) \times (\text{UB} - \text{LB}) + \text{LB}) & \text{otherwise} \end{cases}, \tag{14}$$

end
- Selection operator:

$$x_i^s = \begin{cases} x_i^s & f(x_i^s) \le f(x_{\text{parasite}}) \\ x_{\text{parasite}} & \text{otherwise} \end{cases}, \tag{15}$$

where x_i is the i-th organism vector of the ecosystem, x_{ii} is the ii-th organism vector of the ecosystem where $ii \neq i$, x_{parasite} is the artificial parasite organism created to compete with the host organism x_i^s, x_i^s represents the most similar parent to x_{parasite}, $f(x_i^s)$ is the fitness value of x_i^s, $f(x_{\text{parasite}})$ is the fitness value of x_{parasite}, UB and LB, respectively, are the upper and lower bounds of the problem, and F is a scaling factor to determine the perturbation size with an initial value of 10^{-5}.

Crowding-based selection operator

As shown before in Eq. (15), this research proposes to replace the original selection operator in the SOS with the crowding-based selection operator (Jong 1975; Mahfoud 1995). The new selection process is proposed to decelerate convergence and to preserve population diversity. Its efficacy in dealing with multimodal optimization problems aimed at locating multiple globally optimal or suboptimal solutions simultaneously has been demonstrated (Das et al. 2011). Generally, considering the replacement policy in crowding, a candidate solution (offspring vector) and the most similar parent compete for a place in the population (Mahfoud 1995). The similarity is measured using Euclidean distance.

If an offspring vector is better than the most similar parent, which is not necessarily the direct parent vector, then the parent is replaced; otherwise, the candidate solution is discarded (see Fig. 3). Hence, besides fitness value, the crowding-based selection operator considers the similarity of individuals as quantified by distances among them. The algorithm basically prefers competition among similar individuals and maintains the diversity of the population. This selection process facilitates the algorithm to explore the search space thoroughly. On the other hand, it is beneficial to keep the traditional greedy selection operator on the previous mutualism and commensalism phases to exploit the currently found solutions.

Practical implementation of MSOS for resource leveling

This section describes the proposed optimization model in solving the resource leveling problem shown in Fig. 4. The objective of this optimization model was to minimize daily fluctuations in resource utilization without altering the total project duration. The resource leveling problem for construction projects is proved to be complex because the objective function landscape may harbor many suboptimal solutions (Cheng et al. 2017; Geng et al. 2011). Furthermore, there can be several scheduling solutions that feature the same resource profiles (Christodoulou et al. 2009). Hence, the resource leveling problem is shown to be both complicated and multimodal. Consequently, the proposed MSOS can provide a potential alternative to deal with the problem at hand.

Resource leveling is accomplished by reducing fluctuations between a desirable uniform resource level and resource requirements. It is important that the activity relationship, resource demand, and activity duration are encompassed by the model. The users must also specify the parameter setting of the search engine such as population size (NP) and maximum number of iterations (maxIter). The inputs allow the scheduling component to perform calculations that attain the critical path method (CPM) based schedule, and early and late starts of each activity. When all the information has been provided, the model can operate efficiently without the need for human intervention. A type of possible solution created by a uniform random generator is required for the search process to begin. A vector with D elements represents a solution to the resource leveling problem:

$$X = [X_{i,1}, X_{i,2}, \ldots, X_{i,D}], \tag{16}$$

where D represents the amount of decision variables related to the problem. It also denotes the amount of activity within the project network. The index i refers to the i-th individual in the population. X represents the start time of D activities. SOS operates in real-value variables in order to change the start times to integer values for the feasible domain.

$$X_{i,j} = \mathrm{Round}(\mathrm{LB}(j) + \mathrm{rand}(0,1) \times (\mathrm{UB}(j) - \mathrm{LB}(j))), \tag{17}$$

where $X_{i,j}$ denotes the start time of activity j at the i-th individual. rand (0,1) is a uniformly distributed random number between 0 and 1. LB (j) and UB (j) provide the early start and late start of the activity j. To find the most optimal project schedule, SOS considers the outcome from the scheduling component and the shifts in non-critical activities within their float times to find the best possible

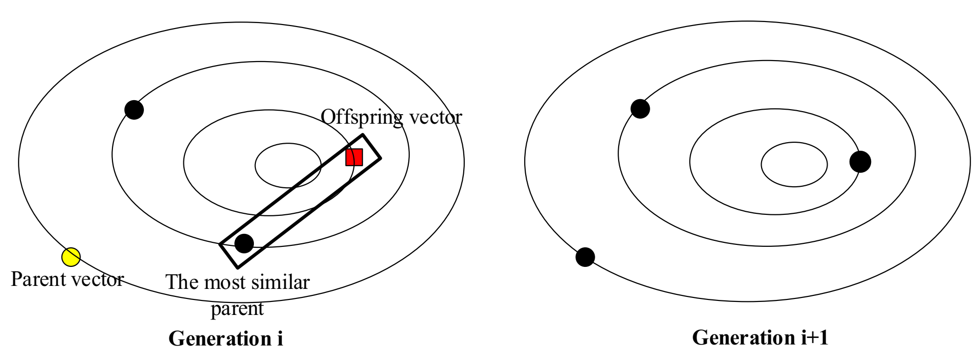

Fig. 3 Crowding-based selection

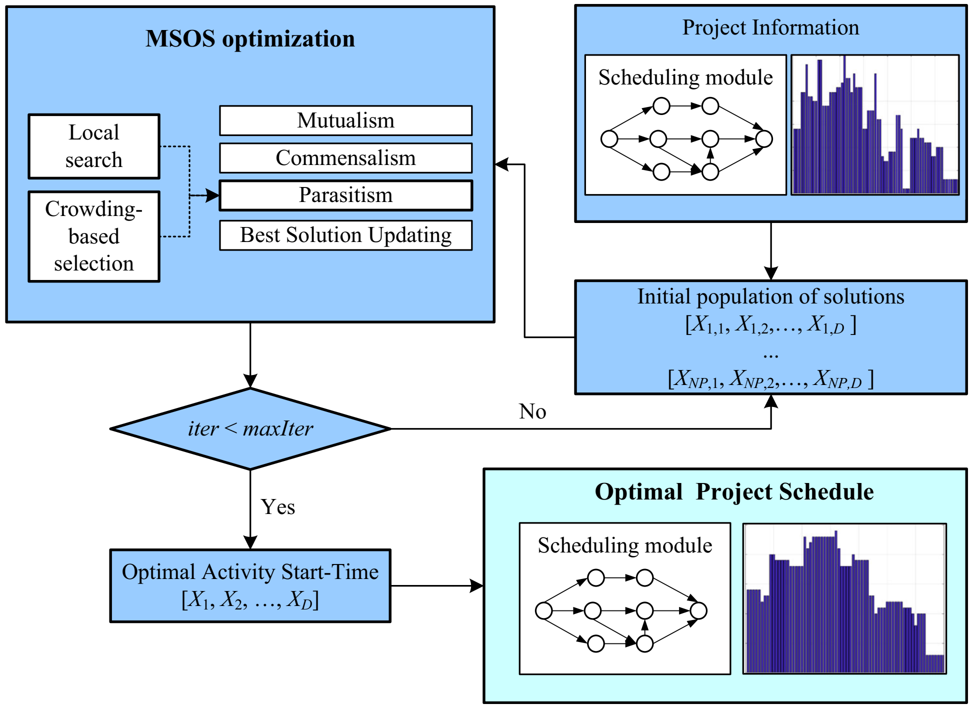

Fig. 4 Flowchart of modified symbiotic organisms search (MSOS) for resource leveling

Table 1 Metrics for performance measurement

Performance metrics	Notation	Calculation
Value of overall fitness function	f	$f = \alpha\frac{1}{2}\sum_{k=1}^{T}(y_k)^2 + \beta\sum_{k=1}^{T-1}(y_{k+1} - y_k) + \gamma \times y_{\max}$ where α, β, and γ are weighting coefficients.
Moment of resource histogram	M_x	$M_x = \frac{1}{2}\sum_{k=1}^{T}(y_k)^2$
Maximum resource demand	$\mathrm{RD}_{\max}$	$\mathrm{RD}_{\max} = y_{\max}$
Cumulative variation of resource demand between consecutive periods	CRV	CRV = $\sum_{k=1}^{T-1}(y_{k+1} - y_k)$
Maximum variation of resource demand between two consecutive periods	$\mathrm{RV}_{\max}$	$\mathrm{RV}_{\max} = \max[(y_2 - y_1), (y_3 - y_2), \ldots, (y_T - y_{T-1})]$ where T is the total project duration

project schedule. In this study, the constraints and objective function are adopted from Cheng et al. (2017) as follows:

$$f = \alpha\frac{1}{2}\sum_{k=1}^{T}(y_k)^2 + \beta\sum_{k=1}^{T-1}(y_{k+1} - y_k) + \gamma \times y_{\max}. \quad (18)$$

Subject to

$$\mathrm{ST}_i - \mathrm{ES}_i \leq \mathrm{TF}_i; \quad \mathrm{ST}_i \geq 0; \quad i = 1, 2, \ldots, D,$$

where T represents the project duration, y_k denotes all resource requirements of activities performed at time unit k, and y_{u} represents a uniform resource level. α, β, and γ are weighting coefficients and are set as 1, 1, and 10, respectively, as suggested by Cheng et al. (2017). $(y_{k+1} - y_{\mathrm{k}})$ determines varying resource usages between two successive time periods. $y_{\max}$ denotes the peak of resource demand during the entire project. ST_i is the start time of activity i. Both ES_i and TF_i denote early start and total float of activity i, respectively. D is the number of activities in the network.

After the searching process terminates, an optimal solution is identified. The project schedule and its corresponding resource histogram are then constructed based on the optimal activities' start time. The user can assess the quality of a project schedule using a set of metrics (see Table 1).

Table 2 Project information

Activity ID	Duration	Predecessors	Daily resource demand	Early start (ES)	Late start (LS)
1	0		0	0	0
2	10	1	5	0	0
3	5	1	2	0	9
4	15	1	3	0	3
5	3	1	2	0	12
6	10	1	2	0	8
7	15	2	6	10	10
8	7	3	10	5	14
9	3	5	6	3	22
10	3	5	2	3	15
11	2	5	2	3	16
12	3	4, 10, 11	6	15	18
13	2	10	1	6	19
14	2	8, 12	5	18	21
15	3	12, 13	2	18	21
16	1	14	6	20	23
17	1	15	7	21	24
18	1	16	7	21	24
19	4	7, 9, 17, 18	13	25	25
20	2	15, 18	9	22	30
21	2	19	4	29	29
22	1	20	6	24	32
23	3	21	8	31	31
24	1	22	3	25	33
25	4	23, 24	8	34	34
26	2	25	7	38	38
27	25	6	10	10	18
28	3	23	6	34	52
29	3	23	2	34	40
30	3	26	9	40	40
31	3	30	10	43	52
32	3	30	3	43	46
33	2	27, 29, 30	4	43	43
34	0	32	0	46	49
35	4	33	1	45	45
36	3	34, 35	12	49	49
37	3	36	12	52	52
38	3	28, 31, 37	3	55	57
39	5	28, 31, 37	8	55	55
40	1	36	2	52	59
41	3	38, 39, 40	10	60	60
42	1	41	3	63	63
43	6	42	3	64	64
44	0	43	0	70	70

Experimental results

Project information

In this section, a construction project adapted from Sears et al. (2008) is used to investigate the capability of the newly developed MSOS model. As mentioned in Table 2, the project consists of 44 activities for 70 days. Manpower is considered as the resource of interest in this study. Figure 5 presents the resource profile of the project prior to resource leveling process.

Optimization results and comparison

This section shows the application of the MSOS model to minimize resource fluctuations significantly. Table 3 presents parameter settings for the MSOS model. Advanced optimization methods—JADE (Zhang et al. 2008), SaDE (Qin et al. 2009), RLDE (Tran and Hoang 2014), and SOS—are used for performance comparison to validate the performance of MSOS. The SOS algorithm used in this study is a discrete adaptation of the original SOS algorithm developed previously by Cheng et al. (2016a). The NP and maxIter of each benchmark algorithm are set to be comparable to those of MSOS.

The optimal solutions, or optimal activities' start times obtained from MSOS and other benchmark algorithms, are listed in the Table 4. The project's resource profiles after optimization are depicted in Fig. 6. Moreover, the optimization result is given in regard to the best result found, the worst result, the mean, and standard deviation throughout 25 simulation iterations to evaluate the stability and accuracy of the benchmark algorithms as shown in Table 5. The optimal results for this case study are found by the MSOS algorithm with the best overall fitness value (f) of 9518.

Table 3 Parameter settings of MSOS

Input parameters	Notation	Setting
Number of decision variables	D	44
Population size	NP	$8 \times D$
Maximum number of iterations	maxIter	1000

Observing from Table 5, the performance of the proposed model is very competitive in terms of stability and accuracy. Considering the overall fitness value, only MSOS is capable of identifying the most desirable objective value of 9518. Results of JADE, SaDE, and SOS are slightly inferior with $f = 9520$, 9520, and 9522, respectively. The average fitness of MSOS is also significantly better than that of other benchmark algorithms. The new optimizer achieves the best average fitness of 9521.40. Meanwhile,

Fig. 5 Project resource profile before resource leveling process

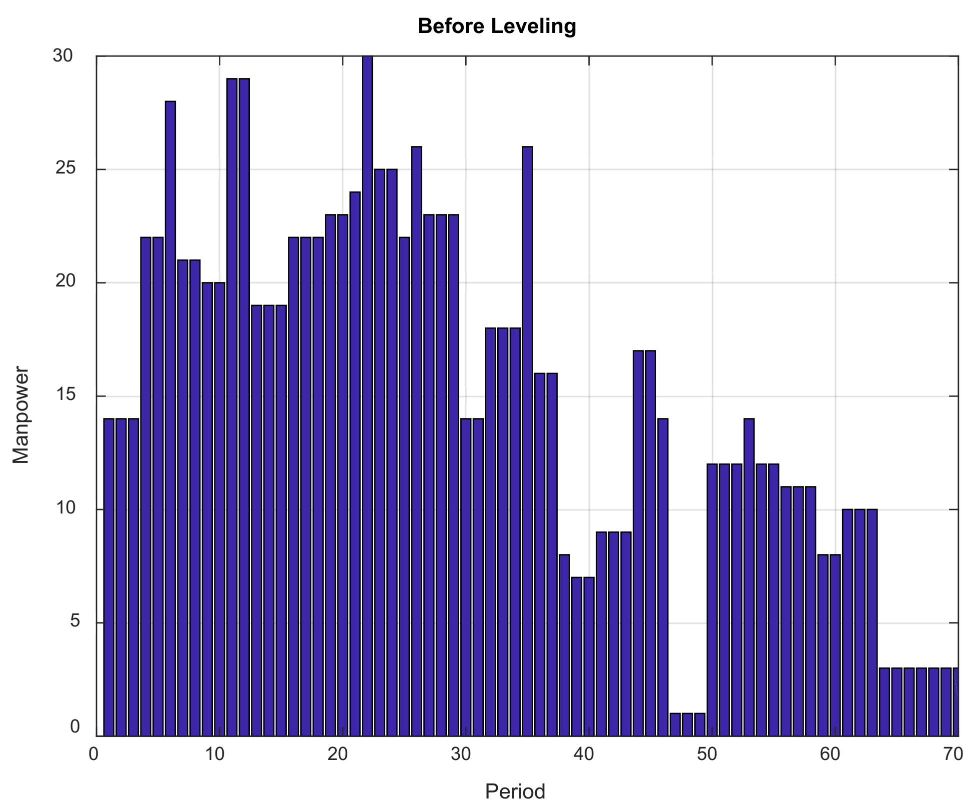

Table 4 Comparison of obtained optimal start times for all activities between MSOS and Benchmark algorithms

Activity ID	Optimal start time for each activity				
	RLDE	JADE	SaDE	SOS	MSOS
1	0	0	0	0	0
2	0	0	0	0	0
3	0	3	3	0	0
4	0	0	0	0	0
5	0	0	0	0	0
6	0	0	0	0	0
7	10	10	10	10	10
8	8	8	8	8	8
9	5	15	15	5	15
10	3	3	3	3	3
11	3	6	6	3	6
12	15	15	15	15	15
13	15	6	6	16	6
14	20	20	20	20	20
15	18	18	18	18	18
16	22	22	22	22	22
17	24	23	23	24	24
18	23	24	24	23	23
19	25	25	25	25	25
20	29	29	29	29	29
21	29	29	29	29	29
22	31	31	31	31	31
23	31	31	31	31	31
24	32	32	32	32	32
25	34	34	34	34	34
26	38	38	38	38	38
27	18	18	18	18	18
28	43	43	43	43	43
29	37	37	37	37	37
30	40	40	40	40	40
31	46	46	46	46	46
32	43	43	43	43	43
33	43	43	43	43	43
34	48	47	47	49	49
35	45	45	45	45	45
36	49	49	49	49	49
37	52	52	52	52	52
38	57	55	55	55	55
39	55	55	55	55	55
40	56	58	58	59	58
41	60	60	60	60	60
42	63	63	63	63	63
43	64	64	64	64	64
44	70	70	70	70	70
f	9524	9520	9520	9522	9518

average fitness values of JADE, SaDE, SOS, and RLDE are 9522.87, 9522.80, 9528.87, and 9534.47, respectively.

Additionally, in terms of the moment of resource histogram (M_x), the maximum resource demand (RD_{max}) and the maximum variation of resource demand between two consecutive periods (RV_{max}), and cumulative variation of resource demand between consecutive periods (CRV), all of the five algorithms have found the optimal values. The best outcomes of M_x, RD_{max}, RV_{max}, and CRV are 9215, 24, 7, and 53, respectively. It is recognizable that the average results and the standard deviations of MSOS in M_x are slightly better than that of other optimization methods while performing equally better in RD_{max} and RV_{max} in comparison with other algorithms. These facts have strongly demonstrated the stability and the accuracy of the new established MSOS model. Furthermore, the convergence curves for MSOS and other benchmark algorithms are shown in Fig. 7. It is shown that MSOS has a better convergence characteristic and is able to achieve the best solution earlier in comparison with other algorithms.

The optimal start times and the results comparison between MSOS and the commonly used project management software Microsoft Project 2007 are shown in Table 6. The comparison of optimized resource profiles between MSOS and Microsoft Project 2007 are depicted in Fig. 8. Obviously, performance of the new model is better than that of the commercial software in which the fitness value of the proposed model (9518) is significantly less than those of the Microsoft Project 2007 (9715). This means that the new model has successfully reduced the resource fluctuation considerably.

Conclusions

This study proposes a new optimization model, namely MSOS, to tackle the complexity of the resource leveling problem. To preserve population diversity, MSOS replaces the original greedy selection operator in standard SOS with a crowding-based selection operator. The crowding technique is crucial as it facilitates the algorithm to explore the search space thoroughly and to preserve the population diversity. By doing so, the possibility of being trapped in a suboptimal solution is diminished considerably. This, in essence, allows the algorithm to adapt itself not only to different optimization problems, but also to various stages during the optimization process. Moreover, to enhance the performance in neighborhood searching ability, the newly developed parasitism mechanism enhances the searching ability in the neighborhood of each organism.

Experimental results and result comparisons have demonstrated that MSOS can deliver accurate and stable results. Additionally, the algorithm operation

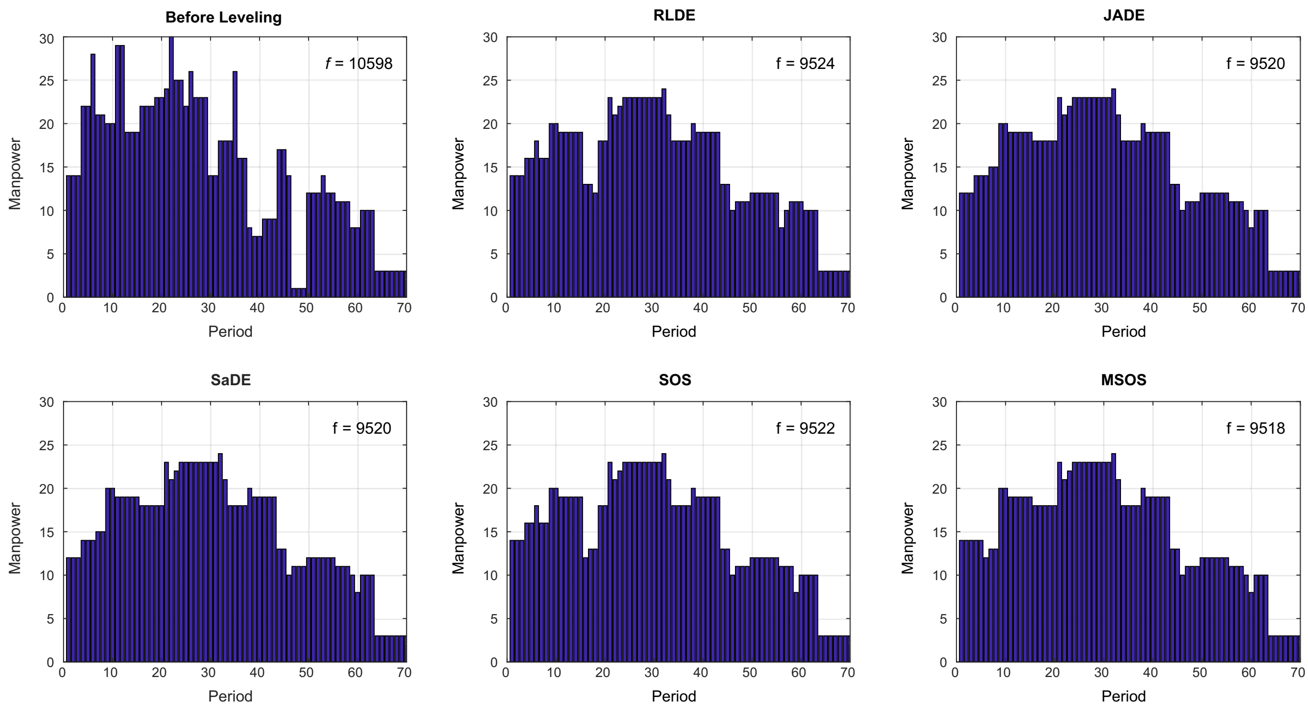

Fig. 6 Project resource profile after being optimized by MSOS and other benchmark algorithms

Table 5 Result comparison between MSOS and Benchmark algorithms

Performance metrics		RLDE	JADE	SaDE	SOS	MSOS
f	Best	9524	9520	9520	9522	9518
	Worst	9542	9528	9534	9538	9524
	Mean	9534.47	9522.87	9522.80	9528.87	9521.40
	Std	5.82	2.21	2.91	6.43	1.40
M_x	Best	9215	9215	9215	9215	9215
	Worst	9227	9227	9227	9227	9227
	Mean	9219.93	9217.40	9218.20	9217.80	9217.40
	Std	5.53	4.88	5.40	5.16	4.53
RD_{max}	Best	24	24	24	24	24
	Worst	24	24	24	24	24
	Mean	24	24	24	24	24
	Std	0	0	0	0	0
RV_{max}	Best	7	7	7	7	7
	Worst	7	7	7	7	7
	Mean	7	7	7	7	7
	Std	0	0	0	0	0
CRV	Best	53	53	53	53	53
	Worst	69	69	69	69	69
	Mean	61.87	64.40	63.67	64.13	64
	Std	6.32	5.83	6.59	6.30	5.72

successfully eliminates human intervention and the trial-and-error process for control parameter settings. These facts have proved that MSOS is a promising tool to assist project managers in dealing with resource leveling problems. The time and resource information in this study is assumed exact although, in the real world, activity time and resources are usually uncertain. Addressing the uncertainty aspects of activity time and resources in the resource leveling problem can become a substantial future research agenda.

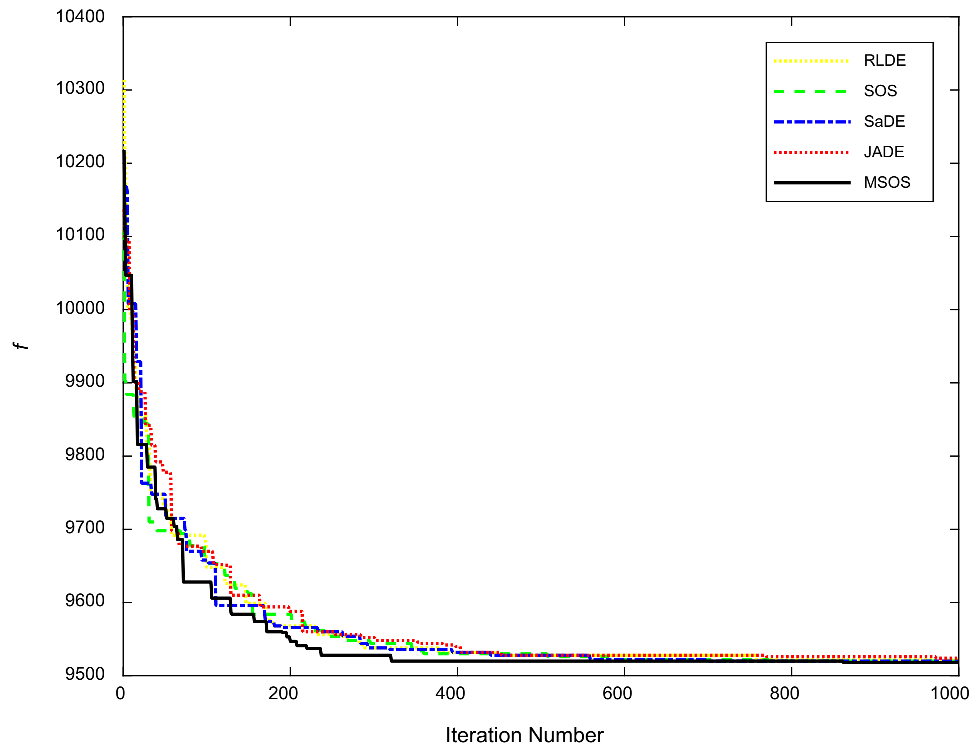

Fig. 7 Convergence curves of MSOS and other benchmark algorithms

Table 6 Comparison of obtained optimal start times for all activities between MSOS and Microsoft project 2007

Activity ID	Optimal start time for each activity	
	Microsoft project 2007	MSOS
1	0	0
2	0	0
3	3	0
4	2	0
5	5	0
6	0	0
7	10	10
8	11	8
9	8	15
10	14	3
11	9	6
12	17	15
13	18	6
14	20	20
15	20	18
16	22	22
17	24	24
18	23	23
19	25	25
20	29	29
21	29	29
22	31	31

Table 6 (continued)

Activity ID	Optimal start time for each activity	
	Microsoft project 2007	MSOS
23	31	31
24	32	32
25	34	34
26	38	38
27	18	18
28	46	43
29	37	37
30	40	40
31	43	46
32	46	43
33	43	43
34	49	49
35	45	45
36	49	49
37	52	52
38	55	55
39	55	55
40	58	58
41	60	60
42	63	63
43	64	64
44	70	70
f	9715	9518

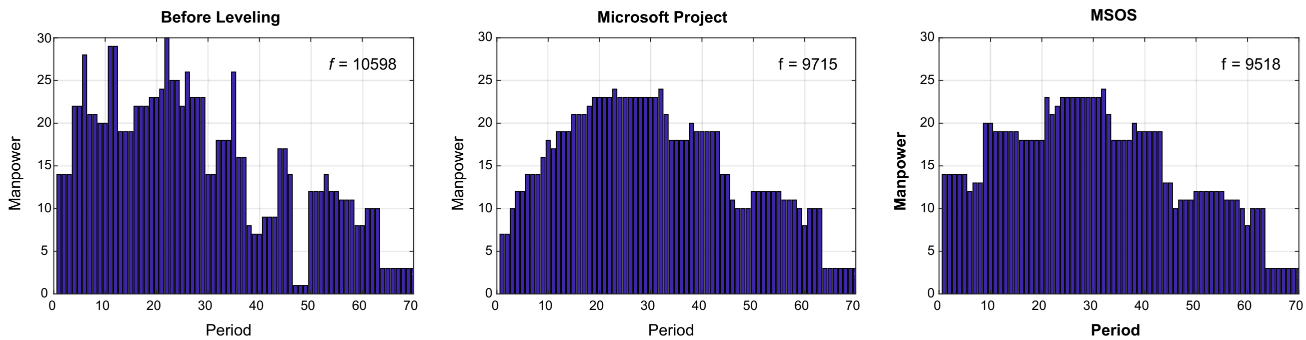

Fig. 8 Project resource profile after being optimized by MSOS and Microsoft project

Acknowledgements The authors gratefully acknowledge that the present research is supported by The Ministry of Research, Technology and Higher Education of the Republic of Indonesia under the "Penelitian Dasar Unggulan Perguruan Tinggi 2018" (PDUPT) Research Grant Scheme (No: 002/SP2H/LT/K7/KM/2017).

References

Arditi, D., & Pattanakitchamroon, T. (2006). Selecting a delay analysis method in resolving construction claims. *International Journal of Project Management, 24*(2), 145–155. https://doi.org/10.1016/j.ijproman.2005.08.005.

Assaf, S. A., & Al-Hejji, S. (2006). Causes of delay in large construction projects. *International Journal of Project Management, 24*(4), 349–357. https://doi.org/10.1016/j.ijproman.2005.11.010.

Cheng, M.-Y., Chiu, C.-K., Chiu, Y.-F., Wu, Y.-W., Syu, Z.-L., Prayogo, D., et al. (2014). SOS optimization model for bridge life cycle risk evaluation and maintenance strategies. *Journal of the Chinese Institute of Civil and Hydraulic Engineering, 26*(4), 293–308.

Cheng, M.-Y., & Prayogo, D. (2014). Symbiotic organisms search: A new metaheuristic optimization algorithm. *Computers & Structures, 139,* 98–112. https://doi.org/10.1016/j.compstruc.2014.03.007.

Cheng, M.-Y., & Prayogo, D. (2017). A novel fuzzy adaptive teaching–learning-based optimization (FATLBO) for solving structural optimization problems. *Engineering with Computers, 33*(1), 55–69. https://doi.org/10.1007/s00366-016-0456-z.

Cheng, M.-Y., Prayogo, D., & Tran, D.-H. (2016a). Optimizing multiple-resources leveling in multiple projects using discrete symbiotic organisms search. *Journal of Computing in Civil Engineering, 30*(3), 04015036. https://doi.org/10.1061/(asce)cp.1943-5487.0000512.

Cheng, M.-Y., Prayogo, D., Wu, Y.-W., & Lukito, M. M. (2016b). A hybrid harmony search algorithm for discrete sizing optimization of truss structure. *Automation in Construction, 69,* 21–33. https://doi.org/10.1016/j.autcon.2016.05.023.

Cheng, M.-Y., Tran, D.-H., & Hoang, N.-D. (2017). Fuzzy clustering chaotic-based differential evolution for resource leveling in construction projects. *Journal of Civil Engineering and Management, 23*(1), 113–124. https://doi.org/10.3846/13923730.2014.982699.

Cheng, M.-Y., Wibowo, D. K., Prayogo, D., & Roy, A. F. V. (2015). Predicting productivity loss caused by change orders using the evolutionary fuzzy support vector machine inference model. *Journal of Civil Engineering and Management, 21*(7), 881–892. https://doi.org/10.3846/13923730.2014.893922.

Christodoulou, S., Ellinas, G., & Michaelidou-Kamenou, A. (2009). Minimum moment method for resource leveling using entropy maximization. *Journal of Construction Engineering and Management, 136*(5), 518–527. https://doi.org/10.1061/(ASCE)CO.1943-7862.0000149.

Das, S., Maity, S., Qu, B.-Y., & Suganthan, P. N. (2011). Real-parameter evolutionary multimodal optimization—A survey of the state-of-the-art. *Swarm and Evolutionary Computation, 1*(2), 71–88. https://doi.org/10.1016/j.swevo.2011.05.005.

Easa, S. (1989). Resource leveling in construction by optimization. *Journal of Construction Engineering and Management, 115*(2), 302–316. https://doi.org/10.1061/(ASCE)0733-9364(1989)115:2(302).

El-Rayes, K., & Jun, D. (2009). Optimizing resource leveling in construction projects. *Journal of Construction Engineering and Management, 135*(11), 1172–1180. https://doi.org/10.1061/(ASCE)CO.1943-7862.0000097.

Geng, J.-Q., Weng, L.-P., & Liu, S.-H. (2011). An improved ant colony optimization algorithm for nonlinear resource-leveling problems. *Computers and Mathematics with Applications, 61*(8), 2300–2305. https://doi.org/10.1016/j.camwa.2010.09.058.

Georgy, M. E. (2008). Evolutionary resource scheduler for linear projects. *Automation in Construction, 17*(5), 573–583. https://doi.org/10.1016/j.autcon.2007.10.005.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Reading, MA: Addison-Wesley Longman Publishing Co., Inc.

Harris, R. (1990). Packing method for resource leveling (pack). *Journal of Construction Engineering and Management, 116*(2), 331–350. https://doi.org/10.1061/(ASCE)0733-9364(1990)116:2(331).

Hegazy, T. (1999). Optimization of resource allocation and leveling using genetic algorithms. *Journal of Construction Engineering and Management, 125*(3), 167–175. https://doi.org/10.1061/(ASCE)0733-9364(1999)125:3(167).

Jong, K. A. D. (1975). *An analysis of the behavior of a class of genetic adaptive systems*. Michigan: University of Michigan.

Karaa, F., & Nasr, A. (1986). Resource management in construction. *Journal of Construction Engineering and Management, 112*(3), 346–357. https://doi.org/10.1061/(ASCE)0733-9364(1986)112:3(346).

Kaveh, A. (2017). *Applications of metaheuristic optimization algorithms in civil engineering*. Cham: Springer.

Kaveh, A., & Ilchi Ghazaan, M. (2018). A new hybrid meta-heuristic algorithm for optimal design of large-scale dome structures. *Engineering Optimization, 50*(2), 235–252. https://doi.org/10.1080/0305215X.2017.1313250.

Kaveh, A., Khanzadi, M., & Alipour, M. (2016). Fuzzy resource constraint project scheduling problem using CBO and CSS algorithms. *International Journal of Civil Engineering, 14*(5), 325–337. https://doi.org/10.1007/s40999-016-0031-4.

Kaveh, A., & Nasrollahi, A. (2013). Engineering design optimization using a hybrid PSO and HS algorithm. *Asian Journal of Civil Engineering (Bhrc), 14*(2), 201–223.

Kennedy, J., & Eberhart, R. (1995). *Particle swarm optimization*. Paper presented at the Proceedings of the IEEE international conference on neural networks, Perth, Australia.

Khanzadi, M., Kaveh, A., Alipour, M., & Aghmiuni, K. H. (2016). Application of CBO and CSS for resource allocation and resource leveling problem. *Iranian Journal of Science and Technology, Transactions of Civil Engineering, 40*(1), 1–10. https://doi.org/10.1007/s40996-016-0004-5.

Leu, S.-S., Yang, C.-H., & Huang, J.-C. (2000). Resource leveling in construction by genetic algorithm-based optimization and its decision support system application. *Automation in Construction, 10*(1), 27–41. https://doi.org/10.1016/S0926-5805(99)00011-4.

Mahfoud, S. W. (1995). *Niching methods for genetic algorithms*. Ph.D. Dissertation, University of Illinois at Urbana-Champaign Champaign, IL, USA.

Martinez, J., & Loannou, P. (1993) Resource leveling based on the modified minimum moment heuristic. *Computing in Civil and Building Engineering*, 287–294.

Prayogo, D., Cheng, M.-Y., & Prayogo, H. (2017). A novel implementation of nature-inspired optimization for civil engineering: A comparative study of symbiotic organisms search. *Civil Engineering Dimension, 19*(1), 36–43.

Prayogo, D., Cheng, M.-Y., Wu, Y.-W., Herdany, A. A., & Prayogo, H. (2018). Differential Big Bang-Big Crunch algorithm for construction-engineering design optimization. *Automation in Construction, 85,* 290–304. https://doi.org/10.1016/j.autcon.2017.10.019.

Qin, A. K., Huang, V. L., & Suganthan, P. N. (2009). Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE Transactions on Evolutionary Computation, 13*(2), 398–417.

Sears, S. K., Sears, G. A., & Clough, R. H. (2008). *Construction project management: A practical guide to field construction management* (5th ed.). New Jersey: Wiley.

Son, J., & Skibniewski, M. J. (1999). Multiheuristic approach for resource leveling problem in construction engineering: Hybrid approach. *Journal of Construction Engineering and Management, 125*(1), 23–31. https://doi.org/10.1061/(ASCE)0733-9364(1999)125:1(23).

Storn, R. M., & Price, K. (1997). Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization, 11,* 341–359.

Tran, D.-H., Cheng, M.-Y., & Prayogo, D. (2016). A novel multiple objective symbiotic organisms search (MOSOS) for time–cost–labor utilization tradeoff problem. *Knowledge-Based Systems, 94,* 132–145. https://doi.org/10.1016/j.knosys.2015.11.016.

Tran, H.-H., & Hoang, N.-D. (2014). A novel resource-leveling approach for construction project based on differential evolution. *Journal of Construction Engineering, 2014,* 7. https://doi.org/10.1155/2014/648938.

Wu, J. I. E., & An, Q. (2012). New approaches for resource allocation via DEA models. *International Journal of Information Technology & Decision Making, 11*(01), 103–117. https://doi.org/10.1142/S0219622012500058.

Yu, V. F., Redi, A. A. N. P., Yang, C.-L., Ruskartina, E., & Santosa, B. (2017). Symbiotic organisms search and two solution representations for solving the capacitated vehicle routing problem. *Applied Soft Computing, 52,* 657–672. https://doi.org/10.1016/j.asoc.2016.10.006.

Zhang, M., Luo, W., & Wang, X. (2008). Differential evolution with dynamic stochastic selection for constrained optimization. *Information Sciences, 178*(15), 3043–3074. https://doi.org/10.1016/j.ins.2008.02.014.