

# Optimization of Resource Leveling Problem under Multiple Objective Criteria Using a Symbiotic Organisms Search

Prayogo, D.<sup>1\*</sup> and Kusuma, C.T.<sup>1</sup>

**Abstract:** Scheduling and resource management is crucial in any construction project. Bad scheduling and resource management can cause delays or cost overruns. Optimization in solving resource leveling is necessary to avoid those problems. A total of nine objective criteria are used to solve resource leveling. Each of them has the same objective, which is to reduce the fluctuation of resource demand of the project. This study compares the performance of symbiotic organisms search (SOS) and particle swarm optimization (PSO) in solving resource leveling problems using separate objective functions in order to find which one produces a better solution. Those nine objective functions generate differing resource demand diagrams since each of them minimizes differing parameters. The results show that SOS produced a better solution in eight of the nine objective functions that are used, and only one objective function produced the same objective value in both SOS and PSO. Further finding reveals that one objective function is better in solving resource leveling than the others.

**Keywords:** Resource leveling; optimization; metaheuristic algorithms; particle swarm optimization; symbiotic organisms search.

## Introduction

In any construction project, proper scheduling is a necessity for success. Several methods have been used in scheduling, such as the program evaluation and review technique (PERT), the critical path method (CPM), and Gantt charts [1]. Since 1950, CPM and PERT have been the most popular methods for planning and controlling schedules of large-scale construction projects [2]. In spite of the fact that CPM is commonly used, it still has many shortcomings. CPM assumes that the available resources are unlimited, which may lead to fluctuation of resource demand. It is impractical and expensive to hire and fire construction workers in the short term in order to meet the fluctuating resource requirements [3], hence this condition may lead to a cost overrun. Therefore, resources should be managed and allocated efficiently to prevent project delay or cost overrun.

Resource leveling is a method of allocating resources in order to reduce the fluctuation of resource demand. This can be done by shifting the start times of non-critical activities within the available float, such that the project duration remains unchanged [1].

One of the earliest resource leveling methods was developed by Burgess and Killebrew [4], in which the sum of the squares of the resource usage was minimized in order to have a smooth rectangle-shaped resource histogram. Wagner et al. [5] proposed several objective criteria for resource leveling, including minimization of the sum of absolute deviations in resource usage for a determined time interval, and minimizing the maximum resource usage for a determined time interval. Later, Hossein Hashemi Doulabi et al. [6], Easa Said [7], and Leu et al. [2] proposed a new objective criterion for resource leveling which is minimization of the sum of the absolute deviations between resource usage for a determined time interval.

At first, mathematical approaches were used to solve resource leveling problems. However, these approaches become impractical when solving large-scale construction projects. Later, heuristics methods were used [8,9], but they still do not produce optimal results [10]. The shortcomings of both mathematical and heuristic methods encouraged many researchers to study metaheuristic approaches in order to find more reliable optimization alternatives in solving resource leveling problems.

Many studies have used some type of metaheuristic method as an alternative to resource leveling, such as genetic algorithm (GA) [2,6], ant colony organization (ACO) [11], particle swarm optimization (PSO) [12], and differential evolution (DE) [10]. However, these methods each have their own drawbacks; one of them is much relying on parameter tuning too much. If the parameter setting is not optimized, more computational time is required to

<sup>1</sup> Department of Civil Engineering, Faculty of Civil Engineering and Planning, Petra Christian University, Jl. Siwalankerto 121-131, Surabaya 60236, INDONESIA

\*Corresponding author; email: prayogo@petra.ac.id

**Note:** Discussion is expected before June, 1<sup>st</sup> 2019, and will be published in the "Civil Engineering Dimension", volume 21, number 2, September 2019.

Received 08 March 2019; revised 14 March 2019; accepted 19 March 2019

find the optimal solution of the problem. Furthermore, the solutions produced by these metaheuristic algorithms might converge to a local optima region.

The past years have seen an increased use of the symbiotic organisms search (SOS) when solving multiple optimization problems in various research fields [13-18]. The searching operators of SOS take inspiration from the phenomenon of organisms' interaction in nature. Unlike most population-based metaheuristic algorithms, SOS does not require any specific parameter tuning since it only uses common parameters, such as population size and iteration numbers, to operate. SOS has three nature-inspired optimization routines to iteratively obtain optimal solutions, which are the mutualism phase, the commensalism phase, and the parasitism phase. Although SOS has been proven to surpass the performance of GA, DE, and PSO in some problems [13], its performance in solving other problems, such as resource leveling, needs to be tested further.

This research investigates the performance of the SOS optimization method in solving the resource leveling problem under multiple objective criteria. A CPM network resource diagram from a construction project is adopted as a case study of this resource problem. A total of nine objective criteria collected from previous studies are used to analyze the consistency of SOS optimization performance under multiple scenarios. These objective functions have the same main purpose, which is to create an even flow of resource demands without changing the project's duration. For comparison purposes, PSO is used as a benchmark for SOS.

### Metaheuristic Algorithms Application on Solving Resource Leveling Problem

Because of its complexity, resource leveling is now considered one of the major issues in construction [19]. Researchers have used mathematical and heuristic methods for dealing with resource leveling [7-9] but it is not practical when solving the more complex problems that people face in real life. Lately, metaheuristic methods have been used to solve resource leveling, yet improvement is still needed to produce a better solution and solve more complex problems. This research investigates the application of SOS alongside PSO in solving the resource leveling problem. These two methods are briefly introduced in the following section.

#### Particle Swarm Optimization

In 1995, Kennedy and Eberhart developed a swarm-intelligence based algorithm, so-called PSO [20]. It simulates the animal social behavior displayed when flocking to a desired place. PSO can be applied to continuous multi-dimensional functions [20]. A particle is used to represent a solution, where the initial

particles are generated randomly. This algorithm is started with initial locations of random particles, which are updated using the velocity vector formulated in Equation (1). Later, each particle moves to a new location using Equation (2).

$$V_i(t+1) = W * V_i(t) + c_1 * (pbest_i - X_i(t)) * rand(0,1) + c_2 * (gbest - X_i(t)) * rand(0,1) \quad (1)$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad (2)$$

where  $W$  is the inertia weight parameter,  $V_i$  is the velocity vector of particle  $i$ ,  $c_1$  is the cognitive factor,  $pbest_i$  is the personal local best location of  $i$ -th particle,  $X_i$  is the location of particle  $i$ ,  $c_2$  is the social factor,  $gbest$  is the location of global best among all swarms of particles, and  $t$  is the number of iterations. At the end of the process, the best solution is represented by the global best particle. The pseudo-code of PSO is shown in Figure 1.

---

#### Algorithm 1. Particle Swarm Optimization

---

```

1: Initializatoptm
2: For iter = 1 to maximum iteration
3:   For each swarm in the population
4:     Evaluate the fitness of each particle
5:     Update personal and global fitness and location
6:     Update velocity and location of particle
7:   End for
8: End for

```

---

Figure 1. Pseudo-Code of PSO

#### Symbiotic Organisms Search

In 2014, Cheng and Prayogo developed SOS as a new and promising metaheuristic algorithm [13]. This algorithm takes its inspiration from symbiosis interactions of living organisms in nature. The three phases used in SOS, the mutualism phase, the commensalism phase, and the parasitism phase, are described as follows:

During the mutualism phase, organism  $X_i$  interacts with  $X_j$  through Equation (3) and Equation (4).

$$X_{i\text{new}} = X_i + rand(0,1) * (X_{\text{best}} - (X_i + X_j) / 2 * (1 + rand(rand(0,1)))) \quad (3)$$

$$X_{j\text{new}} = X_j + rand(0,1) * (X_{\text{best}} - (X_i + X_j) / 2 * (1 + rand(rand(0,1)))) \quad (4)$$

where,  $X_i$  is the  $i$ -th organism in the ecosystem, and  $X_j$  is the  $j$ -th organism in the ecosystem where  $X_i \neq X_j$ .  $X_{\text{best}}$  represent the best organism in the ecosystem. If the objective value of  $X_{i\text{new}}$  is better than  $X_i$ ,  $X_i$  will be replaced by  $X_{i\text{new}}$ , as with  $X_j$  and  $X_{j\text{new}}$ .

During the commensalism phase, organism  $X_i$  interacts with  $X_j$  through Equation (5).

$$X_{i\text{new}} = X_i + rand(-1,1) * (X_{\text{best}} - X_j) \quad (5)$$

where,  $X_i$  is the  $i$ -th organism in the ecosystem, and  $X_j$  is the  $j$ -th organism in the ecosystem where  $X_i \neq X_j$ .  $X_{\text{best}}$  represent the best organism in the eco-

system. If the objective value of  $X_{inew}$  is better than  $X_i$ ,  $X_i$  will be replaced by  $X_{inew}$ .

During the parasitism phase, *parasite\_vector* is created by duplicating organism  $X_i$ , and *parasite\_vector* is randomly modified to mark *parasite\_vector* off from  $X_i$ . Then,  $X_j$  is selected randomly from the ecosystem. If the objective value of *parasite\_vector* is better than  $X_j$ ,  $X_j$  will be replaced by the *parasite\_vector*. Otherwise,  $X_j$  remains in the ecosystem and the *parasite\_vector* dies.

The three phases of SOS are repeated until the maximum numbers of iteration or termination criteria are met. The pseudo-code of SOS is shown in Figure 2.

**Algorithm 2** Symbiotic Organisms Search

- 1: Initialization
- 2: **For**  $iter = 1$  to *maximum iteration*
- 3:   **For** *each organism* in the ecosystem
- 4:     Interact with random organism within mutualism phase
- 5:     Interact with random organism within commensalism phase
- 6:     Interact with random organism within parasitism phase
- 7:     Memorize the best organism
- 8:   **End for**
- 9: **End for**

**Figure 2.** Pseudo-Code of SOS

**Resource Leveling Problem**

**Mathematical Optimization Model**

Resource leveling is a commonly known scheduling method to minimize resource demand fluctuation by shifting non-critical activities start times within their float. The main objective of resource leveling is to smoothen the resource demand diagram by creating a uniform resource flow without changing the project duration. Resource leveling focuses on decreasing the deviation between peak resource demands and daily resource demands, which is the objective function ( $Z$ ) of resource leveling.

$$\begin{aligned} \text{Min} \quad & Z = |R_{i+1} - R_i| & (6) \\ \text{Subject to} \quad & ES_x \leq ST_x \leq LS_x, & (7) \\ & ST_x \geq 0 & (8) \\ & x = 1, 2, \dots, m & (9) \end{aligned}$$

where  $R_i$  is the resource demand on day  $i$ ,  $ES_x$  is the early start time of activity  $x$ ,  $ST_x$  is the start time of activity  $x$ ,  $LS_x$  is the late start time of activity  $x$ , and  $m$  is the total number of activities.

The literature review on resource leveling indicated that nine objective functions have been used to solve resource leveling [1] as shown in Table 1.

**Table 1.** Objective Function for Resource Leveling

No	Objective Criteria	Formula	References
1	Minimize the sum of the absolute deviations in resource usage	$Z = \min \sum_{i=1}^T ( Rdev_i )$	[5,7]
2	Minimize the sum of the only increases in resource usage	$Z = \min \sum_{i=1}^T ( Rinc_i )$	[5]
3	Minimize the sum of the absolute deviations between resource usage for a determined time interval and the average resource usage	$Z = \min \sum_{i=1}^T ( R_i - Arr )$	[2,5-7]
4	Minimize the maximum resource usage	$Z = \min[\max(R_i)]$	[5]
5	Minimize the maximum deviation in resource usage	$Z = \min[\max(Rdev_i)]$	[5]
6	Minimize the maximum absolute deviation between resource usage	$Z = \min[\max  R_i - Arr ]$	[21]
7	Minimize the sum square of resource usage	$Z = \min \sum_{i=1}^T (R_i^2)$	[3,4,8]
8	Minimize the sum of the square of the deviation in resource usage	$Z = \min \sum_{i=1}^T (Rdev_i)^2$	[21]
9	Minimize the sum of the square of deviations between daily resource usage and the average resource usage	$Z = \min \sum_{i=1}^T (R_i - Arr)^2$	[21]

where  $T$  is the project duration,  $i$  is the determined time interval (day, week, etc),  $R_i$  is the resource demand on  $i$ ,  $Rdev_i$  is deviation of the resource demand on  $i$  and  $i+1$ ,  $Rinc_i$  is the increase of resource demand on  $i$  and  $i+1$ ,  $Arr$  is the average of resource demand during the project duration. In this study, a decision variable is represented by early start time of each activity. A resource demands' diagram can be calculated in accordance to the decision variable. Using the information of resource demand per duration, the objective value can be obtained by calculating the objective function shown in Equation (1) and Table 1 while satisfying the three constraints shown in Equations (7)-(9).

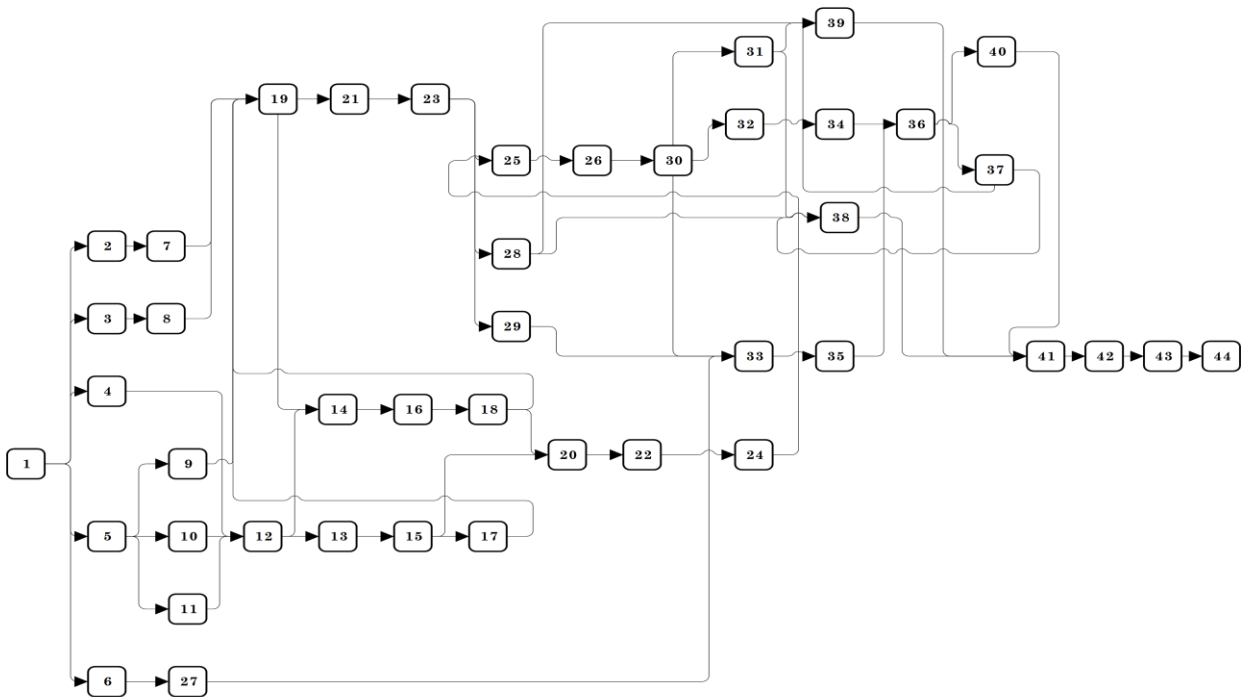


Figure 3. CPM Network Diagram

**Experimental Results and Discussions**

**Project Information and Experimental Settings**

This research compares the performance of SOS and PSO in solving the resource leveling problem with nine objective criteria for resource leveling used in previous literatures. An example of a construction project case study with 44 activities is adopted from Sears et al. [22] for this study. Each algorithm is simulated 30 times with 100 iterations. Details of the project are shown in Table 2 and the CPM network diagram can be seen in Figure 3. The resource demands' diagram before leveling is shown in Figure 4. The parameter settings of each metaheuristic algorithm can be found in Table 3.

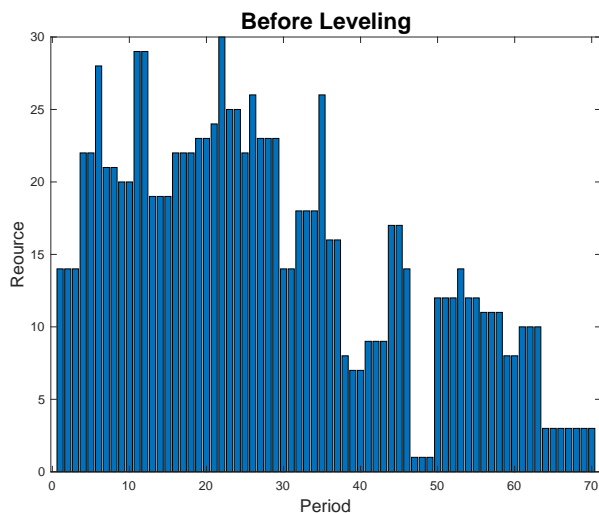


Figure 4. Before Leveling Resources Diagram

Table 2. General Information of Adopted Case Study

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

**Table 3.** Parameter Settings

PSO	SOS
$c1 = 1$	Population size = 352 Iteration numbers = 100
$c2 = 2$	
$w = 0.5$	
Population size = 352	
Iteration numbers = 100	

**Table 4.** Comparison of Objective Value Obtained by SOS and PSO

Objective Function	SOS	PSO
1	Best	80.00
	Worst	114.00
	Mean	66.55
	Std	5.51
2	Best	31.00
	Worst	50.00
	Mean	25.00
	Std	2.15
3	Best	346.69
	Worst	359.54
	Mean	346.79
	Std	3.43
4	Best	23.00
	Worst	24.00
	Mean	23.35
	Std	0.49
5	Best	7.00
	Worst	9.00
	Mean	7.00
	Std	0.00
6	Best	12.11
	Worst	12.11
	Mean	12.11
	Std	0.00
7	Best	18528.00
	Worst	18718.00
	Mean	18434.40
	Std	28.43
8	Best	340.00
	Worst	569.00
	Mean	227.35
	Std	39.02
9	Best	2519.09
	Worst	2731.09
	Mean	2436.29
	Std	17.80

\*Emboldened number indicates best result

### Optimization Results

This section shows the performance of SOS and PSO in solving the resource leveling problem and the comparison between nine objective functions. Between these two algorithms, the population size and the maximum number of iterations are set equal for a fair comparison. The optimization result is achieved through 30 simulation runs to evaluate the consistency and accuracy of the algorithms as shown in Table 4 and Figure 5 respectively. The best objective value founded in the last iteration of each run will be collected. Furthermore, the best, worst, average, and standard deviation values were calculated based from the 30 objective values obtained from simulation runs. Observing from Table 4, the performance of SOS is better than PSO. In eight out of nine different objective functions used, the average fitness and the standard deviation of SOS is significantly better than PSO. However, SOS and PSO obtained equal objective value using objective function number six with a 12.11 average of objective value, and 0.00 standard deviation. Figure 4 shows that in some objective functions, SOS achieves its optimal solution earlier than PSO, such as on objective functions 3, 4,

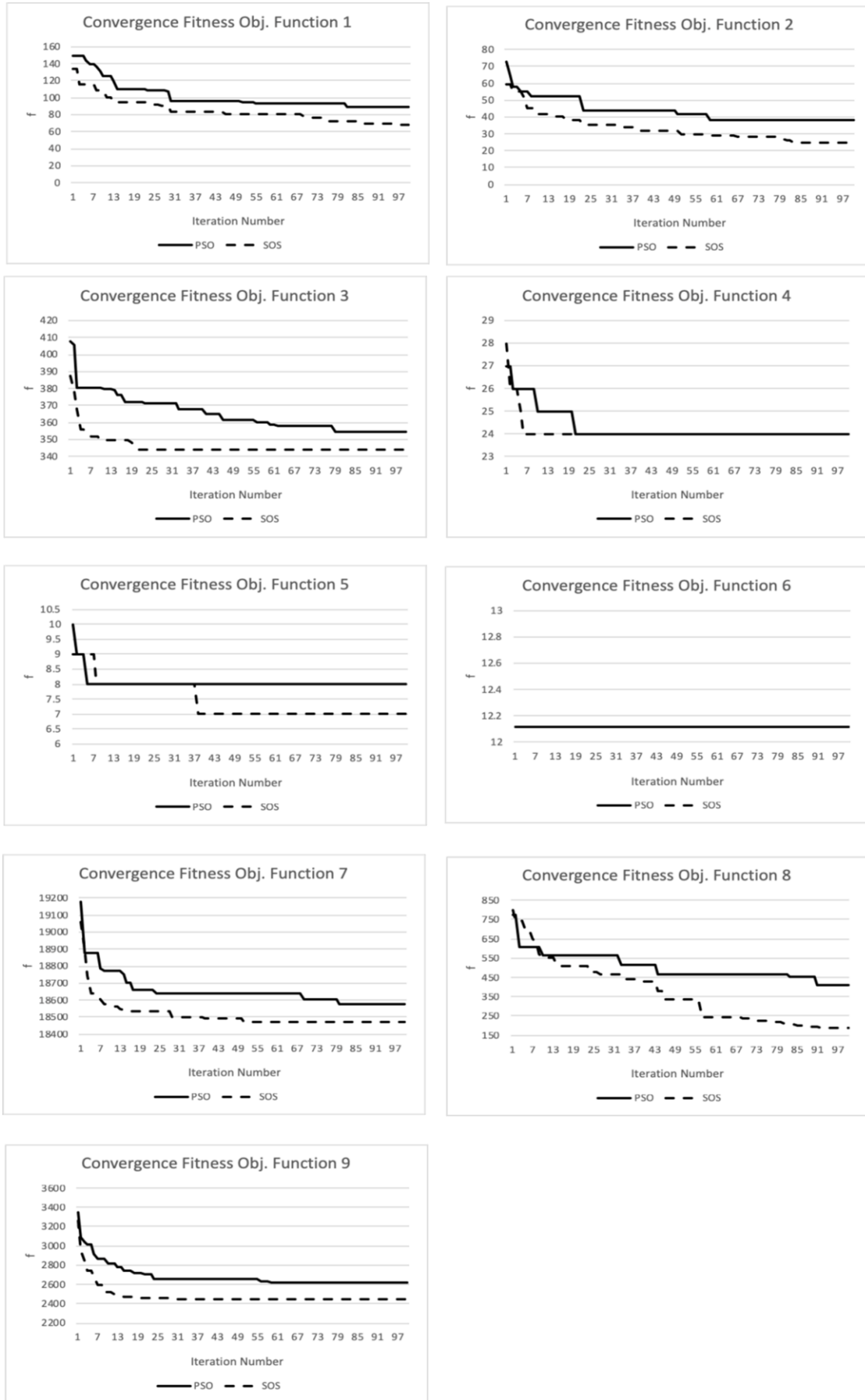
5, 7, and 9, while PSO achieves its optimal solution earlier on objective functions 1, 2, and 8. Although PSO is able to achieve its optimal solution earlier on some functions, the final objective value of the solution is not better than SOS. Hence, it can be stated that SOS is better than PSO in solving the resource leveling problem. After leveling, resource diagrams using the SOS and PSO are shown in Figures 6 and 7.

### Results and Discussions

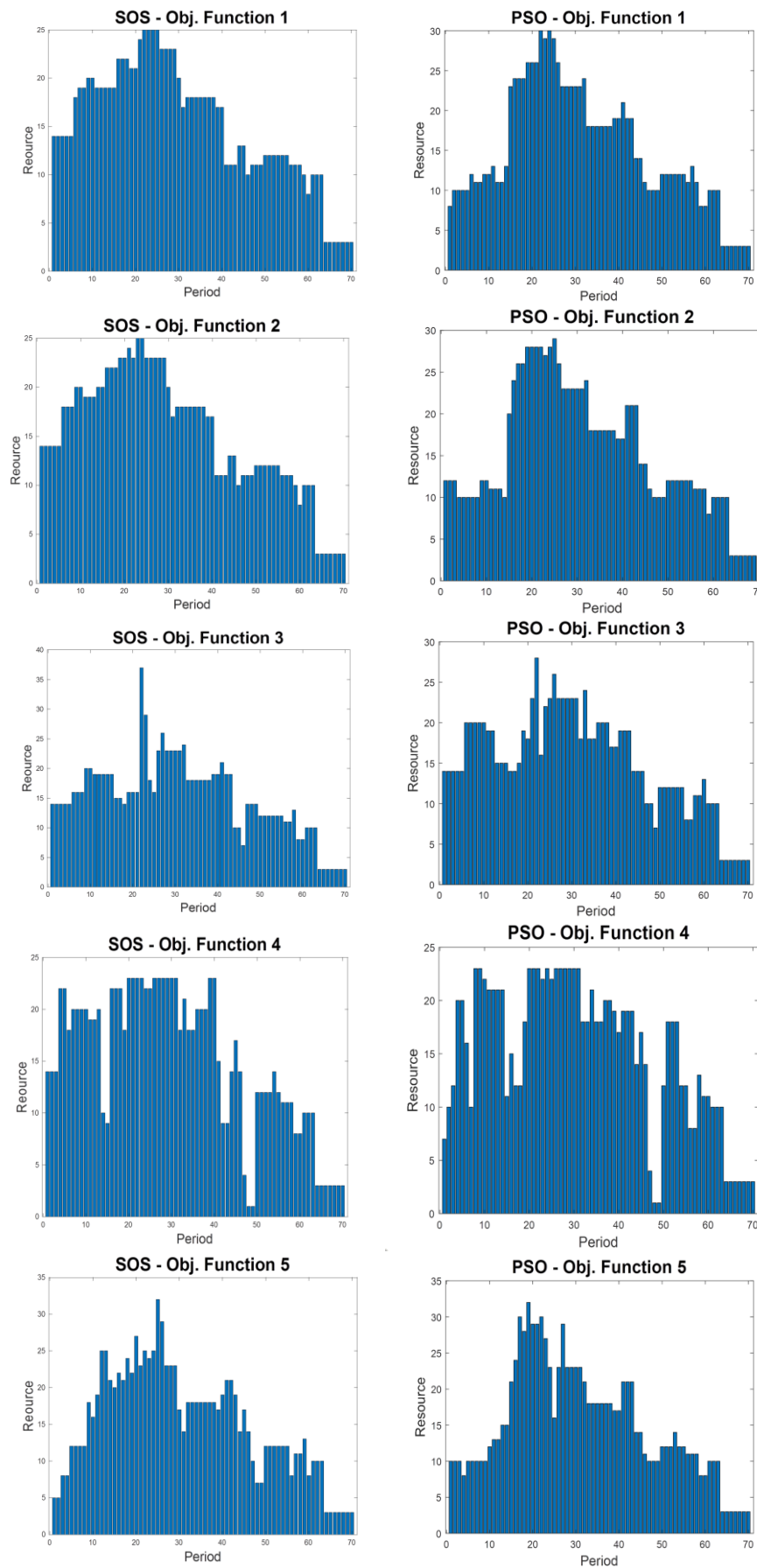
Since SOS is proven to be better than PSO in providing an optimal solution for the resource leveling problem, the comparison of the objective function will use the results of each objective function given by the SOS algorithm. The initial objective values (before leveling objective values) were calculated using the earliest start time of the activities, and will be used to calculate the improvement of the solution provided by the SOS algorithm. Objective values are calculated using the start time of the activities, which is determined after resource leveling has been done by the SOS algorithm. Table 5 presents the after-leveling objective value of each objective function.

In order to determine the objective functions which give the most improvement, each result from the leveling of nine objective functions needs to be calculated using other objective functions. As seen in Table 5, some of the objective values are improved after resource leveling, but others are decreased. The results of objective function 3, when leveled using any objective function are improved (all of the after leveling results are below 468.23), where the highest improvement comes from using objective function 3, which is as expected, as well as using objective functions 4 and 9. The same thing does not happen with the result of objective function 5: the objective value is not improved when calculated using objective functions 3 and 4, while it improves when calculated using other objective functions. Therefore, in order to determine the objective functions that deliver a solution with the most improvement on all objective functions, the cumulative improvement of each objective function was calculated.

As seen in Table 6, each objective function produces a unique improvement over the starting level objective value. Objective function 8 produced the most improvement over the starting level objective value which is 86.20%, followed by objective function 2 with a 73.33% improvement. Objective functions 1, 5, and 9 gained by 67.07%, 46.15%, and 43.71% respectively. Not only did it produce the most improvement, objective function 8 (minimization of the sum of the square of the deviations in resource usage) also had the best average of improvement (58.14%)—slightly above objective function 2 at 44.89%. Objective functions 1, 9, and 7 follow with average percentage of improvements 42.34%, 33.37%, and 23.93% respectively. Objective function 6 has the lowest average improvement with only 6.27%.



**Figure 5.** Convergence Curves Diagrams of SOS and PSO



**Figure 6.** After Leveling Resource Demand Diagrams using SOS and PSO for Objective Function 1 to 5

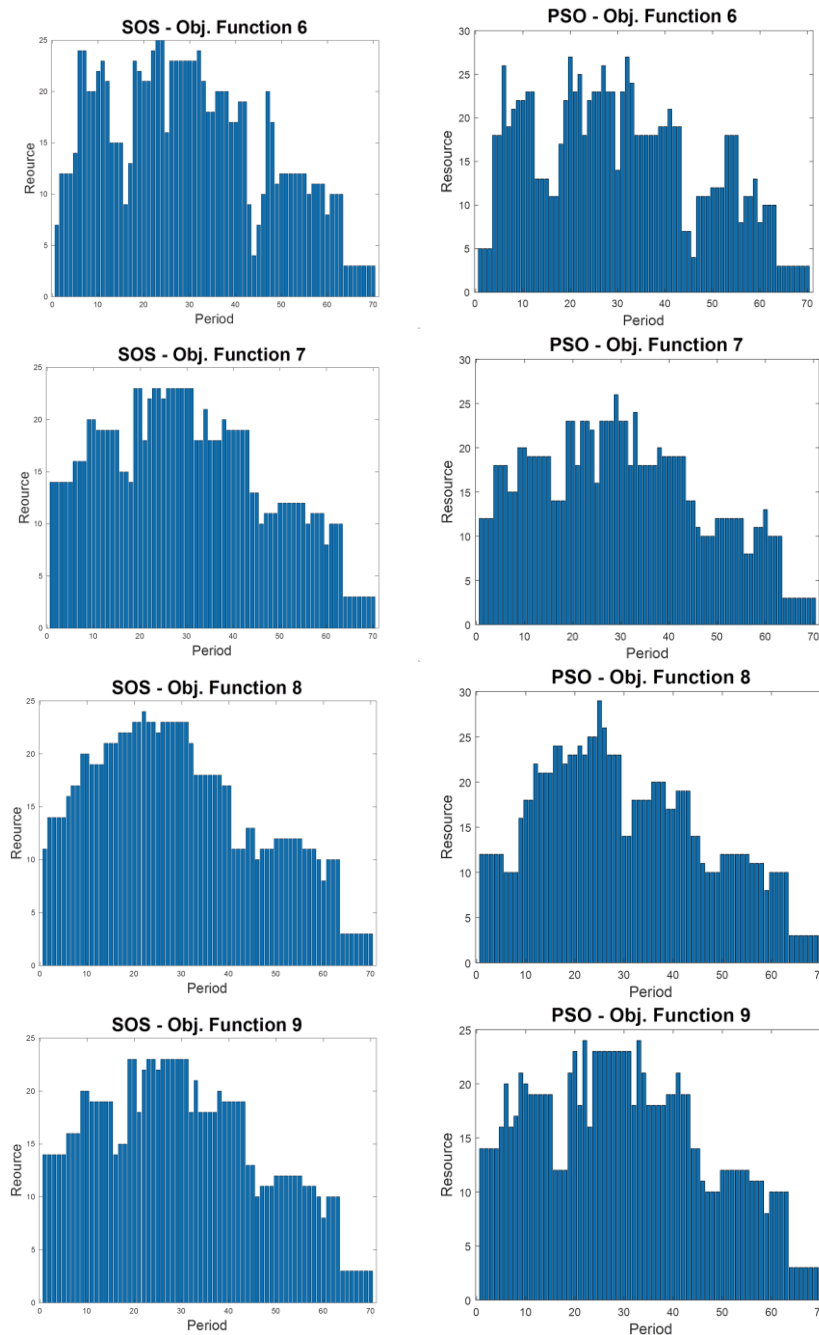


Figure 6. After Leveling Resource Demand Diagrams using SOS and PSO for Objective Function 6 to 9

### Conclusion

The objective of this research is to investigate the performance of the SOS in solving the resource leveling problem from a CPM network resource diagram.. The nine objective functions were optimized using SOS and other optimization method, PSO, as a comparison. Those nine objective functions generate differing resource demand diagrams since each of them minimizes differing parameters. The SOS algorithm surpasses PSO algorithm in solving the resource leveling problem. SOS provide a better objective value in eight of nine objective functions

that are used, and only one objective function produced the same objective value in both SOS and PSO.

In order to find the best objective functions, the average improvement of each was calculated. Objective function 8 (minimization of the sum of the square of the deviations in resource usage) had the best cumulative improvement, followed by objective function 2 and objective function 1 with an average improvement of 41.97%, 41.84%, and 41.74% respectively. However, it is possible that each case may generate its own unique results. Combining those



nine objective functions and adding weighting coefficients by considering the cumulative improvement in order to provide a better resource leveling objective function would be a potential improvement for this study. Furthermore, addressing the uncertainty of resource demand and availability can become a substantial future research agenda for resource leveling development.

### Acknowledgment

The authors gratefully acknowledge that the present research is financially supported by The Ministry of Research, Technology and Higher Education of the Republic of Indonesia under the PDUPT 2019 Research Grant Scheme.

### References

1. Damci, A. and Polat, G., Impacts of Different Objective Functions on Resource Leveling in Construction Projects: A Case Study, *Journal of Civil Engineering and Management*, 20(4), 2014, pp. 537-547.
2. Leu, S.-S., Yang, C.-H., and Huang, J.-C., Resource Leveling in Construction by Genetic Algorithm-Based Optimization and Its Decision Support System Application, *Automation in Construction*, 10(1), 2000, pp. 27-41.
3. Son, J. and Skibniewski, M.J., Multiheuristic Approach for Resource Leveling Problem in Construction Engineering: Hybrid Approach, *Journal of Construction Engineering and Management*, 125(1), 1999, pp. 23-31.
4. Burgess, A.R. and Killebrew, J.B., Variation in Activity Level on a Cyclical Arrow Diagram, *Journal of Industrial Engineering*, 13(2), 1962, pp. 76-83.
5. Wagner, H.M., Giglio, R.J., and Glaser, R.G., Preventive Maintenance Scheduling by Mathematical Programming, *Management Science*, 10(2), 1964, pp. 316-334.
6. Hossein Hashemi Doulabi, S., Seifi, A., and Shariat Seyed, Y., Efficient Hybrid Genetic Algorithm for Resource Leveling Via Activity Splitting, *Journal of Construction Engineering and Management*, 137(2), 2011, pp. 137-146.
7. Easa Said, M., Resource Leveling in Construction by Optimization, *Journal of Construction Engineering and Management*, 115(2), 1989, pp. 302-316.
8. Hiyassat Mohammed, A.S., Applying Modified Minimum Moment Method to Multiple Resource Leveling, *Journal of Construction Engineering and Management*, 127(3), 2001, pp. 192-198.
9. Levy, F.K., Thompson, G.L., and Wiest, J.D., Multiship, Multishop, Workload-Smoothing Program, *Naval Research Logistics Quarterly*, 9(1), 1962, pp. 37-44.
10. Prayogo, D., Cheng, M.-Y., Wong, F.T., Tjandra, D., and Tran, D.-H., Optimization Model for Construction Project Resource Leveling Using a Novel Modified Symbiotic Organisms Search, *Asian Journal of Civil Engineering*, 19(5), 2018, pp. 625-638.
11. Geng, J.-Q., Weng, L.-P., and Liu, S.-H., An Improved Ant Colony Optimization Algorithm for Nonlinear Resource-Leveling Problems, *Computers & Mathematics with Applications*, 61(8), 2011, pp. 2300-2305.
12. Zhang, H. and Yang, Z., Accelerated Particle Swarm Optimization to Solve Large-Scale Network Plan Optimization of Resource-Leveling with a Fixed Duration, *Mathematical Problems in Engineering*, 2018, 2018, pp. 11.
13. Cheng, M.-Y. and Prayogo, D., Symbiotic Organisms Search: A New Metaheuristic Optimization Algorithm, *Computers & Structures*, 139, 2014, pp. 98-112.
14. Prayogo, D., Cheng, M.-Y., and Prayogo, H., A Novel Implementation of Nature-Inspired Optimization for Civil Engineering: A Comparative Study of Symbiotic Organisms Search, *Civil Engineering Dimension*, 19(1), 2017, pp. 36-43.
15. Prayogo, D. and Susanto, Y.T.T., Optimizing the Prediction Accuracy of Friction Capacity of Driven Piles in Cohesive Soil Using a Novel Self-Tuning Least Squares Support Vector Machine, *Advances in Civil Engineering*, 2018(6490169), 2018.
16. Prayogo, D., Sutanto, J.C., Suryo, H.E., and Eric, S., A Comparative Study on Bio-Inspired Algorithms in Layout Optimization of Construction Site Facilities, *Civil Engineering Dimension*, 20(2), 2018, pp. 102-110.
17. Ezugwu, A.E. and Prayogo, D., Symbiotic Organisms Search Algorithm: Theory, Recent Advances and Applications, *Expert Systems with Applications*, 119, 2019, pp. 184-209.
18. Tejani, G.G., Pholdee, N., Bureerat, S., Prayogo, D., and Gandomi, A.H., Structural Optimization Using Multi-Objective Modified Adaptive Symbiotic Organisms Search, *Expert Systems with Applications*, 125, 2019, pp. 425-441.
19. Hegazy, T., Optimization of Resource Allocation and Leveling Using Genetic Algorithms, *Journal of Construction Engineering and Management*, 125(3), 1999, pp. 167-175.
20. Kennedy, J. and Eberhart, R., Particle Swarm Optimization, *Proceedings of IEEE International Conference on Neural Networks*, 1995, pp. 1942-1948.
21. Popescu, C., *How to Use C.P.M in Practice, Part 2. Resources*, Department of Civil Engineering, University of Texas at Austin, 1976.
22. Sears, K.S., Sears, G.A., and Clough, R.H., *Construction Project Management, a Practical Guide to Field Construction Management*, Publisher John Wiley & Sons Inc., 2008.