Impact of Using Multiple Objective Functions on Resource Leveling Using Symbiotic Organisms Search Algorithm

Artya Irvania[1*], Alwafi Pujiraharjo[2], Agus Suharyanto[2]

[1]*Magister Program of Civil Engineering, Brawijaya University, Malang, Indonesia*
[2]*Department of Civil Engineering, Brawijaya University, Malang, Indonesia*

[*1]Corresponding author: artyairvania11@gmail.com

Abstract. The allocation of labor is often a problem that can hinder projects. Optimization is needed in completing resource leveling so that fluctuations in labor requirements can be avoided. This study uses five objective functions to solve resource leveling with the same objective, namely reducing fluctuations in the histogram of labor. This study aims to evaluate the performance and impact of the five objective functions to find out which objective function can produce an efficient histogram of labor and to determine the effect of resource leveling on changes in fluctuations in labor costs. The research used the symbiotic organisms search (SOS) algorithm. The results show that the performance and impact of objective function 4 is more effective in improving the quality of the optimal solution individually and as a whole, resulting in a smoother labor allocation histogram compared to other objective functions so as to improve the efficiency and effectiveness of project implementation.

Keywords: Fluctuation, Resource Leveling, SOS Algorithm, Symbiotic Organism Search Algorithm.

1. Introduction

Construction projects consist of a series of interrelated actions, have unique characteristics, and require management and sufficient resources to achieve the set goals [1]. In this regard, concepts such as project management, project control, and project scheduling with resource constraints have been put forward [2]. The project scheduling problem is defined as scheduling activities and allocating different resources to optimize the problem criteria [3]. In addition, project scheduling can also be used to ensure that the time duration and allocation of labor resources used are in accordance with the goals and objectives of the project [4]. The allocation of labor resources often experiences fluctuations that can hinder the project, so the allocation must be done properly to stabilize the graph of labor resource requirements [5]. Therefore, an effective method is needed to handle and anticipate these problems, namely by using resource leveling. Resource leveling is essential to ensure efficient use of resources for real-world projects. Resource leveling can prevent several negative impacts, such as quickly distributing temporary resources, constantly hiring and firing staff, and idle resources caused by high security levels [6].

Generally, Contractors can complete resource leveling using commercial management software such as Microsoft Project. However, based on Selvam & Tadepalli's (2019) research, resource leveling completed using Microsoft Project shows resource allocation with fluctuations that are still clearly visible, indicating program inefficiency [7]. Furthermore, researchers can solve resource leveling with a mathematical approach, but this approach becomes impractical for large-scale projects. Further researchers tried the heuristic approach, but it did not provide optimal results, so some researchers studied the meta-heuristic approach to find more reliable optimization alternatives, including in handling resource leveling problems [8].

Some meta-heuristic methods that are often used by researchers to perform resource leveling are Ant Colony Optimization (ACO) by [9], Genetic Algorithm (GA) by [10], Differential Evolution (DE) by [11], Particle Swarm Optimization (PSO) by [12], and Symbiotic Organisms Search (SOS) [8]. However, according to Cheng et al. (2016), meta-heuristic algorithms such as ACO, GA, DE, and PSO may be less effective due to the inherent limitations of these algorithms, namely that they are too dependent on tuning parameters, which if the tuning parameter setting is not correct, it will increase the computation time until finding the optimal solution to the problem [13]. Therefore, this study will use the Symbiotic Organisms Search (SOS) algorithm, which requires further implementation to address the resource leveling problem and assess its effectiveness in various project scenarios. The SOS algorithm offers several advantages, including the absence of specific parameters usually required by other metaheuristic algorithms and commendable exploitation capabilities through mutualism and commensalism [14].

Damci and Polat (2014) stated that resource leveling is a minimization problem, so it is necessary to determine the right objective function so that labor resources can be distributed as uniformly as possible. Several studies studied the impact of using various objective functions on resource leveling results using several meta-heuristic methods, namely (1) Damci and Polat (2014) investigated nine objective functions using GA, with results showing that the three best objective functions were generated by objective functions 8, 1, and 5 based on the case studied, (2) Cheng et al. (2016) solved resource leveling with nine different objective functions using DE, with the results of three objective functions that were able to provide the best average increase, namely objective functions 8, 2, and 1, and (3) Prayogo and Kusuma (2019) investigated the performance of the SOS algorithm with nine different objective criteria, with the results of the three best objective functions that were able to provide the largest average increase generated by objective functions 8, 2, and 1.

Based on the empirical studies above, this research wants to continue previous research by using the best objective function on resource leveling produced by [10], [11], and [8], namely objective function 1 (minimum number of absolute deviations of daily resource usage), objective function 2 (minimum number of days with only the addition of resource usage from the previous day), objective function 5 (minimum number of maximum deviations of daily resource usage), and objective function 8 (minimum number of squared deviations of daily resource usage). Furthermore, it is compared with the objective function modeled by El-Rayes and Jun (2009) with the criterion of optimizing the minimum number of idle and unproductive resource days during the project duration [15].

This study aims to determine the impact of using five different objective functions on resource leveling using the SOS algorithm on construction projects with different project types, namely the X building construction project in Surabaya. This research investigates which objective function can give the largest average improvement so as to produce a labor histogram with the least fluctuation and overcome the resource leveling challenge more efficiently.

2. Methods

The research begins with analyzing project data to generate information related to the type of work, duration, and dependency relationships between activities. Then processed using the precedence diagram method (PDM) with the help of Microsoft Project 2013 to create a network diagram. Network diagram is analyzed to generate information related to early start (ES) and late start (LS) for resource leveling optimization purposes.

2.1 Resource Leveling

Resource leveling is a method that aims to make the use of resources as uniform as possible or make the distribution of labor resources according to the type of resources and project needs [7]. Resource leveling is an important aspect of project management, where project managers need a schedule to maximize the efficient utilization of resources required in project completion [16]. This can be achieved by choosing the right objective function according to the complexity of the resource leveling problem. According to Prayogo & Kusuma (2019), the concept of resource leveling focuses on reducing the difference between peak resource demand and daily resource demand as the main objective (Z) of resource leveling. In general, resource leveling can be formulated as follows [8]:

$$\min \quad Z = | Ri+1 - Ri | \tag{1}$$

$$\text{with the constraint} \quad ESx \leq STx \leq LSx \tag{2}$$

$$STx \geq 0 \tag{3}$$

$$x = 1, 2, \ldots n$$

where,

ESx = the early start time of activity x
STx = the start time of activity x
LSx = the late start time of activity x
i = unit of time in the project implementation schedule
x = activity reviewed
n = the total number of activities
Ri = the resource demand on day i

However, this study uses five objective functions on resource leveling obtained from previous research, shown in Table 1.

Table 1. Objective function on resource leveling

No	*Criteria*	*Equation*	*Notation Description*	*Source*
1	The minimum amount of absolute deviation in daily resource usage	$Z = \min \sum_{i=1}^{T} \lvert Rdev_i \rvert$	*min* = minimize *i* = day under consideration *T* = project duration $Rdev_i$ = deviation between required resources on a day i and i+1	Damci and Polat (2014), Cheng et al. (2016), Prayogo and Kusuma (2019)
2	The minimum number of days only with incremental resource usage from the previous day	$Z = \min \sum_{i=1}^{T} \lvert Rinc_i \rvert$	*min* = minimize *i* = day under consideration *T* = project duration $Rinc_i$ = increase between required resources on a day i and i+1	
3	The minimum amount of maximum deviation in daily resource usage	$Z = \min\left[\max \lvert Rdev_i \rvert\right]$	*min* = minimize *max* = maximum *i* = day under consideration $Rdev_i$ = deviation between required resources on a day i and i+1	
4	The minimum sum of the squared deviation in daily resource usage	$Z = \min \sum_{i=1}^{T} \left(Rdev_i\right)^2$	*min* = minimize *i* = days considered *T* = project duration *Rdevi* = deviation between resources required on a day i and i+1	
5	The minimum number of idle and unproductive resource days during the project duration	$Z = \min \Sigma_{i=1}^{T} [\min(A, B) - R_i]$	*min* = minimize *i* = days considered *T* = project duration *Ri* = resources needed on a day i *A* = max (R1, R2, …, Ri) *B* = max (Ri, Ri+1, …, RT)	Rayes and Jun (2009)

Source: [10], [11], and [8].

2.2 Symbiotic Organisms Search

SOS algorithm is a meta-heuristic algorithm first introduced by Cheng & Prayogo (2014). The SOS algorithm is inspired by the interactive behavior between organisms in nature. Organisms rarely live in isolation due to their dependence on other species for survival. This interaction relationship is known as symbiosis [17].

The optimization process is performed using the SOS algorithm with the help of MATLAB 2022 trial version software. The Ecosize and MaxIt parameters must be initialized first to start the resource leveling optimization process. Ecosize is used to determine the number of organisms in an ecosystem calculated using the formula (8*nVar) [8]. The optimization process was repeated 30 times with 1000 iterations using MATLAB 2022 trial version software to obtain accurate and consistent results for each objective function. Table 2 presents the SOS algorithm parameters used in this study.

Table 2. SOS parameters

Control	***Parameters***
Ecosize	688
MaxIt	1000

In the process of finding the optimal solution, the algorithm moves the population of solutions from a possible search space to a more optimal search space. Each solution in the population is called an organism and has a fitness value that represents its survival advantage in the new environment. In the process of iteration, the fitness value of each organism is improved by simulating symbiotic interactions. The iteration process in the SOS algorithm, which follows three search phases, is repeated until an optimal solution is found that meets predetermined criteria. The three search phases are mutualism phase, commensalism phase, and parasitism phase.

Furthermore, the SOS algorithm performs optimization to find the best fitness value of the candidate solution through the main looping process and organism modification with symbiotic interactions consisting of three search phases, namely [18]:

2.2.1 Mutualism Phase

In this phase, Xi+1 organisms are randomly selected from the ecosystem to interact with Xi organisms which aim to build a mutually beneficial relationship. Xi is the vector of the ith organism from the ecosystem, while Xi+1 is the vector of the i+1 organism from the ecosystem (where i ≠ i+1). However, the relationship between the two organisms is only to increase the mutual survival rate of the two organisms Xi and Xi+1 in the population [14]. The new candidate solutions Xinew and Xi+1new will be generated through equations (4) and (5).

Xinew=Xi+rand(0,1)*(Xbest–(Xi+Xi+1)/2*(1+round(rand(0,1))) (4)

Xi+1new=Xi+1+rand(0,1*(Xbest–(Xi+Xi+1)/2*(1+round(rand(0,1)) (5)

2.2.2 Commensalism Phase

In this phase, Xi+1 organisms are randomly selected from the ecosystem to interact with Xi organisms. The two organisms establish a relationship in which Xi seeks to increase its benefits, while the Xi+1 organisms neither benefit nor lose from the relationship. This means that the Xi organism is in an advantageous position over the Xi+1 organism, while the Xi+1 organism is not harmed in the process. Xi's new solution candidate is obtained from calculations based on commensalism symbiotic interactions formulated in equation (6) [14].

Xinew=Xi+rand(-1,1)*(Xbest –Xi+1) (6)

2.2.3 Parasitism Phase

Xi's organism has a role like the anopheles mosquito by creating an artificial parasite (Xparasite). Xparasite is created in the solution search space by duplicating the organism Xi, then modifying the randomly selected dimensions using random numbers. The Xi+1 organism is randomly selected from the ecosystem and used as the Xparasite host, then Xparasite tries to replace Xi+1 in the ecosystem. Each organism is evaluated to measure its fitness value based on the interaction operator. In the selection operator, if the fitness value of Xparasite is better than organism Xi+1, then Xparasite will kill organism Xi+1, so Xi+1 is replaced by Xparasite. If, on the other hand, the fitness value of the Xi+1 organism is better than Xparasite, then Xi+1 will develop immunity from Xparasite, so that Xparasite can no longer live in the ecosystem [14].

2.3 Case Study

Table 3. Project Information Data

ID	***Duration***	***Resource***	***ES***	***LS***		***ID***	***Duration***	***Resource***	***ES***	***LS***
(1)	***(2)***	***(3)***	***(4)***	***(5)***		***(1)***	***(2)***	***(3)***	***(4)***	***(5)***
1	0	0	0	0		44	8	21	83	83
2	58	21	1	1		45	8	27	83	83
3	170	8	1	4		46	8	26	90	90
4	16	18	27	27		47	8	22	88	114
5	65	29	3	3		48	8	28	88	114
6	25	31	27	27		49	8	16	95	121
7	14	17	44	129		50	8	21	91	114
8	14	14	56	129		51	8	25	91	114
9	10	18	45	45		52	8	24	98	121
10	10	21	52	52		53	8	21	93	114
11	10	23	52	52		54	8	27	93	114
12	10	19	59	59		55	8	15	100	121
13	10	24	48	55		56	21	15	90	115

ID (1)	Duration (2)	Resource (3)	ES (4)	LS (5)		ID (1)	Duration (2)	Resource (3)	ES (4)	LS (5)
14	10	23	55	111		57	21	18	90	115
15	10	26	55	111		58	21	11	97	122
16	10	39	62	118		59	8	24	97	97
17	10	19	56	63		60	8	22	97	97
18	10	23	63	86		61	8	13	104	104
19	10	21	63	86		62	8	20	102	128
20	10	19	70	93		63	8	23	102	128
21	10	28	58	65		64	8	16	109	135
22	10	25	63	70		65	8	22	105	128
23	10	21	63	70		66	8	16	105	128
24	10	19	72	79		67	8	15	112	135
25	21	16	59	108		68	8	23	107	128
26	21	21	59	108		69	8	19	107	128
27	21	18	66	84		70	8	18	114	135
28	21	13	73	153		71	21	14	104	115
29	8	23	66	66		72	21	18	104	115
30	8	27	66	66		73	21	13	111	122
31	8	17	76	76		74	8	16	122	122
32	8	23	69	125		75	8	11	122	135
33	8	28	69	125		76	8	15	130	130
34	8	17	79	135		77	8	18	130	135
35	8	23	77	100		78	8	14	132	132
36	8	28	77	100		79	8	14	132	135
37	8	16	84	107		80	8	13	135	135
38	8	21	79	86		81	8	12	135	135
39	8	27	79	86		82	14	9	94	101
40	8	17	86	107		83	14	8	108	115
41	18	15	76	118		84	14	11	122	129
42	18	21	76	118		85	21	7	122	122
43	18	14	83	125		86	10	8	133	133

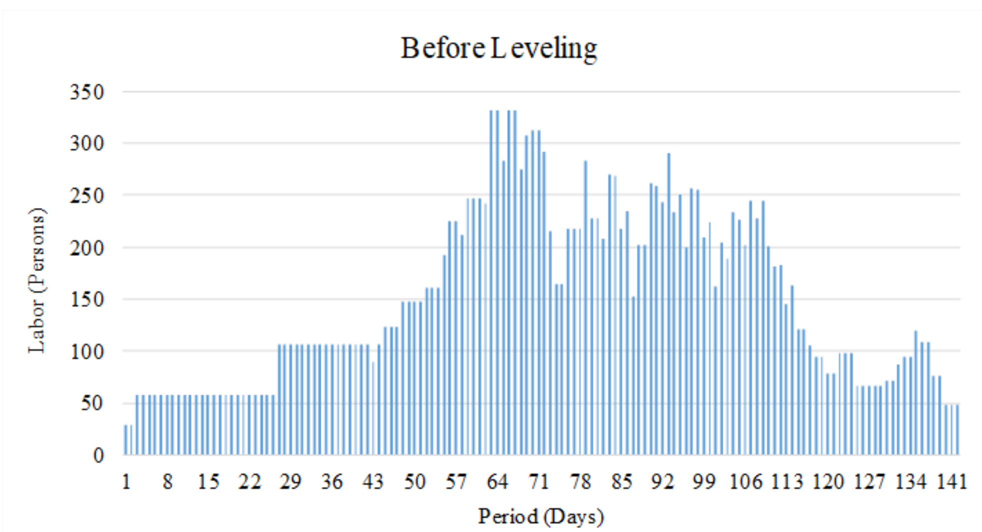

Figure 1 Labor histogram before leveling.

This research uses quantitative descriptive research methods. The object studied was a medium-scale project, namely the X building project in Surabaya consisting of five floors. Project plan data in the form of time schedule, unit price analysis, and labor schedule were collected. The main tool needed for this research is MATLAB 2022 trial version.

The data used consisted of 86 activities and lasted for 141 days. Figure 1 presents a histogram showing the daily labor requirement of the project before resource leveling. The project information data is shown in Table 3.

3. Results And Discussion

The fitness values of 30 simulations using the SOS algorithm are presented in the form of the best result, worst result, average, and standard deviation shown in Table 4.

Based on Table 4, each objective function displays different standard deviation values. The standard deviation can give an idea of the extent to which the solution performance varies among different objective functions. Objective Function 1 has a fairly high standard deviation of 4.4%, which indicates greater variation in the results. This indicates that the performance of this function produces highly variable fitness values so that the results are not stable or tend to change, as is the case with objective functions 2 and 4. Meanwhile, objective function 3 produces a standard deviation of 0%, which means that all fitness values generated from 30 simulations are the same. These objective functions show stable and consistent results without any variation, so it can be said that the two objective functions are quite easy to solve by the SOS algorithm. Based on these results, it can be concluded that the higher the percentage of standard deviation, the greater the variability of data around its mean value. An objective function that produces a low variation in fitness value indicates that the solutions produced indicate several things, namely good stability and consistency in the algorithm used (Table 4).

Table 4. Project Information Fitness value of objective functions

Objective Function	**Fitness Value After Leveling**				
	Best	**Worst**	**Mean**	**Std**	**Std (%)**
1	752	892	827	37	4.4
2	420	459	438	11	2.5
3	49	49	49	0	0
4	15.650	17.832	16.863	600	3.6
5	21.497	21.616	21.539	25	0.1

However, an objective function that has a standard deviation of 0 or low does not guarantee that it is the most effective, as the performance of the algorithm in solving the problem can be affected by the complexity of each objective function itself. Therefore, further evaluation is needed to see the impact of using the tested objective function in other objective functions to find out which objective function is able to provide the best solution performance improvement.

3.1 Resource Histogram After Resource Leveling

The comparison between the 5 objective functions will use the results of the optimal solution provided by the SOS algorithm. The resulting optimal solution is the new start time of each work activity that has been modified by the SOS algorithm. The start time generated by each objective function is put into a barchart which is analyzed to produce a labor histogram of each objective function.

In Figure 1, the resource histogram before leveling shows sharp fluctuations in labor usage. The SOS algorithm performs optimization by shifting start times that are not critical paths to produce the most optimum start time. In Figure 2, the resulting resource histogram after leveling looks smoother than before leveling. However, the five objective functions produce different histograms because each objective function has a different focus and criteria to achieve the optimal solution according to the needs and preferences given. The results show that objective function 4 after leveling can produce a smoother histogram than other objective functions

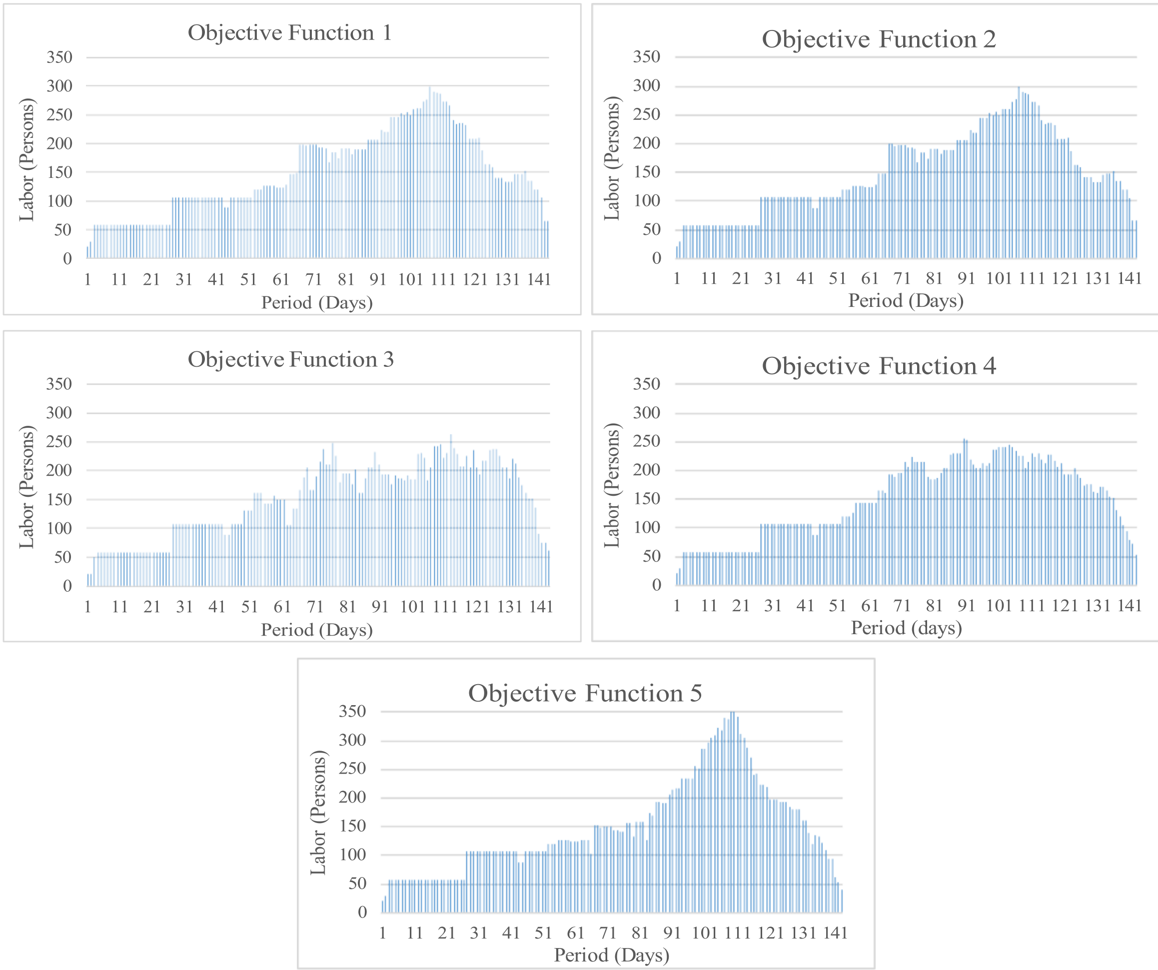

Figure 2 Resource histogram after leveling.

3.2 Impact of Using the Multiple Objective Function on Resource Leveling

Table 5. Fitness Value of multiple objective functions

Objective Function	Objective Function				
	1	2	3	4	5
1	**752**	409	66	20.548	21.896
2	840	**420**	58	19.448	21.932
3	1.618	813	**49**	45.616	23.857
4	898	453	54	**15.650**	22.256
5	1.018	513	50	25.698	**21.497**
Before Leveling	2.262	1135	90	102.768	24.109

Table 6. Percentage increase in fitness value after resource leveling

Objective Function	Objective Function (%)					Average (%)
	1	2	3	4	5	
1	**67**	64	27	80	9	49
2	63	**63**	36	81	9	50
3	28	28	**46**	56	1	32
4	60	60	40	**85**	8	**51**
5	55	55	44	75	**11**	48

The objective function value before leveling has been calculated using the early start data of each work activity. The objective function value before leveling will be used to calculate how much improvement has been generated from the SOS algorithm. To determine which objective function produces the greatest improvement, each leveling result of the 5 objective functions needs to be calculated using the other objective functions. The objective function value is calculated using the start time of the job activity resulting from resource leveling by the SOS algorithm. The objective function values after leveling of the various objective functions are presented in Table 5.

Table 5 shows that all fitness values of each objective function have increased after leveling using any objective function. This result shows that each objective function can have a positive impact on the quality of solutions for other objective functions. However, the increase in fitness value for objective function 5 is insignificant.

Furthermore, by dividing the difference in fitness value between after and before smoothing, it is important to calculate the average improvement in fitness value to determine which objective function can produce solutions with the greatest improvement across all objective functions.

Table 6 shows that each objective function produces a different improvement in solution performance. Objective function 4 produces the best fitness value improvement after resource leveling using the result of the objective function itself, which is 85%. Objective function 4 can also produce the largest average fitness value improvement of 51%, followed by objective functions 2 and 1 of 50% and 49% respectively. The results obtained in this study are consistent with the results of several previous studies, namely research conducted by Damci and Polat (2014), Cheng et al. (2016), and Prayogo and Kusuma [8] with different case studies, where the objective function that provides the best performance improvement is the objective function 4 with the optimization criterion of the minimum amount of the square deviation of daily resource usage.

3.3 Effectiveness of The New Objective Function on Resource Leveling

Table 6 shows that the new objective function applied in this study, namely objective function 5, has not been able to provide the best average increase in fitness value. In terms of overall average improvement, objective function 5 is also able to produce a low average increase in fitness value, which is only 23%. Ideally, a goal function should be able to produce an optimal solution and can improve the overall quality of the solution population. When viewed from this aspect, objective function 4 is still more effective than objective function 5.

However, this does not mean that objective function 4 is the most suitable objective function to use in solving the resource leveling problem, as the results may differ if the research is applied to different project conditions. In real life, a project manager can determine which objective function to use according to the criteria he needs and also the conditions on the project.

4. Conclusions

The best average increase in fitness value is produced by objective function 4 (the minimum sum of the squared deviation in daily resource usage), which is 46%. Objective function 4 can produce a smoother histogram than other objective functions, so it can be the best choice to solve the resource leveling optimization problem

In this research, the objective functions used do not directly consider the cost aspect in resource leveling. Each objective function has its own focus, which is more related to the efficient use of resources and project schedules. For this reason, future researchers can consider modifying or combining the proposed objective functions with cost aspects in an effort to make project planning more holistic and sustainable. The inclusion of objective functions that consider cost aspects will enhance the analysis' ability to ensure efficient and optimal resource allocation from various perspectives, including resource utilization efficiency, project schedule, and cost impact.

5. Acknowledgments

The authors would like to thank Universitas Brawijaya and also express their deepest gratitude and appreciation to Alwafi Pujiraharjo and Agus Suharyanto for the encouragement and advice always given during the author's education.

References

[1]. Hartati G 2019 Dampak Resource Leveling Terhadap Alokasi Tenaga Kerja Dalam Proyek Konstruksi, Jurnal Media Teknologi 4(1) 93–106

[2]. Keynezhad B and Goharshenasan A 2022 A multi-purpose model for optimising project selection and activities scheduling by balancing resource allocation A multi-purpose model for optimising project selection, Int. J. Project Organisation and Management 14(1) 20-35

[3]. Ghasemifard M and Amelian S S 2022 Optimising resource-constrained project probabilistic scheduling problem through a combination of simulation and meta-heuristic algorithm (case study: Govah Sanat Company), Int. J. Project Organisation and Management 14(2) 126

[4]. Saputra N Handayani E and Dwiretnani A 2021 Analisa Penjadwalan Proyek dengan Metode Critical Path Method (CPM) Studi Kasus Pembangunan Gedung Rawat Inap RSUD Abdul Manap Kota Jambi, Jurnal Talenta Sipil 4(1) 44-52

[5]. Loleh M A Inkirawang R L and Mangare J B 2022 Perataan Sumber Daya Tenaga Kerja pada Proyek Konstruksi Bangunan Gedung Menggunakan Microsoft Project (Studi Kasus Pembangunan Lanjutan Aula Dan Gedung RKB SMA Negeri 1 Kawangkoan), Jurnal Sipil Statik 10(1) 47–56

[6]. Li H and Dong X 2018 Multi-mode resource leveling in projects with mode-dependent generalized precedence relations, Expert Systems with Applications 9 193–204

[7]. Selvam G and Tadepalli T C M 2019 Genetic algorithm based optimization for resource leveling problem with precedence constrained scheduling, International Journal of Construction Management 22(4) 659–668

[8]. Prayogo D and Kusuma C T 2019 Optimization of resource leveling problem under multiple objective criteria using a symbiotic organisms search, Civil Engineering Dimension 21(1) 43–51

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

[10]. Damci A and Polat G 2014 Impacts of different objective functions on resource leveling in construction projects: A case study, Journal of Civil Engineering and Management 20(4) 537–547

[11]. Cheng M Y Tran D H and Hoang N D 2016 Fuzzy clustering chaotic-based differential evolution for resource leveling in construction projects, Journal of Civil Engineering and Management 23(1) 113–124

[12]. Zhang H and Yang Z 2018 Accelerated Particle Swarm Optimization to Solve Large-Scale Network Plan Optimization of Resource-Leveling with a Fixed Duration, Mathematical Problems in Engineering 2018

[13]. Cheng M Y Prayogo D and Tran D H 2016 Optimizing Multiple-Resources Leveling in Multiple Projects Using Discrete Symbiotic Organisms Search, Journal of Computing in Civil Engineering 30(3) 1-9

[14]. Ezugwu A E and Prayogo D 2019 Symbiotic Organisms Search Algorithm: theory, recent advances and applications, Expert Systems Applications 119 184–209

[15]. El-Rayes K and Heon Jun D 2009 Optimizing Resource Leveling in Construction Projects, Journal of Construction Engineering and Management pp 1172–1180

[16]. Shadkam E 2022 The problem of resource levelling in multi-project mode by cuckoo optimisation algorithm, International Journal of Mathematics in Operational Research 23(1) 25

[17]. Cheng M Y and Prayogo D 2014 Symbiotic Organisms Search: A new metaheuristic optimization algorithm, Computers and Structures 139 98–112

[18]. Prayogo D Cheng M Y Wong F T Tjandra D, and Tran D H 2018 Optimization model for construction project resource leveling using a novel modified symbiotic organisms search, Asian Journal of Civil Engineering 19(5) 625–638