Indonesian Journal of Electrical Engineering and Computer Science
Vol. 37, No. 1, January 2025, pp. 186~196
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v37.i1.pp186-196

Single search investigation of various searches in recent swarm-based metaheuristics

Purba Daru Kusuma, Ashri Dinimaharawati
Department of Computer Engineering, Faculty of Electrical Engineering, Telkom University, Bandung, Indonesia

Article Info

Article history:

Received May 13, 2024
Revised Aug 31, 2024
Accepted Sep 7, 2024

Keywords:

Metaheuristic
Multiple searches
Optimization
Stochastic system
Swarm intelligence

ABSTRACT

Swarm intelligence has become a popular framework for developing new metaheuristics or stochastic optimization methods in recent years. Many swarm-based metaheuristics are developed by employing multiple searches whether it is conducted through swarm split, serial searches, stochastic choose. Unfortunately, many existing studies that introduced new metaheuristic focused on assessing the performance of the proposed method as a single package. On the other hand, the contribution of each search constructing the metaheuristic is still unknown as the consequence of the missing of single or individual search assessment. Based on this problem, this work is aimed to investigate the performance of five directed searches that are commonly found in recent swarm-based metaheuristics individually. These five searches include: motion toward the highest quality member, motion relative to a randomly chosen member, motion relative to a random solution along the space, motion toward a randomly chosen higher quality member, and motion toward the middle among higher quality members. In this assessment, these five searches are challenged to find the optimal solution of 23 classic functions. The result shows that the first, fourth, and five searches perform better than the second and third searches.

Corresponding Author:

Purba Daru Kusuma
Department of Computer Engineering, Faculty of Electrical Engineering, Telkom University
Telecommunication Street, Buah Batu, Bandung, Indonesia
Email: purbodaru@telkomuniversity.ac.id

1. INTRODUCTION

Swarm intelligence is a popular branch of stochastic optimization that has been employed in many optimizations works. Particle swarm optimization (PSO) as an old swarm intelligence has been extensively utilized in many subjects, such as power distribution network configuration [1], content-based image retrieval system [2], differential amplifier design [3], energy management system for campus building [4], big data clustering [5], predictive control in two-tank system [6], under-frequency load shedding [7], vehicle routing problem [8], path planning for robot system [9], wireless communication network [10]. A new northern goshawk optimization (NGO) has been used also in many subjects, such as power transformation [11], camera imaging [12], wind power prediction [13], dissolved oxygen concentration prediction [14]. Slime mold algorithm has been utilized for optimization in prediction of longitudinal surface settlement [15], strength prediction of high-performance concrete (HPC) [16], 123-bus unbalanced power distribution reconfiguration [17]. Coati optimization algorithm (COA) has been employed in certain subjects, such as health related image synthesizing [18], emotion recognition based on EEG [19], wind power prediction [20].

The popularity of the employment of swarm intelligence is also correlated with the massive development of swarm intelligence. Many recent swarm-based metaheuristics are associated with some researchers. Some swarm-based metaheuristics that are associated to Mirjalili are hippopotamus optimization

(HO) [21], electric eel foraging optimization (EEFO) [22], graylag goose optimization (GGO) [23], crayfish optimization algorithm (COA) [24], marine predator algorithm (MPA) [25]. Some others that are associated to Dehghani are zebra optimization algorithm (ZOA) [26], walrus optimization algorithm (WaOA) [27], lyrebird optimization algorithm (LOA) [28], NGO [29], kookaburra optimization algorithm (KOA) [30]. Some metaheuristics are associated to Kusuma, such as three on three optimization (TOTO) [31], multiple interaction optimizer (MIO) [32], adaptive balance optimizer (ABO) [33], swarm flip-crossover algorithm (SFCA) [34]. Some metaheuristics that are associated to Braik are elk herd optimizer (EHO) [35], white shark optimizer (WSO) [36], chameleon swarm algorithm (CSA) [37]. Some swarm-based metaheuristics that are associated to Malik are pufferfish optimization algorithm (POA) [38], giant armadillo optimization (GAO) [39], one-to-one optimizer (OOBO) [40], golf optimization algorithm (GOA) [41].

In these swarm-based metaheuristics, directed search becomes the backbone of the searching method. Different from full random search or neighbourhood search where the searching process is performed sporadically in any direction, in the directed search, the direction of searching is based on the reference it is employed. The direction can be toward the reference or away from the reference. There are many references employed in these swarm intelligences, such as the highest quality member, random chosen member, random chosen higher quality member, the middle of higher quality member, the mixture between the highest quality member and a randomly chosen member. A new metaheuristic can be developed by selecting this reference, combining some references to create a new reference, or creating a new motion based on the existing reference.

Many recent swarm intelligences or swarm-based metaheuristics were developed by employing multiple searches. These multiple searches are performed in certain ways, such as swarm split, serial searches, randomly chosen search, or iteration-controlled search. Komodo mlipir algorithm (KMA) [42] and COA [43] are the example of metaheuristics that employ swarm split. MPA is the example of metaheuristics where the decision of choosing a search is controlled by the iteration [25]. Osprey optimization algorithm (OOA) [44] and NGO [29] is the example of metaheuristics that employ serial searches where the directed search is enriched with neighborhood search.

One problem in this massive development of metaheuristics is that the evaluation of each search in many studies is hard to find. In many of these studies, the focus of the evaluation is assessing the performance of the proposed metaheuristics as a single package in solving the problem and the superiority of these metaheuristics compared to their comparators. This circumstance makes the contribution of each search that constructs this metaheuristic is difficult to investigate. This problem also becomes the limitation of existing studies that introduced new metaheuristic. For example, NGO consists of two searches where the first search is the directed search relative to a randomly chosen other member and the second search is the neighbourhood search with declining local space as iteration goes [29]. There are several questions that can be obtained based on this strategy. First, how is the performance of NGO if it’s both searches are directed searches? Second, what will happen if the first search is replaced with the motion toward the highest quality member? This circumstance becomes a critical problem in the development of swarm intelligence so that a researcher does not perform mix and match only.

Due to this problem, this work is aimed to investigate several directed searches that are commonly found in some recent swarm-based metaheuristics individually. By performing individual search investigation, the performance of each search can be assessed in a clearer way so that the development of future metaheuristics can be focused on exploiting the searches that have good or acceptable performance. There are five directed searches that are investigated in this work: motion toward the highest quality member, motion relative to a randomly chosen member, motion relative to a random generated solution along the space, motion toward a randomly chosen higher quality member, and motion toward the middle among higher quality members.

Based on the previous explanation, the scientific contributions of this work are listed as follows.

- This work investigates the individual performance of five directed searches that are commonly found in existing swarm intelligences.
- These five directed searches are challenged to solve the theoretical optimization problems which are 23 classic functions.

This paper is divided into four sections. The first section is the introduction that presents the background of this work, problem statement, research objective and the scientific contributions. The second section is the method that is split into two parts. The first part describes the general model of swarm intelligence and the directed searches that are investigated in this paper. The second part describes the assessment method that is used to investigate the performance of these chosen directed searches. The third section exhibits the assessment result and the discussion following the result. Finally, the fourth section is the conclusion that elaborates the concluding remark and the tracks for future works.

2. METHOD
2.1. Swarm intelligence model

In swarm intelligence, the system consists of a certain number of autonomous agents representing the solution. Each agent is active in every iteration and moves independently along the search space to improve its current quality. There is not any central command that gives orders to these agents. Meanwhile, there is interaction among agents so that knowledge sharing occurs among them. This approach is different with single solution-based metaheuristic that there is only one solution or agent in the system. Swarm-based metaheuristic is also different from the population-based metaheuristics that employ evolutionary system so that not all population members perform search in every iteration. It is because there is certain selection in every iteration to decide which members will perform search in every iteration. Roulette wheel is the common method to perform selection process in this population-based metaheuristic that employs evolutionary system. Then, a sorting process is performed to decide the population members that is permitted for the next iteration. On the other hand, in swarm intelligence, there is not any selection or sorting mechanism to decide the population members in the next iteration as the acceptance rule is performed to decide whether the new solution that is produced by the searching process is accepted to replace the current solution.

As common in metaheuristic, the optimization process is split into two stages. The first stage is initialization while the second stage is iteration. It is also common that the initial solution of each agent is generated uniformly along the space. This approach gives equal opportunity to any solution along the space to be chosen as initial solution. After the initial solution for all agents are generated, the optimization enters the second stage which is the iteration. Each agent performs a specific search in every iteration. In general, the final solution is the highest quality member. This optimization process is formalized using algorithm 1. In algorithm 1, the initialization phase is presented from lines 2 to 4. Then, the iteration phase is presented from lines 5 to 9.

Algorithm 1. General algorithm of single directed search
```
1.  begin
2.    for all s ∈ S
3.      initialize s then update s_highest
4.    end for
5.    for t = 1 to t_max
6.      for all s ∈ S
7.        perform search for s then update s_highest
8.      end for
9.    end for
10.   return s_highest as final solution
11. end
```

The formalization of swarm intelligence is by defining the swarm using (1) where s represents the swarm member and S represents the swarm. Then, the initial solution of each member is formalized using (2) representing the uniform random along the search space. Variable α represents the uniform floating point random number between 0 and 1. Then, s_{ub} represents the upper boundary of space and s_{lb} represents the lower boundary of space. Variable i represents the member's index and j represents the dimension index. Then, the updating of the highest quality member is formalized using (3) where $s_{highest}$ represents the highest quality solution and f represents the objective function.

$$S = \{s_1, s_2, s_3, \dots, s_n\} \tag{1}$$

$$s_{i,j} = s_{lb,j} + \alpha\left(s_{ub,j} - s_{lb,j}\right) \tag{2}$$

$$s_{highest}' = \begin{cases} s_i, f(s_i) < f(s_{highest}) \\ s_{highest}, else \end{cases} \tag{3}$$

There are five directed searches whose performance is investigated in this work. The first search is the motion toward the highest quality member. This search can be found for example in COA [43] or ZOA [26]. The second search is the motion relative to a randomly chosen member. This search can be found for example in ZOA [26], TIA [45], or NGO [29]. The third search is the motion relative to a random solution along the space. This search can be found for example in pelican optimization algorithm (POA) [46] or COA [43].

The fourth search is the motion toward a randomly chosen higher quality member, which can be found in KOA [30], OOA [44]. The fifth search is the motion toward the middle among the higher quality members.

The illustration of these five searches is exhibited in Figure 1. Figure 1(a) illustrates the first search which is the motion toward the highest quality member. Figure 1(b) illustrates the second search which is the motion relative to a randomly chosen member. Figure 1(c) illustrates the third search which is the motion relative to a randomly generated solution within the space. Figure 1(d) illustrates the fourth search which is the motion toward a randomly chosen higher quality member. Figure 1(e) illustrates the fifth search which is the motion toward the middle among the higher quality members.

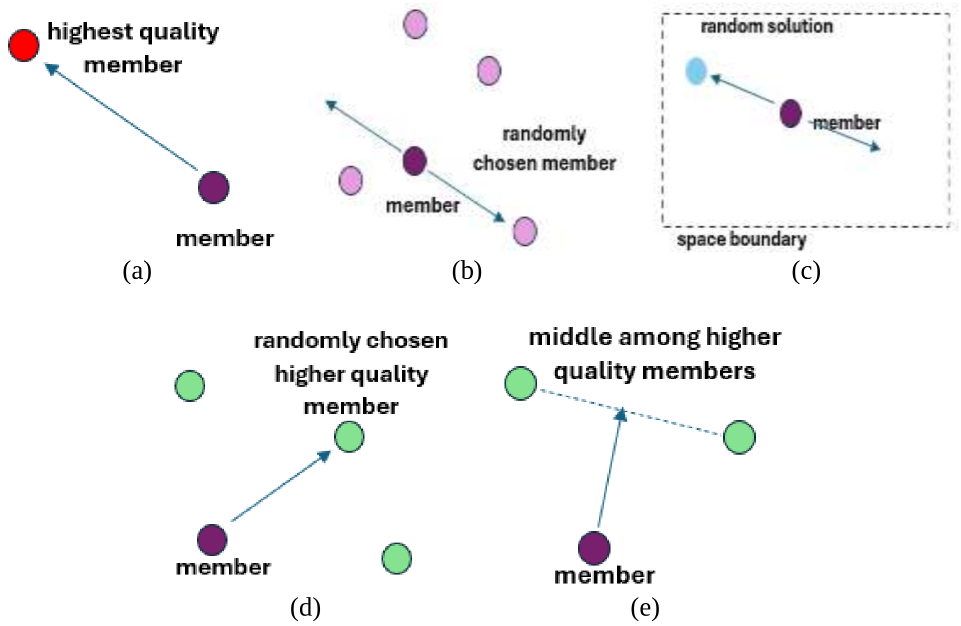

Figure 1. Five directed searches: (a) first search, (b) second search, (c) third search, (d) fourth search, and (e) fifth search

The formalization of these five searches is presented in (4) to (13). In (4) formalizes the first search. In (5) formalizes the uniform random selection of member within the swarm. In (6) formalizes the second search. In (7) formalizes the generating of a random solution along the space. In (8) formalizes the third search. In (9) formalizes the set that consists of all higher quality members relative to the relative member plus the highest quality member. In (10) formalizes the uniform random selection of a member within the pool. In (11) formalizes the fourth search. In (12) formalizes the fifth search. In (13) formalizes the acceptance rule of the candidate to replace the current solution of the related member.

$$c_{i,j} = s_{i,j} + \alpha(s_{highest,j} - \beta s_{i,j}) \tag{4}$$

$$s_{rm} = \gamma(S) \tag{5}$$

$$c_{i,j} = \begin{cases} s_{i,j} + \alpha(s_{rm,j} - \beta s_{i,j}), f(s_{rm}) < f(s_i) \\ s_{i,j} + \alpha(s_{i,j} - \beta s_{rm,j}), else \end{cases} \tag{6}$$

$$s_{rs,j} = s_{lb,j} + \alpha(s_{ub,j} - s_{lb,j}) \tag{7}$$

$$c_{i,j} = \begin{cases} s_{i,j} + \alpha(s_{rs,j} - \beta s_{i,j}), f(s_{rs}) < f(s_i) \\ s_{i,j} + \alpha(s_{i,j} - \beta s_{rs,j}), else \end{cases} \tag{8}$$

$$S_{higher,i} = \{s \in S | f(s) < f(s_i)\} \cup s_{highest} \tag{9}$$

$$s_{rh,i} = \gamma(S_{higher,i}) \tag{10}$$

$$c_{i,j} = s_{i,j} + \alpha(s_{rh,j} - \beta s_{i,j}) \tag{11}$$

$$c_{i,j} = s_{i,j} + \alpha(\frac{\sum_{s_{higher,i}} s_{higher,i,k,j}}{n(S_{higher,i})} - \beta s_{i,j}) \tag{12}$$

$$s_i' = \begin{cases} c_i, f(c_i) < f(s_i) \\ s_i, else \end{cases} \tag{13}$$

The explanation of parameters that are used in (4) to (13). There are two more uniform random numbers where β is an integer random number [1], [2] while γ is a random among a population. s_{rm} represents randomly chosen member among the swarm, s_{rs} represents a random solution along the space and s_{rh} represents a randomly chosen member from the pool that is constructed in (9).

2.2. Assessment scenario

An assessment is performed to investigate the performance of these five searches individually. The assessment is performed by challenging these searches to find the global optimal solution of the 23 classic functions. These functions are well-known due to their variety in the number of optimal solutions (unimodal or multimodal), dimension, search space from narrow to wide, and the terrain of the function. These functions can be split into three groups. The first group consists of seven high dimension unimodal functions (f_1 to f_7). The second group consists of six high dimension multimodal functions (f_8 to f_{13}). The third group consists of ten fixed dimension multimodal functions (f_{14} to f_{23}).

There are three aspects that should be set first which are: objective function, type of search, and the adjusted parameters. Then, the outputs of the assessment are the final solution and the fitness score. The illustration of the assessment scenario is exhibited in Figure 2. The parameter setup for the assessment is as follows. The swarm size is set to 5. Meanwhile, there are two maximum iterations that are observed: 10 and 20. Both values represent low maximum iteration. The dimension for high dimension functions is set to 30.

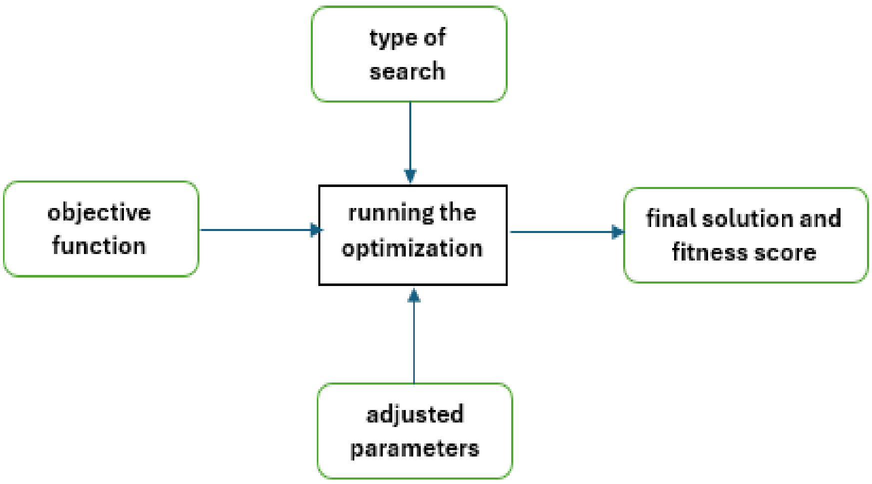

Figure 2. Assessment scenario

3. RESULTS AND DISCUSSION

3.1. Evaluation result

The assessment result is presented in Tables. Table 1 exhibits the result for the assessment when the maximum iteration is 10. Table 2 recapitulates the result in Table 2 based on the mean rank. Meanwhile, Table 3 exhibits the result when the maximum iteration is 20. Then, the result in Table 3 is recapitulated based on the mean rank and it is presented in Table 4.

The result in Table 1, which is recapitulated in Table 2 shows that the fifth search performs the best in solving 23 functions. It achieves the first rank in 13 functions. Then, the first and fourth searches follow as the second and third best searches. The second search becomes the second worst as it is on the fourth rank in 19 functions while the third search becomes the worst as it is on the fifth rank in 22 functions. There are two functions where multiple searches become the best. Three searches are on the first rank in solving f_2 (first, fourth, and fifth searches). Meanwhile, four searches are on the first rank in solving f_{19} (first, second, fourth, and fifth searches). Based on their performance, these searches can be split into two groups in solving high dimension functions. The first group consists of three searches (first, fourth, and fifth searches). The second group consists of two searches (second and third searches). In general, there is significant performance disparity between these two groups in solving high dimension functions in general.

Table 1. Assessment result where maximum iteration is 10

F		Average fitness score				
	Parameter	First search	Second search	Third search	Fourth search	Fifth search
1	mean	4.410×10^{1}	3.649×10^{3}	8.198×10^{4}	1.074×10^{2}	3.224×10^{1}
	mean rank	2	4	5	3	1
2	mean	0.000	2.019×10^{23}	2.696×10^{46}	0.000	0.000
	mean rank	1	4	5	1	1
3	mean	3.496×10^{3}	3.181×10^{4}	2.076×10^{5}	5.246×10^{3}	1.904×10^{3}
	mean rank	2	4	5	3	1
4	mean	6.491	3.893×10^{1}	9.504×10^{1}	7.170	4.981
	mean rank	2	4	5	3	1
5	mean	1.322×10^{3}	1.522×10^{6}	3.681×10^{8}	3.095×10^{3}	5.995×10^{2}
	mean rank	2	4	5	3	1
6	mean	4.724×10^{1}	3.565×10^{3}	8.136×10^{4}	9.791×10^{1}	3.656×10^{1}
	mean rank	2	4	5	3	1
7	mean	0.105	1.633	1.663×10^{2}	0.105	0.115
	mean rank	1	4	5	1	3
8	mean	-2.265×10^{3}	-2.371×10^{3}	-2.160×10^{3}	-1.992×10^{3}	-1.984×10^{3}
	mean rank	2	1	3	4	5
9	mean	7.303×10^{1}	2.609×10^{2}	4.907×10^{2}	8.888×10^{1}	4.909×10^{1}
	mean rank	2	4	5	3	1
10	mean	2.906	1.137×10^{1}	1.999×10^{1}	3.402	2.516
	mean rank	2	4	5	3	1
11	mean	1.483	3.766×10^{1}	7.556×10^{2}	1.893	1.337
	mean rank	2	4	5	3	1
12	mean	2.044	5.374×10^{5}	7.802×10^{8}	2.549	1.601
	mean rank	2	4	5	3	1
13	mean	6.814	3.825×10^{6}	1.473×10^{9}	1.011×10^{1}	5.480
	mean rank	2	4	5	3	1
14	mean	1.681×10^{1}	3.262×10^{1}	1.754×10^{2}	1.809×10^{1}	1.808×10^{1}
	mean rank	1	4	5	3	2
15	mean	0.023	0.045	1.286	0.027	0.035
	mean rank	1	4	5	2	3
16	mean	-0.875	-0.839	6.462	-0.866	-0.871
	mean rank	1	4	5	3	2
17	mean	2.196	1.251	4.904	3.049	4.493
	mean rank	1	4	5	2	3
18	mean	3.608×10^{1}	3.927×10^{1}	2.630×10^{2}	4.930×10^{1}	2.986×10^{1}
	mean rank	2	3	5	4	1
19	mean	-0.049	-0.049	-0.032	-0.049	-0.049
	mean rank	1	1	5	1	1
20	mean	-1.822	-2.126	-1.194	-2.135	-1.784
	mean rank	3	2	5	1	4
21	mean	-1.146	-0.925	-0.320	-1.364	-1.151
	mean rank	3	4	5	1	2
22	mean	-1.092	-1.065	-0.497	-1.336	-1.278
	mean rank	3	4	5	1	2
23	mean	-1.791	-1.105	-0.718	-1.504	-1.624
	mean rank	1	4	5	3	2

Table 2. Mean rank recapitulation when maximum iteration is 10

Search	Mean rank				
	1st rank	2nd rank	3rd rank	4th rank	5th rank
1st search	8	12	3	0	0
2nd search	2	1	1	19	0
3rd search	0	0	1	0	22
4th search	6	2	13	2	0
5th search	13	5	3	1	1

Result in Table 3 which is recapitulated in Table 4 strengthens the dominance of the fifth search. The fifth search is still on the first rank in 12 functions. Then, the second search pushes forward as it achieves the first rank in eight functions, but it is still on the fourth rank in eleven functions. Meanwhile, the first search still achieves the first rank in seven functions. The fourth search achieves the first rank only in three functions but still manages its second and third rank. The third search still becomes the worst as it is on the fifth rank in twenty functions.

Table 3. Assessment result where maximum iteration is 20

F	Parameter	Fitness score				
		First search	Second search	Third search	Fourth search	Fifth search
1	mean	0.019	1.593×10^{2}	8.209×10^{4}	0.070	0.015
	mean rank	2	4	5	3	1
2	mean	0.000	0.000	2.330×10^{43}	0.000	0.000
	mean rank	1	1	5	1	1
3	mean	2.491×10^{2}	1.439×10^{4}	2.266×10^{5}	4.699×10^{2}	1.345×10^{2}
	mean rank	2	4	5	3	1
4	mean	0.237	1.457×10^{1}	$9.438\text{x}10^{1}$	0.296	0.123
	mean rank	2	4	5	3	1
5	mean	2.921×10^{1}	4.463×10^{3}	3.520×10^{8}	3.008×10^{1}	2.906×10^{1}
	mean rank	2	4	5	3	1
6	mean	6.020	1.391×10^{2}	8.140×10^{4}	6.160	6.102
	mean rank	1	4	5	3	2
7	mean	0.024	0.132	1.642×10^{2}	0.027	0.024
	mean rank	1	4	5	3	1
8	mean	-2.177×10^{3}	-2.541×10^{3}	-2.470×10^{3}	-2.268×10^{3}	-1.924×10^{3}
	mean rank	4	1	2	3	5
9	mean	0.366	1.952×10^{2}	4.804×10^{2}	1.005	0.617
	mean rank	2	4	5	3	1
10	mean	0.035	3.877	1.997×10^{1}	0.059	0.024
	mean rank	2	4	5	3	1
11	mean	0.082	2.113	7.537×10^{2}	0.069	0.028
	mean rank	3	4	5	2	1
12	mean	1.019	3.977	9.061×10^{8}	1.038	1.074
	mean rank	1	4	5	2	3
13	mean	3.164	2.227×10^{2}	1.593×10^{9}	3.167	3.217
	mean rank	1	4	5	2	3
14	mean	1.025×10^{1}	1.190×10^{1}	7.818×10^{1}	1.184×10^{1}	1.245×10^{1}
	mean rank	1	3	5	2	4
15	mean	0.024	0.013	0.039	0.015	0.032
	mean rank	3	1	5	2	4
16	mean	-0.912	-0.979	5.283	-0.834	-0.910
	mean rank	2	1	5	4	3
17	mean	6.016	0.665	3.127	3.549	5.172
	mean rank	5	1	2	3	4
18	mean	3.569×10^{1}	$1.590\text{x}10^{1}$	6.146×10^{1}	2.879×10^{1}	6.678×10^{1}
	mean rank	3	1	4	2	5
19	mean	-0.049	-0.049	-0.039	-0.049	-0.049
	mean rank	1	1	5	1	1
20	mean	-2.077	-2.224	-1.718	-1.786	-1.912
	mean rank	2	1	5	4	3
21	mean	-1.829	-1.495	-0.469	-2.486	-2.203
	mean rank	4	3	5	1	2
22	mean	-1.447	-1.731	-0.679	-1.830	-2.028
	mean rank	4	3	5	2	1
23	mean	-1.629	-1.726	-0.721	-1.889	-2.266
	mean rank	4	3	5	2	1

Table 4. Mean rank recapitulation when maximum iteration is 20

Search	Mean rank				
	1st rank	2nd rank	3rd rank	4th rank	5th rank
1st search	7	8	3	4	1
2nd search	8	0	4	11	0
3rd search	0	2	0	1	20
4th search	3	8	10	2	0
5th search	12	2	4	3	2

3.2. Discussion

This sub section presents a comprehensive discussion regarding the assessment result. This result is then utilized to construct the findings of this work. The exploration includes the investigation of the strength of weakness of each search. This sub section also provides different conversation compared to many existing studies in metaheuristics which focused on glorifying the superior performance of the proposed method in tackling various sets of optimization problems without deep diving into the nature of the proposed method.

The main finding in this work is that the existence of highest quality member becomes critical factor in constructing the reference. The first, fourth, and fifth searches perform better than the second and the third searches due to its utilization of the highest quality member. The fifth search becomes the best search while the first search becomes the third best search. Both searches guarantee the utilization of the highest quality member in every movement in different ways. Meanwhile, the fourth search does not guarantee the utilization of the highest quality member due to the stochastic picking among the higher quality members.

The second finding is the searching process cannot rely on only one entity, in this case the highest quality member. It is because moving toward the highest quality member does not guarantee a better improvement. On the other hand, moving toward the area where higher quality members including the highest quality member exist becomes wiser. This strategy is adopted in the fifth search which is based on the assessment result, the fifth search performs better than the first search. This strategy is also adopted in KMA for the male lizards [42] or in grey wolf optimizer (GWO) [47].

The third finding is that random search performs worst as it does not utilize the value of other members. The third search can be seen as an improved random search as the reference is a random solution along the space. Meanwhile, the improvement is the selective direction of the motion where forward motion is performed when the target or reference is better than the member. But the neglection of the existence of other members makes the convergence is difficult to achieve.

The implications of these findings and its relationship with future development of metaheuristic is as follows. First, a new metaheuristic is better developed based on multiple searches rather than single search only. This implication is caused by the circumstance where there is not any search which is superior in all cases. This circumstance is also related to the no-free-lunch theory where there is not any technique that is superior in handling all problems. The weakness of certain search should be covered with other searches. Second, the existence of the highest quality member is important in any future metaheuristic as it plays very important role. This highest quality member can be chosen solely or blended with other entities, such as a randomly picked member, a randomly picked higher quality member.

There are limitations in this work which can be used for the opportunity for future studies. This study investigates only five directed searches which are commonly found in many swarm-based metaheuristics. But there are various spaces are still not observed. The first space is regarding the acceptance approach in facing the worse candidate that is produced by the search. There are three common roles in accepting worse solution: accept, reject, or conditional accept. Many metaheuristics associated with Dehghani reject the worse solution. On the other hand, many metaheuristics that are associated with Mirjalili still accept the worse solution. Meanwhile, some metaheuristics, such as simulated annealing (SA), employ conditional acceptance. The second space is regarding the hybrid reference that is employed in the directed search. This hybrid reference is obtained by combining multiple entities based on certain portion. The common example is the mixture of the global highest quality member and the local highest quality member which can be found in PSO [48] or golden search optimization (GSO) [49]. The second example is the mixture between the highest quality member and a randomly chosen member. The third example is the mixture between multiple randomly chosen members. The third space is regarding the distribution of the random number that is used in the metaheuristics. Many metaheuristics employ uniform distribution due to its simplicity. The fourth space is regarding the swarm split. Some metaheuristics, such as KMA [42] and COA [43] perform swam split in a rigid manner. On the other hand, many other metaheuristics does not differentiate the strategy so that all members perform same search or searches. Meanwhile, some metaheuristics employ other distributions, such as normal distribution, Brownian motion, or levy flight as in MPA [25]. The studies that investigate or compare the performance of these spaces are still hard to find and this circumstance can be used as opportunity for further studies.

4. CONCLUSION

This work has presented the performance investigation of five searches which are commonly found in many swarm-based metaheuristics individually. This work is important in the development of metaheuristic, especially the swarm-based ones as it provides the significance of searching process which are common in many swarm-based metaheuristics. This work also fills the limitation in many studies introducing new metaheuristics as the performance of the proposed technique is performed as whole package. This common approach can make mislead in the investigation as the contribution of each search cannot be traced.

Based on this problem, this work is important to provide clearer view of the contribution of every search in finding the best solution. Through this investigation, it is found that the existence of the highest quality member plays an important role in boosting the performance of the search which becomes the first finding. Then, the second finding is that moving toward the area of higher quality members proof better than moving toward only the highest quality member. The third finding is that performing the random search performs worst as it ignores the existence or contribution of other members within the swarm. Future studies regarding in this work can be performed in several ways. First, future investigation studies on metaheuristics can be conducted by analyzing the acceptance rule, swarm split, and hybrid reference in a more comprehensive manner. Second, future studies can also be conducted by performing single search investigation for more various searching methods. Third, it is important for future studies proposing new metaheuristics are enriched with single search investigation rather than only employing the proposed technique as a single package to solve certain optimization problems, whether the standard cases or the practical ones.

ACKNOWLEDGEMENTS

Publication of this paper is supported by Telkom University, Indonesia.

REFERENCES

[1] Y. Siregar, T. S. J. Tambun, S. P. Panjaitan, K. Tanjung, and S. Yana, "Distribution network reconfiguration utilizing the particle swarm optimization algorithm and exhaustive search methods," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 13, no. 2, pp. 821–831, 2024, doi: 10.11591/eei.v13i2.6206.

[2] A. Bassel, M. Jameel, and M. A. Saad, "Jumping particle swarm optimization algorithm framework for content-based image retrieval system," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 12, no. 6, pp. 3665–3676, 2023, doi: 10.11591/eei.v12i6.5024.

[3] P. P. Prajapati, A. J. Kshatriya, D. N. Patel, S. K. Gonsai, H. B. Tank, and K. R. Sheth, "Comparative analysis of meta heuristics algorithm for differential amplifier design," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 6, pp. 3395–3401, 2023, doi: 10.11591/eei.v12i6.6153.

[4] N. A. Mahmud and N. Y. Dahlan, "Optimum energy management strategy with enhanced time of use tariff for campus building using particle swarm optimization," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 28, no. 2, pp. 644–653, 2022, doi: 10.11591/ijeecs.v28.i2.pp644-653.

[5] S. I. Boushaki, B. H. Mahammed, O. Bendjeghaba, and M. Mosbah, "Big data clustering based on spark chaotic improved particle swarm optimization," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 34, no. 1, pp. 419–429, 2024, doi: 10.11591/ijeecs.v34.i1.pp419-429.

[6] A. K. Kojouri and J. M. Fard, "An efficient application of particle swarm optimization in model predictive control of constrained two-tank system," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 4, pp. 3540–3550, 2022, doi: 10.11591/ijece.v12i4.pp3540-3550.

[7] N. T. Le, T. T. Phung, A. H. Quyen, B. L. P. Nguyen, and A. N. Nguyen, "A hybrid approach of artificial neural network-particle swarm optimization algorithm for optimal load shedding strategy," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 4, pp. 4253–4263, 2022, doi: 10.11591/ijece.v12i4.pp4253-4263.

[8] Q. Qiao, F. Tao, H. Wu, X. Yu, and M. Zhang, "Optimization of a capacitated vehicle routing problem for sustainable municipal solid waste collection management using the PSO-TS algorithm," *International Journal of Environmental Research and Public Health*, vol. 17, no. 6, 2020, doi: 10.3390/ijerph17062163.

[9] A. A. Alabdalbari and I. A. Abed, "New robot path planning optimization using hybrid GWO-PSO algorithm," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 11, no. 3, pp. 1289–1296, 2022, doi: 10.11591/eei.v11i3.3677.

[10] H. Mohammed Hussein, K. Katzis, L. P. Mfupe, and E. T. Bekele, "Performance Optimization of High-Altitude Platform Wireless Communication Network Exploiting TVWS Spectrums Based on Modified PSO," *IEEE Open Journal of Vehicular Technology*, vol. 3, pp. 356–366, 2022, doi: 10.1109/OJVT.2022.3191762.

[11] L. Z. Wang, J. F. Chi, Y. Q. Ding, H. Y. Yao, Q. Guo, and H. Q. Yang, "Transformer fault diagnosis method based on SMOTE and NGO-GBDT," *Scientific Reports*, vol. 14, no. 1, 2024, doi: 10.1038/s41598-024-57509-w.

[12] Q. Liu, M. Dong, P. Sun, B. Yan, J. Wang, and L. Zhu, "A calibration and prediction method of the camera imaging parameters in variable temperature environment," *Optics Communications*, vol. 560, 2024, doi: 10.1016/j.optcom.2024.130428.

[13] G. Hou, J. Wang, Y. Fan, J. Zhang, and C. Huang, "A novel wind power deterministic and interval prediction framework based on the critic weight method, improved northern goshawk optimization, and kernel density estimation," *Renewable Energy*, vol. 226, 2024, doi: 10.1016/j.renene.2024.120360.

[14] J. Jiao, Q. Ma, F. Liu, L. Zhao, and S. Huang, "A hybrid prediction model of dissolved oxygen concentration based on secondary decomposition and bidirectional gate recurrent unit," *Environmental Geochemistry and Health*, vol. 46, no. 4, 2024, doi: 10.1007/s10653-024-01884-w.

[15] W. Shang, Y. Li, H. Wei, Y. Qiu, C. Chen, and X. Gao, "Prediction method of longitudinal surface settlement caused by double shield tunnelling based on deep learning," *Scientific Reports*, vol. 14, no. 1, 2024, doi: 10.1038/s41598-023-49096-z.

[16] L. Yu, "Strength properties prediction of RCA concrete via hybrid regression framework," *Journal of Engineering and Applied Science*, vol. 71, no. 1, 2024, doi: 10.1186/s44147-023-00332-3.

[17] N. Nacar Cikan and M. Cikan, "Reconfiguration of 123-bus unbalanced power distribution network analysis by considering minimization of current & voltage unbalanced indexes and power loss," *International Journal of Electrical Power and Energy Systems*, vol. 157, 2024, doi: 10.1016/j.ijepes.2024.109796.

[18] T. H. H. Le, P. H. Dinh, V. H. Vu, and N. L. Giang, "A new approach to medical image fusion based on the improved Extended difference-of-Gaussians combined with the Coati optimization algorithm," *Biomedical Signal Processing and Control*, vol. 93, 2024, doi: 10.1016/j.bspc.2024.106175.

[19] E. H. Houssein, A. Hammad, M. M. Emam, and A. A. Ali, "An enhanced coati optimization algorithm for global optimization and feature selection in EEG emotion recognition," *Computers in Biology and Medicine*, vol. 173, 2024, doi: 10.1016/j.compbiomed.2024.108329.
[20] C. Wang, H. Lin, H. Hu, M. Yang, and L. Ma, "A hybrid model with combined feature selection based on optimized VMD and improved multi-objective coati optimization algorithm for short-term wind power prediction," *Energy*, vol. 293, 2024, doi: 10.1016/j.energy.2024.130684.
[21] M. H. Amiri, N. Mehrabi Hashjin, M. Montazeri, S. Mirjalili, and N. Khodadadi, "Hippopotamus optimization algorithm: a novel nature-inspired optimization algorithm," *Scientific Reports*, vol. 14, no. 1, 2024, doi: 10.1038/s41598-024-54910-3.
[22] W. Zhao *et al.*, "Electric eel foraging optimization: A new bio-inspired optimizer for engineering applications," *Expert Systems with Applications*, vol. 238, 2024, doi: 10.1016/j.eswa.2023.122200.
[23] E. S. M. El-kenawy, N. Khodadadi, S. Mirjalili, A. A. Abdelhamid, M. M. Eid, and A. Ibrahim, "Greylag goose optimization: nature-inspired optimization algorithm," *Expert Systems with Applications*, vol. 238, 2024, doi: 10.1016/j.eswa.2023.122147.
[24] H. Jia, H. Rao, C. Wen, and S. Mirjalili, "Crayfish optimization algorithm," *Artificial Intelligence Review*, vol. 56, pp. 1919–1979, 2023, doi: 10.1007/s10462-023-10567-4.
[25] A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, "Marine predators algorithm: a nature-inspired metaheuristic," *Expert Systems with Applications*, vol. 152, 2020, doi: 10.1016/j.eswa.2020.113377.
[26] E. Trojovska, M. Dehghani, and P. Trojovsky, "Zebra optimization algorithm: a new bio-inspired optimization algorithm for solving optimization algorithm," *IEEE Access*, vol. 10, pp. 49445–49473, 2022, doi: 10.1109/ACCESS.2022.3172789.
[27] P. Trojovský and M. Dehghani, "A new bio-inspired metaheuristic algorithm for solving optimization problems based on walruses behavior," *Scientific Reports*, vol. 13, no. 1, 2023, doi: 10.1038/s41598-023-35863-5.
[28] M. Dehghani, G. Bektemyssova, Z. Montazeri, G. Shaikemelev, O. P. Malik, and G. Dhiman, "Lyrebird optimization algorithm: a new bio-inspired metaheuristic algorithm for solving optimization problems," *Biomimetics*, vol. 8, no. 6, 2023, doi: 10.3390/biomimetics8060507.
[29] M. Dehghani, S. Hubalovsky, and P. Trojovsky, "Northern goshawk optimization: a new swarm-based algorithm for solving optimization problems," *IEEE Access*, vol. 9, pp. 162059–162080, 2021, doi: 10.1109/ACCESS.2021.3133286.
[30] M. Dehghani, Z. Montazeri, G. Bektemyssova, O. P. Malik, G. Dhiman, and A. E. M. Ahmed, "Kookaburra optimization algorithm: a new bio-inspired metaheuristic algorithm for solving optimization problems," *Biomimetics*, vol. 8, no. 6, 2023, doi: 10.3390/biomimetics8060470.
[31] P. D. Kusuma and A. Dinimaharawati, "Three on Three Optimizer: A New Metaheuristic with Three Guided Searches and Three Random Searches," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 1, pp. 420–429, 2023, doi: 10.14569/IJACSA.2023.0140145.
[32] P. D. Kusuma and A. Novianty, "Multiple interaction optimizer: a novel metaheuristic and its application to solve order allocation problem," *International Journal of Intelligent Engineering and Systems*, vol. 16, no. 2, pp. 440–453, 2023, doi: 10.22266/ijies2023.0430.35.
[33] P. D. Kusuma and A. Dinimaharawati, "Adaptive balance optimizer: a new adaptive metaheuristic and its application in solving optimization problem in finance," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 4, pp. 130–142, 2023, doi: 10.14569/IJACSA.2023.0140415.
[34] P. D. Kusuma and F. C. Hasibuan, "Swarm flip-crossover algorithm: a new swarm-based metaheuristic enriched with a crossover strategy," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 14, no. 2, pp. 2145–2155, 2024, doi: 10.11591/ijece.v14i2.pp2145-2155.
[35] M. A. Al-Betar, M. A. Awadallah, M. S. Braik, S. Makhadmeh, and I. A. Doush, "Elk herd optimizer: a novel nature-inspired metaheuristic algorithm," *Artificial Intelligence Review*, vol. 57, no. 3, 2024, doi: 10.1007/s10462-023-10680-4.
[36] M. Braik, A. Hammouri, J. Atwan, M. A. Al-Betar, and M. A. Awadallah, "White shark optimizer: a novel bio-inspired meta-heuristic algorithm for global optimization problems," *Knowledge-Based Systems*, vol. 243, 2022, doi: 10.1016/j.knosys.2022.108457.
[37] M. S. Braik, "Chameleon swarm algorithm: a bio-inspired optimizer for solving engineering design problems," *Expert Systems with Applications*, vol. 174, 2021, doi: 10.1016/j.eswa.2021.114685.
[38] O. Al-Baik *et al.*, "Pufferfish optimization algorithm: a new bio-inspired metaheuristic algorithm for solving optimization problems," *Biomimetics*, vol. 9, no. 2, 2024, doi: 10.3390/biomimetics9020065.
[39] O. Alsayyed *et al.*, "Giant armadillo optimization: a new bio-inspired metaheuristic algorithm for solving optimization problems," *Biomimetics*, vol. 8, no. 8, 2023, doi: 10.3390/biomimetics8080619.
[40] M. Dehghani, E. Trojovská, P. Trojovský, and O. P. Malik, "OOBO: a new metaheuristic algorithm for solving optimization problems," *Biomimetics*, vol. 8, no. 6, 2023, doi: 10.3390/biomimetics8060468.
[41] Z. Montazeri, T. Niknam, J. Aghaei, O. P. Malik, M. Dehghani, and G. Dhiman, "Golf optimization algorithm: a new game-based metaheuristic algorithm and its application to energy commitment problem considering resilience," *Biomimetics*, vol. 8, no. 5, 2023, doi: 10.3390/biomimetics8050386.
[42] S. Suyanto, A. A. Ariyanto, and A. F. Ariyanto, "Komodo Mlipir Algorithm," *Applied Soft Computing*, vol. 114, 2022, doi: 10.1016/j.asoc.2021.108043.
[43] M. Dehghani, Z. Montazeri, E. Trojovská, and P. Trojovský, "Coati optimization algorithm: a new bio-inspired metaheuristic algorithm for solving optimization problems," *Knowledge-Based Systems*, vol. 259, 2023, doi: 10.1016/j.knosys.2022.110011.
[44] M. Dehghani and P. Trojovský, "Osprey optimization algorithm: a new bio-inspired metaheuristic algorithm for solving engineering optimization problems," *Frontiers in Mechanical Engineering*, vol. 8, 2023, doi: 10.3389/fmech.2022.1126450.
[45] P. D. Kusuma and A. Novianty, "Total interaction algorithm: a metaheuristic in which each agent interacts with all other agents," *International Journal of Intelligent Engineering and Systems*, vol. 16, no. 1, pp. 224–234, 2023, doi: 10.22266/ijies2023.0228.20.
[46] P. Trojovský and M. Dehghani, "Pelican optimization algorithm: a novel nature-inspired algorithm for engineering applications," *Sensors*, vol. 22, no. 3, 2022, doi: 10.3390/s22030855.
[47] A. Q. H. Badar, "Grey Wolf Optimizer," *Evolutionary Optimization Algorithms*, pp. 165–190, 2021, doi: 10.1201/9781003206477-8.
[48] A. G. Gad, "Particle swarm optimization algorithm and its applications: a systematic review," *Archives of Computational Methods in Engineering*, vol. 29, no. 5, pp. 2531–2561, 2022, doi: 10.1007/s11831-021-09694-4.
[49] M. Noroozi, H. Mohammadi, E. Efatinasab, A. Lashgari, M. Eslami, and B. Khan, "Golden search optimization algorithm," *IEEE Access*, vol. 10, pp. 37515–37532, 2022, doi: 10.1109/ACCESS.2022.3162853.

BIOGRAPHIES OF AUTHORS

Purba Daru Kusuma received his bachelor, and master's degrees in electrical engineering from Bandung Institute of Technology, Indonesia, in 2002 and 2009 respectively. He received his doctoral degree in computer science from Gadjah Mada University, Indonesia, in 2017. Currently, he is an assistant professor in Telkom University, Indonesia. His research area is artificial intelligence, machine learning, and operational research. He can be contacted at email: purbodaru@telkomuniversity.ac.id.

Ashri Dinimaharawati received her master's degree in electrical engineering from Bandung Institute of Technology, Indonesia in 2016-2018. She is a Computer Engineering lecturer at Telkom University, Bandung, West Java, Indonesia. Her research area software engineering, game digital, and programming. She can be contacted at email: ashridini@telkomuniversity.ac.id.