

Research Article

A Comparative Performance Investigation of Swarm Optimisers on the Design of Hydrostatic Thrust Bearing

Ismail Sahin 

Gazi University, Technology Faculty, Department of Industrial Design Engineering, Teknikokullar-06500, Ankara, Turkey

Correspondence should be addressed to Ismail Sahin; isahin@gazi.edu.tr

Received 14 June 2020; Revised 8 July 2020; Accepted 21 August 2020; Published 1 September 2020

Academic Editor: Tingsong Wang

Copyright © 2020 Ismail Sahin. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In the design of hydrostatic thrust bearings, power loss that occurs during operation is an important parameter that affects the design, and due to such features, it falls within the interest of design optimisation studies. The fact that the decimal places of the constraints and design variables used for minimum power loss optimisation of hydrostatic thrust bearings are highly effective on the result is a challenge for the design optimisation studies carried out on the problem and has yet made it rather attractive for the researchers. In this study, it is this feature of the problem that makes it the most important motivator in researching the performance of different metaheuristic optimisers in solving the minimum power loss problem. To this end, 7 different optimisers, four of them for the first time, were applied under equal conditions with various pop sizes and a number of iterations, and their performances were addressed under this challenging benchmark problem. The performances of these methods were compared to each other. In addition to the success of optimisers in reaching a solution, their performance in different populations and iterations is also discussed in the study. Considering the results, it is seen that MVO is the most effective optimiser in solving the problem and is followed by the WOA, PSO, and GWO. The application of WOA, MVO, CS, and SSA, for the first time, on the problem has exhibited that these methods could be used in optimisation of such delicate engineering problems.

1. Introduction

Plain bearings are the machine components of choice by virtue of their vibration, impact, and noise damping properties where high speed and load are required, and it is a fact that they have long operating lives. Plain bearings are classified as journal, linear, and thrust bearing, depending on the direction of the load they carry. Thrust bearings can be examined in three groups as hydrostatic, hydrodynamic, and hydrostatic-hydrodynamic bearings [1].

In hydrostatic thrust bearings, pressure is applied to balance the external force and separate the surfaces [2]. With the pressure generated by an external pump, oil is conveyed on the bearings and the bearing surfaces are separated by the oil film. As there is no contact between moving parts, it allows hydrostatic thrust bearings to have lower friction, wear, and vibration [3]. Due to their advantages such as higher accuracy, higher load capacity, incomparable smoothness, higher hardness, and lower friction values that

may be achieved in motion and positioning, hydrostatic thrust bearings are widely used in the industry [4–6]. The power loss that occurs during operation and the increase in oil temperature are considered performance measures in optimum hydrostatic thrust bearing design [7]. For this reason, minimisation of power loss during operation has been a core issue in the studies aimed at the optimisation of hydrostatic thrust bearings.

In the studies carried out for optimum design of hydrostatic thrust bearings, metaheuristic methods are heavily used [1, 2, 8–13]. Studies conducted on this subject are recently focused on swarm intelligence methods. Swarm intelligence algorithms mimic the intelligence of swarms, herds, or flocks of creatures in nature [14].

In this study, the power loss minimisation problem of hydrostatic thrust bearings defined by Siddall [12] was attempted to be solved through 7 different optimisation methods. The performances of the swarm intelligence approaches used in the study for the minimisation problem of

power loss during operation of hydrostatic thrust bearings were comprehensively discussed. The problem is interesting, compelling, and yet utterly attractive by reason of the fact that the decimal places of the active constraints and design variables are highly sensitive on the results. This has made it a rather good benchmark problem [11]. This structure of the problem makes it difficult to achieve feasible results. In the study, more delicate search was carried out in the search space to overcome this difficulty. The challenging structure of the problem was an important motivator in the research of the performance of different metaheuristic optimisers to produce optimum solutions. With this motivation, the problem was solved through popular swarm intelligence optimisers such as particle swarm optimisation (PSO), multiverse optimiser (MVO), grey wolf optimiser (GWO), cuckoo search (CS), whale optimisation algorithm (WOA), salp swarm algorithm (SSA), and artificial bee colony (ABC). Out of these, MVO, CS, SSA, and WOA methods have been applied to the problem for the first time. The performances of these methods were compared with each other. Another important goal of the study is to investigate the impact of population size and number of iterations on the solution. For this reason, the population size was selected as 100, 400, and 800 and the number of iterations was selected as 100, 1000, and 5000.

The remainder of this article is organised as follows: in Section 2, a literature search in which the studies on the subject are examined is presented; in Section 3, the problem is thrust bearing; in Section 4, optimisation methods used in experimental studies are introduced; in Section 5, comparisons of experimental results obtained with optimisers are explained; and Section 6 includes the conclusion and suggestions for future studies.

2. Related Works

The advantages of hydrostatic thrust bearings date back to 1940s. Automatic fluid pressure balancing system of Hoffer is considered the pioneer of hydrostatic thrust bearings [15]. Over the next half-century, a series of patent and research studies carried out covering topics such as the structure of hydrostatic thrust bearings, oil flow, balance problems, and infinite stiffness and have formed the basis of modern hydrostatic thrust bearing designs [16–21]. The hydrostatic system developed by Slochum et al. for use in high-pressure press benches was the first hydrostatic bearing [20]. In this new system, the support equipment used for the shaft's bearing had high strength and friction resistance.

In the subsequent studies, it was observed that oil film thickness, recess pressure, pressure distribution, and oil flow rate were effective on the performance of hydrostatic bearings [22]. These characteristics affect high wear resistance, which is the most important feature expected from thrust bearings [23]. Bearing materials are expected, in their selection, to have features such as lower friction coefficient, higher wear resistance, higher loading capacity, better erosion resistance, better thermal conductivity, and lower thermal expansion. However, oil viscosity, oil film thickness, oil flow rate, and pressure amount also have significant effects on the

performance of the system. In addition to experimental and analytical studies aimed at achieving the optimum values of these features, computer aided studies date back to early 1980s. Siddall's study, in which the problem of minimising power loss during operation of hydrostatic thrust bearings is described, is one of the pioneering studies in this respect [12]. Siddall reduced the power loss in existing bearings by 2,288.0 ft-lb/s (4.14 hp) through the Hooke and Jeeves (HJ) method.

Siddall's problem has become a benchmark problem used for testing intelligent optimisation approaches today. Among them, population-based approaches stand out. These studies used the minimisation of the power loss of hydrostatic thrust bearings extensively, inspired by Siddall's identified problem, in testing the metaheuristic methods they developed. Deb and Goyal's genetic adaptive search (GeneAS) method is the first of these [9]. Deb and Goyal demonstrated that the thrust bearing they designed could withstand more weight with smaller film thickness and would exhibit lesser power loss as per the algorithm of Siddall. Afterwards, Coello achieved faster and better results using a multipurpose optimisation technique in place of the penalty functions used in GA [8]. Another study focused on inconsistencies in previous studies with regard to unit and design criteria, using the particle swarm optimisation method [10]. Improved PSO algorithm of He et al. reduced power loss by around 30% compared to Siddall's study. Rao et al. achieved good results through the teaching-learning-based optimisation (TLBO) method they developed, compared to the studies in the literature [11]. Kentli and Sahbaz have solved the problem through sequential quadratic programming (SQP) method and compared the results with Siddall's study [1]. Sahin et al. [2] have developed a mathematical model of GWO, a highly popular population-based approach in recent years, and solved the problem through a new model (enhanced GWO) that increases the diversity of the valid solutions by keeping the search area wider [2]. In the study, all the studies in the literature were examined together, and a comprehensive comparison was made and very successful results were obtained through E-GWO. Talatahari and Azizi solved the problem through chaos game optimiser in an up-to-date study [13]. Best fitness results and design variable values obtained in studies that address the minimum power loss problem are presented in Table 1.

The optimisation of features such as maximum load-carrying capacity of bearings [24–26], stiffness [26], film thickness [7], and optimal recess shape [4, 27] is among other important research topics in hydrostatic thrust bearings.

3. Hydrostatic Thrust Bearing Design for Minimum Power Loss

The hydrostatic thrust bearing design problem discussed as part of this study was defined by Siddall [12]. The purpose function of the problem in which minimisation of power loss is targeted during the operation of a thrust bearing exposed to the axial load seen in Figure 1 is formulated in the following equation [12]:

TABLE 1: Algorithms and its optimal solutions for minimum power loss problem in the literature.

	Variables				Best fitness $f(x)$
	R	R_0	μ ($\times 10^{-6}$)	Q	
Siddall [12], HJ	7.1550805	6.688682	8.320765	9.168461	29221.321
Deb and Goyal [9], GeneAS	6.778	6.234	6.096	3.809	25937.058
Deb and Goyal [9], BGA	7.7077	6.549	6.619	4.849	27554.5428
Coello [8], GASO	6.271	12.901	5.605	2.938	23403.432
He et al. [10], PSO	5.956868685	5.389175	5.402133	2.301546	19586.5788
Rao et al. [11], TLBO	5.9557805	5.389013	5.358697	2.269655	19505.3131
Kentli and Sahbaz [1], SQP	5.95580	5.38904	8.63332	8.00001	26114.545
Şahin et al. [2], GWO	5.9576206	5.390681	0.000005	2.278085	19530.6552
Şahin et al. [2], E-GWO	5.9557805	5.389013	5.358697	2.269656	19505.3134
Talatahari and Azizi [13], CGO	5.963440023	5.395587861	5.36	2.264822	19454.9541

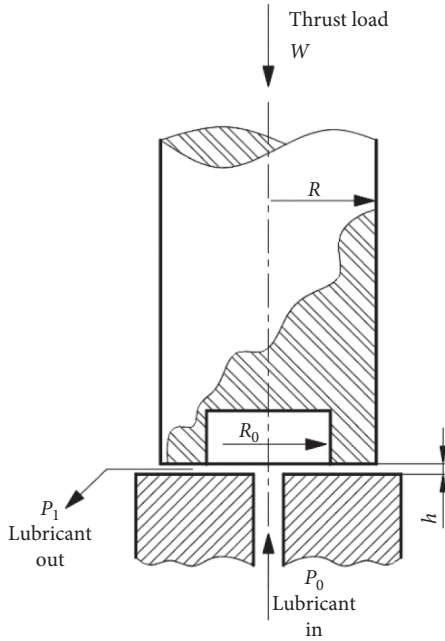


FIGURE 1: Hydrostatic thrust bearing.

$$\text{minimise: } f(x) = \frac{QP_0}{0.7} + E_f. \quad (1)$$

Four design variables were used in the problem [12]: flow rate (Q), recess radius (R_0), bearing step radius (R), and viscosity (μ) (equation (2)).

Design vector:

$$X = (Q, R_0, R, \mu). \quad (2)$$

The value ranges of the design variables of the problem are as follows:

$$\begin{aligned} 1,000 &\leq R \leq 16,000, \\ 1,000 &\leq R_0 \leq 16,000, \\ 1.0 \times 10^{-6} &\leq \mu \leq 16 \times 10^{-6}, \\ 1,000 &\leq Q \leq 16,000. \end{aligned} \quad (3)$$

Seven nonlinear constraints were identified in the problem (equations (4)–(10)). These are minimum load-carrying capacity, weight capacity, inlet oil pressure, oil

temperature rise, oil film thickness, step radius, exit loss, and contact pressure [2, 10, 12]. Out of 7 constraints, 6 are active constraints considering an accuracy of 3 decimal places, and all the design variables are highly sensitive in the optimisation problem subject to

$$g_1(x) = W - W_s \geq 0, \quad (4)$$

$$g_2(x) = P_{\max} - P_0 \geq 0, \quad (5)$$

$$g_3(x) = \Delta T_{\max} - T_0 \geq 0, \quad (6)$$

$$g_4(x) = h - h_{\min} \geq 0, \quad (7)$$

$$g_5(x) = R - R_0 \geq 0, \quad (8)$$

$$g_6(x) = 0.001 - \frac{\gamma}{gP_0} \left(\frac{Q}{2\pi Rh} \right) \geq 0, \quad (9)$$

$$g_7(x) = 5000 - \frac{W}{\pi(R^2 - R_0^2)} \geq 0, \quad (10)$$

where $g_1(x)$ is for weight capacity, which must be greater than weight of generator, $g_2(x)$ is for inlet oil pressure, $g_3(x)$ is for oil temperature rise, $g_4(x)$ is for oil film thickness, $g_5(x)$ is for step radius and must be greater than recess radius, $g_6(x)$ is for limits on significance off exit loss and must be greater than 0001, and $g_7(x)$ is the limit for contact pressure and must be greater than 5000.

The parameters given in the constraints are calculated by the following equations:

$$\begin{aligned} W &= \left(\frac{\pi P_0}{2} \right) \left[(R^2 - R_0^2) \left(\ln \frac{R}{R_0} \right) \right], \\ P_0 &= \frac{6\mu Q}{\pi h^3} \ln \frac{R}{R_0}, \\ E_f &= 9336Q\gamma C\Delta T, \end{aligned} \quad (11)$$

$$\Delta T = 2(10^P - 560),$$

$$P = \frac{\log_{10} \log_{10} (8.122e6\mu + 0.8) - C_1}{n},$$

where W is the load-carrying capacity, P_0 is the inlet pressure, E_f is the friction loss, ΔT is the temperature, γ is the weight density of oil (0.0307 lb/in^3), C is the specific heat of oil ($0.5 \text{ Btu/lb}^\circ\text{F}$), n and C_1 are the oil constants, and h is the film thickness. $C_1 = 10.04$ and $n = -3.55$ were chosen for SAE 20 grade oil (see Table 2).

Other specifications of the design problem are as follows: W_s (weight of generator) = 101000 lb (45804.99 kg); P_{\max} (maximum pressure available) = 1000 psi ($6.89655 \times 10^6 \text{ Pa}$); ΔT_{\max} (maximum temperature rise) = 50°F (10°C); h_{\min} (minimum oil thickness) = 0.001 in. (0.00254 cm); $g = 32.3 \times 12 = 386.4 \text{ in./seg}^2$ (981.465 cm/seg^2), and N (angular speed of shaft) = 750 RPM.

4. Optimisation Methods Applied to the Problem

In this study, seven population-based optimisation methods were applied to the problem. These are PSO, ABC, GWO, MVO, CS, WOA, and SSA. MVO, CS, WOA, and SSA optimisation methods were applied to the problem for the first time.

4.1. Salp Swarm Algorithm (SSA). The salp swarm algorithm (SSA) is an optimisation algorithm developed, inspired by the swarming mechanism of salps resembling jelly fish [14]. The inspiration for the method is the swarming behaviour that salp chains exhibit while searching and collecting in the ocean. In the SSA swarm model, the leading salp moving towards the food source is followed by the follower salps. If the food source is replaced with a global optimum, the salp chain automatically moves towards it. The results in mathematical functions exhibit that the SSA algorithm can effectively develop the initial random solutions and zoom closer to the optimum level [28].

4.2. Cuckoo Search (CS). CS is a metaintuitive search algorithm inspired by the reproductive strategy of cuckoo birds developed by Yang and Deb [29]. The method is based on some cuckoo species placing their eggs in the nests of other cuckoos or different species. In this method called the brood parasitism method, the cuckoo that desires to increase the chances of its own eggs may destroy other eggs, engage directly with other birds, or leave the nest [30]. Yang and Deb have set three rules to use this to solve optimisation problems [29]:

- (i) Each cuckoo bird leaves one egg at a time in a randomly selected nest.
- (ii) The best nests with high quality eggs are conveyed onto future generations.
- (iii) The number of host nests is constant. Likelihood of the landlord finding, a foreign egg is $pa \in [0, 1]$. In this case, the host bird can throw the egg or leave the nest to build a new nest.

4.3. Multiverse Optimiser (MVO). Developed by Mirjalili and his colleagues, MVO is inspired by the theory of multiverse and big bang theory in physics [31]. In the

TABLE 2: Values of n and C_1 for various grades of oil.

Oil	C_1	N
SAE 5	10.85	-3.91
SAE 10	10.45	-3.72
SAE 20	10.04	-3.55
SAE 30	9.88	-3.48
SAE 40	9.83	-3.46
SAE 50	9.82	-3.44

method in which the mathematical model of white hole, black hole, and worm hole concepts in cosmology is created, the search process is a two-step process, as in other population-based methods. But here, white hole and black hole concepts are used in the discovery of search spaces. Worm hole supports MVO in exploiting the search spaces. In MVO, each solution is assumed to resemble a universe, and every variable in the solution is an object in this universe. Another difference of MVO compared to other algorithms is the use of the concept of time, which is a common term in multiverse theory and cosmology [31].

4.4. Artificial Bee Colony (ABC). ABC is a swarm intelligence-based algorithm developed by Karaboga [32]. The algorithm was inspired by the behaviour of honey bees that collect nectars. Bees are divided into three groups: scout, onlooker, and employed bees. Worker bees seek food in nature with a special dance (the waggle dance) and communicate with each other. The onlookers watch the dance to make a choice and follow the bee to the food source [33]. Scooters discover abandoned food sources and replace them with new resources. In this method, the location of food refers to the possible solution to the optimisation problem and the amount of nectar in food refers to the suitability of the solution [34].

4.5. Particle Swarm Optimisation (PSO). PSO, one of the most popular approaches to swarm intelligence, was developed by Kennedy and Eberhart [35]. The algorithm is inspired by the foraging and navigation capabilities of bird flocks. The algorithm consists of particles that are placed in a search space and move around by combining their own previous location and the current global optimal solution. Potential solutions are encoded as randomly initiated particles and directed to move within the search space to find the most appropriate solution [36]. In PSO, the global optimum solution is searched taking into account the speed and location of each particle that makes up the flock [34].

4.6. Whale Optimisation Algorithm (WOA). WOA, which is based on the modelling of the hunting behaviour of humpback whales, was developed by Mirjalili and Lewis [33]. Humpback whales exhibit a three-stage behaviour, namely, search for prey, encircling prey, and bubble-net foraging, as they hunt. In WOA, this behaviour is mimicked.

The method uses updating whales (individuals) their position based on the position of the prey as base as in GWO.

4.7. Grey Wolf Optimiser (GWO). GWO was developed by Mirjalili and Lewis [37], inspired by the hunting method of grey wolves in nature and the social hierarchy within the pack. In GWO, each solution in the population corresponds to a wolf in the pack. There are four different hierarchical levels in the pack, which are called alpha, beta, delta, and omega. Mathematical model of GWO is constituted by this hierarchical structure and hunting method [38].

5. Results and Discussion

This study presented is aimed at minimising the power loss during operation of hydrostatic thrust bearings formulated by Siddall [12]. Siddall's ADRANS algorithm was based on the Hooke and Jeeves (HJ) pattern search method for the solution. In the study, the achievements of SSA, MVO, CS, GWO, ABC, WOA, and PSO methods were tested in solving the problem using the purpose function, design constraints, design vector, and parameters defined by Siddall.

In the first phase of experimental studies, statistical achievements of the algorithms were evaluated. Algorithms were run on a computer with an Intel® Core™ i7, 2.6 GHz CPU, and 8 GB RAM, running on the Windows 10 × 64 operating system using Python 3.7 software. In order to investigate the impact of population size and the number of iterations on the solution, the population was selected as 100, 400, and 800 in all algorithms and the number of iterations, defined as the stop criteria, was set to 100, 1000, and 5000. The algorithms were run 20 times for each pairwise of population size and number of iterations.

Siddall used the inches-pounds per second as unit of fitness value in his study. However, some studies were confirmed to use feet-pounds per second [9, 11, 13]. In this study, calculations were made in the units used by Siddall, sticking to the original state of the problem.

Each algorithm was run 20 times, and best, worst, average, and success percentage (SP) values of the results were calculated. SP takes the results obtained after running of the algorithm in different numbers and demonstrates the stability of the algorithm. In this context, the difference between the best value (F) achieved and the global optimum value achieved must be less than 0.1% of the global optimum (F^*) value (equation (12)), in order for the result to be considered successful:

$$\left| \frac{F^* - F}{F^*} \right| < 1E - 3. \quad (12)$$

5.1. Statistical Comparison of the Algorithms. The study aims to observe the impact of different population sizes and number of iterations on the achievement of the optimisers. Therefore, the performance of the algorithms is with 100, 400, and 800 population sizes and numbers of iterations is 100, 1000, and 5000 (Table 3). As seen in Table 3, the lowest objective function value in the study was achieved by the

MVO with a population of 800 and in 5000 iterations, as $f(x) = 19508,39014$. The results obtained by other algorithms, except for the ABC, are also quite successful. After MVO, the lowest objective function value was achieved by WOA, PSO, and GWO. The power loss value obtained with WOA is only 0.003% behind of MVO. SSA, another algorithm that is applied on the problem, has, however, achieved worse values than that of MVO by as much as 0.88%. This demonstrates that MVO, WOA, and SSA are rather successful in solving the power loss minimisation problem. Worst objective function values were materialised in ABC and CS as $f(x) = 22830,65571$ and $f(x) = 19950,96346$, respectively. With PSO, the third lowest objective function value was achieved following MVO and WOA. This result obtained is better than results of previous PSO study [10].

The performance of algorithms in solving the problem with different population sizes and in iterations differs. PSO, for example, performed poorly in low numbers of iteration. As can be seen from the table, performance of PSO in lower population sizes (100 and 400) and lower iteration numbers (100 and 1000) is worse compared to the population size and number of iterations where it achieved the best fitness. MVO has achieved the best objective function results in all populations and iterations except 100 populations and 5000 iterations. MVO has generated the best results at 5000 iterations for the whole population. This exhibits that MVO performs decisively. Algorithms other than ABC and CS exhibited closer performance at best fitness values at 400 and 800 populations and 1000 iterations. All algorithms except SSA and CS yielded the most successful fitness values during the experiments conducted with a population size of 800 and 5000 iterations.

As the number of iterations increased, success also increased in algorithms. It is not, however, possible to come up with the same conclusion for the results related to increase in population size. For instance, SSA, which acquired the lowest power loss in population size of 400, did not attain the same performance as the population size increased. The solution CS achieved with a population size of 800 is worse than what it achieved with a population size of 400. This may, consequently, indicate that the number of iterations is more effective on the solution than the population size.

In Table 4, the statistical results of algorithms were compared with each other. These values are the values obtained in a population size of 800 in 5000 iterations, where the best fitness value is achieved. In Table 4, the best optimal power loss value is seen to be achieved by MVO. In parallel with this, it can be seen that MVO is more successful, compared to other studies in terms of SP, mean, and worst values. MVO exhibited a steady achievement in terms of success percentage. MVO, with an SP rate of 1, is followed by PSO, GWO, and WOA with an achievement rate of 0.85, 0.40, and 0.15, respectively. However, WOA has the second-best fitness value, and the SP rate occurs to be 0.15. SP rates of SSA, ABC, and CS were 0.00.

In Table 5, the values of design variable used for solutions that occur in 800 populations and 5000 iterations, where the best objective function value is achieved, are given comparatively. When the table is analysed, it will be seen that the

TABLE 3: Best, worst, and mean fitness results of the algorithms at different populations and iteration sizes.

Algorithms	Population size # iterations	100			400			800		
		Worst	Mean	Best	Worst	Mean	Best	Worst	Mean	Best
ABC	100	97880,22377	59756,19037	35199,122	61438,2	38997,04537	23972,79965	45816,66109	35635,12678	25911,96344
	1000	42477,76908	33791,24852	25929,378	36734,93	30543,36807	25276,78226	37473,99432	30138,08625	23780,59516
	5000	33425,55472	28379,06693	24551,577	30956,61847	26934,92081	23447,8621	26339,46279	24666,75976	22830,65571
CS	100	86472,45356	52863,75942	37332,47	73568,12	46060,36489	22864,9523	49565,0751	37655,52983	29197,89212
	1000	33805,431	28370,40843	24409,907	29135,1	26901,81866	22247,30483	28196,3445	26395,48853	23922,48927
	5000	21419,78879	20663,07987	20165,94	20842,00943	20455,23573	19926,80827	20466,14966	20297,33609	19950,96346
GWO	100	96943,09892	49335,12227	25103,291	49501,18	29989,26663	21571,10275	46829,225	26831,94961	20747,87574
	1000	80124,80937	51458,42565	23085,525	28241,92	20625,69807	19629,68579	20070,02109	19742,61778	19562,81207
	5000	36928,31868	24291,30478	19586,232	19673,73006	19575,76898	19542,05678	19556,18396	19533,09898	19516,83771
MVO	100	36290,93605	28601,01781	24000,85	33412,84	26288,74334	20299,44124	31998,05736	24091,47227	19827,22283
	1000	28544,07865	21942,57516	19605,425	20334,92	19739,74148	19537,59644	20306,92897	19632,75373	19517,10515
	5000	25179,91496	19985,30306	19529,24	19937,75722	19595,30664	19514,8083	19517,78342	19511,16351	19508,39014
PSO	100	41159,18268	32404,95809	26166,869	34563,34	28483,1693	24047,96276	30706,99491	26603,35596	23125,08598
	1000	27831,00564	23986,12466	21747,726	25286,4	21347,28043	20117,52228	21623,69067	20614,88738	19948,8345
	5000	20707,32489	19861,48152	19588,604	19638,34448	19560,9795	19520,64307	19529,52256	19522,63534	19511,3608
SSA	100	40696,77805	30843,76731	27046,163	41522,23	31848,27576	23198,94943	40628,97117	31973,0032	26916,72938
	1000	63824,98431	31286,77464	21030,705	31282,91	25501,85922	20211,64085	47549,89143	28910,74458	21365,3382
	5000	28169,30759	23867,89307	19525,274	30032,02749	23406,51657	19572,13612	26034,91429	22685,63005	19671,90412
WOA	100	56481,05312	43441,94424	24296,775	51397,87	33718,33034	22550,37552	41141,30307	27294,16099	20671,39233
	1000	31215,27977	25190,74613	20333,709	41090,55	25311,34203	19764,81337	23666,92413	20918,93497	19522,75021
	5000	35441,19553	23217,44358	19595,925	23441,77209	20477,36342	19521,03023	21765,56684	20590,50532	19509,09474

TABLE 4: Statistical comparison of best fitness value of the algorithms.

Algorithm	Best ($f(X)$)	Worst ($f(X)$)	Mean ($f(X)$)	SP
ABC	22830,65571	26339,46279	24666,75976	0
CS	19950,96346	20466,14966	20297,33609	0
GWO	19516,83771	19556,18396	19533,09898	0.4
MVO	19508,39014	19517,78342	19511,16351	1
PSO	19511,3608	19529,52256	19522,63534	0.85
SSA	19671,90412	26034,91429	22685,63005	0
WOA	19509,09474	21765,56684	20590,50532	0.15

TABLE 5: Design variables for the best solutions.

Algorithms	Variables			
	R_0	R	$\mu (10^{-6})$	Q
ABC	6,42110644	5,87339441	6,03021956	3,27847435
CS	5,99669606	5,42618025	5,47140755	2,39732714
GWO	5,95622341	5,38916513	5,36189828	2,27293937
MVO	5,95590533	5,38909289	5,35983982	2,27070608
PSO	5,95635114	5,38953695	5,36015715	2,27129760
SSA	5,95578188	5,38901445	5,48694850	2,36847888
WOA	5,95584921	5,38908429	5,36133333	2,27169540

MVO with the lowest objective function value also has the optimum variable values. Looking at the variable values of MVO and WOA, it can be seen that there is little difference between them, which is compatible with best fitness results. When the results of design variables are considered, it can be said that PSO, MVO, and WOA are very competitive approaches in obtaining minimum power loss and are followed by GWO.

If a comment is to be made, in terms of the algorithms that are applied on the problem for the first time, it can be concluded that MVO and WOA produce high-quality solutions both in objective function values as well as in variable values.

5.2. *Convergence Performances of Optimisers.* In this section, the convergence graphs with population sizes of 100, 400, and 800 in 100, 1000, and 5000 iterations are presented in order to comparatively see the convergence ratios of the algorithms in all population sizes and number of iterations. A general interpretation of the convergence speeds and solution performance of algorithms in different population sizes and number of iterations is thus made.

In Figure 2, convergence rates of algorithms with population of 800 and 5000 iterations where the best result is obtained across all populations and iterations are presented. As can be clearly seen from the figure, although MVO had a slower convergence in comparison with PSO and WOA in the beginning, it reached the best fitness value by surpassing PSO and WOA in the last iterations. WOA reached the best fitness value after the 1764th and PSO after the 4670th iterations. GWO, which exhibited a similar convergence performance to MVO, reached the best fitness value in the last iterations. ABC and CS demonstrated the worst convergence performances, respectively. Considering the

change in the convergence curves, the performance exhibited by WOA and PSO towards solution is assessed to be more stable. There are gradual changes in the convergence curves of MVO, GWO, and SSA. Among these algorithms, MVO and GWO achieved the best fitness values in the last iterations. SSA reached its best fitness value after the 4120th iteration.

The fastest convergence and best quality solutions in a population size of 100 and 100 iterations are seen in WOA and MVO (Figure 3). They are followed by GWO, SSA, and PSO. Although SSA is seen to be converging faster than GWO after the 48th iteration, GWO exhibits a better convergence after the 85th iteration. WOA reaches its best values at iteration #81 and MVO at iteration #100. Although MVO has a slower convergence rate than that of WOA, it is the algorithm that reaches the best solution. CS and ABC are the algorithms with the worst performance in convergence.

As shown in Figure 4, the convergence of WOA, MVO, and SSA in population of 100 and 1000 iterations is better than that of GWO and PSO. CS and ABC exhibits the worst convergence performances. Although WOA reaches its best value (616th iteration) before MVO, MVO reaches the best solution by exhibiting the best convergence performance after the 890th iteration ($f_x = 19605.42531$). MVO and WOA are followed by SSA and PSO.

With a population of 100 and 5000 iterations, WOA, CS, and PSO are seen to converge faster initially and they are followed by SSA, MVO, and GWO (Figure 5). As can be seen in the convergence graph in Figure 5, SSA exhibits better convergence performance in the last 1500 iterations and reached the best solution. Exhibiting rather slow convergence in the beginning, MVO reached the 2nd best solution after demonstrating a faster convergence in the last 500 iterations. ABC has the worst convergence performance. It is therefore observed that there is analogy between the

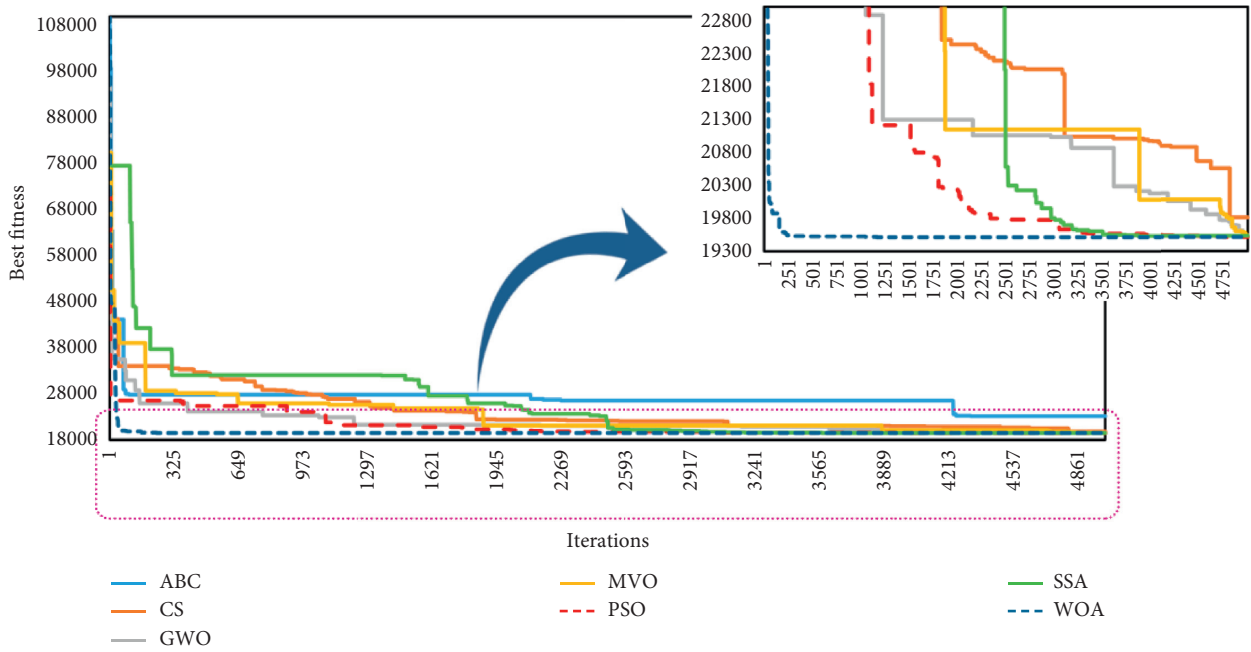


FIGURE 2: Convergence graph for best optimal solutions (in population size of 800 and number of iterations of 5000).

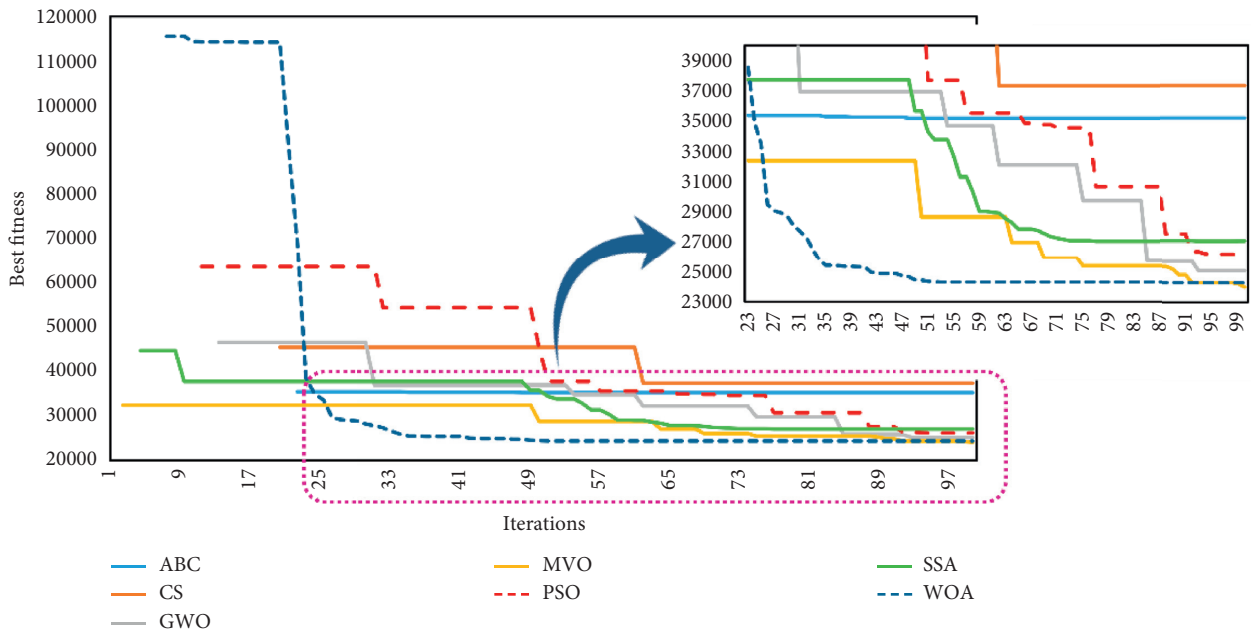


FIGURE 3: Convergence graph for a population size of 100 and number of iterations of 100.

optimum solutions achieved by the algorithms and the convergence speeds.

As seen in Figure 6, having a higher initial convergence speed with population of 400 and 100 iterations, CS is the algorithm with the highest initial convergence speed and it is followed by GWO, WOA, and SSA. While MVO, reaching the best solution ($f(x) = 20299,44124$), with a population of 400 and 100 iterations, has a slower initial convergence, it is seen that it exhibits a faster convergence

after the 30th iteration and reaches the solution. GWO exhibits a faster convergence after iteration #70 and reaches the second-best objective function value of $f(x) = 21571,10275$. PSO and ABC exhibit the worst convergence performances.

In the experiments conducted with a population of 400 and 1000 iterations, as number of iterations of ABC, GWO, PSO, and CS algorithms, excluding PSO, increased, the convergence speed got slower (Figure 7). As can be seen

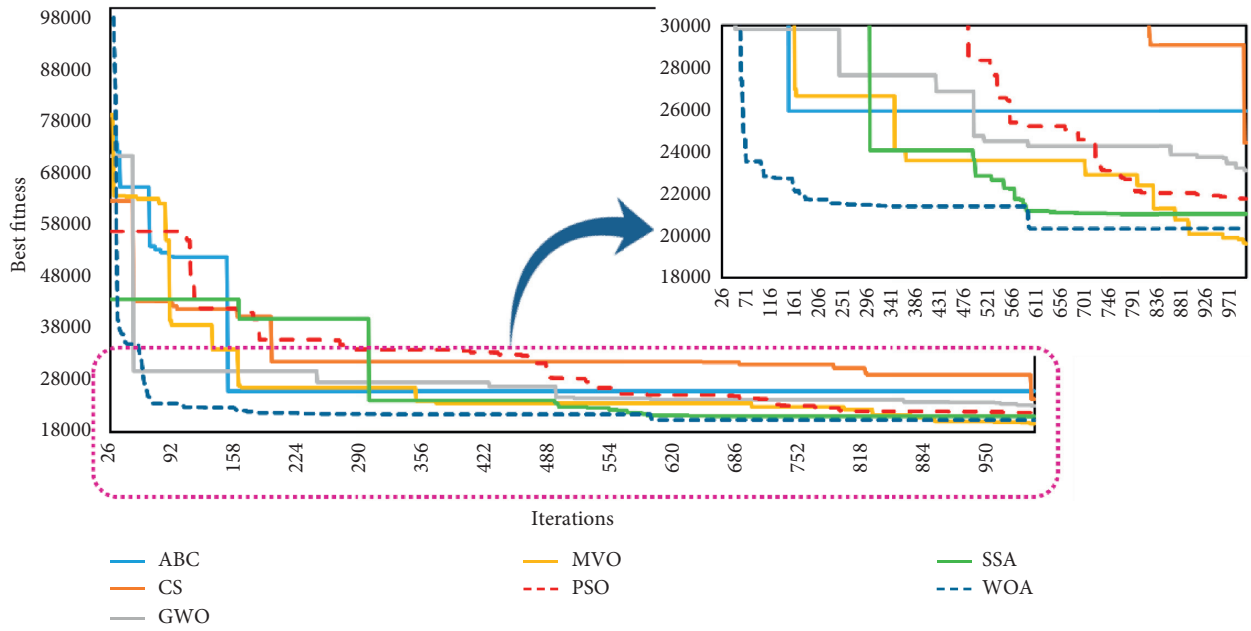


FIGURE 4: Convergence graph for a population size of 100 and number of iterations of 1000.

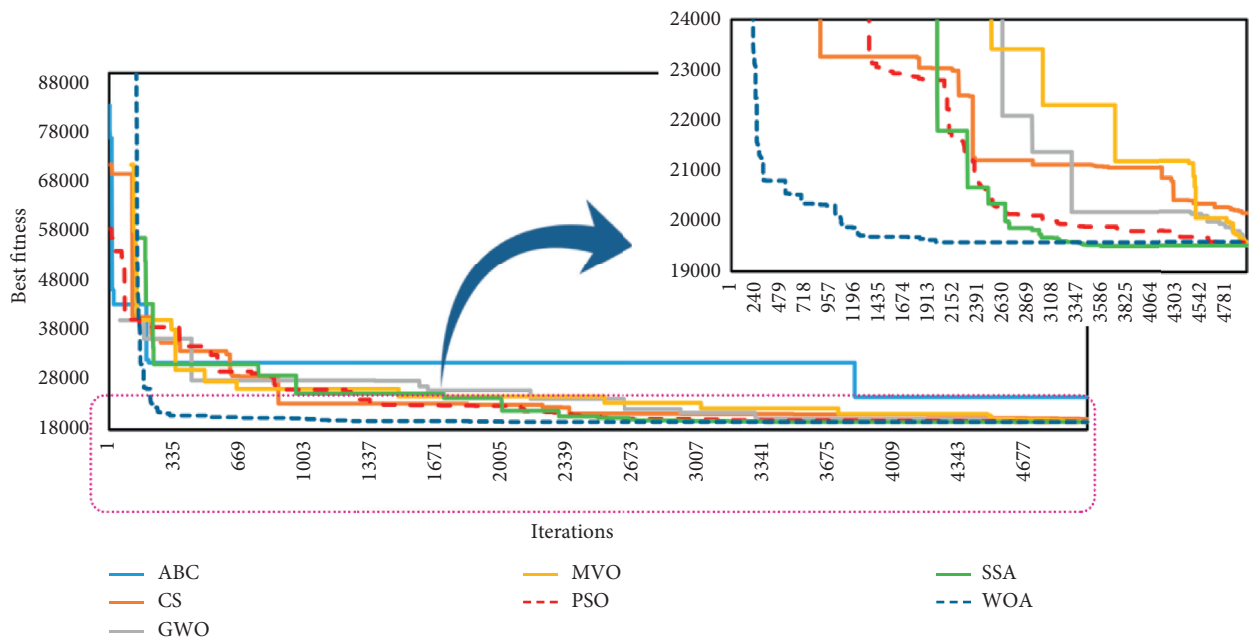


FIGURE 5: Convergence graph for population size of 100 and number of iterations of 5000.

from the convergence graph, having a slower initial convergence speed, WOA exhibits a faster convergence speed after the 80th iteration and SSA after the 570th iteration and they reach the optimal values. MVO, yielding the best solution ($f(x) = 19537,59644$), converges to its optimum value after the iteration #960, and GWO, yielding the second-best solution ($f(x) = 19629.68579$), after iteration #930.

It is seen in Figure 8 that initial convergence rates of WOA, PSO, and GWO are faster than that of ABC, CS, MVO, and SSA in the experiments conducted with a population size of 400

and 5000 iterations. MVO, with a slower initial convergence speed, exhibits a better convergence after the iteration #4500 and generates the best quality solution. PSO exhibits its performance in convergence speed and quality solution, here again, with iteration number of 5000, and generates the second-best objective function value of $f(x) = 19520,64307$. As can be seen from Figure 8, WOA, which converged to the best fitness value in early iterations by showing a rapid convergence in the beginning, was able to produce the third best solution. Having a good initial convergence performance, GWO generates the

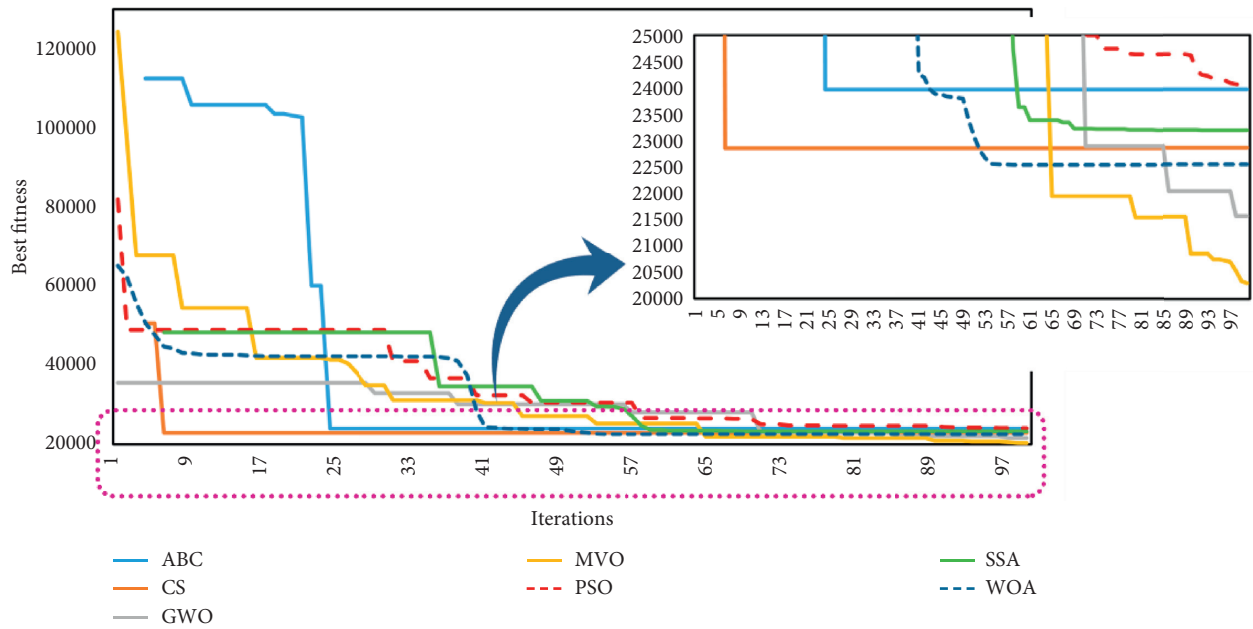


FIGURE 6: Convergence graph for a population size of 400 and number of iterations of 100.

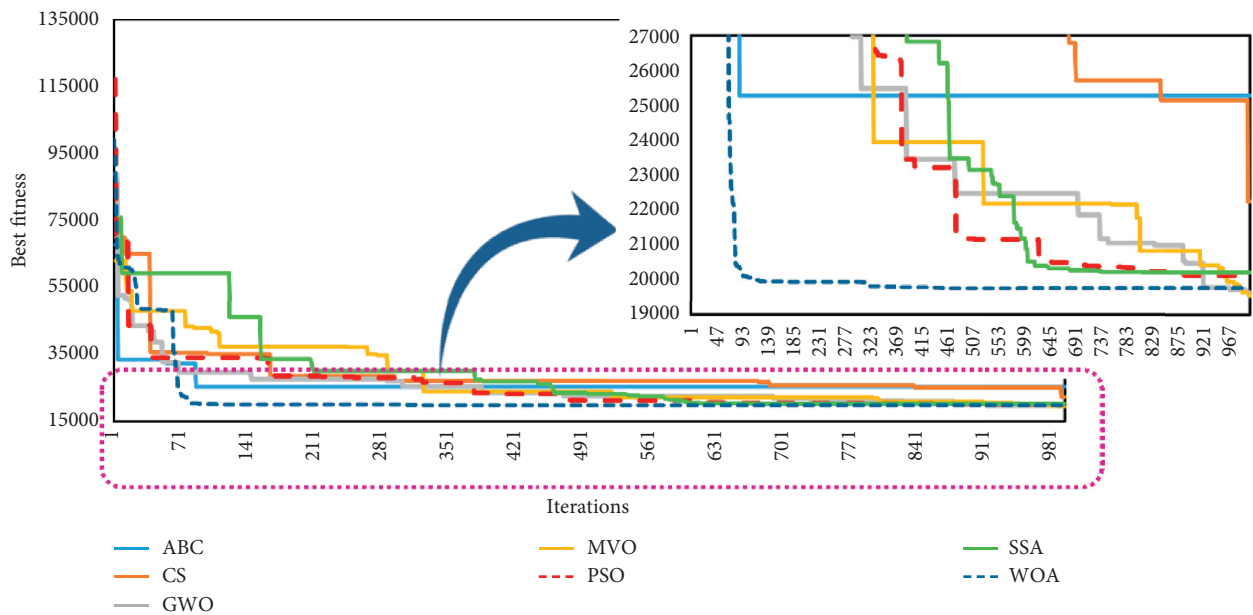


FIGURE 7: Convergence graph for a population size of 400 and number of iterations of 1000.

third-worst solution and ABC generates the worst solutions by exhibiting the worst convergence performance.

Figure 9 shows the convergence graphs of optimisers to their best values with a population size of 800 and 100 iterations. It can be seen here that the convergence rates of WOA, MVO, and GWO are better than that of ABC, PSO, SSA, and CS. While PSO exhibits a faster initial convergence, it exhibits a slower convergence in advancing iterations. In this series, MVO, WOA and GWO present quality solutions both in convergence speed and in best fitness values.

In the experiments carried out with a population size of 800 and 1000 iterations, it can be seen that the convergence speeds of WOA, GWO, and MVO are faster than PSO, SSA,

ABC, and CS (Figure 10). It is seen that MVO, WOA, and GWO generate quality solutions in both convergence speed and best fitness values with this population size and number of iterations. Although the initial convergence of the MVO is slower compared to that of GWO and WOA, it reaches the best objective function value of $f(x) = 19517,10515$.

5.3. Results of Execution Time Analysis. Execution time analysis is an important parameter used for evaluating the performance of optimisers. In the study, the speeds were compared with each other based on the time optimisers took for the solution of the problem where the best fitness value was

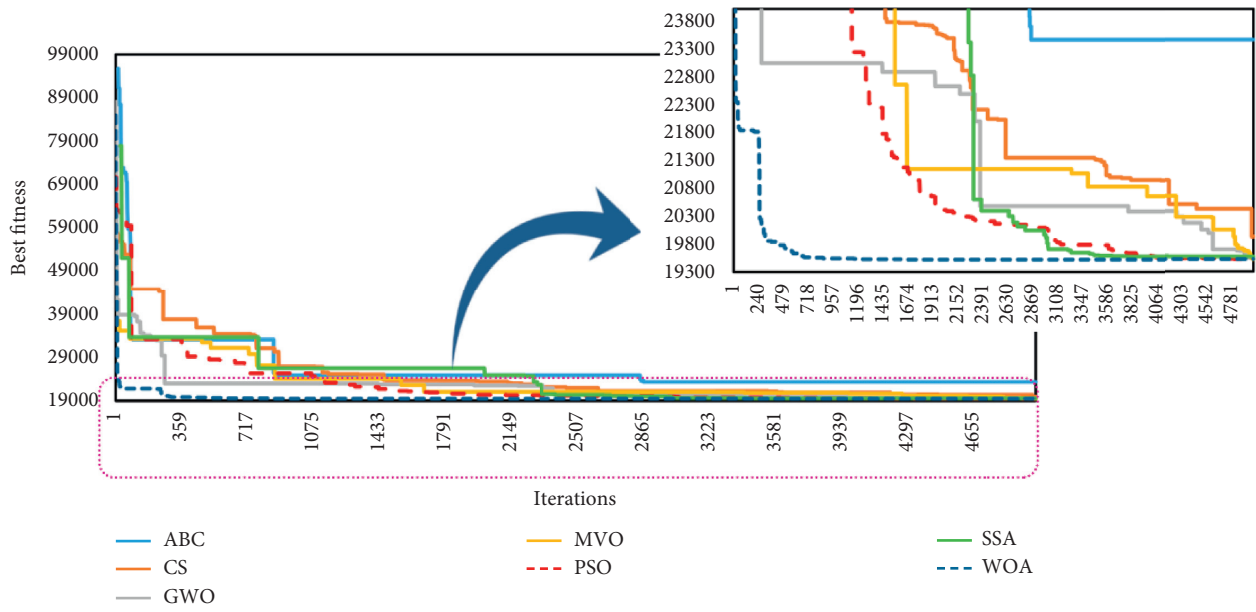


FIGURE 8: Convergence graph for a population size of 400 and number of iterations of 5000.

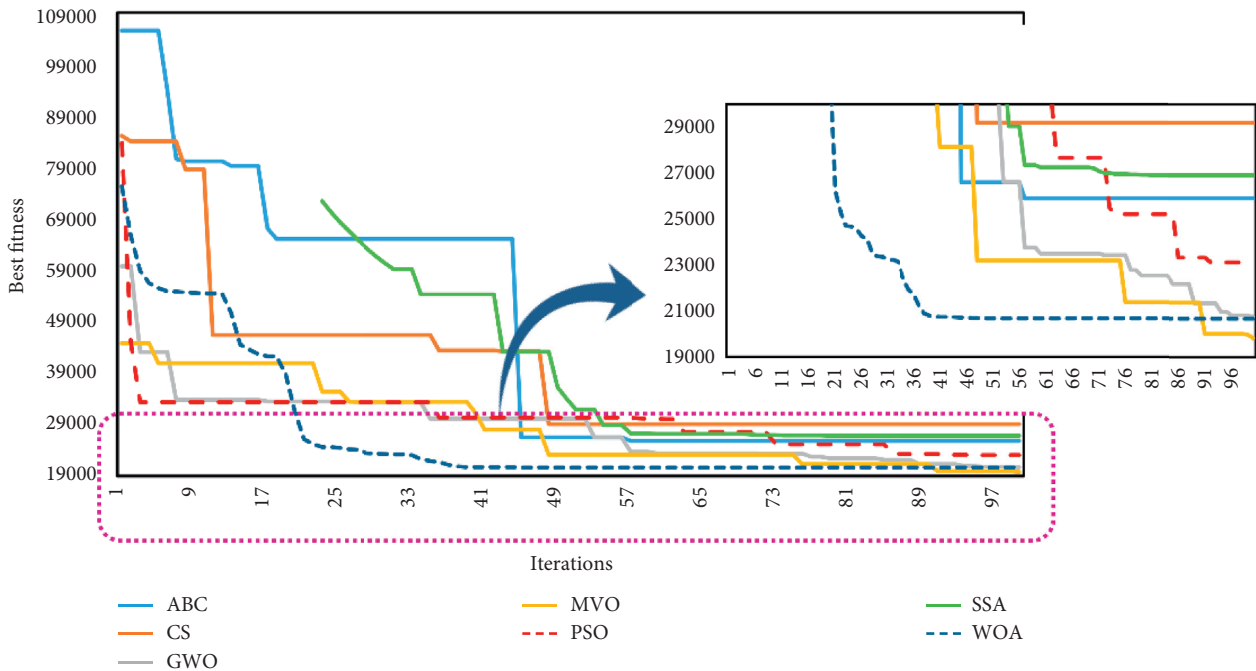


FIGURE 9: Convergence graph for a population size of 800 and number of iterations of 100.

reached with a population size of 800 and iteration number of 5000 and 20 studies (Figure 11). With this comparison, it is possible to construct the time complexity of algorithms. As seen in the graph shown in Figure 11, the longest calculation time for solving the problem is seen in ABC. It is followed by SSA and WOA. Having the best fitness value, PSO achieved this success by solving the problem at an average of 19.51 seconds. PSO, GWO, and MVO, which are most successful ones in solving the problem, are also more successful than other algorithms in terms of average calculation time. The

average running times of algorithms are close to each other. Although algorithms differ in terms of running time, it can be said that, with a general assessment, all algorithms solve the problem in a reasonable running time.

6. Conclusion

In this study, the hydrostatic thrust bearing design problem defined by Siddall [12] is discussed through 7 different swarm intelligence approaches. The purpose of the problem

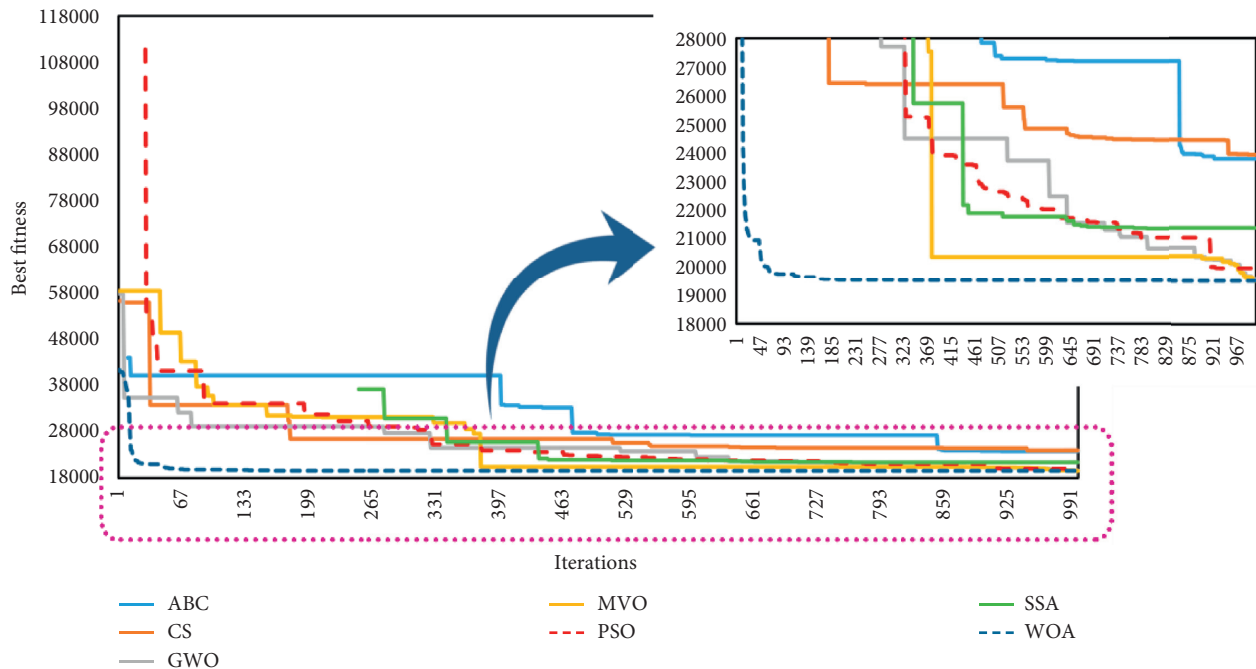


FIGURE 10: Convergence graph for a population size of 800 and number of iterations of 1000.

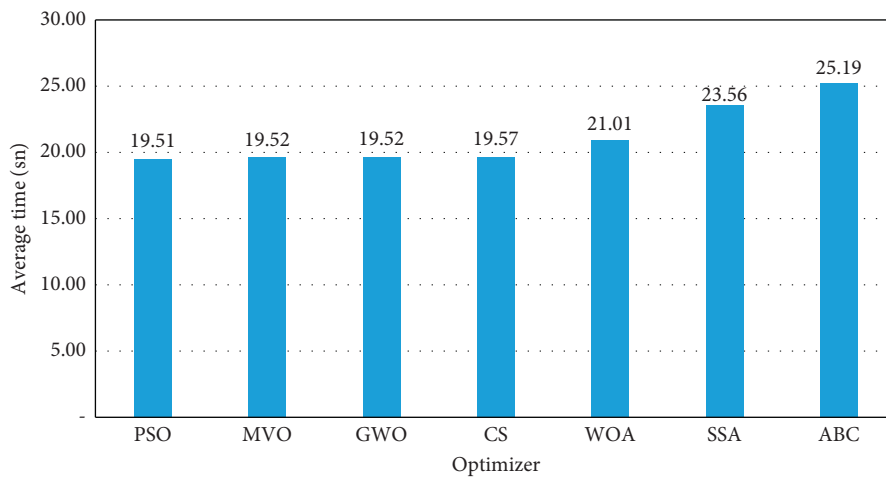


FIGURE 11: Average execution time of all optimisers for a population size of 800 and number of iterations of 5000.

is to minimize the power loss that occurs during operation of hydrostatic thrust bearings. In addition to the successes of 7 different optimisers in achieving an optimum solution, their performances in different population sizes and numbers of iterations are also discussed as the research subject. For this reason, optimisers were run with population sizes of 100, 400, and 800 in 100, 1000, and 5000 iterations. According to the results obtained, the lowest objective function value in the study was obtained by MVO with a population size of 800 and 5000 iterations. Following MVO, the best solution was reached through WOA, PSO, and GWO. The performances of WOA, MVO, CS, and SSA, which were applied for the first time to solve the minimum power loss problem, are particularly successful in higher iteration numbers.

The successful performance of MVO and WOA in the populations and iterations where the best fitness value is reached is remarkable. The results obtained with the other two algorithms (SSA and CS) applied for the first time to the solution of the problem are also very close to the best fitness value. This reveals the competitive aspects of algorithms that have not been applied to the problem earlier. The results of objective functions obtained with PSO [10] and GWO [2], which were earlier applied to the solution of the problem, are better than that of the studies in the literature.

When the performances of algorithms in different population sizes and iterations are analysed, it is seen that, as the number of iterations increased, the algorithms reached the solutions more easily. The increase in the number of iterations increased the precision of the solution.

Contribution of the increase in population size into quality solutions was not as much as that of the number of iterations. The outcomes of ABC and CS with a population size of 800 support this remark. Another remarkable outcome of the study is the speed of the algorithms. The best solution was reached in 19.51 seconds and the worst solution in 25.19 seconds with a population size of 800 and 5000 iteration numbers where the best fitness value was reached.

Quality solutions produced by swarm optimisers used for the first time in the study demonstrate that these methods can be used in delicate engineering design problems, such as the problem discussed in this article. However, the obligation for going up to higher numbers of iterations to reach the ideal solution is a significant issue here. In the upcoming studies, the focus will be on solving benchmark problems with hybrid models, such as the problem discussed in this study.

Data Availability

The data used to support the findings of this study are included within the article, and obtained other datasets can be requested from the corresponding author if needed.

Conflicts of Interest

The author declares that there are no conflicts of interest.

References

- [1] A. Kentli and M. Sahbaz, "Optimisation of hydrostatic thrust bearing using sequential quadratic programming," *Oxidation Communications*, vol. 37, no. 4, pp. 1144–1152, 2014.
- [2] İ. Şahin, M. Dörterler, and H. Gökçe, "Optimization of hydrostatic thrust bearing using enhanced grey wolf optimizer," *Mechanics*, vol. 25, no. 6, pp. 480–486, 2019.
- [3] W. B. Rowe, *Hydrostatic, Aerostatic and Hybrid Bearing Design*, Elsevier, Amsterdam, Netherlands, 2012.
- [4] O. J. Bakker and R. A. J. Van Ostayen, "Recess depth optimization for rotating, annular, and circular recess hydrostatic thrust bearings," *Journal of Tribology*, vol. 132, no. 1, Article ID 011103, 2010.
- [5] Z. Liu, Y. Wang, L. Cai, Y. Zhao, Q. Cheng, and X. Dong, "A review of hydrostatic bearing system: researches and applications," *Advances in Mechanical Engineering*, vol. 9, no. 10, Article ID 168781401773053, 2017.
- [6] W. B. Rowe, *Hydrostatic and hybrid bearing design*, Butterworth & Co. (Publishers) Ltd., Oxford, UK, 1983.
- [7] E. Solmaz, F. C. Babalik, and F. Öztürk, "Multicriteria optimization approach for hydrostatic bearing design," *Industrial Lubrication and Tribology*, vol. 54, no. 1, pp. 20–25, 2002.
- [8] C. A. C. Coello, "Treating constraints as objectives for single-objective evolutionary optimization," *Engineering Optimization*, vol. 32, no. 3, pp. 275–308, 2000.
- [9] K. Deb and M. Goyal, "Optimizing engineering designs using a combined genetic search," in *Proceedings of the Sixth International Conference on Genetic Algorithms*, pp. 521–528, San Francisco, CA, USA, July 1997.
- [10] S. He, E. Prempan, and Q. H. Wu, "An improved particle swarm optimizer for mechanical design optimization problems," *Engineering Optimization*, vol. 36, no. 5, pp. 585–605, 2004.
- [11] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems," *Computer-Aided Design*, vol. 43, no. 3, pp. 303–315, 2011.
- [12] J. N. Siddall, *Optimal Engineering Design*, Marcel Dekker, New York, NY, USA, 1982.
- [13] S. Talatahari and M. Azizi, "Optimization of constrained mathematical and engineering design problems using chaos game optimization," *Computers & Industrial Engineering*, vol. 145, Article ID 106560, 2020.
- [14] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, "Salp Swarm algorithm: a bio-inspired optimizer for engineering design problems," *Advances in Engineering Software*, vol. 114, pp. 163–191, 2017.
- [15] F. W. Hoffer: "Automatic fluid pressure balancing system," U.S. Patent and Trademark Office, Washington, DC, USA, U.S. Patent No. 2,449,297, 1948.
- [16] F. G. Arneson: "Externally pressurized bearing structure," U.S. Patent and Trademark Office, Washington, DC, US, U.S. Patent No. 3,472,565, 1969.
- [17] H. E. Arneson: "Hydrostatic bearing structure," U.S. Patent and Trademark Office, Washington, DC, USA, U.S. Patent No. 3,305,282, 1967.
- [18] N. R. Kane and A. H. Slocum: "Modular hydrostatic bearing with carriage form-fit to PR," U.S. Patent and Trademark Office, Washington, DC, USA, U.S. Patent No. 5,971,614, 1999.
- [19] A. H. Slocum, U.S. Patent and Trademark Office, Washington, DC, USA, U.S. Patent No. 5,104,237, 1992.
- [20] A. H. Slocum, P. A. Scagnetti, N. R. Kane, and C. Brunner, "Design of self-compensated, water-hydrostatic bearings," *Precision Engineering*, vol. 17, no. 3, pp. 173–185, 1995.
- [21] N. Tully, "Static and dynamic performance of an infinite stiffness hydrostatic thrust bearing," *Journal of Lubrication Technology*, vol. 99, no. 1, pp. 106–112, 1977.
- [22] T. A. Osman, M. Dorid, Z. S. Safar, and M. O. A. Mokhtar, "Experimental assessment of hydrostatic thrust bearing performance," *Tribology International*, vol. 29, no. 3, pp. 233–239, 1996.
- [23] H. Shen, J. Zhou, Q. Peng, H. Li, and J. Li, "Multi-objective interplanetary trajectory optimization combining low-thrust propulsion and gravity-assist maneuvers," *Science China Technological Sciences*, vol. 55, no. 3, pp. 841–847, 2012.
- [24] M. Fesanghary and M. M. Khonsari, "Topological and shape optimization of thrust bearings for enhanced load-carrying capacity," *Tribology International*, vol. 53, pp. 12–21, 2012.
- [25] W. Wang, Y. He, J. Zhao, Y. Li, and J. Luo, "Numerical optimization of the groove texture bottom profile for thrust bearings," *Tribology International*, vol. 109, pp. 69–77, 2017.
- [26] C. Weißbacher, C. Schellnegger, A. John, T. Buchgraber, and W. Pscheidt, "Optimization of journal bearing profiles with respect to stiffness and load-carrying capacity," *Journal of Tribology*, vol. 136, no. 3, Article ID 031709, 2014.
- [27] S. K. Yadav and S. C. Sharma, "Performance of hydrostatic tilted thrust pad bearings of various recess shapes operating with non-Newtonian lubricant," *Finite Elements in Analysis and Design*, vol. 87, pp. 43–55, 2014.
- [28] A. E. Hegazy, M. A. Makhlof, and G. S. El-Tawel, "Improved salp swarm algorithm for feature selection," *Journal of King Saud University - Computer and Information Sciences*, vol. 32, no. 3, pp. 335–344, 2020.
- [29] X. S. Yang and S. Deb, "Engineering optimisation by cuckoo search," *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 1, no. 4, pp. 330–343, 2010.

- [30] M. Mareli and B. Twala, "An adaptive Cuckoo search algorithm for optimisation," *Applied Computing and Informatics*, vol. 14, no. 2, pp. 107–115, 2018.
- [31] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-verse optimizer: a nature-inspired algorithm for global optimization," *Neural Computing and Applications*, vol. 27, no. 2, pp. 495–513, 2016.
- [32] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," vol. 200 Technical report-TR06, pp. 1–10, Erciyes University, Engineering Faculty, Computer Engineering Department, Kayseri, Turkey, 2005.
- [33] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.
- [34] Ü. Atila, M. Dörterler, R. Durgut, and İ. Şahin, "A comprehensive investigation into the performance of optimization methods in spur gear design," *Engineering Optimization*, vol. 52, no. 6, pp. 1052–1067, 2020.
- [35] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, vol. IV, pp. 1942–1948, Perth, Australia, November 1995.
- [36] H. Shi, S. Liu, H. Wu et al., "Oscillatory particle swarm optimizer," *Applied Soft Computing*, vol. 73, pp. 316–327, 2018.
- [37] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.
- [38] M. Dörterler, İ. Şahin, and H. Gökçe, "A grey wolf optimizer approach for optimal weight design problem of the spur gear," *Engineering Optimization*, vol. 51, no. 6, pp. 1013–1027, 2019.