Sigma J Eng Nat Sci, Vol. 40, No. 3, pp. 529–541, September, 2022

Sigma Journal of Engineering and Natural Sciences
Web page info: https://sigma.yildiz.edu.tr
DOI: 10.14744/sigma.2022.00056

Research Article

A novel modified arithmetic optimization algorithm for power system stabilizer design

Davut IZCI[1,*]

[1]*Department of Electronics and Automation, Batman University, Batman, Türkiye*

ARTICLE INFO

Article history
Received: 14 January 2021
Accepted: 07 June 2021

Keywords:
Power System Stability;
Metaheuristic Algorithms;
Hybrid Algorithm

ABSTRACT

The development of a novel hybrid algorithm by modifying the arithmetic optimization algorithm (AOA) with the aid of simulated annealing technique is discussed in this paper. The novel algorithm, named modified arithmetic optimization algorithm (mAOA), is proposed as an effective tool for optimizing power system stabilizer (PSS) adopted in a single-machine infinite-bus power system. To perform the assessments, MATLAB/Simulink software was used. The evaluations on the proposed algorithm are initially performed using several benchmark functions that have unimodal and multimodal natures. The results are then compared with five of the other competitive approaches (arithmetic optimization algorithm, simulated annealing algorithm, genetic algorithm, particle swarm optimization and gravitational search algorithm). The comparisons with respect to those algorithms demonstrate the great promise of the constructed hybrid mAOA algorithm. This shows the greater balance between global and local search stages achieved by the mAOA algorithm. The performance of the developed mAOA algorithm is also assessed through designing an optimally performing PSS for further evaluation which allows the observation of its capability for complex real-world engineering problems. To do so, PSS damping controller is formulated as an optimization problem and the constructed mAOA algorithm is used to search for optimal controller parameters to demonstrate the applicability and the greater performance of the proposed hybrid algorithm for such a complex real-world engineering problem. The obtained results for the latter case are compared with the sine-cosine and symbiotic organisms search algorithms as they are the best performing reported algorithms. The comparisons have demonstrated the superiority of the mAOA algorithm over reported best performing algorithms in terms of PSS design, as well.

Cite this article as: Izci D. A novel modified arithmetic optimization algorithm for power system stabilizer design. Sigma J Eng Nat Sci 2022;40(3):529–541.

***Corresponding author.**
*E-mail address: davut.izci@batman.edu.tr

This paper was recommended for publication in revised form by Regional Editor N. Özlem Ünverdi

Published by Yıldız Technical University Press, İstanbul, Turkey

INTRODUCTION

One of the main challenges in power systems is to deal with the low frequency oscillations as they may cause the failure in terms of integrity of power system interconnections if they are not damped appropriately. Such a failure is highly undesired as it causes interrupts in terms of power supply which may lead to financial loss. Due to its high impact on the stability of the system, it is quite vital to damp out the low frequency oscillations so that the transfer capability of the power system can be preserved. A power system stabilizer (PSS) is one of the mainly employed structures to deal with such oscillations for both single-machine infinite-bus (SMIB) and multi-machine power systems. However, such structures are required to be designed appropriately by considering the nature of the power system.

In terms of designing conventional PSS, a linearized model is considered which means a nominal operating point is taken into account to determine the parameters of the PSS. However, due to nonlinear nature of power systems, consistent fluctuations over a wide range may occur which makes the conventional PSS insufficient to achieve an optimum performance. Several approaches such as pole shifting and pole placement along with self-tuning regulators and feedback linearization are available for PSS design. However, longer computational times are the lack of those approaches [1]. To address the latter issue and to come up with a more efficient solution, metaheuristic algorithms have widely been utilized recently as an alternative method [2].

There has been a breathtaking effort in terms of development of new metaheuristic algorithms over the last decade for tackling with difficult and complex nonlinear optimization problems. Traditionally, deterministic optimization schemes are used for solving optimization problems, however, those schemes suffer from unbalanced exploitation-exploration, local optima stagnation and being derivative dependent [3]. Because of those major issues, a growing interest has shifted towards stochastic optimization approaches [4]. The metaheuristic algorithms are stochastic optimization techniques that are simple, durable, self-organized and skillful approaches [5]. Few of the recent examples of metaheuristic algorithms are moth-flame optimization algorithm [6], elephant herding optimization [7], whale optimization algorithm [8], grasshopper optimization algorithm [9], Henry gas solubility optimization [10] and Harris hawks optimization algorithm [11]. Meanwhile, all the listed algorithms would exhibit an equivalent performance on average in case they were used for all potential optimization problems [12] which means some of them would present greater performance for specific problems.

The promise of metaheuristic algorithms has already been demonstrated in terms of offline tuning of PSS parameters by considering a wide range of operating conditions. Some of the recently reported metaheuristic algorithms based PSS design examples can be listed as salp swarm algorithm [13], artificial bee colony algorithm [14], chaotic versions of sunflower optimization [15] and particle swarm optimization [16], grasshopper optimization [17], particle swarm optimization [18], kidney-inspired algorithm [19], farmland fertility algorithm [20], sine-cosine algorithm [1] along with its modified version with grey wolf optimization [21] and improved whale optimization [22]. The demonstrated promise of those algorithms has motivated this study to further improve the PSS ability by utilizing a more recent metaheuristic algorithm as a competitive and efficient approach in terms of designing the related parameters. Therefore, arithmetic optimization algorithm (AOA) [23] has been used for this study which is one of the latest published and population-based metaheuristic algorithms.

The greater promise of the arithmetic optimization algorithm has so far been demonstrated through several test functions and engineering design problems. It is feasible to achieve a good exploration in arithmetic optimization algorithm. Considering this fact, the overall ability of this algorithm can further be improved via integrating another algorithm with better exploitative capability which would consequently lead to a better balance between exploration and exploitation phases. In this way, the algorithm can avoid from local optima and the early convergence issue can be prevented. One of the popular algorithms that can be used to achieve such a structure is the simulated annealing (SA) technique [24]. The latter algorithm has an excellent local search ability that can easily be hybridized with another metaheuristic approach. Besides its easy implementable structure, it also does not require longer computational times which makes it an excellent candidate for hybridization. The latter case is another, and fundamental, motivation of this paper, therefore, simulated annealing technique has been integrated with the arithmetic optimization algorithm in order to construct a better structure.

Bearing the discussion so far in mind, this paper proposes a novel hybrid algorithm, named modified arithmetic optimization algorithm (mAOA), which has been developed by considering greater local search behavior of simulated annealing algorithm in order to enhance the capability of the arithmetic optimization algorithm. The constructed hybrid algorithm aims to address the issue related to lack of balance between exploration and exploitation phases of the arithmetic optimization algorithm. The simulated annealing technique is used to operate on worse solutions such that the potential of neighborhood solutions is not neglected.

To evaluate the developed mAOA algorithm, MATLAB/Simulink environment was used. The performance evaluation was firstly carried out against benchmark functions of Step, Sphere, Rastrigin, Rosenbrock, Quartic, Griewank, Schwefel, and Ackley [25]. The obtained results on those well-known benchmark functions were comparatively assessed with respect to five other competitive approaches such as original arithmetic optimization algorithm, simulated annealing algorithm, genetic algorithm, particle swarm optimization and gravitational search algorithm. The achieved results have demonstrated highly competitive behavior of the proposed hybrid mAOA algorithm which is an indication of better balance between exploration and exploitation. After comparative assessment of the proposed mAOA algorithm on those test functions, the design of a power system stabilizer (PSS), which is employed in a single-machine infinite-bus system, was considered as a complex real-world engineering problem for further performance evaluation. The ability of the designed system was observed through comparing it with the PSS controllers that are designed by utilizing sine cosine and symbiotic organisms search algorithms [1] as the latter two structures had the same power system and the limits of the PSS parameters. The comparisons have also demonstrated better performance of the mAOA algorithm for such a complex real-world engineering problem, as well. In brief, the assessments have shown the mAOA algorithm to be a useful and efficient optimization algorithm for benchmark functions and PSS controller design for power system. The contribution of this work can be listed briefly as follows: (1) A novel hybrid structure, named mAOA, is achieved by modifying arithmetic optimization algorithm with the aid of simulated annealing technique. The constructed hybrid mAOA algorithm has an enhanced balance between exploration and exploitation phases; (2) The proposed mAOA algorithm is utilized to achieve optimum parameters for stabilizing power system; (3) The obtained statistical results from unimodal and multimodal benchmark functions demonstrate the greater capability of the mAOA algorithm in terms of achieving the best, mean and standard deviation values compared to simulated annealing algorithm, genetic algorithm, particle swarm optimization, gravitational search algorithm and original arithmetic optimization algorithm; (4) The mAOA algorithm is shown to be a good choice for complex optimization problems as it enhances the transient stability of the SMIB system greatly as well as providing improvement on the damping characteristics of electromechanical modes; (5) Overall, the proposed mAOA algorithm is demonstrated to be a powerful approach for optimizing problems with different nature compared to other reported competitive and best performing algorithms.

ARITHMETIC OPTIMIZATION ALGORITHM

The arithmetic optimization algorithm is one of the latest reported population based stochastic algorithms which is inspired from a fundamental component of number theory known as the arithmetic [23]. It was proposed to solve optimization problems in a derivative-free manner. In this algorithm, the simple mathematical operators of addition, subtraction, multiplication, and division are used for determining the best solution from candidate solutions. In the initialization stage of this algorithm, a set of random solutions of $\boldsymbol{X}$ are generated as given in (1):

$$X = \begin{bmatrix} x_{1,1} & \cdots & \cdots & x_{1,j} & x_{1,n-1} & x_{1,n} \\ x_{2,1} & \cdots & \cdots & x_{2,j} & \cdots & x_{2,n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{N-1,1} & \cdots & \cdots & x_{N-1,j} & \cdots & x_{N-1,n} \\ x_{N,1} & \cdots & \cdots & x_{N,j} & x_{N,n-1} & x_{N,n} \end{bmatrix} \tag{1}$$

In each iteration, the best candidate solution is the best obtained solution so far. After the initialization step, the search phases of exploration or exploitation are selected based on a function called Math Optimizer Accelerated (***MOA***) function which is given in (2):

$$MOA(t_c) = Min + t_c \times \left(\frac{Max - Min}{t_M} \right) \tag{2}$$

where the value of the function at current iteration is denoted by $\boldsymbol{MOA(t_c)}$. The current iteration is represented by $\boldsymbol{t_c}$ and has a range between 1 and maximum number of iterations ($\boldsymbol{t_M}$). The minimum and maximum values of the accelerated function are denoted by ***Min*** and ***Max***, respectively. The exploration phase is conditioned by ***MOA*** function for $\boldsymbol{r_1} >$ ***MOA*** where $\boldsymbol{r_1}$ is a random number. In terms of exploration, the arithmetic operators of multiplication (***M***) or division (***D***) are used for random search on several regions. Therefore, two different strategies, known as multiplication and division search strategies, are exist in this algorithm which are mathematically modeled as follows:

$$x_{i,j}(t_c+1) = \begin{cases} best(x_j) \times MOP \times \big((UB_j - LB_j) \times \\ \qquad \mu + LB_j\big), \ for \ r_2 > 0.5 \\ best(x_j) \div (MOP + \in) \times \big((UB_j - LB_j) \times \\ \qquad \mu + LB_j\big), \ for \ r_2 < 0.5 \end{cases} \tag{3}$$

where the solution of $\boldsymbol{i}$ in the next iteration is represented by $\boldsymbol{x_i(t_c+1)}$, the $\boldsymbol{j^{th}}$ position of solution $\boldsymbol{i}$ for current iteration is denoted by $\boldsymbol{x_{i,j}(t_c)}$ and the $\boldsymbol{j^{th}}$ position of the best solution obtained so far is given by $\boldsymbol{best(x_j)}$. In the respective equation, $\in$ denotes a small integer number whereas $\boldsymbol{\mu}$ represents the control parameter for search process adjustment. The upper and lower bounds of position j are represented by $\boldsymbol{UB_j}$ and $\boldsymbol{LB_j}$, respectively. The ***MOP*** function given in the latter equation is used to represent Math Optimizer probability which is calculated as in (4):

$$MOP(t_c) = 1 - \frac{(t_c)^{1/\alpha}}{(t_M)^{1/\alpha}} \tag{4}$$

In Eq, (4), $\boldsymbol{MOP}(\boldsymbol{t_c})$ is used to denote the value of the function for current iteration whereas $\boldsymbol{\alpha}$ is a sensitive parameter which defines the accuracy of the search through iterations. The execution of $\boldsymbol{M}$ or $\boldsymbol{D}$, given in Eq. (3), are decided based on another random number which is denoted by $\boldsymbol{r_2}$. $\boldsymbol{M}$ operator is performing the task, as can be seen from Eq. (3), for $\boldsymbol{r_2} > 0.5$. In this stage, the $\boldsymbol{D}$ operator is neglected until the $\boldsymbol{M}$ operator completes the task. In case of $\boldsymbol{r_2} < 0.5$, the execution of the task occurs vice versa (see Eq. (3)). In this way, the position update is performed for exploration phase. The mathematical operators of addition ($\boldsymbol{A}$) and subtraction ($\boldsymbol{S}$) are used for exploitation. As is the case for exploration, the $\boldsymbol{MOA}$ function is also used for conditioning this phase. As opposed to the exploration, this phase is performed for $\boldsymbol{r_1} < \boldsymbol{MOA}$. The exploitation phase of $\boldsymbol{AOA}$ is modeled using Eq. (5) where $\boldsymbol{r_3}$ is a random number.

$$x_{i,j}(t_c+1) = \begin{cases} best(x_j) + MOP \times \big((UB_j - LB_j) \times \\ \qquad \mu + LB_j\big),\ for\ r_3 > 0.5 \\ best(x_j) - MOP \times \big((UB_j - LB_j) \times \\ \qquad \mu + LB_j\big),\ for\ r_3 < 0.5 \end{cases} \tag{5}$$

The execution of $\boldsymbol{A}$ or $\boldsymbol{S}$ are decided based on the value of $\boldsymbol{r_3}$. The A operator is executed for $\boldsymbol{r_3} > 0.5$ and $\boldsymbol{S}$ operator is neglected until $\boldsymbol{A}$ operator finishes the task whereas for $\boldsymbol{r_3} < 0.5$ vice versa occurs.

SIMULATED ANNEALING ALGORITHM

The SA algorithm is basically a mathematical representation of the metallurgical annealing process [24]. The stated annealing process helps formation of uniform crystals because of adopted heating and cooling stages. A random solution, $\boldsymbol{X_i}$, is the starting point of SA technique. This solution is then used to determine a neighborhood solution $\boldsymbol{X_i'}$. The fitness values of both random solution and its neighborhood solution are then computed and compared. In case of $\boldsymbol{F}(\boldsymbol{X_i'}) < \boldsymbol{F}(\boldsymbol{X_i'})$, SA sets $\boldsymbol{X_i} = \boldsymbol{X_i'}$. Apart from the latter arrangement, SA may still decide to use the neighborhood solution even if it does not satisfy the above condition. Such a case depends on the probability of $\boldsymbol{p}$ which is defined as follows.

$$p = e^{-\frac{\Delta F}{T_K}}; \Delta F = F(X_i') - F(X_i) \tag{6}$$

In here, $\boldsymbol{F}$ is the control parameter for the fitness whereas $\boldsymbol{T}$ is of the temperature. The SA does not perform the replacement of $\boldsymbol{X_i}$ by $\boldsymbol{X_i'}$ in case where $\boldsymbol{p}$ is smaller than a randomly generated number within [0, 1] range. On the other hand, the replacement would occur for a contrary case. The following equation is used by SA to reduce the temperature valueswhere the cooling coefficient (which has a random constant value in [0, 1]) is denoted by $\boldsymbol{\mu}$.

$$T_{k+1} = \mu T_k \tag{7}$$

PROPOSED ALGORITHM

As discussed in the previous related sections, the original version of arithmetic optimization algorithm has been demonstrated to be successful for various test functions and a bunch of engineering design problems. Further improvement is feasible for this algorithm to make it an even more efficient tool to deal with problems of different natures. To achieve such an enhancement, simulated annealing algorithm has been used as a complementary structure. Therefore, the developed mAOA algorithm is basically a hybridized structure that consists of the arithmetic optimization and simulated annealing algorithms.

In the developed mAOA algorithm, the simulated annealing technique deals with the exploitation enhancement as it is a good local search algorithm [24]. The latter technique initializes random solutions as a necessary starting point in the search space. The solution of the problem is then evaluated by generating and accepting/rejecting the neighbor solutions which shifts the solution towards the better neighbor's solution to explore the search space. This behavior helps the arithmetic optimization algorithm to avoid local minimum. On the other hand, the original arithmetic optimization algorithm takes care of the global search. The flowchart given in Figure 1 illustrates the process of the developed mAOA algorithm. As shown in the respective flowchart, the algorithm is initialized by the mAOA parameters and the candidate solutions. Then the arithmetic optimization algorithm is performed to find a global best solution. The latter obtained solution is used as an initial solution for the simulated annealing algorithm which is followed by the procedure of the latter technique. In each generation, the best solution of the arithmetic optimization algorithm for the respective iteration is adopted as the starting solution of simulated annealing algorithm. In this way, the ability of the arithmetic optimization algorithm is enhanced in terms of achieving better solutions.

SIMULATION SETUP AND RESULTS

Benchmark Functions

In this study, four unimodal and four multimodal benchmark functions have been employed for initial performance evaluation of the constructed hybrid mAOA algorithm. The details related to those benchmark functions (name, mathematical definition, dimension (n), range and minimum point (F_{min})) are listed in Table 1.

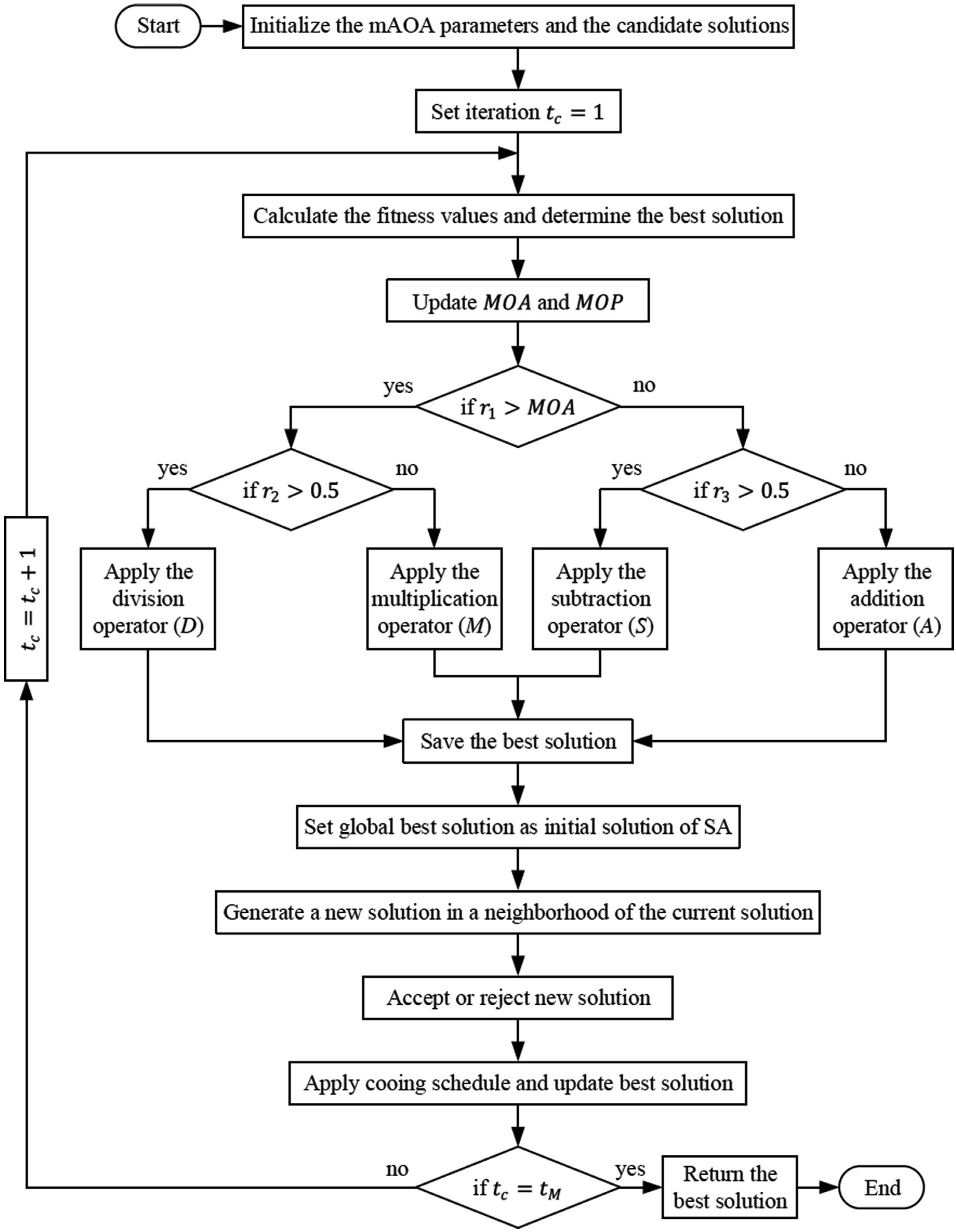

Figure 1. Flowchart of the constructed mAOA algorithm.

Table 1. Adopted benchmark functions for performance evaluation

Name	Function equation	n	Range	F_{min}
Sphere	$F_1(x)=\sum_{i=1}^{n} x_i^2$	30	$[-100,100]^n$	0
Rosenbrock	$F_2(x)=\sum_{i=1}^{n-1}(100(x_{i+1}-x_i^2)^2+(x_i-1)^2)$	30	$[-30,30]^n$	0
Step	$F_3(x)=\sum_{i=1}^{n}(x_i+0.5)^2$	30	$[-100,100]^n$	0
Quartic	$F_4(x)=\sum_{i=1}^{n} ix_i^4+random[0,1)$	30	$[-1.28,1.28]^n$	0
Schwefel	$F_5(x)=-\sum_{i=1}^{n}\left(x_i \sin\left(\sqrt{\lvert x_i \rvert}\right)\right)$	30	$[-500,500]^n$	$-418.9829 \times n$
Rastrigin	$F_6(x)=\sum_{i=1}^{n}[x_i^2-10\cos(2\pi x_i)+10]$	30	$[-5.12,5.12]^n$	0
Ackley	$F_7(x)=-20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}\right)-\exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right)+20+e$	30	$[-32,32]^n$	0
Griewank	$F_8(x)=\frac{1}{4000}\sum_{i=1}^{n}x_i^2-\prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right)+1$	30	$[-600,600]^n$	0

As it is obvious from the table, effective performance evaluation of the developed mAOA algorithm can be performed via listed test functions since the ones from $\boldsymbol{F_1(x)}$ to $\boldsymbol{F_4(x)}$ (Sphere, Rosenbrock, Step, Quartic) are unimodal functions and good tools for exploitation assessment whereas $\boldsymbol{F_5(x)}$ to $\boldsymbol{F_8(x)}$ (Schwefel, Rastrigin, Ackley, Griewank) are multimodal functions and good for evaluating the exploration ability of the algorithms. A detailed description of the employed test functions can be found in Ref. [25].

Table 2. Initial parameter values of compared algorithms

Algorithm	Parameter	Value
SA [24]	Initial temperature	0.10
	Cooling factor	0.98
	Mutation rate	0.50
GA [26]	Selection	Roulette wheel
	Crossover rate	0.80
	Mutation rate	0.40
PSO [27]	Cognitive coefficient	2
	Social coefficient	2
	Inertia constant	Linearly decrease from 0.80 to 0.20
GSA [25]	Gravitational constant	100
	Decreasing coefficient	20
AOA [23]	Sensitive parameter	5
	Control parameter	0.499

Compared Algorithms

The performance of the developed hybrid mAOA algorithm has comparatively been assessed against the previously mentioned benchmark functions using both popular (genetic algorithm, simulated annealing algorithm and particle swarm optimization) and recent (arithmetic optimization algorithm and gravitational search algorithm) competitive algorithms.The initial parameters chosen for those algorithms are listed in Table 2.

Results and Discussion

The population size and the maximum number of iterations for the developed mAOA algorthm were respectively chosen to be 50 and 1000 in order to perform a fair comparison with particle swarm optimization (PSO), genetic algorithm (GA), simulated annealing (SA) algorithm, gravitational search algorithm (GSA) and arithmetic optimization algorithm (AOA). The obtained statistical results (mean, standard deviation (StDev), best and rank) from the employed benchmark functions are listed in Table 3. Compared with other algorithms, the overall best results can easily be seen to be achieved via the developed mAOA algorithm. The respective results are a good demonstration of better balance between the diversification and the intensification phases of the mAOA algorithm.

PSS DESIGN FOR SMIB SYSTEM

Power System Model

Figure 2 shows the power system considered for this study. The respective power system consists of a single-machine infinite-bus through a double circuit transmission line. The related machine was assumed to be equipped with a fast exciter. To improve the small oscillations, a PSS was integrated with this system. The 4th order model of the considered power system can be given as follows [28]:

$$\dot{\delta} = \omega_0(\omega - 1) \tag{8}$$

$$\dot{\omega} = [P_m - P_e - D(\omega - 1)]/M \tag{9}$$

$$\dot{E}_q' = [E_{fd} - E_q' - (x_d - x_d')i_d]/T_{d0}' \tag{10}$$

Table 3. The obtained comparative results for unimodal and multimodal functions

Functions	Metric	SA	GA	PSO	GSA	AOA	mAOA (proposed)
$F_1(x)$	Mean	1.97E−13	1.05E−02	1.43E−04	2.06E−17	2.09E−83	**0**
	StDev	6.47E−14	5.12E−03	1.12E−04	6.57E−18	6.62E−83	0
	Best	6.88E−14	4.23E−03	9.38E−06	9.83E−18	0	0
	Rank	4	6	5	3	2	**1**
$F_2(x)$	Mean	1.07E+03	9.67E+01	1.33E+02	2.83E+01	2.77E+01	**4.90E+00**
	StDev	2.04E+03	1.28E+02	1.26E+02	1.14E+01	6.86E−01	2.06E+00
	Best	2.45E+01	9.52E+00	2.49E+01	2.61E+01	2.61E+01	3.18E+00
	Rank	6	4	5	3	2	**1**
$F_3(x)$	Mean	5.72E−01	**0**	1.29E−01	**0**	2.37E+00	**0**
	StDev	7.53E−01	0	3.38E−01	0	3.18E−01	0
	Best	0	0	0	0	1.96E+00	0
	Rank	5	**1**	4	**1**	**6**	**1**
$F_4(x)$	Mean	1.22E−01	5.12E−02	6.58E−02	2.11E−02	1.31E−05	**1.24E−05**
	StDev	3.85E−02	2.31E−02	1.92E−02	8.27E−03	1.95E−05	1.16E−05
	Best	5.74E−02	1.37E−02	2.94E−02	7.36E−03	1.47E−06	1.10E−06
	Rank	6	4	5	3	2	**1**
$F_5(x)$	Mean	−9.34E+03	−6.84E+03	−5.23E+03	−2.71E+03	−6.26E+03	**−9.76E+03**
	StDev	3.78E+02	6.29E+02	5.04E+02	3.45E+02	5.26E+02	4.58E+02
	Best	−9.97E+03	−8.16E+03	−6.60E+03	−3.52E+03	−7.11E+03	−1.09E+04
	Rank	2	3	5	6	4	**1**
$F_6(x)$	Mean	5.42E+01	1.26E+01	2.94E+01	1.51E+01	**0**	**0**
	StDev	1.35E+01	3.88E+00	7.09E+00	4.42E+00	0	0
	Best	2.66E+01	4.57E+00	1.76E+01	7.94E+00	0	0
	Rank	6	3	5	4	**1**	**1**
$F_7(x)$	Mean	3.46E−01	2.17E−02	7.39E−03	3.72E−09	**8.88E−16**	**8.88E−16**
	StDev	4.55E−01	4.38E−03	2.54E−03	3.85E−10	0	0
	Best	1.68E−07	1.23E−02	6.75E−04	2.98E−09	8.88E−16	8.88E−16
	Rank	6	5	4	3	1	**1**
$F_8(x)$	Mean	1.28E−02	1.82E−02	2.29E−02	4.46E+00	7.81E−02	**6.64E−13**
	StDev	8.25E−03	1.10E−02	2.82E−02	2.13E+00	3.95E−02	1.60E−12
	Best	4.53E−06	5.91E−03	7.35E−05	1.92E+00	9.79E−04	4.66E−15
	Rank	2	3	4	6	5	**1**
Average rank		4.6250	3.6250	4.6250	3.6250	2.8750	**1**
Overall rank		5	3	5	3	2	**1**

$$\dot{E}_{fd} = [K_A(V_{ref} - V_T + U_{PSS}) - E_{fd}]/T_A \quad (11)$$

$$P_e = E'_q i_q + (x_q - x'_d) i_d i_q \quad (12)$$

where ω_0, δ and ω are the synchronous speed, rotor angle and rotor speed, respectively. P_m represent mechanical power input whereas P_e is electrical power output. The inertia constant and damping coefficient are denoted by M and D, respectively. E'_q and E_{fd} are internal voltage behind x'_d and excitation system voltage, respectively, whereas T'_{d0} is the d-axis open-circuit transient time constant. K_A denotes the constant of gain whilst T_A is of the time for the excitation circuit. i_d is the stator current in d-axis whereas i_q is of the q-axis circuits. x_d and x_q represent d-axis reactance and q-axis synchronous reactance, respectively, and x'_d denotes the d-axis transient reactance. V_T, V_{ref} and U_{PSS} represent the terminal voltage, reference voltage and PSS output signal, respectively.

The linearized model of the considered power system is obtained as follows where K_1 to K_6 are the well-established constants showing the interaction amongst the variables in power system [28].

$$\Delta\dot{\delta} = \omega_0 \Delta\omega \quad (13)$$

$$\Delta\dot{\omega} = \frac{K_1}{M}\Delta\delta - \frac{D}{M}\Delta\omega - \frac{K_2}{M}\Delta E'_q \quad (14)$$

$$\Delta\dot{E}'_q = \frac{K_4}{T'_{d0}}\Delta\delta - \frac{1}{K_3 T'_{d0}}\Delta\dot{E}'_q + \frac{1}{T'_{d0}}\Delta E_{fd} \quad (15)$$

$$\Delta\dot{E}'_{fd} = -\frac{K_A K_5}{T_A}\Delta\delta - \frac{K_A K_6}{T_A}\Delta\dot{E}'_q - \frac{1}{T_A}\Delta E_{fd} + \frac{K_A}{T_A}\Delta U_{PSS} \quad (16)$$

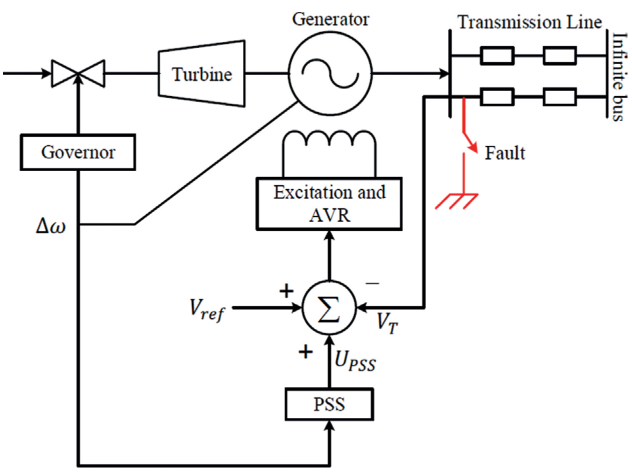

Figure 2. Single-machine infinite-bus system with PSS.

The following equations can be used to arrange the state space form of the system given in (13) - (16):

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (17)$$

where:

$$x(t) = [\Delta\delta \Delta\omega \Delta E'_q \Delta E_{fd}]^T \quad (18)$$

$$x(t) = \begin{bmatrix} 0 & 2\pi f & 0 & 0 \\ -\frac{K_1}{M} & -\frac{D}{M} & -\frac{K_2}{M} & 0 \\ -\frac{K_4}{T'_{d0}} & 0 & -\frac{1}{K_3 T'_{do}} & -\frac{1}{T'_{do}} \\ -\frac{K_A K_5}{T_A} & 0 & -\frac{K_A K_6}{T_A} & -\frac{1}{T_A} \end{bmatrix} \quad (19)$$

$$B = \begin{bmatrix} 0 & 0 & 0 & \frac{K_A}{T_A} \end{bmatrix} \quad (20)$$

The machine, transformer and transmission along with the exciter and PSS data of the analyzed single-machine infinite-bus system are listed in Table 4.

Structure of PSS

Operating an auxiliary stabilizing signal through the excitation system is the main function of a PSS as it is used to damp out the generator rotor oscillations [28]. The structure of a widely used conventional lead-lag PSS is given in (21).

$$U_{PSS} = K_{PSS}\left(\frac{sT_W}{1+sT_W}\right)\left(\frac{1+sT_1}{1+sT_2}\right)\left(\frac{1+sT_3}{1+sT_4}\right)\Delta\omega \quad (21)$$

This structure consists of a stabilizer gain K_{PSS}, a washout filter with a time constant T_W, two lead-lag blocks for phase compensation with time constants T_1, T_2, T_3 and T_4 and a limiter as shown in Figure 3. U_{PSS} is the output voltage of the PSS which is added to the generator exciter input. The generator speed deviation $\Delta\omega$ is typically used as the PSS input signal. In this study, washout time constant T_W was chosen to be **5***s*.

mAOA Algorithm based PSS Design

In this study, the performance index known as the integral of time multiplied absolute error (ITAE), given in (22), was used as an objective function since it is feasible to choose the parameters of the PSS for minimization of the respective function.

$$ITAE = \int_0^{t_{sim}} t \cdot |\Delta\omega(t)| \cdot dt \quad (22)$$

Table 4. Parameters of the test system [1]

Machine	$M = 2H = 6.4\ pu,\ x_d = 2.5\ pu,\ x_q = 2.1\ pu,\ x'_d = 0.39\ pu,\ T'_{d0} = 9.6\ s,$ $\omega_0 = 2\pi \times 60\ rad/s,\ P_{e0} = 0.5\ pu,\ D = 0,\ \delta_0 = 42.4°$
Transmission line and transformer	$X_{L1} = X_{L2} = 0.8\ pu,\ X_T = 0.1\ pu$
Exciter and PSS	$K_A = 400,\ T_A = 0.2\ s,\ E_{fd}^{min} = -5\ pu,\ E_{fd}^{max} = 5\ pu,\ U_{PSS}^{max} = -0.2\ pu,\ U_{PSS}^{max} = 0.2\ pu$

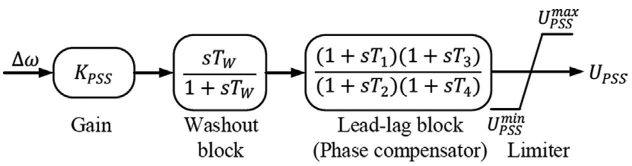

Figure 3. Block diagram of two-stage lead-lag PSS.

In the definition of ITAE, $\boldsymbol{\Delta\omega(t)}$ stands for the rotor speed deviation following a large disturbance whereas $\boldsymbol{t_{sim}}$ is the simulation time. The advantage of this performance index is the requirement of minimal dynamic plant information. The related optimization problem can be minimized with the criteria of $0.01 \le K_{PSS} \le 100$ and $0.01 \le T_i \le 1$ where $\boldsymbol{i = 1,2,3,4}$. The implementation of the mAOA algorithm to PSS design for single-machine infinite-bus system is illustrated in Figure 4.

The stability of the respective system is increased after performing the optimization procedure detailed in Figure 4. The population and the maximum iteration number (stopping criteria) were respectively set to 50 and 30 in the optimization procedure depicted in Figure. MATLAB/Simulink environment was used to calculate the ITAE objective function via integrating the mAOA algorithm with single-machine infinite-bus power system. The mAOA algorithm was run for 25 times and the corresponding minimum ITAE function value was found with the PSS parameters of $\boldsymbol{K_{PSS}} = 42.9745$, $\boldsymbol{T_1} = 0.08039$, $\boldsymbol{T_2} = 0.01023$, $\boldsymbol{T_3} = 0.07818$ and $\boldsymbol{T_4} = 0.01015$. The average execution time was found to be 51.39 *s* per run. Besides, Figure 5 has been plotted to demonstrate how the adopted ITAE objective function is minimized with respect to number of iterations.

Simulation Results

The competitiveness of the constructed mAOA algorithm in designing PSS was assessed by comparing it with the most recently published study through eigenvalue analysis and nonlinear time domain simulation. The Simulink model used for the simulations can be found in Ref [1]. The most convenient approaches chosen for comparisons were symbiotic organisms search algorithm based PSS (SOS-PSS) [1] and sine cosine algorithm based PSS (SCA-PSS) [1] damping controllers since those approaches adopted the same limits of the PSS and power system parameters. The optimized PSS parameters were obtained by utilizing the mAOA and the compared algorithms are presented in Table 5.

Eigenvalue Analysis

Eigenvalue analysis is used to investigate the small signal stability behavior of a power system by considering different characteristic frequencies. In a power system, the stability of eigenvalues (to be in the left side of the s–plane) is not the only criteria for stability. The desired eigenvalues must also be damped as quickly as possible for electromechanical oscillations. The eigenvalue analysis was performed, from this point of view, to verify that the constructed mAOA algorithm-based controller improves the linear model stability of the system.

The system eigenvalues ($\boldsymbol{\lambda = \sigma \pm j\omega}$) and damping ratios ($\boldsymbol{\xi}$) of the electromechanical modes related to the systems without using the stabilizer and with optimized PSS controller parameters using mAOA, SCA and SOS algorithms are given in Table 6.

As can be seen from the table, the system is insufficiently damped in case of no PSS ($\boldsymbol{\xi} = 0.0115$). In addition, compared to the SOS-PSS [1] and SCA-PSS [1] controllers, the electromechanical modes of the proposed mAOA-PSS are further to the left of the s-plane (damping factor $\sigma = -2.6040$) along with a greater damping ratio ($\boldsymbol{\xi} = 54.36\%$). Therefore, mAOA algorithm-based PSS greatly enhances the small signal stability of the single-machine infinite-bus power system and improves the damping characteristics of electromechanical modes.

Nonlinear Time Domain Simulation

At the generator terminal busbar, a three-phase fault was applied at $t = 2\ s$ and then cleared after 6-cycle (**0.10*s***). The original system has been restored after the clearance of the fault. The response of the system in terms of rotor angle ($\boldsymbol{\delta}$), speed deviation ($\boldsymbol{\Delta\omega}$), electrical power ($\boldsymbol{P_e}$) and terminal voltage ($\boldsymbol{V_T}$) are shown in Figures 6, 7, 8 and 9, respectively. It is obvious from these figures that the power system oscillations are inadequately damped although the system is stable without any controller. The stability of the single-machine infinite-bus system was maintained, and the oscillations of the power system were effectively suppressed with

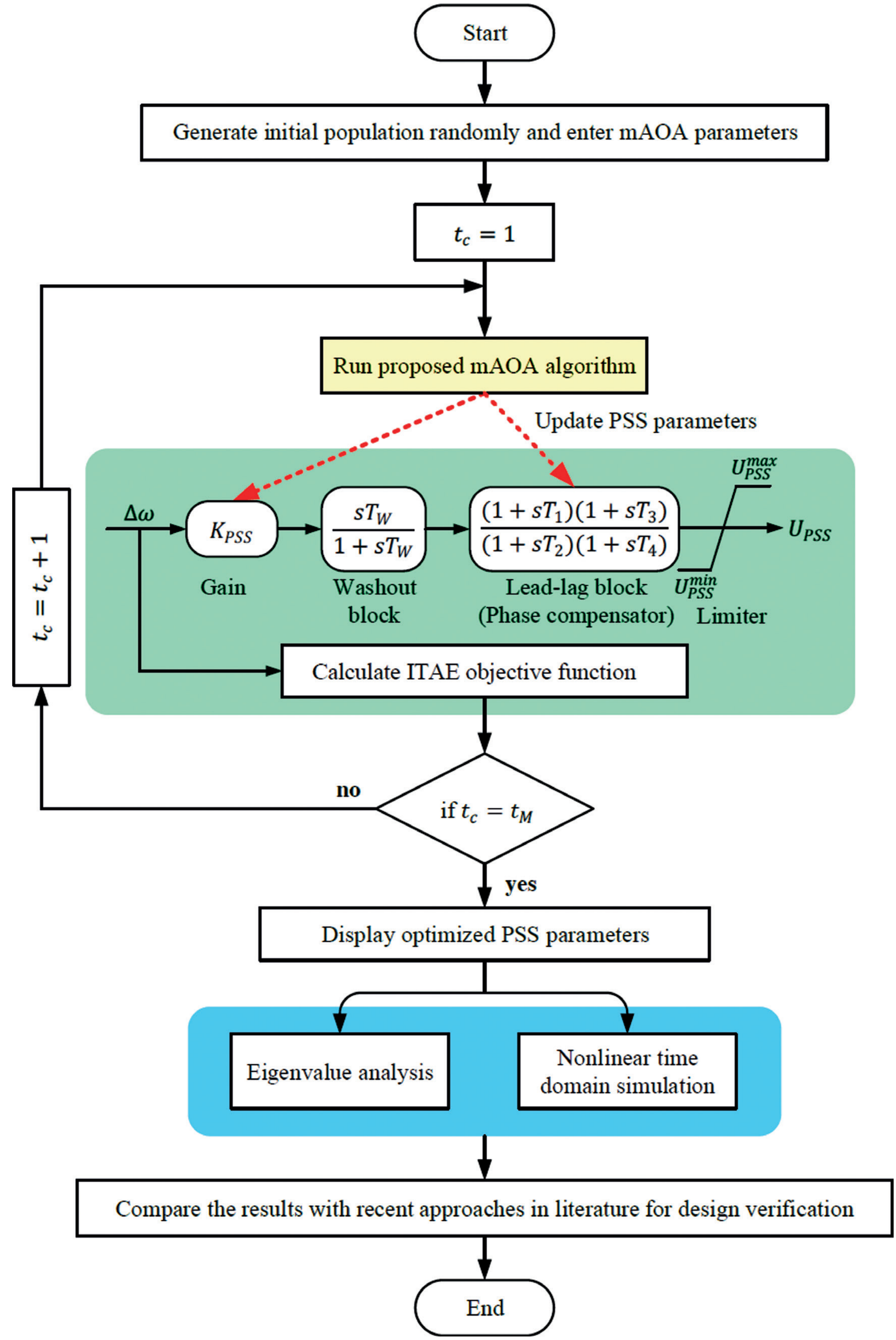

Figure 4. Detailed flowchart of the developed mAOA algorithm-based PSS design.

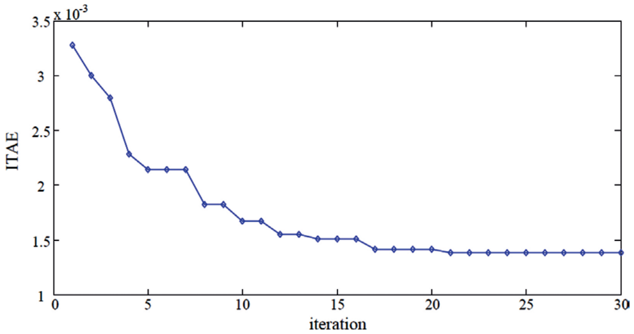

Figure 5. Change of ITAE function with respect to iterations.

Table 5. Optimized PSS parameters using different algorithms

PSS type	K_{PSS}	T_1	T_2	T_3	T_4
SOS-PSS [1]	16.1361	0.1888	0.0211	0.7916	0.5550
SCA-PSS [1]	46.8866	0.09288	0.0100	0.1238	0.0100
mAOA-PSS (proposed)	42.9745	0.08039	0.01023	0.07818	0.01015

Table 6. Eigenvalues and damping ratios of the electromechanical modes

PSS type	Dominant eigenvalue	Damping ratio
No stabilizer	$-0.0803 \pm j6.9824$	1.15%
SOS-PSS [1]	$-1.7449 \pm j4.6057$	35.43%
SCA-PSS [1]	$-2.0165 \pm j3.6201$	48.66%
mAOA-PSS (proposed)	$-2.6040 \pm j4.0211$	54.36%

applications of SOS-PSS [1] and SCA-PSS [1]. In addition, unlike SOS-PSS and SCA-PSS controllers, the oscillations in the rotor angle, speed and electrical power were prevented with the employment of the proposed mAOA-PSS controller. Moreover, it provided good damping characteristics to low-frequency oscillations by quickly stabilizing the system.

CONCLUSION

In this work, a modified arithmetic optimization algorithm has been developed by combining the original version of the arithmetic optimization algorithm with the simulated annealing technique in order to achieve a novel and enhanced metaheuristic algorithm. The search capability of the arithmetic optimization algorithm has been improved

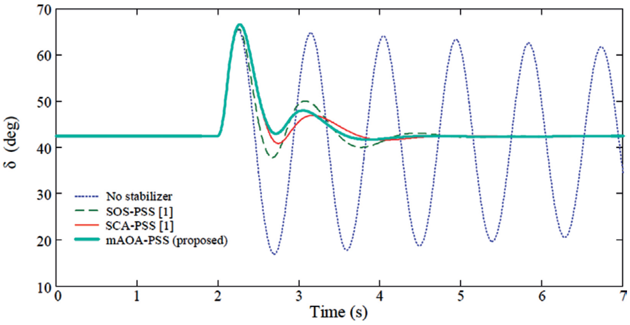

Figure 6. Change of δ rotor angle.

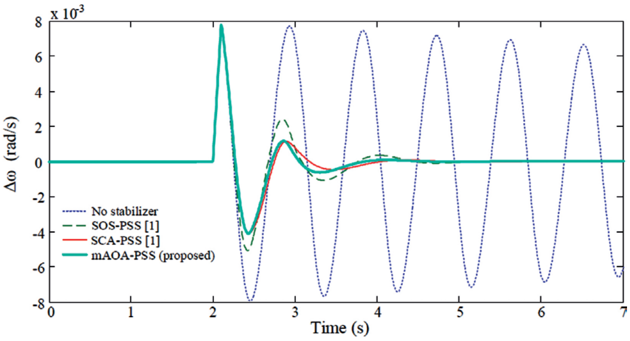

Figure 7. Change of $\Delta\omega$ speed deviation.

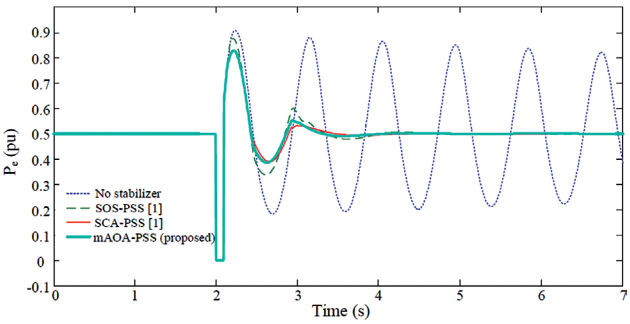

Figure 8. Change of P_e electrical power.

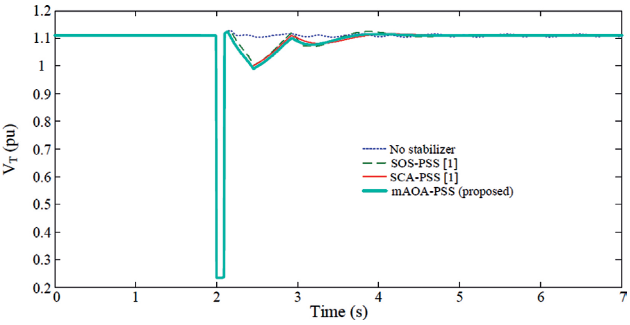

Figure 9. Change of V_T terminal voltage.

via inserting simulated annealing algorithm with an appropriate mechanism. Test functions with unimodal and multimodal features along with a real-world engineering problem have been employed to assess the capability of the developed mAOA algorithm and observe its promise. The evaluation process has been started by testing the proposed approach against the benchmark functions of Step, Sphere, Rastrigin, Rosenbrock, Quartic, Griewank, Schwefel, and Ackley. The obtained results have been compared with the original arithmetic optimization algorithm, simulated annealing algorithm, genetic algorithm, particle swarm optimization and gravitational search algorithm. The statistical performance of the proposed algorithm has demonstrated the highly competitive capability of the mAOA algorithm in terms of achieving the best, mean and standard deviation values. Further assessment of the proposed mAOA algorithm has been performed by testing it for complex systems through designing a PSS employed in a single-machine infinite-bus power system. The achieved system capability has been compared with sine-cosine and symbiotic organisms search algorithms-based PSS damping controllers since those approaches adopted the same power system parameters and the limits of the PSS parameters. The transient stability of the considered power system has greatly been enhanced and the damping characteristics of electromechanical modes has been improved via utilization of the proposed mAOA based PSS which confirmed its superior performance. To sum up, the implementation of the proposed hybrid mAOA algorithm to unimodal and multimodal test functions and one of the complex real-world engineering problems demonstrated that the mAOA algorithm is a powerful approach for optimization problems and a promising tool for complex real-world engineering problems.

AUTHORSHIP CONTRIBUTIONS

Authors equally contributed to this work.

DATA AVAILABILITY STATEMENT

The authors confirm that the data that supports the findings of this study are available within the article. Raw data that support the finding of this study are available from the corresponding author, upon reasonable request.

CONFLICT OF INTEREST

The author declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

ETHICS

There are no ethical issues with the publication of this manuscript.

REFERENCES

[1] Ekinci S. Optimal design of power system stabilizer using sine cosine algorithm. J Fac Eng Archit Gazi Univ 2019;34:1330–1350.

[2] Ekinci S. Power system stabilizer design for multimachine power system using bat search algorithm. Sigma J Eng Nat Sci 2015;33:627–637.

[3] Eker E, Kayri M, Ekinci S, Izci D. A New Fusion of ASO with SA Algorithm and Its Applications to MLP Training and DC Motor Speed Control. Arab J Sci Eng 2021;46:3889–3911. [CrossRef]

[4] Boussaïd I, Lepagnot J, Siarry P. A survey on optimization metaheuristics. Inf Sci (Ny) 2013;237:82–117.

[5] Yang X-S, Deb S, Zhao Y-X, Fong S, He X. Swarm intelligence: past, present and future. Soft Comput 2017;22:5923–5933. [CrossRef]

[6] Mirjalili S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. Knowledge-Based Syst 2015;89:228–249. [CrossRef]

[7] Wang G, Deb S, dos S. Coelho L. Elephant Herding Optimization. In: 2015 3rd International Symposium on Computational and Business Intelligence (ISCBI). 2015 3rd International Symposium on Computational and Business Intelligence (ISCBI), 07-09 December 2015. [CrossRef]

[8] Mirjalili S, Lewis A. The whale optimization algorithm. Adv Eng Softw 2016;95:51–67. [CrossRef]

[9] Saremi S, Mirjalili S, Lewis A. Grasshopper optimisation algorithm: Theory and application. Adv Eng Softw 2017;105:30–47. [CrossRef]

[10] Hashim FA, Houssein EH, Mabrouk MS, Al-Atabany W, Mirjalilid S. Henry gas solubility optimization: A novel physics-based algorithm. Futur Gener Comput Syst 2019;101:646–667. [CrossRef]

[11] Heidari AA, Mirjalili S, Faris H, Aljarah I, Mafarja M, Chen H. Harris hawks optimization: Algorithm and applications. Futur Gener Comput Syst 2019;97:849–872. [CrossRef]

[12] Wolpert DH, Macready WG. No free lunch theorems for optimization. IEEE Trans Evol Comput 1997;1:67–82. [CrossRef]

[13] Ekinci S, Hekimoglu B. Parameter optimization of power system stabilizer via Salp Swarm algorithm. In: 2018 5th International Conference on Electrical and Electronic Engineering (ICEEE). pp 143–147, 2018. [CrossRef]

[14] Ekinci S, Demirören A. Modeling, simulation, and optimal design of power system stabilizers using ABC algorithm. Turkish J Electr Eng Comput Sci 2016;24:1532–1546. [CrossRef]

[15] Alshammari BM, Guesmi T. New chaotic sunflower optimization algorithm for optimal tuning of power system Stabilizers. J Electr Eng Technol 2020;15:1985–1997. [CrossRef]

[16] Mijbas AF, Hasan BAA, Salah HA. Optimal stabilizer PID parameters tuned by chaotic particle swarm optimization for damping low frequency oscillations (LFO) for single machine infinite bus system (SMIB). J Electr Eng Technol 2020;15:1577–1584. [CrossRef]

[17] Hekimoğlu B. Robust fractional order PID stabilizer design for multi-machine power system using grasshopper optimization algorithm. J Fac Eng Archit Gazi Univ 2020;35:165–180.

[18] Verdejo H, Pino V, Kliemann W, Becker C, Delpiano J. Implementation of particle swarm optimization (PSO) algorithm for tuning of power system stabilizers in multimachine electric power systems. 2020;13:2093. [CrossRef]

[19] Ekinci S, Demiroren A, Hekimoglu B . Parameter optimization of power system stabilizers via kidney-inspired algorithm. Trans Inst Meas Control 2019;41:1405–1417. [CrossRef]

[20] Sabo A, Abdul Wahab NI, Othman ML, Mohd Jaffar MZA, Beiranvandet H. Optimal design of power system stabilizer for multimachine power system using farmland fertility algorithm. Int Trans Electr Energy Syst 2020;30: e12657. [CrossRef]

[21] Devarapalli R, Bhattacharyya B. A hybrid modified grey wolf optimization-sine cosine algorithm-based power system stabilizer parameter tuning in a multimachine power system. Optim Control Appl Methods 2020;41:1143–1159. [CrossRef]

[22] Butti D, Mangipudi SK, Rayapudi SR. An improved whale optimization algorithm for the design of multi-machine power system stabilizer. Int Trans Electr Energy Syst 2020;30:e12314. [CrossRef]

[23] Abualigah L, Diabat A, Mirjalili S, Abd Elaziz M, Gandomih . The arithmetic optimization algorithm. Comput Methods Appl Mech Eng 2021;376:113609. [CrossRef]

[24] Kirkpatrick S, Gelatt CD, Vecchi MP. Optimization by simulated annealing. Science (80-) 1983;220: 671–680.

[25] Rashedi E, Nezamabadi-pour H, Saryazdi S. GSA: A gravitational search algorithm. Inf Sci (Ny) 2009;179:2232–2248. [CrossRef]

[26] Holland JH. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, And Artificial Intelligence. Massachusett: MIT Press; 1992. [CrossRef]

[27] Kennedy J, Eberhart R. Particle swarm optimization. In: Proceedings of ICNN'95-International Conference on Neural Networks. IEEE, 1995;1942–1948.

[28] Sauer PW, Pai MA, Chow HJ. Power System Dynamics and Stability: With Synchrophasor Measurement and Power System Toolbox. 2nd ed. Hoboken: Wiley Press; 2017. [CrossRef]