Evolutionary Intelligence
https://doi.org/10.1007/s12065-021-00615-9

RESEARCH PAPER

A novel improved atom search optimization algorithm for designing power system stabilizer

Davut Izci[1]

Received: 19 November 2020 / Revised: 18 April 2021 / Accepted: 2 May 2021

Abstract
A novel hybrid algorithm developed by merging atom search optimization and simulated annealing algorithms is presented. The constructed improved algorithm, named as improved atom search optimization algorithm, was proposed for optimizing a power system stabilizer adopted in a single-machine infinite-bus power system. The evaluations were initially performed using several benchmark functions by comparing the results with genetic algorithm, simulated annealing technique, particle swarm optimization, gravitational search algorithm and the original version of atom search optimization algorithm. The obtained results showed the great promise of the developed hybrid algorithm in terms of the balance between exploration and exploitation phases. The performance of the proposed hybrid algorithm was also assessed through designing an optimally performing power system stabilizer for further evaluation. To do so, a power system stabilizer damping controller was formulated as an optimization problem and the improved algorithm was used to search for optimal controller parameters in order to show the applicability and greater performance of the proposed algorithm for such a complex real-world engineering problem. The obtained results for the latter case were compared with the best performing reported approaches of sine cosine algorithm and symbiotic organisms search algorithm. The comparisons clearly demonstrated the superiority of the proposed algorithm over other recently reported best performing algorithms for power system stabilizer design.

Keywords Improved atom search algorithm · Simulated annealing · Power system stability · Metaheuristic algorithms

1 Introduction

The integrity of power system interconnections is prone to failure due to low frequency oscillations that are not damped appropriately. In such a case, interruption of power supply and consequently financial loss may be observed. Therefore, damping low frequency oscillations in power system is of crucial importance for stabilizing the system and enhancing the power transfer capability. One way of dealing with such oscillations is to employ a power system stabilizer (PSS) for both single-machine infinite-bus and multi-machine power systems. A linearized model is considered for design of the conventional PSS. This approach takes a nominal operating point into account for determining the PSS parameters. 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.

Different approaches, such as self-tuning regulators and feedback linearization along with pole shifting and pole placement, are available for the design of PSS. However, those approaches present disadvantages such as intensive computations and long computer processing times [1]. Machine learning approaches [2] may be considered as a better alternative to those listed techniques in terms of dealing with PSS design, however, such mechanisms require a data set that covers a wide range of operating conditions. Besides, the system needs to be trained via the respective data set which would cause a longer computational time. Due to those requirements, such approaches possess disadvantage. To cope with those listed disadvantages, metaheuristic algorithms have been utilized in recent decades as an alternative approach to the previously listed techniques and they have gained considerable attention from the researchers in the field. The latter types of optimization algorithms are inspired from natural phenomena such as physical laws or biological behavior of living organisms; thus, it is feasible in practice to

✉ Davut Izci
davut.izci@batman.edu.tr

[1] Department of Electronics and Automation, Batman University, Batman 72060, Turkey

Published online: 13 May 2021

categorize them as evolution-based, swarm-based and physics-based approaches [3–5]. The advantage of metaheuristic algorithms arises from their ability to successfully run away from local optima which is achieved due to their stochastic nature and problem independent structure [6]. Therefore, metaheuristic algorithms can be thought of as black boxes that provides solutions for the problem of interest without concerning their nature. This feature makes them great tools to deal with real-world optimization problems.

Metaheuristic algorithms have already been demonstrated as an alternative method to tackle with offline tuning of PSS parameters by taking a wide range of operating conditions into account. A considerable number of such algorithms have recently been offered which can be listed as, cuckoo search [7], water cycle [8], moth search [9], slime mould [10], cultural [11], artificial bee colony [12], salp swarm [13], sine cosine [1], kidney-inspired [14], chaotic sunflower optimization [15], modified sine cosine and grey wolf optimization [16], particle swarm [17] and chaotic particle swarm [18] optimizations, improved whale optimization [19], grasshopper optimization [20] and farmland fertility [21] algorithms. Those metaheuristic algorithms listed above have led to better performing power systems with greater stability due to their ability to tune PSS controllers more efficiently. It is a fact that achieving further improvement of the PSS controller may still be feasible with novel algorithms despite the presented promise of the methods listed above which can be explained with the well-known no free lunch theorem [22]. The latter fact together with the promise shown by those listed approaches have motivated this study to further improve the capability of PSS by utilizing a more recent metaheuristic approach as an alternative algorithm in designing the related parameters. Therefore, atom search optimization (ASO) [23] algorithm was adopted for this study which is one of the recently proposed physics inspired and population-based technique proposed to achieve more efficient solutions for various optimization problems [24].

A wide variety of systems can be listed in terms of implementation of ASO algorithm. Few of the reported works on successful implementation of ASO can be listed as centralized thermoelectric generation system in heterogeneous temperature difference [25], line loss and cost minimization of shunt capacitors [26], feature selection [27], hydrogeologic parameter estimation [23], automatic voltage regulator control [28], modular multilevel converters design [29], wind power prediction [30], modelling fuel cells [31] and dispersion coefficient estimation in groundwater [24] along with tension/compression spring, welded beam, pressure vessel and speed reducer design problems [32]. Meanwhile, the successful demonstration of this algorithm for various systems may not be considered as the only assurance that it can be a perfect tool to achieve the optimal solution for the system under consideration [22]. This is caused due to undesired balance between global and local search phases of the algorithms which is a main issue for ASO algorithm, as well. The latter drawback makes ASO algorithm to suffer from stacking local optima and early convergence [33] despite its widely demonstrated capability (similar to many other metaheuristics) and requires to be dealt with.

Hybridization is one of the different options that are available to address such an issue [32] and can be considered as a significantly successful approach since it increases the feasibility of achieving more effective results. To achieve such a structure, simulated annealing (SA) algorithm can be adopted which is one of the popular algorithms that can be hybridized with other stochastic approaches to provide better solutions for optimization problems [34]. The SA algorithm [35] is a single-solution based approach and allows an easy implementation. Besides, it does not require longer computational time and has so far been shown to be a capable technique for efficient local search. Therefore, novel hybrid models can be constructed via convenient adaptation of SA algorithm with other metaheuristic approaches. Considering the latter case, the SA technique has been adopted to enhance the ASO algorithm in terms of abovementioned issues. Due to complementary features, the appropriate combination of ASO and SA algorithms has the potential to construct a more capable structure that can be used for optimization. Bearing the discussion so far, this paper proposes a novel hybrid algorithm, named improved atom search optimization (IASO), which has been developed by considering greater exploitative behavior of SA algorithm to address the issue that caused by the lack of balance between global and local search stages of ASO algorithm. The developed hybrid IASO algorithm utilizes SA technique to operate on worse solutions, as well, such that the potential of neighborhood solutions is not neglected.

The performance evaluation was firstly carried out by using benchmark functions of Step, Sphere, Rastrigin, Rosenbrock, Quartic, Griewank, Schwefel, and Ackley [36]. The obtained results were compared with five stochastic algorithms such as original ASO, SA, genetic algorithm (GA), particle swarm optimization (PSO) and gravitational search (GSA) algorithms. Then, designing a power system stabilizer employed in a single-machine infinite-bus (SMIB) system was considered for further performance assessment of the proposed IASO algorithm as a complex real-world engineering system. The ability of the designed system was observed through comparing it with SCA and symbiotic organisms search (SOS) algorithms based PSS controllers [1] as the latter two structures had the same power system and the limits of the PSS parameters. The assessment of the proposed hybrid algorithm has shown the IASO algorithm to be a useful and efficient optimization approach for the design of PSS controller employed in a power system. The comparisons demonstrated that the IASO algorithm has better

performance for a complex real-world engineering problem as the IASO algorithm-based PSS controller has greatly enhanced the small signal stability of the SMIB system and has improved the damping characteristics of electromechanical modes compared to other competitive approaches. The contribution of this work can be listed briefly as follows:

1. A novel hybrid structure, named IASO, was achieved using ASO and SA which has an enhanced balance between exploration and exploitation phases.
2. The obtained statistical results from unimodal and multimodal benchmark functions demonstrated the greater capability of the IASO in terms of achieving the best, mean and standard deviation values compared to original ASO, SA, GA, PSO and GSA algorithms.
3. The proposed IASO was utilized to achieve optimum parameters for stabilizing power system.
4. The IASO algorithm was shown to be a good choice for complex optimization problems as it enhanced the transient stability of the SMIB system greatly as well as providing improvement on the damping characteristics of electromechanical modes compared to other available competitive approaches.
5. Overall, the proposed IASO was demonstrated to be a powerful approach for optimizing problems with different nature compared to other competitive and reported best performing algorithms.

2 ASO algorithm

This algorithm is a population based global optimization approach that is inspired from molecular dynamics [24]. In another word, atomic motion behaving according to classical mechanics is mathematically represented via ASO. The following is a mathematical demonstration of relationship of an atomic system that is derived from Newton's second law where F_i denotes the force of the interaction whereas G_i is of the constraint together operating on atom i.

$$a_i = \frac{F_i + G_i}{m_i} \tag{1}$$

In the mathematical representation given in (1), a_i represents the acceleration of the atom i whereas m_i is the mass of the respective atom. The following equation expresses the interaction force (in dimension d and time t) acting on an atom i because of atom j. This definition (a revised version of the Lennard–Jones (L–J)) potential [37]) helps atoms to be able to converge to a specific point.

$$F'_{ij}(t) = -\eta(t)\left[2\left(h_{ij}(t)\right)^{13} - \left(h_{ij}(t)\right)^{7}\right] \tag{2}$$

In the above expression, $\eta(t)$ stands for the depth function which is defined in (3):

$$\eta(t) = \alpha\left(1 - \frac{t-1}{T}\right)^{3} e^{-\frac{20t}{T}} \tag{3}$$

where α and T represent the depth weight and the maximum number of iterations, respectively. The latter function is used for adjusting repulsion or attraction regions. The function of $h_{ij}(t)$ is defined in (4):

$$h_{ij}(t) = \begin{cases} h_{min}, \frac{r_{ij}(t)}{\sigma(t)} < h_{min} \\ \frac{r_{ij}(t)}{\sigma(t)}, h_{min} \le \frac{r_{ij}(t)}{\sigma(t)} \le h_{max} \\ h_{max}, \frac{r_{ij}(t)}{\sigma(t)} > h_{max} \end{cases} \tag{4}$$

where the distance between two atoms is represented by r whereas h_{min} stands for lower bound h_{max} for upper bound. The function of $h_{ij}(t)$ helps the occurrence of repulsion, attraction, or equilibrium. The terms of g_0 and u, given in (5), are chosen to be 1.1 and 1.24, respectively, in order to represent the limits.

$$h_{min} = g_0 + g(t), \quad h_{max} = u \tag{5}$$

g expresses the drift factor and given in (6). The algorithm is drifted from exploration to exploitation via the help of this expression.

$$g(t) = 0.1 \times \sin\left(\frac{\pi}{2} \times \frac{t}{T}\right) \tag{6}$$

The length scale is denoted by $\sigma(t)$ and given in (7). The latter term represents the collision diameter.

$$\sigma(t) = || \, x_{ij}(t), \frac{\sum_{j \in Kbest} x_{ij}(t)}{K(t)} \, ||_2 \tag{7}$$

In here, an atom population containing the best first K atoms are denoted by $Kbest$. The corresponding F' function behavior with respect to values of h is demonstrated in Fig. 1.

The total force acting on atom i (in dth dimension) can be written as in (8):

$$F_i^d(t) = \sum_{j \in Kbest} rand_j F_{ij}^d(t) \tag{8}$$

where $rand_j$ is the representation for a random number ranging between [0, 1]. The geometric constraint has a great effect on atomic motion which is simplified in ASO as assuming a covalent bond between the best and the other atoms. This can be written as given in (9) for atom i:

$$\theta_i(t) = \left[\left|x_i(t) - x_{best}(t)\right|^2 - (b_{i,best})^2\right] \tag{9}$$

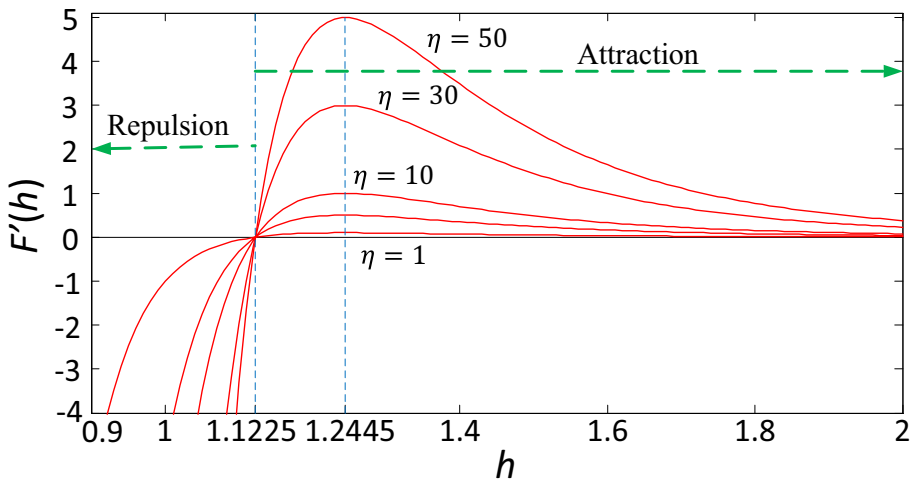

Fig. 1 Corresponding F' function behavior with different η values

where the best atom position is denoted by $x_{best}(t)$ (at iteration t) and the fixed length between the i^{th} and the best atoms is represented by $b_{i,best}$. The constraint force then can be written as in (10):

$$G_i^d(t) = \lambda(t)\left(x_{best}^d(t) - x_i^d(t)\right) \tag{10}$$

where the Lagrangian multiplier is represented by $\lambda(t)$ and given as follows:

$$\lambda(t) = \beta e^{-\frac{20t}{T}} \tag{11}$$

In the above equation, β denotes the multiplier weight. The i^{th} atom's acceleration at time t is given in (12):

$$a_i^d(t) = \frac{F_i^d(t)}{m_i^d(t)} + \frac{G_i^d(t)}{m_i^d(t)} \tag{12}$$

where the mass of i^{th} atom at time t is represented by $m_i(t)$. The latter expression briefly states that a better fitness value can be achieved via an atom with bigger mass which causes the acceleration to reduce. The following equation is used to compute the i^{th} atom's mass:

$$m_i(t) = \frac{M_i(t)}{\sum_{j=1}^{N} M_j(t)} \tag{13}$$

$$M_i(t) = e^{-\frac{Fit_i(t) - Fit_{best}(t)}{Fit_{worst}(t) - Fit_{best}(t)}} \tag{14}$$

where $Fit_{best}(t)$ and $Fit_{worst}(t)$ are representation of the atoms with the minimum and the maximum fitness values at time t, and are given in (15) and (16), respectively:

$$Fit_{best}(t) = min_{i\in(1,2,\ldots N)} Fit_i(t) \tag{15}$$

$$Fit_{worst}(t) = max_{i\in(1,2,\ldots N)} Fit_i(t) \tag{16}$$

In here, $Fit_i(t)$ is a representation of function fitness value of atom i at iteration t. To simplify the algorithm, the following can be used to express the velocity and the i^{th} atom's position at iteration $(t+1)$.

$$v_i^d(t+1) = rand_i^d v_i^d(t) + a_i^d(t) \tag{17}$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \tag{18}$$

For exploration enhancement, the possibility of interactions with the atoms having better fitness values as K neighbors of each atoms are required whereas for exploitation this should be fewer. In here, K is a function (time dependent) which is calculated as follows.

$$K(t) = N - (N-2) \times \sqrt{\frac{t}{T}} \tag{19}$$

3 SA algorithm

The SA algorithm is basically a mathematical representation of the metallurgical annealing process [38]. The stated annealing process helps formation of uniform crystals because of adopted heating and cooling stages. A random solution, X_i, is the starting point of SA technique. This solution is then used to determine a neighborhood solution X_i'. The fitness values of both random solution and its neighborhood solution are then computed and compared. If the fitness value of X_i' ($F(X_i')$) is smaller than that of X_i ($F(X_i)$), then SA sets $= X_i = X_i'$. 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 p which is defined as follows.

$$p = e^{-\frac{\Delta F}{T_k}}; \Delta F = F\left(X_i^{'}\right) - F\left(X_i\right) \tag{20}$$

In here, F is the control parameter for the fitness whereas T is of the temperature. The SA does not perform the replacement of X_i by $X_i^{'}$ in case where 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 values:

$$T_{k+1} = \mu T_k \tag{21}$$

where the cooling coefficient (which has a random constant value in [0, 1]) is denoted by μ.

4 Proposed IASO algorithm

4.1 Framework of IASO algorithm

As mentioned earlier, the original version of ASO have been adopted to resolve a variety of engineering problems successfully. However, similar to many other global optimization techniques, ASO is not efficient enough for providing sufficient solutions to problems with large or small dimensions. To overcome the latter issue, SA was adopted to enhance effectiveness of the ASO. Therefore, the proposed IASO is a hybrid version of ASO and SA algorithms. The pseudocode of the proposed IASO algorithm is provided in Algorithm 1.

Algorithm 1
Pseudocode of proposed IASO algorithm

Inputs: Fitness function, population size, boundary conditions, maximum number of iterations (t_{max}), dimension of optimization problem, depth weight, multiplier weight, initial temperature, minimum temperature, and cooling rate.
Outputs: The best solution X_{best} and its objective function $F(X_{best})$.
Randomly initialize X_i (solutions) and their velocity V_i and set $Fit_{best} = \infty$.
Put $t = 1$.
while (stopping condition is not met ($t < t_{max}$)) **do**
  for each atom X_i **do** *% Procedure of ASO algorithm*
    Calculate the fitness value Fit_i;
    if ($Fit_i < Fit_{best}$) **then**
      $Fit_{best} = Fit_i$ and $X_{best} = X_i$;
    end if
    Calculate the atom's mass using equations (13) and (14);
    Determine its K neighbors using equation (19);
    Calculate F_i, G_i and acceleration using equations (8), (10) and (12), respectively;
    Update the velocities and positions using equations (17) and (18), respectively;
  end for
    Set best solution found by ASO as starting solution for SA *% Procedure of SA algorithm*
    Generate a new solution X_i' in a neighborhood of the current solution X_i;
    Calculate $\Delta F = F(X_i') - F(X_i)$;
    if ($\Delta F \leq 0$ or $rand(0,1) < e^{-\frac{\Delta F}{T_k}}$) **then**
      Accept the new solution, $X_i = X_i'$ and $F(X_i) = F(X_i')$;
    end if
    Apply cooling schedule: $T_{k+1} = \mu T_k$;
    Update the best solution X_{best}
  $t = t + 1$
end while
Return the best solution X_{best}

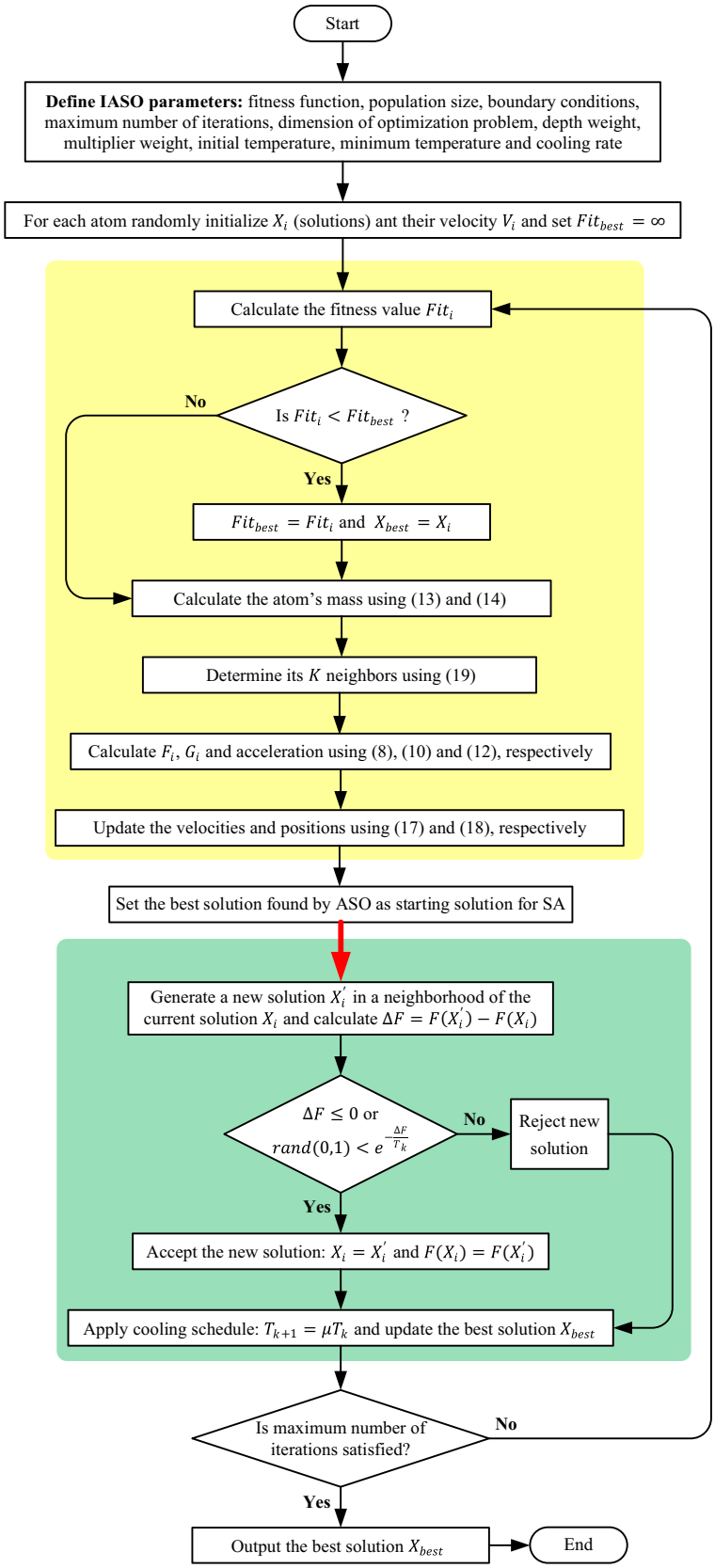

Fig. 2 Flowchart for the IASO algorithm

In the proposed IASO algorithm, the SA takes care of the exploitation enhancement as it is a good local search algorithm that has hill-climbing ability [38]. In SA technique, random solutions are initialized as a necessary starting point in the search space. The solution of the problem is evaluated by generating and accepting or rejecting the neighbor solutions. The solution is shifted towards the better neighbor's solution to explore the search space. This behavior helps assisting the ASO to avoid local minimum. On the other hand, ASO deals with the global search. In another word, the IASO algorithm makes use of the fast-optimal search capability of ASO algorithm together with hill-climbing feature of SA technique.

The flowchart given in Fig. 2 illustrates the process of the proposed IASO algorithm. As shown in the respective flowchart, the algorithm initially defines the ASO and SA related parameters as well as creating fitness value (set to infinity) and a random set of atoms with respective velocities. Then, through iterations, the fitness values for each atom are calculated and compared with the best fitness value. If a better value is obtained, then this is used to update the best solution. The algorithm then executes the rest of the steps shown in the respective flowchart. In each generation, the best solution of ASO algorithm of respective iteration is adopted as the starting solution of SA algorithm. In this way, the ability of ASO algorithm in terms of exploring the search space is enhanced.

4.2 Computational complexity of IASO algorithm

The computational complexity of an optimization algorithm allows evaluation of the run time of an algorithm since it is a key metric for such a task. The structure of an algorithm can be used to define the computational complexity. Bearing this in mind, the computational complexity of the proposed IASO algorithm essentially depends on three factors which are the initialization, fitness function evaluation and updating of solutions. Therefore, the computational complexity for basic ASO ($O(ASO)$) and the proposed IASO ($O(IASO)$) algorithms are defined as follows by considering the steps of those algorithms:

$$O(ASO) = O(N) + O(N \times T) + O(N \times D \times T) \tag{22}$$

$$O(IASO) = O(ASO) + O(T) + O(D \times T) \tag{23}$$

where N, T and D respectively represent the swarm size, maximum number of iterations and dimension of the problem.

5 Experimental setup and results

5.1 Benchmark functions

Table 1 lists the employed benchmark functions (with related dimension (n) and range) that have been used for initial performance evaluation of the proposed IASO algorithm. As can be seen, the table contains both unimodal and multimodal types of test functions which together helped measuring the performance of the IASO algorithm effectively.

The functions from $F_1(x)$ to $F_4(x)$ are unimodal functions and good tools for exploitation assessment whereas the ones from $F_5(x)$ to $F_8(x)$ are multimodal functions and

Table 1 The details of adopted benchmark functions for performance evaluation

Name	Type	Function equation	n	Range	F_{min}
Sphere	Unimodal	$F_1(x) = \sum_{i=1}^{n} x_i^2$	30	$[-100, 100]^n$	0
Rosenbrock	Unimodal	$F_2(x) = \sum_{i=1}^{n-1} \left(100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2\right)$	30	$[-30, 30]^n$	0
Step	Unimodal	$F_3(x) = \sum_{i=1}^{n} (x_i + 0.5)^2$	30	$[-100, 100]^n$	0
Quartic	Unimodal	$F_4(x) = \sum_{i=1}^{n} ix_i^4 + random[0, 1)$	30	$[-1.28, 1.28]^n$	0
Schwefel	Multimodal	$F_5(x) = -\sum_{i=1}^{n} \left(x_i \sin\left(\sqrt{\lvert x_i \rvert}\right)\right)$	30	$[-500, 500]^n$	$-12,569.5$
Rastrigin	Multimodal	$F_6(x) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos\left(2\pi x_i\right) + 10\right]$	30	$[-5.12, 5.12]^n$	0
Ackley	Multimodal	$F_7(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos\left(2\pi x_i\right)\right) + 20 + e$	30	$[-32, 32]^n$	0
Griewank	Multimodal	$F_8(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30	$[-600, 600]^n$	0

Table 2 Initial parameter values of compared algorithms

Algorithm	Parameter	Value
SA [38]	Initial temperature	0.10
	Cooling factor	0.98
	Mutation rate	0.50
GA [39]	Selection	Roulette wheel
	Crossover rate	0.80
	Mutation rate	0.40
PSO [40]	Cognitive coefficient	2
	Social coefficient	2
	Inertia constant	Linearly decrease from 0.80 to 0.20
GSA [36]	Gravitational constant	100
	Decreasing coefficient	20
ASO [23]	Depth weight	50
	Multiplier weight	0.20

good for evaluating the exploration ability of the algorithms. A detailed description of the employed test functions can be found in Ref. [36].

5.2 Compared algorithms

Five stochastic algorithms were used to perform the comparisons using the benchmark functions mentioned in the previous subsection. Those algorithms are consisted of three popular ones such as GA, SA, and PSO along with two recently proposed ones such as the original version of ASO and GSA algorithms. The chosen parameters of the respective algorithms for this study are listed in Table 2.

5.3 Results and discussion

The swarm size was chosen as 50 whereas maximum iterations was set to 1000 to run the proposed IASO algorithm. The IASO was run for 50 times in order to provide a fair comparison with PSO, GA, SA, ASO and GSA algorithms [23]. The obtained statistical results via listed algorithms for utilized benchmark functions of $F_1(x)$ to $F_8(x)$ are provided in Table 3. The best results are highlighted in bold. Compared with other algorithms, the best results in terms of mean, standard deviation (StDev) and best were achieved via the proposed IASO algorithm in overall. This can also be observed from the ranking of the algorithms. This is a good indication of desired balance between global and local search phases of IASO algorithm which demonstrates its promise for optimization problems.

Apart from the statistical results, the compared algorithms have also been analyzed in terms of computational complexity. In terms of original ASO and the proposed IASO algorithms, the related discussion has been provided in sub-Sect. 4.2. A similar discussion is valid for the compared algorithms, as well. Therefore, it is worth to note that the computational complexities of the compared algorithms are the same as the original ASO. However, in terms of IASO algorithm, the computational complexity is only a little higher than those algorithms due to inclusion of SA.

6 PSS design for SMIB system

6.1 Power system model

An infinite bus is the source of constant frequency and voltage either in magnitude or angle. A single-machine infinite-bus model was mathematically analyzed in Heffron-Philips model and was used by DeMello and Concordia extensively for small signal stability analysis. In this paper, a power system consisting of single-machine infinite-bus through a double circuit transmission line, as shown in Fig. 3, has been considered. The machine was assumed to be equipped with a fast exciter. A PSS was integrated with this system for improving small signal oscillations.

The 4th order model of a single-machine infinite-bus power system can be given as follows [41]:

$$\dot{\delta} = \omega_0(\omega - 1) \tag{24}$$

$$\dot{\omega} = \left[P_m - P_e - D(\omega - 1)\right]/M \tag{25}$$

$$\dot{E}_q^{'} = \left[E_{fd} - E_q^{'} - \left(x_d - x_d^{'}\right)i_d\right]/T_{d0}^{'} \tag{26}$$

$$\dot{E}_{fd} = \left[K_A\left(V_{ref} - V_T + U_{PSS}\right) - E_{fd}\right]/T_A \tag{27}$$

$$P_e = E_q^{'}i_q + \left(x_q - x_d^{'}\right)i_d i_q \tag{28}$$

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.

Table 3 The obtained comparative results for unimodal and multimodal functions

Functions	Metric	GSA [23]	PSO [23]	GA [23]	SA [23]	ASO [23]	IASO (proposed)
$F_1(x)$	Mean	2.11E−17	1.46E−04	1.02E−02	2.04E−13	2.68E−21	7.88E−28
	StDev	6.67E−18	1.19E−04	5.07E−03	6.14E−14	3.65E−21	2.74E−27
	Best	1.06E−17	9.56E−06	2.98E−03	7.76E−14	3.52E−22	5.61E−36
	Rank	3	5	6	4	2	1
$F_2(x)$	Mean	2.81E+01	1.34E+02	9.71E+01	1.06E+03	2.48E+01	5.81E+00
	StDev	1.13E+01	1.29E+02	1.29E+02	2.05E+03	5.16E−01	4.32E−01
	Best	2.58E+01	2.62E+01	9.70E+00	2.32E+01	1.66E+01	4.97E+00
	Rank	3	5	4	6	2	1
$F_3(x)$	Mean	0	1.33E−01	0	5.67E−01	0	0
	StDev	0	3.46E−01	0	7.28E−01	0	0
	Best	0	0	0	0	0	0
	Rank	1	5	1	6	1	1
$F_4(x)$	Mean	2.08E−02	6.63E−02	5.05E−02	1.24E−01	3.56E−02	6.28E−04
	StDev	7.72E−03	1.95E−02	2.18E−02	3.97E−02	1.95E−02	4.63E−04
	Best	7.01E−03	2.98E−02	1.32E−02	6.25E−02	3.61E−02	1.54E−04
	Rank	2	5	4	6	3	1
$F_5(x)$	Mean	−2.65E+03	−5.20E+03	−6.80E+03	−9.29E+03	−7.43E+03	−1.16E+04
	StDev	3.42E+02	5.29E+02	6.33E+02	4.02E+02	4.22E+02	2.58E+02
	Best	−3.49E+03	−6.79E+03	−8.15E+03	−1.01E+04	−5.56E+03	−1.23E+04
	Rank	6	5	4	2	3	1
$F_6(x)$	Mean	1.51E+01	2.93E+01	1.25E+01	5.44E+01	0	0
	StDev	4.44E+00	6.88E+00	2.92E+00	1.38E+01	0	0
	Best	7.96E+00	1.73E+01	6.09E+00	2.69E+01	0	0
	Rank	4	5	3	6	1	1
$F_7(x)$	Mean	3.69E−09	7.43E−03	2.12E−02	3.44E−01	3.00E−11	8.88E−16
	StDev	3.96E−10	1.42E−02	4.55E−03	4.47E−01	2.15E−11	0
	Best	2.96E−09	5.49E−04	1.07E−02	8.29E−08	1.13E−11	8.88E−16
	Rank	3	4	5	6	2	1
$F_8(x)$	Mean	4.47E+00	2.28E−02	1.84E−02	1.24E−02	0	0
	StDev	2.05E+00	2.74E−02	9.77E−03	1.03E−02	0	0
	Best	1.94E+00	5.88E−05	6.12E−03	3.16E−06	0	0
	Rank	6	5	4	3	1	1
Average rank		3.5000	4.8750	3.8750	4.8750	1.8750	1
Overall rank		3	5	4	5	2	1

The linearized model of a single-machine infinite-bus power system is obtained as follows where six well-established constants (represented by $K_1 - K_6$) describes the interaction among the variables in power system [41].

$$\Delta\dot{\delta} = \omega_0 \Delta\omega \tag{29}$$

$$\Delta\dot{\omega} = -\frac{K_1}{M}\Delta\delta - \frac{D}{M}\Delta\omega - \frac{K_2}{M}\Delta E_q^{'} \tag{30}$$

$$\Delta\dot{E}_q^{'} = -\frac{K_4}{T_{d0}^{'}}\Delta\delta - \frac{1}{K_3 T_{d0}^{'}}\Delta E_q^{'} + \frac{1}{T_{d0}^{'}}\Delta E_{fd} \tag{31}$$

$$\Delta\dot{E}_{fd} = -\frac{K_A K_5}{T_A}\Delta\delta - \frac{K_A K_6}{T_A}\Delta E_q^{'} - \frac{1}{T_A}\Delta E_{fd} + \frac{K_A}{T_A}\Delta U_{PSS} \tag{32}$$

The following equations can be used to arrange the state space form of the system given in (29)–(32):

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{33}$$

where:

$$x(t) = \left[\Delta\delta\Delta\omega\Delta E_q^{'}\Delta E_{fd}\right]^T, u(t) = \Delta U_{PSS} \tag{34}$$

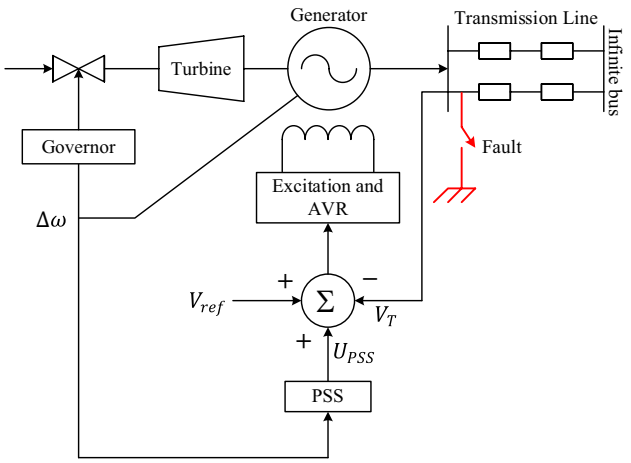

Fig. 3 Single-machine infinite-bus system with PSS

$$A = \begin{bmatrix} 0 & 2\pi f & 0 & 0 \\ -\frac{K_1}{M} & -\frac{D}{M} & -\frac{K_2}{M} & 0 \\ -\frac{K_4}{T'_{d0}} & 0 & -\frac{1}{K_3 T'_{d0}} & \frac{1}{T'_{d0}} \\ -\frac{K_A K_5}{T_A} & 0 & -\frac{K_A K_6}{T_A} & -\frac{1}{T_A} \end{bmatrix} \tag{35}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & \frac{K_A}{T_A} \end{bmatrix}^T \tag{36}$$

The linearized dynamic model of SMIB system with PSS controller is represented in Fig. 4. The machine, transformer and transmission along with the exciter and PSS data of the analyzed SMIB system are listed in Table 4.

6.2 Structure of PSS

Operating an auxiliary stabilizing signal through the excitation system is the main function of a PSS as it is used to

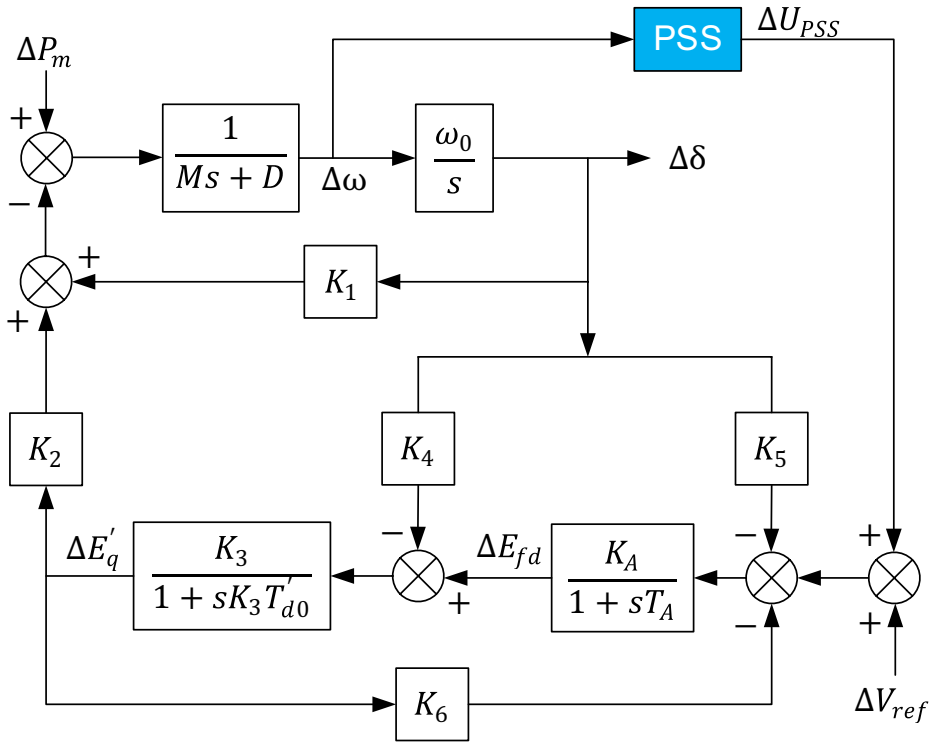

Fig. 4 Representation of Heffron-Philips model with PSS controller

Table 4 Parameters of test system [1]

Machine	$M = 2H = 6.4pu$, $x_d = 2.5pu$, $x_q = 2.1pu$, $x'_d = 0.39pu$, $T'_{d0} = 9.6s$, $\omega_0 = 2\pi \times 60rad/s$, $P_{e0} = 0.5pu$, $D = 0$,$\delta_0 = 42.4°$
Transmission line and transformer	$X_{L1} = X_{L2} = 0.8pu$, $X_T = 0.1pu$
Exciter and PSS	$K_A = 400$, $T_A = 0.02s$, $E_{fd}^{min} = -5pu$, $E_{fd}^{max} = 5pu$, $U_{PSS}^{min} = -0.2pu$,$U_{PSS}^{max} = 0.2pu$

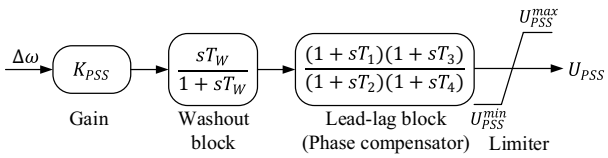

Fig. 5 Block diagram of two-stage lead-lag PSS

damp out the generator rotor oscillations [41]. The structure of a widely used conventional lead-lag PSS is given in (37).

$$U_{PSS} = K_{PSS}\left(\frac{sT_W}{1+sT_W}\right)\left(\frac{1+sT_1}{1+sT_2}\right)\left(\frac{1+sT_3}{1+sT_4}\right)\Delta\omega \quad (37)$$

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 Fig. 5. 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 $5s$.

6.3 IASO based PSS design

The performance index provided in (38) is known as the integral of time multiplied absolute error (ITAE) which can be used as an objective function for such a design problem since it is feasible to choose the parameters of the PSS for minimizing the respective function:

$$ITAE = \int_0^{t_{sim}} t \cdot |\Delta\omega(t)| \cdot dt \quad (38)$$

In the above definition, $\Delta\omega(t)$ is the rotor speed deviation following a large disturbance and t_{sim} is simulation time. The requirement of minimal dynamic plant information is the advantage of this performance index. The related optimization problem can be minimized via the respective objective function subjected to the following criteria:

$$0.01 \le K_{PSS} \le 100; \quad 0.01 \le T_i \le 1 \quad (39)$$

where $i = 1, 2, 3, 4$. The application of the proposed IASO algorithm to PSS design in SMIB system is illustrated in Fig. 6.

After performing the optimization procedure given in the respective figure, the stability of the system can be observed to increase to the highest level. The swarm size (atom population) was set to 40 whereas the maximum iteration number (stopping criteria) was chosen to be 50 for the optimization module given in the respective figure. The parameters of the recommended IASO algorithm for effective PSS design are listed in Table 5. The ITAE function value was calculated for each atom in the swarm via integration of IASO with the SMIB system using MATLAB/Simulink software. The IASO was run for 25 times using a simulation time (t_{sim}) of $10s$. The PSS parameters corresponding minimum ITAE value were found to be $K_{PSS} = 41.3806$, $T_1 = 0.0669$, $T_2 = 0.0101$, $T_3 = 0.0891$ and $T_4 = 0.0104$.

6.4 Simulation results

The superiority of proposed IASO algorithm in designing PSS was evaluated by comparing it with the most recent study published in a prestigious journal through eigenvalue analysis and nonlinear time domain simulation. The most convenient approaches chosen for comparison were SCA-PSS [1] and SOS-PSS [1] damping controllers since those approaches adopted the same power system parameters and the limits of the PSS parameters. PSS parameters that were optimized with different algorithms are presented in Table 6.

6.4.1 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 proposed IASO based controller improves the linear model stability of the system.

The system eigenvalues ($\lambda = \sigma \pm j\omega$) and damping ratios ($\xi$) of the electromechanical modes related to the systems without using the stabilizer and with optimized PSS controller parameters using other algorithms (IASO, SCA and SOS) are given in Table 7.

As can be seen from the table, the system is insufficiently damped in case of no PSS (ξ=0.0115). In addition, compared to the SCA-PSS [1] and SOS-PSS [1] controllers, the electromechanical modes of the proposed IASO-PSS are further to the left of the s-plane (damping factor $\sigma = -2.5985$) along with a greater damping ratio ($\xi = 0.5345$). Therefore, IASO based PSS greatly enhances the small signal stability of the SMIB system and improves the damping characteristics of electromechanical modes.

6.4.2 Nonlinear time domain simulation

A three-phase fault has been applied at the generator terminal busbar at $t = 2s$ and cleared after 6-cycle (0.10 s). The original system was restored upon the fault clearance. The system rotor angle (δ), speed deviation ($\Delta\omega$) and electrical power (P_e) responses are shown in Figs. 7, 8 and 9, respectively. It is obvious from these figures that the power system oscillations are inadequately damped although the system

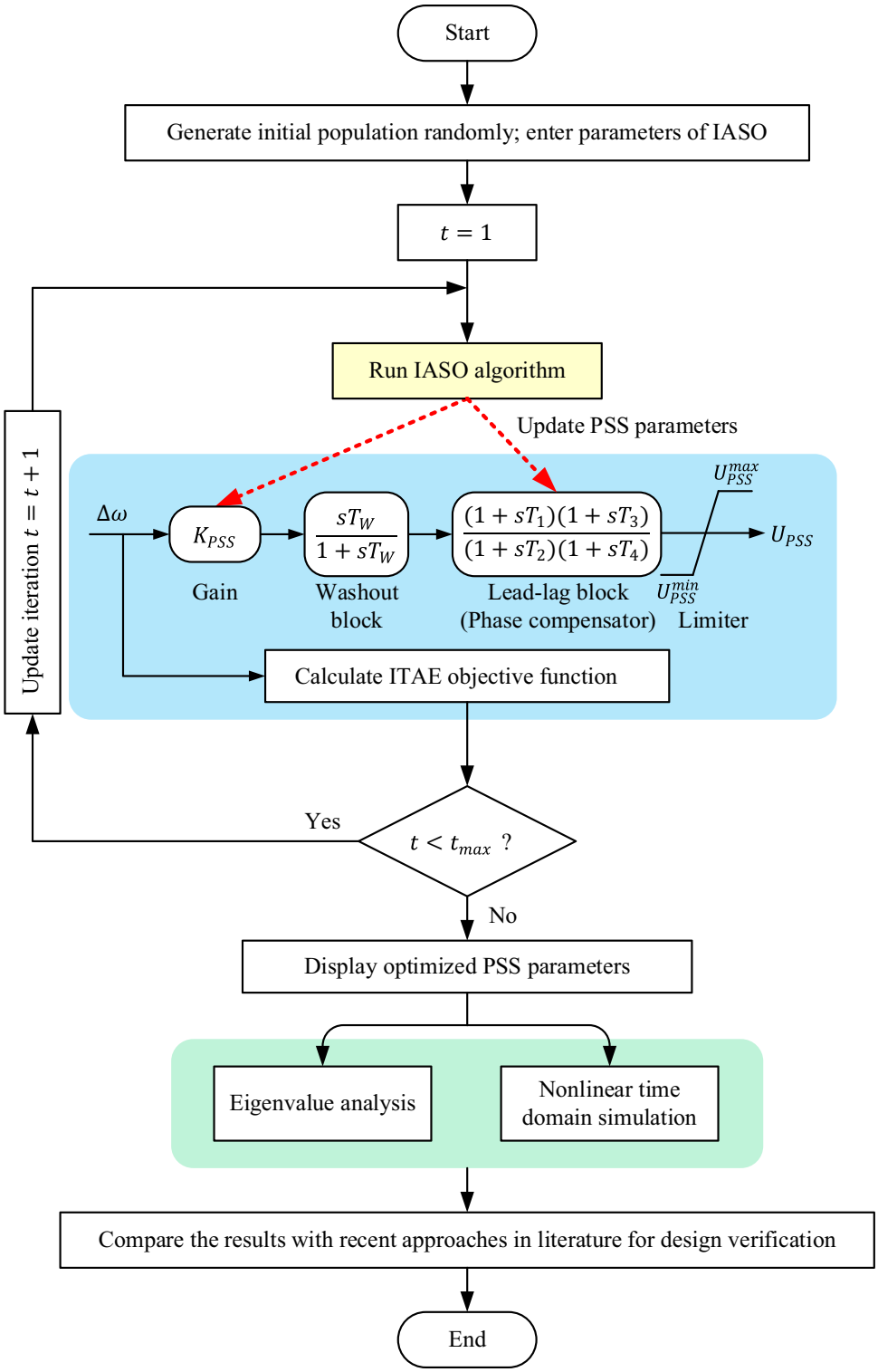

Fig. 6 Detailed flowchart of the proposed IASO based PSS design

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

Table 5 Parameters of IASO algorithm for PSS design

Parameter	Value
Depth and multiplier weights	50 and 0.20
Initial temperature and cooling factor	0.10 and 0.98
Swarm size	40
Maximum number of iterations	50
Variable number	5 (K_{PSS}, T_1, T_2, T_3 and T_4)

Table 6 Optimized PSS parameters using different algorithms

PSS type	K_{PSS}	T_1	T_2	T_3	T_4
IASO-PSS (proposed)	41.3806	0.0669	0.0101	0.0891	0.0104
SCA-PSS [1]	46.8866	0.09288	0.0100	0.1238	0.0100
SOS-PSS [1]	16.1361	0.1888	0.0211	0.7916	0.5550

Table 7 Eigenvalues and damping ratios of the electromechanical modes

PSS type	Dominant eigenvalue	Damping ratio
No stabilizer	$-0.0803 \pm j6.9824$	1.15%
IASO-PSS (proposed)	$-2.5985 \pm j4.1092$	53.45%
SCA-PSS [1]	$-2.0165 \pm j3.6201$	48.66%
SOS-PSS [1]	$-1.7449 \pm j4.6057$	35.43%

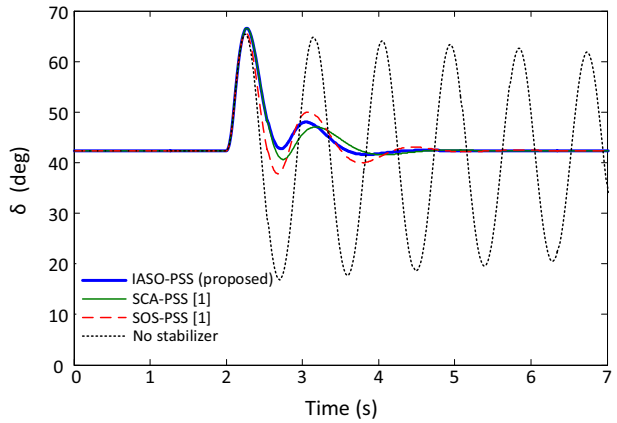

Fig. 7 Change of δ rotor angle

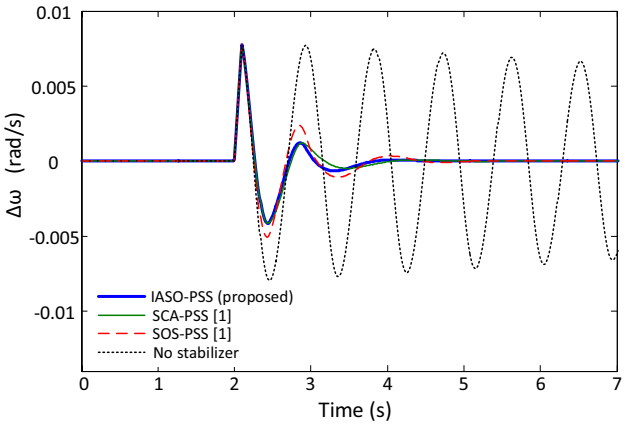

Fig. 8 Change of $\Delta\omega$ speed deviation

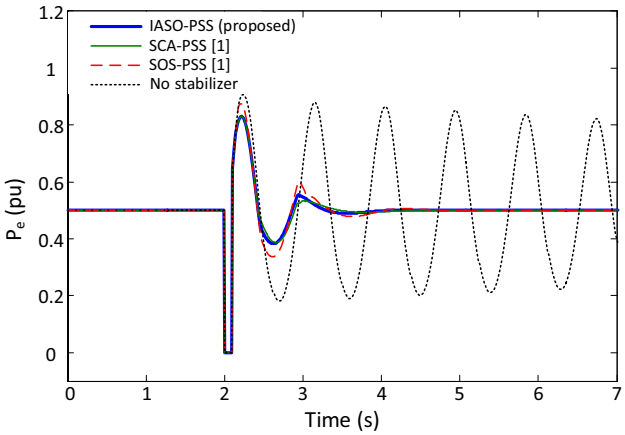

Fig. 9 Change of P_e electrical power

7 Conclusion

In this work, a novel improved ASO algorithm was developed by combining the original ASO with SA technique. In this way, the search capability of ASO was improved. Test functions and a real-world engineering problem were employed to evaluate the promise of the proposed IASO algorithm. The evaluation process started by testing the proposed approach against the benchmark functions of Step, Sphere, Rastrigin, Rosenbrock, Quartic, Griewank, Schwefel, and Ackley. The obtained results were compared with original ASO, SA, GA, PSO and GSA algorithms. The statistical performance of the proposed algorithm demonstrated the greater capability of the IASO in terms of achieving the best, mean and standard deviation values. Further assessment of the proposed IASO was performed by testing it for complex systems through designing a PSS employed in a SMIB system. The achieved system capability was compared with SCA and SOS 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 SMIB system was greatly enhanced and the damping characteristics of electromechanical modes was improved via utilization of the proposed IASO based PSS which confirmed the superior performance of the algorithm. To sum up, the implementation of the proposed hybrid IASO

algorithm to unimodal and multimodal test functions and one of the complex real-world engineering problems showed the stated algorithm to be a powerful approach for optimization problems.

Declarations

Conflict of interest The author declares that he has no conflict of interest.

References

1. Ekinci S (2019) Optimal design of power system stabilizer using sine cosine algorithm. J Fac Eng Archit Gazi Univ 34:1330–1350
2. Huang XL, Ma X, Hu F (2018) Machine learning and intelligent communications. Mob Networks Appl 23:68–70. https://doi.org/10.1007/s11036-017-0962-2
3. Abualigah L, Yousri D, Elaziz MA et al (2021) Aquila optimizer: a novel meta-heuristic optimization algorithm. Comput Ind Eng. https://doi.org/10.1016/j.cie.2021.107250
4. Abualigah LMQ (2019) Feature selection and enhanced Krill Herd algorithm for text document clustering. Springer International Publishing, Cham
5. Abualigah L, Diabat A (2021) Advances in sine cosine algorithm: a comprehensive survey. Artif Intell Rev 54:2567–2608. https://doi.org/10.1007/s10462-020-09909-3
6. Abualigah L, Diabat A, Mirjalili S et al (2021) The arithmetic optimization algorithm. Comput Methods Appl Mech Eng. https://doi.org/10.1016/j.cma.2020.113609
7. Chitara D, Niazi KR, Swarnkar A, Gupta N (2018) Cuckoo search optimization algorithm for designing of a multimachine power system stabilizer. IEEE Trans Ind Appl 54:3056–3065. https://doi.org/10.1109/TIA.2018.2811725
8. Ghaffarzadeh N (2015) Water cycle algorithm based power system stabilizer robust design for power systems. J Electr Eng 66:91–96. https://doi.org/10.1515/jee-2015-0014
9. Razmjooy N, Razmjooy S, Vahedi Z et al (2021) A new design for robust control of power system stabilizer based on Moth search algorithm. In: Razmjooy N, Ashourian M, Foroozandeh Z (eds) Lecture notes in electrical engineering. Springer International Publishing, Cham, pp 187–202
10. Ekinci S, Izci D, Zeynelgil HL, Orenc S (2020) An application of Slime Mould algorithm for optimizing parameters of power system stabilizer. In: 2020 4th international symposium on multidisciplinary studies and innovative technologies (ISMSIT). IEEE, pp 1–5
11. Khodabakhshian A, Hemmati R (2013) Multi-machine power system stabilizer design by using cultural algorithms. Int J Electr Power Energy Syst 44:571–580
12. Ekinci S, Demirören A (2016) Modeling, simulation, and optimal design of power system stabilizers using ABC algorithm. Turkish J Electr Eng Comput Sci 24:1532–1546. https://doi.org/10.3906/elk-1311-208
13. Ekinci S, Hekimoglu B (2018) Parameter optimization of power system stabilizer via Salp Swarm algorithm. In: 2018 5th international conference on electrical and electronic engineering (ICEEE). pp 143–147
14. Ekinci S, Demiroren A, Hekimoglu B (2019) Parameter optimization of power system stabilizers via kidney-inspired algorithm. Trans Inst Meas Control 41:1405–1417. https://doi.org/10.1177/0142331218780947
15. Alshammari BM, Guesmi T (2020) New chaotic sunflower optimization algorithm for optimal tuning of power system stabilizers. J Electr Eng Technol 15:1985–1997. https://doi.org/10.1007/s42835-020-00470-1
16. Devarapalli R, Bhattacharyya B (2020) A hybrid modified grey wolf optimization-sine cosine algorithm-based power system stabilizer parameter tuning in a multimachine power system. Optim Control Appl Methods 41:1143–1159
17. Verdejo H, Pino V, Kliemann W, et al (2020) Implementation of particle swarm optimization (PSO) algorithm for tuning of power system stabilizers in multimachine electric power systems. Energies 13
18. Mijbas AF, Hasan BAA, Salah HA (2020) 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 15:1577–1584. https://doi.org/10.1007/s42835-020-00442-5
19. Butti D, Mangipudi SK, Rayapudi SR (2020) An improved whale optimization algorithm for the design of multi-machine power system stabilizer. Int Trans Electr Energy Syst. https://doi.org/10.1002/2050-7038.12314
20. Hekimoğlu B (2020) Robust fractional order PID stabilizer design for multi-machine power system using grasshopper optimization algorithm. J Fac Eng Archit Gazi Univ 35:165–180
21. Sabo A, Abdul Wahab NI, Othman ML et al (2020) Optimal design of power system stabilizer for multimachine power system using farmland fertility algorithm. Int Trans Electr Energy Syst. https://doi.org/10.1002/2050-7038.12657
22. Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. IEEE Trans Evol Comput 1:67–82. https://doi.org/10.1109/4235.585893
23. Zhao W, Wang L, Zhang Z (2019) Atom search optimization and its application to solve a hydrogeologic parameter estimation problem. Knowl Based Syst 163:283–304. https://doi.org/10.1016/j.knosys.2018.08.030
24. Zhao W, Wang L, Zhang Z (2019) A novel atom search optimization for dispersion coefficient estimation in groundwater. Futur Gener Comput Syst 91:601–610. https://doi.org/10.1016/j.future.2018.05.037
25. Yang B, Zhang M, Zhang X et al (2020) Fast atom search optimization based MPPT design of centralized thermoelectric generation system under heterogeneous temperature difference. J Clean Prod. https://doi.org/10.1016/j.jclepro.2019.119301
26. Rizk-Allah RM, Hassanien AE, Oliva D (2020) An enhanced sitting–sizing scheme for shunt capacitors in radial distribution systems using improved atom search optimization. Neural Comput Appl 32:13971–13999. https://doi.org/10.1007/s00521-020-04799-6
27. Too J, Abdullah AR (2020) Chaotic atom search optimization for feature selection. Arab J Sci Eng. https://doi.org/10.1007/s13369-020-04486-7
28. Ekinci S, Demiroren A, Zeynelgil H, Hekimoğlu B (2020) An opposition-based atom search optimization algorithm for automatic voltage regulator system. J Fac Eng Archit Gazi Univ 35:1141–1158
29. Diab AAZ, Ebraheem T, Aljendy R et al (2020) Optimal design and control of MMC STATCOM for improving power quality indicators. Appl Sci 10:2490. https://doi.org/10.3390/app10072490
30. Li LL, Chang YB, Tseng ML et al (2020) Wind power prediction using a novel model on wavelet decomposition-support vector machines-improved atomic search algorithm. J Clean Prod. https://doi.org/10.1016/j.jclepro.2020.121817
31. Agwa AM, El-Fergany AA, Sarhan GM (2019) Steady-state modeling of fuel cells based on atom search optimizer. Energies 12:1884. https://doi.org/10.3390/en12101884

32. Sun P, Liu H, Zhang Y et al (2021) An intensify atom search optimization for engineering design problems. Appl Math Model 89:837–859. https://doi.org/10.1016/j.apm.2020.07.052
33. Sun P, Zhang Y, Liu J, Bi J (2020) An improved atom search optimization with cellular automata, a Lévy flight and an adaptive weight strategy. IEEE Access 8:49137–49159. https://doi.org/10.1109/ACCESS.2020.2979921
34. Pan X, Xue L, Lu Y, Sun N (2019) Hybrid particle swarm optimization with simulated annealing. Multimed Tools Appl 78:29921–29936. https://doi.org/10.1007/s11042-018-6602-4
35. Javidrad F, Nazari M (2017) A new hybrid particle swarm and simulated annealing stochastic optimization method. Appl Soft Comput J 60:634–654. https://doi.org/10.1016/j.asoc.2017.07.023
36. Rashedi E, Nezamabadi-pour H, Saryazdi S (2009) GSA: a gravitational search algorithm. Inf Sci (Ny) 179:2232–2248. https://doi.org/10.1016/j.ins.2009.03.004
37. Lennard-Jones JE (1924) On the determination of molecular fields. Proc Roy Soc A 106:463–477
38. Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing. Science 220(80):671–680
39. Holland JH (1992) Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press, Cambridge
40. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95-International Conference on Neural Networks. IEEE, pp 1942–1948
41. Sauer PW, Pai MA, Chow H. J (2017) Power system dynamics and stability: with synchrophasor measurement and power system toolbox, 2nd ed