



Bio-Inspired Optimization Algorithms Based Design of Robust Controller for Single Machine Power System Stabilizer

Manogna Bojugu¹, Satish Kumar Injeti¹(✉), and Dasu Butti²

¹ Electrical Engineering Department, NIT Warangal, Hanumakonda 506004, Telangana, India
drinjetisatishkumar@nitw.ac.in

² Faculty of Electrical and Electronics Engineering Department, SRGEC (A),
Gudlavaluru 521356, A.P., India

Abstract. A power system stabilizer (PSS) is a device that provides additional damping torque to the power system during the mitigation of oscillations caused by various sorts of disturbances. Bio-Inspired Optimization Algorithms are used to construct PSS on a Single Machine Infinite Bus (SMIB) system and two areas, respectively, in this research. The study examines four different multi-machine generator systems working under a variety of disturbances. The SMIB system uses ISE as an objective function, while the Multimachine system uses Eigen value shifting. Modified Heffron-Phillips (MHP) model is used in the SMIB system to lower the system complexity and processing time. The SMIB system's response time is improved by integrating a PID controller with MPSS. Particle swarm optimization (PSO), whale optimization (WO) and butterfly optimization (BO) techniques are used to find the best parameters for PSS and PID. A thorough evaluation of the MPSS, MPSS-PID, and other optimization algorithms' performance is offered based on the simulation findings.

Keywords: Bio-inspired Optimization · Power System Stabilizer · SMIB · PID controller · Dynamic Stability Improvement

1 Introduction

A variety of disruptions can occur in the power system, including faults, changes in torque, and a sudden shift in load, all of which can impact the network's power transmission capacity constraints and result in synchronism loss, system blackouts, and eventually system failure. Low-frequency oscillations (between 0.2 and 3.0 Hz) caused by these disturbances have a significant impact on the dynamic stability and performance of the system. Power System Stabilizer reduces these oscillations by providing the necessary damping torque.

In order to induce either +ve or -ve damping, the excitation control dV/dt is used in the research of PSS and excitation control in [1]. Robust Control [2] and Artificial Intelligence Techniques [3–5] have been used to tune PSS parameters, while the use of Eigen value drift as an objective function in Robust Control has made the tuning mechanism

flexible to changes in operating conditions. Complex and time-consuming approaches like this necessitate a substantial amount of computation time. As a result, heuristic algorithms such as Tabu Search [6], Simulated Annealing [7], Genetic Algorithms [8], Particle Swarm Optimization [9–11], Differential Evolution [12], Artificial Bee Colony [13–15] and Honey Bee Algorithm [16] are being considered in recent research to solve complex problems. As our power system is highly non-linear coupled and operates under a variety of operational situations, these approaches necessitate the use of an optimal PSS design strategy. PSO stands out as a strategy with the capacity to address a wide range of optimization issues. Modern optimization and Hybrid research on PSO and GA has been increasing in recent years [17–19, 26]. By using an infinite bus voltage, all of the strategies outlined above can be compared.

An HP model PSS design relies on data that may not be immediately available from the framework's respective generator. The PSS structure for such a system necessitates extensive knowledge of system parameters external to the generating station. If the system settings aren't correct, it may be impossible to do an accurate PSS configuration. Because consumer demand is constantly shifting, it's challenging to develop a PSS that can withstand these constantly shifting operational conditions (due to change in load demand). Such a wide range of operating conditions makes conventional PSS parameter design problematic, as complexity and computation time will increase. Researchers are inspired by this to develop a PSS design that does not rely on external system data. Additionally, PSS incorporates a PID controller to enhance the system's dynamic performance. The transformer secondary voltage of the generator is used instead of the infinite bus voltage in the Modified Heffron Phillip Model [20, 21] for the design of PSS parameters. This update allows PSS to be constructed in a way that takes into account local information and data, rather of relying solely on data from the external system. However, the PSS design process will become tedious and complex while working in a variety of environments. Thus, Modified Power System Stabilizer (MPSS) is developed to overcome the above drawback for the SMIB system. P. Kundur [22] had built up a precise strategy for PSS tuning and executed on Ontario Hydro station. To check PSS performance and robustness, the PSS designing technique is the extent to benchmark Multi-Machine system (Two Area-Four Generator system) [23]. Many authors reported great results on this particular engineering problem. From the literature, it has been observed that the solution to this problem is still evolving because of the latest methodologies and evolution of novel optimization algorithms availability. So, in this paper an attempt has been made to find a better solution through implementing efficient optimization algorithms such as PSO, WOA and BOA to optimal parameter selection of power system stabilizer in case of single as well as multi-machine two are power system.

2 System Modelling

2.1 Mathematical Model for MHP (for SMIB) Model

Modified Heffron-Phillips model is constructed in this study by using the transformer secondary bus voltage as the reference instead of infinite bus voltage shown in Fig. 1(a). SMIB system for MHP model is depicted in Fig. 1(a) using a single line diagram.

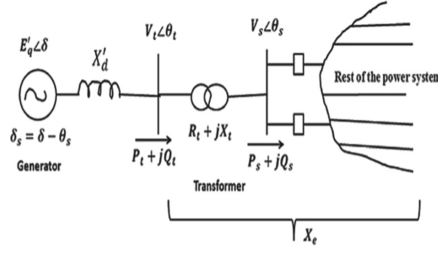


Fig. 1(a). SMIB system for MHP model

This Modified mode yields the G-constants shown below

$$G_1 = \frac{V_{so} E_{qo} \sin \delta_{so}}{X_q + X_t} + \frac{X_q - X'_d}{X_t + X'_d} V_{so} \sin \delta_{so} \quad (1)$$

$$G_2 = \frac{X_q + X_t}{X_t + X'_d} i_{qo} \quad (2)$$

$$G_3 = \frac{X_t + X'_d}{X_d + X_t} \quad (3)$$

$$G_4 = \frac{X_d - X'_d}{X_t + X'_d} V_{so} \sin \delta_{so} \quad (4)$$

$$G_5 = \frac{X_q V_{do} V_{so} \cos \delta_{so}}{(X_q + X_t) V_{to}} + \frac{X'_d V_{qo} V_{so} \sin \delta_{so}}{(X_t + X'_d) V_{to}} \quad (5)$$

$$G_6 = \frac{X_t}{X_t + X'_d} \frac{V_{qo}}{V_{to}} \quad (6)$$

$$G_{v1} = \frac{E_{qo} \sin \delta_{so}}{(X_q + X_t)} + \frac{(X_q - X'_d) I_{qo} \cos \delta_{so}}{(X_t + X'_d)} \quad (7)$$

$$G_{v2} = - \frac{(X_d - X'_d) \cos \delta_{so}}{(X_t + X'_d)} \quad (8)$$

$$G_{v3} = \frac{X_{qo} V_{do} \sin \delta_{so}}{(X_q + X_t)} + \frac{X'_d V_{qo} \cos \delta_{so}}{(X_t + X'_d) V_{to}} \quad (9)$$

We have derived the MHP model's G-constants above, which are similar to the HP model K-constants, except that they are assessed by taking as a reference at generator side the secondary side of the step-up transformer, rather than the infinite bus voltage. The MHP model's block diagram is shown in Fig. 1(b).

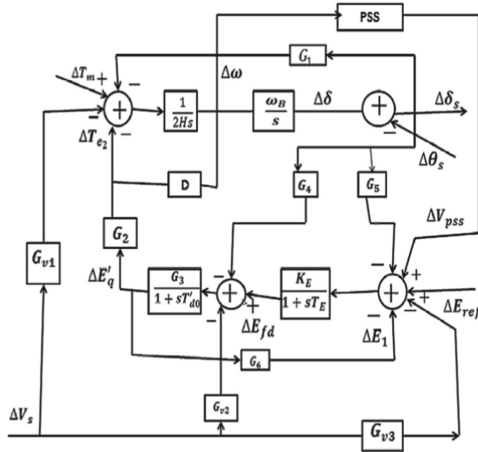


Fig. 1(b). MHP model's block diagram

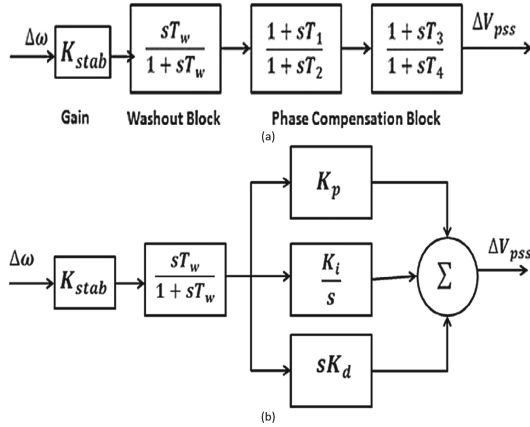


Fig. 2. (a) Structure of MPSS (b) Block diagram of PID-MPSS

2.2 Proposed PID-MPSS

A CSMIB system's parameters might vary under normal operating conditions, and these parameters are difficult to determine because they are not easily accessible. The P-I-D controllers are well-known for their ability to improve the dynamic performance of power systems. The SMIB system's dynamic performance can be improved by integrating MPSS with various algorithm-tuning based P-I-D controllers. The P-I-D-MPSS block diagram can be seen in Fig. 2(b). PSO, WOA, or PSO-PID-MPSS are used to determine the gain setting and parameter of MPSS in the next section, respectively.

3 Problem Formulation

3.1 SMIB System

Integral Square Error (ISE) is the most accurate and reliable objective function for assessing the dynamic performance of a power system. The following is an illustration of the performance index:

$$J = \int_0^t (\Delta\omega)^2 dt \quad (10)$$

where t is the simulation time and $\Delta\omega$ is the change in speed. Based on the optimization function J , the fitness function can be expressed as:

$$K_{pss}^{min} \leq K_{pss} \leq K_{pss}^{max} \quad (11)$$

$$K_p^{min} \leq K_p \leq K_p^{max} \quad (12)$$

$$K_i^{min} \leq K_i \leq K_i^{max} \quad (13)$$

$$K_d^{min} \leq K_d \leq K_d^{max} \quad (14)$$

3.2 Multi-machine System

The Eigenvalues of the test systems determine the goal function. It is here that an objective function is created in order to shift Eigenvalues into proper s-plane position. The oscillating behaviour of the system is attributed solely to the use of gently damped Eigenvalues in the construction of the objective function. Because the system's oscillatory behaviour can only be explained by gently damped Eigenvalues, these are taken into account while constructing the goal function. By using this objective function, only the desired poles are regarded to be relocated to their new places.

$$J = J_1 + c * J_2 \quad (15)$$

$$\text{Where, } J_1 = \sum_{j=1}^{Np} \sum_{\sigma_i \leq \sigma_o} (\sigma_o - \sigma_i)^2 \text{ \& } J_2 = \sum_{j=1}^{Np} \sum_{\zeta_i \leq \zeta_o} (\zeta_o - \zeta_i)^2 \quad (16)$$

Relative stability (σ_o) and damping ratio I are used in this example, where Np is the population size, I the i^{th} Eigenvalue of the population, and σ_o is set as 0.3. Eigen value is set to 0.15 as well. As shown in Fig. 3(a), if just J_1 is taken into account, eigenvalues will be located in the highlighted areas. If just J_2 is taken into account, the eigenvalues will be shifted to the area shown in Fig. 3(b). An objective function can be created by combining two single objective functions J_1 and J_2 by assigning them a weighting factor of $c = 10$. Instance No. 3(c).

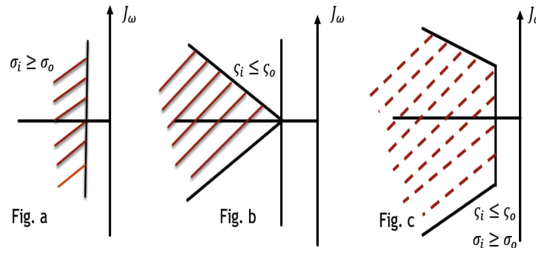


Fig. 3. Eigen control plane

4 Optimization Algorithms

The suggested objective function is optimised using PSO, WOA, and BO bio-inspired algorithms. For solving engineering optimization issues, it has been found that PSO (1995) and WOA (2016) well-proven optimization algorithms because of their advantages such ease of implementation and understanding. Using the most recent version of the BO optimization algorithm (2018), a comparison study was conducted to see if the PSO and WOA had provided a better solution than the others. No need to memorise the specific best placements of each agent, as the BO method has no memory requirements, is a major benefit. It is a newly created optimization algorithm with a new means of propagating information about an agent's fitness via fragrance.

4.1 Particle Swarm Optimization Algorithm

Based on the social comparison between fish and birds, Particle Swarm is a biologically-inspired programme. Within an initialization zone, the PSO algorithm generates random placements for the particles. The velocities of particles can be set to zero or too small random values in order to prevent them from leaving the search area during the first iterations of the search algorithm. This technique uses Eq. 17 and Eq. 18 to iteratively update the particle velocities and positions [24] during the main loop of the process.

$$\overline{V}_i^{k+1} = w * \overline{V}_i^k + C_1 R_1 (\overline{x}_i^{\text{best},k} - \overline{x}_i^k) + C_2 R_2 (\overline{G}_i^{\text{best},k} - \overline{x}_i^k) \quad (17)$$

$$\overline{x}_i^{k+1} = \overline{x}_i^k + \overline{V}_i^{k+1} \quad (18)$$

the i^{th} particle's velocity vector \overline{V}_i^k at the k^{th} iteration is given by, where each value of \overline{V}_i^k must lie within the range. A particle's current position vector is given by \overline{x}_i at each iteration. The best position vector for that particle up to the k^{th} iteration is given by $\overline{x}_i^{\text{best},k}$ and the best position vector for all particles up to that iteration is given by $(\overline{G}_i^{\text{best},k})$. w is the weighing function or inertia weight factor.

4.2 Whale Optimization Algorithm

As a group, most whales spend much of their time together. As the largest humpback whale in its kind, it's also one of the most unique. The term for this kind of hunting

is “bubble-net feeding.” Fish schools near the surface are a favourite hunting ground for humpback whales. This foraging is done by blowing distinct bubbles in a circle, as has been observed. In its hunting method, bubbles are related with upward spirals and double loops. A humpback whale dives to a depth of about 12 m before rising and swimming toward the surface in a meandering pattern around its food. Third, a catch circle is used in conjunction with all three of the previous manoeuvres. Spiral bubble-net feeding movement is theoretically simulated in order to do optimization [25].

In any optimization algorithm, initially, exploration takes place then exploitation. The Eq. 19 and Eq. 20 represent the exploration stage of WOA.

$$\bar{D} = |\bar{C} \cdot \bar{X}_{\text{rand}} - \bar{X}(t)| \quad (19)$$

$$\bar{X}(t+1) = \bar{X}_{\text{rand}} - \bar{A} \cdot \bar{D} \quad \text{if } |\bar{A}| > 1 \quad (20)$$

$$\bar{A} = 2\bar{a} \cdot \bar{r} - \bar{a} \quad (21)$$

$$\bar{C} = 2\bar{r} \quad (22)$$

Coefficient vectors \bar{A} and \bar{C} are defined. In both the exploration and exploitation stages, \bar{a} is linearly decreased from 2 to 0 while \bar{r} is a random vector in $[0, 1]$. Experimentation begins in Eq. 20 when the vector $|\bar{A}| > 1$ is used, and it ends when the vector $|\bar{A}| < 1$ is used. Models for two exploitation strategies are shown in this way.

$$\bar{X}(t+1) = \begin{cases} \bar{X}^*(it) - \bar{A} \cdot \bar{D} & \text{if } p < 0.5 \\ \bar{D} \cdot e^{bl} \cdot \cos(2\pi l) + \bar{X}^*(it) & \text{if } p > 0.5 \end{cases} \quad (23)$$

$x^*(t) - x^*(t)$, where $||$ denotes the absolute value, is the position vector we're interested in; (\cdot) represents an element-by-element multiplication; The distance between the prey and the whales is $(D') |X - (t) - X - (t)|$ and indicates that the best solution so far, b is the constant for shaping the logarithmic spiral, and the random integer in [1] is 1. X^* should be modified in each iteration if there is a better option. Using Eq. 23, we can see that when p is less than or equal to 0, we can use a shrinking encircling method of exploitation or a spiral updating mechanism of exploitation. The WOA will come to an end once the necessary conditions have been met.

4.3 Butterfly Optimization Algorithm

Developed by Sankalp Arora and Satvir Singh in 2018, the butterfly optimization (BO) technique is a new meta-heuristic optimization algorithm. In order to create the BO algorithm, we take into account the butterfly's food-gathering and mating habits. Butterflies rely on their keen sense of smell to locate food and a mate in the wild. Each butterfly emits a distinct scent as it searches for food, and the strength of the scent is a direct reflection of the quality and amount of food available in the immediate area. The butterfly's scent will go for a long way. As soon as the scent was detectable, additional butterflies would fly towards it. To get to a suitable food supply, butterflies will have to travel around in the real world in this manner.

Every butterfly in the BO algorithm is viewed as a search agent. Each agent is assigned a role and a distinct scent to go with it. As each agent's ability to carry out its objective function improves, so does its scent. Equation 24 provides a mathematical model for the aroma.

$$f = cI^a \quad (24)$$

f is the perceived scent magnitude, I is the stimulus intensity, c is the sensor mode, and a is the power exponent. The search agent's or butterfly's fitness is represented by the letter I in the BO algorithm. BO algorithm control parameters c and a were analysed in detail in [27] and are referred to in this article. If the best agent is chosen, all agents will migrate to their new locations based on a switch probability " p " and the magnitudes of all their perfumes, with the exception of one. P , the likelihood of switching between local and global search options, determines which path the agent will take. Below are the formulae for updating your position.

Perform a global search using Eq. 25 if $rand < P$

$$x_i^d(t+1) = x_i^d(t) + (r^2 * gbest - x_i^d(t)) * f_i \quad (25)$$

or local search using Eq. 26 if $rand > P$

$$x_i^d(t+1) = x_i^d(t) + (r^2 * x_j^d(t) - x_k^d(t)) * f_i \quad (26)$$

where $x_j^d(t)$ and $x_k^d(t)$ are Butterflies from the same swarm in the solution space as J^{th} and K^{th} and a random number in the range of $[0, 1]$.

PSO, WOA, and BO algorithms can be implemented using the following detailed procedures.

The first step is to set up the problem and algorithm settings.

Prior to implementing the algorithm, a number of parameters must be set up, including population size, dimension, maximum number of iterations (*itermax*), and acceleration constants $c1$, $c2$, as well as the probability switch P , power exponent a , and sensor modality for PSO, WOA, A, and C and BO algorithms, respectively Set the upper and lower bounds of variables to their default values (MPSS or PID parameters).

Step 2: Random generation of PID gains

$$X = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_{d-1}^1 & x_d^1 \\ x_1^2 & x_2^2 & \cdots & x_{d-1}^2 & x_d^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_1^{pop-1} & x_2^{pop-1} & \cdots & x_{d-1}^{pop-1} & x_d^{pop-1} \\ x_1^{pop} & x_2^{pop} & \cdots & x_{d-1}^{pop} & x_d^{pop} \end{bmatrix} \quad (27)$$

$$x_i^j = x_{min,i} + (x_{max} - x_{min,i}) * rand() \quad (28)$$

PID gains, i.e., the j^{th} population of the i^{th} parameter, are created randomly between the limits as $x(\max, i)$ and $x(\min, i)$ are the i^{th} parameter limits, and $\text{rand}()$ is a random number in between 0 and 1. d is the number of decision variables.

$$\text{Soln} = [K_p, K_i, K_D \dots] \quad (29)$$

Soln symbolises a swarm of particles in the PSO algorithm. In each particle, there is a PID gain. Soln is a collection of search agents in WOA, BO.

Step 3: Fitness evaluation (Objective function).

Equation 10 and Eq. 15 should be used to calculate the fitness value for each initial solution, and the g_{best} solution should be recorded in the case of BO, X should be recorded in the case of the WOA algorithm,

Step 4: Set iteration count = 0.

Step 5: During this step, PSO, WOA, and BO algorithms begin their evolutionary process. Update the number of iterations by one.

Step 6: Use Eq. 17 to update particle velocities, and then Eq. 18 to update particle position for the PSO algorithm.

In WOA algorithm update \bar{X} using Eq. 23.

Figure 4 depicts a thorough flow chart for implementing optimization methods.

For BO algorithm, calculate the fragrance f_N for each agent or butterfly using Eq. 24 and then perform a global search and local search as follows.

If $\text{rand} < \text{probability } P$ perform global search using Eq. 30.

$$\text{soln}_N^d(t+1) = \text{soln}_N^d(t) + \left(r^2 * g_{\text{best}} - \text{soln}_N^d(t) \right) * f_N \quad (30)$$

If $\text{rand} > \text{probability } P$ perform a local search using Eq. 31

$$\text{soln}_N^d(t+1) = \text{soln}_N^d(t) + \left(r^2 * \text{soln}_j^d(t) - \text{soln}_k^d(t) \right) * f_N \quad (31)$$

where $\text{soln}_j^d(t)$ and $\text{soln}_k^d(t)$ are A random value between $[0, 1]$ is chosen as the r^{th} element of the solution space and used as the initialization vector.

Step 7: Fitness evaluation (Objective function).

Each initial solution's fitness value is calculated using Eq. 10 and Eq. 15; the best solution is recorded for each solution in BO and X in WOA.algorithm, $\bar{x}_i^{\text{best},k}$, $\bar{G}_i^{\text{best},k}$ for the PSO algorithm.

Step 8: Stopping criterion.

The computation is finished and the results are printed if the iteration count surpasses the specified maximum. Steps 7 through 10 must be repeated if this is not the case.

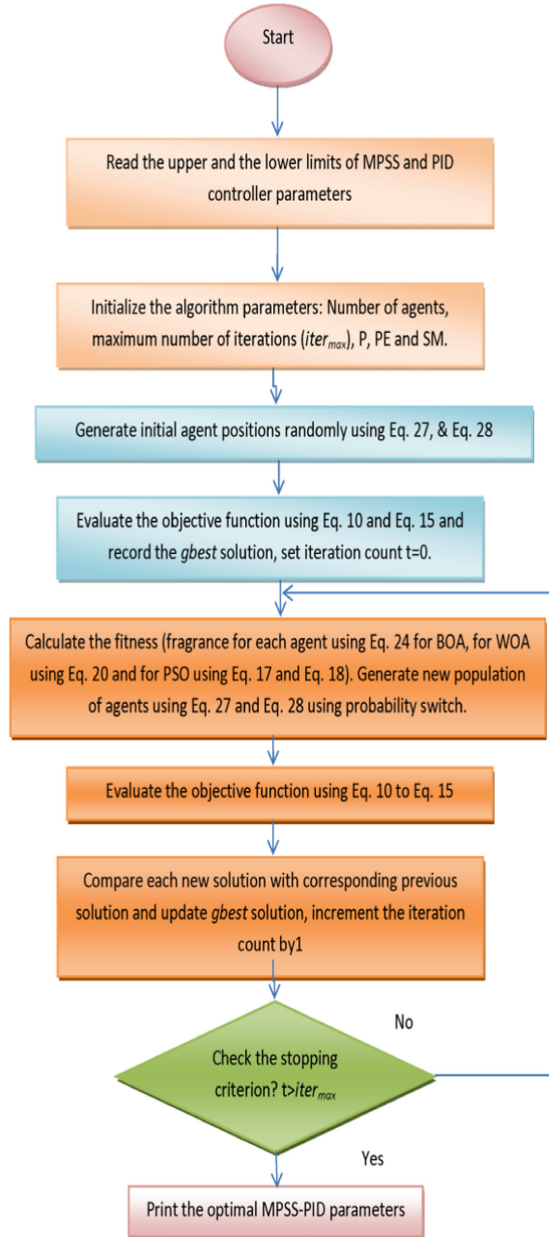


Fig. 4. Flow chart for implementing optimization methods.

5 Simulation Results and Discussion

At four different loading circumstances, PSS is tested in two cases (i.e. two different generator parameters) for the SMIB system: higher loading, nominal loading, weak

loading and lead power factor loading. For each of the two scenarios, detailed system information is provided in Appendix A. The proposed methods were tested on the MHP model in the MATLAB environment to see how well they worked. All the proposed algorithms were run for several times with Integral Square Error (ISE) was used as an objective function to test each algorithm. The suggested stabilizer's efficacy and robustness are examined using two disturbances, which are a 10% step change in T_m and a 10% step change in V_{ref} for all operational points in each example.

5.1 SMIB Case-1

The MPSS system's performance is tested under four different operating situations using the SMIB system, which uses G-constants that have been adjusted. Integral Square Error (ISE) serves as the objective function for the PSO, WOA, and BOA optimization algorithms. The usage of a stabiliser in a classical manner can reduce calculation time and increase dynamic performance. Appendix A contains the generator's SMIB system data parameters. The dynamic performance of the system has been improved by utilising a PID controller. Gain settings, i.e., K_p , K_i , and K_d , for PID controllers are optimised using presented techniques. Table 1 lists the ranges of various PSS parameters. Table 2 is a list of the various operating points. Two types of disturbances are looked for in the system: a 10% step change at T_m and a 10% step change at V_{ref} .

The simulation results for PSO, WOA & BOA Algorithms for four operating points for disturbances of 10% step change at ΔT_m for SMIB with MPSS and MPSS-PID

Table 1. Range of various PSS parameters (SMIB)

S. No.	Range for Operating Points			
	1	2	3	4
K_{pss}	1 & 50	1 & 50	1 & 50	1 & 50
$T_1, T_2,$ T_3, T_4	0.001 & 1	0.001 & 1	0.001 & 1	0.001 & 1
K_P	0.1 & 12	0.1 & 12	0.1 & 12	0.1 & 12
K_I	0.1 & 12	0.1 & 12	0.1 & 12	0.1 & 12
K_d	0.01 & 5	0.01 & 5	0.01 & 5	0.01 & 5

Table 2. SMIB Case-1 Operating conditions

Operating Points	X_e	P_t (p.u.)	Q_t (p.u.)
1	0.3-Higher Loading	1.1	0.5
2	0.4-Nominal Loading	0.8	0.41
3	0.8-Weak Loading	0.4	0.1
4	0.4-Lead p.f. Loading	0.7	-0.2

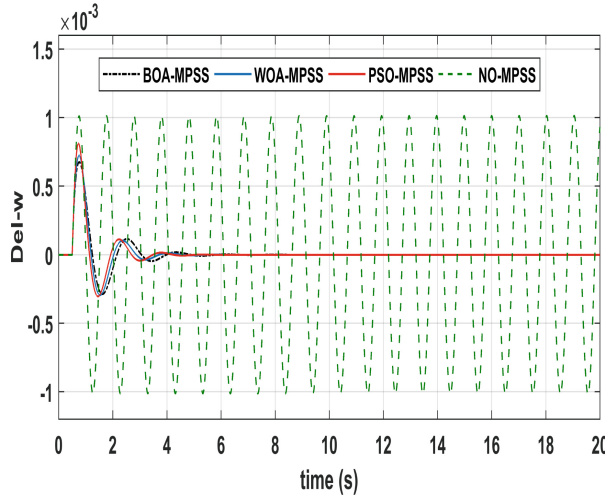


Fig. 5. Speed deviation for 10% step change at T_{ref} for MPSS-SMIB Case-1 at Operating Point 1

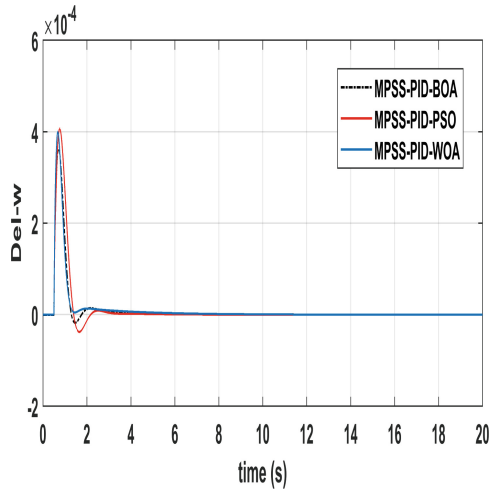


Fig. 6. Speed deviation for 10% step change at T_{ref} for MPSS-PID-SMIB Case-1 at Operating Point 1

are shown in Fig. 5 and Fig. 6 and disturbances of 10% step change ΔV_{ref} for SMIB with MPSS and MPSS-PID are shown in Fig. 7 and Fig. 8 respectively for operating condition 1. In each figure is having four curves namely: No MPSS (green), MPSS-PSO (red), MPSS-WOA (blue) and MPSS-BOA (black). The gain settings are determined by considering Integral Square Error (ISE) as an Objective Function. An algorithm has been run for 50 iterations and optimised parameters for MPSS-PSO, MPSS-WOA and MPSS-BOA with and without tuned PID Controller are given in Table 3 and Table 4.

Table 3. Optimized parameters of MPSS with PSO, WOA and BOA for SMIB Case-1

Parameters	PSO-MPSS				WOA-MPSS				BOA-MPSS			
	Operating Points											
	1	2	3	4	1	2	3	4	1	2	3	4
K_{pss}	10.234	10.354	10.014	10.128	10.098	10.074	10.014	10.129	10.154	9.9954	10.032	10.945
T_1	0.9865	0.9025	0.9925	0.9854	0.9465	0.9654	0.0100	0.9965	0.978	0.9632	0.9547	0.9832
T_2	0.393	0.0257	0.4249	0.2681	0.5222	0.3981	0.424	0.4468	0.1220	0.3981	0.1541	0.4468
T_3	0.9865	0.0063	0.9925	0.9854	0.6105	0.9654	0.9951	0.9965	0.978	0.9632	0.9547	0.9832
T_4	0.393	0.4622	0.4249	0.268	0.3713	0.3981	0.424	0.4468	0.1220	0.3981	0.1541	0.4468

Table 4. Optimized parameters of PID-MPSS with PSO, WOA and BOA for SMIB Case-1

Operating Points	PSO-PID-MPSS				WOA-PID-MPSS				BOA-PID-MPSS			
	Optimization Parameters											
	K_{pss}	K_p	K_i	K_d	K_{pss}	K_p	K_i	K_d	K_{pss}	K_p	K_i	K_d
1	20.354	4.654	8.114	0.105	20.095	5.095	9.0354	0.198	8.824	11.098	8.9325	0.832
2	19.654	4.001	7.981	0.1032	8.0147	10.62	8.354	0.852	5.025	5.0251	9.9654	2.2798
3	20.065	3.987	8.147	0.0541	20.659	5.032	9.521	0.0431	8.154	10.692	9.1467	0.9903
4	18.475	5.258	8.394	0.325	10.521	4.958	3.324	1.085	9.996	4.8254	3.325	0.9460

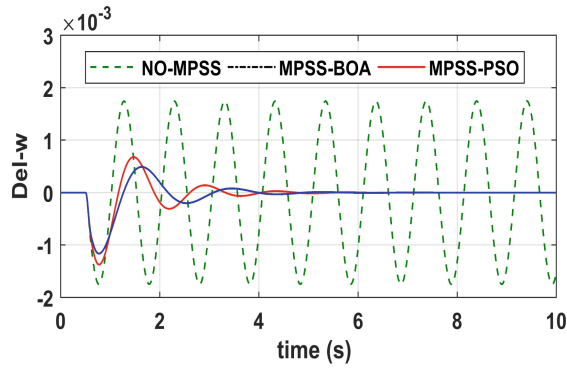
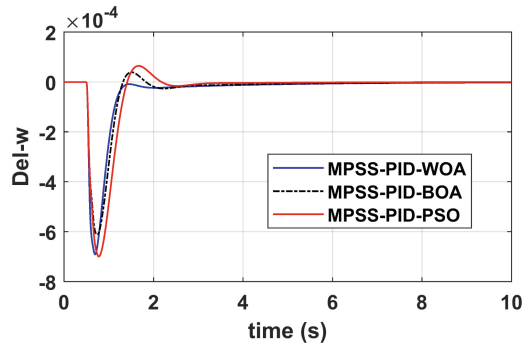
Table 5. Time Response Specification for SMIB Case-1 using PSO, WOA and BOA Algorithms

Operating Points	PSO					
	With MPSS			With PID-MPSS		
	Settling Time (t_s)	Peak Time (t_p)	Rise Time (t_r)	Settling Time (t_s)	Peak Time (t_p)	Rise Time (t_r)
1	4.4543	0.7600	0.0000000202	3.5022	0.6800	0.0000000245
2	6.1625	0.7900	0.0000001846	3.3695	0.7000	0.0000033627
3	9.364	0.8800	0.0000003727	3.3400	0.7700	0.0000459879
4	4.6780	0.8000	0.0000000497	3.8400	0.7300	0.0000126464
	WOA					
1	4.5399	0.7500	0.0000000174	3.1451	0.6700	0.0000000184
2	5.4530	0.8000	0.0000000467	3.2834	0.6900	0.0000643617
3	8.1825	0.9000	0.0000236033	3.1254	0.7700	0.0001250593
4	4.5326	0.8000	0.0000000632	3.5124	0.7400	0.0000000787
	BOA					
1	3.3344	0.7000	0.0000012152	1.2522	0.6400	0.0002973060

(continued)

Table 5. (continued)

Operating Points	PSO					
	With MPSS			With PID-MPSS		
	Settling Time (t_s)	Peak Time (t_p)	Rise Time (t_r)	Settling Time (t_s)	Peak Time (t_p)	Rise Time (t_r)
2	2.9519	0.7000	0.0000094311	1.1904	0.6300	0.0002336126
3	2.6175	0.7000	0.0000052016	1.7873	0.6700	0.0006541999
4	4.3412	0.765	0.0001165013	1.6291	0.6300	0.0000587452


Fig. 7. Speed deviation for 10% step change at V_{ref} for MPSS-SMIB Case-1 at Operating Point 1

Fig. 8. Speed deviation for 10% step change at V_{ref} for MPSS-PID-SMIB Case-1 at Operating Point 1

Values presented in Table 5 shows the time response specifications i.e., rise time (t_r), peak time (t_p) & settling time (t_s) for MPSS with and without PID controller tuned using three bio-inspired algorithms such as PSO-PID, WOA-PID and BOA-PID. It has been observed that system without PSS becomes unstable for variable operating points

with introducing forced disturbances and when the fixed gain stabilizer is introduced, the system becomes stable. For Higher Loading, the Settling time for MPSS-PSO is improved from 4.4543 s to 3.5022 s with the use of PID Controller. For MPSS-WOA, settling time is improved from 5.4530 s to 3.2834 s and for MPSS-BOA it is improved from 3.3344 s to 1.2522 s. Peak time and Rise time also gets improved and the same follows with other Loading conditions (Operating Points). Figure 9 and Fig. 10 shows the comparison of convergence characteristics.

Table 6. SMIB Case-2 Operating conditions

Operating Points	X_e	P_t (p.u.)	Q_t (p.u.)
1	0.3-Higher Loading	1.0	0.2
2	0.4-Nominal Loading	0.8	0.41
3	0.8-Weak Loading	1.0	0.5
4	0.4-Lead p.f. Loading	1.0	-0.5

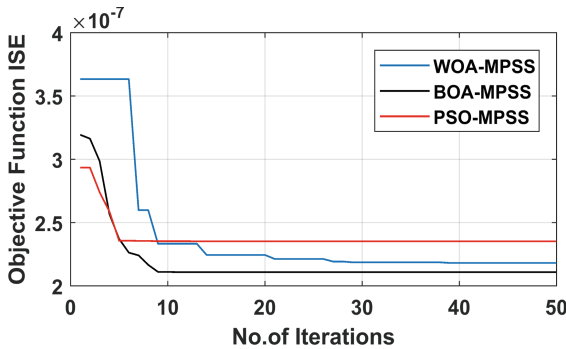


Fig. 9. Convergence Plots for operating condition 1 of MPSS-SMIB Case-1 with PSO, WOA and BOA at T_{ref}

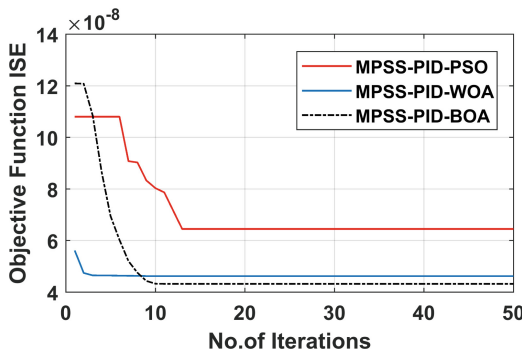


Fig. 10. Convergence Plots for operating condition 1 of MPSS-PID-SMIB Case-1 with PSO, WOA and BOA

5.2 SMIB Case-2

The machine data and exciter data for this case is given Appendix A. In this case, the performance of proposed stabilizer is connected with PID controller and is tuned with PSO, WOA & BOA optimization algorithms. The Operating point for this case is listed in Table 6. After several iterations, optimised parameters are given in Table 7 and Table 8. Table 7 depicts the parameters for MPSS tuned with algorithms and Table 8 gives the parameters for MPSS tuned with PID Controller. The gain settings are determined by considering Integral Square Error (ISE) as an Objective Function.

In this case also the performance of Algorithms tested on four operating conditions under the two disturbances i.e., 10% step change at ΔT_m and 10% step change at ΔV_{ref} . The simulation results for PSO, WOA & BOA Algorithm for disturbances of 10% step change at ΔT_m are shown in Fig. 11 and Fig. 12 and disturbances of 10% step change ΔV_{ref} are shown in Fig. 13 and Fig. 14.

The results clearly show that speed deviation (oscillations) is much more reduced with the PID controller with proposed algorithms and settled faster compared to the conventional approach. Table 9 given below is for time response specifications for MPSS tuned PSO, WOA and BOA Algorithms with & without PID Controller. Rise time (t_r), Peak time (t_p) & Settling time (t_s) are compared.

In this case also, without MPSS system becomes unstable for all operating points. And with MPSS system becomes stable. For 1st operating condition, settling time for PSO-MPSS is improved from 4.4543 s to 2.0062 s with PID controller tuned PSO i.e., MPSS-PID-PSO. For WOA, settling time is shifted from 3.9005 s (MPSS-WOA) to 1.6692 s (MPSS-PID-WOA). For BOA, settling time is shifted from 3.9005 s (MPSS-BOA) to 1.6691 s (MPSS-PID-BOA).

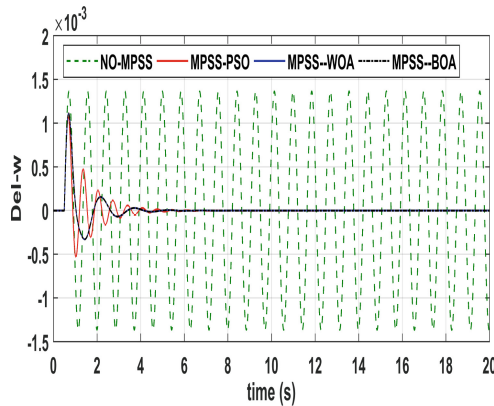


Fig. 11. Speed deviation for 10% step change at T_{ref} for MPSS-SMIB Case-2 at Operating Point 1

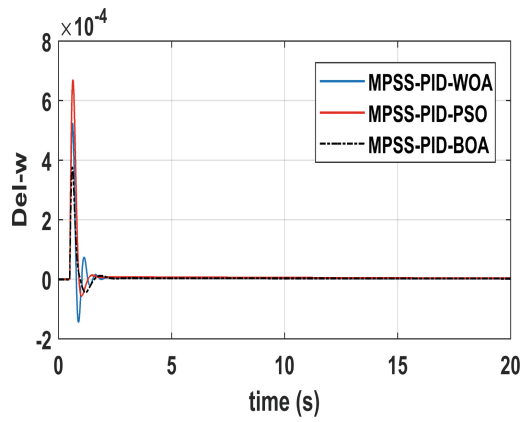


Fig. 12. Speed deviation for 10% step change at T_{ref} for MPSS-PID-SMIB Case-2

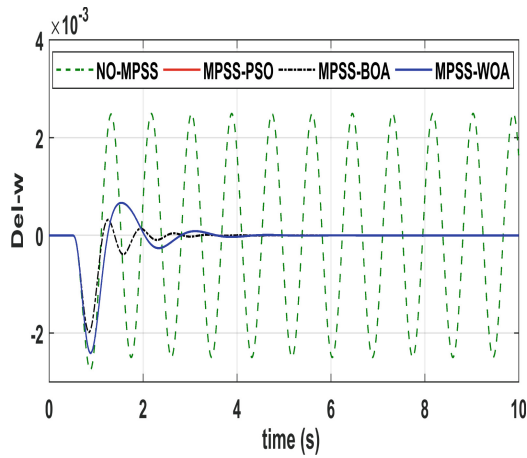


Fig. 13. Speed deviation for 10% step change at V_{ref} for MPSS-SMIB Case-2 at Operating Point 1

Table 7. Optimized parameters of MPSS with PSO, WOA and BOA for SMIB Case-2

Parameters	PSO-MPSS				WOA-MPSS				BOA-MPSS			
	Operating Points											
	1	2	3	4	1	2	3	4	1	2	3	4
K_{pss}	2.354	2.246	6.241	10.08	40.21	39.45	35.73	40.025	2.032	2.032	6.52	10.214
T_1	0.965	1.012	0.998	0.996	0.0198	0.895	1.001	0.1100	0.967	1.012	1.000	1.000
T_2	0.176	0.207	0.315	0.347	0.0135	0.599	0.738	0.0284	0.1760	0.2272	0.361	0.3479
T_3	0.965	1.012	0.998	0.996	0.0135	0.895	1.001	0.120	0.967	1.012	1.00	1.00
T_4	0.176	0.207	0.315	0.347	0.0135	0.895	0.468	0.0284	0.1760	0.2272	0.361	0.3479

Table 8. Optimized parameters of PID-MPSS with PSO, WOA and BOA for SMIB Case-2

Parameters	PSO-PID-MPSS				WOA-PID-MPSS				BOA-PID-MPSS			
	Optimization Parameters											
	K_{pss}	K_p	K_i	K_d	K_{pss}	K_p	K_i	K_d	K_{pss}	K_p	K_i	K_d
Operating Point 1	40.021	2.821	11.832	0.0054	40.321	2.594	11.521	0.5004	40.021	2.821	8.1134	0.200
Operating Point 2	5.041	5.00	10.54	2.41	40.21	2.510	7.00	0.200	5.014	5.21	10.33	2.012
Operating Point 3	6.0352	5.0014	8.064	2.0014	25.0012	3.625	0.5012	0.1501	6.021	8.002	10.021	0.200
Operating Point 4	6.512	5.014	8.654	2.014	40.021	2.001	14.52	0.151	6.021	10.35	10.120	2.001

Table 9. Time Response Specification for SMIB Case-2 using PSO, WOA and BOA Algorithms

Operating Points	PSO					
	With MPSS			With PID-MPSS		
	Settling Time (t_s)	Peak Time (t_p)	Rise Time (t_r)	Settling Time (t_s)	Peak Time (t_p)	Rise Time (t_r)
1	4.4543	0.6800	0.0000095464	2.0062	0.6400	0.0002281065
2	4.4056	0.7300	0.0000005871	3.2834	0.6900	0.0000643617
3	4.9278	0.7500	0.0000017379	2.4815	0.7100	0.0000636908
4	4.9946	0.7000	0.0000055255	3.1115	0.6400	0.0000508992
	WOA					
1	3.9005	0.7000	0.0000012152	1.6692	0.6200	0.0002258219
2	4.4046	0.7300	0.0000005868	2.6078	0.6700	0.0000744554
3	4.9115	0.7500	0.0000017379	1.8783	0.6900	0.0000042916
4	4.6073	0.7200	0.0000046539	2.6693	0.6400	0.0001855305
	BOA					
1	3.9005	0.7000	0.0000012541	1.2520	0.6400	0.0002973060
2	4.5846	0.7200	0.0000094311	1.1904	0.6300	0.0002336126
3	4.6715	0.7120	0.0000065201	1.7873	0.6700	0.0000465199
4	4.5539	0.7650	0.0000165013	1.6291	0.6300	0.0001187452

Also, it can be observed that peak time is reduced thereby improving the stability of the system by using a PID controller. Rise time also increased that implies improved response of the system. Settling time, Peak time & Rise time improves in each case following all other operating conditions.

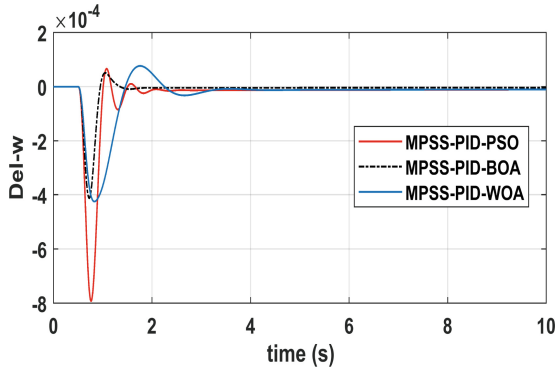


Fig. 14. Speed deviation for 10% step change at V_{ref} for MPSS-PID-SMIB Case-2 at Operating Point 1

6 Conclusion

Two different generating parameters (i.e., generator parameters) for the SMIB system are tested in this research to examine the system's effectiveness and efficiency. To enhance the power system's dynamic stability, this research proposes the use of PID-MPSS, an optimization algorithm based on three bio-inspired bio-inspirational algorithms. The MPSS on the MHP model is constructed by taking the secondary bus voltage from the generator side transformer rather than the infinite bus voltage reference. Adopting this approach has the benefit of removing the reliance on external system data for local knowledge on the generator side. The PID controller is used to find the best settings for MPSS utilising the proposed algorithms. Also, this test system is tested in a variety of operational circumstances to ensure that the suggested approach works as expected. Using simulation data, it can be determined that MPSS-PID outperforms other approaches in terms of peak and settling time in every situation. BOA, however, is the best solution for the MHP (SMIB) model since it requires the fewest number of parameter tuning rounds and has the lowest error rate (ISE).

This paper proposes a PSS design method based on three bio-inspired optimization algorithms for an interconnected power system. Three-phase fault at $t = 10$ s has been successfully realised using the design technique on the aforementioned case studies. BOA-PSS gives greater damping performance than PSO and WOA for all generating scenarios. In comparison to the other two algorithms, the BOA's Peak Overshoot, Settling Time, and Peak Time have all been enhanced.

Acknowledgments. Authors are thankful to the authorities of the NIT Warangal.

Authors' Contributions. All authors contributed equally.

Appendix A

Case-I Machine Data

$X_d = 0.1026$; $X_q = 0.658$; $= 5.67$; $H = 8$; $D = 0$; $f_B = 60$ Hz; $K_e = 400$; $T_e = 0.02$ s;
 $E_{fdmax} = 6$ p.u.; $E_{fdmin} = 6$ p.u;

Case-II Machine Data

$X_d = 1.6$; $X_q = 1.55$; $= 6$; $H = 5$; $D = 0$; $f_B = 60$ Hz; $E_B = 1$ p.u.; $K_e = 200$; $T_e = 0.05$ s;
 $E_{fdmax} = 6$ p.u.; $E_{fdmin} = 6$ p.u;

References

- Schleif, F.R., Hunkins, H.D.: Excitation control to improve powerline **2**(6), 1426–1434 (1968)
- Gupta, R.: Robust Nondynamic Multirate Output Feedback Technique based Power System Stabilizers (2003)
- Ghoshal, S.P., Chatterjee, A., Mukherjee, V.: Expert systems with applications bio-inspired fuzzy logic-based tuning of power system stabilizer. *Exp. Syst. Appl.* **36**(5), 9281–9292 (2009)
- Chaturvedi, D.K., Malik, O.P., Fellow, L.: Neurofuzzy power system stabilizer **23**(3), 887–894 (2008)
- Khalil, A.A., Ahmied, N.M.: Optimal tuning of lead-lag and fuzzy logic power system stabilizers using particle swarm optimization. *Exp. Syst. Appl.* **36**(2), 2097–2106 (2009)
- Abido, M.A.: A novel approach to conventional power system stabilizer design using tabu search. *Int. J. Electr. Power Energy Syst.* **21**(6), 443–454 (1999)
- Abido, M.A.: Robust design of multimachine power system stabilizers using simulated annealing. *IEEE Trans. Energy Convers.* **15**(3), 297–304 (2000)
- Kim, D.H.: Hybrid GA–BF based intelligent PID controller tuning for AVR system **11**, 11–22 (2011)
- Mostafa, H.E., El-sharkawy, M.A., Emary, A.A., Yassin, K.: Electrical power and energy systems design and allocation of power system stabilizers using the particle swarm optimization technique for an interconnected power system. *Int. J. Electr. Power Energy Syst.* **34**(1), 57–65 (2012)
- Olivas, F., Amador-Angulo, L., Perez, J., Caraveo, C., Valdez, F., Castillo, O.: Comparative study of type-2 fuzzy particle swarm, Bee Colony and Bat Algorithms in optimization of fuzzy controllers (2017)
- Abido, M.A.: Optimal design of power-system stabilizers using particle swarm optimization **17**(3), 406–413 (2002)
- Wang, Z., Chung, C.Y., Wong, K.P., Tse, C.T.: Robust power system stabiliser design under multi-operating conditions using differential evolution **2**(5), 690–700 (2008)
- Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, 459–471 (2007)
- Karaboga, D., Basturk, B.: On the performance of artificial bee colony (ABC) algorithm. *Appl. Soft Comput.* **8**, 687–697 (2008)
- Mohammadi, M., Ghadimi, N.: Optimal location and optimized parameters for robust power system, 1–17 (2014)

16. Elazim, S.M.A., Ali, E.S.: Electrical power and energy systems optimal power system stabilizers design via Cuckoo Search algorithm. *Int. J. Electr. Power Energy Syst.* **75**, 99–107 (2016)
17. Shakarami, M.R., Davoudkhani, I.F.: Wide-area power system stabilizer design based on Grey Wolf Optimization algorithm considering the time delay. *Electr. Power Syst. Res.* **133**, 149–159 (2016)
18. Islam, N.N., Hannan, M.A., Shareef, H., Mohamed, A.: An application of backtracking search algorithm in designing power system stabilizers for large multi-machine system. *Neurocomputing* **237**, 175–184 (2016)
19. Abdulkhader, H.K., Jacob, J., Mathew, A.T.: Fractional-order lead-lag compensator-based multi-band power system stabiliser design using a hybrid dynamic GA-PSO algorithm (2018)
20. Gurralla, G., Sen, I.: A modified Heffron-Phillip's model for the design of power system stabilizers (2008)
21. Butti, D., Kumar, S., Rao, S.: Design of robust modified power system stabilizer for dynamic stability improvement using particle swarm optimization technique. *Ain Shams Eng. J.* **10** (2019)
22. Kundur, P., Klein, M., Rogers, G.J., Zywno, M.S.: Application of power system stabilizers for enhancement of overall system stability. *IEEE Trans. Power Syst.* **4**(2), 614–626 (1989)
23. Dasu, B., Sivakumar, M., Srinivasarao, R.: Interconnected multi-machine power system stabilizer design using whale optimization algorithm. *Prot. Control Mod. Power Syst.* **4**, 2 (2019). <https://doi.org/10.1186/s41601-019-0116-6>
24. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: *Proceedings of the IEEE International Conference on Neural Networks, ICNN'95*, vol. 11, no. 1, pp. 111–117 (1995)
25. Mirjalili, S., Lewis, A.: The Whale optimization algorithm. *Adv. Eng. Softw.* **95**, 51–67 (2016)
26. Arora, S., Singh, S.: Butterfly optimization algorithm: a novel approach for global optimization. *Soft. Comput.* **23**(3), 715–734 (2018). <https://doi.org/10.1007/s00500-018-3102-4>
27. Injeti, S.K., Thunuguntla, V.K.: Optimal integration of DGs into radial distribution network in the presence of plug-in electric vehicles to minimize daily active power losses and to improve the voltage profile of the system using bio-inspired optimization algorithms. *Prot. Control Mod. Power Syst.* **5**(1) (2020)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

