A New Modified Shuffled Frog Leaping Algorithm for Optimal Design of Damping Controllers

Hamed Akbari, Mahdiyeh Eslami
Department of Engineering, Kerman Branch, Islamic Azad University, Kerman, Iran

Abstract
This paper presents a simultaneous coordinated tuning of damping controllers by using a modified shuffled frog leaping algorithm to damp power system low frequency oscillations. The control strategy is to choose the best controller parameters in such a manner that the lightly damped and un-damped electromechanical modes of all machines are shifted to the left-hand side of s-plane as far as possible. To illustrate the capability of the proposed approach, the numerical results are presented on a 5-area 16-machine system while one power system stabilizer (PSS) and one supplementary controller are designed simultaneously for a generator and Static Var Compensator (SVC), respectively. To show the feasibility of the designed controllers, the study power system is tested under a three phase fault at a bus. The simulation studies show that the designed controllers by proposed MSFL perform better in comparison to modified SFL.

Keywords- Shuffled frog leaping algorithm, Low frequency oscillation, PSS, SVC

1. INTRODUCTION

Electromechanical oscillations are inherent phenomena in electric power systems. With the development of extensive power systems, especially with the interconnection of these systems by weak tie-lines, electromechanical oscillations restrict the steady-state power transfer limits and affect operational system economics and security. Therefore, they have become one of the major problems in the power system stability area and have received a great deal of attention. Over the last three decades, there has been extensive research on the stabilization of electromechanical oscillations to enhance system small-signal stability by designing supplemental damping controllers.

To enhance system damping, the generators are equipped with power system stabilizers (PSSs) that provide supplementary feedback stabilizing signals in the excitation systems. PSSs augment the power system stability limit and extend the power-transfer capability by enhancing the system damping of low-frequency oscillations in the order of 0.2 to 2.5 Hz [1], [2]. Furthermore, it is well known that the supplementary controllers for Flexible AC Transmission System (FACT) devices are efficient tools for improving the stability of power systems through damping of low frequency modes.

In recent years, several approaches based on modern control theory have been applied to the PSS design problem including; pole-placement, optimal control, adaptive control, variable structure control [3]-[8]. Despite the potential of different control techniques with different structures, fixed-structure lead/lag type of controllers is widely used in power utilities because of its simple structure and ease of physical realization [9].

Recently, there has been a growing interest in algorithms inspired from the observation of natural phenomenon to seek the optimal design of PSS in a power system by the researches around the world [10]-[20]. Besides, different control efforts are reported for controlling the FACTS devices including modern control techniques as well as intelligent control [21]-[24].

Since there are some shortcomings with the SFL, a modified SFL (MSFL) algorithm is used in this paper to design coordinated damping controllers in a power systems, where the design problem is formulated as a multi-objective optimization problem with the weighted-sum approach. To show the feasibility and effectiveness of the proposed method, numerical results are presented on 5-area 16-machine system by designing a PSS for a generator and a supplementary controller for the SVC.

The paper is organized as follows: to make a proper background, the basic concept of the SFLA is briefly explained in section II. The modified SFLA with a new frog leaping rule is presented in section III. In section IV, the optimization problem is formulated. The results of the SFLA and MSFL in the study systems are given in section V. Finally, section VI concludes paper.
2. **OVER VIEW of SFLA**

The SFL algorithm is a memetic meta-heuristic method that is derived from a virtual population of frogs in which individual frogs represent a set of possible solutions. Each frog is distributed to a different subset of the whole population described as memeplexes.

The SFLA begins with an initial population of “P” frogs \( F=\{X_1,X_2,\ldots,X_n\} \) created randomly within the feasible space \( \Omega \). For \( S \)-dimensional problems (\( S \) variables), the position of the \( i^{th} \) frog is represented as \( X_i=[x_{i1},x_{i2},\ldots,x_{in}]^T \). A fitness function is defined to evaluate the frog’s position. Afterward the performance of each frog is computed based on its position. The frogs are sorted in a descending order according to their fitness. Then, the entire population is divided into \( m \) memeplexes, each of which consisting of \( n \) frogs (i.e. \( P=n \times m \)). The division is done with the first frog goes to the first memeplex, the second frog goes to the second memeplex, frog \( m \) goes to the \( m^{th} \) memeplex, and the \((m+1)^{th}\) frog back to the first memeplex, and so on.

During memeplex evolution, the position of frog \( i^{th} \) \( \left(D_i\right) \) is adjusted according to the different between the frog with the worst fitness \( X_w \) and the frog with the best fitness \( X_b \) as shown in (1). Then, the worst frog \( X_w \) leaps toward the best frog \( X_b \) and the position of the worst frog is updated based on the leaping rule, as shown in (2):

\[
\text{Position change (} D_i \text{)} = \text{rand()} \times (X_b - X_w) \\
X_w(\text{new}) = X_w + D_i \left( ||D|| < D_{\text{max}} \right) 
\]

where \( \text{rand()} \) is a random number in the range \([0,1]\) and \( D_{\text{max}} \) is the maximum allowed change of frog’s position in one jump. If this repositioning process produces a frog with better fitness, it replaces the worst frog, otherwise, the calculation in (1) and (2) are repeated with respect to the global best frog \( X_g \), (i.e. \( X_g \) replaces \( X_w \)). If no improvement becomes possible in this case, then a new frog within the feasible space is randomly generated to replace the worst frog. The evolution process is continued until the termination criterion is met. The termination criterion could be the number of iterations or when a frog of optimum fitness is found [25]. In the search based heuristic methods two aspects should be investigated, exploration and exploitation. Exploration deals with searching the solution space while exploitation is able to find optimal solution around the best solution. For a good performance it is necessary to make a suitable tradeoff between exploration and exploitation. The SFLA may not be able to find the global optimum in some optimization problems and may get trapped in local optima. Two most common reasons that cause this problem in SFLA are as follows:

3. **MODIFIED SHUFFLED FROG LEAPING ALGORITHM**

1. In the original SFLA, dividing of population to memeplexes is done by sorting the population in a decreasing order in terms of function evaluation value of each member. This strategy is inhomogeneous, that causes the performance of the first memeplexes to be better than the last ones, and hence learning process in the last memeplexes cannot be well performed because there are fewer frogs with a better fitness value in the last memeplexes than the first ones [29].

2. According to the original frog leaping rule demonstrated in Fig. 1, the possible new position of the worst frog is restricted in the line segment between its current position and the best frog’s position, and the worst frog will never jump over the best one. As a result, this frog leaping rule limits the local search space during each memeplex evolution step. Also, according to (1) and (2), the worst frog is only affected by the best frog; therefore the best frog has less chance of evolution during the leaping process. These issues make the algorithm having an insufficient learning mechanism and cause premature convergence and lead the algorithm to be trapped in local optimum easily.
As mentioned before, the learning mechanism of original SFLA possess some difficulties such as weak search ability, trapping in the local optimums and the premature convergence. Therefore, in order to obviate these problems and increase ability of the algorithm in the search space exploration, a new frog leaping rule is proposed as follows.

To overcome the above problem, a suggestion is given below. This suggestion is based on the modification in [28]. In [28], the authors define an uncertainty terms since in nature, the worst frog cannot jump exactly to its target position. Due to imperfect perception, a modified frog leaping rule is defined as:

$$D = r \times c \times (X_b - X_w) + W$$

(3)

$$W = [r_1 w_{1, \text{max}}, r_2 w_{2, \text{max}}, \ldots, r_d w_{d, \text{max}}]$$

(4)

$$W_{\text{max}}^{\text{iteration}} = \lambda^{\text{iteration}} W_{\text{max}}^{0}$$

(5)

$$X_w^{(\text{new})} = \begin{cases} X_w + D & \text{if } \|D\| \leq D_{\text{max}} \\ X_w + \frac{D}{\sqrt{D \cdot D}} D_{\text{max}} & \text{if } \|D\| > D_{\text{max}} \end{cases}$$

(6)

where \( r \) is a uniformly distributed random number in the interval \([0, 1]\); \( c \) is a constant chosen in the range between 1 and 2; \( r_i \) (\( 1 \leq i \leq d \)) are uniformly distributed random numbers in the interval \([-1, 1]\); \( w_{i, \text{max}} \) is the maximum allowed perception and action uncertainties in the \( i \)-th dimension of the search space, the maximum uncertainties are exponentially decreased according to (5) where \( \lambda \) is a decay factor in the range between 0 and 1; \( w_{\text{max}}^{0} \) is the initial maximum uncertainties where can be chosen equal to 10-20 percent of the initial range of each dimensions and \( D_{\text{max}} \) is the maximum allowed distance of one jump.

The rest of the algorithm is similar to the standard SFL as explained in the previous subsection. With the above given modification in [28], the algorithm may trap in the local optimum since there is a still insufficient learning mechanism (as it is evident in Section V Table I). In this paper, learning mechanism is improved while keeps the uncertainties term the same, which are defined in (4)-(5).

Instead of learning from the best frog, all the frogs are considered based on the following equation:

$$D_i = r \times c_i \times (X_i - X_w) + W$$

(7)

Then, the new position of the frog is obtained as follows:

$$X_w^{(\text{new})} = X_w^{(\text{old})} + D_{\text{min}} \leq D_{i} \leq D_{\text{max}}$$

(8)

If the repositioning process produces a frog with better fitness, it replaces the worst frog. Otherwise, the process is repeated with respect to the global best frog (\( X_b \)) with the \( X_i \). Also, \( c_i \) is the learning factor of the worst solution in a memeplex from the best solution in the whole population which varies within the range \([0, \text{ASF}]\). Where ASF is a control parameter which is called the adaptive scaling factor (ASF) and is selected before running the algorithm. A lower value of ASF causes the search to fine tuning the process in small steps while causing slow convergence. A larger value of ASF speeds up the search, but it reduces the exploitation capability of the perturbation process. For some classes of problems, lower values of ASF are appropriate while for some, higher ones are convenient. Automatic tuning of ASF is performed by employing the Rechenberg’s 1/5 mutation rule which states that the ratio of successful mutations to all mutations should be 1/5. Changing step size according to 1/5 rule in every \( k \) number of iterations is performed as the following equation:

$$D = r \times c \times (X_b - X_w) + W$$

(3)
If the algorithm cannot improve the solution with respect to Rechenberg’s 1/5 rule, that is the ratio of successful mutations to all mutations ($C_1(k)$) is less than 1/5, then ASF is decreased. If $C_1(k)$ is greater than 1/5 then ASF is increased in order to speed up the search. In MSFLA, the exploration is controlled $ASF$. If they are tuned properly, the exploration will be increased and the algorithm avoids the premature convergence. By the above mechanism, the diversity of the population is controlled. In other words, the exploration and exploitation of the search space are increased, resulting in avoiding premature convergence and better performance. In the case of improvement, it replaces the worst frog. In the case of no improvement, the algorithm goes toward stagnation and it needs a new movement to explore the new position in the search space. First, a new frog within the feasible space is randomly generated to replace the worst frog. Then, $X_i$ is leaped to explore the new positions as follows:

$$D_i = r \times c \times (X_e - X_i) + W$$

(10)

Then, the new position of the frog is obtained based on following equation:

$$X_{i,new} = \begin{cases} X_i + D_i & \text{if } \|D_i\| \leq D_{max} \\ X_i + \frac{D_i}{\sqrt{D_i^T D_i}} D_{max} & \text{if } \|D_i\| > D_{max} \end{cases}$$

(11)

If the repositioning process produces a frog with better fitness, it replaces $X_i$. Otherwise the algorithm goes to the next jump and the evolution process is continued until the termination criterion is met. The termination criterion could be the number of iterations or when a frog of maximal fitness is found. The principle of the MSFL is given by Fig.2.

4. STUDY SYSTEM AND PROBLEM FORMULATION

A. Study System

5-area-16-machine system:

This system is shown in Fig. 3, consisting of 16 machines and 68 buses for 5 interconnected areas. The first nine machines, G1 to G9, constitute the simple representation of Area 1. Next four machines G10 to G13 represent Area 2. The last three machines, G14 to G16, are the dynamic equivalents of the three large neighboring areas interconnected to Area 2. The subtransient reactance model for the generators, the first-order simplified model for the excitation systems, and the linear models for the loads and ac network are used. The system data and the concept of the small-signal stability are adopted from [32].
Based on earlier studies in [30], [31], a 546 MVar SVC is placed at bus 1 in the 5-area-16-machine system. The supplementary controller for the SVC and a PSS to be placed in machine 9 are going to be designed simultaneously by using the SFLA, MSFL. The structure shown in Fig. 4 is used for both the PSS and the supplementary controller, where the generator speed (GS) is considered as input to the PSS and the input to the supplementary controller is the active power flow in line 1-27.
Fig. 3. Single line diagram of a 5-area-16-machine study system.

Fig. 4. Block diagram for PSS and Supplementary Controller of SVC.

B. Problem Formulation

The first step to implement the SFL and MSFL is to generate the initial population (of n members). Each population is a solution to the problem which determines the parameters of the PSS and supplementary controller; that is, $K, T, T_i, T_s, T_K, T, T_i, T_s, T_k$. In the case of placing each solution into the study system, different degree of the stability is obtained for the system where the stability of the system can be shown by the related eigenvalues. For each solution the worst eigenvalue is selected and the corresponding damping ratio ($\xi$) is calculated. Now a vector $\xi = \{\xi_1, ..., \xi_n\}$ is defined whose elements are the damping ratio of the worst eigenvalue for each solution. Also, $\sigma = \{\sigma_1, ..., \sigma_n\}$ is defined as those elements are the real parts of the eigenvalues with the damping ratios less than 0.36. With these two vectors the following objective functions are defined:

$$f_1 = \min_{i=1,...,n} (\xi_i)$$  \hspace{1cm} (12)

$$f_2 = \min_{i=1,...,n} (-\sigma_i)$$  \hspace{1cm} (13)

and the multi-objective optimization problem can be defined as follows:

maximize $\{f_1, f_2\}$.

According to these objectives the damping controllers are designed so that the damping ratio of the close-loop system (about 0.3-0.4) is increased as well as shifting the eigenvalues of the close-loop system to the left hand side. In the other word, this fitness function will place the system closed-loop eigenvalues in the D-shape sector characterized by $\sigma_i < \sigma_0$ and $\xi_i > \xi_0$, as shown in Fig. 5.

To implement the algorithms, a weighted-sum-approach is used for (12)-(13). The weighted-sum-approach considers the above two objective to a single objective function. Therefore to place the system closed-loop eigenvalues in the D-shape sector shown in Fig. 5, the following objective function can be considered:

$$\max F = f_1 + f_2$$  \hspace{1cm} (14)
Furthermore, the design problem can be formulated as the constrained optimization problem, where the constraints are the bounds on the PSS parameters:

\[
\begin{align*}
1 & \leq K \leq 50 \\
1 & \leq T \leq 10 \\
0 & \leq T_1 \leq 2 \\
0 & \leq T_2 \leq 2 \\
0 & \leq T_3 \leq 2 \\
0 & \leq T_4 \leq 2 \\
\end{align*}
\]

and the bounds on the SVC supplementary controller parameters are the same as the bounds on PSS parameters, except the gain:

\[
1 \leq K \leq 100
\]

5. DESIGN OF POWER SYSTEM STABILIZER AND SUPPLEMENTARY CONTROLLER FOR SVC

The goal of the optimization is to find the best value for PSS and supplementary controller in the 5-area-16-machine system. Therefore, a configuration is considered for each solution as a vector \((K,T,T_1,T_2,T_3,T_4)\). The first step to implement the SFLA is generating the initial population \(N\) frogs where \(N\) is considered to be 100. The number of memeplex is considered to be 10 and the number of evaluation for local search is set to 10. Also \(D_{\text{max}}\) is chosen as inf. Each population is a solution to the problem which determines the parameters of the PSSs; i.e.: \((K,T,T_1,T_2,T_3,T_4)\).

In SFLA, during each generation, the frogs are evaluated with some measure of fitness, which is calculated from the objective function defined in (8), subject to (9). Then the best frogs are chosen. In the current problem, the best frog is the one that has minimum fitness. Based on Fig.1 the local search and shuffling processes (global relocation) continue until the last iteration is met. In this paper, the number of iteration is set to be 100.

To find the best value for the controller parameters, \((K,T,T_1,T_2,T_3,T_4)\); the algorithms are run for 10 independent runs under different random seeds. The results obtained by two algorithms for \((K,T,T_1,T_2,T_3,T_4)\) are shown in Table I. It should be noted that the boundary of D-shape sector for eigenvalues (Fig. 5) is defined with \(\alpha = -0.16\) and \(\beta = 0.049\). Table II shows the system close-loop eigenvalue with minimum damping ratio for designed controllers by each algorithm. It shows that the SFL cannot move the worst eigenvalue to the D-shape sector, while the MSFL can.

**Table I. The results obtained by SFL and MFL.**

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>(K)</th>
<th>(T)</th>
<th>(T_1)</th>
<th>(T_2)</th>
<th>(T_3)</th>
<th>(T_4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SFL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PSS</td>
<td>33.34</td>
<td>1.264</td>
<td>0.857</td>
<td>1.847</td>
<td>0.987</td>
<td>0.347</td>
</tr>
<tr>
<td>Supp.-Controller</td>
<td>59.74</td>
<td>4.278</td>
<td>1.938</td>
<td>0.7436</td>
<td>0.9280</td>
<td>0.547</td>
</tr>
<tr>
<td>MSFL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PSS</td>
<td>39.84</td>
<td>2.847</td>
<td>1.827</td>
<td>0.9480</td>
<td>1.098</td>
<td>0.193</td>
</tr>
<tr>
<td>Supp.-Controller</td>
<td>40.84</td>
<td>4.821</td>
<td>0.193</td>
<td>1.095</td>
<td>0.851</td>
<td>1.637</td>
</tr>
</tbody>
</table>

The obtained PSS and supplementary controller by SFLA and MSFL are placed in the 5-area-16-machine
system (Fig. 4). To show the effectiveness of the designed controller, a time-domain analysis is performed for the study system. A line-to-ground fault is applied in one of the tie lines at bus 26. The fault persisted for 70.0 ms. The behavior of the system was evaluated for 20 s. Fig. 6 shows the voltage magnitude at the fault bus. The machine angles, $\delta$, with respect to a particular machine (machine 13), were computed over the simulation period and shown in Figs. 7-9. These figures show that both controllers provide a good damping for the study system, but the one designed by MSFL performs better.

Once again to show the robustness of the designed controllers, a three-phase fault is applied in one of the tie circuits at bus 26. The dynamic behavior of the system was evaluated for 20 s. The voltage magnitude at the fault bus and machine angles, $\delta$, were computed over the simulation period and shown in Figs. 10-13. Again, these responses are similar to the responses in Figs. 9-12 for a line-to-ground fault, showing the robustness of the designed controllers.

Fig. 6. The response of the system to a line-to-ground fault at bus 26.

Fig. 7. The response of generator 1 to a line-to-ground fault at bus 26.
Fig. 8. The response of generator 3 to a line-to-ground fault at bus 26.

Fig. 9. The response of generator 9 to a line-to-ground fault at bus 26.

Fig. 10. The response of the system to three-phase fault at bus 26.
Fig. 11. The response of generator 1 to a three-phase fault at bus 26.

Conclusion

In this paper a new real version of shuffled frog leaping Algorithm and a modified shuffled frog leaping...
Algorithm (MSFL), is used to simultaneously design coordinated damping controllers. For this the parameters of the controllers are determined using an eigenvalue-based objective function. In SFL, the local search is done through the evolution in memeplexes. The issue of exploration and exploitation is taken into account by a frog leaping rule for local search and a memetic shuffling rule for global information exchange. In this paper, learning mechanism is improved. To show the effectiveness and robustness of the designed controller, a line-to-ground fault is applied at a bus. The simulation studies show that the designed controllers improve the stability of the system. The obtained results show that the MSFL has a better feature comparing to SFL for the current problem.

REFERENCES