Improving Power System Damping using EP Based PI Controller

N. A. Mohamed Kamari, Member, IEEE, I. Musirin, Member, IEEE, M. M. Othman, Member, IEEE

Abstract—This paper presents a new stochastic optimization technique for tuning conventional PI controller parameters of a static var compensator (SVC) which controls a synchronous machine. As a type of Flexible AC Transmission Systems (FACTS) device, SVC is designed and implemented to improve the damping of a synchronous generator. Computational intelligence technique based PI controller using SVC is implemented in this study. The study involves the development of PI controller for SVC placement, while computational intelligence technique is used to optimize the values of proportional gain, \( K_p \) and interval gain, \( K_i \) parameters of PI controller. Validation with respect to eigenvalues determination and synchronizing and damping torque coefficients (\( \Delta T \) and \( \Delta T_k \)) value confirmed that the proposed technique is effective to improve the angle stability problem.

Keywords—Transient Stability, Synchronizing Torque Coefficient, Damping Torque Coefficient, Evolutionary Programming, Particle Swarm Optimization.

I. INTRODUCTION

Static var compensator (SVC) is one of the FACTS technologies that are used widely for power transmission systems applications. The main application of SVC is to regulate the voltage of transmission systems. Over the last decades, many techniques have been proposed for the damping controllers for SVC to improve the damping of synchronous machines oscillations mode. Some techniques have been explored by means of the lead lag controllers [5], proportional-integral (PI) controllers [7] and proportional-integral-derivative (PID) controllers [8].

Recently, Evolutionary Programming (EP) has received much attention for global optimization problems. This heuristic population-based search method used both random variation and selection. The search for an optimal solution is based on the natural process of biological evolution and is accomplished in a parallel method in the parameter search space. EP-based method has been applied in various researches in static and dynamic system stability [6], [9].

This paper presents an efficient technique to determine the parameters of SVC damping controller in solving angle stability problems. PI controller has been chosen for the damping controller and its fixed-gains are determined using EP optimization technique. The goal is to stabilize the system in minimum time. A commonly used evolutionary computing technique, Particle Swarm Optimization (PSO) is selected for comparison purpose. In this paper, 3 conditions were considered: system with SVC and conventional PI controller, system with SVC and PSO based PI controller, and system with SVC and EP based PI controller.

II. THE SYSTEM MODEL

A single machine to infinite bus (SMIB) system model is considered in this study. The SVC is placed at the middle of the transmission line which is generally considered to be the ideal site. The following equations represent SMIB system with ignored SVC:

\[
\frac{\Delta \omega}{\Delta t} = \frac{\Delta T_m - K_1 \Delta \delta - K_2 \Delta \omega - K_3 \Delta \psi}{2H} \\
\frac{\Delta \delta}{\Delta t} = \Delta \omega, \Delta \omega = \frac{\Delta \omega}{\Delta t} \\
\frac{\Delta \psi_{\text{fd}}}{\Delta t} = -\frac{K_3 K_4 \Delta \delta + K_5 K_4 \Delta \psi_{\text{fd}} - K_3 \Delta E_{\text{fd}}}{T_3} \\
\frac{\Delta E_{\text{fd}}}{\Delta t} = -\frac{K_4 K_5 \Delta \delta + K_5 K_6 \Delta \psi_{\text{fd}} + \Delta E_{\text{fd}}}{T_R}
\]

where \( T_m \) is a mechanical torque, \( H \) is the inertia constant, \( K_D \) is the damping torque coefficient, \( K_i \) and \( T_R \) are the constant and time constant of the exciter oscillation system, respectively. \( \omega_0 \) is equal to \( 2\pi f_0 \). In this representation, the dynamic characteristics of the system are expressed in terms of \( K \) constants, \( K_1, K_2, K_3, K_4, K_5, \) and \( T_3 \) with linearized SMIB system. This model is related with some variables such as electrical torque, rotor speed, rotor angle and exciter output voltage.

The PI controller is designed to increase the damping torque of the SMIB system. The structure of PI controller is shown in Fig. 1. Value of proportional gain, \( K_p \) and interval gain, \( K_i \) parameters of PI controller should be kept within specified limits. In this paper, the EP algorithm is proposed for the optimal computation of the PI controller parameters.

\[
\begin{align*}
\frac{\Delta \omega}{\Delta t} &= \frac{K_p}{s} \Delta \beta \\
\max \Delta \beta &= \frac{K_p}{1 + sT_R} \max \Delta \sigma
\end{align*}
\]

Fig 1. Structure of SVC-PI controller

The following state-space form is developed from SMIB system model with SVC-PI damping controller:

\[
X = A \cdot X + B \cdot U
\]

where

N. A. Mohamed Kamari is with Universiti Teknologi Mara, Shah Alam, Selangor, Malaysia (e-mail: azwan_ki@hotmail.com).
I. Musirin is with Universiti Teknologi Mara, Shah Alam, Selangor, Malaysia (e-mail: ismailbm@salam.uitm.edu.my).
M. M. Othman is with Universiti Teknologi Mara, Shah Alam, Selangor, Malaysia (e-mail: mamat505my@yahoo.com).

978-1-4577-0354-6/11/$26.00 ©2011 IEEE
A. Detail calculation of parameters operating real and reactive loading as well as the excitation only a function of the ratio of impedance, are a function of the solution is based on the natural process of biological evolution and can yield global solutions to any problem, regardless of the form of the objective function.

In the EP algorithm, the population has \( n \) candidate solutions with each candidate solution is an \( m \)-dimensional vector, where \( m \) is the number of optimized parameters. The EP algorithm can be described as:

a) Step 1 (Initialization): Generation counter \( i \) is set to 0, and generate \( n \) random solutions \((x_k, k=1,\ldots,n)\). The \( k^{th} \) trial solution \( x_k \) can be written as \( x_k=[p_1,\ldots,p_n] \), where the \( r \)th optimized parameter \( p_r \) is generated by random value in the range of \([p_r^{\min}, p_r^{\max}]\) with uniform probability. Each individual is evaluated using the objective function \( J \). In this initial population, maximum value of objective function \( J_{\text{max}} \) will be searched, the target is to find the best solution \( x_{\text{best}} \) with objective function \( J_{\text{best}} \).

b) Step 2 (Mutation): Each parent \( x_k \) produces one offspring \( x_{k+n} \). Each optimized parameter \( p_i \) is perturbed by a Gaussian random variable \( N(0, \sigma^2) \). The standard deviation \( \sigma \) specifies the range of the optimized parameter perturbation in the offspring. \( \sigma \) equation is as follows:

\[
\sigma_l = \beta \times \frac{J(x_k)}{J_{\text{max}}} (p_l^{\max} - p_l^{\min})
\]

where \( \beta \) is a scaling factor, and \( J(x_k) \) is the objective function of the trial solution \( x_k \). The value of optimized parameter will be set at certain limit if any value violates its specified range. The offspring \( x_{k+n} \) can be described as:

\[
x_{k+n} = x_k + N(0, \sigma^2_l) \ldots N(0, \sigma^2_m)
\]

III. COMPUTATIONAL INTELLIGENCE TECHNIQUE

A. Evolutionary Programming

The Evolutionary Programming (EP) is one of the evolutionary computing techniques which use the models of biological evolutionary process to obtain the solution for complex engineering problems. The search for an optimal solution is based on the natural process of biological evolution and is accomplished in a parallel method in the parameter search space. EP belongs to the generic fields of the simulated evolution and artificial life. It is robust, flexible and adaptable and it can yield global solutions to any problem, regardless any form of the objective function.

In the EP algorithm, the population has \( n \) candidate solutions with each candidate solution is an \( m \)-dimensional vector, where \( m \) is the number of optimized parameters. The EP algorithm can be described as:

- Step 1 (Initialization): Generation counter \( i \) is set to 0, and generate \( n \) random solutions \((x_k, k=1,\ldots,n)\). The \( k^{th} \) trial solution \( x_k \) can be written as \( x_k=[p_1,\ldots,p_n] \), where the \( r \)th optimized parameter \( p_r \) is generated by random value in the range of \([p_r^{\min}, p_r^{\max}]\) with uniform probability. Each individual is evaluated using the objective function \( J \). In this initial population, maximum value of objective function \( J_{\text{max}} \) will be searched, the target is to find the best solution \( x_{\text{best}} \) with objective function \( J_{\text{best}} \).

- Step 2 (Mutation): Each parent \( x_k \) produces one offspring \( x_{k+n} \). Each optimized parameter \( p_i \) is perturbed by a Gaussian random variable \( N(0, \sigma^2) \). The standard deviation \( \sigma_i \) specifies the range of the optimized parameter perturbation in the offspring. \( \sigma \) equation is as follows:

\[
\sigma_l = \beta \times \frac{J(x_k)}{J_{\text{max}}} (p_l^{\max} - p_l^{\min})
\]

where \( \beta \) is a scaling factor, and \( J(x_k) \) is the objective function of the trial solution \( x_k \). The value of optimized parameter will be set at certain limit if any value violates its specified range. The offspring \( x_{k+n} \) can be described as:

\[
x_{k+n} = x_k + N(0, \sigma^2_l) \ldots N(0, \sigma^2_m)
\]

- Step 3 (Statistics): The minimum objective function \( J_{\text{min}} \) and the maximum objective function \( J_{\text{max}} \) of all individuals are calculated.

- Step 4 (Update the best solution): If \( J_{\text{min}} \) is smaller than \( J_{\text{best}} \), go to Step 5, or else, update the best solution, \( x_{\text{best}} \). Set \( J_{\text{best}} \) as \( J_{\text{best}} \), and go to Step 5.

- Step 5 (Combination): All members in the population \( x_k \) are combined with all members from the offspring \( x_{k+n} \) to become \( 2n \) candidates. These individuals are then ranked in descending order, based on their objective function as their weight.

- Step 6 (Selection): The first \( n \) individuals with higher weights are selected along with their objective functions as parents of the next generation.

- Step 7 (Stopping criteria): The search process will be terminated if one of the followings is satisfied:

  - It reaches the maximum number of generations
  - The value of \( (J_{\text{max}} - J_{\text{min}}) \) is very close to 0.
If the process is not terminated, the generation counter will be set to \( i = i + 1 \) and algorithm will start again from Step 2.

### B. Particle Swarm Optimization

Particle Swarm Optimization (PSO) was introduced by Dr. Russ Eberhart and Dr. James Kennedy in 1995. Similar to EP, PSO is an evolutionary based optimization technique, which imitates the behaviour of birds flocking and fish schooling. The technique is initialized with a population of random particles where each particle is a candidate solution. The particles fly through the problem space by following the current optimum particles. Then, it searches for optimal solution by updating positions of each particle. In this paper, the PSO algorithm works as follows:

a) Step 1 (Initialization): The velocity \( v_i \) and position \( x_i \) of \( N \) particles \((i = 1, \ldots, N)\) are randomly created to form initial population. Similar to EP, each particle is evaluated using the objective function \( J \). In this initialization process, \( J_i \) is set as personal best objective function \( J_{p} \) for \( i \)th particle. The maximum objective function of all particles \( J_{\text{max}} \) is set as global best objective function \( J_{g} \). The position \( x_i \) for \( J_{p} \), \( J_{\text{max}} \) and \( J_{g} \) is set as personal best position \( p_{i} \), position with maximum objective function \( p_{m} \) and global best position \( g \), respectively.

b) Step 2 (Update the velocity and positions): At \( j \)th iteration, the velocity and position of \( i \)th particle is updated according to the following equations:

\[
\begin{align*}
v_i(j) &= \omega v_i(j-1) + c_1 r_1 \{ p_i(j-1) - x_i(j-1) \} \\
&\quad + c_2 r_2 \{ g(j-1) - x_i(j-1) \} \\
x_i(j) &= v_i(j) + x_i(j-1)
\end{align*}
\]  

where, \( \omega \) is the inertia weight, \( c_1 \) and \( c_2 \) are acceleration coefficients, and \( r \) is random function in the range \([0,1]\).

c) Step 3 (Calculate objective functions): The new \( J, J_{\text{max}} \) and the minimum objective function of all particles \( J_{\text{min}} \) are calculated.

d) Step 4 (Update the best positions): \( p_i \) and \( g \) are updated when the following conditions are met:
   - If \( J_i \) is bigger than \( J_{p} \), set \( J_{p} \) as \( J_i \) and set \( x_i \) as \( p_i \). Else, the value of \( J_{p} \) and \( p_i \) are maintained.
   - If \( J_{\text{max}} \) is bigger than \( J_{p} \), set \( J_{\text{max}} \) as \( J_{p} \) and set \( p_{m} \) as \( g \). Else, the value of \( J_{\text{max}} \) and \( g \) are maintained.

e) Step 5 (Stopping criteria): The search process will be terminated if one of the followings is satisfied:
   - It reaches the maximum number of iterations
   - The value of \( J_{\text{max}} - J_{\text{min}} \) is very close to 0.

If the process is not terminated, the iteration counter will be set to \( j = j + 1 \) and algorithm will start again from Step 2.

### C. Objective Function

The SVC controller is designed to minimize the power angle deviation after a disturbance and to accelerate the damping of the power system oscillations. In this work, the objective function can be formulated as the maximization of:

\[
J = \text{inv} \left( 1 + \sum_{t=1}^{T_{\text{sim}}} \Delta \omega(t) \right)
\]  

where, \( \Delta \omega(t) \) is the change in rotor speed of the system at the \( p \) loading condition and \( T_{\text{sim}} \) is the time range of the simulation. Hence, the design problem can be formulated as:

Maximize \( J \)

Subject to

\[
\begin{align*}
K_P^m &\leq K_P \leq K_P^m \\
K_I^m &\leq K_I \leq K_I^m
\end{align*}
\]

With the proposed approach, optimum proportional \( K_P \) and integral gain \( K_I \) settings of the PI controller were searched using EP and PSO for different operating points simultaneously.

### IV. CONCEPT OF SYNCHRONIZING & DAMPING TORQUE AND LEAST SQUARE (LS) METHOD

The change of electromagnetic torque \( \Delta T_e(t) \) can be broken down into two components: the synchronizing torque \( K_S \) and damping torque \( K_D \). The \( K_S \) component is proportional to the change in rotor angle \( \Delta \delta(t) \) and the \( K_D \) component is proportional to the change in rotor speed \( \Delta \omega(t) \). The change of estimated torque \( \Delta T_e^*(t) \) equation is shown below [2]-[4]:

\[
\Delta T_e^*(t) = K_S \Delta \delta(t) + K_D \Delta \omega(t)
\]  

where:

\( \Delta \delta(t) \): Change in rotor angle

\( \Delta \omega(t) \): Change in rotor speed

\( K_S \): Synchronizing torque coefficients

\( K_D \): Damping torque coefficients

All the data for \( \Delta \delta(t) \), \( \Delta \omega(t) \) and \( \Delta T_e(t) \) can be obtained from either offline simulation or online measurements. Following a small disturbance, the time responses of these three items are recorded. The least square (LS) technique is then used to minimize the sum of the square of the differences between the estimated torque \( \Delta T_e^*(t) \) and the change of \\

### V. RESULTS AND DISCUSSION

In this paper, simulation studies of a SMIB power system with SVC are carried out in MATLAB environment. The MATLAB program simulation is conducted using all the 7 parameters: \( K_P, K_I, T_P, T_I, K_S, K_D \) and \( K_{\text{c}} \). In this simulation, value of proportional gain parameter, \( K_P \) and integral gain parameter, \( K_I \) of PI controller are optimized until maximum value of the objective function \( J \) is defined with selected value of \( K_P \) and \( K_I \). A Simulink model of SMIB system with SVC is
developed based on these 7 calculated parameters and 2 optimized parameters: $K_P$ and $K_I$ to produce 3 system responses (speed deviation, $\Delta \omega (t)$, angle deviation, $\Delta \delta (t)$ and torque deviation, $\Delta T_e (t)$). Based on these system responses, $K_S$ and $K_D$ are then calculated using LS technique.

In this case, the performance of SVC with conventional based PI controller (C-PI) is compared to SVC with EP based PI controller (EP-PI) and SVC with PSO based PI controller (PSO-PI). Following three different loading conditions are simulated:

i. Case 1 ($P = -0.1 \text{ p.u.}, Q = 0.35 \text{ p.u.}$)

ii. Case 2 ($P = 0.5 \text{ p.u.}, Q = 0.25 \text{ p.u.}$)

iii. Case 3 ($P = -0.7 \text{ p.u.}, Q = -0.37 \text{ p.u.}$)

The response of speed deviation for Case 1 is shown in Figure 2(a). The system with C-PI is poorly damped and becomes stable for more than 2 seconds. Table I tabulates the results for comparative studies using C-PI, EP-PI, and PSO-PI. Both EP-PI and PSO-PI systems become stable in below 1.5 seconds. Although both techniques show almost the same result, the response of speed deviation for the proposed EP-PI controller is damp faster than the PSO-PI.

The response for torque deviation for Case 1 is shown in Figure 2(b). Here also, the proposed PI controller shows better damping compared to C-PI and PSO-PI. As compared to C-PI and PSO-PI, the proposed EP-PI system has low oscillation.

<table>
<thead>
<tr>
<th>Type</th>
<th>$K_P$</th>
<th>$K_I$</th>
<th>$K_S$</th>
<th>$K_D$</th>
<th>$J$</th>
<th>Eigenvalues</th>
</tr>
</thead>
<tbody>
<tr>
<td>C-PI</td>
<td>0.5000</td>
<td>15.0000</td>
<td>0.1003</td>
<td>0.6282</td>
<td>-19.4812</td>
<td>-8.4509 ± 4.3040i</td>
</tr>
<tr>
<td>EP-PI</td>
<td>0.0267</td>
<td>0.0058</td>
<td>0.9638</td>
<td>20.5568</td>
<td>-16.9473</td>
<td>-5.0437 ± 8.6848i</td>
</tr>
<tr>
<td>PSO-PI</td>
<td>0.1810</td>
<td>0.0056</td>
<td>0.9592</td>
<td>19.9516</td>
<td>-17.9936</td>
<td>-5.6366 ± 8.6848i</td>
</tr>
</tbody>
</table>

The response of speed deviation and torque deviation for Case 2 are shown in Figure 3(a) and 3(b), respectively. Both responses indicated that the proposed PI controller technique give result compared to C-PI and PSO-PI. The response for the proposed EP-PI is damping faster than the other two techniques.

$K_S$, $K_D$, eigenvalues, $J$ and simulation time for Case 2 are shown in Table II. The results obtained verify that proposed PI controller give better result compared the other two techniques.
### TABLE II

<table>
<thead>
<tr>
<th>Type</th>
<th>$K_p$</th>
<th>$K_s$</th>
<th>Eigenvalues</th>
</tr>
</thead>
<tbody>
<tr>
<td>C-PI</td>
<td>0.5000</td>
<td>0.1039</td>
<td>-20.0000</td>
</tr>
<tr>
<td></td>
<td>15.0000</td>
<td>0.5340</td>
<td>-8.6786 ± 4.7537i</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>-1.5281 ± 8.0918i</td>
<td></td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>-0.0000</td>
<td></td>
</tr>
<tr>
<td>EP-PI (Simulation time: 38.30s)</td>
<td>0.9894</td>
<td>0.0024</td>
<td>-20.6474</td>
</tr>
<tr>
<td></td>
<td>14.6090</td>
<td>3.6607</td>
<td>-6.7912 ± 4.6508i</td>
</tr>
<tr>
<td></td>
<td>0.9447</td>
<td>-3.0918 ± 7.5914i</td>
<td></td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>-0.0000</td>
<td></td>
</tr>
<tr>
<td>PSO-PI (Simulation time: 46.09s)</td>
<td>0.7269</td>
<td>0.0002</td>
<td>-20.2537</td>
</tr>
<tr>
<td></td>
<td>12.6327</td>
<td>6.5158</td>
<td>-2.2436 ± 7.9073i</td>
</tr>
<tr>
<td></td>
<td>0.9430</td>
<td>-7.8363 ± 4.4095i</td>
<td></td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>-0.0000</td>
<td></td>
</tr>
</tbody>
</table>

**TABLE III**

<table>
<thead>
<tr>
<th>Type</th>
<th>$K_p$</th>
<th>$K_s$</th>
<th>Eigenvalues</th>
</tr>
</thead>
<tbody>
<tr>
<td>C-PI</td>
<td>0.5000</td>
<td>0.0898</td>
<td>-21.3097</td>
</tr>
<tr>
<td></td>
<td>15.0000</td>
<td>0.2261</td>
<td>-1.3035 ± 6.9859i</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>-0.0000</td>
<td></td>
</tr>
<tr>
<td>EP-PI (Simulation time: 40.12s)</td>
<td>0.7847</td>
<td>0.0001</td>
<td>-20.0283</td>
</tr>
<tr>
<td></td>
<td>15.5333</td>
<td>0.9329</td>
<td>-10.4745</td>
</tr>
<tr>
<td></td>
<td>0.0001</td>
<td>-2.8485 ± 8.5146i</td>
<td></td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>-4.2273</td>
<td></td>
</tr>
<tr>
<td>PSO-PI (Simulation time: 47.31s)</td>
<td>0.5655</td>
<td>0.0014</td>
<td>-19.4749</td>
</tr>
<tr>
<td></td>
<td>13.9994</td>
<td>28.4348</td>
<td>-11.5262</td>
</tr>
<tr>
<td></td>
<td>0.9231</td>
<td>-2.2541 ± 8.5146i</td>
<td></td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>-4.9179</td>
<td></td>
</tr>
</tbody>
</table>

The response of speed deviation and torque deviation for Case 3 are shown in Figure 4(a) and 4(b), respectively. The responses observed in both figures indicate that the proposed EP-based PI controller gives better results compared to C-PI and PSO-PI systems. It is also shown that the EP-PI technique damps faster than the others.

**VI. CONCLUSIONS**

This paper has presented computational intelligence techniques based PI controller. Two methods based on EP and PSO computation intelligence techniques for optimizing $K_p$ and $K_s$ have been developed. Results obtained from the study indicated that EP outperformed PSO in terms of giving better $K_p$ and $K_s$ values, which are responsible for stability point determination. The performance of EP compared to PSO is validated with respect to speed and torque deviation response as well as eigenvalues, $K_s$ and $K_D$ determination. EP also manages to perform much faster than PSO.

**REFERENCES**


BIOGRAPHIES

Nor Azwan Mohamed Kamari was born in Perak, Malaysia on January 5, 1976. He received bachelor degree in Electrical & Electronics Engineering from Meiji University, Japan on 2000 and master degree from Ehime University, Japan on 2004, respectively. He is currently pursuing a PhD degree in power system at the Universiti Teknologi MARA, Malaysia. His research interests include angle stability analysis and optimization techniques.

Ismail Musirin obtained Diploma of Electrical Power Engineering in 1987, Bachelor of Electrical Engineering (Hons) in 1990; both from Universiti Teknologi Malaysia, MSc in Pulsed Power Technology in 1992 from University of Strathclyde, United Kingdom and PhD in Electrical Engineering from Universiti Teknologi MARA, Malaysia in 2005. He has published 2 books and more than 80 technical papers in the international and national conferences, and also in the international journals. He is a reviewer of IEEE transactions and IET journals. He is also appointed as the permanent reviewer for the World Scientific and Engineering Academy and Society (WSEAS) centered in Greece. His research interest includes power system stability, optimization techniques, distributed generator and artificial intelligence. He is also a member of IEEE, IEEE Power Engineering Society and Artificial Immune System Society (ARTIST).

Muhammad Murtadha Othman received Bachelor of Electrical Engineering (Hons) in 1998 from The Staffordshire University, UK, MSc in Electrical Engineering in 2000 from Universiti Putra Malaysia and PhD in Electrical Engineering in 2006 from Universiti Kebangsaan Malaysia. He is currently a lecturer at The Faculty of Electrical Engineering, Universiti Teknologi MARA, Malaysia. He has authored and co-authored more than 40 technical papers published in the international journals, national and international conferences. His research interest includes power system optimization, artificial intelligence application in power system, biological computing techniques, voltage stability studies, available transfer capability and power system reliability. He is also appointed as international journal reviewers.