A FREE MODEL BASED CONTROLLER DESIGN FOR POWER SYSTEM STABILIZATION

Hee-Sang Ko†, Kwang Y. Lee††, and Ho-Chan Kim†††

† *Dept. of Electrical and Computer Eng., Univ. of British Columbia, Vancouver, Canada*
†† *Dept. of Electrical Eng., Pennsylvania State Univ., University Park, USA*
††† *Dept. of Electrical Eng., Cheju National Univ., Cheju, Korea*

Abstract: This paper presents an intelligent modeling approach, named as free model approach, for a closed-loop system identification using input and output data and its application to design a power system stabilizer (PSS). The free model concept is introduced as an alternative intelligent system technique to design a controller for an unknown system with input and output data only, and without the detailed knowledge of mathematical model for the system. In the free model, the data used has incremental forms using backward difference operators. The parameters of the free model can be obtained by the simultaneous perturbation stochastic approximation (SPSA) method. The feasibility of the proposed method is demonstrated in a one-machine infinite-bus power system. The linear quadratic regulator method is applied to the free model to design a PSS for the system, and compared with the conventional PSS in different loading conditions and system failures such as the outage of a major transmission line or a three phase to ground fault which causes the change of the system structure.

Keywords: Free model, power system stabilization, simultaneous perturbation stochastic approximation, linear quadratic regulator.

1. INTRODUCTION

Traditionally, controllers are designed on the basis of a mathematical description of a system and its linearized model. Therefore, it is difficult to implement these model-based controllers to a real system, especially, to a system, which is complex and nonlinear such as power systems. A power system stabilizer (PSS) with the excitation system is the most common tool used to enhance the damping of low frequency oscillations of a power system. Considerable effort has been made to design PSS for power systems, most of which is based on deMello and Concordia's pioneering work (deMello and Concordia, 1969). They use a linearized model to find a proper set of parameters in a fixed structure PSS. Linear optimal control and modern control theories are also introduced to improve the dynamic performance of power systems under the uncertainty of power system models (Doi, 1984; Law et al., 1994). These techniques, however, depend on the accuracy of the model, which is less reliable as the power system becomes larger. Adaptive techniques are also employed in the PSS design for a wide range of operations (Ghosh et al., 1984; Gu and Bollinger, 1989). Recently, there has been a great deal of research that reports on artificial neural network and fuzzy logic and their applications to control and power systems (Zhang et al., 1994; El-Metwally and Malik, 1996).

This paper presents the free model approach for system identification using input and output data and its application to a PSS design. The free model concept is introduced as an alternative intelligent system technique to design a controller for an unknown dynamic system with input and output data only, and it does not require the knowledge of mathematical model for the system. The idea of free model comes from the Taylor series approximation, where an output trajectory can be estimated when such data as position, velocity, and acceleration are known.

One of the techniques using only loss function measurements that have attracted considerable attention for difficult multivariate problems is the simultaneous perturbation stochastic approximation (SPSA) method (Spall, 2003). The SPSA is based on a highly efficient and easily implemented "simultaneous perturbation" approximation to the gradient: this gradient approximation uses only two cost function measurements independent of the number of parameters being optimized. The parameters of the free model can be obtained by the SPSA method using the input-output data and a controller can be designed based on the free model. The free model is then transformed to a linear state space model and the linear quadratic regulator (LQR) method (Anderson and Moore, 1990) is used to design a controller. In this paper, one machine infinite-bus system (Sauer and Pai, 1998) is studied to demonstrate the feasibility of the proposed method.

The LQR method is applied to the free model to design a PSS for the systems, and compared with the conventional PSS (CPSS). This proposed SPSA based LQR controller is applied to the test systems and compared with the CPSS. Although no mathematical model is used to design the controller, the proposed controller is robust in different loading conditions and system failures such as the outage of a major transmission line or a three phase to ground fault.

2. DESCRIPTION OF THE FREE MODEL

We consider an arbitrary nonlinear time-invariant discrete-time system, represented by

$$\begin{aligned} y(k+1) = f\,(&y(k), y(k-1), \cdots, y(k-N),\\ &u(k), u(k-1), \cdots, u(k-M)) \end{aligned} \tag{1}$$

where $y(k-i)$ and $u(k-j)$, for $i = 0,1,\cdots,N,\quad j = 0,1,\cdots,M$, denote the delayed outputs and inputs, respectively.

It can be shown that the delayed signals are made of increments or differences. Using the backward difference operator (Phillips and Nagle, 1997) defined by

$$\begin{aligned} &\Delta^n f(k) = \Delta^{n-1} f(k) - \Delta^{n-1} f(k-1), \quad n \geq 1\\ &\Delta^0 f(k) = f(k) \end{aligned} \tag{2}$$

the system (1) can be represented as

$$\begin{aligned} y(k+1) = f\,(&y(k), \Delta y(k), \cdots, \Delta^N y(k), u(k),\\ &u(k-1), \Delta u(k-1), \cdots, \Delta^M u(k-1)) \end{aligned} \tag{3}$$

Equation (3) is expanded into Taylor series.

$$\begin{aligned} y(k+1) = y(k) &+ \sum_{i=1}^{N} a_i \Delta^i y(k) + b_0 \Delta u(k)\\ &+ \sum_{i=1}^{M} b_i \Delta^i u(k-1) + O(k) \end{aligned} \tag{4}$$

where $a_i = \partial f \big/ \partial \Delta^i y(k-1)$, $b_0 = \partial f \big/ \partial u(k-1)$, $b_i = \partial f \big/ \partial \Delta^i u(k-2)$ and $O(k)$ represents the high order terms.

By subtracting $y(k)$ from both sides of (4), the above equation is represented as following:

$$\begin{aligned} \Delta y(k+1) = &\sum_{i=1}^{N} a_i \Delta^i y(k) + b_0 \Delta u(k)\\ &+ \sum_{i=1}^{M} b_i \Delta^i u(k-1) + O(k) \end{aligned}$$

By neglecting high order terms and dividing both sides with Δ, the free model is then defined as following:

$$\begin{aligned} \hat{y}(k+1) = &\sum_{i=1}^{N} a_i \Delta^{i-1} y(k) + b_0 u(k)\\ &+ \sum_{i=1}^{M} b_i \Delta^{i-1} u(k-1) \end{aligned} \tag{5}$$

where N and M are, respectively, the output and input orders of the free model, and $\hat{y}(k+1)$ denotes the estimate of $y(k+1)$. The remaining problem is how to determine parameters a_i, b_0, and b_i. Here, we use the SPSA method (Spall, 2003) in determining these parameters. The SPSA method is based on the least squares problem, which is designed to minimize the loss function $E(\theta)$:

$$\min\ E(\theta) = \sum_{i=1}^{n} (y(k-i+1) - \hat{y}(k-i+1))^2 \tag{6}$$

where $\theta = [a_1 \cdots a_N b_0 \cdots b_M]^T$ is the parameter vector of the free model, y is the plant output and $\hat{y}$ indicates the estimated output of the free model.

3. SPSA BASED FREE MODEL APPROXIMATION

3.1 The Basic SPSA Algorithm (Spall, 2003)

The goal of the SPSA is to minimize a loss function $L(\theta)$, where the loss function is a scalar-valued "performance measure" and θ is a continuous-valued p-dimensional vector of parameters to be adjusted. The SPSA algorithm works by iterating from an initial guess, where the iteration process depends on the "simultaneous perturbation" approximation to the gradient $g(\theta) \equiv \partial L(\theta) / \partial \theta$.

Assume that the measurements of the loss function are available at any value of θ:

$$E(\theta) = L(\theta) + noise$$

where $L(\theta)$ is a differentiable function of θ and the minimum point θ^* corresponds to a zero point of the gradient, i.e.,

$$g(\theta^*) = \left.\frac{\partial L(\theta)}{\partial \theta}\right|_{\theta=\theta^*} = 0 \tag{7}$$

In cases where more than one point satisfies (7), then the algorithm may only converge to a local minimum. The modifications of basic SPSA algorithm allow it to search for the global solution among multiple local solutions. Note also that (7) is generally associated with unconstrained optimization; however, through the application of penalty function and/or projection methods, it is possible to use (7) in a constrained problem.

The basic unconstrained SPSA algorithm is in the general recursive stochastic approximation (SA) form

$$\hat{\theta}_{k+1} = \hat{\theta}_k - a_k \hat{g}_k(\hat{\theta}_k) \tag{8}$$

where $\hat{g}_k(\hat{\theta}_k)$ is the simultaneous perturbation estimate of the gradient $g(\theta) \equiv \partial L(\theta)/\partial\theta$ at the iterate $\hat{\theta}_k$ based on the measurements of the loss function and a_k is a nonnegative scalar gain coefficient.

The essential part of (8) is the gradient approximation $\hat{g}_k(\hat{\theta}_k)$. This gradient approximation is formed by perturbing the components of $\hat{\theta}_k$ one at a time and collecting a loss measurement $E(\bullet)$ at each of the perturbations (in practice, the loss measurements are sometimes noise-free, $E(\bullet) = L(\bullet)$). This requires $2p$ loss measurements for a two-sided finite difference approximation. All elements of $\hat{\theta}_k$ are randomly perturbed together to obtain two loss measurements $E(\bullet)$. For the two-sided simultaneous perturbation gradient approximation, this leads to

$$\hat{g}_k(\hat{\theta}_k) = \frac{E(\hat{\theta}_k + c_k\Delta_k) - E(\hat{\theta}_k - c_k\Delta_k)}{2c_k}\begin{bmatrix}\Delta_{k1}^{-1}\\ \Delta_{k2}^{-1}\\ \vdots\\ \Delta_{kp}^{-1}\end{bmatrix} \quad (9)$$

where the mean-zero p -dimensional random perturbation vector, $\Delta_k = [\Delta_{k1}, \Delta_{k2}, \cdots, \Delta_{kp}]^T$, has a user-specified distribution and c_k is a positive scalar. Because the numerator is the same in all p components of $\hat{g}_k(\hat{\theta}_k)$, the number of loss measurements needed to estimate the gradient in SPSA is two, regardless of the dimension p.

3.2 The SPSA Algorithm Implementation for Free model Approximation

The step-by-step summary below shows how SPSA iteratively produces a sequence of estimates.

***Step 1 Initialization and coefficient selection*:**
Set counter index $k = 0$. Pick initial guess $\hat{\theta}_0$ in (5) and nonnegative coefficients a, c, A, α , and γ in the SPSA gain sequences $a_k = a/(A+k+1)^{\alpha}$ and $c_k = c/(k+1)^{\gamma}$. Practically effective values for α and γ are 0.602 and 0.101, respectively.

***Step 2 Generation of simultaneous perturbation vector*:**
Generate by Monte Carlo a p-dimensional random perturbation vector Δ_k , where each of the p components of Δ_k are independently generated from a zero-mean probability distribution satisfying the conditions in Spall (Spall, 2003). A simple (and theoretically valid) choice for each component of Δ_k is to use a Bernoulli ±1 distribution with probability of 0.5 for each ±1 outcome. Note that uniform and normal random variables are not allowed for the elements of Δ_k by the SPSA regularity conditions since they have infinite inverse moments.

***Step 3 Loss function evaluations*:**
Obtain two measurements of the loss function based on the simultaneous perturbation around the current $\hat{\theta}_k$: $E(\hat{\theta}_k + c_k\Delta_k)$ and $E(\hat{\theta}_k - c_k\Delta_k)$ in (6) with the c_k and Δ_k from Steps 1 and 2.

***Step 4 Gradient approximations*:**
Generate the simultaneous perturbation approximation to the unknown gradient $\hat{g}_k(\hat{\theta}_k)$ according to (9). It is sometimes useful to average several gradient approximations at $\hat{\theta}_k$, each formed from an independent generation of Δ_k .

***Step 5 Updating θ Estimate*:**
Use the standard stochastic approximation form in (8) to update $\hat{\theta}_k$ to a new value $\hat{\theta}_{k+1}$. Check for constraint violation and modify the updated θ.

***Step 6 Iteration or Termination*:**
Return to Step 2 with $k+1$ replacing k . Terminate the algorithm if there is little change in several successive iterates or the maximum allowable number of iterations has been reached.

The choice of the gain sequences (a_k and c_k) is critical to the performance of SPSA. With α and γ as specified in Step 1, one typically finds that in a high-noise setting (i.e., poor quality measurements of $L(\theta)$) it is necessary to pick a smaller a and larger c than in a low-noise setting. Although the asymptotically optimal values of α and γ are 1.0 and $1/6$, respectively, it appears that $\alpha < 1.0$ choosing usually yields better finite-sample performance through maintaining a larger step size; hence the recommendation in Step 1 to use values (α and γ) that are effectively the lowest allowable satisfying the theoretical conditions mentioned (Spall, 2003).

4. STATE SPACE REALIZATION AND LQR DESIGN

Free model can be easily adopted to design controllers with conventional design method. In this paper, an LQR is applied to design a controller that is called the SPSA based optimal controller. First, a linear transformation is introduced to convert the free model into a linear model so that the LQR design method can be applied (Anderson and Moore, 1990). The state variables are defined by the following linear transformation:

$$\begin{aligned} x_1(k) &= y(k)\\ x_2(k) &= \Delta y(k) + \beta_1 u(k-1)\\ &\vdots\\ x_N(k) &= \Delta^{N-1}y(k) + \beta_{N-1}u(k-1) + \ldots + \beta_1\Delta^{N-2}u(k-1) \end{aligned} \quad (10)$$

From the linear transformation (10), the ith state variable is defined by

$$x_i(k) = \Delta^{i-1} y(k) + \sum_{m=0}^{i-2} \beta_{i-m-1} \Delta^m u(k-1) \qquad (11)$$

where $i = 1, 2, \cdots, N$, and $\beta_0 = 0$. Solving (10) for the output increments,

$$\begin{aligned} y(k) &= x_1(k) \\ \Delta y(k) &= x_2(k) - \beta_1 u(k-1) \\ &\vdots \\ \Delta^{i-1} y(k) &= x_i(k) - \beta_{i-1} \Delta u(k-1) - \beta_{i-2} \Delta u(k-1) \\ &\quad - \beta_{i-3} \Delta^2 u(k-1) - \cdots - \beta_1 \Delta^{i-2} u(k-1) \end{aligned} \qquad (12)$$

Then applying (12) into (5) and replacing $\hat{y}(k+1)$ with $y(k+1)$,

$$y(k+1) = \sum_{i=1}^{N} a_i \Delta^{i-1} y(k) + b_0 u(k) + \sum_{i=1}^{N-1} b_i \Delta^{i-1} u(k-1) \qquad (13)$$

which, from (12), can be represented as the following equation:

$$\begin{aligned} x_1(k+1) &= \sum_{i=1}^{N} a_i x_i(k) + b_0 u(k) \\ &\quad + (b_1 - a_2 \beta_1 - a_3 \beta_2 \cdots - a_N \beta_{N-1}) u(k-1) \\ &\quad + (b_2 - a_3 \beta_1 - a_4 \beta_2 \cdots - a_N \beta_{N-2}) \Delta^1 u(k-1) \\ &\quad + \cdots + (b_{N-1} - a_N \beta_1) \Delta^{N-2} u(k-1) \end{aligned} \qquad (14)$$

Choose β_i so that the coefficients of $\Delta^i u(k-1)$ become zeros, i.e.,

$$\begin{bmatrix} a_2 & a_3 & \cdots & a_N \\ a_3 & a_4 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ a_N & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_{N-1} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_{N-1} \end{bmatrix} \qquad (15)$$

Then, (14) becomes

$$x_1(k+1) = \sum_{m=1}^{N} a_m x_m(k) + b_0 u(k) \qquad (16)$$

Now, it remains to derive the $x_i(k+1)$ for $i \geq 2$. From the definition of the backward difference operator, and (11),

$$\begin{aligned} \Delta x_{i-1}(k+1) &= x_{i-1}(k+1) - x_{i-1}(k) \\ &= \left\{ \Delta^{i-2} y(k+1) + \sum_{m=1}^{i-2} \beta_{i-m-1} \Delta^{m-1} u(k) \right\} \\ &\quad - \left\{ \Delta^{i-2} y(k) + \sum_{m=1}^{i-2} \beta_{i-m-1} \Delta^{m-1} u(k-1) \right\} \\ &= \Delta^{i-1} y(k+1) + \sum_{m=1}^{i-2} \beta_{i-m-1} \Delta^m u(k) \end{aligned} \qquad (17)$$

From (17)

$$\begin{aligned} \Delta^{i-1} y(k+1) &= x_{i-1}(k+1) - x_{i-1}(k) \\ &\quad - \sum_{m=1}^{i-2} \beta_{i-m-1} \Delta^m u(k) \end{aligned} \qquad (18)$$

In (11), the state equation of the ith state variable is defined as

$$x_i(k+1) = \Delta^{i-1} y(k+1) + \sum_{m=0}^{i-2} \beta_{i-m-1} \Delta^m u(k) \qquad (19)$$

then by substituting (18) into (19),

$$x_i(k+1) = x_{i-1}(k+1) - x_{i-1}(k) + \beta_{i-1} u(k) \qquad (20)$$

By using (20) recursively,

$$\begin{aligned} x_i(k+1) &= x_{i-2}(k+1) - x_{i-2}(k) - x_{i-1}(k) \\ &\quad + \beta_{i-2} u(k) + \beta_{i-1} u(k) \\ &\quad \vdots \\ &= x_1(k+1) - x_1(k) - x_2(k) - \ldots - x_{i-1}(k) \\ &\quad + \beta_1 u(k) + \beta_2 u(k) + \ldots + \beta_{i-1} u(k) \end{aligned}$$

Using (16),

$$\begin{aligned} x_i(k+1) &= \sum_{m=1}^{i} a_m x_m(k) - \sum_{m=1}^{i-1} x_m(k) + b_0 u(k) \\ &\quad + \sum_{m=1}^{i-1} \beta_m u(k) \qquad \text{for } 2 \leq i \leq N \end{aligned} \qquad (21)$$

In a matrix form, the state-difference equations of the free model in (16) and (21) is then transformed into the following linear system:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) \end{aligned} \qquad (22)$$

where

$$A = \begin{bmatrix} a_1 & a_2 & \cdots & a_N \\ a_1\text{-}1 & a_2 & \cdots & a_N \\ \vdots & \vdots & \ddots & \vdots \\ a_1\text{-}1 & a_2\text{-}1 & \cdots & a_N \end{bmatrix}, \; B = \begin{bmatrix} b_0 \\ b_0 + \beta_1 \\ \vdots \\ b_0 + \beta_1 + \cdots + \beta_{N-1} \end{bmatrix},$$

$$C = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}$$

$$x(k) = [x_1(k) \; x_2(k) \; \cdots \; x_N(k)]^T$$

In this paper, the LQR technique is applied to the free model to design a power system stabilizer. The object of the LQR design is to determine the optimal control law u which can transfer the system from its initial state to the final state such that a given performance index is minimized. The performance index is given in the quadratic form

$$J = \sum_{k=0}^{\infty} \left(x^T(k) Q x(k) + u^T(k) R u(k) \right) \qquad (23)$$

where Q is positive semi-definite, and R is positive-definite. To design the LQR controller, the first step is to select the weighting matrices Q and R. Then, the feedback gain K can be computed and the closed-loop system responses can be found by simulation. This method has an advantage of allowing all control loops in a multi-loop system to be closed simultaneously, while guaranteeing closed-loop stability. The LQR controller is given by

$$u(k) = -Kx(k) \qquad (24)$$

where K is the constant feedback gain obtained from the solution of the discrete algebraic Ricatti equation:

$$\begin{aligned} K &= \left(B^T S B + R \right)^{-1} B^T S A \\ S &= A^T S A - A^T S B K + C^T Q C \end{aligned} \qquad (25)$$

In a conventional method to design the LQR controller, the controller requires all state variables and often an observer is needed. However, the free-model based realization (22) is observable since all the states are constructed from the input–output data via (5). Therefore, an observer is not required for state feedback control. Since the realization is linear, any linear controller design method can be used.

5. SPSA BASED FREE MODEL APPROXIMATION

The free model concept is applied to design a PSS for a one-machine infinite-bus (OMIB) power system (Pai et al., 1997). For the OMIB power system, the q-axis generator model, the static excitation, and turbine and governor models are used. Three simulation tasks are conducted: first, torque angle deviation is simulated in a normal load condition. Second, torque deviation is performed in a heavy load condition. Third, a three-phase fault is considered. All simulations are shown by the comparisons between the CPSS and proposed SPSA based LQR controller. The proposed controller shows the improvement of damping performance for a simple second order free model approximation ($N = 2$). The weight R is 10^{-6} and the Q matrix has elements $Q_{11}=10^6$, $Q_{12}=Q_{21}=0$, $Q_{11}=1$. The initial conditions for simulations are torque angle $d = 0.9767$, the d-q axis stator currents $I_d = 0.6232$ and $I_q = 0.8072$, the d-q axis stator voltages $V_d = 0.4439$ and $V_q = 0.8960$, the internal voltage $E_{pq} = 1.0144$, the field voltage $E_{fd} = 1.5023$, and the reference voltage $V_{ref} = 1.06$.

SPSA Based Free Model Approximation: The system is disturbed by small noise signals. Then, the system input-output pairs are obtained. The system input is the controller output in the CPSS, and the system output is the angular speed (ϖ). The reason to use the angular speed is that the controller is to improve the damping by reducing the coupling effect between the governor system and excitation system since the governor system acts much slower than excitation system. To obtain the system input and output data, first exciter signal, which covers interest bandwidth (say, 1Hz~5Hz), is given with the input signal of conventional PSS. Then, controller input signal and its output signal are taken as input and output for system identification. Fig. 1 shows the comparison between the system output and the SPSA based free model output. The SPSA based free model output is very close to the system output and the root-mean square error is very small as 0.0004106. The coefficients of the second-order free model are a_1=-0.9977, a_2=0.6570, b_0=-0.5331, and b_1=0.5826.

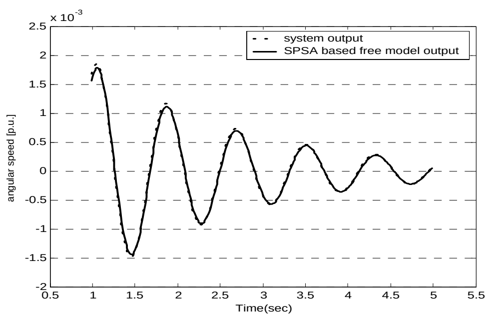

Fig. 1. Comparison of the estimation between the system output and the SPSA based free model output.

Normal Load Condition: In this case, the torque angle is decreased by 0.7767 with P_{load}=1 and Q_{load}=0.2 . Fig. 2 shows the system performance between the CPSS and proposed controller. Faster damping is recognized in the proposed controller case.

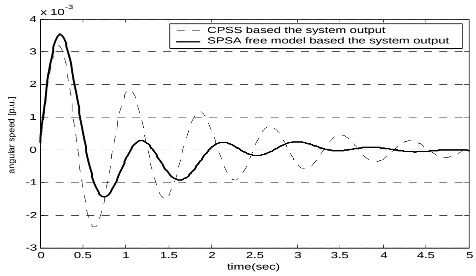

Fig. 2. Comparison of the system output between the CPSS and the proposed controller.

Heavy Load Condition: In this task, the conditions of the torque angle and Q_{load} are the same as the case B. However, to evaluate the heavy load condition, P_{load} is increased by 1.2. Fig. 3 shows the system performance between the CPSS and the proposed controller. The better performance in proposed controller is also shown.

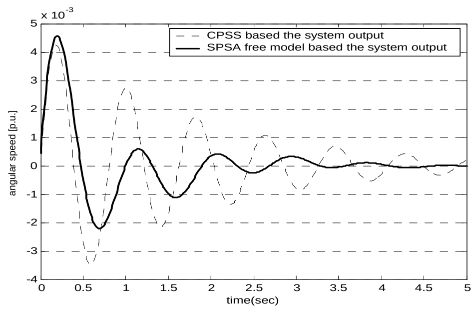

Fig. 3. Comparison of the system output between the CPSS and the proposed controller.

Three-Phased Fault Condition: In this task, a fault is occurred at 1 second, and the fault line is disconnected at 1.04 second. Then, the faulted line is reconnected at 1.1 second. The line impedance is changed to conduct the fault conditions. For example, R=0.12 and $X = 0.2$ during the fault. R=0.6 and X=1 for the removal faulted line. Fig. 4 shows that the faster damping can be recognized in the proposed controller.

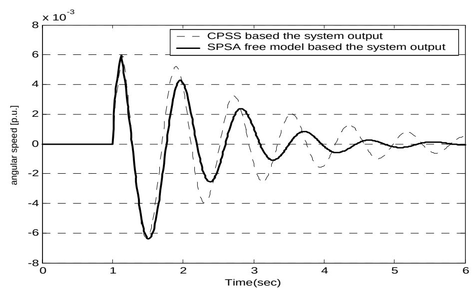

Fig. 4. Comparison of the system output between the CPSS and the proposed controller.

Therefore, Figs. 2, 3, and 4 show that the proposed SPSA based LQR controller is robust for a wide range of operating conditions

6. **CONCLUSION**

This paper presented the SPSA based free-model approximation for system identification using input and output data and its application to the design of a PSS. The SPSA method is used to find the parameters of the free model. The free model is then transformed to a linear state space model and the LQR technique is used to design a PSS. The SPSA based LQR controller was implemented in a one-machine infinite-bus power system. The proposed controller was tested in various operating conditions and compared with the conventional PSS. In all cases, the proposed controller out-performed the conventional PSS and thus demonstrated the usefulness of the SPSA based LQR controller. For multi-machine power system case, same procedure can be applied. Each machine can be excited by its own excite signal. In the future, a multi-machine power system will be presented.

REFERENCES

Anderson, B.D.O. and J.B. More (1990). *Linear Optimal Control*, Prentice Hall, New Jersey.

DeMello, F.P. and C.A. Concordia (1969). Concept of synchronous machine stability as affected by excitation control. *IEEE Trans. PAS*, **103**, 316–319.

Doi, S.A. (1984). Coordinated synthesis of power system stabilizers in multimachine power systems. *IEEE Trans. PAS*, **103**, 1473–1479.

El-Metwally, K.A. and O.P. Malik (1996). Fuzzy logic power system stabilizer. *IEE Proc. Generation Trans. Distribution*, **143**, 263–268.

Ghosh, A., G. Ledwich, O.P. Malik, and G.S. Hope (1984). Power system stabilizer based on adaptive control techniques. *IEEE Trans. PAS*, **103**, 1983–1989.

Gu, W. and K.E. Bollinger (1989). A self-tuning power system stabilizer for wide-range synchronous generator operation. *IEEE Trans. PWRS*, **4**, 1191–1199.

Law, K.T., D.J. Hill, and N.R. Godfrey (1994). Robust controller structure for coordinate power system voltage regulator and stabilizer design. *IEEE Trans. Control Sys. Tech.*, **2**, 220–232.

Pai, M.A., C.D. Vournas, A.N. Michel, and H. Ye (1997). Application of interval matrices in power system stabilizer design. *Int. J. Elec. Power Energy Syst.*, **19**, 179-184.

Phillips, C. and H.T. Nagle (1997). *Digital Control System Analysis and Design*, Prentice Hall.

Sauer, P.W. and M.A. Pai (1998). *Power System Dynamics and Stability*, Prentice Hall, New Jersey.

Spall, J.C. (2003). *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control*, Wiely-Interscience.

Zhang, Y., O.P. Malik, G.S. Hope, and G.P. Chen (1994). Application of an inverse input/output mapped ANN as a power system stabilizer. *IEEE Trans. Energy Convers.*, **9**, 433–441.

APPENDIX

One-machine infinite-bus power system is shown in the one-line diagram in Fig. A1, and a conventional power system stabilizer is presented in Fig. A2. The $q-$axis model is used for generator-turbine system (Pai et al., 1997).

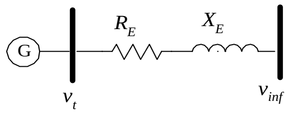

Fig. A1. A one-machine infinite-bus power system.

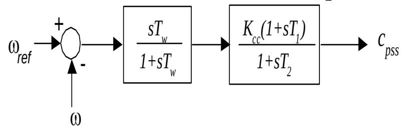

Fig. A2. A conventional PSS for comparison.

<u>Generator-Turbine</u>:

$$\frac{d\delta_i}{dt} = \omega_b(\omega_i - \omega_0)$$

$$M_i \frac{d\omega_i}{dt} = (T_{m_i} - P_{e_i} - D_i(\omega_i - \omega_0))$$

$$T'_{d0_i} \frac{dE'_{q_i}}{dt} = (E_{fd_i} - E'_{q_i} - (x_{d_i} - x'_{d_i}) I_{d_i})$$

$$T_{C_i} \frac{dT_{m_i}}{dt} = (F_{hp_i} U_{g_i} - T_{m_i} + T_{mr_i})$$

$$v_d = x_q i_q,\ v_q = e'_q - x'_d i_d,$$

$$v_t^2 = v_d^2 + v_q^2,\ T_e \cong v_d i_d + v_q i_q$$

<u>Network equation</u>:

$$Zi = (1 + ZY)v_t - v_0$$

$$\begin{bmatrix} R & -X \\ X & R \end{bmatrix} \begin{bmatrix} i_d \\ i_q \end{bmatrix} = \begin{bmatrix} C_1 & -C_2 \\ C_2 & C_1 \end{bmatrix} \begin{bmatrix} v_d \\ v_q \end{bmatrix} - v_0 \begin{bmatrix} \sin\delta \\ \cos\delta \end{bmatrix}$$

$$C_1 = 1 + RG - XB,\ C_2 = XG + RB$$

<u>AVR and exciter</u>:

$$T_{Ai} \frac{dE_{fdi}}{dt} = (K_{Ai}(V_{refi} - V_i + U_{pssi}) - E_{fdi})$$

<u>Governor(GOV)</u>:

$$T_{g_i} \frac{dU_{g_i}}{dt} = (K_{g_i}(\omega_{refi} - \omega_i) - U_{g_i})$$

<u>Generator parameters in p.u.</u>:

$$x_d = 0.973 \quad x'_d = 0.19 \quad x_q = 0.55 \quad T'_{d0} = 7.76$$

$$M = 9.26 \quad D = 0.01 \quad F_{hp} = 1 \quad T_c = 0.1$$

<u>AVR and GOV parameters</u>:

$$K_A = 25 \quad T_A = 0.05 \quad K_g = 10 \quad T_g = 0.1$$

<u>Transmission line parameters in p.u.</u>:

$$R_E = 0.03 \quad X_E = 0.5$$

<u>Constants of a conventional PSS for comparison</u>:

$$T_1 = 0.685 \quad T_2 = 0.1 \quad T_w = 3 \quad K_{cc} = 7.091$$