Proceedings of International Joint Conference on Neural Networks, Orlando, Florida, USA, August 12-17, 2007

Parameter Optimization of PSS Based on Estimated Hessian Matrix from Trajectory Sensitivities

Seung-Mook Baek, Jung-Wook Park, and Ganesh K. Venayagamoorthy

Abstract—**This paper describes the optimal tuning for the output limits of the power system stabilizer (PSS), which can improve the system damping performance immediately following a large disturbance. The non-smooth nonlinear parameters such as the saturation limits of the PSS cannot be tuned by the conventional methods based on linear approaches. To implement the systematic optimal tuning for the output limits of the PSS, a feedforward neural network (FFNN) is applied to the hybrid system model based on the differential-algebraic- impulsive-switched (DAIS) structure. The FFNN is firstly designed to identify the trajectory sensitivities obtained from the DAIS structure. Thereafter, it estimates the second-order derivatives of an objective function J, which is used during iterations of optimization process. The performance of the optimal output limits tuned by the proposed method is evaluated by applying a large disturbance to a power system.**

I. Introduction

THE dynamic behaviors of the power system stabilizer (PSS) are affected by the linear parameters (gain and time constant of phase compensator) with smoothness and the constrained parameter (output limits) with non-smooth nonlinearities. The appropriate selection of linear parameters has been usually made by using the conventional tuning techniques [1]-[4] based on the small signal stability analysis. However, by focusing only on small signal conditions, the dynamic damping performance immediately following a large disturbance is often degraded. The PSS output limits (which cannot be determined by the linear approach) can provide the solution to balance theses competing effects. In particular, these limit values attempt to prevent the machine terminal voltage from falling below the exciter reference level while speed is also falling, which means that it can improve the reduced transient recovery after disturbance (faster recovery to its initial steady state points, therefore, it allows to save system energy), especially in multi-machine power system [1].

On the other hands, power systems frequently exhibit interactions between continuous-time dynamics, discrete-time, discrete-event dynamics, switching action, and jump phenomenon. Such systems are known generically as *hybrid systems* [5]-[6], which can be modeled by a set of *differential-algebraic-impulsive-switched* (DAIS) structure [7]. Especially, this hybrid system model provides the effective and insightful analysis of the PSS with non-smooth nonlinear dynamics due to saturation limits.

For the systematic optimal tuning of the output limits of the PSS, a feedforward neural network (FFNN) is applied to the hybrid system model based on the DAIS structure. It is designed to adaptively identify the first-order derivatives of an objective function **J** (which are available from the DAIS structure) with respect to the saturation limits. Then, the FFNN estimates the second-order derivatives by using the backpropagation algorithm [8] to form the approximated Hessian matrix, which is used during optimization process.

The performances of nonlinear parameters optimized by the proposed method are evaluated on a single-machine infinite bus (SMIB) power system by the time-domain simulation.

II. Hybrid System Presentation

As mentioned in Section I, the hybrid systems, which include power systems, are characterized by the continuous and discrete states, continuous dynamics, discrete events or triggers, and mappings that define the evolution of discrete states at events.

In other words, the hybrid system is a mathematical model of physical process consisting of an interacting continuous and discrete event system [7], [9]. A formal presentation of the hybrid system is given in [9], where a general hybrid dynamical system is defined as H = [Q, Σ, A, G] and

- Q is the set of discrete states;
- $\Sigma = \{\Sigma_q\}_{q\in Q}$ is the collection of dynamical systems $\Sigma_q = [X_q, \Gamma_q, f_q]$ where each X_q is an arbitrary topological space forming the continuous state space of Σ_q, Γ_q is a semigroup over which the states evolve, and f_q generates the continuous state dynamics;
- $A = \{A_q\}_{q\in Q}$, $A_q \subset X_q$ for each $q\in Q$, is the collection of autonomous jump sets, i.e., the conditions which trigger jumps;
- $G = \{G_q\}_{q\in Q}$, where $G_q: A_q \rightarrow S = \cup_{q\in Q} (X_q \times \{q\})$ is the autonomous jump transition map. The hybrid state-space of H is given by S.

Manuscript received January 31, 2007.

This work was supported by the Seoul Research and Business Development (R&BD) program under Grant No. 10988.

Seung-Mook Baek and Jung-Wook Park are with the School of Electrical and Electronic Engineering, Yonsei University, Seoul 120-749, Korea (corresponding author to provide phone: +82-2-2123-5867; Fax: +82-2-313-2879; e-mail: jungpark@yonsei.ac.kr).

G.K. Venayagamoorthy is with the Real-Time Power and Intelligent Systems Laboratory, the Department of Electrical and Computer Engineering, University of Missouri-Rolla, MO 65409-0249, USA (e-mail: gkumar@ieee.org).

The above level of abstraction of the general hybrid system does not suit the implementation of numerical optimization method described in this study, for which the trajectory sensitivities can be exploited efficiently. A hybrid model with the DAIS structure, which is more conductive to such analysis, can be presented without loss of generalities as follows.

$$\dot{\underline{x}} = \underline{f}(\underline{x}, y) \tag{1}$$

$$0 = g(\underline{x}, y), \tag{2}$$

$$0 = \begin{cases} g^{(i-)}(\underline{x}, y) & y_{d,i} < 0, \\ g^{(i+)}(\underline{x}, y) & y_{d,i} > 0, \end{cases} \quad i = 1, \cdots, d, \tag{3}$$

$$\underline{x}^+ = \underline{h}_j(\underline{x}^-, y^-) \quad y_{e,j} = 0, \quad j \in \{1, \cdots, e\}, \tag{4}$$

where:

$$\underline{x} = \begin{bmatrix} x \\ z \\ \lambda \end{bmatrix}, \quad \underline{f} = \begin{bmatrix} f \\ 0 \\ 0 \end{bmatrix}, \quad \underline{h}_j = \begin{bmatrix} x \\ h_j \\ \lambda \end{bmatrix},$$

$\underline{x} \in X \subseteq \Re^{\underline{n}}, y \in Y \subseteq \Re^m, z \in Z \subseteq \Re^l, \lambda \in L \subseteq \Re^p,$

and x are the continuous dynamic states, for example generator angles, speed, and fluxes; z are discrete dynamic states such as transformer tap positions and protection relay logic states; y are algebraic states, e.g. load bus voltage magnitudes and angles; λ are parameters such as generator reactance, controller gains, switching times, and limit values.

The reset equations $\underline{h}_j$ in (4) ensure that x and λ remain constant at reset events, but the dynamics states z are reset to new values according to $z^+ = h_j(\underline{x}^-, y^-)$. The algebraic function g in (2) is composed of $g^{(0)}$ together with appropriate choices of $g^{(i-)}$ or $g^{(i+)}$, depending on the signs of the corresponding elements of y_d in (3). An event is triggered by an element of y_d changing sign and/or an element of y_e in (4) passing through zero. In other words, at an event, the composition of g changes and/or elements of z are reset. Then, the system flows ϕ are defined accordingly as

$$\phi(\underline{x}_0, t) = \begin{bmatrix} \phi_{\underline{x}}(\underline{x}_0, t) \\ \phi_y(\underline{x}_0, t) \end{bmatrix} = \begin{bmatrix} \underline{x}(t) \\ y(t) \end{bmatrix} \tag{5}$$

More detailed explanation and associated mathematical equations of the DAIS model (especially for the switching and impulse effects) are given in [7] with the comprehensive studies of the hybrid system.

Example-1 (*A switched hybrid system*)

This example is taken from [7]. The system description is

$\dot{x} = A_i x$,

where $A_1 = \begin{bmatrix} 1 & -100 \\ 10 & 1 \end{bmatrix}$, $A_2 = \begin{bmatrix} 1 & 10 \\ -100 & 1 \end{bmatrix}$

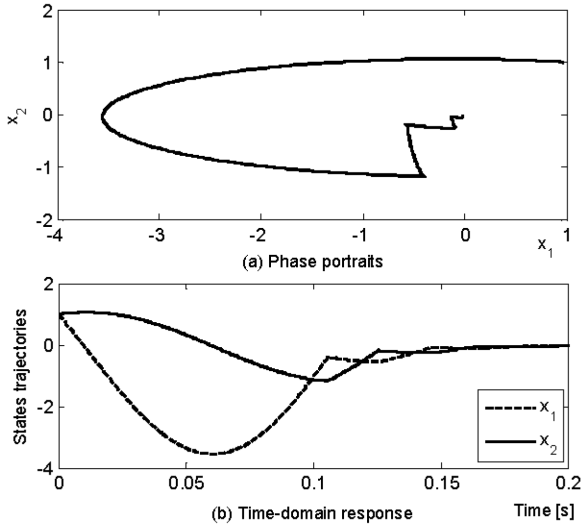
The index i (of matrix A_i) changes from 1 to 2 when $x_2 = 2.75 \cdot x_1$ and from 2 to 1 when $x_2 = 0.36 \cdot x_1$. Initially, $x_0 = [1\ 1]^t$ and $i = 1$. This model can be written in the DAIS form in (1) ~ (4) as

$$\dot{x} = \begin{bmatrix} 1 & z_1 \\ z_2 & 1 \end{bmatrix} x$$

$$0 = \begin{cases} \lambda_1 x_1 - x_2 - z_3 y, & y < 0 \\ x_2 - \lambda_2 x_1 - z_3 y, & y > 0 \end{cases}$$

$$\left. \begin{aligned} z_1^+ &= z_2^- \\ z_2^+ &= z_1^- \\ z_3^+ &= -z_3^- \end{aligned} \right\}, \quad \text{when} \quad y = 0.$$

where $\underline{x}_0 = [x_0\ z_0\ \lambda]^t = [1\ 1\ -100\ 10\ 1\ 2.75\ 0.32]^t$ ($x_0 = [1\ \ 1]^t$, $z_0 = [-100\ \ 10\ \ 1]^t$, $\lambda = [2.75\ \ 0.32]^t$) and $y_0 = -1$. The phase portrait and the time-domain response are shown in Fig. 1.



(a) Phase portraits

(b) Time-domain response

Fig. 1. A switched hybrid system.

The change between A_1 and A_2 in the above example is achieved by resetting the matrix elements z_1 and z_2 whenever a switch is saturated. The switching saturations are given by the algebraic constraints. In other words, the alternating between active switching saturations corresponds to flipping the sign of z_3. This example gives a good illustration for the fact that the two unstable sub-systems (the eigenvalues of both A_1 and A_2 are equal to $\lambda = 1 \pm j\omega_0$ where $\omega_0 = \sqrt{1000}$) can make the overall system stable through the proper switching action by the hybrid system modeling, independently of initial states (which means that this hybrid system is guaranteed to be asymptotically stable).

III. Nonlinear Controller Optimization

A. *Trajectory Sensitivities*

Trajectory sensitivity provides a way of quantifying the changes in the flow (5) that result from (small) changes to parameters and/or initial conditions. The development of these sensitivity concepts will be based on the DAIS model in

(1)~(4). Trajectory sensitivities follow from the Taylor series expansion (neglecting higher order terms) of the flows $\phi_{\underline{x}}$ and ϕ_y in (5), which can be expressed as

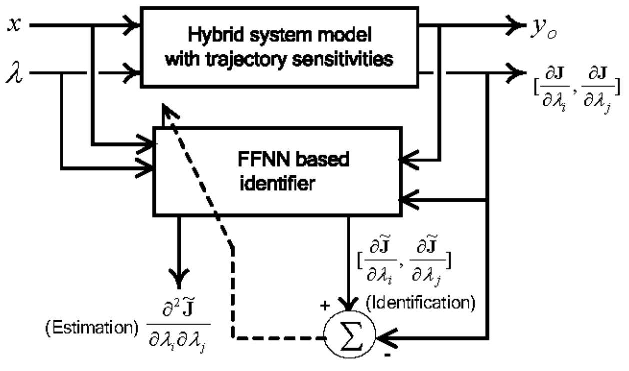
$$\Delta\underline{x}(t) = \Delta\phi_{\underline{x}}(\underline{x}_0, t) \approx \frac{\partial\phi_{\underline{x}}(\underline{x}_0, t)}{\partial\underline{x}_0}\Delta\underline{x}_0 \equiv \mathbf{\Gamma}_{\underline{x}_0}(t)\Delta\underline{x}_0 \tag{6}$$

$$\Delta y(t) = \Delta\phi_y(\underline{x}_0, t) \approx \frac{\partial\phi_y(\underline{x}_0, t)}{\partial\underline{x}_0}\Delta\underline{x}_0 \equiv \mathbf{\Gamma}_y(t)\Delta\underline{x}_0 \tag{7}$$

where $\mathbf{\Gamma}_{\underline{x}_0} \in \Re^{\underline{n}\times\underline{n}}$ and $\mathbf{\Gamma}_y \in \Re^{m\times\underline{n}}$ are partial derivatives matrices of system flows and known as the trajectory sensitivities. Recall that $\underline{x}_0$ incorporates parameters λ, therefore, the sensitivities to initial conditions $\underline{x}_0$ include parameter sensitivities.

B. The FFNN Identifier for Hessian Matrix Estimation

As shown in Fig. 2, the FFNN identifier is applied to accurately identify the dynamics of the hybrid system, which are the trajectory sensitivities in (6) and (7). Thereafter, it validates the system model with its converged weights and estimates the second-order derivatives of a user-defined objective function **J** with respect to the nonlinear parameters λ to be optimized. The FFNN (with the multilayer perceptron structure [8]) consists of three-layers of neurons (which are the input, hidden, and output layer) interconnected by the weight matrices $\mathbf{W}_l$ and $\mathbf{W}_L$ (see the Fig. 2 in [8]), and it is first trained to identify the dynamics of the plant until its weights are converged with sufficient accuracy. In other words, the FFNN starts with random initial values for its weights, and then, computes a one-pass backpropagation algorithm at each time step *k*, which consists of a forward pass propagating the input vector through the network layer by layer, and a backward pass to update the weights with the error between [$\partial\mathbf{J}/\partial\lambda_i$, $\partial\mathbf{J}/\partial\lambda_j$] and [$\partial\tilde{\mathbf{J}}/\partial\lambda_i$, $\partial\tilde{\mathbf{J}}/\partial\lambda_j$] shown in Fig. 2.



x
λ
Hybrid system model with trajectory sensitivities
y_o
$[\frac{\partial\mathbf{J}}{\partial\lambda_i}, \frac{\partial\mathbf{J}}{\partial\lambda_j}]$
FFNN based identifier
$[\frac{\partial\tilde{\mathbf{J}}}{\partial\lambda_i}, \frac{\partial\tilde{\mathbf{J}}}{\partial\lambda_j}]$
(Identification)
+
-
Σ
(Estimation) $\frac{\partial^2\tilde{\mathbf{J}}}{\partial\lambda_i\partial\lambda_j}$

Fig. 2. FFNN applied to the hybrid system.

The objective function **J** required during the optimization process is normally defined as a positive and quadratic form, therefore, its derivative exists. With the identified gradient $\nabla\tilde{\mathbf{J}}$ in Fig. 2, one component of the second-order derivatives $\nabla^2\tilde{\mathbf{J}}$ can be estimated by the one-step backpropagation computation in (8). In the similar manner, the associated Hessian matrix $\tilde{\mathbf{H}}$ is expressed by (9).

$$\frac{\partial^2\tilde{\mathbf{J}}}{\partial\lambda_i\partial\lambda_j} = \frac{\partial\nabla\tilde{\mathbf{J}}_i}{\partial p_L}\frac{\partial p_L}{\partial q_L}\frac{\partial q_L}{\partial p_l}\frac{\partial p_l}{\partial q_l}\frac{\partial q_l}{\partial\lambda_j} = \left\{s(q_l)\left(1 - s(q_l)\right)\mathbf{W}_l(\lambda_j)\right\}\sum_{j=1}^{m_l}\nabla\tilde{\mathbf{J}}_i\cdot\mathbf{W}_L \tag{8}$$

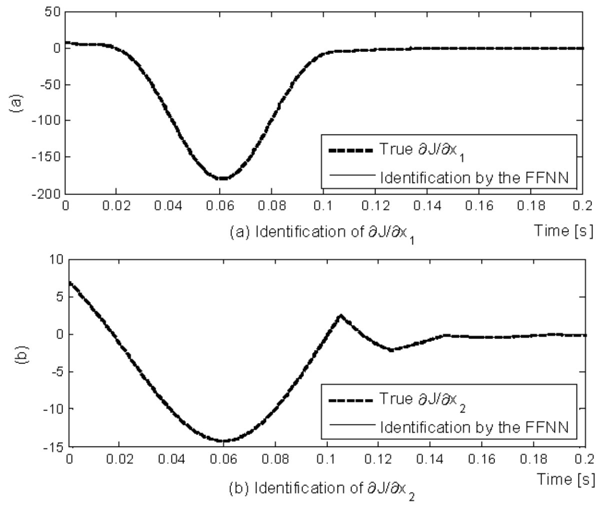
where m_l is the number of neurons in the hidden layer; *p* is the output of the activation function for a neuron; *q* is the regression vector as the activity of a neuron; **W** is the weight matrix; *L* and *l* denote the output and hidden layer, respectively; $\nabla\tilde{\mathbf{J}}$ is the value identified by FFNN; the function *s* in (8) is the sigmoidal function given as $s(x) = 1/\{1+\exp(-x)\}$.

$$\tilde{\mathbf{H}}(\lambda) = \begin{bmatrix} \partial^2\tilde{\mathbf{J}}/\partial\lambda_i^2 & \partial^2\tilde{\mathbf{J}}/\partial\lambda_i\partial\lambda_j \\ \partial^2\tilde{\mathbf{J}}/\partial\lambda_j\partial\lambda_i & \partial^2\tilde{\mathbf{J}}/\partial\lambda_j^2 \end{bmatrix} \tag{9}$$

Example-2 (Estimation of Hessian by the FFNN)

For the hybrid system in Example-1 (see Section II), it is assumed that the FFNN identifier is designed to identify the objective function $\mathbf{J} = (x_1^4 + x_2^3 + 4x_1x_2)$ and its partial derivatives with respect to states x_1 and x_2, which are $\partial\mathbf{J}/\partial x_1 = (4x_1^3 + 4x_2)$ and $\partial\mathbf{J}/\partial x_2 = (3x_2^2 + 4x_1)$.

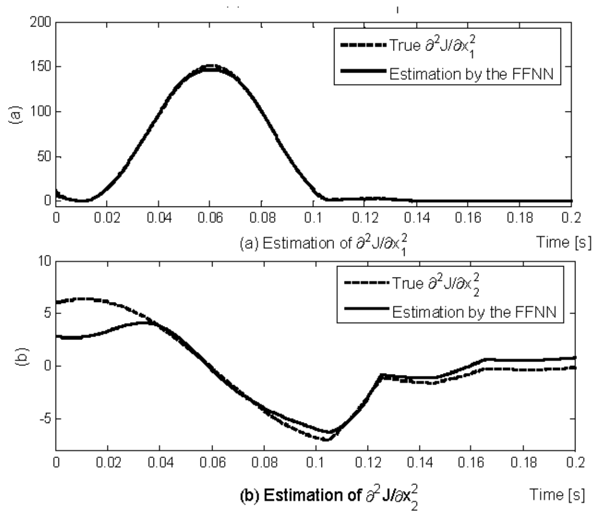
After taking the necessary steps (training→testing→fixed weights), the identification performance of the first-order derivatives by the FFNN with sufficient accuracy is shown in Fig. 3. Thereafter, it estimates the second-order derivatives, which are $\partial^2\mathbf{J}/\partial x_1^2 = 12x_1^2$ and $\partial^2\mathbf{J}/\partial x_2^2 = 6x_2$ by using (8). The results are shown in Fig. 4. The FFNN approximates the second-order partial derivatives of this hybrid system with the acceptable accuracy, which can be used to apply the numerical optimization technique described in the next sub-section.



(a) Identification of ∂J/∂x1

(b) Identification of ∂J/∂x2

Fig. 3. Performance of identification by the FFNN.


(a) Estimation of $\partial^2 J/\partial x_1^2$

(b) Estimation of $\partial^2 J/\partial x_2^2$

Fig. 4. Performance of estimation by the FFNN.

C. Implementation of Optimal Tuning Applied to PSS Design

The Hessian matrix estimated by the FFNN identifier is now applied to implement the optimal tuning of the PSS output limits (V_{max} and V_{min} shown in Fig. 5), which are the non-smooth nonlinear parameters. The generator (**G**) of the SMIB system in Fig. 6 is accurately represented by a six-order machine model, viz., a two axis (*d-q*) model with two damper windings in each axis [10]. Figure 5 shows the control block diagram of the PSS and automatic voltage regulator (AVR)/Exciter.

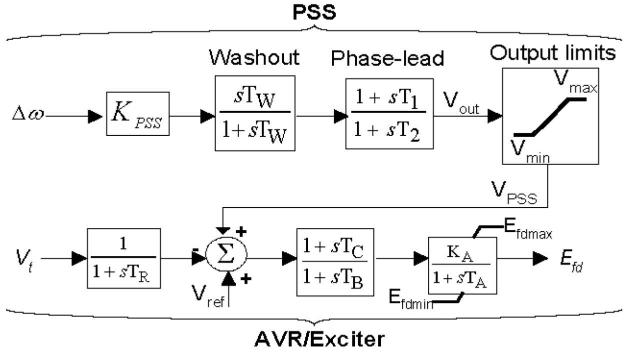


Fig. 5. AVR/PSS block representation.

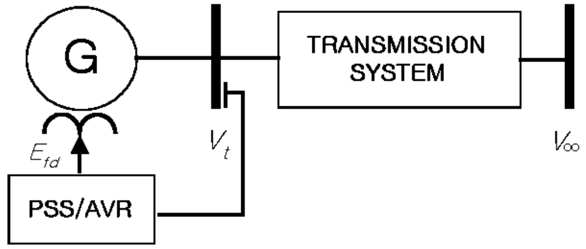


Fig. 6. Single-machine infinite bus (SMIB) system.

The output (clipping) tunable limits on the PSS output V_{pss} and the anti-windup limits (these are physical limits, so are not tunable) on the field voltage E_{fd} in Fig. 5 introduce events that can be captured by the DAIS model. In other words, the event occurs when a controller signal saturates in the response to the large inputs ($\Delta\omega$ and V_t) due to the disturbance. This indicated phenomenon is implemented by the DAIS structure as given in (10) for the PSS clipping limits.

$$
\begin{aligned}
&y_1 = V_{max} - V_{out};\\
&y_2 = V_{out} - V_{min};\\
&0 = \begin{cases} g_1^{(i-)}(\underline{x}, y) = V_{PSS} - V_{max} & y_1 < 0,\\ g_2^{(i-)}(\underline{x}, y) = V_{PSS} - V_{min} & y_2 < 0,\\ g_1^{(i+)}(\underline{x}, y) = g_2^{(i+)}(\underline{x}, y) = V_{PSS} - V_{out} & y_1 > 0, y_2 > 0. \end{cases}
\end{aligned}
\tag{10}
$$

Many practical optimization problems can be formulated by using a Bolza form of the objective function **J**

$$\min_{\lambda,\, t_f} \mathbf{J}(\underline{x}, y, \lambda, t_f) \tag{11}$$

subject to $[\underline{x}(t)\ y(t)]^{\mathrm{T}} = \phi(\underline{x}_0, t)$ in (5), and

$$\mathbf{J} = \varphi\big(\underline{x}(t_f), y(t_f), \lambda, t_f\big) + \int_{t_0}^{t_f} \psi\big(\underline{x}(t), y(t), \lambda, t\big) dt. \tag{12}$$

where λ are the optimized parameters (output limits of the PSS), which are adjusted to minimize the value of objective function **J**, and t_f is the final time. Also, φ is the cost or penalty associated with the error in the terminal state at time t_f, and ψ is the cost function associated with transient state errors. The objective of the PSS is to mitigate system damping and force the system to recover to the post-disturbance stable operating point as quickly as possible. The speed deviation ($\Delta\omega$) and terminal voltage deviation (ΔV_t) of the generator in Fig. 6 are considered as good assessments of the damping and recovery [1]. Therefore, the objective function **J** in (12) can be re-formulated for the optimal tuning of the PSS with specific final time t_f as the following

$$\mathbf{J}(\lambda) = \int_{t_0}^{t_f} \left(\begin{bmatrix} \omega(\lambda, t) - \omega^s \\ V_t(\lambda, t) - V_t^s \end{bmatrix}^T \mathbf{V} \begin{bmatrix} \omega(\lambda, t) - \omega^s \\ V_t(\lambda, t) - V_t^s \end{bmatrix} \right) dt, \tag{13}$$

where **V** is the diagonal matrix with weighting factors. The ω^s and V_t^s are the post-fault steady state values of ω and V_t, respectively. Minimization of the value of **J** in (13) is straightforward even though the cost is obtained by integrating over the system flows (trajectories). The simplest way of obtaining **J** is to introduce a new state variable $\underline{x}_{\text{cost}}$, with $\dot{\underline{x}}_{\text{cost}}$ equal to the integrand of (13). Thereafter, $\underline{x}_{\text{cost}}(t_f) = \mathbf{J}$. The trajectory sensitivities in (6) with respect to λ directly provide the gradient by

$$\nabla\mathbf{J} = \mathbf{\Gamma}_{\underline{x}_{\text{cost}}}(t_f). \tag{14}$$

With the use of $\nabla\mathbf{J}$ obtained from (14) as the inputs, the FFNN consists of the total 7 inputs (threshold input of 1, $\Delta\omega$, ΔV_t, $V_{\max}$, $V_{\min}$, $\partial\mathbf{J}/\partial V_{\max}$, $\partial\mathbf{J}/\partial V_{\min}$) and 10 neurons in the hidden layer. Then, it generates (identifies) two outputs $\partial\tilde{\mathbf{J}}/\partial V_{\max}$ and $\partial\tilde{\mathbf{J}}/\partial V_{\min}$ as shown in Fig. 2. Thereafter, it computes the estimated Hessian $\tilde{\mathbf{H}}$ by using (8) and (9) for the nonlinear parameters λ, which are the upper and lower limits of the PSS, with its converged weights. During the optimization process, these parameters λ are updated by using (15) at each iteration *k*.

$$\lambda_{k+1} = \lambda_k + \alpha \cdot \tilde{\mathbf{H}}^{-1}(\lambda) \cdot \nabla\mathbf{J}(\lambda) \tag{15}$$

where α is the step-length to ensure that the optimal path (search) is the descent direction vector.

IV. Simulation Results

A. Hessian Matrix Estimation by the FFNN

To evaluate the performance of identification and estimation by the FFNN, the SMIB system in Fig. 6 is disturbed by applying a 200 ms three-phase short circuit fault with the fault-impedance of 0.05 pu to the generator terminal bus at 0.05 s. The results in Fig. 7 show the very good identification performances of the FFNN identifier for the $\partial\tilde{\mathbf{J}}/\partial V_{\max}$ and $\partial\tilde{\mathbf{J}}/\partial V_{\min}$, at the particular iteration (which is the tenth iteration) with the time duration of 0 s to 5 s (t_f) in the optimization process. Also, Fig. 8 shows the results of the corresponding second-order derivatives estimation, $\partial^2\tilde{\mathbf{J}}/\partial V_{\max}^2$ and $\partial^2\tilde{\mathbf{J}}/\partial V_{\min}^2$, which are the diagonal components of the estimated Hessian matrix $\tilde{\mathbf{H}}$ in (9) at the tenth iteration. The off-diagonal components of $\tilde{\mathbf{H}}$ are obtained by the same manner.

B. Performances of Convergence

During the optimization process, the convergence performances of the proposed method are shown in Fig. 9 with comparison of those by the steepest descent algorithm [11] for the value of the objective function **J** and maximum relative gradient $f_{r\text{-}gradient}$ in (16).

$$f_{r-gradient} = \left\| \frac{\text{relative rate of change in } \mathbf{J}}{\text{relative rate of change in } \underline{x}} \right\|_\infty = \left\| \frac{\nabla\mathbf{J}(\underline{x}_k)\cdot\underline{x}_k}{\mathbf{J}(\underline{x}_k)} \right\|_\infty \tag{16}$$

It is clearly shown from Fig. 9 that the proposed method by the estimated Hessian improves the convergence speed very effectively. In other words, the values of both **J** and $f_{r\text{-}gradient}$ are almost converged with only 5 iterations when the tuning process is implemented by the proposed method. The nonlinear parameters λ by the steepest descent method are updated by using (17). For the fair comparison, the same value of the fixed step-length α (= 0.22) is used in both methods.

$$\lambda_{k+1} = \lambda_k + \alpha \cdot \nabla\mathbf{J}(\lambda) \tag{17}$$

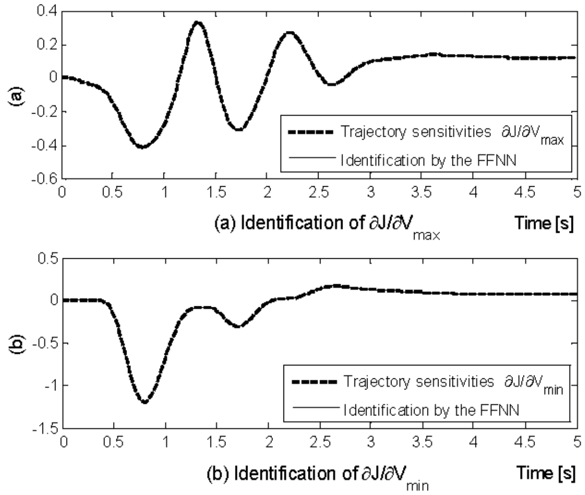


Fig. 7. Performance of the identification by the FFNN.

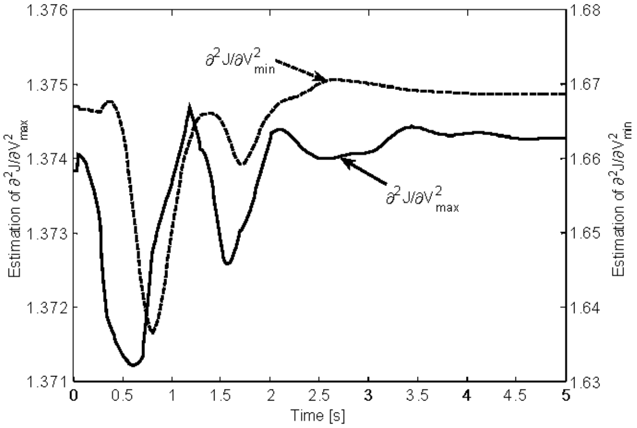


Fig. 8. The second-order derivatives estimated by the FFNN.

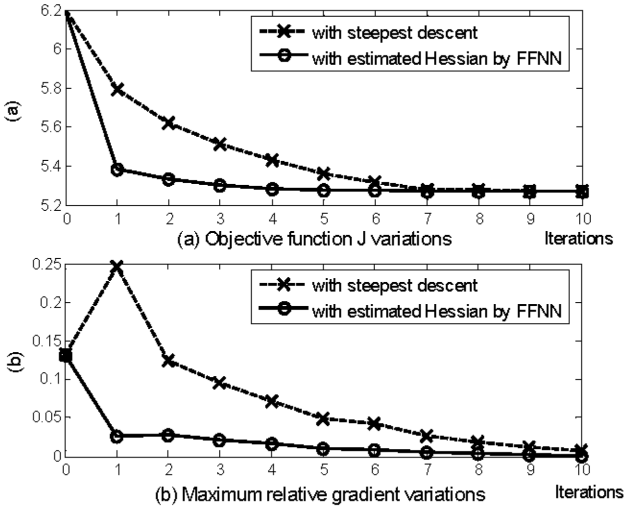


Fig. 9. Comparisons of performances (convergence property).

C. Damping Performance by Optimized Parameters

The damping performance of the output limits (which are [0.1152 –0.3135] for [$V_{\max}$ $V_{\min}$]) of the PSS optimized after 5 iterations is compared with that of the initial output limits

([0.1 –0.1]) and other possible limits ([0.2 –0.2]) in Figs. 10 and 11.

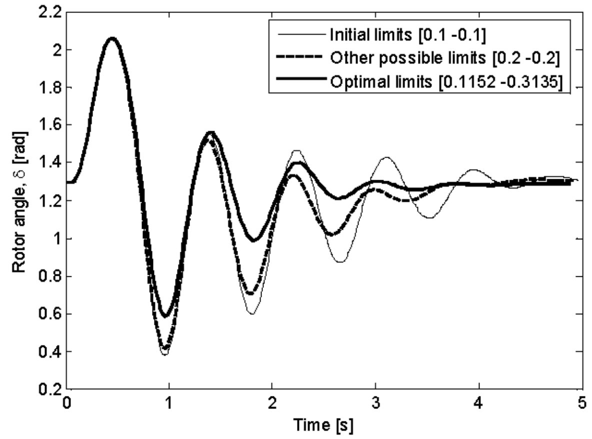


Fig. 10. Generator rotor angle response [rad].

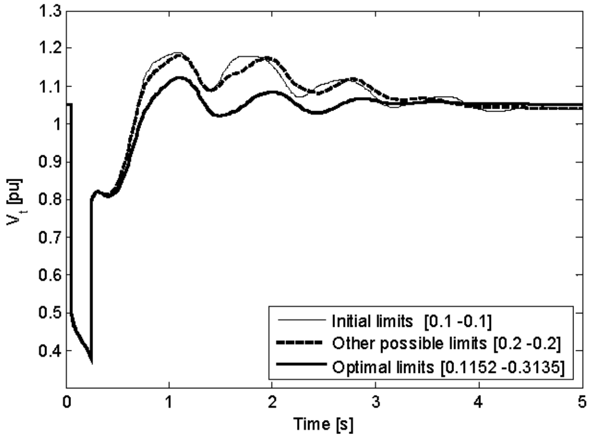


Fig. 11. Generator terminal voltage response [rad].

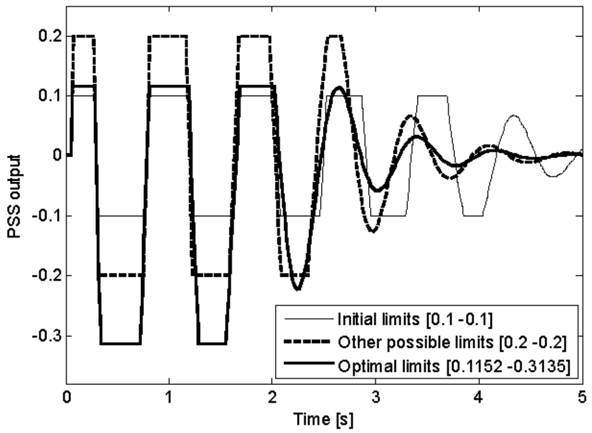


Fig. 12. PSS output response.

It is clearly verified that the optimal saturation limits determined by the proposed method improve the low-frequency oscillation damping and transient terminal voltage response effectively. The value of V_{max} has been changed a litter from 0.1 to 0.1152, but the value of V_{min} has moved significantly from –0.1 to –0.3135. Note that the manual tuning would likely not even search in that direction for improved response. The effect of optimal tuning for these saturation limits is rather dramatic and quite evident for a large disturbance (such as a three-phase short circuit) applied to a power system. The corresponding PSS output response (V_{PSS} in Fig. 5) in Fig. 12 exhibits the non-smooth nonlinear dynamic behaviors. The effectiveness of the proposed method for the systematic optimal tuning has been validated on the IEEE benchmark four-machines, two-area test system. Due to page limitation, the results will be shown in the authors' other paper.

V. Conclusions

This paper made the new contribution by applying the feedforward neural network (FFNN) to the hybrid system modeling with the differential-algebraic-impulsive-switched (DAIS) structure to estimate the Hessian matrix used for the nonlinear parameter optimization of the power system stabilizer (PSS). Trajectory sensitivities obtained from the DAIS structure were used to estimate the second-order derivatives by the FFNN. The optimized output limits of the PSS, which are the non-smooth nonlinear parameters, improved the damping performance of low-frequency oscillations effectively. The proposed method based on the estimated Hessian matrix provides the much faster convergence properties in the systematic optimal tuning problem especially when compared to the steepest descent method using only the first-order derivatives information.

References

[1] Prabha Kundur, ed., Power System Stability and Control, EPRI Editors, McGraw-Hill, Inc. 1993.

[2] P. Kundur, M. Klein, G.J. Rogers, and M.S. Zywno, "Application of power system stabilizers for enhancement of overall system stability," *IEEE Trans. on Power Systems*, vol.4, no.2, pp. 614-626, May 1989.

[3] M. Klein, G. J. Rogers, S. Moorty, and P. Kundur, "Analytical investigation of factors influencing power system stabilizers performance", *IEEE Trans. on Energy Conversion*, vol. 7, no. 3, pp. 382-388, September 1992.

[4] N. Martins and L. T. G. Lima, "Determination of suitable locations for power system stabilizers and static VAr compensators for damping electromechanical oscillations in large scale power systems", in *Proc. of Power Industry Computer Application*, pp.74-82, May 1989.

[5] A. van der Schaft and H. Schumacher, An Introduction to Hybrid Dynamical Systems, Springer-Verlag, London, 2000.

[6] D. Liberzon, Switching in Systems and Control, Birkhauser, Boston, 2003.

[7] Ian A. Hiskens, "Trajectory sensitivity analysis of hybrid systems", *IEEE Trans. on Circuits and Systems-Part I: Fundamental Theory and Applications*, vol.47, no.2, pp. 204-220, February 2000.

[8] Jung-Wook Park, G.K. Venayagamoorthy, and R.G. Harley, "MLP/RBF Neural-Networks-Based Online Global Model Identification of Synchronous Generator", *IEEE Transactions on Industrial Electronics*, Vol.52, No. 6, pp. 1685-1695, December 2005.

[9] M. S. Branicky, V. S. Borkar, and S. K. Mitter, "A unified framework for hybrid control: Model and optimal control theory", *IEEE Trans. on Automat. Contr.,* vol. 43, pp. 31-45, January 1998.

[10] P. W. Sauer and M. A. Pai, ed., Power System Dynamics and Stability. Englewood Cliffs, NJ: Prentice-Hall, 1998.

[11] J. Nocedal and S. J. Wright, ed., Numerical Optimization, Springer-Verlag, New York, 1999.