Dynamic System Eigenvalue Extraction using a Linear Echo State Network for Small-Signal Stability Analysis – a Novel Application

Jiaqi Liang, Jing Dai, Ganesh K. Venayagamoorthy, and Ronald G. Harley

***Abstract*— A large nonlinear dynamic system usually has complex dynamic modes corresponding to the system's eigenvalues. These eigenvalues govern the system's local behavior and thus are critical information for designing system operation and control strategies. Without the availability of the system's analytical model, which is often the case for large nonlinear systems, the system's eigenvalues need to be estimated. A linear echo state network (ESN) based method for extracting observable eigenvalues of a dynamic system together with the participation factors of these eigenvalues in the accessible system states is presented in this paper. A linear ESN is first trained to track the dynamic system's local responses under injected small perturbation signals. The dynamic system's eigenvalues are then extracted from the ESN's weight matrices. Given the merit of fast training of ESNs, the ESN can be quickly retrained once the system operating point changes, and the system eigenvalues can be reestimated. Application of the proposed eigenvalue extraction method in the power system small-signal analysis is presented to demonstrate the effectiveness of the proposed method.**

I. Introduction

ARTIFICIAL neural networks (ANNs) are well-known effective universal function approximators. Different types of ANNs have been proposed in the past few decades and broadly used in nonlinear system modeling and intelligent neural control [1]. The multi-layer perceptron (MLP) network and the radial basis function (RBF) network have relatively simple structures, but their approximation abilities are limited by their feed-forward structures. Recurrent neural networks (RNNs) have advantages in modeling dynamic systems due to their internal dynamic memory [2]. However, the training algorithms for RNNs usually have high computational complexity and long training cycles. Recently, the echo state network (ESN) [3] has been introduced as a special type of RNN, which has the same dynamic modeling capability as a RNN but requires a simpler training process. ESNs have been successfully applied to the areas of dynamic system modeling and neural adaptive controls [4]-[7]. However, most of the previous works on dynamic system identification have focused on how to train an ANN such that it tracks a certain trajectory or minimizes a certain performance index, but seldom has anyone investigated the information buried inside an ANN after it has been trained.

On the other side, a complex nonlinear dynamic system (plant) being modeled by an ANN often does not have an accurate analytical model for designers to study its dynamic characteristics and stability margins. This characteristic information, such as system eigenvalues (or modes), however, is often critical for system operators to maintain the system within a safe stability margin as well as designing system controllers. Existing methods to estimate a dynamic system's observable modes are based on directly analyzing the frequency components in the system outputs, for example, Prony's method [8]. However, if the system has multiple outputs and multiple observable modes, this method does not provide information on the coupling between different system modes and states, i.e., the participation factors.

This paper proposes a linear ESN (an ESN with linear activation function) based method to extract observable eigenvalues of a dynamic system as well as the participation factors of these eigenvalues in the accessible system states. For nonlinear system identification, nonlinear activation functions are necessary. However, eigenvalues of a nonlinear dynamic system only represent its local dynamics around its operating point, where the nonlinear system behaves like a linear system. Therefore, a linear ESN is sufficient to extract this information. A brief overview of the ESN structure and its training algorithms, as well as the mathematical simplicity of a linear ESN, is presented in Section II. The proposed method for dynamic system observable eigenvalue extraction is discussed in Section III. Section IV provides further discussion on finding the participation factors between the extracted system eigenvalues and the system accessible states. The application of the proposed method to evaluate the small-signal stability of a power system is presented in Section V.

II. Overview of Echo State Networks

This section gives a brief overview of ESNs. More detailed descriptions can be found in [3],[9]-[12].

A. Structure of ESNs

The structure of an ESN is shown in Fig. 1, where the output of the internal hidden layer (dynamic reservoir) depends on the network input vector at time k, $\boldsymbol{u}(k)_{(n \times 1)}$, the internal state (hidden layer output) vector at time k, $\boldsymbol{x}(k)_{(q \times 1)}$, and the network output vector at time k, $\boldsymbol{y}(k)_{(m \times 1)}$. Note that time k here does not denote the k^{th} execution of the ESN, but represents the k^{th} sampling instant of all signals. $W_{i\ (q \times n)}$

Manuscript received May 2, 2010. This work was supported in part by the National Science Foundation, USA, under Grants ECCS 0802047 and EFRI 0836017.

J. Liang, J. Dai and R. G. Harley are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA. R. G. Harley is also Professor Emeritus and an Honorary Research Associate, University of KwaZulu-Natal, Durban, South Africa. (e-mail: jliang@gatech.edu; jingdai@gatech.edu; rharley@ece.gatech. edu)

G. K. Venayagamoorthy is the Director of the Real-Time Power and Intelligent Systems Laboratory, Missouri University of Science and Technology, Rolla, MO 65409, USA (e-mail: gkumar@ieee.org).

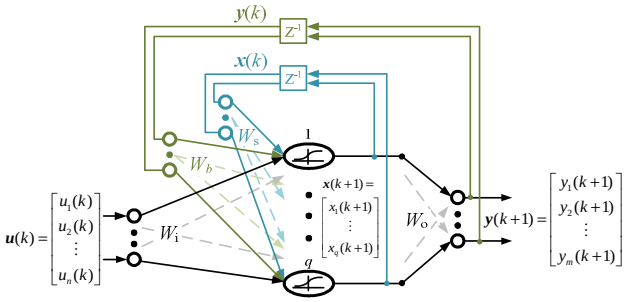

Fig. 1. Structure of an ESN.

denotes the input weight matrix, $W_{b\ (q\ x\ m)}$ denotes the output feedback weight matrix, $W_{s\ (q\ x\ q)}$ denotes the state feedback weight matrix, and $W_{o\ (m\ x\ q)}$ denotes the output weight matrix. The ESN in Fig. 1 can be mathematically expressed as

$$\boldsymbol{x}(k+1) = f\,[W_{\mathrm{i}}\,\boldsymbol{u}(k) + W_{\mathrm{s}}\,\boldsymbol{x}(k) + W_{\mathrm{b}}\,\boldsymbol{y}(k)], \tag{1}$$

$$\boldsymbol{y}(k+1) = W_{\mathrm{o}}\,\boldsymbol{x}(k+1), \tag{2}$$

where the activation function, $f(\cdot)$, is an element-wise operation that can be chosen as a linear function, or a nonlinear function such as the sigmoid function and the tansig function.

W_{i}, W_{b} and, W_{s} are generated randomly beforehand and they are fixed during the training process. W_{o} is the only matrix that will be updated during the network training. The random generation of W_{s} follows these specific steps [9]:

1) Randomly generate a sparse matrix, $W_{\mathrm{s\_init}}$, with all elements between -1 and 1.

2) Normalize $W_{\mathrm{s\_init}}$ into a matrix, W_{s}', with unity spectral radius:

$$W_{\mathrm{s}}' = W_{\mathrm{s\_init}} / |\,\lambda_{\max}\,|, \tag{3}$$

where $\lambda_{\max}$ is the spectral radius of $W_{\mathrm{s\_init}}$, i.e., the maximum absolute eigenvalue of $W_{\mathrm{s\_init}}$.

3) W_{s} is obtained as

$$W_{\mathrm{s}} = \alpha W_{\mathrm{s}}', \tag{4}$$

where $\alpha < 1$, and usually α is selected between 0.9 and 1.

B. Training Algorithms for ESNs

Both offline and online training algorithms have been developed for training the ESN [9]-[11]. Fig. 2 shows the general scheme for training an ESN to identify the plant dynamics, and the objective of the training is to minimize the error, $\boldsymbol{e}$(k), between the ESN and the plant outputs.

The offline training algorithm is based on the pseudo-inverse method [9], [10]. First, a set of plant input and output samples are taken, and then a set of over-determined equations are formed with W_{o} unknown (the number of data samples is usually much more than the number of unknown weights in W_{o}). A solution for W_{o} with minimum mean squared error is obtained by performing a pseudo-inverse operation. Therefore, the offline training is fast and it leads to an optimal output weight matrix with respect to the sample data set.

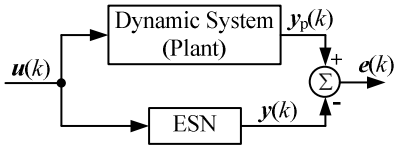

Fig. 2. General scheme of plant dynamics identification.

The online training algorithm continuously updates the output weight matrix based on the error signal and the gradient descent algorithm [11]. The online training algorithm offers a method for tracking the variations of the plant dynamics, but the training cycle typically takes more time than the offline training algorithm.

C. ESN with Linear Activation Function

If a linear function is used as the activation function in an ESN, then the ESN can be described by

$$\boldsymbol{x}(k+1) = a[W_{\mathrm{i}}\,\boldsymbol{u}(k) + W_{\mathrm{s}}\,\boldsymbol{x}(k) + W_{\mathrm{b}}\,\boldsymbol{y}(k)], \tag{5}$$

$$\boldsymbol{y}(k+1) = W_{\mathrm{o}}\,\boldsymbol{x}(k+1), \tag{6}$$

where a is a constant. Since a can be absorbed by the output matrix W_{o}, without loss of generality, a is set to 1 and the resulting ESN is called a linear ESN in this paper. Substitute (6) at time k into (5), to obtain the following equation:

$$\begin{aligned}\boldsymbol{x}(k+1) &= W_i\boldsymbol{u}(k) + W_s\boldsymbol{x}(k) + W_bW_o\boldsymbol{x}(k)\\ &= W_A\boldsymbol{x}(k) + W_i\boldsymbol{u}(k)\end{aligned}, \tag{7}$$

where $W_{\mathrm{A}}=W_{\mathrm{s}}+W_{\mathrm{b}}W_{\mathrm{o}}$, which is the state update matrix of the discrete-time dynamic system (7), i.e. the linear ESN. The modes of the linear ESN are completely represented by the eigenvalues of W_{A}. Combine (6) and (7), and the linear ESN is described by

$$\boldsymbol{y}(k+1) = W_oW_A\boldsymbol{x}(k) + W_oW_i\boldsymbol{u}(k)\,. \tag{8}$$

When a linear ESN is trained to track the outputs of a nonlinear dynamic system under small perturbation signals, the system's local dynamics around its operating point are learned by the linear ESN, i.e., the linear ESN learns the locally linearized dynamic system. Note that the linear ESN is in the discrete-time domain, and thus it indeed learns the time-discretized version of the dynamic system. It will be shown in Section III.B that a time-discretized locally linearized dynamic system can also be represented in the same analytical form as (8).

III. Dynamic System Eigenvalue Extraction

A. Small-Signal Analysis of a Dynamic System

A multiple-input multiple-output (MIMO) time-invariant dynamic system without feed-through outputs can be described in the state equation form, as in

$$\begin{aligned}\dot{\boldsymbol{x}}_p(t) &= \boldsymbol{g}[\boldsymbol{x}_p(t), \boldsymbol{u}(t)]\\ \boldsymbol{y}_p(t) &= \boldsymbol{h}[\boldsymbol{x}_p(t)]\end{aligned}, \tag{9}$$

where $\boldsymbol{u}(t)$ is the input vector to the MIMO system, $\boldsymbol{x}_p(t)$ is the system internal state vector, and $\boldsymbol{y}_p(t)$ is the system output vector. The overall stability of such a nonlinear system is usually difficult to analyze and prove. However, the small-signal stability of the linearized system, which is necessary for the overall system to be asymptotically stable at the operating point, is often of interest. The locally linearized system of (9) around an operating point ($\boldsymbol{u}^*$, $\boldsymbol{x}_p^*$, $\boldsymbol{y}_p^*$) is given by

$$\begin{aligned} \Delta\dot{\boldsymbol{x}}_p(t) &= A\cdot\Delta\boldsymbol{x}_p(t) + B\cdot\Delta\boldsymbol{u}(t) \\ \Delta\boldsymbol{y}_p(t) &= C\cdot\Delta\boldsymbol{x}_p(t) \end{aligned}, \tag{10}$$

where $\boldsymbol{u}(t)=\Delta\boldsymbol{u}(t)+\boldsymbol{u}^*$, $\boldsymbol{x}_p(t)=\Delta\boldsymbol{x}_p(t)+\boldsymbol{x}_p^*$, $\boldsymbol{y}_p(t)=\Delta\boldsymbol{y}_p(t)+\boldsymbol{y}_p^*$, $A=\left.\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}_p}\right|_{\boldsymbol{u}=\boldsymbol{u}^*,\ \boldsymbol{x}_p=\boldsymbol{x}_p^*}$, $B=\left.\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{u}=\boldsymbol{u}^*,\ \boldsymbol{x}_p=\boldsymbol{x}_p^*}$, and $C=\left.\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_p=\boldsymbol{x}_p^*}$.

The locally linearized system (10) is stable, i.e., the nonlinear system (9) is small-signal stable, if and only if all eigenvalues of matrix A have real parts less than zero. For a small-signal stable nonlinear system, the closer an eigenvalue lies to the imaginary axis, the less stable margin the system would have for the corresponding mode. In the real world, the analytical model of a complex dynamic system is often unknown. Therefore, the direct eigenvalue analysis on a complex system is often not possible.

However, the system modes, or part of the modes, can be observed from the system output, $\Delta\boldsymbol{y}_p(t)$, given that the observability condition is satisfied. Therefore, these observable system modes can be estimated from the system outputs, and the small-signal analysis can then be performed on these observable modes.

B. System Discretization in Time

Before introducing the method of system eigenvalue extraction using a linear ESN, it is necessary to first introduce the time-discretized version of (10), because the linear ESN learns in the discrete time domain. The continuous-time system (10) can be discretized [13] to

$$\begin{aligned} \Delta\boldsymbol{x}_p(k+1) &= A_d\Delta\boldsymbol{x}_p(k) + B_d\Delta\boldsymbol{u}(k) \\ \Delta\boldsymbol{y}_p(k+1) &= C_d\Delta\boldsymbol{x}_p(k+1) \end{aligned}, \tag{11}$$

where $A_d = \mathcal{L}^{-1}\{(sI-A)^{-1}\}_{t=Ts}$ and $\mathcal{L}^{-1}\{\cdot\}_{t=Ts}$ is the Laplace inverse transform with the sampling time being T_s, $B_d = A^{-1}(A_d-I)B$, and $C_d = C$. An approximation of A_d exists based on the bilinear transform which preserves the stability of the continuous-time system [13], given by

$$A_d = (I+\frac{1}{2}AT_s)(I-\frac{1}{2}AT_s)^{-1}. \tag{12}$$

Note that a smaller sampling time step results in a better approximation by using (12). The modes of the dicretized locally linearized system (11) are completely represented by the eigenvalues of A_d. System (11) can be further written as

$$\Delta\boldsymbol{y}_p(k+1) = C_dA_d\Delta\boldsymbol{x}_p(k) + C_dB_d\Delta\boldsymbol{u}(k), \tag{13}$$

which has the exact same form as the linear ESN in (8). Note that the linearization and time discretization of the nonlinear system (9) are not assumptions. Under small perturbations and in discrete time domain, the nonlinear system (9) behaves exactly like (11).

Therefore, when a linear ESN is trained to model the nonlinear dynamic system (9) under small perturbations, the modeling task is indeed to find out the output weight matrix, W_o, such that (8) and (13) respond as closely as possible. Note that $\boldsymbol{x}(k)$ in (8) and $\Delta\boldsymbol{x}_p(k)$ in (13) may have different dimensions and different values.

C. System Eigenvalue Extraction using Linear ESN

The idea of the proposed method is to reconstruct the internal local dynamics of an unknown nonlinear system by training a linear model to follow the system's local responses and then analyzing the eigenvalues of the trained model. As discussed above, the small-signal stability of a nonlinear system is based on its local dynamics (the linearized system), and thus the reconstruction can be achieved by using a linear dynamic ANN, for example the linear ESN.

Because the dynamics of a linear ESN and the local dynamics of the nonlinear system (plant) are completely represented by the eigenvalues of W_A in (8) and the eigenvalues of A in (10) respectively, if the linear ESN response tracks the plant response closely around a local region, then the plant's observable eigenvalues should be shared by the linear ESN. However, because the linear ESN is in the discrete-time domain, the trained linear ESN indeed contains the observable eigenvalues of A_d in (11). In order to find the observable eigenvalues of A, it is then necessary to find the continuous-time version of the linear ESN matrix W_A.

Once the linear ESN is trained, by solving (12), the continuous-time version of W_A is given by

$$W_{A\_cts} = \frac{2}{T_s}(W_A - I)(W_A + I)^{-1}. \tag{14}$$

The eigenvalues of W_{A_cts} then contain the plant's observable eigenvalues. If the plant's operating point changes, the ESN can be retrained as necessary, and a new set of eigenvalues can be obtained for the new operating point.

Because the number of the linear ESN internal states is usually larger than the number of the plant's observable eigenvalues, there are some extra eigenvalues from W_{A_cts} that do not correspond to any of the plant modes. However, since the linear ESN output dynamics depends mostly on those observable plant modes, the extra eigenvalues are usually well damped and they do not have much effect on the ESN outputs. Therefore, those eigenvalues of the linear ESN that are closer to the imaginary axis are more likely to be the plant modes. Another way of distinguishing the plant modes from the extra eigenvalues is to use the participation factors, which is

introduced in the next section. The eigenvalues of the linear ESN that have high participation factors in the ESN outputs are more likely to be the plant modes.

IV. Participation Factor Extraction

The extraction of the participation factors between different observable modes and different accessible states is discussed in this section. The accessible states are defined as plant states that are also plant outputs. For the analysis below, part of the plant outputs are assumed to be plant states. This assumption is often true, because most measured quantities in an actual system cannot change abruptly and thus they can be chosen as state variables.

A. Participation Factor

The definition of participation factor is briefly reviewed below. The plant states $\Delta \boldsymbol{x}_p$ in (10) can be decoupled by the plant's modal matrix U [14], where the columns of U are the eigenvectors of matrix A in (10) with unity norms, i.e.,

$$AU = U\Lambda, \tag{15}$$

where Λ = diag ($\lambda_1, \lambda_2 \ldots, \lambda_n$) and λ_i's are the eigenvalues of A. Define a new set of state variables, $\boldsymbol{z}_p$, as

$$\boldsymbol{z}_p(t) = U^{-1}\Delta \boldsymbol{x}_p(t). \tag{16}$$

Then the plant dynamics described by (10) can be rewritten as

$$\dot{\boldsymbol{z}}_p(t) = \Lambda \cdot \boldsymbol{z}_p(t) + U^{-1}B \cdot \Delta \boldsymbol{u}(t). \tag{17}$$

From (17), each state in $\boldsymbol{z}_p$ is only related to one eigenvalue and thus only related to one system mode. The original state vector is related to $\boldsymbol{z}_p$ by (16). For example, the k^{th} state in $\Delta \boldsymbol{x}_p$ is related to the i^{th} element in $\boldsymbol{z}_p$ by a factor of u_{ki}. Define a matrix V as

$$V = U^{-1}. \tag{18}$$

The rows of V are then the left eigenvectors of A with the same eigenvalue matrix Λ. The participation factor of the i^{th} mode in the k^{th} state of $\Delta \boldsymbol{x}_p$ is defined as [14]

$$p_{ki} = u_{ki} v_{ik}. \tag{19}$$

In the case of complex eigenvalues and eigenvectors, the normalized participation factor p_{ki} is given by

$$p_{ki} = \frac{|u_{ki} v_{ik}|}{\sum_{j=1}^{q} |u_{kj} v_{jk}|}. \tag{20}$$

The participation factor p_{ki} is a measurement of the relative coupling between the k^{th} state and the i^{th} mode. It is useful when a damping controller for increasing damping of the i^{th} mode is desired [15].

B. Participation Factor Extraction using Linear ESN

Suppose that the first r plant outputs are the plant states and they are distinct (r is less than the total number of the plant states). As a result from the linear ESN modeling, the first r outputs from a trained linear ESN are estimates of the plant states, and the linear ESN has part of the eigenvalues that are corresponding to the observable plant modes. The goal is then to find the participation factors between the plant observable modes and the r accessible states.

In order to find out these participation factors, the first r outputs from the trained linear ESN is first transformed to be part of the linear ESN states by a linear transformation defined as

$$T = \begin{bmatrix} W_{o,1:r} \\ 0 \quad I_{(p-r)\times(p-r)} \end{bmatrix}_{p\times p}, \tag{21}$$

where $W_{o,1:r}$ denotes the first to the r^{th} rows of W_o, which are linear independent due to the assumption that the r plant states are distinct. If T is full rank, which is true in most cases, a new set of linear ESN states, $\boldsymbol{x}^{new}(k)$, is defined as

$$\boldsymbol{x}^{new}(k) = T \cdot \boldsymbol{x}(k) = \begin{bmatrix} y_1(k) \\ \vdots \\ y_r(k) \\ x_{r+1}(k) \\ \vdots \\ x_q(k) \end{bmatrix}. \tag{22}$$

With this new set of states, the linear ESN described by (6) and (7) is then rewritten as

$$\begin{aligned} \boldsymbol{x}^{new}(k+1) &= (TW_A T^{-1}) \cdot \boldsymbol{x}^{new}(k) + TW_i \boldsymbol{u}(k) \\ \boldsymbol{y}(k+1) &= \begin{bmatrix} I_{r\times r} & 0 \\ W_{o,\ r+1:m} \end{bmatrix} \boldsymbol{x}^{new}(k+1) \end{aligned}. \tag{23}$$

Define $W_A{}^{new} = TW_A T^{-1}$, which has the same eigenvalues as W_A, i.e., they have the same modes. After finding the continuous-time version of $W_A{}^{new}$ from (14) and calculating its right eigenvector matrix U, the participation factors between the first r states in $\boldsymbol{x}^{new}$, i.e., $y_1, y_2, \ldots y_r$, and all the modes of the linear ESN are obtained by following the same procedure as described in section IV.A. In other words, the participation factors between the r accessible plant states and the observable plant modes are obtained.

V. Application in Power System Small-Signal Analysis

A electric power grid is a nonlinear complex MIMO dynamic system. Most of its internal states are not accessible and cross-coupled. The dynamic model for a large power system is often not available due to lack of accurate system parameters. However, the system dynamic characteristics, such as eigenvalues around different operating points, are critical information for operating the system within a safe

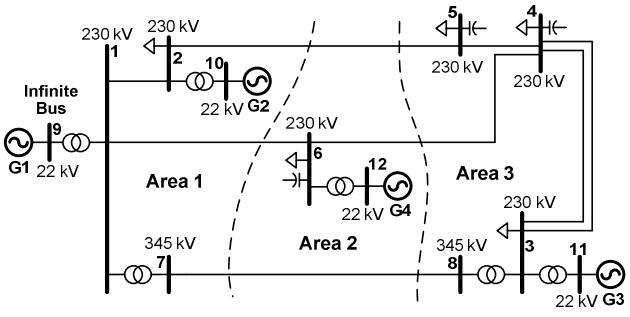

Fig. 3. Benchmark 12 bus power system.

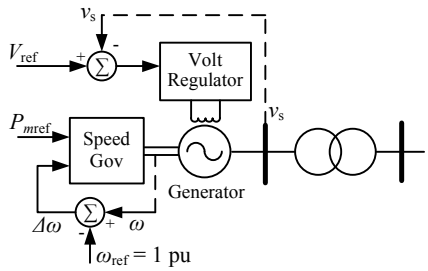

Fig. 4. Block diagram of controller on each generator.

stability margin, as well as allocating and designing system damping controllers. The generator speed oscillation modes in a 12 bus power system [15] are studied in this section as an example application of the proposed dynamic system eigenvalue extraction technique.

A. 12 bus Power System

The one-line diagram [15] of the benchmark 12 bus power system is shown in Fig. 3. It has three detailed modeled generators (G2, G3 and G4) and one voltage source (G1).

Two different models of the same 12 bus system are developed. One detailed dynamic system model, which represents all three generators by their 8th order models, is developed in the PSCAD/EMTDC commercial time domain simulation software. The detailed models for the terminal voltage regulators and generator speed governors are also included, as shown in Fig. 4. This model is used as the plant for generating training data for the linear ESN.

Another simplified system analytical model is developed, which represents each generator as only a 3rd order system and represents each voltage regulator as a 1st order system. The speed governors are neglected due to their relatively slow responses, compared to the generator speed oscillations. The analytical modeling and linearizing method of a power system was discussed in detail in [14] and [15]. In each generator and voltage regulator unit, there are 4 states, including the rotor speed ω, the rotor angle (integral of rotor speed) δ, the internal induced voltage E_q', and the voltage regulator output E_f. This simplified analytical model, with a total of 12 states, is linearized around its steady-state operating point to obtain the system matrix A and its eigenvalues. These eigenvalues are then used to verify the proposed technique.

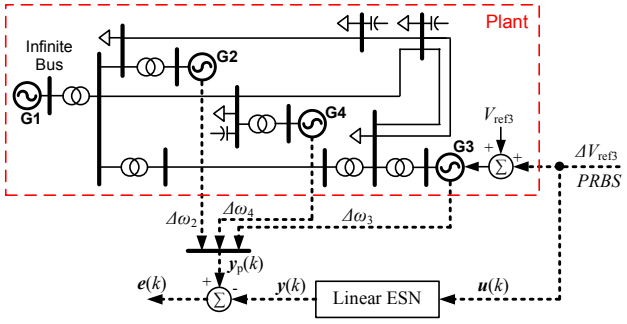

Fig. 5. Plant and linear ESN input-output signals.

TABLE I
CASES 1: GENERATOR SET POINTS

Gen #	MVA Rating	P_{mref} (MW)	V_{ref} (pu)
Gen 2	700	500	1.02
Gen 3	500	200	1.01
Gen 4	500	300	1.02

B. Training of the Linear ESN Model

Fig. 5 shows the input-output signals of the plant and the linear ESN. The plant has three outputs, which are speed deviation signals from the three generators. The generator speed deviations in a power system are often coupled with modes with long time constants, i.e., their corresponding eigenvalues are close to the imaginary axis. Note that all three plant outputs are also the plant states.

A small pseudo-random binary signal (PRBS) is fed into the voltage reference signal of generator 3 to perturb the 12 bus system for generating ESN training data. The PRBS is generated by

$$PRBS(k) = 0.03[r_{0.05}(k) + r_{0.1}(k)]/2 \ \text{(pu)}, \qquad (24)$$

where $r_{0.05}$(k) and $r_{0.1}$(k) are sequences of i.i.d. random numbers uniformly distributed in [-1, 1]. $r_{0.05}$(k) updates at the frequency of 0.05 Hz and $r_{0.1}$(k) updates at the frequency of 0.1 Hz. The maximum value for the PRBS is set to ±0.03 per unit (pu), so that the generator terminal voltage stays between 0.95 pu and 1.05 pu, which is a requirement in an actual power system. A linear ESN is trained to track the plant outputs during this series input PRBS disturbances.

The linear ESN thus has one input (n = 1) and three outputs (m = 3). Thirty internal neurons are used (q = 30), i.e., the linear ESN has 30 eigenvalues. α is set to 0.95 in the linear ESN W_s initialization. Two different cases corresponding to two different operating points of the 12 bus system are studied below. For both cases, the offline training based on pseudo-inverse is used for fast convergence.

C. Case 1: Base Case for Verifying the Proposed Eigenvalue Extraction Technique

In this case, the set points of the three generators, i.e., the reference output powers and the reference terminal votlages, are listed in Table I. From the simplified analytical model, the

TABLE II
CASE 1: OSCILLATION MODES FROM ANALYTICAL MODEL

Eigenvalues	Frequency (Hz)	Damping Ratio (%)
-0.297 ± j7.078	1.13	4.19
-0.102 ± j5.337	0.85	1.91
-0.340 ± j4.738	0.75	7.16

TABLE III
CASE 1: PARTICIPATION FACTORS FROM ANALYTICAL MODEL

State	1.13 Hz Mode	0.85 Hz Mode	0.75 Hz Mode
$\Delta\delta_{G2}$	0.00355	**0.45015**	0.03599
$\Delta\omega_{G2}$	0.00356	**0.45023**	0.03603
$\Delta E_q'_{G2}$	0.00134	0.00907	0.00513
ΔE_{fG2}	0.00053	0.00252	0.00104
$\Delta\delta_{G3}$	**0.45471**	0.00096	0.02553
$\Delta\omega_{G3}$	**0.45503**	0.00096	0.02558
$\Delta E_q'_{G3}$	0.03257	0.00021	0.00835
ΔE_{fG3}	0.00946	0.00007	0.00202
$\Delta\delta_{G4}$	0.01892	0.04115	**0.40328**
$\Delta\omega_{G4}$	0.01893	0.04116	**0.40380**
$\Delta E_q'_{G4}$	0.00088	0.00297	0.04496
ΔE_{fG4}	0.00049	0.00056	0.00828

system's eigenvalues at this operating point are obtained, among which those corresponding to the oscillation modes are listed in Table II. There are three oscillation modes in this system, the 1.13 Hz mode, the 0.85 Hz mode, and the 0.75 Hz mode. Table III shows the participation factors between these three modes and each of the system states from the analytical model. All of the three oscillation modes are primarily coupled with the generator mechanical states, i.e., the rotor speed and angle. Generator 3 dominates in the 1.13 Hz mode, as indicated by the relatively high participation factors of $\Delta\delta_{G3}$ and $\Delta\omega_{G3}$ in the 1.13 Hz mode. Generator 2 dominates in the 0.85 Hz mode, and Generator 4 dominates in the 0.75 Hz mode. However, all of the above information is based on the system analytical model, which is often not available for a large complex real-world system.

The linear ESN is now trained with data generated from the detailed dynamic system PSCAD model. 70 s of training data, as shown in Fig. 6, are generated with a sampling time of 0.02 s (T_s = 0.02). The linear ESN is trained offline by pseudo-inverse with a final mean squared error (MSE) of 2.19e-016, which is 12 orders of magnitude smaller than the $\Delta\omega$ data. After the training, all the weights in the linear ESN are fixed and the network is tested under a different set of PRBS disturbance. The testing results with fixed weights are shown in Fig. 7. Under step change inputs, the trained linear ESN shows good tracking to the dynamic oscillation of the plant, which indicates that the trained linear ESN contains information of the plant oscillation modes.

With the mode extraction technique introduced in section III, all the 30 eigenvalues corresponding to the continuous-time version of the trained linear ESN are listed in Table IV. It can be seen from Table IV that the linear ESN contains the plant's three oscillation modes, with accurate

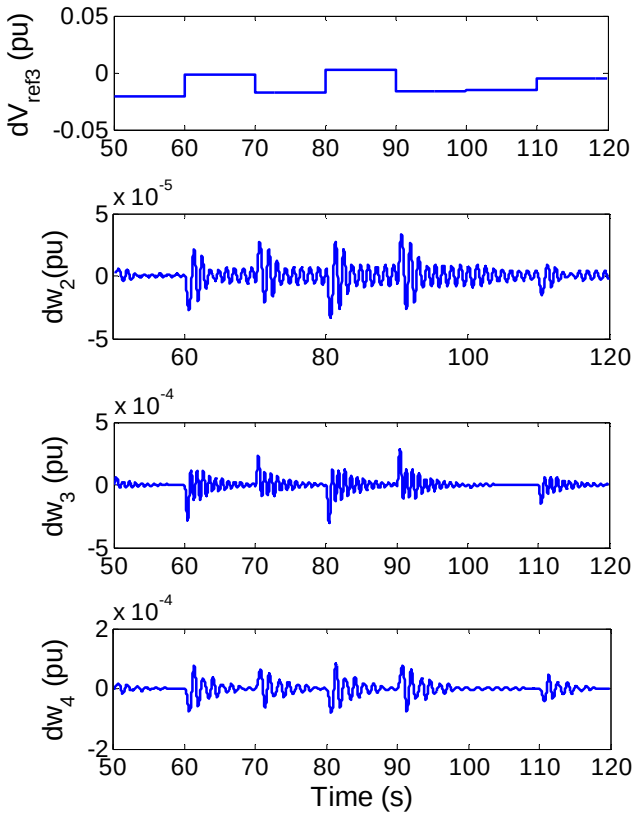

Fig. 6. Linear ESN offline training data in case 1. Top graph: injected PRBS. Bottom three graphs: plant responses under the injected PRBS.

estimation on both the oscillation frequencies and the damping ratios. These three oscillation modes can be separated from the rest of the linear ESN eigenvalues from either the damping ratios or the oscillation frequencies. The other extra eigenvalues of the linear ESN either have a high damping ratio or a high oscillation frequency, which are clearly not the dominant components in the $\Delta\omega$ waveforms.

Table V shows the participation factors between the 30 eigenvalues and the three linear ESN outputs, which are calculated based on the technique introduced in section IV. From the participation factors, the three modes that correspond to the plant oscillation modes clearly stand out by having relatively high participation factors. From Table V, it can also be seen that the 0.75 Hz mode is mostly coupled with generator 4, the 0.84 Hz mode is mostly coupled with generator 2, and the 1.13 Hz mode is mostly coupled with generator 3. The results are consistent with those derived from the analytical model, i.e., Tables II and III.

The proposed technique would be more useful when a certain mode is coupled with more than one state, or one state is coupled with more than one mode. In such a case, the participation factor table could show the couplings between multiple states and multiple modes.

D. Case 2: Same System with a Different Operating Point

In this case, the power generations from all three generators are increased to reduce the energy demand from the voltage

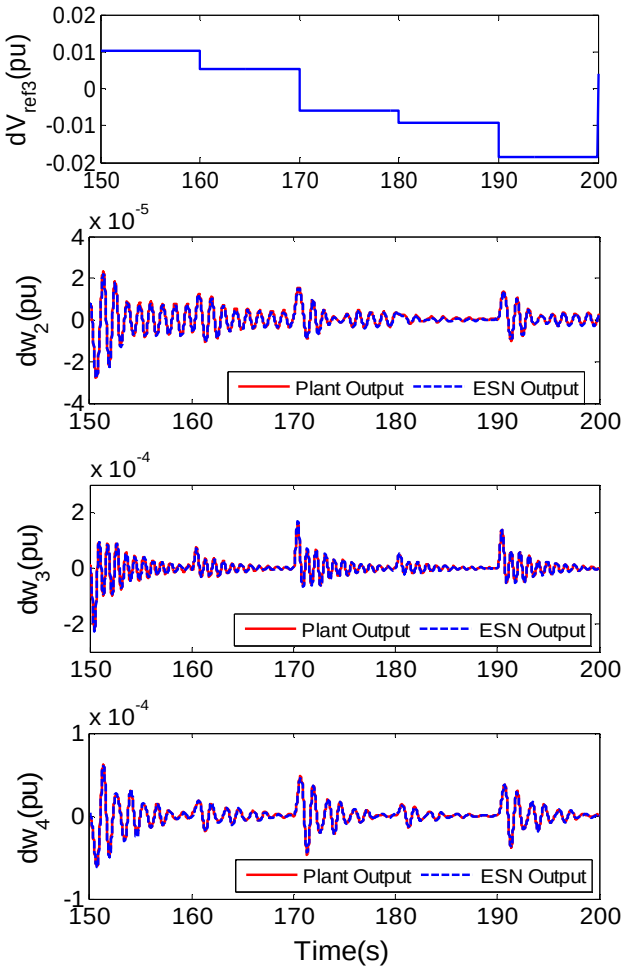

Fig. 7. Testing results of the trained linear ESN (fixed weights) in case 1. Top graph: injected PRBS. Bottom three graphs: plant and the linear ESN responses under the injected PRBS.

source (G1). The set points in this case are listed in Table VI.

From the analytical model, the plant oscillation modes are calculated and listed in Table VII for comparison with results extracted from the linear ESN. After changing the system operating point, both the oscillation frequency and damping ratio of the mode coupled with G3 increase, while the damping ratios of the modes coupled with G2 and G4 decrease.

70 s of new data are collected at the new operating point under the PRBS perturbation. When using the weights trained in case 1, the MSE increases to 3.41e-014 under this new data set. This is primarily because the change of the plant operating point also changes the eigenvalues of the plant. The linear ESN is then retrained with the new data. After training, the MSE drops back to 2.64e-016. The retrained linear ESN with fixed weights is tested again at the new operating point. Fig. 8 shows the testing results, which reveal good tracking performance of the retrained linear ESN.

The system oscillation modes are extracted again from the retrained linear ESN, as shown in Table VIII. Comparing Tables VII and VIII, the modes extracted from the linear ESN are accurate. Note that by monitoring the oscillation modes of the 12 bus system using the linear ESN, it can be seen that generator 2 is close to instability at the new operating point.

TABLE IV
CASE 1: ALL 30 EIGENVALUES OF THE LINEAR ESN

Eigenvalues	Frequency (Hz)	Damping Ratio (%)
-2.519	0	N/A
-0.318 ± j4.681	**0.75**	**6.78**
-0.098 ± j5.285	**0.84**	**1.85**
-0.290 ± j7.094	**1.13**	**4.08**
-30.551 ± j3.943	0.63	99.18
-49.585	0	N/A
-67.928 ± j12.421	1.98	98.37
-69.618 ± j40.168	6.39	86.62
-78.297 ± j45.788	7.29	86.32
-109.972 ± j12.355	1.97	99.37
-130.788	0	N/A
-53.094 ± j74.914	11.92	57.82
-0.367 ± j72.425	11.53	0.51
-156.367	0	N/A
-57.946 ± j103.742	16.51	48.76
-143.823 ± j83.419	13.28	86.50
-218.834 ± j204.580	32.56	73.05

VI. CONCLUSIONS

This paper proposes a linear ESN based method for extracting observable eigenvalues of a large nonlinear dynamic system around its operating point, as well as extracting the participation factors between the observable eigenvalues and the system's accessible states. A linear ESN is first trained by using offline training algorithm to learn the nonlinear dynamic system's local dynamics. The system's observable eigenvalues and their participation factors are then extracted from the linear ESN's weight matrices.

Because of the availability of the fast offline training algorithm for ESNs, the proposed method is able to quickly retrain the ESN once the system operating point changes, and reestimate the system's eigenvalues. Moreover, for ESNs, it is only necessary to change the ESN output weights in order to follow the changes of the plant operating point. Thus, the ESN demands

TABLE V
CASE 1: PARTICIPATION FACTORS (ROUNDED TO THE 3RD DECIMAL) BETWEEN THE LINEAR ESN OUTPUTS AND THE LINEAR ESN MODES

State	0 Hz (-2.52)	**0.75 Hz**	**0.84 Hz**	**1.13 Hz**	0.63 Hz	0 Hz (-49.6)	1.98 Hz	6.39 Hz	7.29 Hz	1.97 Hz	0 Hz (-131)	11.92 Hz	11.53 Hz	0 Hz (-156)	16.51 Hz	13.28 Hz	32.56 Hz
$\Delta\omega_{G2}$	0.000	0.059	**0.372**	0.003	0.000	0.001	0.003	0.003	0.002	0.001	0.001	0.001	0.052	0.002	0.000	0.001	0.000
$\Delta\omega_{G3}$	0.004	0.052	0.024	**0.378**	0.000	0.004	0.005	0.004	0.004	0.003	0.001	0.001	0.020	0.004	0.000	0.002	0.000
$\Delta\omega_{G4}$	0.002	**0.416**	0.045	0.007	0.004	0.005	0.002	0.001	0.001	0.000	0.000	0.000	0.020	0.000	0.000	0.000	0.000

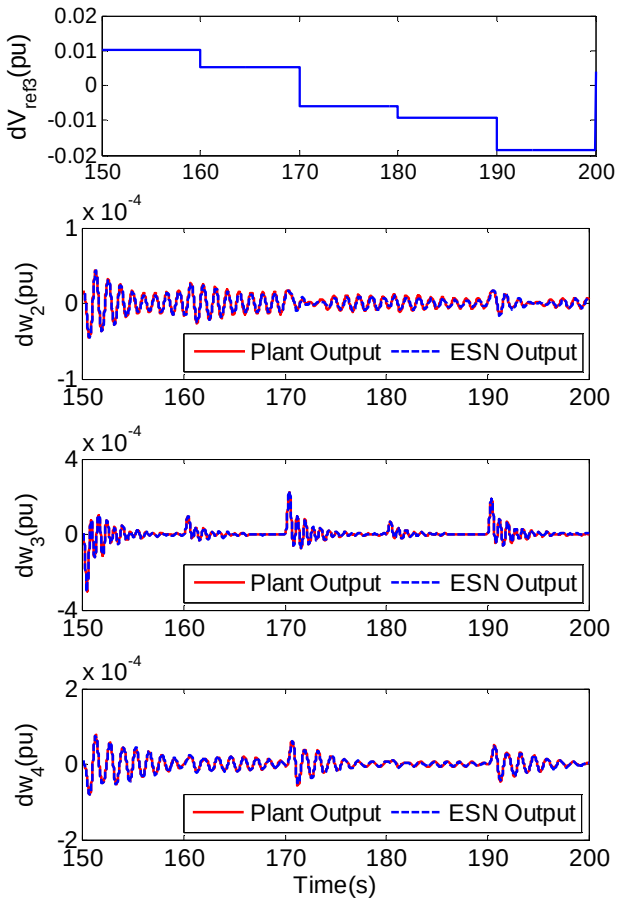

Fig. 8. Testing results of the retrained linear ESN (fixed weights) at the new operating point in case 2. Top graph: injected PRBS. Bottom three graphs: plant and the linear ESN responses under the injected PRBS.

less storage if the weights of the linear ESN are to be saved for different plant operating points.

An example in estimating the eigenvalues of a 12 bus power system under different operating points is presented in this paper and demonstrates the effectiveness of the proposed method.

TABLE VI
CASES 2: GENERATOR SET POINTS

Gen #	MVA Rating	P_{mref} (MW)	V_{ref} (pu)
Gen 2	700	600	1.02
Gen 3	500	420	1.01
Gen 4	500	420	1.02

TABLE VII
CASE 2: OSCILLATION MODES FROM ANALYTICAL MODEL

Eigenvalues	Frequency (Hz)	Damping Ratio (%)	Dominant Gen
-0.410 ± j8.050	1.28	5.09	G3
-0.037 ± j5.393	0.86	0.69	G2
-0.205 ± j4.813	0.77	4.26	G4

TABLE VIII
CASE 2: OSCILLATION MODES ESTIMATED FROM LINEAR ESN

Eigenvalues	Frequency (Hz)	Damping Ratio (%)	Dominant Gen
-0.458 ± j8.099	1.29	5.64	G3
-0.035 ± j5.316	0.85	0.66	G2
-0.211 ± j4.744	0.76	4.44	G4

REFERENCES

[1] S. Haykin, "Neural networks: a comprehensive foundation," 2nd ed., Prentice Hall, 1998. ISBN-13: 978-0132733502.

[2] D. Prokhorov, G. Puskorius, and L. Feldkamp, "Dynamical neural networks for control," in J. Kolen and S. Kremer (Eds.), "A Field Guide to Dynamical Recurrent Networks," IEEE Press, 2001. ISBN-13: 978-0780353695.

[3] H. Jaeger, "The 'echo state' approach to analysing and training recurrent neural networks," *GMD Report 148*, German National Research Center for Information Technology, 2001.

[4] J.Mazumdar, R. G. Harley, "Utilization of echo state networks for differentiating source and nonlinear load harmonics in the utility network," *IEEE Trans. Power Electronics*, vol 23, no. 6, pp:2738 – 2745, Nov. 2008.

[5] J. Mazumdar, G. K. Venayagamoorthy, R. G. Harley and F. C. Lambert, "Echo state networks for determining harmonic contributions from nonlinear loads," *Proc. IEEE Int. Joint Conf. on Neural Networks (IJCNN'06)*, Vancouver, Canada, 2006, pp. 1695 – 1701.

[6] M. Salmen and P.G. Ploger, "Echo state networks used for motor control," *Proc. IEEE International Conf. on Robotics and Automation*, Barcelona, Spain, 2005, pp. 1953 – 1958.

[7] D. Xu, J. Lan and J. C. Principe, "Direct adaptive control: an echo state network and genetic algorithm approach," *Proc. IEEE Int. Joint Conf. on Neural Networks (IJCNN'05)*, Montréal, Québec, Canada, Jul./Aug. 2005, vol. 3, pp. 1483 – 1486.

[8] J. F. Hauer, C. J. Demeure, and L. L. Scharf, "Initial results in prony analysis of power system response signals," *IEEE Trans. Power Systems*, vol. 5, no. 1, pp. 80-89, Feb 1990.

[9] H. Jaeger, "Tutorial on training recurrent neural networks, covering BPTT, RTRL, EKF and the 'Echo State Network' approach," *GMD Report 159*. German National Research Center for Information Technology, Sankt.

[10] H. Jaeger, "Adaptive nonlinear system identification with echo state networks," *Proc. Advances in Neural Information Processing Systems*, pp. 593-600, MIT Press, Cambridge, MA, 2003.

[11] G. Venayagamoorthy, "Online design of an echo state network based wide area monitor for a multimachine power system," *Neural Networks*, vol. 20, pp. 404-413, 2007.

[12] J. Dai, G. Venayagamoorthy, and R. Harley, "An introduction to the echo state network and its applications in power system," *Proc. Intelligent System Applications to Power Systems (ISAP 2009)*, Curitiba, Brazil, Nov. 8-12, 2009.

[13] R. A. DeCarlo, "Linear systems: a state variable approach with numerical implementation," Prentice Hall, NJ, 1989. ISBN-13: 978-0135368145.

[14] P. Kundur, "Power system stability and control", New York: McGraw-Hill, 1994. ISBN-13: 978-0070359581.

[15] S. Jiang, U. D. Annakkage, and A. M. Gole, "A platform for validation of FACTS models," *IEEE Trans. Power Delivery*, vol. 21, no. 1, pp. 484-491, Jan 2006.