A Kalman Filter Approach to Quasi-Static State Estimation in Electric Power Systems

Ellery A. Blood, *Student Member, IEEE*, Marija D. Ilić, *Fellow, IEEE*, Jovan Ilić, and Bruce H. Krogh, *Fellow, IEEE*

***Abstract*—Static state estimation in electric power systems is normally accomplished without the use of time-history data. This paper presents preliminary work on the use of the discrete-time Kalman filter to incorporate time history into the state estimators. Transitional state equations are derived via linearization of the power flow equations. A simplified example using a decoupled real power flow solution demonstrates this technique.**

***Index Terms*—Kalman Filter, Electric Power System, Static State Estimation, Time History**

I. Introduction

ESTIMATION of the static state of an electric power system (the vector of complex voltage magnitudes and angles at each bus) is a nontrivial task that is performed continuously on electric power systems with hundreds (often thousands) of buses. The static state is typically determined through minimizing a cost function using a recursive weighted, least squares approach. The cost function is typically designed to be a weighted sum of the squares of the error between the actual measurements and calculations based on the estimated state [1]. Due to the complexity and inherent nonlinearity of the power system, convergence of the solution to the static state estimation is not guaranteed. In fact, various methods of static state estimation may have wildly different regions of convergence. A particularly challenging aspect in today's state estimation comes from topological changes and bad data. In this paper, we explore preliminary results of a method that may improve the convergence of the static state estimate through the use of the Kalman filter formulation.

Several contributions regarding Kalman filtering approach to the state estimation problem have been proposed [5, 10, 11]. Our proposed formulation emphasizes a transitional state equation based on physical laws. The Kalman filter [2] is a four-step algorithm that consist of the following: 1) Predicting the future state from inputs, 2) calculating the Kalman gain, 3) correcting the prediction with new measurements, 4) updating the covariance to reflect the integrating the measurements.

This work was supported in part by the U.S. National Science Foundation under awards CCR-0325892, SES-034578, and CNS-0428404, and, in part, by the U.S. Department of Energy, National Energy Technology Laboratory, under Research and Development Solutions, LLC contract number DE-AM26-04NT41817.305.01.21.002.

E.A. Blood is a Ph.D. candidate at Carnegie Mellon University, Pittsburgh, PA 15213 USA. (e-mail: eblood@andrew.cmu.edu).

M.D. Ilic is a professor of Electrical and Computer Engineering at Carnegie Mellon University, Pittsburgh, PA (e-mail: milic@ece.cmu.edu)

J. Ilic is a research scientist in the Electrical and Computer Engineering department at Carnegie Mellon University, Pittsburgh, PA

B.H. Krogh is a professor of Electrical and Computer Engineering at Carnegie Mellon University, Pittsburgh, PA

II. Time Invariant Static State Estimation

Static state estimation in today's electric utility control centers has historically been the interpretation of a single sample (either a snapshot or a scan) of measurements to determine the state of the system, **x**. As such, the formulation ignores any information from previous samples (with the exception that the initial guess is typically the previous state estimate). A typical representation of **x** is shown in (1) where δ is a voltage angle and V is a voltage magnitude. Similarly, the measurements, **z**, are represented as in (2).

$$\mathbf{x} = \begin{bmatrix} \delta_1 & \delta_2 & \cdots & \delta_n & V_1 & V_2 & \cdots & V_n \end{bmatrix}^T \quad (1)$$

$$\mathbf{z} = \begin{bmatrix} z_1 & z_2 & \cdots & z_m \end{bmatrix}^T \quad (2)$$

To ensure sufficient redundancy, it is desirable that the number of measurements (m) be at least twice the number of states (2n) [3].

The ideal measurements are assumed to be deterministic and calculable given a known state. Actual measurements are often modeled as ideal measurements that have been corrupted by zero-mean Gaussian noise. Thus:

$$\mathbf{z} = \mathbf{h}(\mathbf{x}) + \upsilon \quad (3)$$

In order to determine the value of **x** that most accurately matches the measurements, a weighted least squares approach is used. A cost function (C) is defined as follows:

$$C(\mathbf{x}) = \frac{1}{2}\left[\mathbf{z} - \mathbf{h}(\mathbf{x})\right]\mathbf{R}^{-1}\left[\mathbf{z} - \mathbf{h}(\mathbf{x})\right]^T \quad (4)$$

where **R** is the measurement covariance matrix, which allows measurements with less noise to be weighted more heavily than less accurate measurements.

The cost function, (4), is minimized where the derivative of (4) is equal to zero as shown in (5).

$$\left.\frac{\partial C(\mathbf{x})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}} = \mathbf{H}(\hat{\mathbf{x}})\mathbf{R}^{-1}\left[\mathbf{z} - \mathbf{h}(\hat{\mathbf{x}})\right] = \mathbf{0} \quad (5)$$

where $$\mathbf{H} = \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}}. \quad (6)$$

Because **h**(**x**) is a nonlinear function, an iterative procedure to solve for the state estimate that minimizes C(**x**) may be

1-4244-0228-X/06/$20.00 ©2006 IEEE

necessary. Schweppe [1] discusses a modification to (5) that results in a suitable formulation to iteratively minimize C(**x**) (7).

$$\mathbf{H}^T(\hat{\mathbf{x}})\mathbf{R}^{-1}\mathbf{H}(\hat{\mathbf{x}})\left[\hat{\mathbf{x}}_{i+1}-\hat{\mathbf{x}}_i\right]=\mathbf{H}^T(\hat{\mathbf{x}})\mathbf{R}^{-1}\left[\mathbf{z}-\mathbf{h}(\hat{\mathbf{x}})\right] \quad (7)$$

Equation (7) can be solved in various ways, (Schweppe discusses square root factorization among others), however for ease of notation, we multiply both sides of (7) by the inverse of the left hand coefficient. This lets us solve for $\hat{\mathbf{x}}_{i+1}$ as in (8).

$$\hat{\mathbf{x}}_{i+1}=\hat{\mathbf{x}}_i+pinv(\mathbf{H}(\hat{\mathbf{x}}),\mathbf{R})\left[\mathbf{z}-\mathbf{h}(\hat{\mathbf{x}})\right] \quad (8)$$

where

$$pinv(\mathbf{H}(\hat{\mathbf{x}}),\mathbf{R})=\left\{\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}(\hat{\mathbf{x}})\right\}^{-1}\mathbf{H}^T(\hat{\mathbf{x}})\mathbf{R}^{-1} \quad (9)$$

is the least-squares pseudo-inverse of **H** weighted by **R**. The iteration repeats until the change in **x** is smaller than the desired tolerance.

III. Dynamics of Quasi-Static State Transitions

The time constants inherent in an electrical power system are typically more than an order of magnitude smaller (faster response) than the rate at which measurements are gathered. Thus, a dynamic model based on the measurements is unable to track the actual dynamics but, instead, track the way the system transitions from one quasi-static state to the next.

Various efforts have been made to monopolize on these quasi-static transitions. Masiello and Schweppe [4] investigate using a dynamic model where the state transition matrix was the identity matrix (zero-order-hold). This method effectively works off the presumption that the state does not dramatically change from one sample to the next. Debs and Larson [5] expand on this by using a diagonal state transition matrix to simulate an uncoupled first order response in the state. The driving input in both methods is assumed to be random noise (i.e. not determined by physical laws).

Both methods use the value for the predicted state as a set of pseudo-measurements. Both methods produce favorable results if the actual state transitions were small so that their assumptions generally held true. The performance tends to suffer, however, if large changes in state were experienced.

In this paper, we speculate that the state transition is deterministic given certain key measurements. Thus, we study the performance of the pseudo-dynamic formulation through more representative state transition model, which uses system measurements instead of random noise to drive the transition. We then apply a Kalman filter to this pseudo-dynamic system.

IV. Pseudo-Dynamic Model

In order to utilize the Kalman filter formulation, we must first generate a model for the state transition of the system. In this paper, we generate this pseudo-dynamic model through the linearization of the power flow equation (10) at bus i.

$$\widetilde{V}_i\sum_k\left(\widetilde{V}_k Y_{ik}\right)^*=P_i+jQ_i \quad (10)$$

The power flow equation relates the real and reactive power injections at the i[th] bus to the complex voltages in the entire system. P_i is the real power injection at bus i, Q_i is the reactive power injection at bus i, Y_{ik} is the (i,k)[th] element of the admittance matrix, and $\widetilde{V}_i=V_i e^{j\delta_i}$ is the complex phasor voltage at bus i.

If we define a vector **u** to represent the real and reactive power injections as in (11), and the state **x** as in (1), we can rewrite (10) in terms of **u** and **x** as (12).

$$\mathbf{u}=\begin{bmatrix}P_1 & P_2 & \cdots & P_n & Q_1 & Q_2 & \cdots & Q_n\end{bmatrix}^T, \quad (11)$$

$$g_i(\mathbf{x},\mathbf{u})=V_i e^{j\delta_i}\sum_k\left(V_k e^{j\delta_k}Y_{ik}\right)^*-\left(P_i+jQ_i\right)=0 \quad (12)$$

We can now define a vector of expressions [1] $\mathbf{f}(\mathbf{x},\mathbf{u})$ to be

$$\mathbf{f}(\mathbf{x},\mathbf{u})=\left[\mathbf{f}_P(\mathbf{x},\mathbf{u})\quad \mathbf{f}_Q(\mathbf{x},\mathbf{u})\right]^T=$$
$$\begin{bmatrix}real(g_1(\mathbf{x},\mathbf{u})) & \cdots & real(g_n(\mathbf{x},\mathbf{u}))\\ imag(g_1(\mathbf{x},\mathbf{u})) & \cdots & imag(g_n(\mathbf{x},\mathbf{u}))\end{bmatrix}^T \quad (13)$$

We then linearize this equation and look at the incremental change. This yields the following equation:

$$\frac{\partial\mathbf{f}(\mathbf{x},\mathbf{u})}{\partial\mathbf{x}}\Delta\mathbf{x}+\frac{\partial\mathbf{f}(\mathbf{x},\mathbf{u})}{\partial\mathbf{u}}\Delta\mathbf{u}+\mathbf{e}_f=0 \quad (14)$$

where $\mathbf{e_f}$ represents the modeling error due to the nonlinearity of (10).

Investigating the partial derivatives more closely,

$$\frac{\partial\mathbf{f}(\mathbf{x},\mathbf{u})}{\partial\mathbf{x}}=\begin{bmatrix}\frac{\partial\mathbf{f}_P}{\partial\delta} & \frac{\partial\mathbf{f}_P}{\partial V}\\ \frac{\partial\mathbf{f}_Q}{\partial\delta} & \frac{\partial\mathbf{f}_Q}{\partial V}\end{bmatrix}=\mathbf{J} \quad (15)$$

is the power flow Jacobian, and

$$\frac{\partial\mathbf{f}(\mathbf{x},\mathbf{u})}{\partial\mathbf{u}}=\begin{bmatrix}\frac{\partial\mathbf{f}_P}{\partial\mathbf{P}} & \frac{\partial\mathbf{f}_P}{\partial\mathbf{Q}}\\ \frac{\partial\mathbf{f}_Q}{\partial\mathbf{P}} & \frac{\partial\mathbf{f}_Q}{\partial\mathbf{Q}}\end{bmatrix}=\begin{bmatrix}\mathbf{I} & 0\\ 0 & \mathbf{I}\end{bmatrix}=\mathbf{I} \quad (16)$$

is the 2n by 2n identity matrix.

We can now solve for **Δx.** Recognizing that **Δx** represents the increment in our state, we have now arrived at our model for the state transition of the system:

$$\mathbf{x}(t+1)=\mathbf{x}(t)+\mathbf{J}^{-1}\left[\mathbf{u}(t+1)-\mathbf{u}(t)\right]+\mathbf{w}(t) \quad (17)$$

where **w**(t) represents the error due to linearization of a nonlinear system.

[1] For a typical constant power flow model, it is sufficient to use only (n-1) components of real(g_i(**x**,**u**)), and *l* components of imag(g_i(**x**,**u**)) of **f**(**x**,**u**) where *i* is the number of loads in the system. This results in a sufficient number of structurally consistent equations that define the state **x** for a given input **u**.

V. Kalman Filtering

Now that we have defined how we expect the pseudo-dynamics of our system to behave, we can explore the Kalman filtering methodology as it applies to an electrical power system.

The first step in the Kalman filter is prediction. We calculate the prediction $\bar{\mathbf{x}}(t+1)$ from the previous state estimate $\hat{\mathbf{x}}(t)$, our pseudo-dynamic model (16), and the input. Thus equation (17) becomes:

$$\bar{\mathbf{x}}(t+1) = \hat{\mathbf{x}}(t) + \mathbf{J}^{-1}\left[\mathbf{u}(t+1) - \mathbf{u}(t)\right] \tag{18}$$

The second step in the Kalman filter is calculation of the Kalman gain. We calculate the Kalman gain from the state covariance matrix (**S**(t)), the measurement covariance (**R**), the input covariance (**W**), the pseudo-dynamic model, and the measurement model. The Kalman gain **L**(t) is:

$$\mathbf{L}(t+1) = \left[\mathbf{S}(t) + \mathbf{J}^{-1}\mathbf{R}\left(\mathbf{J}^{-1}\right)^{T}\right]\mathbf{H}^{T} \left\{\mathbf{H}\left[\mathbf{S}(t) + \mathbf{J}^{-1}\mathbf{R}\left[\mathbf{J}^{-1}\right]^{T}\right]\mathbf{H}^{T} + \mathbf{W}\right\}^{-1} \tag{19}$$

The third step in the Kalman filter is to update the state estimate with the measurements. We calculate the error between the actual measurements and their values calculated from the predicted state becomes the innovation vector. Combining this with the Kalman gain, we have:

$$\hat{\mathbf{x}}(t+1) = \bar{\mathbf{x}}(t+1) + \mathbf{L}(t+1)\left\{\mathbf{z}(t+1) - \mathbf{h}(\bar{\mathbf{x}}(t+1))\right\} \tag{20}$$

The fourth step in the Kalman filter is to update the covariance matrix **S**. This gives us an updated value for how confident we are in the value of the state.

$$\mathbf{S}(t+1) = \left[\mathbf{I} - \mathbf{L}(t+1)\mathbf{H}\right]\left[\mathbf{S}(t) + \mathbf{J}^{-1}\mathbf{R}\left[\mathbf{J}^{-1}\right]^{T}\right]\left[\mathbf{I} - \mathbf{L}(t+1)\mathbf{H}\right]^{T} + \mathbf{L}(t+1)\mathbf{W}\mathbf{L}^{T}(t+1) \tag{21}$$

VI. Simulation Model and Methodology

In order to test the performance of the model discussed above, we created the 4-bus system shown in Fig. 1. All lines are lossless and bus 1 is the slack (angle reference) bus.

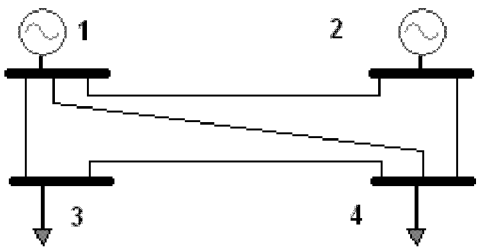

Fig. 1: The 4-bus test system.

In the simulations, the voltage magnitudes at all the buses are held constant at 1 per/unit. This reduces the number of state variables from 8 (4 angles and 4 magnitudes) to 4. Assigning bus 1 as the slack bus, further reduces the number of free variables by constraining the angle at bus 1 to be identically equal to zero. Assignment of a slack bus is necessary to prevent the power flow Jacobian (**J**) from being structurally singular.

By decoupling the solutions for real and reactive power [6-7] and recognizing that all voltage magnitudes are held constant, the problem reduces to the solution of the real power flow as a function of the angles on buses 2 through 4.

The simulation is therefore run for a given real-power-only scenario with loads (negative real power injections into buses 3 and 4) as shown in Fig. 2.

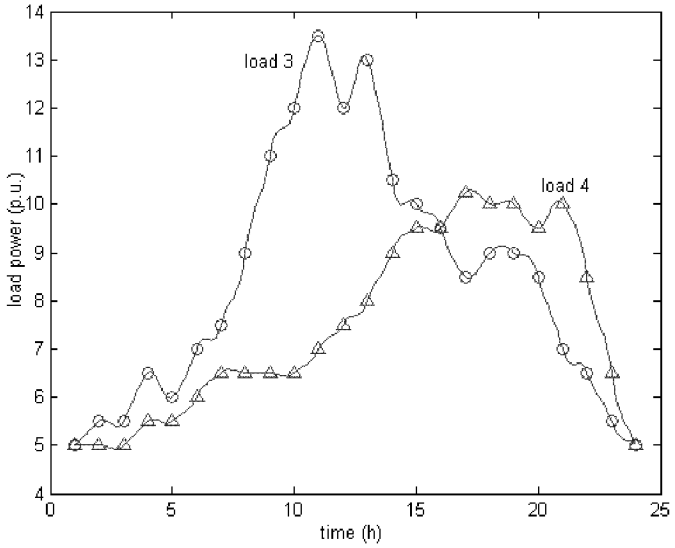

Fig. 2: Graph of the real loads applied to buses 3 (circles) and 4 (triangles) through a typical 24 hour period. The curve represents cubic spline interpolations between the specified points for samples taken off the hour.

For each step in the simulation, the loads for buses 3 and 4 are assigned the values from Fig 2. The generation on bus 2 is assigned to be 1/3 of the sum of the loads. A Newton-Raphson load flow solution is performed on this system to determine a value for the state. This will be considered the true state for our purposes of this simulation. All generated measurements are calculated from the true state and the network model.

This paper evaluates three algorithms of increasing complexity. Each algorithm accomplishes the state estimation process in two steps.

Step 1) Prediction: A starting point for the state estimation process is generated through a prediction based on the previous state estimate, certain key measurements, and a weighting matrix **M** (21).

$$\bar{\mathbf{x}}(t+1) = \hat{\mathbf{x}}(t) + \mathbf{M}\left[\mathbf{u}(t+1) - \mathbf{u}(t)\right] \tag{21}$$

Step 2) Correction: The predicted state is corrected through a single step, linearized least-squares iteration based on a weighting matrix **N** and a measurement error (22). **N** may be

dependent on **x**, and thus will be evaluated at $\mathbf{x} = \bar{\mathbf{x}}(t+1)$.

$$\hat{\mathbf{x}}(t+1) = \bar{\mathbf{x}}(t+1) + \mathbf{N}\left[\mathbf{z} - \mathbf{h}(\bar{\mathbf{x}}(t+1))\right] \quad (22)$$

Algorithm 1 uses the zero matrix for **M** and (9) for **N**. Thus, it simply predicts that the new state is the previous state and performs a single iteration weighted, least squares correction.

Algorithm 2 uses $\mathbf{J}^{-1}$ for **M** and (9) for **N**. Thus, it predicts the state using the load flow equations and change in the real power injections and performs a single iteration weighted, least squares correction.

Algorithm 3 uses $\mathbf{J}^{-1}$ for **M** and (19) for **N**. Thus, it predicts the state using the load flow equations and change in the real power injections and uses the Kalman gain for correction.

VII. Simulation Results

A typical data run of the simulation appears below. Fig. 3 through Fig. 5 show the angle for buses 2 through 4 respectively. The angle at bus 1 (the slack bus) is identically equal to zero, and is not shown. The solid line represents the true state (voltage angle) for that bus. The circle shows the predicted state from step 1), and the star shows the state estimate after correction in step 2). The data run begins with a flat start (all bus angles initialized to zero radians).

The initial value for the state covariance matrix is 50 for the diagonal elements and 0 for the off diagonal elements. The input covariance is assigned to be 5 per unit (p.u.) for all injections and the measurement covariance is assigned to be 0.1 p.u. for all branch flows. The noise injected to the branch measurements is a zero mean random variable with a variance of 0.1 p.u.

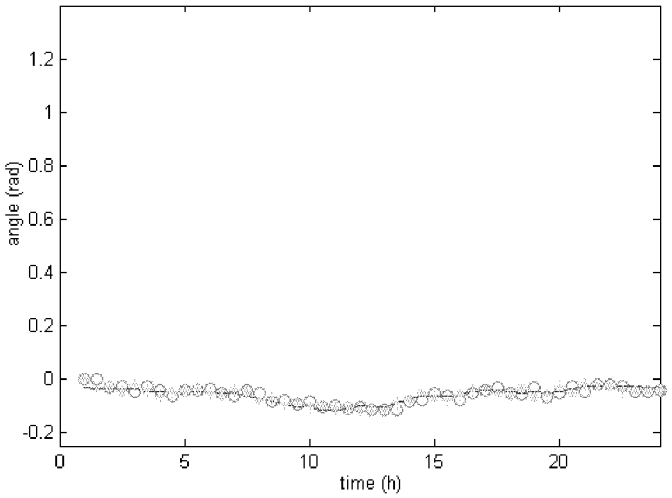

Fig. 3: Voltage angle on bus 2 for a typical data run using algorithm 3 with 30 minute samples.

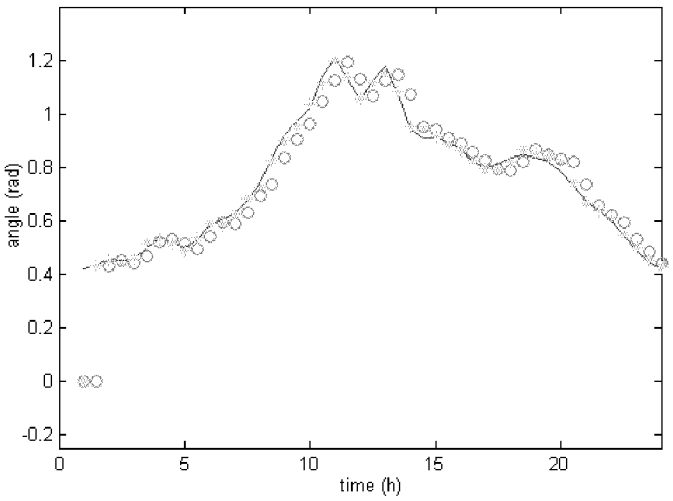

Fig. 4: Voltage angle on bus 3 for a typical data run using algorithm 3 with 30 minute samples.

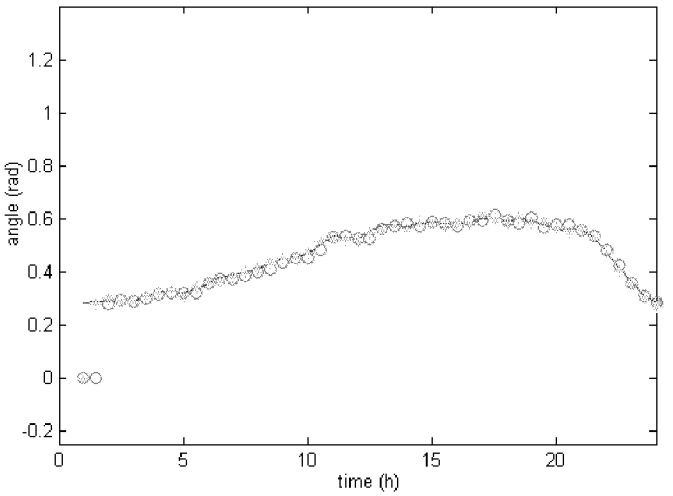

Fig. 5: Voltage angle on bus 4 for a typical data run using algorithm 3 with 30 minute samples.

For this evaluation, 1000 data runs were performed to provide the necessary collection of data to generate statistically significant values for the ensemble mean and standard deviation of the state estimate error. Data runs were performed with 60 minute, 30 minute, 20 minute, and 15 minute sample periods

Fig. 6 shows the mean state estimate error for the 30 minute sample time. The solid line represents algorithm 1, the dashed line represents algorithm 2, and the dotted line represents algorithm 3. In each of the three plots the mean error generally decreases as we progress from algorithm 1 to 3. This is most pronounced on bus 2.

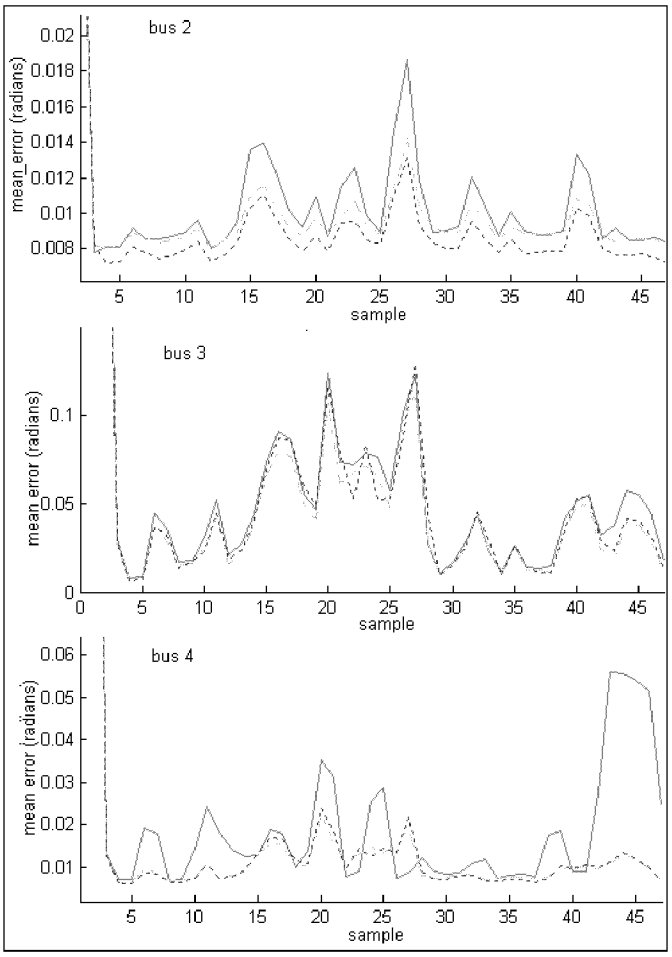

Fig. 6: Mean error for 30 minute samples for buses 2-4.

Similarly, if we look at the standard deviation of the error for the same 30 minute sample time (Fig. 7), we can see that the solid line (algorithm 1) has the largest standard deviation, the dashed line (algorithm 2) is smaller, and the dotted line (algorithm 3) generally has the smallest standard deviation.

The sampling periodicity strongly correlates to the average magnitude of the state transition between samples. Thus, if we compare the performance of the different algorithms given varied sampling frequencies, we can get an idea of how the algorithms perform under varied sizes of state transitions.

Fig. 8 shows the mean error for bus 2 for various sampling rates. In each of the four plots, we can observe that the solid line (algorithm 1) consistently has the highest mean error. The dashed and the dotted lines lie close to each other with 60minute sampling, but as the sampling time decreases, the lines diverge and the dotted line (algorithm 3) shows a significantly smaller mean error than the dashed line (algorithm 2).

Since algorithms 2 and 3 show significant improvements over algorithm 1 for large state changes, a logical conclusion is that the improvement is due to the state prediction using (18). Similarly, since algorithm 3 shows significant improvements over algorithms 1 and 2 for small state changes, a logical conclusion is that the improvement is due to the tracking of the state error covariance and calculation of the Kalman gain in (19) and (20).

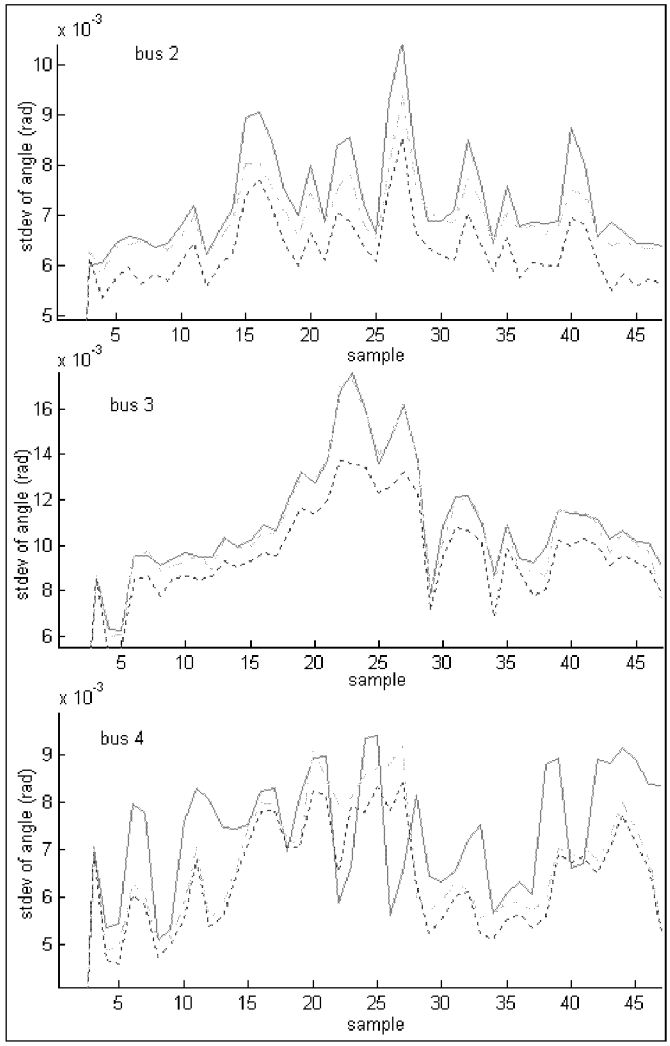

Fig. 7: Standard deviation of the error for 30 minute samples for buses 2-4.

VIII. Discussion

The simulation results show that a combination of state prediction based on linearization of the power flow equations and Kalman filter formulation results in improved estimates of the static state given only a single iteration. It is important to note that, because the three algorithms described in section VI only perform a single iteration of each step per sample, errors from the linearization of the system are present. Commercial state estimation algorithms, therefore, repeat the iteration until the solution converges to within the desired tolerance.

Multiple iterations are not used here so that a state estimate can be continuously updated as new measurements arrive. Larson, Tinney, and Peschon [8] discuss multiple decompositions to support sequential evaluation of measurements. Additionally, Debs and Larson discuss [5] how sequentially processed measurement can be applied while using a Kalman filter in their dynamic formulation. Applying these ideas, we can consider groups of measurements arriving together. Each group could be treated as either inputs or correcting measurements and incorporated

into the state estimate one group at a time. This breaks up the state estimation problem into many small problems that can be handled sequentially.

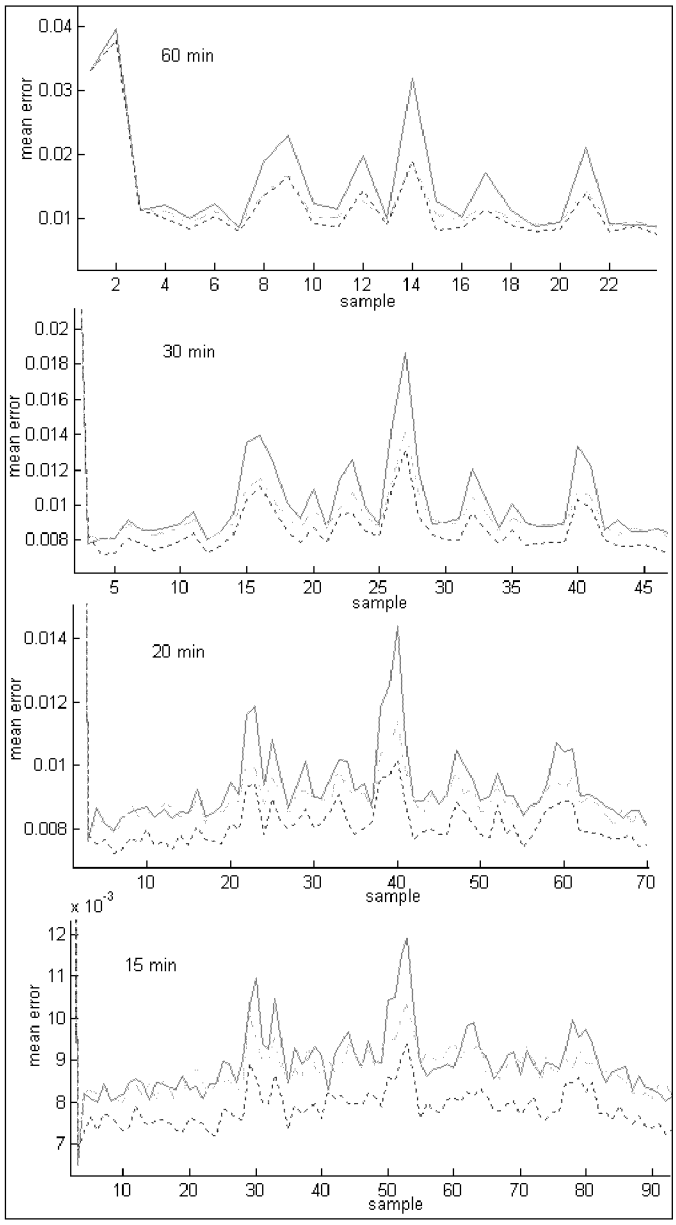

Fig. 8: Mean error for bus 2 sampled at 60, 30, 20, and 15 minute intervals.

Additionally, we recognized that the power injections at the buses may not be the best choice of inputs to predict the change in state. Branch measurements (e.g., power flow, current, voltage differential) generally tend to give a more meaningful result in a noisy environment [9]. We are currently investigating other methods for choosing the measurements that make up the input vector.

IX. Acknowledgments

The authors greatly appreciate funding from the National Science Foundation and the National Energy Technology Laboratory. The authors would like to thank John Allemong and his colleagues at American Electric Power (AEP) Columbus, OH for helpful discussions of current techniques used in commercial static state estimation.

References

[1] F.C. Schweppe, E.J. Handschin, *Static state estimation in electric power systems*, IEEE Proceedings, vol 62, issue 7, pp. 972-982, July 1974

[2] R. E. Kalman, *A new approach to linear filtering and prediction problems*, Trans. ASME, J. Basic Engrg., ser. D, vol. 82, pp. 35-45, 1960

[3] J.J. Allemong, *State Estimation Fundamentals for Successful Deployment,* 2005 IEEE Power Engineering Society General Meeting, vol 1, pp. 800-801, June 2005.

[4] R. D. Masiello, F. C. Schweppe, *A tracking state estimator*, IEEE Transactions on Power Apparatus and Systems, vol PAS-90, iss 3, pp. 1025-1033, May-June 1971

[5] A.S. Debs, R. E. Larson, *A dynamic estimator for tracking the state of a power system*, IEEE Transactions on Power Apparatus and Systems, vol PAS-89, iss. 7, pp. 1670-1678, Sept-Oct 1970

[6] B. Stott, 0. Alsac, *Fast decoupled load flow,* IEEE Trans. Power App. Syst., vol. PAS-93, pp. 859-869, May 1974.

[7] C.H. Joissaint, N.V. Arvanitidis, D.G. Luenberger, *Decomposition of real and reactive power flows: A method suited for online applications*, IEEE Transactions on Power App. Syst., vol. PAS-91, pp. 661-670, Nov/Apr 1972

[8] R.E. Larson, W.F. Tinney, J. Peschon, *State Estimation in Power Systems: Part I: Theory and Feasability*, Vol PAS-89, Iss 3, pp. 345-352, March 1970

[9] J.J. Allemong, L.S. VanSlyck, *Operating Experience with the AEP State Estimator*, IEEE Transactions on Power Systems, Vol 3, Iss 2, pp. 521-528, May 1988

[10] A.K. Sinha, J.K. Modal, *Dynamic State Estimator Using ANN Based Bus Load Prediction,* IEEE Transactions on Power Systems, Vol 14, iss 4, pp. 1219-1225, Nov 1999

[11] K. Shih, S. Huang, *Application of a Robust Algorithm for Dynamic State Estimation of a Power System,* IEEE Transactions on Power Systems, Vol 17, Iss. 1, pp. 141-147, February 2002