

# Data-Driven Power System Operations<sup>\*</sup>

E.H. Abed<sup>1</sup>, N.S. Namachchivaya<sup>2</sup>, T.J. Overbye<sup>2</sup>, M.A. Pai<sup>2</sup>,  
P.W. Sauer<sup>2</sup>, and A. Sussman<sup>1</sup>

<sup>1</sup> University of Maryland, College Park MD 20742, USA

<sup>2</sup> University of Illinois, Urbana, IL, 61801, USA

**Abstract.** In operations, simulation and control of power systems, the presence of real-time data relating to system states can yield precise forecasts and can enable robust active control. In this research we are developing efficient and robust methods to produce “data enhanced” reduced order models and filters for large-scale power systems. The application that this paper focuses on is the creation of new data-driven tools for electric power system operation and control. The applications systems include traditional **SCADA** systems as well as emerging **PMU** data concentrators. A central challenge is to provide near real-time condition assessment for “extreme events,” as well as long-term assessment of the deterioration of the electrical power grid. In order to provide effective guidance for power system control, we are also developing visualization methods for integrating multiple data sets. These visualization methods provide an up-to-date view of the system state, and guide operator-initiated power system control.

## 1 Introduction

The current power system operation paradigm relies heavily on two components: human operator experience and model-based analysis. Sensors currently exist at hundreds of thousands of locations dispersed at generating stations, at major network substations, and at loads. This sensor data is currently transmitted by Intelligent Electronic Devices (**IEDs**) to local and area control centers through Remote Terminal Units (**RTUs**) by a two-way network called a Supervisory Control and Data Acquisition (**SCADA**) system. In the past few years the restructuring of the electric power industry has created Regional Transmission Organizations (**RTOs**) and Independent System Operators (**ISOs**) that may also have access to certain amounts of this data. The area control centers, **RTOs**, and **ISOs** currently utilize this data in raw form for status displays and alarming. This data is also used as the input to a system State Estimator (**SE**) that is designed to filter erroneous data, create topology estimates, and provide the *best estimate* of the current state of the network configuration and the values of key indicators (voltages, currents, power and frequency). Human operators make judgments and decisions based on this data and on traditional analysis tools that utilize this data. At the very local level, automatic protection systems

---

\* This research is supported by NSF Grant #CNS-0540216.

react to measured data as they are programmed, with their primary objectives being to save equipment from damage in the event of abnormal conditions (lightning strikes, short circuits, etc.) and to continue to serve customers even under component failures.

The main objective of this research is to develop new algorithms and tools for the distributed collection, sharing, and harnessing of data for cooperative health monitoring and security assessment of power systems *in real time*. Fundamental concepts and technology for achieving this objective exist today in the areas of power systems, system theory and computer science, but harnessing them to achieve this vision requires the significant multidisciplinary concerted effort presented here. This work aims at making fundamental contributions to the necessarily multidisciplinary systems required for data utilization in monitoring and control of interconnected engineering systems. The application we focus on is the creation of new data-driven tools for electric power system operation and control. The systems include traditional **SCADA** systems as well as emerging Phasor Measurement Unit (**PMU**) data concentrators. Traditional **SCADA** equipment provides massive amounts of data that can be processed for diagnoses, awareness indicators, and other new operational tools. This effort will focus on data-driven tools for power system operations, which includes filtering, low-order modeling, and automated visualization for power system health monitoring and security assessment.

## 2 Data-Driven Tools for Power System Operations

Power system state estimation, security analysis, and real-time operation are currently heavily dependent on model parameters and assumed topology information. With the restructuring of power grid operations, the large geographical nature of the interconnected system is making the computation of state estimation more challenging. In addition, the massive volume of data and real-time alarms is making situational awareness and decision-making issues more severe. At the same time, new technologies are emerging to provide new types of data that can be utilized for both monitoring and control.

### 2.1 Signal Process: Power System Model

As the Data Grid maps the Power Grid in real time, it would be useful to have a view into the future evolution of the various variables. The traditional, time tested way to simulate the dynamic behavior of a power system is from a mathematical parameter-driven model of the system. The difference between the actual measured data and the simulated data can be used to improve the accuracy of the model. A typical dynamic model is an event-driven system so that we have a large-scale hybrid differential algebraic equations(**DAE**) system of the form:

$$\dot{X}_t = f(X_t, Z_t, \lambda), \quad X_0 = x \in \mathbb{R}^n, \quad Z_0 = z \in \mathbb{R}^m \quad (1)$$

$$0 = \begin{cases} g^1(X_t, Z_t, \lambda), & s(X_t, Z_t, \lambda) < 0, \\ g^2(X_t, Z_t, \lambda), & s(X_t, Z_t, \lambda) > 0. \end{cases}$$

In equations (1) a switching occurs when the switching function  $s(x, z, u) = 0$ . For example, if a device is switched into service at time  $t = t_{sw}$ , then the switching function in this case can simply be defined as  $s(x, z, u) = t - t_{sw}$ . In the above model,  $x$  are the dynamic state variables such as machine angles, velocities, etc.;  $z$  are the algebraic variables such as load bus voltage magnitudes and angles; and  $\lambda$  are the system parameters such as line reactances, generator mechanical input power, and fault clearing time. Note that the state variables  $x$  are continuous while the algebraic variables can undergo step changes at switching instants. The initial conditions for equation (1) are such that  $z$  satisfies the equation

$$g(x, z, u) = 0. \quad (2)$$

The model in equation (1) is imperfect. We assume that system (1) is subject to small *random jerks*, a fast perturbation process  $\xi$  that represents the unmodeled dynamics of the system, which we assume to be a Markov process. Then equations (1) become

$$\dot{X}_t = f(X_t, Z_t, \xi_t; \lambda), \quad X_0 = x \in \mathbb{R}^n, \quad (3)$$

$$0 = g(X_t, Z_t; \lambda), \quad Z_0 = z \in \mathbb{R}^m.$$

## 2.2 Observation Process: Phasor Measurement Unit

The introduction of **PMU** technology on a wide-area basis is offering a fruitful direction for dynamic data utilization. The **PMU** technology is providing Global Positioning System (GPS) based time stamping of key power system states, including phase angles of bus voltages. The **PMU** observation process  $\{Y_t : t \geq 0\}$  is a function of  $(X_t, Z_t)$  corrupted by noise  $\eta_t$ , and is given by

$$Y_t = h(X_t, Z_t) + \eta_t \quad (4)$$

where  $h$  is called the sensor function that transforms the model variables into observed variables. Assuming data gathering and concentration at the rate of 60 measurements per second, this information can be used to detect dynamic trends as well as steady-state conditions. Initial research in this area indicates that the dynamic filtering of the **PMU** is very nearly equivalent to the mathematical filtering of the singular perturbation process in traditional power system dynamic modeling. This is important because it creates a one-to-one link between the dynamic model states and the measured data. This makes the data a valuable input for model validation, post-mortem analysis, and remedial action schemes in real-time control. Since most dynamic modes of interest are considerably slower than 60 Hz, these measurements will be useful in predicting damping of critical modes of oscillation.

### 2.3 Filtering

The theory of nonlinear filtering forms the framework in which problems of data assimilation for the nonlinear power network models will be treated. In this work, we consider the situation where the nature of the monitored environment will be captured by a Markovian state-space model that involves potentially nonlinear dynamics, nonlinear observations, and Gaussian observation noises. Our goal is to perform sequential estimation of the current system state from equation (2) at multiple sensor nodes from equation (4) in the power network system. In a nonlinear environment, particle methods, which includes sequential Monte Carlo and interacting particle filters are very useful. We will design distributed particle filters (or sequential algorithms) that can dynamically fuse the information recorded by the sensors without requiring excessive exchange of data. This work is being initiated through deterministic analysis of data measurements that determine initial conditions for model-based dynamic simulations. The results will be used in security assessment algorithms.

Using the algebraic equations (2), the variable  $Z_t$  can be eliminated in the model (1), as well as in the observer (4). Hence, in a discrete state space version of the model the state  $x_k \in \mathbb{R}^n$  and the observation  $y_k \in \mathbb{R}^p$  is described by

$$x_{k+1} = f_k(x_k, \xi_k) \quad (5)$$

$$y_k = h_k(x_k, \eta_k), \quad (6)$$

where  $f_k : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$  and  $h_k : \mathbb{R}^n \times \mathbb{R}^r \rightarrow \mathbb{R}^p$  are possibly nonlinear functions of the state  $x_k$ . The process and observation errors are represented by Gaussian white noise sequences  $w_k$  and  $v_k$  respectively and assumed to be independent to each other. Given the initial density  $p(x_0)$ , we seek to find the filtered estimation of the state i.e., the estimation of the state  $x_k$  based on the history of the past observation  $y_{1:k}$  defined by  $\{y_i : i = 1 \dots k\}$ . Since  $x_k$  can be considered as a random variable, the characterization of  $x_k$  is given by the conditional probability density,  $p(x_k | y_{1:k})$ . In the Bayesian approach we follow, this can be obtained through recursive predictions and observations. Given an initial state, a *a priori* estimation of the state is calculated according to equation (5) in the prediction stage. After a new observation is taken a *a priori* estimation is updated to become a *a posteriori* state estimation.

By Bayesian filtering, we mean any technique that uses Bayes' formula to estimate the posterior density  $p(x_k | y_{1:st_k})$  for each  $k = 1, 2, \dots$ . This usually consists of two important steps called *prediction* and *update*.

- i. (Bayesian Prediction) This step maps the previous *posterior density*  $p(x_{k-1} | y_{1:st_{k-1}})$  into the one-step *prediction density*  $p(x_k | y_{1:st_{k-1}})$ . The formula for this step follows immediately from Bayes' formula and the Markov property of the system dynamics as

$$p(x_k | y_{1:k-1}) = \int p(x_k | x_{k-1}) p(x_{k-1} | y_{1:k-1}) dx_{k-1}. \quad (7)$$

- ii. (Bayesian Measurement Update) Now we want to use  $y_k$  and the Bayesian prediction  $p(x_k|y_{1:st_{k-1}})$  from the first step to calculate the posterior density  $p(x_k|y_{1:st_k})$ . Then *a priori* density estimation at  $k$  will be updated according to a new observation  $y_k$  via Bayes' formula and using once again the Markov property of the system dynamics as the *posteriori* estimation

$$p(x_k|y_{1:k}) = \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{\int p(y_k|x_k)p(x_k|y_{1:k-1})dx_k}. \quad (8)$$

Under the assumption of linearity, where system and observation dynamics are linear functions of  $x_k$ , the Kalman filter can be derived from the recursive relations (7) and (8). Due to the linearity the conditional probability density function  $p(x_n|y_{1:n})$  becomes Gaussian and characterized by its mean and variance.

The power systems that we are interested in are nonlinear and the probability density functions are no longer Gaussian. As a remedy, an extended Kalman filter (**EKF**) is used in which the state and observation equations are linearized around the current state as a first approximation. In a strongly nonlinear environment these assumptions may undermine filter performance. Furthermore, the application of **EKF** to very high dimensional problems such as power systems, where the dimension of  $x$  and  $z$  for a typical Eastern U.S. power network can be of the order of 5,000 and 50,000 respectively, would be a computationally demanding problem even for state-of-art supercomputers. This is because the **EKF** needs to store and manipulate error covariance matrices, which becomes unmanageable for high dimensional models. In addition to this, the **EKF** assumes that the errors in the state estimates are small and that the state errors propagate through a separate linearized system. Because of this, **EKF** has a fundamental flaw since it is based on the attempt to approximate non-Gaussian density with the Gaussian one through a closure scheme, and this causes several nontrivial problems such as an unbounded error variance growth [4].

A recent, more efficient class of filtering methods is called particle methods, which includes sequential Monte Carlo, ensemble Kalman filters and interacting particle filters. Particle algorithms are techniques for implementing a recursive Bayesian filter by Monte Carlo simulations (see for example, Arulampalam et al [1]). Ensemble Kalman filters (**EnKF**) were proposed to resolve these defects in **EnKF** [4]. Approximating an error covariance matrix by an ensemble of states instead of storing a full matrix certainly reduces the storage demand. Replacing the linearized state evaluation in **EnKF** with integration of each member of the ensemble forward in time using Markov Chain Monte Carlo method made **EnKF** possible to handle nonlinear problems and succeed in many areas of application [5] since its development.

In a similar vein to **EnKF**, the particle filter approach uses an ensemble of states. But the ensemble is used to represent the probability density functions and the update from measurements is done by Bayes' rule (8) instead of the Kalman filter equation. For this reason, particle filters, in principle, are applicable to any nonlinear dynamical system. While there are many variants of particle filters, the basic algorithm underlying them is same. First, the prior density is

approximated by sample particles with weights. These particles are evolved according to system dynamics and once the new measurement  $y_k$  is obtained, new weights are assigned proportional to the likelihood of each prior sample. To avoid degeneracy of the samples and make sure the convergence of the algorithm, re-sampling is used if necessary. These steps are recursively done for each iteration.

In this work we will apply the extended Kalman filter for almost linear applications and the particle filters to the highly nonlinear system state estimation framework and compare their performance. Armed with these algorithms, we should be able to handle a large class of nonlinear filtering problems.

### 3 Software Tools

The heart of the work is the development of an overall system framework for a new approach to power system security monitoring. We are developing globally coordinated monitoring and decision making tools, and this requires global data sharing and coordination. The software tools and methodologies must be designed taking into account the physical power system dynamics and behavior in response to control actions and contingencies. The enabling computer science technologies to achieve this objective are Grid computing.

We are incorporating existing Grid computational tools and developing new tools, both to analyze and interpret electrical power grid state data and to integrate such data into power grid simulations to predict the future state of the system, as described in this paper. The data grid technology to be used in this project includes Web and Grid service technologies and standards, such as the Globus toolkit [6] and the Storage Resource Broker [2]. One way in which we are contributing to Grid computing technology lies in building effective methods for integrating power grid measurement data into both data analysis and simulations in a distributed computing and sensing environment.

The data grid work builds on our prior work on component-based tools for Grid environments, including DataCutter [3] and InterComm [8]. Much of this prior work has focused on providing high performance and flexibility in order to enable components to be effectively deployed onto distributed sets of computational resources. However, for this effort we are developing methods for providing performance guarantees. In the power grid monitoring and control environment, limiting response times is crucial. DataCutter is a system for enabling application developers to write simple functions (called filters) to perform parts of a data analysis application, connect the filters into a graph that performs all the desired processing of the data, and then deploy the filters onto remote computational resources. Standard Grid technologies are used for authentication across multiple sites and for starting up the required services. In addition, we have been building tools to index datasets stored across multiple sites [7], to enable efficient access for certain types of data requests (multidimensional range queries). InterComm is a programming model and runtime library for achieving direct data transfers between data structures managed by multiple parallel and sequential programs. Each program does not need to know in advance (i.e., before a data transfer is

desired) any information about the program on the other side of the transfer. All required information for the transfer is computed at runtime. InterComm allows a program to specify data distribution across processors in each program in a flexible manner. The InterComm framework also includes the infrastructure required to decouple the specification of when data transfers occur between coupled sets of components from the logic within the components. A separate coupling specification, written by the application integrator (the person building the coupled set of components), is used by the framework to specify when data transfers between coupled components should occur.

In this work, both InterComm and DataCutter services are being extended and enhanced to deal with the large, frequent data transfers required from multiple sources to parallel data analysis, control modules, and simulations. More specifically, InterComm services are being integrated with DataCutter services to deploy various components at distributed sites, and tie them all together to perform the required analyses. These services will also be used to couple the sensor data into the simulation models as it is produced and consumed, in a flexible manner. Much of the work also involve providing standard Web and Grid service interfaces, to enable the use of services developed in the wider Grid and Web communities, for security, authentication, information discovery, etc. The design and implementation of those services, including tools and interfaces that can be easily employed by power and control systems engineers, is a major challenge.

## 4 Automated Visualization for Power System Health Monitoring

Visualization techniques are being investigated to maximize the benefit of data streams and data content. Efforts to enable a more meaningful operator view include color contouring of wide-area data and 3-D animation. Examples of the data visualized are network bus voltages, line percent loadings, and area economic data. Displays are being tested under a variety of different simulated operating points to represent different emergency system conditions. The goal of this testing is to determine the displays that are best at helping users actually correct system problems. Our current hypothesis is that such visualization would be useful in helping people see subtle changes in key measures that might be important precursors to impending system problems. For operational use we envision the ability to "play back" animated visualizations showing the variation in the system conditions over a specified time period, similar to the animated weather radar images used to track storms. We will experiment with visualizing other system quantities, such as real power flow, and the hundreds of power transfers that simultaneously take place between the different control areas.

An example of such a data-driven visualization that we have performed is the contours of the per unit voltage magnitudes at several thousand 115/138 kV buses in the Ohio region for the simulated August 14, 2003 conditions immediately before the beginning of the events that led to the infamous blackout. Those contours provide an overview of the voltage profile of the entire region.

Hence the display is quite effective for providing situational awareness. For actual power systems this data has been traditionally supplied in real-time by the SCADA system with typical refresh rates varying between 5 and 30 seconds. In the near future, as PMUs become more ubiquitous, much faster refresh rates will be possible, allowing visualization of a system as it is moving toward voltage instability. The contours may show subtle oscillations in bus voltage magnitudes or angles.

Another possible advantage of using data driven, wide-area visualizations arises because power system operation is it tends to be cyclic, with strong a strong daily, weekly and seasonal cycles (e.g., very high loads on weekday afternoons during the summer, and much lower loads during the weekend nights in the spring or fall). Hence an experienced system operator might readily detect patterns that seem abnormal. For operational use one might envision the ability to "playback" animated visualizations showing the variation in the system conditions over a specified time period, similar to the animated weather radar images used to track storms. For human factors testing we might create animation sequences of different visualizations and test how readily the experiment participants detect something abnormal. We will also experiment with visualizing other system quantities, such as real power flow, and the hundreds of power transfers which simultaneously take place between the different control areas.

## References

1. M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, Vol.50(2), 174-188, 2002.
2. C. Baru, R. Moore, A. Rajasekar, and M. Wan. The SDSC Storage Resource Broker. In *Proceedings of CASCON'98 Conference*, Dec. 1998. <http://www.cas.ibm.com/archives/1998/index.html>.
3. M. Beynon, C. Chang, U. Çatalyürek, T. Kurc, A. Sussman, H. Andrade, R. Ferreira, and J. Saltz. Processing large-scale multidimensional data in parallel and distributed environments. *Parallel Computing*, 28(5):827-859, May 2002. Special issue on Data Intensive Computing.
4. G. Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics, *J. Geophys. Res.*, 99, 10, 14310,162, 1994.
5. G. Evensen. The Ensemble Kalman Filter: Theoretical formulation and practical implementation, *Ocean Dynamics*, 53, 343367, 2003.
6. I. Foster and C. Kesselman. Globus: A metacomputing infrastructure toolkit. *International Journal of Supercomputer Applications and High Performance Computing*, 11(2):115-128, 1997.
7. B. Nam and A. Sussman. Spatial indexing of distributed multidimensional datasets. In *Proceedings of CCGrid2005: IEEE/ACM International Symposium on Cluster Computing and the Grid*. IEEE Computer Society Press, May 2005.
8. J. S. Wu and A. Sussman. Flexible control of data transfers between parallel programs. In *Proceedings of the Fifth International Workshop on Grid Computing - GRID 2004*, pages 226-234. IEEE Computer Society Press, Nov. 2004.