Electric Load Transient Recognition With a Cluster Weighted Modeling Method

Tao Zhu, *Member, IEEE*, Steven R. Shaw, *Senior Member, IEEE*, and Steven B. Leeb, *Fellow, IEEE*

***Abstract*—This paper considers the use of sequential cluster weighted modeling (SCWM) for electric load transient recognition and energy consumption prediction that are promising for isolating the deleterious load transients from delicate renewable sources. Two computational processes co-exist in the SCWM scheme. In the training process, we propose a cluster weighted normalized least mean squares modification of the expectation maximization method to address the singular matrix inversion problem in updating the local model parameters. For the prediction process, we propose a sequential version of the CWM prediction that not only improves the real time performance of load transient recognition, but also resolves online overlapping transients. Other real time transient processing issues are also addressed. The methods are demonstrated using benchmark electric load transients.**

***Index Terms*—Adaptive estimation, clustering methods, electric variables measurement, expectation maximization, Gaussian distributions, least-mean-squares, load forecasting, maximum likelihood estimation, statistical learning.**

I. Introduction

FUTURE SMART grid applications will likely focus on improving the stability and frequency regulation of the electric utility grid. Storage mechanisms may be essential for mitigating fluctuations from renewable sources and demand-responsive loads. A variety of technologies are under consideration for providing electrical storage. A common feature across many electrical storage schemes is a need to protect delicate storages or conversion devices from deleterious electric load transients. In this paper, we present new robust techniques for identifying electric load transients in real time and mitigating the effect of these transients on delicate sources, for example fuel cells, in favor of sources capable of responding to transient demands.

Electric load transients often have significant initial peaks in power relative to the long-range requirements. These transient peaks may have an impact on fuel cell system efficiency and lifetime [1], [2]. Multi-source fuel cell systems incorporating rapid energy storage devices have been proposed to reduce transient effects on fuel cells [3]–[5]. In these applications, the mechanisms for determining how much power the fuel cell delivers during a transient are generally not specific to the type of transient, even though many classes of electric loads have repeatable transients that can be modeled and recognized [6]. We proposed a transient recognition control (TRC) method in [7], in which the TRC module provides a prediction of the long-range behavior of the incoming load based on the initial transient peak information. The TRC model was developed based on the cluster weighted modeling (CWM) method which was originally proposed for stochastic time series analysis and modeling [8]–[10]. The prediction of transient load behavior is of great utility in smart-grid applications more generally. In [11], a load features are selected and analyzed using a neural network with an evolutionary algorithm to forecast load demand. The method is demonstrated on demand data. In [12], wide area synchrophasor measurements are incorporated in an neural network for real-time prediction. This technique is applied to a distribution level model, and a remedial action scheme based on the predictions is proposed. The authors of [13] consider transient detection on a vehicle-level power system using a wavelet based approach. This predictions are used to optimize and facilitate the sharing of demand between a battery and super-capacitor. This paper provides a novel, real-time CWM technique that can use physically based models to provide transient behavior predictions for applications in smart grids.

CWM is a divide-and-conquer method that addresses the global modeling problem by dividing the sample set into a number of subsets where a simple local model can be applied in each subset. CWM addresses the modeling problem through a collection of clusters, using a Gaussian mixture framework to associate inputs with clusters and map inputs to outputs using the local model in each cluster. Model parameters are solved iteratively through an expectation maximization (EM) process that maximizes the joint log-likelihood of all the samples in the training set. The training samples are effectively partitioned by clusters. Only the samples strongly correlated with a cluster will contribute significantly to the parameter tuning of the corresponding local model.

The foundation of the TRC method is presented in mathematical detail in this paper and is further improved to handle real-time transient scaling and situations where multiple transients overlap. Here, we propose a sequential reformulation of the CWM prediction process called *sequential CWM* (SCWM). This method is distinct from other contemporary discriminators used for machine learning, like support vector machines

Manuscript received September 04, 2012; revised January 02, 2013, February 21, 2013; accepted March 27, 2013. Date of publication May 24, 2013; date of current version November 25, 2013. This work was supported by NSF under Grant ECS-0135229 and the High Temperature Electrochemistry Center (HiTEC), Pacific Northwest National Lab contract No. 3971 (413030-A). Paper no. TSG-00548-2012.

T. Zhu was with the Department of Electrical and Computer Engineering, Montana State University, Bozeman, MT 59717 USA. He is now with IPG Photonics, Marlborough, MA 01752 USA (e-mail: zhutao73@yahoo.com).

S. R. Shaw is with the Department of Electrical and Computer Engineering, Montana State University, Bozeman, MT 59717 USA (e-mail: sshaw@matrix.coe.montana.edu).

S. B. Leeb is a with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139, USA, (e-mail: sbleeb@mit.edu).

Digital Object Identifier 10.1109/TSG.2013.2256804

(SVM) and the k-nearest neighbors (kNN) algorithm. These approaches to machine learning do not support sequential estimation. As such, SCWM is qualitatively different from many other approaches to machine learning, and distinctly different from other schemes like artificial neural networks, which fail to provide the high degree of localization offered by SCWM. These benefits of SCWM are essential for smart grid applications, in that SCWM affords real-time, evolving predictions with the ability to discriminate fine differences between electrical loads with similar but distinct transients. SCWM, for example, has the unique benefit of producing a *tail prediction* for a partially received transient, which makes the method uniquely robust to certain transient overlap situations.

In the case of electric load transient modeling, the EM training algorithm may have a singular matrix inversion problem in adjusting the local model parameters because the load transient dimension is large. In this paper, we also develop a normalized least-mean-squares method based on a proposed posterior likelihood weighted mean square error criterion. The method does not have an explicit matrix inversion and requires little computational effort.

The standard cluster weighted modeling method is briefly reviewed in Section II. The proposed cluster weighted normalized least-mean-squares algorithm is also developed in this section. In Section III, the sequential CWM for real time load transient recognition under both overlapping and scaling situations is developed, and transient detection technique is discussed. Simulations are conducted on several benchmark load transients in Section IV, demonstrating the success of the proposed methods. Finally, conclusions and possibilities for future work are outlined in the last section.

II. Cluster Weighted Modeling and Numerical Improvements

A variety of machine-learning and classification techniques, including neural networks and support vector machines, can be applied to the problem of transient recognition in smart grid applications. However, these methods generally do not generalize to real time by producing a new estimate as each point of the incoming transient is received. Also, these methods are not directly adapted to the use of physically based, verifiable local models as is CWM.

A. Cluster Weighted Modeling

CWM is reviewed in this subsection based on the presentation and notations in [14] where more detailed derivations can be found. CWM is a divide-and-conquer method for maximizing the joint probability density $p(y,\vec{x})$ and constructing a function mapping between the input vectors $\vec{x}$ and desired outputs y over a sample set $\{\vec{x}_n, y_n\}_{n=1}^N$ drawn from the problem being studied. The output y is assumed to be scalar in this paper, but generalization to the vector case is straightforward. CWM solves the modeling problem with a mixture of clusters. The structure of a cluster is described by three priors, $\{P(c_m), p(\vec{x}|c_m), p(y|\vec{x},c_m)\}$. The cluster probability $P(c_m)$, where c_m is the cluster label, is the normalized cluster weight, reflecting the relative importance of one cluster to the modeling problem. The input probability density $p(\vec{x}|c_m)$ is the likelihood between the input vector and the cluster that describes the global distribution of clusters in the input space and the local grouping information of input patterns around each cluster. The input-output dependance probability density $p(y|\vec{x},c_m)$ describes the local functional relationship between the input vectors and the desired outputs in one cluster and the uncertainty of the local mapping. The joint likelihood $p(y,\vec{x})$ is a measure of the error between the established global CWM model and the true mapping behind the sample data. $p(y,\vec{x})$ is estimated by

$$p(y,\vec{x}) = \sum_{m=1}^{M} p(y|\vec{x},c_m)\cdot p(\vec{x}|c_m)\cdot P(c_m), \tag{1}$$

where M is the pre-determined number of clusters. For the electric load transient modeling problem, CWM can be applied by assigning the load transient pattern as the CWM input vector $\vec{x}$ and the corresponding load transient long-range value as the desired output y. A cluster c_m could represent a specific transient or a class of transients [7].

The prior probability densities are assumed to be Gaussian distributions, i.e.,

$$p(\vec{x}|c_m) = \prod_{d=1}^{D} \frac{1}{\sqrt{2\pi\sigma_{m,d}^2}} \exp\left[\frac{-(x_d-\mu_{m,d})^2}{2\sigma_{m,d}^2}\right], \tag{2}$$

$$p(y|\vec{x},c_m) = \frac{1}{\sqrt{2\pi\sigma_{m,y}^2}} \exp\left[\frac{-(y-f(\vec{x},\vec{\beta}_m))^2}{2\sigma_{m,y}^2}\right]. \tag{3}$$

In (2), D is the pre-defined vector dimension for the input vectors, $\sigma_{m,d}$ is the dth component of the diagonal variance matrix of the input vector, and $\mu_{m,d}$ is the dth component of the center vector $\vec{\mu}_m$ of cluster c_m. A diagonal variance matrix is used for $p(\vec{x}|c_m)$ to save computation and storage resources because the pattern dimension may be large in this work. In (3), a local model $f(\vec{x},\vec{\beta}_m)$ with parameter vector $\vec{\beta}_m$ is embedded in each cluster as the mean value for $p(y|\vec{x},c_m)$, and $\sigma_{m,y}$ describes the uncertainty of the local model. The role of local modeling of $f(\vec{x},\vec{\beta}_m)$ is apparently shown in the CWM output (prediction) which is defined as the conditional expectation of the model output $\hat{y}$ given the input vector $\vec{x}$,

$$\langle \hat{y}|\vec{x}\rangle = \frac{\sum_{m=1}^{M} f(\vec{x},\vec{\beta}_m)p(\vec{x}|c_m)P(c_m)}{\sum_{m=1}^{M} p(\vec{x}|c_m)P(c_m)}. \tag{4}$$

In (4), the local model outputs are combined by the normalized cluster weighted prior likelihoods $(p(\vec{x}|c_m)P(c_m))/(\sum_{m=1}^{M} p(\vec{x}|c_m)P(c_m))$ of the input pattern to the associated cluster to generate the CWM model output. Distant local models for an input pattern have little effects on the CWM output because their likelihoods are small. The local model could be as simple as a linear model,

$$f(\vec{x},\vec{\beta}_m) = \vec{\beta}_m^T\cdot\vec{x} = \sum_{d=1}^{D}\beta_{m,d}\cdot x_d. \tag{5}$$

This model may suffice even for complicated nonlinear modeling problem because the Gaussian clusters have good localization properties. The linear local model also helps simplify the parameter optimization process.

The parameters in the cluster priors are optimized in the sense of maximizing the joint log likelihood $J = \sum_{n=1}^{N} \log p(y_n, \vec{x}_n)$ over the sample set. The closed form solutions of the optimized parameters can be found by setting the partial derivatives of J with respect to the parameters to zero.

The cluster prior parameters are estimated through a expectation maximization (EM) process that iteratively estimates the prior and posterior parameters.

First, the cluster prior parameters, $\{P(c_m), \vec{\mu}_m, \vec{\sigma}_m^2, \vec{\beta}_m, \sigma_{m,y}^2\}$, are respectively estimated using the posterior likelihood evaluated from the sample set in the last iteration:

$$P(c_m) = \frac{1}{N} \sum_{n=1}^{N} p(c_m | y_n, \vec{x}_n), \tag{6}$$

$$\vec{\mu}_m = \langle \vec{x}_n \rangle_m, \tag{7}$$

$$\sigma_{m,d}^2 = \langle (x_{n,d} - \mu_{m,d})^2 \rangle_m, \tag{8}$$

$$\vec{\beta}_m = \left\langle \vec{x}_n \cdot \vec{x}_n^T \right\rangle_m^{-1} \cdot \langle y_n \cdot \vec{x}_n \rangle_m, \tag{9}$$

$$\sigma_{m,y}^2 = \langle (y_n - f(\vec{x}_n, \vec{\beta}_m))^2 \rangle_m, \tag{10}$$

where the notation $\langle *_n \rangle_m$ is the normalized posterior likelihood weighted estimation that is defined by

$$\langle *_n \rangle_m \equiv \frac{\sum_{n=1}^{N} *_n p(c_m | y_n, \vec{x}_n)}{\sum_{n=1}^{N} p(c_m | y_n, \vec{x}_n)}. \tag{11}$$

In (11), $*_n$ is the individual estimation of the parameter by $(\vec{x}_n, y_n)$; and $(p(c_m|y_n, \vec{x}_n))/(\sum_{n=1}^{N} p(c_m|y_n, \vec{x}_n))$ is the normalized posterior likelihood weight reflecting the relative importance of the nth sample pair in estimating the parameters in the cluster c_m.

Second, the posterior likelihood can be estimated using the updated cluster prior likelihoods and applying the Bayes theorem,

$$p(c_m | y_n, \vec{x}_n) = \frac{p(y_n | \vec{x}_n, c_m) \cdot p(\vec{x}_n | c_m) \cdot P(c_m)}{\sum_{m=1}^{M} p(y_n | \vec{x}_n, c_m) \cdot p(\vec{x}_n | c_m) \cdot P(c_m)}. \tag{12}$$

From a signal processing point of view, CWM can be thought of as a bank of filters as shown in Fig. 1. Each filter can be tuned individually to a specific group (cluster) of signals with similar features. The filter bank output is a probabilistic combination of the outputs of the activated filters that match the input signal. The criterion of choosing the activated filters is the cluster weighted likelihood of input signals to the associated filter.

B. Numerical Improvement

A badly conditioned or singular matrix inversion may occur for the matrix $\langle \vec{x}_n \cdot \vec{x}_n^T \rangle_m^{-1}$ in (9) when D is large. This can happen if the number of parameters of the local model is more than the number of available sample data, i.e., the problem is under-determined. Singular value decomposition (SVD) is suggested in [15] for calculating the inverse matrix. However, SVD involves high computation effort for a large D. A much simpler least-mean-squares (LMS) algorithm based on the CWM framework is proposed to iteratively find the optimal value of $\vec{\beta}_m$. Accordingly, a posterior likelihood weighted mean squares error criterion is defined on the local model output residuals of the given cluster,

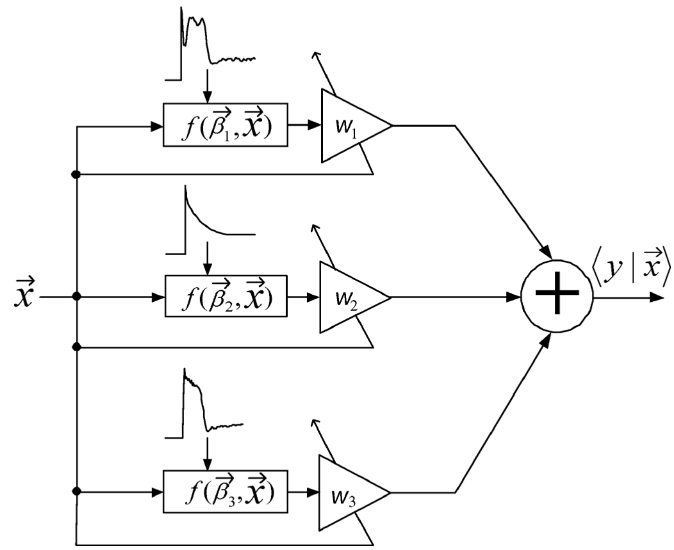

Fig. 1. Graphical explanation of CWM. CWM achieves the complicate nonlinear modeling problem by partitioning the global modeling problem into a series of sub-modeling problems corresponding to clusters. The input x can include a variety of time-resolved physical information, including reactive power, phase angle, or other measurements.

$$J_m = \langle (y_n - f(\vec{x}_n, \vec{\beta}_m))^2 \rangle_m. \tag{13}$$

A LMS algorithm is derived based on the method of stochastic gradient descendent regarding (13). A problem for the LMS estimation of the electric load transients is that the transients normally have significant initial peaks relative to the transient tails. Similar to the gradient estimate noise defined in [16], the estimate of $\beta_{m,j}$ is computed using a gradient with noise

$$N_j = -2\left(y - \vec{\beta}_m^T \vec{x}\right) x_j, j = 1, 2, \ldots, D. \tag{14}$$

which contains terms proportional to x_j^2. Therefore $\hat{\beta}_{m,j}$ for large amplitude x_j may have a long, erratic convergence. A scalar normalized LMS (NLMS) algorithm with respect to an individual $\beta_{m,j}$ that can mitigate the problem was derived in [17]:

$$\hat{\beta}_{m,j}^{(k)} = \hat{\beta}_{m,j}^{(k-1)} + \frac{\rho}{\langle x_{n,j}^2 \rangle_m} \left[\langle y_n x_{n,j} \rangle_m - \hat{\beta}_m^{T,(k-1)} \langle \vec{x}_n x_{n,j} \rangle_m \right]. \tag{15}$$

In (15), the superscript (k) represents the iteration step. $\langle x_{n,j}^2 \rangle_m$ is introduced to normalize the step size ρ for $\beta_{m,j}$. (15) can directly replace (9) in the CWM EM training routine.

As with any other data-driven technique, there is the possibility when updating the parameters of the local models that poor data can produce poor results. However, unlike many other methods, over training is significantly less of a concern when the local model has been selected, e.g., by picking an appropriate order for β, to reflecting the underlying physics of the load. It is anticipated that in smart grid applications, the number of clusters and order of the associated models will be considered in advance and updated as needed through a supervised process.

III. Sequential CWM for Multiple Overlapped Load Transients Recognition

A. Sequential CWM Prediction

The CWM input vector dimension D is a critical parameter for transient recognition control [7]. Conventionally, CWM requires a full set of D data points of an input pattern to make a prediction and results in a delay of D steps. If D is set to be small, the transient pattern may not be effectively modeled and the resulting prediction is not reliable. If D is too large, the output delay may render the prediction useless. In contrast, sequential CWM (SCWM) can generate a prediction when the first data point of a transient is detected, and can update the prediction as more data is received. Under the SCWM prediction scenario, the modeling dimension D is not critical and can be set large enough to include the maximum physical length of transients in the sample set $\{y_n, \vec{x}_n\}_{n=1}^{N}$. In effect, SCWM automatically finds a minimum delay for each transient $\vec{x}_n$ to output an accurate prediction.

Suppose a transient $\vec{x}$ is detected at the time point 1 and the transient recorded by the time point k is $\vec{x}_{(1:k)}$ where the subscript $(1:k)$ marks the segment from the first point to the kth point. For $k \leq D$, the SCWM prediction at the time k is

$$\langle \hat{y}|\vec{x}_{(1:k)}\rangle = \int y p(y|\vec{x}_{(1:k)})dy = \int y \frac{p(y,\vec{x}_{(1:k)})}{p(\vec{x}_{(1:k)})}dy. \tag{16}$$

The dummy un-received transient tail $\vec{x}_{(k+1:D)}$ can be added into the likelihoods $p(y, \vec{x}_{(1:k)})$ and $p(\vec{x}_{(1:k)})$ by

$$p(y,\vec{x}_{(1:k)}) \equiv \int p(y,\vec{x}_{(1:D)})d\vec{x}_{(k+1:D)}, \tag{17}$$

$$p(\vec{x}_{(1:k)}) \equiv \int p(\vec{x}_{(1:D)})d\vec{x}_{(k+1:D)}. \tag{18}$$

Substitute (17) and (18) into (16), and further decompose $p(y, \vec{x}_{(1:D)})$ and $p(\vec{x}_{(1:D)})$ into M clusters and apply the probability expectation for variable y, we have

$$\langle \hat{y}|\vec{x}_{(1:k)}\rangle = \frac{\sum_{m=1}^{M}\int f(\vec{x}_{(1:D)},\vec{\beta}_{m,(1:D)})p(\vec{x}_{(1:D)}|c_m)d\vec{x}_{(k+1:D)}P(c_m)}{\sum_{m=1}^{M}\int p(\vec{x}_{(1:D)}|c_m)d\vec{x}_{(k+1:D)}P(c_m)}, \tag{19}$$

The linear local model output in (19) can be separated into two parts at the time point k,

$$f(\vec{x}_{(1:D)},\vec{\beta}_{m,(1:D)}) = \vec{\beta}_{m,(1:k)}^{T}\cdot\vec{x}_{(1:k)} + \vec{\beta}_{m,(k+1:D)}^{T}\cdot\vec{x}_{(k+1:D)}. \tag{20}$$

Substitute (20) back into (19) and note that

$$\begin{aligned}\int \vec{x}_{(k+1:D)}p(\vec{x}_{(1:D)}|c_m)d\vec{x}_{(k+1:D)} &= p(\vec{x}_{(1:k)}|c_m)\int \vec{x}_{(k+1:D)}p(\vec{x}_{(k+1:D)}|c_m)d\vec{x}_{(k+1:D)}\\ &= \vec{\mu}_{m,(k+1:D)}p(\vec{x}_{(1:k)}|c_m),\end{aligned} \tag{21}$$

therefore the SCWM prediction at time k is derived to be

$$\langle \hat{y}|\vec{x}_{(1:k)}\rangle = \frac{\sum_{m=1}^{M}[f(\vec{q}_m,\vec{\beta}_m)p(\vec{x}_{(1:k)}|c_m)P(c_m)]}{\sum_{m=1}^{M}[p(\vec{x}_{(1:k)}|c_m)P(c_m)]}, \tag{22}$$

where

$$\vec{q}_m = \left(\vec{x}_{(1:k)}^{T}\ \vec{\mu}_{m,(k+1:D)}^{T}\right)^{T} \tag{23}$$

is the reconstruction of the complete input transient vector by cluster c_m. $\vec{\mu}_{m,(k+1:D)}$ is the prediction by cluster c_m for the transient tail $\vec{x}_{(k+1:D)}$ not yet received. A reconstruction $\vec{q}_m$ of the complete input pattern $\vec{x}_{(1:D)}$ by cluster c_m is available at any time of the sequential prediction process. The reasonableness of each cluster's prediction is evaluated by the corresponding normalized cluster weighted partial input prior likelihood $(p(\vec{x}_{(1:k)}|c_m)P(c_m))/(\sum_{m=1}^{M} p(\vec{x}_{(1:k)}|c_m)P(c_m))$. Implausible reconstructions, such as $\vec{x}_{(1:k)}$ being the first points of an incoming bulb transient and $\vec{\mu}_{m,(k+1:D)}$ being the remaining tail of the lathe's cluster center, may significantly affect the model output accuracy at the beginning—it is clearly impossible to predict transient behavior if only one or two points have been received. Empirically, only a small initial part of a load transient is needed for accurate prediction of the remaining transient tail, and the delay of the transient recognition is small.

B. Sequential Resolution of Overlapping Load Transients

Time-overlap of transients poses a challenge for SCWM in real load monitoring environments. As shown in Fig. 2, the transient $\vec{x}_2$ overlaps and is significantly deformed by the tail of transient $\vec{x}_1$ after time k. This could cause the recognition and the long-range behavior predictions for both $\vec{x}_1$ and $\vec{x}_2$ to fail. However, the prediction of the un-received transient tail of $\vec{x}_1$ by each cluster derived in the last subsection can be subtracted from the composite signal to recover $\vec{x}_2$. This combined prediction is the *Cluster Weighted Tail Prediction* and is defined as $\langle \hat{\vec{x}}_{(k+1:L)}|\vec{x}_{(1:k)}\rangle$, where L is the physical length of transient $\vec{x}$, in excess of the CWM input vector dimension D used in the last subsection. Through a derivation similar to the one in SCWM prediction in the last subsection, the closed form of cluster weighted tail prediction is

$$\langle \hat{\vec{x}}_{(k+1:L)}|\vec{x}_{(1:k)}\rangle = \frac{\sum_{m=1}^{M}\left[\vec{\mu}_{m,(k+1:L)}p(\vec{x}_{(1:k)}|c_m)P(c_m)\right]}{\sum_{m=1}^{M}\left[p(\vec{x}_{(1:k)}|c_m)P(c_m)\right]}. \tag{24}$$

As shown in (24), the cluster weighted tail prediction is determined by weighting all of the clusters' predictions with the corresponding normalized cluster weighted partial input prior likelihoods, with the same structure as (4) and (22). When k is greater than the necessary modeling dimension of $\vec{x}$, the irrelevant predictions vanish in (24) because the associated likelihoods are small, and the tail prediction $\langle \hat{\vec{x}}_{(k+1:L)}|\vec{x}_{(1:k)}\rangle$ approximates the true signal $\vec{x}_{(k+1:L)}$.

The transient overlapping resolution is illustrated in Fig. 2. The transient $\vec{x}_{2,(1:L_2)}$ can be recovered by subtracting $\langle \hat{\vec{x}}_{1,(k+1:k+L_2)}|\vec{x}_{1,(1:k)}\rangle$ from the overlapped signals. The dashed line in Fig. 2 is the predicted tail of $\vec{x}_{1,(1:k)}$ calculated

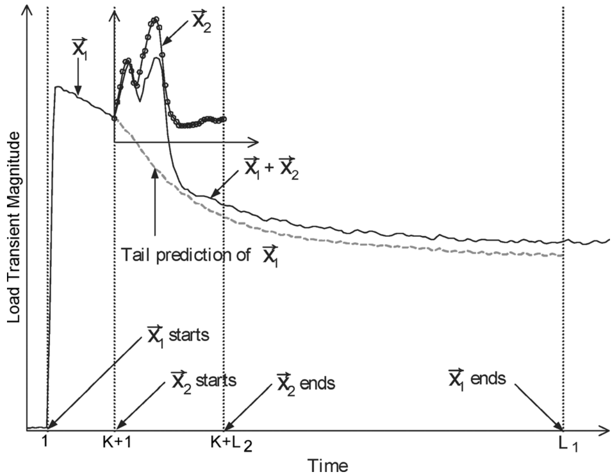

Fig. 2. Transient overlap elimination. Transient $\vec{x}_1$ (with physical length L_1) is detected at time 1. Transient $\vec{x}_2$ (with physical length L_2) is detected at time $k+1$. $\vec{x}_2$ is overlapped and therefore deformed by the tail of $\vec{x}_1$. The true $\vec{x}_2$ vector (shown by the line marked with circles) can be recovered by subtracting the tail of $\vec{x}_1$ from the overlapped signals. $\vec{x}_1$'s tail can be estimated by the cluster weighted tail prediction defined in (24) and is shown by the dashed line in the figure.

by (24). The recovered $\vec{x}_2$ shape is shown by the line marked with circles. Like (22), the cluster weighted tail prediction in (24) is also formulated sequentially, therefore the transient overlapping resolution can be performed online, along with the SCWM prediction process. The tail prediction error at time k for the ith component $x_i (k < i \le L)$ is

$$\begin{aligned}\varepsilon_i^2(k) &= \int (x_i - \langle \hat{x}_i | \vec{x}_{(1:k)} \rangle)^2 p(x_i|\vec{x}_{(1:k)}) dx_i,\\ &= \frac{\sum_{m=1}^{M}\left[\sigma_{m,i}^2 p(\vec{x}_{(1:k)}|c_m)P(c_m)\right]}{\sum_{m=1}^{M}\left[p(\vec{x}_{(1:k)}|c_m)P(c_m)\right]}\\ &\quad + \frac{\sum_{m=1}^{M}\left[\mu_{m,i}^2 p(\vec{x}_{(1:k)}|c_m)P(c_m)\right]}{\sum_{m=1}^{M}\left[p(\vec{x}_{(1:k)}|c_m)P(c_m)\right]}\\ &\quad - \langle \hat{x}_i | \vec{x}_{(1:k)} \rangle^2. \end{aligned} \tag{25}$$

The first item in (25), $(\sum_{m=1}^{M}[\sigma_{m,i}^2 p(\vec{x}_{(1:k)}|c_m)P(c_m)])/(\sum_{m=1}^{M}[p(\vec{x}_{(1:k)}|c_m)P(c_m)])$, is the essential uncertainty of the transient signals and is therefore the theoretical upper limit of SCWM tail prediction precision. The item $(\sum_{m=1}^{M}[\mu_{m,i}^2 p(\vec{x}_{(1:k)}|c_m)P(c_m)])/(\sum_{m=1}^{M}[p(\vec{x}_{(1:k)}|c_m)P(c_m)]) - \langle \hat{x}_i|\vec{x}_{(1:k)}\rangle^2$ is the extra "model prediction error" due to the implausible transient recognition to the irrelevant cluster in the early stage of the prediction. The model will converge to the right cluster (i.e., the relevant likelihoods converge to 1 and irrelevant likelihoods varnish) very soon as more points of transient signals being received. The model prediction error finally disappears and the total prediction error converges to $\sigma_{m,i}^2$. Meanwhile, $\sigma_{m,i}^2$ empirically drops quickly along the time, i.e., $\sigma_{m,i}^2 < \sigma_{m,j}^2$ for $i > j$. The SCWM predictions converge along these two co-exist error dropping curves.

In online applications, the true physical length of the current transient needs to be estimated to determine whether the current transient is still active when a new transient is detected, and accordingly to turn on/off the transient overlapping resolution procedure. The expected transient length of $\vec{x}_n$ can be estimated by an equation similar to (24), i.e.,

$$\langle \hat{L}_n | \vec{x}_{n,(1:k)} \rangle = \frac{\sum_{m=1}^{M}\left[L_m p(\vec{x}_{n,(1:k)}|c_m)P(c_m)\right]}{\sum_{m=1}^{M}\left[p(\vec{x}_{n,(1:k)}|c_m)P(c_m)\right]}, \tag{26}$$

where L_m is the expected "transient length" of cluster c_m that is evaluated in the training process. Similar to (6)–(10) in the training process, L_m can be estimated by

$$\begin{aligned} L_m &= \frac{\sum_{n=1}^{N} L_n \cdot p(c_m|y_n, \vec{x}_{n,(1:D)})}{\sum_{n=1}^{N} p(c_m|y_n, \vec{x}_{n,(1:D)})}\\ &= \langle L_n \rangle_m. \end{aligned} \tag{27}$$

Illustrated by Fig. 2, if $k+1 < \langle \hat{L}_1 | \vec{x}_{1,(1:k)} \rangle$, then the transient overlapping resolution procedure will be turned on to recover transient $\vec{x}_2$; if $k+1 > \langle \hat{L}_1 | \vec{x}_{1,(1:k)} \rangle$, then $\vec{x}_2$ is not overlapped by $\vec{x}_1$ and can be processed by SCWM directly.

Tail prediction method for transient overlapping resolution, illustrated in Fig. 2, may not work perfectly in all situations. A worst case for tail prediction is that two different transients have identical initial transient segments (e.g., $\vec{x}_{1,(1:k)}$ in Fig. 2), and different remaining tails (e.g., $\vec{x}_{1,(k+1:L_1)}$ in Fig. 2) from each other. The tail prediction in this situation could be false, and cause the transient overlapping resolution fail. However, the possibility of the worst case of tail prediction can be predicted from the CWM model cluster mean parameters. Any two clusters having the identical initial segments and different tails could be marked.

C. Online Load Transient Scaling

An empirical observation important to the non-intrusive load monitor [6] is that transients from differently sized but physically similar loads tend to be similar up to scale factors in amplitude and time. The storage and computational efficiency of the transient recognition can be improved by introducing an amplitude scale factor a and an amplitude offset b between the detected transient and the cluster centers. The offset b can be continuously estimated by an appropriate unit gain low pass filter. The scale factor a for the most relevant cluster is determined by sequentially solving the two-step maximum likelihood estimation problem,

$$a = \operatorname{argmax}_{a_m} \log p(\tilde{\vec{x}}_{(1:k)}|c_m), \tag{28}$$

$$\tilde{\vec{x}}_{(1:k)} = a_m \cdot (\vec{x}_{(1:k)} - b), \tag{29}$$

where a_m is the maximum likelihood scale factor between $\vec{x}_{(1:k)}$ and $\vec{\mu}_{m,(1:k)}$ for cluster c_m. The maximum likelihood solution for a_m is

$$a_m = \frac{\sum_{i=1}^{k}[\mu_{m,i}(x_i - b)/\sigma_{m,i}^2]}{\sum_{i=1}^{k}[(x_i - b)^2/\sigma_{m,i}^2]}. \tag{30}$$

Load transients are pre-scaled by the coefficients a and b before being used for the SCWM prediction. There are limits imposed on the scaling range for each a_m to prevent scaling that may lead to poor results, especially at the beginning of the sequential prediction. The pre-scaling operation is conducted sequentially and integrated with the SCWM process.

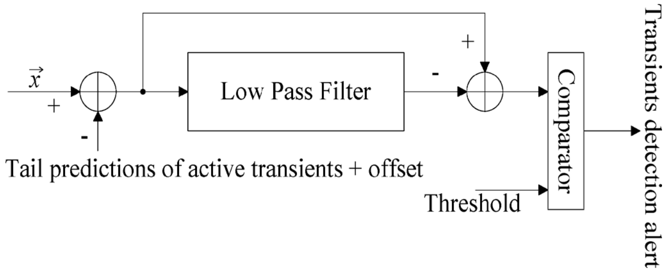

Fig. 3. Change of mean detector combining tail prediction method for transient detection in overlapping situation.

D. Online Transient Detection and Off-Training Set Transient Indication

Electric load transients typically have a sudden change at the beginning of the transient, making them relatively easy to detect. The change-of-mean detector proposed in [6] is applied in this paper for transient detection. The detector compares the difference between the input and a low-pass filtered version of the input. If the difference is greater than a threshold, a new transient is detected.

A difficulty with this simple scheme is that it may not work properly when transients overlap. Fig. 3 shows one method to solve the problem by using the SCWM tail prediction of the overlapped transient. In Fig. 3, the tail predictions of currently evolving transients are subtracted from the input before the change-of-mean detector.

One of the advantages of SCWM is that the likelihood of the input with respect to the clusters is continuously available as new data arrive. This likelihood can be used to detect "off training set" inputs that might produce spurious outputs. In a practical implementation, a threshold monitoring this likelihood could be used to both protect the system from "new" transients and also to isolate new transients for possible future training.

IV. Results

The load transient recognition algorithm introduced here is a "silver bullet" for bringing energy storage to smart grids. This method permits optimization of energy and power flow from different sources. For example, it can optimize the relative contributions of sources with high power and low energy densities in combination with sources that provide low power density but higher energy density. Optimized combinations perfectly supply load transient and steady-state demands while protecting expensive energy storage sources. The proposed approach is demonstrated with a scenario using real-time electric load transients, including situations with multiple transient overlap and while other loads are operating.

Load transients can be characterized by one or more narrow segments with relatively high derivative or mean value variation information, called v-sections [18]. Using v-sections instead of the entire transient to model the load transients is useful because the v-sections involve less computation and storage requirements. The individual v-section can be itself viewed as an independent transient because it is only necessary to predict the long-range value change following each v-section. In this paper, informative v-sections were used for load transient modeling.

TABLE I
Assignments of Training/Testing Samples and Clusters With Respect to Different Load Transients

Transient	Number of training transients	Number of testing transients	Number of clusters
Lathe	40	8	3
Monitor v-section1	30	6	6
Monitor v-section2	30	6	6
Bulb	51	17	2
Drill	30	10	3
Vacuum	28	9	2
Total	209	56	22

Simulations were conducted in Matlab. Five types of benchmark load transients (including transients from a lathe, computer monitor, bulb, drill, and vacuum cleaner) were measured from ac loads supplied by an invertor. Current was measured from the dc bus between the converter and inverter. The recorded data stream was pre-processed to eliminate the 120 Hz ripple from the inverter. Then the v-sections of the load transients were extracted, sorted to different transient classes, and synchronized with others within the same class. A total of 265 transients were recorded. Each class of transients is split into two parts, one subset used for CWM modeling (training), and the other subset used for the primitive testing of the SCWM prediction performance after training. The assignments of load transients with respect to training and testing subsets are shown in Table I. Finally a fully functional SCWM module was implemented using CWM model parameters and verified using test data assembled from the measured current transients but with added noise, scaling in amplitude of transients, and overlapping of transient profiles.

A. CWM Training

The expectation maximization and normalized least-mean-squares mixed algorithm was used to initialize the parameters $\{\vec{\mu}_m, \vec{\sigma}_m^2, \vec{\beta}_m, \sigma_{m,y}^2, P(c_m)\}_{m=1}^M$. $\vec{\sigma}_m^2$ includes the main diagonal components in the variance matrix of $p(\vec{x}|c_m)$. The Gaussian likelihoods were manipulated in log domain to overcome the finite machine precision problem. A numerical improvement involves scaling the likelihood $p(\vec{x}_n|c_m)$ and $p(y_n|\vec{x}_n, c_m)$ in (12) by the same scale factor, which does not affect evaluation of the posterior $p(c_m|y_n, \vec{x}_n)$ or the EM update. A useful scale factor is the maximum value of $p(\vec{x}_n|c_m) \cdot p(y_n|\vec{x}_n, c_m) \cdot P(c_m)$, so that at least one item in the summation in the denominator of (12) has a non-zero value that can be easily represented in the machine.

The CWM input vector dimension for transients was set to $D = 50$ and a 100 ms sampling interval was used. Cluster probabilities $\{P(c_m)\}_{m=1}^M$ were equally initialized to $1/M$ because uniform distribution suggests the maximum likelihood estimation if no prior information exist at all. Twenty-two clusters were used in this example to model six classes of transients/v-sections. The allocation of clusters to different transient classes is summarized in Table I. The number of clusters per transient can be adjusted in a supervised off-line process to

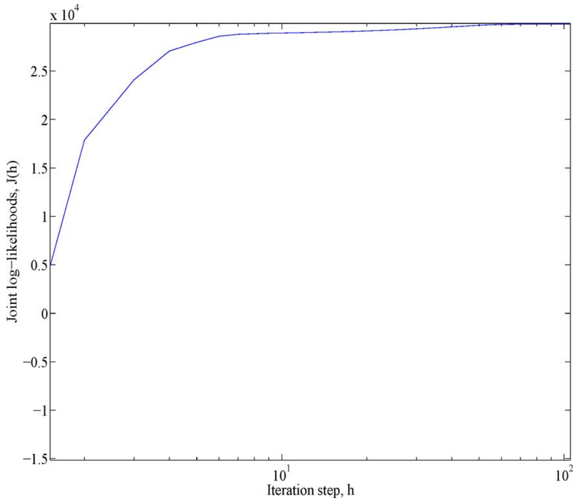

Fig. 4. The convergence curve of the joint log-likelihood over the training samples. The variable $J(h) = \sum_{n=1}^{N} \log p(y_n, \vec{x}_n)$, as defined in Section II. The variable $J(h)$ extends beyond zero because the two Gaussian $p(\vec{x}_n|c_m)$ and $p(y_n|\vec{x}_n, c_m)$ are re-scaled to prevent division by zero.

handle the range of variability within a class of transients. Because CWM generates the probability of an observation with respect to the set of clusters, the method is self-diagnostic with respect to whether the number of clusters is sufficient. The clusters used to model each class are initialized to the average center of that class, adding small random perturbations so that the clusters adapt to span the transient variability. Local model parameters $\{\vec{\beta}_m\}_{m=1}^{M}$ were initialized with small random numbers. The variances of two Gaussian distributions were initialized to be a constant 10. During training, it is also necessary to add small constants to the variances to prevent them from shrinking to zero [14]. The adaptation step size for NLMS is 0.002. One thousand iterations were performed for training. Cluster parameters were checked by hand after training to ensure that no degenerate cases, i.e., two identical clusters, resulted. The final convergence curve of the joint log-likelihood over the sample set is shown in Fig. 4, which suggests that indeed 50–100 iterations are sufficient in this example.

B. Testing

A separate set of transient data, *not* used for training, was used for testing. Training and test data sets are different. The test data sets include noise and different power amplitudes from the training set data. Two kinds of testing were conducted. The first test used 56 primitive v-sections, similar to the ones used in training, to verify whether the SCWM prediction converged accurately and quickly in the transient scaling situation. The second test used the continuous transient stream data to evaluate the fully functional SCWM module implemented in Matlab, including the pseudo-online signal pre-filtering, transient detection, scaling, recognition, and elimination of transient overlap.

The results of the first test are shown in Fig. 5. The dashed lines shown in each sub-figure describe the family of load transients under scales different from the default scale used in training. The solid lines show the SCWM prediction of the transients' long-range behaviors, and the cross-marked solid lines show the tail prediction errors of the corresponding partially received transients. The tail prediction error is defined as

$$e(k) = \|\vec{x}_{(k+1:D)} - \langle \hat{\vec{x}}_{(k+1:D)} | \vec{x}_{(1:k)} \rangle \|. \qquad (31)$$

Early in the observation of the transient, there are significant errors in the tail predictions and consequently in the SCWM outputs. This is acceptable since it is impossible to predict the future behavior of a transient depending on only one or two data points. However, in almost all cases, the tail prediction and the SCWM output settle accurately and quickly compared to the transient length. In practice, a "lock-out" interval can help block the initial behavior.

A fully functional SCWM module for the second test includes signal filtering, transient detection, scaling, overlap resolution, and long-range transient behavior prediction. A new transient processing line is set up for each newly detected transient, and any processing lines are marked as active if the associated transients are still evolving. The active status is checked by the associated transient starting point and expected transient length calculated by (26). When a new transient is detected, the outcomes of the currently active transient processing lines (such as the tail prediction, the expected transient length, and the long-range behavior prediction) are not updated anymore but remain available for future use. The long-range behavior outcomes of the inactive transient processing line are used to update the system offset one time before the line is closed. We assume a new transient comes in not overlapping the current transient which is still in "lock-out" period. The performance of the fully functional SCWM module is verified in the pseudo real-time Matlab environment.

Three simulations were conducted to verify the fully functional SCWM module, including a full computer monitor transient (including two v-sections), a mixed drill-bulb transients stream, and a mixed vacuum cleaner-lathe transients stream. The results are shown in Fig. 6. The transient signals after pre-filtering are shown in figures for the purpose of visibility. The SCWM output within the "lock-out" period is not blocked in the figure in order to show the full details of the SCWM action. The performance of the fully functional SCWM module, especially the transient overlapping resolution through the idea of tail prediction, were verified by the presented results.

V. Conclusion

Cluster weighted modeling was studied in this paper for electric load transient recognition. A mixed expectation maximization and normalized least-mean-squares algorithm was proposed to solve the singular matrix inversion problem in the model training process and also consequently to improve the numerical stability and save computational cost. A sequential modification of CWM prediction (SCWM) was developed to solve the real time transient recognizing and predicting problem. A sequential transient overlapping elimination method was also developed based on the idea of tail prediction for the partially received transient head. Other electric transient processing issues like scaling and transient detection under overlapping were also discussed. Several benchmark transient

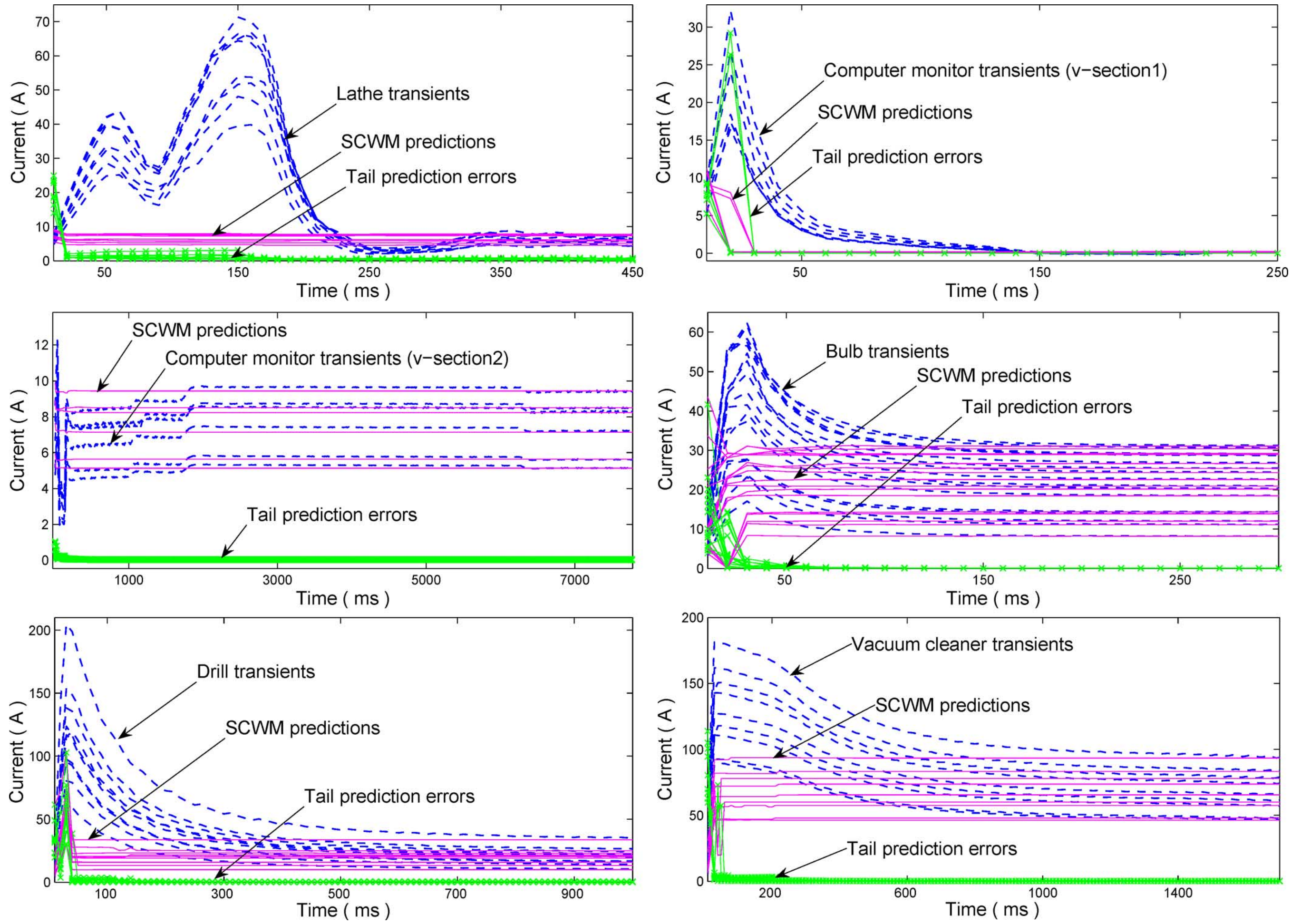

Fig. 5. SCWM test of prototype electric load transients with different scale factors. The dashed lines describe the family of load transients under scales different from the default scale used in training. The solid lines show the SCWM prediction of the transients' long-range behaviors, and the cross-marked solid lines show the tail prediction errors of the corresponding partially received transients.

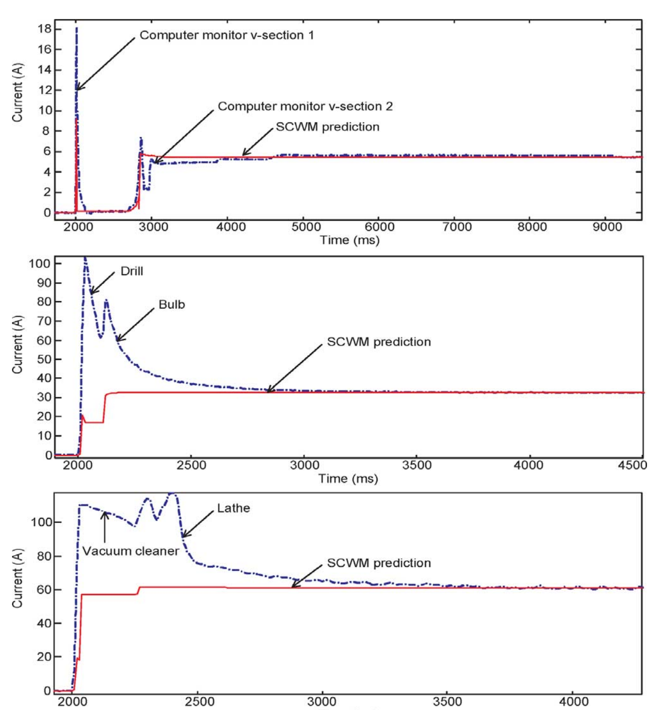

Fig. 6. SCWM test of concatenated and overlapping load transients.

examples were used in the simulation to verify the developed model and the possibility of implementing the SCWM in real time parallel processing hardware. Future work will be concentrated on the improvement of the real time implementation of the SCWM module on FPGA that is to be embedded into real power system controls.

REFERENCES

[1] R. S. Gemmen, "Analysis for the effect of inverter ripple current on fuel cell operating condition," *Trans. ASME, J. Fluids Eng.*, vol. 125, pp. 576–585, May 2003.

[2] K. Acharya, S. K. Mazumder, and P. K. Burra, "System interaction analyses of solid oxide fuel cell (SOFC) power conditioning system," in *Proc. IEEE Ind. Appl. Conf., 38th IAS Annu. Meet.*, Oct. 2003, pp. 2026–2032.

[3] K. W. Park, H. J. Ahn, and Y. E. Sung, "All-solid-state supercapacitor using a nafion polymer membrane and its hybridization with a direct methanol fuel cell," *J. Power Sources*, vol. 109, pp. 500–506, 2002.

[4] R. Nojima, I. Takano, and Y. Sawada, "Transient performance of a new-type hybrid electric power distribution system with fuel cell and SMES," in *Proc. 27th Annu. Conf. IEEE Ind. Electron. Soc. (IECON'01)*, pp. 1303–1308.

[5] L. P. Jarvis, P. J. Cygan, and M. P. Roberts, "Hybrid power source for manportable applications," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 18, pp. 13–16, Jan. 2003.

[6] S. R. Shaw, "System identification techniques and modeling for non-intrusive load diagnostics," Ph.D. thesis, Massachusetts Inst. Technol., Cambridge, MA, USA, Feb. 2000.

[7] T. Zhu, S. R. Shaw, and S. B. Leeb, "Transient recognition control for hybrid fuel cell systems," *IEEE Trans. Energy Convers.*, vol. 21, no. 1, pp. 195–201, Mar. 2006.

[8] B. Schoner, C. Cooper, C. Douglas, and N. Gershenfeld, "Data-driven modeling and synthesis of acoustical instruments," in *Proc. Int. Comput. Music Conf.*, Ann Arbor, MI, USA, Oct. 1998.

[9] B. Schoner and N. Gershenfeld, "Data-driven modeling of nonlinear microwave devices," in *Dig. 53rd ARFTG Conf. Nonlinearity Characterization*, Anaheim, CA, USA, Jun. 1999.

[10] B. Schoner, C. Cooper, and N. Gershenfeld, "Cluster-weighted sampling for synthesis and cross-synthesis of violin family instruments," in *Proc. Int. Comput. Music Conf.*, Berlin, Germany, Aug. 2000.

[11] N. Amjady, F. Keynia, and H. Zareipour, "Short-term load forecast of microgrids by a new bilevel prediction strategy," *IEEE Trans. Smart Grid*, vol. 1, no. 3, pp. 286–294, Dec. 2010.

[12] F. Hashiesh, H. E. Mostafa, A. R. Khatib, I. Helal, and M. M. Mansour, "An intelligent wide area synchrophasor based system for predicting and mitigating transient instabilities," *IEEE Trans. Smart Grid*, vol. 3, no. 2, pp. 645–652, Jun. 2012.

[13] C. Sen, Y. Usama, T. Carciumaru, X. Lu, and N. C. Kar, "Design of a novelwavelet based transient detection unit for in-vehicle fault determination and hybrid energy storage utilization," *IEEE Trans. Smart Grid*, vol. 3, no. 1, pp. 422–433, Mar. 2012.

[14] N. Gershenfeld, *The Nature of Mathematical Modeling*. Cambridge, U.K.: Cambridge Univ. Press, 1999.

[15] B. Schoner and N. Gershenfeld, , A. Mees, Ed., "Cluster-weighted modeling: Probabilistic time series prediction, characterization and synthesis," in *Nonlinear Dynamics and Statistics*. Boston, MA, USA: Birkhaeuser, 2000, pp. 365–385.

[16] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1985.

[17] T. Zhu, "Extended cluster weighted modeling methods for transient recognition control," Ph.D. thesis, Montana State University, Bozeman, MT, USA, Jun. 2006.

[18] S. R. Shaw, C. B. Abler, R. F. Lepard, D. Luo, S. B. Leeb, and L. K. Norford, "Instrumentation for high performance nonintrusive electrical load monitoring," *J. Sol. Energy Eng.*, vol. 120, pp. 224–229, 1998.

Tao Zhu received his B.E. and M.E. degrees, both from Tianjin University, China, in 1996 and 1999, respectively. He graduated from the Electrical and Computer Engineering Department, Montana State University, Bozeman, MT, USA. He is interested in signal processing, pattern recognition, and system identification. He is currently an electrical engineer in IPG Photonics, Marlborough, MA, USA.

Steven R. Shaw received his Ph.D. degree in electrical engineering from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2000. He is currently an Assistant Professor in the Department of Electrical and Computer Engineering, Montana State University, Bozeman, MT, USA. He is interested in sensors, instrumentation, modeling, and numerical and computational methods associated with control and measurement problems.

Steven B. Leeb received his Ph.D. degree in electrical engineering and computer science from the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, in 1993. He has been a member of the MIT faculty in the Department of Electrical Engineering and Computer Science since 1993. He currently serves as an Associate Professor in the Laboratory for Electromagnetic and Electronic Systems. Dr. Leeb is concerned with the design, analysis, development, and maintenance processes for all kinds of machinery with electrical actuators, sensors, or power electronic drives.