Load Forecasting via Diversified State Prediction in Multi-Area Power Networks

Ali Tajer, *Senior Member, IEEE*

***Abstract*—Load forecasting has a pivotal role in operating electric utilities. This paper proposes a learning-based tool for short-term load forecasting via first predicting the state parameters of the grid and then delineating load forecasts as functions of the predicted state parameters. Such indirect load forecasting via predicting state parameters leads to improved forecast quality due to the inherent diversity in predictions for state parameters. Specifically, due to the strong inter-connectivity among network components, some state parameters are shared by multiple substations. Hence, allowing each substation to provide local predictions for its associated state parameters based on its local dynamics enables providing multiple predictions for each shared state parameter, leading to a diversified set of predictions for it. Our analysis shows that the proposed aggregation framework provides a prediction for each state parameter such that the quality of this prediction equals to that of the best local prediction provided for that state parameter. This implies that this framework can identify and track the best local prediction for each state parameter without requiring any knowledge about network dynamics.**

***Index Terms*—Diversity, load forecasting, multi-area network, state prediction, support vector regression.**

I. Introduction

A. Background

DRIVEN by the needs for efficient economic dispatch, load forecasting, which is responsible for delineating the amount of energy to be delivered at different geographical locations over different future time horizons, is instrumental for production planning, designing energy trading approaches and investment portfolios, and enhancing the operational reliability of the grid. Moreover, the need for accurate prediction approaches is expected to grow well into the future with the advent of new technologies (e.g., electric vehicles), advances in sensing and data acquisition technologies, institutional deregulations that allow integrating private and small producers into the grid, and abundant presence of intermittent energy sources.

There exists a rich literature on electricity load forecasting for different time horizons (i.e., short, middle, and long terms) and different demand scales (i.e., individual and regional). Short-term load forecasting is essential for monitoring and controlling the power flow in the network on the sub-annual scales (e.g., hourly, daily, weekly, and monthly). Long-term load forecasting, on the other hand, copes with forecasting the demands on the annual-basis and is governed by multiple factors including the spatial and temporal expansions of the networks, end-user appliances, urbanization dynamics, customers behavior, and population variations, to name a few.

A wide variety of prediction models exist in the literature that differ in complexity, underlying statistical assumptions, availability of historical data, and prediction procedures. Some major existing approaches include the following methods (and references therein): semi-parametric models [1], stochastic methods [2], hierarchical and clustering methods [3], [4], auto-regression algorithms [5]–[7], multiple regression [8], support vector regression [9], high-dimensional and non-linear regression [10], iterative weighted least-squares [11], consumption segmentation [12], neural networks [13], support vector machines [14], fuzzy inductive reasoning [15], and knowledge-based approaches. Depending on the network dynamics these methods offer different advantages and, consequently, selecting an appropriate method for a network strongly hinges on the model known about the network.

B. Contribution

Multi-area power grids are growing in scale, interconnectivity, complexity, and their distributed structures. While these lead to increased volume and complexity of the data generated in power grids, processing which creates challenges in operating the power grids, they also create structures in the generated data, which once exploited judiciously, can bring about improvements in operating the power grid. Specifically:

1) Different subnetworks in large multi-area networks overlap through shared consuming, generating, sensing, and controlling modules. Such overlaps imply coupling among the dynamics of power flow in different areas, which in turn indicates that different areas can provide different amount of information with variable levels of accuracy about different load constituents.
2) There exists redundancy in the measurements in the network and number of sensors significantly dominates the number of state (bus voltage and phase angle) parameters.

Manuscript received June 23, 2015; revised October 31, 2015 and January 21, 2016; accepted February 23, 2016. Date of publication March 11, 2016; date of current version October 19, 2017. This work was supported by the National Priorities Research Program through the Qatar National Research Fund (a Member of Qatar Foundation) under Grant 6-149-2-058. Paper no. TSG-00718-2015.

The author is with the Electrical, Computer, and Systems Engineering Department, Rensselaer Polytechnic Institute, Troy, NY 12180 USA (e-mail: tajer@ecse.rpi.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSG.2016.2535200

By capitalizing on these measurement redundancies and overlaps in multi-area networks, the contribution of this paper is to design a framework that exploits these measurement redundancy and structures and uses them *in conjunction* with any prediction method in order to improve their prediction qualities. Specifically, in order to exploit measurement redundancy and structure overlaps, in this framework we propose to perform load forecasting indirectly via first predicting the state parameters (bus voltages and phase angles) of the system and then predicting the loads by using the connection between state parameters and electric loads. Such an indirect approach is especially beneficial in multi-area networks in which some consuming, generating, sensing, or controlling modules are shared by multiple areas, and their associated state parameters can be predicted independently in different areas, and possibly based on different prediction models. These diversify the number of predictions available for some state parameters, and if such diverse predictions are combined appropriately, they can collectively produce a more reliable prediction. Hence, through this indirect approach, we can leverage the structure and the inherent *diversity* in the data to improve the quality of the predictions of the state parameters, which in turn leads to improvement in the quality of load forecasts. Other specifications of the proposed framework are:

- Different subnetworks can choose their local best prediction model that matches their local physical characteristics and power flow dynamics.
- The subnetworks do not need to be aware of each others prediction strategies and no exchange of such information among the subnetworks is necessary for this purpose.
- For state parameters with multiple predictions, the framework is guaranteed (asymptotically) to track the best prediction over time.
- Prediction does not necessitate any central supervision and can be implemented in a fully distributed manner. This is inline with the envisioned futuristic grids that need to implement appropriate measures that allow for distributed information processing. The advantages of distributed processing include reduced communication load, controlled computational complexity, data privacy, and data security.
- Combining different predictions has very low computational complexity.

The notion of forecasting individual bus loads instead of forecasting the aggregate load is also studied in [16], such bus load prediction is performed to facilitate operations whereas evaluating stability margins and estimating voltage collapse points. The major distinction of [16] is that it deploys a combination of extended Kalman filters and neural networks to predict the state parameters and bus loads, and does not exploit the inherent diversity in the state parameters. The aggregation method deployed in this framework follows the multiple expert advice framework (see [17]). This expert advice framework is also used in [18] for the purpose of load forecasting. The major distinction of this paper with [18] is that, in this paper the focus is on performing state prediction as the initial step, and then leveraging it for load prediction, while [18] performs load prediction directly.

II. Preliminaries and Notations

A. Network Model

An energy grid consisting of N interconnected and geographically distributed subnetworks is considered. Each subnetwork is comprised of a variety of consuming, generating, sensing, and controlling modules, which can be potentially shared by multiple subnetworks, especially adjacent ones. The state of the grid, that is the voltage magnitudes and phase angles of the buses, at time t is denoted by

$$\boldsymbol{x}_t \overset{\triangle}{=} [x_t[1], \dots, x_t[m]]. \tag{1}$$

The measurements at time t is denoted by $\boldsymbol{y}_t \in \mathbb{R}^k$, and is related to the state vector according to:

$$\boldsymbol{y}_t = h_t(\boldsymbol{x}_t) + z_t, \tag{2}$$

where the non-linear function $h_t(\cdot)$ captures instantaneous network dynamics and $z_t \in \mathbb{R}^n$ accounts for measurements noise. Furthermore, we define $\boldsymbol{x}_t^n$ as the state parameters that affect the power flow of subnetwork $n \in \{1, \dots, N\}$. Similarly, we define $\boldsymbol{y}_t^n$ as the measurements taken in subnetwork n and $h_t^n(\cdot)$ as the non-linear function that relates the measurements in subnetwork n to its state parameters,[1] i.e.,

$$\boldsymbol{y}_t^n = h_t^n(\boldsymbol{x}_t^n) + z_t^n, \tag{3}$$

where z_t^n denotes the pertinent measurements noise. Subnetworks could potentially share some consuming, generating, sensing, or controlling modules, and therefore, some state parameters are affected by the power flows of multiple subnetwork. We define $\mathcal{S}_i$ as the set of indices of the subnetworks in which the power flow affects the state parameter $x_t[i]$ for $i \in \{1, \dots, m\}$, i.e.,

$$\mathcal{S}_i \overset{\triangle}{=} \{n \mid x_t[i] \text{ affects power flow of subnetwork } n\}. \tag{4}$$

The ultimate objective is to provide an accurate prediction for the magnitude and geographical location of *future* electric demands over different time horizons. As discussed in detail in [13], such load demands can be determined uniquely by alternatively determining the predictions of the future state parameters $\boldsymbol{x}_t$. Hence, in the rest of the paper is the attention is focused on providing accurate predictions for $\boldsymbol{x}_t$, which in turn can be used to find the desired load forecasting.

Remark 1: We remark that the state *prediction* operation discussed in this paper is different from the objective of state *estimation* introduced by Schweppe and Wildes [19]. Specifically, state estimation is responsible for using the measurements collected from the network in order to recover the variations in the *current* state of the system caused by load fluctuations [19]–[23]. State *prediction*, which is used for the ultimate purpose of load forecasting, on the other hand, leverages the information about the current measurements, states, and the historical data in order to form predictions for the *future* variations of the state parameters. Such predictions for the states can be used to uniquely predict loads at different geographical locations over different time horizons.

[1] In this paper we assume that functions $h_t^n(\cdot)$ can take any arbitrary form that is not necessarily known or specified and could be deterministic, stochastic, or even adversarial to network state.

B. Functional State Prediction

We are interested in predicting the temporal evolution of the network state vectors $\{\boldsymbol{x}_t : t \in \mathbb{N}\}$. Specifically, we consider a dynamic *state* prediction model, which at any given time t uses the current and past states of the network $\{\boldsymbol{x}_1, \dots, \boldsymbol{x}_t\}$, and provides a prediction for the state values at a future instance $t+\tau$, i.e., $\boldsymbol{x}_{t+\tau}$ for $\tau \in \mathbb{N}$. The choice of τ delineates how much in advance we want to perform prediction, and we denote the prediction of $\boldsymbol{x}_{t+\tau}$ available at time t, by $\boldsymbol{s}_{t+\tau}$.

Perfect acquisition of the current and past states through noisy measurements $\{\boldsymbol{y}_1, \dots, \boldsymbol{y}_t\}$ is, however, not feasible and only noisy estimates of the states can be obtained via state estimation. Hence, we define $\hat{\boldsymbol{x}}_t$ and $\hat{\boldsymbol{x}}_t^n$ as the noisy estimates of $\boldsymbol{x}_t$ and $\boldsymbol{x}_t^n$, respectively, available to the predictors. The prediction of $\boldsymbol{x}_{t+\tau}$ is denoted by $\boldsymbol{s}_{t+\tau}$ and is related to the state estimates up to time t via

$$\boldsymbol{s}_{t+\tau} \;=\; f_t(\hat{\boldsymbol{x}}_1, \dots, \hat{\boldsymbol{x}}_t), \tag{5}$$

where function $f_t : \mathbb{C}^{m\times t} \to \mathbb{C}^m$ captures the structure of the predictor at time t. It is noteworthy that in this paper the goal is to design a framework that can be applied in conjunction with any desired predictors. To this end, we do not impose any structure on the predictors f_t.

Due to the complexity and scale of the entire grid, delineating a statistical model that captures the dynamics of the entire grid is prohibitive, which makes the design of the predictor f_t extremely hard, if not impossible. On the other hand, given the limited size and complexity of the individual subnetworks, furnishing statistical models and designing local state predictors for the individual subnetworks is more viable. Motivated by this premise, we consider *local* predictors for the subnetworks and the proposed framework combines these local predictions for obtaining an aggregated prediction for the network state. For this purpose, we denote the predictor of subnetwork n at time t by f_t^n, which maps the current and past local state parameters $\{\hat{\boldsymbol{x}}_1^n, \dots, \hat{\boldsymbol{x}}_t^n\}$ into a prediction for $\boldsymbol{x}_{t+\tau}^n$, denoted by $\boldsymbol{s}_{t+\tau}^n$, i.e.,

$$\boldsymbol{s}_{t+\tau}^n \;=\; f_t^n(\hat{\boldsymbol{x}}_1^n, \dots, \hat{\boldsymbol{x}}_t^n). \tag{6}$$

Functions f_n^t can take arbitrary forms suitable for capturing the local dynamics in different subnetworks.

Due to different sizes and complexities of different subnetworks, the local predictions $\{\boldsymbol{s}_{t+\tau}^1, \dots, \boldsymbol{s}_{t+\tau}^N\}$ provided by the subnetworks have potentially different qualities. Furthermore, the prediction strategies of different subnetworks, captured by functions $\{f_t^n\}_{n=1}^N$, are not necessarily identical across the subnetworks and can be designed based on their local dynamics and computational capabilities. Also, the prediction approach of subnetwork n (i.e., function f_t^n) is not necessarily known to other subnetworks. Our proposed prediction framework, which is discussed in details in Section III, specifies approaches for aggregating these local predictions in order to obtain the accurate aggregated prediction $\boldsymbol{s}_{t+\tau}$.

C. Connection Between State Parameters and Loads

By defining $u_t[i]$ and $v_t[i]$ as the real and imaginary parts of $x_t[i]$, power injections at busbar i are [13]

$$\begin{aligned} P_i \;=& \sum_{i=1}^{M} u_t[i](u_t[j]G_{ij} - v_t[j]B_{ij}) \\ &\qquad + v_t[i](v_t[j]G_{ij} + u_t[j]B_{ij}), \end{aligned} \tag{7}$$

and

$$\begin{aligned} Q_i \;=& \sum_{j=1}^{M} v_t[i](u_t[j]G_{ij} - v_t[j]B_{ij}) \\ &\qquad - u_t[i](v_t[j]G_{ij} + u_t[i]B_{ij}). \end{aligned} \tag{8}$$

Also, flows in the line $i-j$ are [13]

$$\begin{aligned} P_{ij} = u_t[i](&u_t[i]G_{ij} - u_t[j]G_{ij} - v_t[i]B_{ij} + v_t[j]B_{ij}) \\ &+ v_t[i](u_t[i]B_{ij} - u_t[j]B_{ij} + v_t[i]G_{ij} - v_t[j]G_{ij}) \\ &+ G'(u_t^2[i] + v_t^2[i]), \end{aligned} \tag{9}$$

and

$$\begin{aligned} Q_{ij} = v_t[i](&u_t[i]G_{ij} - u_t[j]G_{ij} - v_t[i]B_{ij} + v_t[i]B_{ij}) \\ &- u_t[i](e_iB_{ij} - u_t[j]B_{ij} + v_t[i]G_{ij} - v_t[j]G_{ij}) \\ &- B'(u_t^2[i] + v_t^2[i]), \end{aligned} \tag{10}$$

where $Y_{ij} = G_{ij} + jB_{ij}$ is the (i, j) element of the nodal admittance matrix and $Y'_{ij} = G'_{ij} + jB'_{ij}$ is the line (i, j) charging admittance. Based on the relationships in (7)–(10) the prediction for the state parameter $x_t[i] = u_t[i] + j \cdot v_t[i]$ one can find predictions for the injected and line power values.

III. State Prediction Methodology

The framework developed in this section can be deployed in conjunction with any set of prediction models for its subnetworks. For any set of desirable predictors, abstracted by functions $\{f_t^1, \dots, f_t^N\}$ for different areas, the details of *diversified* predictions of the state parameters shared by multiple areas are provided in Section III-B, the details of how such distinct local predictions are combined are discussed in Section III-C, and the performance analysis (which is valid for any arbitrary choices of the prediction approaches) is provided in Section III-D, where it is shown that without any knowledge of the dynamics in the subnetworks, the proposed learning-based framework can identify the best prediction for each state parameter.

Such inherent diversity in state predictions leads to higher level of prediction quality for the state parameter, which then, by using the relationships in Section II-C, leads to a higher level of accuracy for load forecasting.

A. Prediction Performance Measure

In order to assess the performance of the predictors f_t and $\{f_t^1, \dots, f_t^n\}$ over the entire time horizon $t = 1, 2, 3, \dots$, we define local and aggregated prediction accuracy measures. Specifically, for quantifying the accuracy of $s_t[i]$ as the aggregated prediction for the state parameter $x_t[i]$, we define a *non-negative* prediction cost function

$$C(s_t[i],\ \hat{x}_t[i]), \tag{11}$$

that measures the distance between the prediction for $x_t[i]$ which is provided before taking the measurements $\boldsymbol{y}_t$ and its estimate after taking these measurements. One popular prediction cost function corresponding to the minimum-square error (MSE) criterion is $C(u,\ v) = |u-v|^2$. Given the prediction cost $C(s_t[i],\ \hat{x}_t[i])$, which quantifies the prediction fidelity at time t, we define a *cumulative* prediction cost function, that aggregates the prediction inaccuracies over the interval $t = 1, \dots, T$ for $T \in \mathbb{N}$. Specifically, corresponding to each state parameter $x_t[i]$ we define the *cumulative* prediction cost accumulated up to time T as

$$\mathsf{C}_T[i] \ \overset{\triangle}{=} \ \sum_{t=1}^{T} C(s_t[i],\ \hat{x}_t[i]). \tag{12}$$

Similarly, we also define the instantaneous and cumulative prediction costs for the local prediction on the state parameter $x_t[i]$ as follows. We denote the prediction for $x_t[i]$ provided by subnetwork $i \in \mathcal{S}_i$ by $s_t[i; n]$, for which the prediction cost is $C(s_t[i; n], \hat{x}_t[i])$. By accumulating this cost function over the interval $t \in \{1, \dots, T\}$ we define

$$\mathsf{C}_T[i; n] \overset{\triangle}{=} \sum_{t=1}^{T} C(s_t[i; n],\ \hat{x}_t[i]). \tag{13}$$

B. Measurement Diversity

Different subnetworks follow possibly different stochastic models, have different uncertainties about their underlying models, enjoy different levels of computational capabilities, and based on their underlying models and historical data deploy different prediction models. Different prediction models provide multiple predictions for the state parameters that are shared by more than one subnetwork. Such inherent prediction diversity for a state parameter, if exploited judiciously, can provide a prediction for that state parameter that is more accurate than each of the individual predictions. This observation is the core premise of the proposed prediction framework, which tracks and uses the *best*[2] prediction model for each state parameter. In order to trace the quality of the aggregated prediction provided for each state parameter and compare its performance over the local ones (provided by the subnetworks) we define a *relative* prediction cost function that captures the difference between the prediction cost incurred by the aggregating predictor and the local predictors. For each state parameter $x_t[i]$ and for all subnetworks $n \in \mathcal{S}_i$ we define the relative prediction cost as

$$\Delta L_t[i; n] \ \overset{\triangle}{=} \ C(s_t[i],\ \hat{x}_t[i]) \ - \ C(s_t[i; n],\ \hat{x}_t[i]). \tag{14}$$

We also define the cumulative relative prediction cost up to T

$$L_T[i; n] \ \overset{\triangle}{=} \ \sum_{t=1}^{T} \Delta L_t[i; n] \ \overset{(a)}{=} \ \mathsf{C}_T[i] \ - \ \mathsf{C}_T[i; n], \tag{15}$$

where the equality in (a) holds according to the definitions of $\mathsf{C}_T[i]$ and $\mathsf{C}_T[i; n]$ in (11) and (12). Larger (and positive) values of $L_T[i; n]$ indicate that the cost of the aggregated prediction for the state parameter $x_t[i]$ up to time T (i.e., $\mathsf{C}_T[i]$) is higher than that of the prediction of subnetwork n for the same state parameter up to time T (i.e., $\mathsf{C}_T[i; n]$), or in other words, the local predictor is more accurate than the aggregating predictor. Conversely, smaller (and negative) values of $L_T[i; n]$ indicate that the aggregating predictor outperforms the local predictor at subnetwork n.

C. Aggregated Prediction

In order to focus on the core components of the proposed framework we start with a model in which we assume that the state estimates are reliable (i.e., $x_t[i]$ is close enough to $\hat{x}_t[i]$) so that the cost function $C(s_t[i], \hat{x}_t[i])$ is a reliable prediction measure. The necessary modifications to the framework for incorporating unreliable state estimates are relegated to Section III-E.

The core step of the prediction procedure is the mechanism for aggregating the real-time information and local predictions for each state parameter provided by different subnetworks in order to obtain the aggregated prediction for the state of the network. In this mechanism, upon obtaining the local predictions, corresponding to each state parameter $x_t[i]$, all the subnetworks that have predicted this state parameter (i.e., the subnetworks in $\mathcal{S}_i$) share their local predictions for $x_t[i]$ (i.e., $\{s_t[i; n]\}_{n \in \mathcal{S}_i}$). Given all these local predictions, the aggregated prediction for $x_t[i]$, denoted by $s_t[i]$, is computed as a weighted average of the local predictions according to

$$s_t[i] \ = \ \Big(\sum_{n \in \mathcal{S}_i} \alpha_t[i; n]\ s_t[i; n]\Big)\Big(\sum_{n \in \mathcal{S}_i} \alpha_t[i; n]\Big)^{-1}, \tag{16}$$

where $\{\alpha_t[i; n]\}$ are the weights assigned to the local prediction of $x_t[i]$, for which we have $\alpha_t[i; n] = 0$ when $n \notin \mathcal{S}_i$. The side information about the past prediction costs incurred by different predictors can be exploited to dynamically adjust these weighting factors. Intuitively, at time t, if $L^n_{t-1}[i]$, which is the *relative* cumulative cost for predictor n up to time $t-1$, is large, it means that prediction of subnetwork n is performing better than the aggregating predictor and we need to assign a larger weighting factor $\alpha_t[i; n]$ to this prediction. This approach, in other words, results in increasing the weights of the predictors whose cumulative cost $\mathsf{C}_{t-1}[i; n]$ is small. Therefore, the weights are increasing functions of the cumulative relative cost. For this purpose, for $n \in \mathcal{S}_i$ we write the weight $\alpha_t[i; n]$ as the *derivative* of a non-negative, convex, and increasing function $\phi : \mathbb{R} \to \mathbb{R}$ of $L_{t-1}[i; n]$, i.e.,

$$\alpha_t[i; n] \ = \ \phi'(L_{t-1}[i; n]), \qquad \forall n \in \mathcal{S}_i. \tag{17}$$

Corresponding to each state parameter $x_t[i]$ we define the *relative* cost vector $\boldsymbol{r}^i_t \overset{\triangle}{=} [r_t[i; 1], \dots, r_t[i; N]] \in \mathbb{R}^N$ as the vector of the predictions provided for $x_t[i]$ by different subnetworks such that

$$r_t[i; n] \ \overset{\triangle}{=} \ \begin{cases} L_t[i; n] & \text{if} \ \ n \in \mathcal{S}_i \\ \\ 0 & \text{if} \ \ n \notin \mathcal{S}_i \end{cases}, \tag{18}$$

[2] The best prediction model is the one that yields the best prediction performance after a transient initial duration.

and the corresponding *cumulative* cost vector $\boldsymbol{R}^i_T \in \mathbb{R}^N$ as

$$\boldsymbol{R}^i_T \overset{\triangle}{=} \sum_{t=1}^{T} \boldsymbol{r}^i_t, \tag{19}$$

where $\boldsymbol{R}^i_t \overset{\triangle}{=} [R^i_t[1], \dots, R^i_t[N]]$. Also, we introduce the *potential* function $\Phi : \mathbb{R}^N \to \mathbb{R}$ as

$$\Phi(\boldsymbol{u}) \overset{\triangle}{=} \psi\Big(\sum_{n=1}^{N} \phi(u_n)\Big), \tag{20}$$

where $\psi : \mathbb{R} \to \mathbb{R}$ is any non-negative, strictly increasing, concave, and twice differentiable auxiliary function. Based on this definition and (17) we find that for each state parameter $i \in \{1, \dots, m\}$ and for all $n \in \mathcal{S}_i$

$$\alpha_t[i; n] = \nabla\Phi(\boldsymbol{R}^i_{t-1})_n = \frac{\partial\Phi(\boldsymbol{R}^i_{t-1})}{\partial L_{t-1}[i; n]}. \tag{21}$$

In the next step, we provide guarantees for the performance of the weighting average predictors of the form in (16) and the weighting coefficients specified in (21).

D. Prediction Performance

We evaluate the performance of the predictor, which is the performance after some initial transient period, for the following specific potential function.

$$\Phi(\boldsymbol{u}) = \frac{1}{\eta} \sum_{n=1}^{N} \exp(\eta\, u_n), \tag{22}$$

where η is a positive real number. For this given potential function the weights assigned to the predictors are given by

$$\alpha_t[i; n] = \nabla\Phi(\boldsymbol{R}^i_{t-1})_n = \frac{\exp(\eta\, R^i_{t-1}[n])}{\sum_{j=1}^{N} \exp(\eta\, R^i_{t-1}[j])}. \tag{23}$$

For the above given set of weighting factors, the following theorem establishes the asymptotic performance of the aggregating predictor.

Theorem 1: When the prediction cost function $C(\cdot, \cdot)$ is convex in its first argument, and the potential function is $\Phi(\boldsymbol{u}) = \frac{1}{\eta}\sum_{n=1}^{N} \exp(\eta u_n)$, the relative cumulative prediction cost of the aggregating predictor with respect to the most reliable predictor (with the smallest cumulative cost) satisfies $\max_{n\in\mathcal{S}_i} L_T[i; n] = o(T)$, or by invoking (15), it equivalently satisfies[3]

$$\lim_{T\to\infty} \frac{1}{T}\Big(\mathsf{C}_T[i] - \min_{n\in\mathcal{S}_i} \mathsf{C}_T[i; n]\Big) = 0. \tag{24}$$

The theorem above demonstrates that for each state parameter $x_t[i]$, for which the subnetworks in $\mathcal{S}_i$ provide predictions, the aggregating predictor $s_t[i]$ has the same long-term average performance (captured by $\mathsf{C}_T[i]$) as the best prediction provided by the subnetworks $\mathcal{S}_i$ (captured by $\min_{i\in\mathcal{S}_i} \mathsf{C}_T[i; n]$). This guarantee (i.e., identifying the best local prediction) holds in spite of the fact that the aggregating predictor has no information about the statistical model, historic data, and structure of the local predictors.

Remark 2: Throughout the analysis we have assumed that functions $h^n_t(\cdot)$ (which relate the measurements to the state parameters) can take any arbitrary form that is not necessarily known or specified, and could be deterministic, stochastic, or even *adversarial* to network state. Based on this, any aggregation method that does not explicitly incorporate the values of all the state predictions can potentially fail.

Remark 3: Theorem 1 establishes that the upper bound on the prediction cost increases as the number of areas N increases. Hence, networks with a larger number of subnetworks are expected to experience reduced quality of prediction.

E. Aggregated Prediction via Unreliable Data

The major challenge in the presence of noisy estimates and using them as the benchmark for evaluating prediction quality is that state estimates are prone to be contaminated with error levels large enough to affect the performance of aggregating predictor. This necessitates implementing a measure of feedback mechanism that constantly monitors the accuracy level of the state estimates and avoids using them when they are deemed to be erroneous beyond a tolerable level. For this purpose, by recalling the relationship between the state parameters and the measurements of subnetwork n given in (3), we define the residual vector of measurements $\boldsymbol{e}^n_t$ as

$$\boldsymbol{e}^n_t \overset{\triangle}{=} \boldsymbol{y}^n_t - h^n_t(\hat{\boldsymbol{x}}^n_t). \tag{25}$$

Hence, the average residual value in subnetwork n at time t is

$$E^n_t \overset{\triangle}{=} \frac{1}{\dim(\boldsymbol{e}^n_t)} \cdot \|\boldsymbol{e}^n_t\|_2^2. \tag{26}$$

Consequently, the average residual error of state parameter $\hat{x}_t[i]$ at time t is

$$E_t[i] \overset{\triangle}{=} \frac{1}{|\mathcal{S}_i|} \cdot \sum_{n\in\mathcal{S}_i} E^n_t. \tag{27}$$

The state estimate $\hat{x}_t[i]$ is deemed to be accurate enough if $E_t[i] \leq \varepsilon$, where ε is a predefined threshold value. In order to incorporate this admissibility condition for the state estimates, the weighting coefficients defined in (21) are modified as follows. For each $i \in \{1, \dots, m\}$ and $n \in \mathcal{S}_i$ we set

$$\alpha_t[i; n] \overset{\triangle}{=} \begin{cases} \frac{\partial\Phi(\boldsymbol{R}^i_{t-1})}{\partial L_{t-1}[i;n]} & \text{if } E_t[i] \leq \varepsilon \\ \alpha_{t-1}[i; n] & \text{if } E_t[i] > \varepsilon \end{cases}. \tag{28}$$

Hence, the weighting coefficients are updated only when the state estimates are accurate enough, otherwise, the previous weighting factors are retained.

The appropriate selection of ε hinges on balancing the interplay between how much prediction inaccuracy a network operator can tolerate and how fast the algorithms converge. The selection of the desired balance these figures of merit relate power engineering of the problem on one hand, and the statistical learning machinery on the other hand. Specifically,

[3] Function $f(x)$ is said to be asymptotically dominated by $g(x)$ and denoted by $f(x) = o(g(x))$ when $\lim_{x\to\infty} \frac{f(x)}{g(x)} = 0$.

depending on the operational and economic aspects of grid such as the energy markets, resource type (e.g., intermittent versus fossil-based), and time horizon of prediction, the network operator decides what quality of prediction it desires. Clearly, aiming for higher quality of prediction, generally (but not necessarily always) requires setting ε closer to small ranges such that any prediction with high level of disturbance is filtered out in order not to contribute to prediction approach. On the other hand, ε at the same time acts as a hyperparameter (or in a non-Bayesian setting a regular parameter), selection of which affects the quality of learning. Hence, the network can dynamically decide about the relevant and desirable level of prediction quality, based on which it can decide upon the right range of ε.

F. Error Propagation

Since the aggregation method relies on the learning accumulated over time, operations at each time instance hinge on the input from the operations in the past. Therefore, operations can be vulnerable to error propagation. Next we discuss that depending on the nature of the error, the propagation issue manifests in different forms and is coped with differently.

Specifically, the proposed aggregation framework in the previous subsections establishes that the aggregated prediction for each state parameter enjoys a prediction quality that is similar to that of the best local predictor for that specific state parameter. In this sense, the proposed aggregation identifies and tracks the best predictor over time. This implies that the erroneous predictions during the transient period will gradually diminish over time and such initial errors do not propagate. Therefore, on an aggregated level there exists no error propagation. In order to formalize this we define the mean square error (MSE) term between the actual state parameter $x_t[i]$ and its prediction $s_t[i;n]$ provided by the subnetwork n as

$$\mathsf{mse}_t(i;n) \overset{\triangle}{=} \|x_t[i] - s_t[i;n]\|^2. \tag{29}$$

By taking all such local predictions provided for the state parameter $x_t[i]$, we denote the MSE corresponding the best prediction provided by all subnetworks by

$$\mathsf{mse}^*_t(i) \overset{\triangle}{=} \min_{n\in\mathcal{S}_i} \mathsf{mse}_t(i;n), \tag{30}$$

and denote the MSE of the prediction provided by the proposed framework, i.e., $s_t[i]$, by

$$\mathsf{mse}_t(i) \overset{\triangle}{=} \|x_t[i] - s_t[i]\|^2. \tag{31}$$

The following theorem establishes that the average error between the prediction provided and that of the best local prediction diminishes over time, implying that the effect of prediction error does not propagate through the process.

Theorem 2: The mean square error of each state predictor approaches that of the most reliable predictor, i.e.,

$$\lim_{T\to\infty} \frac{1}{T}\big(\mathsf{mse}_t(i) - \mathsf{mse}^*_t(i)\big) \;=\; 0. \tag{32}$$

IV. Case Study: Support Vector Regression

As discussed earlier, the framework developed in Section III can be deployed in conjunction with any desired set of prediction models for its subnetworks, selection of which is widely studied in different contexts (see [24]). For any set of desirable predictors, abstracted by functions $\{f^1_t,\dots,f^N_t\}$ for different areas, the details for diversified prediction of the state parameters shared by multiple areas were discussed in Section III-B, and how such local predictions are combined to form a better prediction is discussed in Section III-C, with performance analysis (which is valid for all choices of the predictors) provided in Section III-D.

In this section we select specific structures for state predictors f_t and $\{f^1_t,\dots,f^N_t\}$ defined in (5) and (6), respectively, based on support machine vector approaches and then discuss how to deploy them in the context of the framework developed in Section III.

A. Local Prediction

Inspired by support vector machine (SVM), support vector regression (SVR) is a supervised learning algorithm that analyzes past data in order to predict a model for future variations in data. Unlike SVM algorithms, SVR algorithms are only interested in a *subset* of past data by ignoring the data segments that lie beyond certain margins. Specifically, the local predictor of area n analyzes the state parameters in the past ℓ time instances in that area and uses them to form a prediction for its relevant state parameters. In order to formalize this, corresponding to the last ℓ of the state parameter $\hat{x}_t[i]$ leading to time t we define

$$\boldsymbol{v}_t[i] \;\overset{\triangle}{=}\; \big[\hat{x}_{t-\ell+1}[i],\dots,\hat{x}_t[i]\big]. \tag{33}$$

By using SVR, our objective is to find the prediction functions f^n_t, as defined in (6) such that the prediction of $x_{t+\tau}[i]$ provided by area n (i.e., $s_{t+\tau}[i;n]$) is related to the past state parameters, captured by $\boldsymbol{v}_t[i]$, via the following model

$$s_{t+\tau}[i;n] \;\overset{\triangle}{=}\; \boldsymbol{w}_t[i;n]\;\cdot\;\boldsymbol{v}^T_t[i] \;+\; b[i;n], \tag{34}$$

where $\boldsymbol{w}_t[i;n]$ is a weighting vector, and $b[i;n]\in\mathbb{R}$ is a constant. Both $\boldsymbol{w}_t[i;n]$ and $b[i;n]$ are parameters to be computed such that:

1) All the previous predictions $s_j[i;n]$ have at most a controlled level of deviation λ from the true estimates $\hat{x}_j[i]$ for all previous time instances $j=t-\ell+1,\dots,t$.

2) The mapping from $\boldsymbol{v}^n_t$ to $s_{t+\tau}[i;n]$ is as flat as possible.

These two objectives can be achieved simultaneously by solving the following optimization problem [25].

$$\begin{array}{ll} \underset{\boldsymbol{w}_t[i;n],b[i;n]}{\text{minimize}} & \frac{1}{2}\|\boldsymbol{w}_t[i;n]\|^2 \\ \text{subject to} & x_j[i;n] \;-\; s_{j+\tau}[i;n] \;\leq\; \lambda \\ & s_{j+\tau}[i;n] \;-\; x_j[i;n] \;\leq\; \lambda \end{array} \tag{35}$$

This is a convex optimization problem, which has a unique globally optimal solution when it exists. Small choices of λ, however, can make the problem infeasible, in which case as a remedy two slack variables ξ_j and ξ^*_j are introduced so that

$(\hat{x}_j[i;n] - s_{j+\tau}[i;n])$ tolerates deviations beyond the boundaries of the error-insensitive zone (controlled by λ) in favor of rendering the problem feasible. Hence, the problem in (35) is modified to:

$$\begin{aligned} \underset{\boldsymbol{w}_t[i;n],b[i;n]}{\text{minimize}} \quad & \frac{1}{2}\|\boldsymbol{w}_t[i;n]\|^2 + C\sum_{j=1}^{t}(\xi_j + \xi_j^*) \\ \text{subject to} \quad & x_j[i;n] - s_{j+\tau}[i;n] \leq \lambda + \xi_j \\ & s_{j+\tau}[i;n] - x_j[i;n] \leq \lambda + \xi_j^* \\ & \xi_j, \xi_j^* \geq 0. \end{aligned} \tag{36}$$

Solving the optimization problem in (36) in its dual has lower computational complexity [26] and [27]. Solving the dual form shows that $s_{t+\tau}[i;n]$ can be cast as an affine combination of the input patterns $\boldsymbol{v}_j^n$ [25]. Specifically, we have

$$s_{t+\tau}[i;n] = \sum_{j=t-\ell+1}^{t} \beta_j[i;n]\,\hat{x}_j[i;n]\hat{x}_t[i;n] + b[i;n], \tag{37}$$

where the constants $\{\beta_j[i;n]\}$ are obtained from solving the optimization problem. For the purpose of achieving higher accuracy, the SVR algorithm can be extended to take a non-linear form by modifying (37) to:

$$s_{t+\tau}[i;n] = \sum_{j=t-\ell+1}^{t} \beta_j[i;n]\Psi_n(\hat{x}_j[i;n], \hat{x}_t[i;n]) + b[i;n], \tag{38}$$

and by introducing the non-linear kernel functions $\Psi_n(\hat{x}_j[i;n], \hat{x}_t[i;n])$. Popular choices of the kernel functions include those that can be decomposed as

$$\Psi_n(\hat{x}_i[i], \hat{x}_t[i;n]) = \phi_n(\hat{x}_j[i;n]) \cdot \phi_n(\hat{x}_t[i;n]), \tag{39}$$

for some non-linear function $\phi_n(\cdot)$. The discussions on the choices of these non-linear functions for the load forecasting problem at hand is provided in the next subsection.

Finally, we remark that the SVR algorithm in its original form, after providing the prediction and then observing the true value of the predicted parameter, adds them to its training sequence. Such *batch*-based implementation of SVR is computationally inefficient for the online setting we are interested in as the sequence is updated constantly. To overcome this disadvantage, online recursive training methods are introduced [28] and further developed in [29]. We use these algorithms in the implementation of SVR in Section V on simulation results.

B. Predictions Diversification

The core part of our proposed prediction framework is that the individual areas are allowed to adopt a prediction model that suits best their local power flow dynamics. Such diversified local prediction methods allow for reaching a aggregated prediction that can aggregate the local ones in a manner that invests more weight on the more accurate predictors. For implementing the SVR algorithm in area $n \in \{1, \ldots, N\}$, we allow different subnetworks select different kernel functions as defined in (39). Two commonly used kernels are the polynomial kernel and the radial basis function kernel. In this paper, we adopt the polynomial kernel defined as

$$\Psi_n(a,b) \overset{\triangle}{=} (ab+d)^{p_n}, \tag{40}$$

where p_n is degree of the polynomial kernel at area n, and $d \geq 0$ is a constant trading off the influence of higher-order versus lower-order terms in the polynomial. The effectiveness of SVR in each area depends on the selection of the kernel order p_n. It is noteworthy that the performance of SVR predictor is *very sensitive* to the choice of its parameters $\{p_n, \lambda, C\}$ and selecting the appropriate kernel order impacts the prediction quality significantly. We will assume that the polynomial order p_n is independently selected by each area n. This will serve as a source of diversification of prediction qualities across the areas. The prediction algorithm described in Section III is then used to combine the local predictions and assign appropriate weights to different local predictors.

TABLE I
PREDICTION ALGORITHM

```
**Initialize**: $\hat{\boldsymbol{x}}_0$, $\alpha_0[i;n]$, $\tau$
**For** each time $t = 1, 2, 3, \ldots$
  **For** each subnetwork $n = 1, 2, \ldots, N$
    Sub networks update SVR function $\forall n \in \mathcal{S}_i$ using [29]
    Subnetwork $n$ calculates $E_t[i]$ using (25)-(27)
    **If** $E_t[i] \leq \varepsilon$, then
      Update $\alpha_t[i;n]$, $\forall n \in \mathcal{S}_i$ using (42)
    **else**
      $\alpha_t[i;n] = \alpha_{t-1}[i;n]$
    **End**
    Find local predictions $\boldsymbol{s}_{t+\tau}^n$ using (37)
    Obtain the aggregated prediction $s_{t+\tau}$ using (41)
    Calculate predicted total load using $\boldsymbol{s}_{t+\tau}$ using [13]
  **End For**
**End For**
```

C. Aggregated Prediction

As discussed earlier, the states will be predicted by potentially more than one subnetwork. Once the subnetworks provide their local predictions at time t using (37), the subnetworks in $\mathcal{S}_i$ share their predictions for $x_{t+\tau}[i]$. These local predictions are then aggregated in a linear fashion through

$$s_{t+\tau}[i] = \Big(\sum_{n\in\mathcal{S}_i} \alpha_t[i;n]\, s_{t+\tau}[i;n]\Big)\Big(\sum_{n\in\mathcal{S}_i} \alpha_t[i;n]\Big)^{-1} \tag{41}$$

to provide the aggregated prediction for $x_{t+\tau}[i]$. By recalling the characterization of $\alpha_t[i;n]$ in (23) and definition of $\boldsymbol{R}_T^i$ in (19) we find that

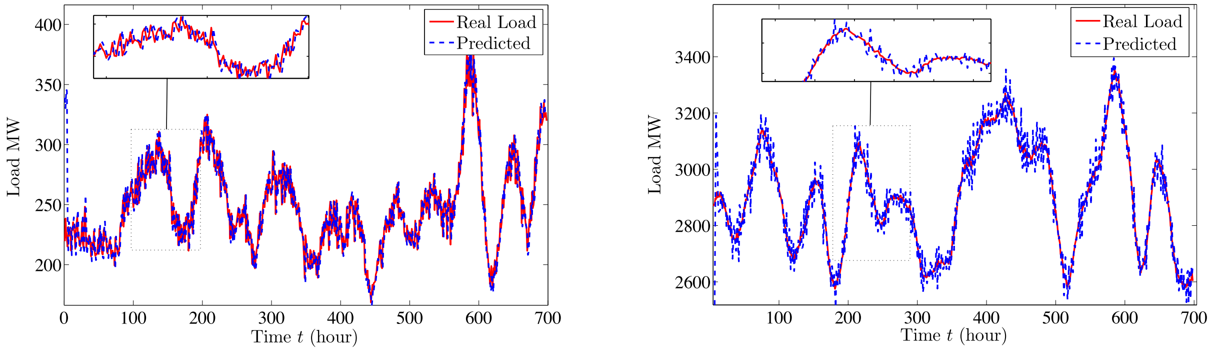
$$\begin{aligned} \alpha_t[i;n] &= \frac{\exp(\eta\, R_{t-1}^i[n])}{\sum_{j=1}^{N}\exp(\eta\, R_{t-1}^i[j])} \\ &= \frac{\alpha_{t-1}[i;n]\, \exp(-\eta\, C(s_t[i;n],\ \hat{x}_t[i]))}{\sum_{i\in\mathcal{S}_i}\alpha_{t-1}[i;n]\, \exp(-\eta\, C(s_t[i;n],\ \hat{x}_t[i]))}, \end{aligned} \tag{42}$$

which shows that the weighting coefficient $\alpha_t[i;n]$ can be found recursively as a function of $\alpha_{t-1}[i;n]$. Also for the cost function we have set $C(u,\ v) = \|u - v\|^2$.

Table I summarizes the technique presented in this section. It is noteworthy that updating the weights for the predictors is stopped when the cumulative regret defined in (12) diminishes.

V. SIMULATION RESULTS

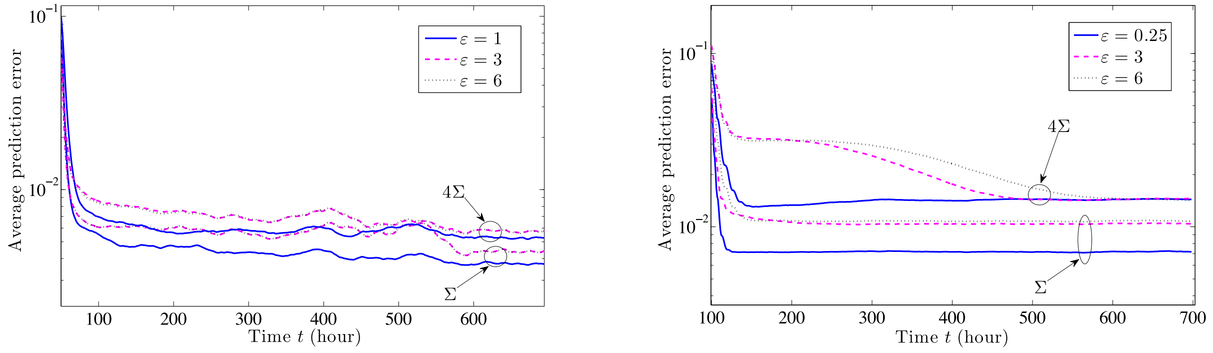
In the simulations, we consider the IEEE 14-bus ($m = 14$ and $N = 4$) and the IEEE 118-bus ($m = 118$ and $N = 8$)



(a) IEEE-14 bus system

(b) IEEE-118 bus system

Fig. 1. Total load forecasting over time (hourly).



(a) IEEE-14 bus system

(b) IEEE-118 bus system

Fig. 2. Mean absolute percent error (MAPE) in prediction for different ε values and noise levels.

systems, where in each system, the subnetworks are interconnected through tie-lines. The time unit in all the time series (i.e., $\boldsymbol{x}_t, \boldsymbol{y}_t, \boldsymbol{s}_t, f_t$) is one hour, that is, the progression from time instance t to $t+1$ is one hour. Hence, load forecasting can be provided on hourly or multi-hourly basis. Throughout the analysis and simulations it is assumed that all the predicted parameters obtained at each cycles by each subnetwork will be shared with the rest of the subnetworks. For simulations we will consider the DC power model in which the grids state includes phase angles of all the buses, which we obtain by power flow simulations using MATPOWER package. The measurement vector of each local area is then constructed such that it consists of power injections at buses, power flows at all branches, and lines currents.

At each subnetwork $n \in \{1, \dots, N\}$, SVR predictor is initialized for each state $x_t[i; n]$ through defining the model order p_n, where p_n is a non-negative integer representing the order of the polynomial kernel defined in (40) (which as mentioned earlier, has a significant impact on the prediction quality). In order to control the computational complexity, an evolutionary algorithm is employed to determine the optimized values. We assume that the measurements in all N areas are contaminated with Gaussian noise with variance values $\Sigma \triangleq [\sigma_1^2, \dots, \sigma_N^2]$, where σ_n^2 is variance of noise observed in area n. Different observation noise levels imply that the measurements and, subsequently, the predictions in different subnetworks can have different accuracy levels.

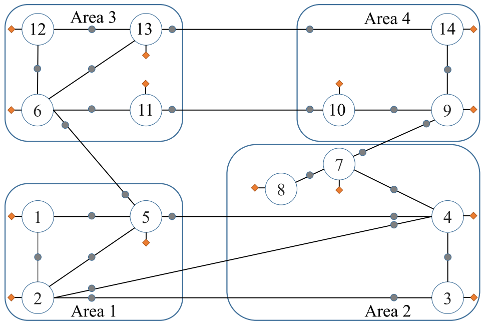


Fig. 3. IEEE 14 bus system.

The weighting factors are initialized by setting $\alpha_0[i; n] = 1$ for all $n \in \{1, \dots, N\}$ and $i \in \mathcal{S}_i$. The simulations are performed for a duration of $T = 700$ hours and the prediction performance is assessed for each state parameter by comparing the predicted states values to the real states of the system through the following relative error.

$$\lambda_t[i] \triangleq \left| \frac{s_t[i] - x_t[i]}{x_t[i]} \right|. \tag{43}$$

Averaging over all state parameters to compute the mean absolute percent error (MAPE), i.e., $\bar{\lambda}_t = \frac{1}{m} \sum_{i=1}^{m} \lambda_t[i]$, provides a network-wide prediction measure. Similarly, by invoking the power flow models we can find the predictions for the loads, which can be compared with the actual loads in order to obtain a load forecasting quality measure. The results in

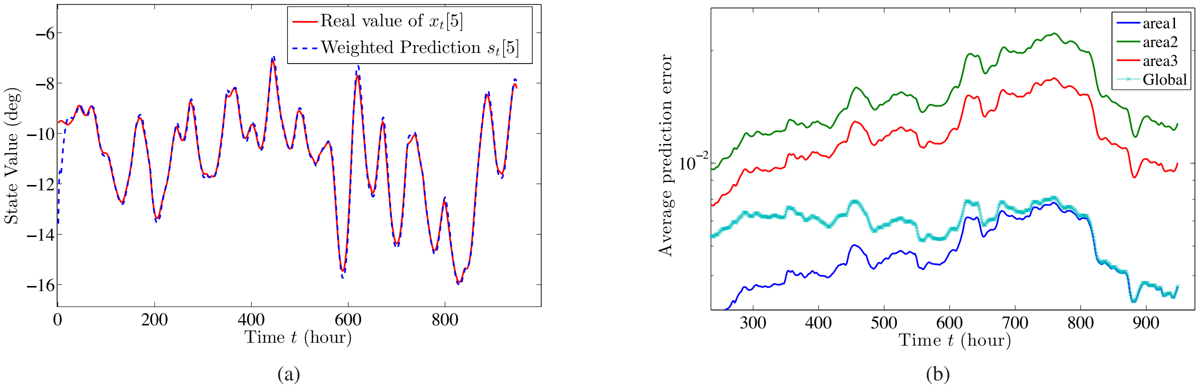


Fig. 4. (a) Moving average MAPE for state $x_t[5]$ in IEEE-14 bus system value prediction over time. (b) Average prediction error of $x_t[5]$ in IEEE-14 bus system evaluated for three different areas and compared to aggregated prediction.

figures 1 and 2 are obtained through Monte Carlo simulation averaged over 500 simulation.

Figures 1(a) and 1(b) depict the fluctuations of the actual load values and the predictions in the IEEE-14 and IEEE-118 bus systems, respectively. In both systems it is observed that except for a short initial learning period, the predictor is closely tracking the true load fluctuations. It is noteworthy that even when the fluctuations are sharp, the predictor is agile in responding to such rapid changes and tracks the actual load closely. One implication of this observation is that the proposed technique is effective for short-term load forecastings in which the role of network dynamics on predictions is more than the historical data.

In order to quantify how closely the predictor is tracking the actual load, figures 2(a) and 2(b) depict the average load forecasting error over time for different values of ε (which controls when the prediction coefficients should be updated based on the estimation accuracy) and for two different values of noise level (Σ and 4Σ). Changing the values of ε has a two-fold effect striking a tradeoff between the quality of prediction on one hand, and the speed of convergence on the other hand. More specifically, it is observed that values of ε, e.g., $\varepsilon = 6$, increase the chance that the weights $\alpha_t[i;n]$ are updated at each cycle, while smaller choices, e.g., $\varepsilon = 0.25$, impose more stringent conditions for updating these weights so that they are updated only when the residual errors are small. Hence, as ε increases the weights are updated more frequently (but they are less accurate) and the algorithm converges faster to a solution, while when ε decreases the weights are updated less frequently (but they are more accurate) and the algorithm converges slower, but will converge to a better solution. It is also observed that lower noise level leads to faster convergence. The reason is that at higher noise levels, identifying the best predictors by the aggregating predictor becomes harder due to the lower quality of measurements.

Figures 4(a) and 4(b) demonstrate the core machinery for concurrent model selection and load forecasting. Since our prediction approach, as an intermediate step, first predicts the state parameters, these figures are provided to illustrate the effectiveness of the prediction algorithm in this intermediate step in predicting the state parameters and the mechanism for selecting the predictor for each state parameter. Specifically, in these figures the temporal variations of the true and predicted values of the state parameters $x_t[5]$ in the IEEE-14 bus system are illustrated. According to the IEEE-14 bus model in Fig. 3 this state parameter can be predicted by areas $\mathcal{S}_5 = \{1, 2, 3\}$. Therefore, Figure 4(a) is provided to compare the real value of the state parameter with its aggregated prediction $s_t[5]$ and Figure 4(b) compares the performance of the aggregating predictor with the local ones. Similar to the behavior observed for load fluctuations, in Figure 4(a) it is observed that except for the initial learning period, the gap between the aggregated prediction and the true state value reduces very quickly. Figure 4(b) aims to illustrate the discrepancy among the predictions provided by local predictions as well as their relevance to the aggregated predictions. For this purpose, the local predictions of parameter ($x_t[5]$) provided by areas $\mathcal{S}_5 = \{1, 2, 3\}$ are compared with the aggregated prediction. As observed, different areas are providing different prediction by using different prediction models that match best their local dynamics and exhibit varying accuracy levels, which alludes to the inherent diversity among the predictions provided by different subnetworks. Such diversity provides the aggregating predictor with the freedom to identify and track the best combination of them. There are two main observations from this plot. First, the aggregating predictor uniformly outperforms all the local predictors by having lower predictor error, and secondly, the aggregating predictor ultimately identifies and tracks the best local predictor. It is noteworthy that the time required for identifying the best local predictor is often much shorter than the results shown in Figure 4(b). In this figure we have selected some of the simulation results for specific settings with slow convergence in order to highlight the distinction between different local predictors and the aggregated one.

VI. Conclusion

In this paper, a learning-based load framework is proposed, which aims to 1) exploit the inherent measurement redundancy and diversity in the measurements accumulated, and 2) uses these structures in conjunction with any desired prediction method in order to improve their prediction quality. The central idea in this framework is to obtain predictions for the state of the network, which in turn, uniquely determine predictions for the loads. Such an indirect approach is especially beneficial in multi-area networks in which some state parameters can be predicted independently in different areas, and possibly based on different prediction models. Such diversified set of predictions for each state parameter allows to combine different predictions appropriately in order to obtain an overall more accurate aggregated prediction for each state parameter. The performance of the proposed approach for combining different local predictors is studied analytically and also via simulations in the standard IEEE-14 and IEEE-118 bus systems.

References

[1] Y. Goude, R. Nedellec, and N. Kong, "Local short and middle term electricity load forecasting with semi-parametric additive models," *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 440–446, Jan. 2014.

[2] T. Hong, J. Wilson, and J. Xie, "Long term probabilistic load forecasting and normalization with hourly information," *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 456–462, Jan. 2014.

[3] J. D. Black and W. L. W. Henson, "Hierarchical load hindcasting using reanalysis weather," *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 447–455, Jan. 2014.

[4] M. Chaouch, "Clustering-based improvement of nonparameteric functional time series forecasting: Application to intra-day household-level load curves," *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 411–419, Jan. 2014.

[5] H.-T. Yang, C.-M. Huang, and C.-L. Huang, "Identification of ARMAX model for short term load forecasting: An evolutionary programming approach," *IEEE Trans. Power Syst.*, vol. 11, no. 1, pp. 403–408, Feb. 1996.

[6] S.-J. Huang and K.-R. Shih, "Short-term load forecasting via ARMA model identification including non-Gaussian process considerations," *IEEE Trans. Power Syst.*, vol. 18, no. 2, pp. 673–679, May 2003.

[7] J. W. Taylor, "Short-term load forecasting with exponentially weighted methods," *IEEE Trans. Power Syst.*, vol. 27, no. 1, pp. 458–464, Feb. 2012.

[8] I. Moghram and S. Rahman, "Analysis and evaluation of five short-term load forecasting techniques," *IEEE Trans. Power Syst.*, vol. 4, no. 4, pp. 1484–1491, Nov. 1989.

[9] L. Ghelardoni, A. Ghio, and D. Anguita, "Energy load forecasting using empirical mode decomposition and support vector regression," *IEEE Trans. Smart Grid*, vol. 4, no. 1, pp. 549–556, Mar. 2013.

[10] R. Campo and P. Ruiz, "Adaptive weather-sensitive short term load forecast," *IEEE Trans. Power App. Syst.*, vol. 2, no. 3, pp. 592–598, Aug. 1987.

[11] G. A. N. Mbamalu and M. E. El-Hawary, "Load forecasting via suboptimal seasonal autoregressive models and iteratively reweighted least squares estimation," *IEEE Trans. Power Syst.*, vol. 8, no. 1, pp. 343–348, Feb. 1993.

[12] J. Kwac, J. Flora, and R. Rajagopal, "Household energy consumption segmentation using hourly data," *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 420–430, Jan. 2014.

[13] A. K. Sinha and J. K. Mondal, "Dynamic state estimator using ANN based bus load prediction," *IEEE Trans. Power Syst.*, vol. 14, no. 4, pp. 1219–1225, Nov. 1999.

[14] B.-J. Chen, M.-W. Chang, and C.-J. Lin, "Load forecasting using support vector machines: A study on EUNITE competition 2001," *IEEE Trans. Power Syst.*, vol. 19, no. 4, pp. 1821–1830, Nov. 2004.

[15] V. H. Hinojosa and A. Hoese, "Short-term load forecasting using fuzzy inductive reasoning and evolutionary algorithms," *IEEE Trans. Power App. Syst.*, vol. 25, no. 1, pp. 565–574, Feb. 2010.

[16] N. Amjady, "Short-term bus load forecasting of power systems by a new hybrid method," *IEEE Trans. Power Syst.*, vol. 22, no. 1, pp. 333–341, Feb. 2007.

[17] N. Cesa-Bianchi and G. Lugosi, *Prediction, Learning, and Games*. Cambridge, U.K.: Cambridge Univ. Press, Mar. 2006.

[18] M. Devaine, P. Gaillard, Y. Goude, and G. Stoltz, "Forecasting electricity consumption by aggregating specialized experts—A review of the sequential aggregation of specialized experts, with an application to Slovakian and French country-wide one-day-ahead (half-)hourly predictions," *Mach. Learn.*, vol. 90, no. 2, pp. 231–260, Feb. 2013.

[19] F. C. Schweppe and J. Wildes, "Power system static-state estimation, part I: Exact model," *IEEE Trans. Power App. Syst.*, vol. PAS-89, no. 1, pp. 120–135, Jan. 1970.

[20] D. M. Falcao, F. F. Wu, and L. Murphy, "Parallel and distributed state estimation," *IEEE Trans. Power Syst.*, vol. 10, no. 2, pp. 724–730, May 1995.

[21] A. Monticelli, "Electric power system state estimation," *Proc. IEEE*, vol. 88, no. 2, pp. 262–282, Feb. 2000.

[22] R. D. Masiello and F. C. Schweppe, "A tracking static state estimator," *IEEE Trans. Power App. Syst.*, vol. PAS-90, no. 3, pp. 1025–1033, May 1971.

[23] K.-R. Shih and S.-J. Huang, "Application of a robust algorithm for dynamic state estimation of a power system," *IEEE Trans. Power Syst.*, vol. 17, no. 1, pp. 141–147, Feb. 2002.

[24] J. O. Bergera and L. R. Pericchi, "The intrinsic Bayes factor for model selection and prediction," *J. Amer. Stat. Assoc.*, vol. 91, no. 433, pp. 109–122, 1996.

[25] D. Basak, S. Pal, and D. C. Patranabis, "Support vector regression," *Neural Inf. Process. Lett. Rev.*, vol. 11, no. 10, pp. 203–224, Oct. 2007.

[26] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Stat. Comput.*, vol. 14, no. 3, pp. 199–222, Aug. 2004.

[27] H. Drucker, C. J. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," in *Proc. Adv. Neural Inf. Process. Syst.*, Denver, CO, USA, Dec. 1996, pp. 155–161.

[28] G. Cauwenberghs and T. Poggio, "Incremental and decremental support vector machine learning," in *Proc. Adv. Neural Inf. Process. Syst.*, Denver, CO, USA, Nov. 2000, pp. 409–415.

[29] J. Ma, J. Theiler, and S. Perkins, "Accurate on-line support vector regression," *Neural Comput.*, vol. 15, no. 11, pp. 2683–2703, Mar. 2003.

Ali Tajer (S'05–M'10–SM'15) received the B.Sc. and M.Sc. degrees in electrical engineering from the Sharif University of Technology in 2002 and 2004, respectively, and the M.A. degree in statistics and the Ph.D. degree in electrical engineering from Columbia University in 2007 and 2010, respectively. He was a Postdoctoral Research Associate at Princeton University from 2010 to 2012. He is currently an Assistant Professor of Electrical, Computer, and Systems Engineering with Rensselaer Polytechnic Institute. His research interests include mathematical statistics, network information theory, wireless communications, and power grids.

Dr. Tajer was a recipient of the 2016 NSF CAREER Award. He serves as an Editor for the IEEE Transactions on Smart Grid and the IEEE Transactions on Communications, and is the Guest Editor-in-Chief for the IEEE Transactions on Smart Grid Special Issue on Theory of Complex Systems With Applications to Smart Grid Operations.