Adaptive Learning in Time-Variant Processes With Application to Wind Power Systems

Eunshin Byon, *Member, IEEE*, Youngjun Choe, *Student Member, IEEE*, and Nattavut Yampikulsakul

***Abstract*—This study develops new adaptive learning methods for a dynamic system where the dependency among variables changes over time. In general, many statistical methods focus on characterizing a system or process with historical data and predicting future observations based on a developed time-invariant model. However, for a nonstationary process with time-varying input-to-output relationship, a single baseline curve may not accurately characterize the system's dynamic behavior. This study develops kernel-based nonparametric regression models that allow the baseline curve to evolve over time. Applying the proposed approach to a real wind power system, we investigate the nonstationary nature of wind effect on the turbine response. The results show that the proposed methods can dynamically update the time-varying dependency pattern and can track changes in the operational wind power system.**

***Note to Practitioners*—This study aims at characterizing the dynamic outputs of a wind turbine such as power generation and load responses on turbine subsystems. The turbine responses evolve over time due to the range of time-varying factors. Changes in both internal and external factors affect the power generation capability and load levels. Some of these factors are not measurable (or quantifiable) and thus, changes in these factors cause the wind-to-power and the wind-to-load relationships to be nonstationary. This study proposes adaptive procedures for capturing the time-varying relationship among variables. The results can improve the prediction capability of wind turbine responses, and can be also applicable to other engineering systems subject to dynamic operating conditions.**

***Index Terms*—Kernel-based learning, nonparametric regression, nonstationary process, prediction, wind turbine.**

I. Introduction

THIS STUDY develops adaptive learning methods for a dynamic system where the output (or response) depends on input factors. Specially, we consider a system where the dependency of output on input factors changes over time. Typical regression analysis focuses on finding the single baseline curve that best fits the historical data. However, for the time-variant system, this static regression curve cannot accurately describe the dependency among variables, thus resulting in poor prediction performance.

This study is motivated by improving the prediction accuracy of responses in wind power systems. Wind power is among the fastest-growing renewable energy sources in the United States [1], [2]. Today, 3–4% of the domestic energy supply is provided by wind power, and the number is expected to rise rapidly in the near future [1]. Given this trend, researchers are now focusing on quantifying wind turbine responses including wind power generation and structural/mechanical load responses. The load prediction and quantification can be used in model-predictive controls (e.g., pitch and torque controls) for mitigating damages and avoiding failures in the wind turbine [3].

In general, a turbine manufacturer provides a power curve that characterizes a turbine's power output as a deterministic function of the hub-height wind speed [4]. Then, wind power is predicted using the deterministic power curve supplied by the manufacturer, given the forecasted wind speed. Therefore, in the past, substantial efforts have been undertaken to improve the forecast accuracy of wind speeds, including time series analysis such as autoregressive moving average and generalized autoregressive conditional heteroscedasticity models [5] and numerical weather prediction models [6].

However, different from the deterministic power curve, the empirical power curve shows more dynamic patterns. Unlike other conventional power systems, wind turbines operate under nonsteady aerodynamic loading and are subjected to stochastic operating conditions. Furthermore, even under the same wind conditions, actual power generation changes over time, reflecting the intrinsic changes that alter the ability of the turbine to respond to wind forces. A combination of several ambient conditions (e.g., humidity), external effects (e.g., dust and insect contamination and/or ice accumulation on blades) and internal effects (e.g., wear and tear on components), changes a turbine's production and affects the aerodynamic property of the turbine [7]. These factors make the wind-to-power relationship nonstationary. The same insights apply to the load response as well. Fig. 1 shows the scatter plots of the 10-min averages of three response variables and wind speeds from the data collected for two different periods of actual operations of a 500 kW turbine (see Section V for a description of the data). From Fig. 1, we note that the dependency of the turbine outputs on wind speeds varies over time, thus demonstrating the nonstationary characteristics.

Manuscript received November 19, 2014; revised April 02, 2015; accepted May 25, 2015. Date of publication July 10, 2015; date of current version April 05, 2016. This paper was recommended for publication by Associate Editor D. Djurdjanovic and Editor J. Wen upon evaluation of the reviewers' comments. This work was supported by the National Science Foundation under Grant CMMI-1362513. This paper was presented in part at the IIE Annual Conference and Expo (ISERC), Montréal, QC, Canada, June 2, 2014, and in part at the Ninth Annual IEEE International Conference on Automation Science and Engineering (IEEE CASE), Madison, WI, USA, August 17, 2013. *(Corresponding author: Eunshin Byon.)*

The authors are with the Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI 48109-2117 USA (e-mail: ebyon@umich.edu; yjchoe@umich.edu; nattavut@umich.edu).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org provided by the authors. The Supplementary Material includes a supplementary document. This material is 1.8 MB in size.

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/TASE.2015.2440093

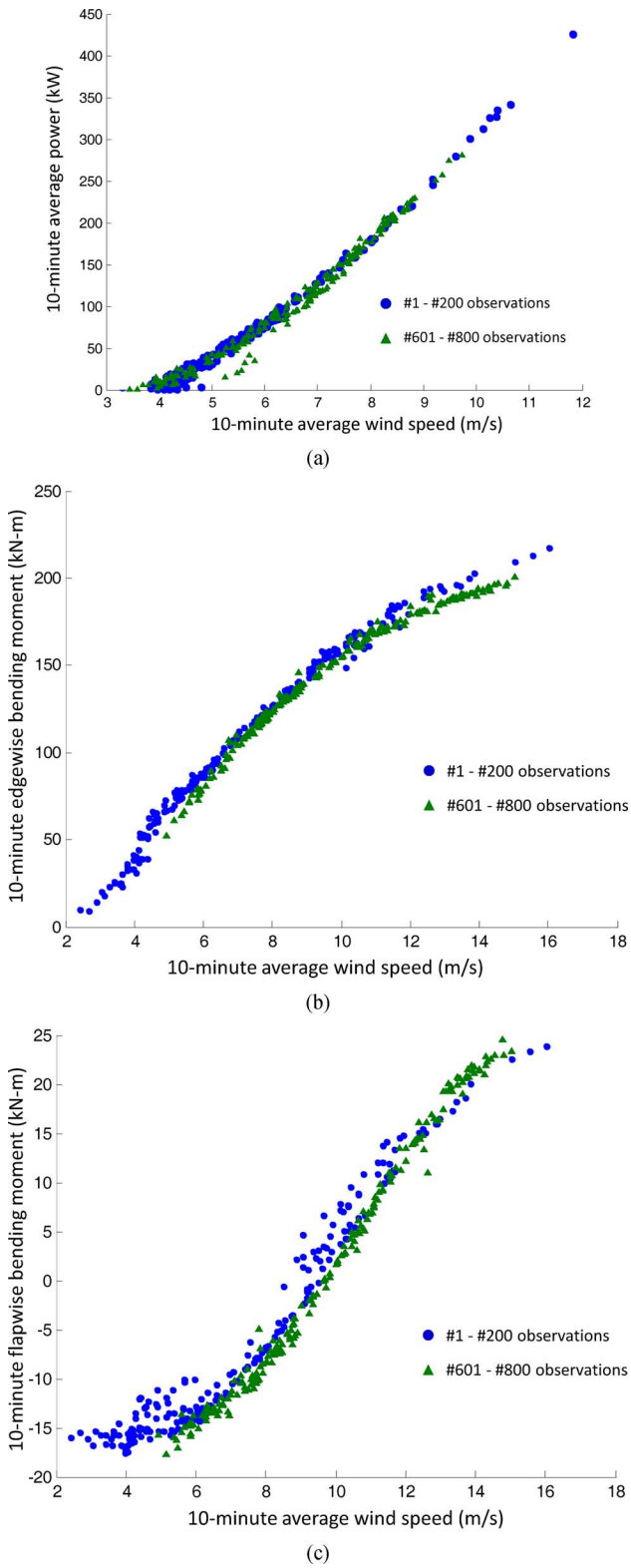

Fig. 1. Scatter plots between 10-min average turbine response (y axis) and 10-min average wind speed (x axis) during two different periods from a 500 kW wind turbine. (a) Power generation. (b) Edgewise bending moment. (c) Flapwise bending moment.

Attempts to capture the stochastic nature of the power curve have been made using data-driven, statistical methods. Sánchez [8] uses a linear regression model for representing the power curve, assuming a specific input-to-output relationship. However, when the relationship between the dependent and explanatory variables exhibits a complicated, nonlinear pattern, linear regression-based methods often fail to characterize the input-to-output dependency. Recent studies suggest that the turbine structure experiences nonlinear stresses, so the power and load responses show nonlinear dependency on wind conditions [9]–[11]. Pinson *et al.* [12] propose a local linear regression model using a first-order Taylor expansion to approximate a nonlinear power curve. Kusiak *et al.* [13] compare the power prediction performances with several data mining techniques. Among the various data mining techniques, neural network (NN) models, including the recurrent NN (RNN) [14], fuzzy NN [15] and ridgelet NN [16], have been employed to capture the nonlinear relationship between wind speed and power. In the study by De Giorgi *et al.* [17], the power prediction capabilities of NN models are compared with those of ARMA models, and the results suggest the necessity of employing a nonlinear power curve for improving power prediction. Barbounis *et al.* [14] address the nonstationary issue by training RNN models for predicting long-term wind speed and power. The studies by Lee and Baldick [18] and Sideratos and Hatziargyriou [15] integrate the wind forecast information from the numerical weather prediction model in the NN framework.

This study develops new learning methods that accommodate the time-varying dependency pattern among variables. To capture the nonlinearity between wind conditions and turbine responses, we use a nonparametric regression without imposing any restriction on the input-to-output relationship. The proposed method adaptively characterizes the change in the baseline curve, but regulates how rapidly the baseline curve can change. The proposed approach is called an *adaptive regularized learning* (ARL) method in this study. We prove that the solution of the proposed model always exists, and provide the closed-form solution to update the baseline function. We also present an alternative model that approximates the original ARL model but can be solved much more efficiently.

The prediction performances of the proposed methods are validated with the data collected from a land-based operational turbine. The results show that the proposed approach can identify the changing reaction of a turbine to wind force and improve the prediction accuracy over a nonparametric regression model that assumes a time-invariant dependency. Comparison with the RNN-based models also reveals that the proposed approach generates better prediction results and is computationally more efficient.

The remainder of this paper is organized as follows. The proposed approach is presented in Section II. Section III includes the implementation details. Section IV and Section V discuss numerical examples and a case study, respectively, for evaluating the performance of the proposed methods. We summarize the paper in Section VI.

II. Mathematical Model

This section models the progressive change of a system's input-to-output relationship and describes the proposed methods and the solution procedures. The detailed derivations and proofs are available in the supplementary document.

A. Formulation of Nonstationary Regression Function

Consider a sequence of observations, $S = \{(y_t, x_t), t = 1, \ldots, N\}$, ordered by time where N denotes the current period. Here, x_t is a vector of input factors at the t^{th} period and $y_t \in \mathbf{R}$ is a response. In the wind power system, x_t could be a vector of weather conditions; y_t could be either the power output or the load response measured at certain hotspots (e.g., bending moment measured from strain sensors). These observations can be collected from sensors installed in a wind turbine and the supervisory control and data acquisition system known as SCADA.

The functional relationship between the input and the output can be written as $y = f_t(x) + e$, where e is a random noise. To identify the baseline function, $f_t(\cdot)$, we use a nonparametric model without assuming a specific function type. First, when a stationary input–output model is considered with $f(\cdot)$, the relationship between x and y can be learned in a manner similar to nonlinear curve fitting. This study employs the kernel method [19] for fulfilling this learning objective because of its flexibility and capability. Specifically, the input vector, x, is mapped into a feature space, $\mathcal{H}$, via a nonlinear map, $x \to \phi(x)$, so that $y = \omega^T \phi(x) + e$ where the superscript ^{T} denotes the transpose of a matrix. The inner product of $\phi(\cdot)$ produces a kernel function, $k(\cdot, \cdot)$, i.e., $k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$. Many choices of $k(\cdot, \cdot)$ are available, thereby providing highly flexible models of nonlinear mappings. In fact, some choices of $k(\cdot, \cdot)$, e.g., the radial basis kernel, have an infinite-dimensional $\phi(\cdot)$ that cannot be written analytically. As such, the resulting kernel learning method is nonparametric, different from parametric curve fitting with basis functions fixed *a priori*. In the kernel method, $k(\cdot, \cdot)$, rather than $\phi(\cdot)$, is explicitly specified, so that the difficulty of dealing with a high dimensional $\phi(\cdot)$ can be avoided; this is the so-called kernel trick [19].

Next, we allow the coefficient, w, to vary over time in order to reflect the nonstationary nature of the system behavior and to characterize the time-variant baseline function. Thus, the model becomes

$$y = \omega_t^T \phi(x) + e, \qquad t = 1, \ldots, N \tag{1}$$

where w_t is a nonparametric regression coefficient vector (hereafter, coefficient) at period t.

However, unremarkable this change may appear at first glance, the new model actually presents a challenge: instead of having a constant coefficient, ω, it has a time-varying coefficient, ω_t, that a typical model fitting procedure fails to estimate. To solve this new model, this study uses the idea of regularized learning. Regularized learning is a popular machine learning approach that places constraints on model parameters and regulates the model complexity [20]. Making use of this general idea, our regularized learning strategy regulates the rapidity of the model's possible changes. In most engineering systems, a system model will not change much over a short period, so that it is reasonable to assume that the system's change in consecutive periods will be gradual. Making use of this idea, this study proposes the two methods for estimating the nonstationary baseline function, $f_t(x)$.

B. Adaptive Regularized Learning (ARL)

Assuming that the initial coefficient, w_0, is known, to update the baseline function at period N with the data set S, the problem is formulated by regulating the norm of the change in the regression coefficient in (1) as well as by minimizing the discrepancy between the actual observations and the estimated responses as

$$\min \quad L_O = \frac{1}{2} \sum_{t=1}^{N} \|w_t - w_{t-1}\|_2^2 + \frac{1}{2} \gamma \sum_{t=1}^{N} e_t^2 \tag{2}$$

$$s.t. \quad y_t = w_t^T \phi(x_t) + e_t, \qquad t = 1, \ldots, N \tag{3}$$

where $\|\cdot\|_2$ represents the L^2-norm. $\gamma > 0$ is a regularization parameter, balancing between the baseline change (the first term) and the quality of model fitting (the second term). This regularization parameter controls the relative importance between the nonstationarity and empirical errors.

As a remark, the quadratic penalty in (2) is considered due to its computational tractability in updating the baseline function with a closed-form solution, which will be addressed in the following discussion. A possible research extension will be to investigate different penalties.

The model in (2)–(3) takes a similar form of the kernel ridge regression or least squares support vector regression [21], [22]. The difference is that assuming a stationary process, the kernel ridge regression uses the constant coefficient, w, in the constraint and regulates the model complexity in the objective.

To solve the proposed ARL model, we follow a similar approach used in the kernel ridge regression; however, due to the time-varying w_t's, the solution procedure is more complicated. Using Lagrangian multipliers, α_t's, $t = 1, \ldots, N$, the optimization problem in (2)–(3) becomes

$$\min \quad L'_O = \frac{1}{2} \sum_{t=1}^{N} \|w_t - w_{t-1}\|_2^2 + \frac{1}{2} \gamma \sum_{t=1}^{N} e_t^2 - \sum_{t=1}^{N} \alpha_t (w_t^T \phi(x_t) + e_t - y_t). \tag{4}$$

The Karush–Kuhn–Tucker conditions are applied for optimality [23]. The Lagrangian multipliers can be obtained by solving the following linear system:

$$\begin{bmatrix} k_{11} + \frac{1}{\gamma} & k_{12} & k_{13} & \cdots & k_{1N} \\ k_{21} & 2k_{22} + \frac{1}{\gamma} & 2k_{23} & \cdots & 2k_{2N} \\ k_{31} & 2k_{32} & 3k_{33} + \frac{1}{\gamma} & \cdots & 3k_{3N} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ k_{N1} & 2k_{N2} & 3k_{N3} & \cdots & Nk_{NN} + \frac{1}{\gamma} \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_N \end{bmatrix} = \begin{bmatrix} y_1 - \hat{y}_{0,1} \\ y_2 - \hat{y}_{0,2} \\ y_3 - \hat{y}_{0,3} \\ \vdots \\ y_N - \hat{y}_{0,N} \end{bmatrix}$$

or

$$\left(K'_N + \frac{1}{\gamma} I_N\right) \boldsymbol{\alpha}_N = Y_N - \hat{Y}_{0,N}. \tag{5}$$

Here, $k_{ij} = k(x_i, x_j)$, and K'_N is an $N \times N$ matrix whose $(i,j)^{th}$ component is $\min(i,j) \times k(x_i, x_j)$. I_N is an identity matrix of size $N \times N$, $\boldsymbol{\alpha}_N = [\alpha_1, \ldots, \alpha_N]^T$, and $Y_N = (y_1, \ldots, y_N)^T$. $\hat{Y}_{0,N}$ is an $N \times 1$ vector whose i^{th} component is $\hat{y}_{0,i} = w_0^T \phi(x_i)$, $i = 1, \ldots, N$.

To find $\boldsymbol{\alpha}_N$ in (5), the matrix, $K'_N + (1/\gamma) I_N$, should be invertible. Lemma 1 proves the invertibility of $K'_N + (1/\gamma) I_N$.

Lemma 1: $K'_N + (1/\gamma) I_N$ is positive definite and thus it is always invertible.

Since $K'_N + (1/\gamma) I_N$ is invertible, $\boldsymbol{\alpha}_N$ can be obtained by

$$\boldsymbol{\alpha}_N = \left(K'_N + \frac{1}{\gamma} I_N\right)^{-1} (Y_N - \hat{Y}_{0,N}). \tag{6}$$

Whenever a new observation is obtained, finding $\boldsymbol{\alpha}$ in (6) by inverting the matrix, $K'_N + (1/\gamma) I_N$, is not computationally efficient. Especially when N is large, the matrix inversion can be computationally demanding, which leads to our use of the block matrix inversion. Let M_{N-1} denote the matrix, $K'_{N-1} + (1/\gamma) I_{N-1}$.

Lemma 2: Suppose that M_{N-1}^{-1} and $\boldsymbol{\alpha}_{N-1}$ are obtained at period $N-1$. Define $\kappa = [k_{1N}, 2k_{2N}, \ldots, (N-1)k_{N-1,N}]^T$ and $h = N k_{NN} + 1/\gamma - \kappa^T M_{N-1}^{-1} \kappa$. At period N, $\boldsymbol{\alpha}_N$ is given by

$$\boldsymbol{\alpha}_N = \begin{bmatrix} \boldsymbol{\alpha}_{N-1} + \frac{1}{h} M_{N-1}^{-1} \kappa \kappa^T \boldsymbol{\alpha}_{N-1} - \frac{1}{h} M_{N-1}^{-1} \kappa (y_N - \hat{y}_{0,N}) \\ -\frac{1}{h} \kappa^T \boldsymbol{\alpha}_{N-1} + \frac{1}{h} (y_N - \hat{y}_{0,N}) \end{bmatrix}.$$

Lemma 2 shows that the solution at the previous period can be used to attain the solution at the current period.

The result of Lemma 1 guarantees the existence and uniqueness of the solution of the model in (2)–(3). Then, the fitted baseline function at period t, $t = 1, \ldots, N$, is given by

$$\hat{y}_t(x) = \left(w_0 + \sum_{i=1}^{N} \min(i,t) \alpha_i \phi(x_i)\right)^T \phi(x) \tag{7}$$

$$= w_0^T \phi(x) + \sum_{i=1}^{N} \min(i,t) \alpha_i k(x, x_i). \tag{8}$$

Detailed derivations is available in the supplementary document.

In (8), the prediction requires the initial value of the coefficient, w_0. We obtain w_0 by applying the kernel ridge regression to the data set with initial N_0 records in the historical time series data, $S_0 = \{(y_{0,t}, x_{0,t}), t = 1, \ldots, N_0\}$. Note that the data in S_0 should not be overlapped with the data in S (see Section III for the detailed discussion on the data partition). Let $\alpha_{0,t}$ denote the Lagrangian multiplier in the kernel ridge regression [19] for $t = 1, \ldots, N_0$. Then, w_0 is

$$w_0 = \sum_{i=1}^{N_0} \alpha_{0,i} \phi(x_{0,i}). \tag{9}$$

After plugging (9) into (8) and utilizing the kernel trick, the fitted baseline function at period N becomes

$$\hat{y}_N(x) = \sum_{i=1}^{N_0} \alpha_{0,i} k(x, x_{0,i}) + \sum_{i=1}^{N} i \alpha_i k(x, x_i). \tag{10}$$

It should be noted that by writing $\hat{y}_N(x)$ in this form, this approach does not need to explicitly define the nonlinear mapping, $\phi(\cdot)$. It also does not need to compute the time-varying regression coefficients, w_t's. Instead, the time-varying information is embedded in the Lagrangian multipliers, α_t's.

C. Sequential Adaptive Regularized Learning

Even with the use of the block matrix inversion technique, solving the original ARL model discussed in Section II-B could pose computational difficulties when applied to a large data set. The original ARL method requires updating the set of coefficient sequences whenever a new observation is obtained; recall that in Lemma 2, with a new observation (y_N, x_N) at period N, we not only obtain α_N, but also update $\alpha_1, \alpha_2, \ldots, \alpha_{N-1}$. In other words, all of the Lagrangian multipliers change. This updating process can take longer as the data size increases.

Therefore, we develop the sequential approximation of the original ARL method. The sequential ARL method obtains w_N using the previous estimate, w_{N-1}, upon a new observation at period N, and only regulates the difference of w_{N-1} and w_N at period N in contrast to the original ARL that regulates all of the differences of w_i's in the two consecutive periods up to the current period, N. The original ARL in (2)-(3) is transformed into the following optimization problem:

$$\min \quad L_s = \frac{1}{2} \|w_N - w_{N-1}\|_2^2 + \frac{1}{2} \gamma e_N^2 \tag{11}$$

$$s.t.$$

$$y_N = w_N^T \phi(x_N) + e_N. \tag{12}$$

In (11), w_{N-1} is the estimate of the coefficient at period $N-1$. Applying the similar procedure used in Section II-B allows us to obtain the Lagrangian multiplier of the sequential learning in (11)-(12) as follows:

$$\alpha_N = \left(k(x_N, x_N) + \frac{1}{\gamma}\right)^{-1} (y_N - \hat{y}_{N-1}), \tag{13}$$

where $\hat{y}_{N-1} = w_{N-1}^T \phi(x_N)$.

With α_N in (13), w_N can be obtained from w_{N-1} as

$$w_N = w_{N-1} + \alpha_N \phi(x_N). \tag{14}$$

By the recursion of (14), w_N can be rewritten as

$$w_N = w_0 + \sum_{i=1}^{N} \alpha_i \phi(x_i),$$

where the Lagrangian multipliers, α_i, $i = 1, \ldots, N-1$, are those obtained at the previous periods.

Now, the fitted baseline function at an input vector, x, at period N is given by

$$\begin{aligned}\hat{y}_N(x) &= w_N^T\phi(x)\\ &= \left(w_0 + \sum_{i=1}^{N}\alpha_i\phi(x_i)\right)^T\phi(x)\\ &= w_0^T\phi(x) + \sum_{i=1}^{N}\alpha_i k(x,x_i) \qquad (15)\\ &= \hat{y}_{N-1}(x) + \alpha_N k(x,x_N). \qquad (16)\end{aligned}$$

In (16), $\hat{y}_{N-1}(x) = w_0^T\phi(x) + \sum_{i=1}^{N-1}\alpha_i k(x,x_i)$ is the fitted baseline function at period $N-1$. Therefore, (16) implies that upon a new observation, (y_N, x_N), the fitted baseline is updated by $\alpha_N k(x,x_N)$ from the previous estimation.

As in the original ARL, the initial regression coefficient, w_0, in (15) can be obtained by using the kernel ridge regression method with the initial N_0 observations (see (9)). Then, the fitted baseline function at period N with the sequential ARL method becomes

$$\hat{y}_N(x) = \sum_{i=1}^{N_0}\alpha_{0,i}k(x,x_{0,i}) + \sum_{i=1}^{N}\alpha_i k(x,x_i). \qquad (17)$$

Similar to the original ARL, the nonlinear mapping, $\phi(\cdot)$, does not have to be explicitly defined in this sequential learning.

III. Implementation Details

This section specifies a few details for implementing the proposed approach, including how the data set is partitioned, the specific methods for parameter selection, and the evaluation criteria used in this study.

The whole data set is divided into three groups. The first two groups serve as a training set to obtain the model parameters. The kernel ridge regression is applied to the first group, consisting of N_0 records, to select the kernel bandwidth in the Gaussian kernel function and obtain w_0. A cross validation is used to get the optimal kernel bandwidth that minimizes the mean squared error. With the selected bandwidth, we get w_0. The second group, consisting of N_0^γ records, is for selecting the regularization parameters, either γ in (2) of the original ARL or γ in (11) of the sequential ARL.

The third group serves as a testing set for evaluating the performance of the proposed methods. The baseline function is updated period by period with new observations. In order to use the kernel trick, the same kernel bandwidth (obtained from the first group) is used in the testing set. It should be noted that in the analysis that does not consider the temporal evolution, data can be randomly divided to form the training and testing sets. The proposed methods, however, track sequential progression of the regression coefficient, w_t, and thus, the first two groups (the training set) of observations should be used for initializing the coefficient, w_0, and determining the regularization parameter, γ. Then, given w_0 and γ, the coefficient can be updated, e.g., $w_1, w_2, \ldots$, sequentially with new observations.

To quantify the effect of the time-variant baseline curve, we use the most updated baseline for the short-term prediction and compute the prediction error. Because our objective is to adaptively characterize the dependency between dependent and explanatory variables, we focus on the prediction of responses given input conditions. As such, it is assumed that the input condition, x_{N+1}, at the next period $N+1$, is given (or, precisely forecasted). In both original and sequential ARL methods, the predicted response at period $N+1$ is $\hat{y}_{N+1}(x_{N+1}) = w_{N+1}^T\phi(x_{N+1})$. However, w_{N+1} is unknown at the current period N. Assuming the change from w_N to w_{N+1} is small, we use $\hat{y}_N(x_{N+1})$ in (10) and (17) for the one-step ahead prediction in the original and sequential ARL, respectively. The model's prediction performance is evaluated with the mean squared error (MSE)

$$\text{MSE} = \frac{1}{M}\sum_{t=1}^{M} r_t^2, \qquad (18)$$

where r_t is the prediction error that is the difference between the actual observation at period t, y_t, and its prediction, $\hat{y}_t$. M is the number of records in the testing data set.

The following summarizes the procedure of each method.

1) Divide the sequence of observations into three groups of sizes N_0, N_0^γ and M, respectively.
2) Initialize ω_0 (or equivalently, obtain $\alpha_{0,t}, t = 1,\ldots,N_0$) using the initial N_0 observations by solving the kernel ridge regression.
3) Determine the regularization parameter, γ in (2) and (11) for the original and sequential ARL, respectively.
   a) For each candidate value for γ, repeat for $N = 1,\ldots,N_0^\gamma$,
      - for the original ARL, upon a new observation (y_N, x_N) at each period N, update the Lagrangian multipliers, $\alpha_t, t = 1,\ldots,N$, using the block matrix inversion in Lemma 2 and update the baseline function using (10);
      - for the sequential ARL, upon a new observation (y_N, x_N) at each period N, obtain α_N using (13) and update the baseline function using (17).
   b) Given the input condition at period $N+1$, x_{N+1}, use $\hat{y}_N(x_{N+1})$ in (10) and (17) for the original and sequential ARL, respectively, to make the one-step ahead prediction.
   c) Compute MSE in (18) with the N_0^γ observations.
   d) Select the best regularization parameter that provides the lowest MSE in each method, and with the N_0^γ observations, update the baseline function using (10) and (17) for the original and sequential ARL, respectively.
4) With the M observations in the testing set, update the baseline function using (10) and (17) for the original and sequential ARL, respectively and compute MSE in (18).

IV. Numerical Examples

This section demonstrates the proposed ARL methods using numerical examples. Specifically, the $sinc$ function is modified to represent a time-varying process as follows:

$$\begin{aligned}y_t &= \beta_t\frac{\sin(\nu_t x_t)}{x} + e,\\ \beta_t &= \beta_{t-1} + \delta\beta,\\ \nu_t &= \nu_{t-1} + \delta\nu,\end{aligned}$$

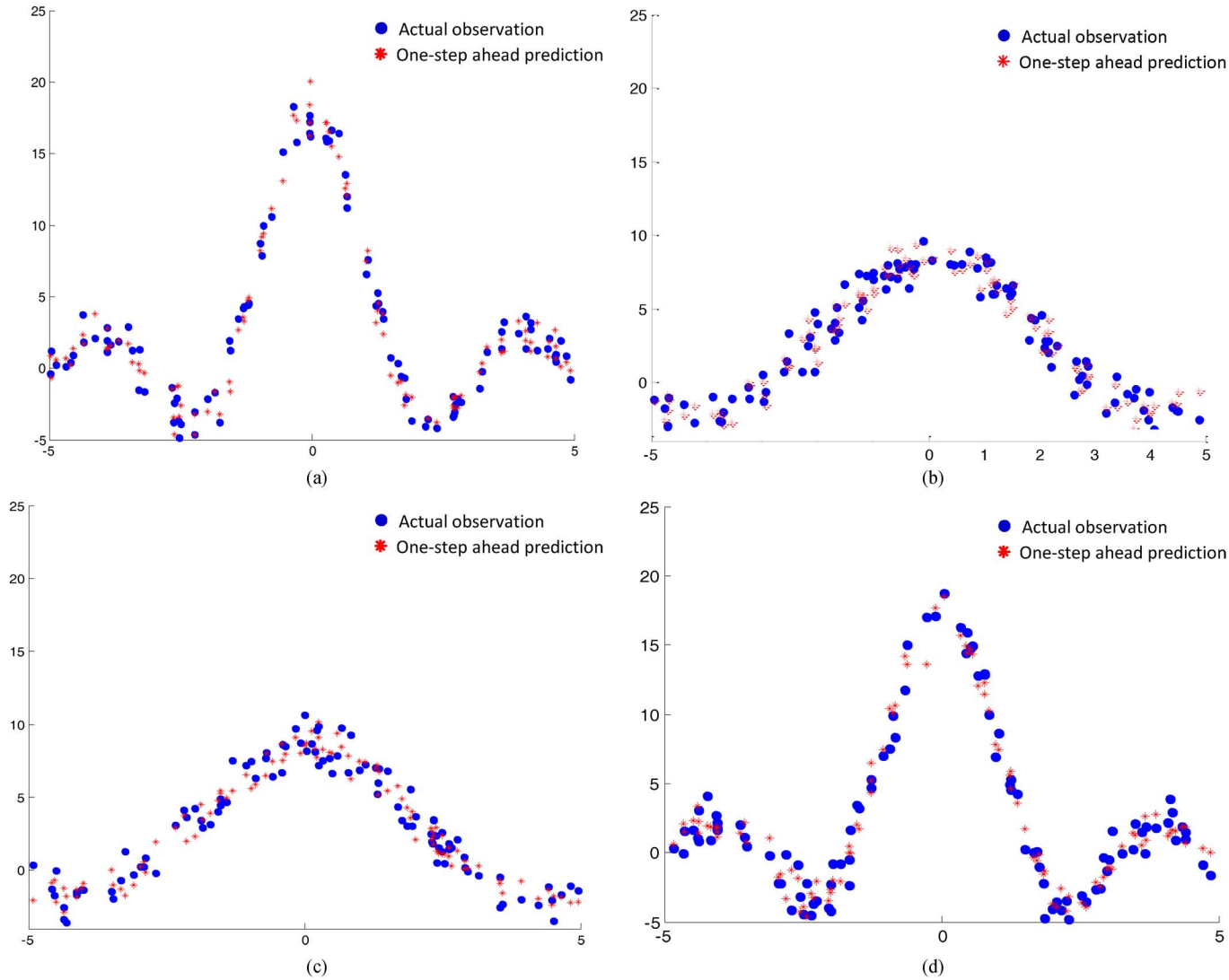

Fig. 2. Comparison between responses and one-step ahead predictions using sequential ARL at different periods (x axis: x_t, y axis: y_t). (a) $N = 1, \ldots, 100$. (b) $N = 901, \ldots, 1000$. (c) $N = 1001, \ldots, 1100$. (d) $N = 1901, \ldots, 2000$.

where $e \sim N(0, 1)$. The initial values of β_t and ν_t are set to be $\beta_0 = 10$ and $\nu_0 = 3$, respectively. In the training set, $N_0 = 600$ and $N_0^\gamma = 400$ are used. x_t is uniformly sampled in $[-5, 5]$. The Gaussian kernel is used in our implementation. Under this setting, we investigate the performance of the proposed approach with different transition rates, $\delta\beta$ and $\delta\nu$.

A. Analysis With Seasonal Variations

In the training set, $\delta\beta = -0.001$ and $\delta\nu = -0.001$ are used to simulate the time-varying baseline. Then, we simulate the 1000 observations during $N = 1, \ldots, 1000$ periods to form a testing set, using the same $\delta\beta$ and $\delta\nu$. The solid dots in Fig. 2(a) and (b) show the heterogeneous response patterns at the first 100 periods and the last 100 periods in this testing set, respectively. During these periods, the observations tend to move closer to zero and the cyclic pattern's frequency gets smaller over time [see the pattern change from Fig. 2(a) to (b)]. The prediction results using the sequential ARL match well with the observations, indicating that the sequential ARL successfully tracks the time-varying input-to-output relationship.

In some processes, the time-varying patterns could change, depending on the variations of operational conditions, e.g., due to seasonal effects. To investigate the performances of the proposed methods in such processes, we additionally simulate 1000 observations with different transition rates, namely, $\delta\beta = 0.001$ and $\delta\nu = 0.001$, during $N = 1001, \ldots, 2000$ periods in the testing set. The solid dots in Fig. 2(c) and (d) depict the observations during the first 100 and the last 100 periods in this second testing set. Unlike the pattern change during the first testing periods, the magnitude of observations tend to increase and the cyclic pattern's frequency gets higher during the second testing periods [see the pattern change from Fig. 2(c) to (d)]. Because the changes occur slowly, the patterns in Fig. 2(b) and 2(c) appear to be similar even though the observations are collected under different $\delta\beta$ and $\delta\nu$ (e.g., different seasons). The good match between the observations and predictions in Fig. 2(c) and (d) is the evidence that the sequential ARL can adapt to the variations of the dynamic process.

Table I summarizes the prediction performance using the original and sequential ARL in the entire testing set. The two

TABLE I
ONE-STEP AHEAD PREDICTION RESULTS (MSES)

Original ARL	Sequential ARL
1.39	1.55
(1.36, 1.41)	(1.55, 1.55)

Note: The regularization parameters used in the original and sequential ARL are $\gamma = 0.1$ and 0.8, respectively. The two values inside each parenthesis represent the MSEs during the first 1000 periods with $\delta\beta = -0.001$ and $\delta\nu = -0.001$ and the next 1000 periods with $\delta\beta = 0.001$ and $\delta\nu = 0.001$, respectively.

TABLE II
ONE-STEP AHEAD PREDICTION RESULTS DURING 10000 PERIODS WITH SLOW TRANSITION RATES (MSES)

	Sequential ARL	Kernel ridge regression
$\delta\beta = \delta\nu = 0.0001$	1.08	18.23
$\delta\beta = \delta\nu = 0.00001$	1.06	1.23

Note: In each case, the regularization parameters used in the sequential ARL are 0.12 and 0.04, whereas the static kernel ridge regression uses 1 and 150, respectively.

values inside each parenthesis represent the MSEs during the first 1000 periods with $\delta\beta = -0.001$ and $\delta\nu = -0.001$ and the next 1000 periods with $\delta\beta = 0.001$ and $\delta\nu = 0.001$ in the testing set, respectively.

B. Analysis With Slower Transition Rates

We further investigate whether the proposed approach accounts for changes when the transitions evolve very slowly during a long duration. As before, $N_0 = 600$, $N_0^\gamma = 400$ are used. For testing periods, a much longer duration of 10000 periods ($N = 1, \ldots, 10000$) is considered. We employ the two sets of transition rates shown in the first column of Table II. To check the adaptability of the proposed approach, the results of the sequential ARL are compared with those from the kernel ridge regression that assumes a stationary process. Note that the kernel ridge regression uses a static baseline estimated using the training data. Table II, which summarizes the one-step ahead prediction results of the sequential ARL and the kernel ridge regression, shows that the sequential ARL exhibits consistently better prediction accuracy. The performance gain of the sequential ARL over the static model is small under the extremely small rate with $\delta\beta = 0.00001$, $\delta\nu = 0.00001$ because the baseline change is minimal. Even in this case, the sequential ARL improves the prediction accuracy over the model assuming a stationary process.

V. CASE STUDY

This section discusses the results of our case study using the data set from a 500 kW land-based turbine at Roskilde in Denmark. The data, provided by Risø-DTU, Technical University of Denmark [24], was collected at 35 Hz in April 2006 for 15 days. Three turbine outputs are investigated: power generation, flapwise bending moment and edgewise bending moment at the turbine's blade root. Flapwise and edgewise bending moments, collected from strain sensors, measure the structural loads in two orthogonal directions.

We use 10-min averages for the power generation, flapwise bending moment and edgewise bending moment as the response variable, respectively. Wind velocity, which is a vector form of wind speed, is used as the explanatory variable. As such, the input vector, x, consists of two variables, wind speed $\times$ $cosine$(wind direction) and wind speed $\times$ $sine$(wind direction). Many types of kernel functions are available in the kernel-based learning. In general, the choice of an appropriate kernel function depends on each problem. The implementation results with different kernel functions including Gaussian, linear and polynomial kernels suggest that the Gaussian kernel produces robust outputs in characterizing the wind-to-response relationship in all of the three responses.

Before implementing the proposed approaches, we remove the outliers which do not follow the general pattern of turbine responses by imposing simple rules. For example, in the power response, we exclude the data points where the power is negative. The training set includes $N_0 = 400$ and $N_0^\gamma = 100$ observations for each response, and the testing set includes $M = 1196$, 1382 and 1363 observations for the power generation, flapwise bending moment and edgewise bending moment, respectively.

A. Comparison With Other Methods

The proposed ARL methods are compared with several benchmark methods. The first method is the kernel ridge regression method that regulates the model complexity, but assumes a stationary process. This method generates a static baseline function.

Next, as discussed earlier, NN has been used for predictions of wind speed and power generation in the literature. Among many available network architectures, the RNN architecture, specifically, infinite impulse response multilayer perceptron (IIR-MLP) [25] proved particularly useful through the extensive experimentations in the study by Barbounis *et al.* [14]. The detailed descriptions are available in the supplementary document. We consider two approaches: (a) RNN without updating the model parameters, where we train the network based on a training set and do not update the parameters during testing, as in [14] and (b) RNN with updating the model parameters, where we do retrain the network by updating the parameters upon every new observation. We call the two approaches as nonadaptive RNN and adaptive RNN, respectively. It should be noted that both approaches address the nonstationary issue by using the internal states memorized in the IIR filters. In the adaptive RNN, updating the parameters with all the data available up to the current period requires unreasonably lengthy computational time (more than 6 hours), which is impractical in real applications. Therefore, we update the parameters with the most new observation, similar to the sequential ARL.

Finally, we also implement the exponentially weighted moving average (EWMA) which is one of the most commonly used forecasting methods in the time-series data analysis [26]. At period N, the predicted response at period $N+1$ is

$$\begin{aligned}\hat{y}_{N+1} &= (1-\lambda)\sum_{r=0}^{N-1}\lambda^r y_{N-r} \\ &= (1-\lambda)y_N + \lambda\hat{y}_N.\end{aligned}$$

TABLE III
ONE-STEP AHEAD PREDICTION RESULTS (MSES)

	Power generation $(kW)^2$	Edgewise moment $(kN\text{-}m)^2$	Flapwise moment $(kN\text{-}m)^2$
Original ARL	30.3	12.8	0.6
Sequential ARL	48.4	19.3	0.8
Kernel ridge regression	160.7	110.9	12.7
Adaptive RNN	249.9	62.2	5.2
Nonadaptive RNN	353.5	307.6	16.7
EWMA	1515.1	179.9	9.1

Note: The regularization parameters used in the original, sequential and kernel ridge regression for power generation are 1, 0.7 and 5000, respectively. The regularization parameters used for edgewise moment are 10, 3.6 and 900, whereas the regularization parameters used for flapwise moment are 10, 4.1 and 900, respectively.

The forgetting factor, $0 < \lambda < 1$, determines the weights on the past observations; a smaller weight is given to an observation farther from the current period; a large (small) λ indicates the long (short) memory of the temporal process. Accordingly, the choice of λ determines the prediction performance. In our implementation, we choose λ that minimizes MSE of the model in the training set.

B. Prediction Results

Table III summarizes the one-step ahead prediction results for the three turbine response variables. In each prediction, the input and output are based on 10-minute averages. The original ARL generates the lowest prediction errors in all of the three responses. The sequential ARL generates slightly higher prediction errors than the original ARL, but the differences are minimal. The result also indicates that the kernel-based learning algorithms generally provide better prediction capabilities than RNN methods. The nonadaptive RNN generates much higher predictions errors than the kernel ridge regression. The adaptive RNN improves the prediction accuracy over the nonadaptive RNN, however, its MSEs are three to nine times higher than the proposed ARL approaches. EWMA produces significantly high prediction errors because it does not utilize the information of the wind speed and direction, but only uses the previous observed responses for its predictions.

Figs. 3–4 compare the actual observations of the edgewise bending moment with their corresponding prediction results for the two different sets of periods, i.e., 1*st*–200*th* and 601*st*–800*th* periods, in the testing set. The proposed ARL methods show a good match between the actual observations and the predictions. On the contrary, the kernel ridge regression and the adaptive RNN display many discrepancies. The proposed approaches also show superior performance for other responses (see the comparison plots in the supplementary document).

As a remark, there are two reasons for the presence of multiple predictions at the same wind speed in these figures. First, we use the wind *velocity* that considers wind direction as the explanatory variables instead of using wind *speed*. Second, whenever a new observation is obtained, we update the baseline function in the ARL approaches.

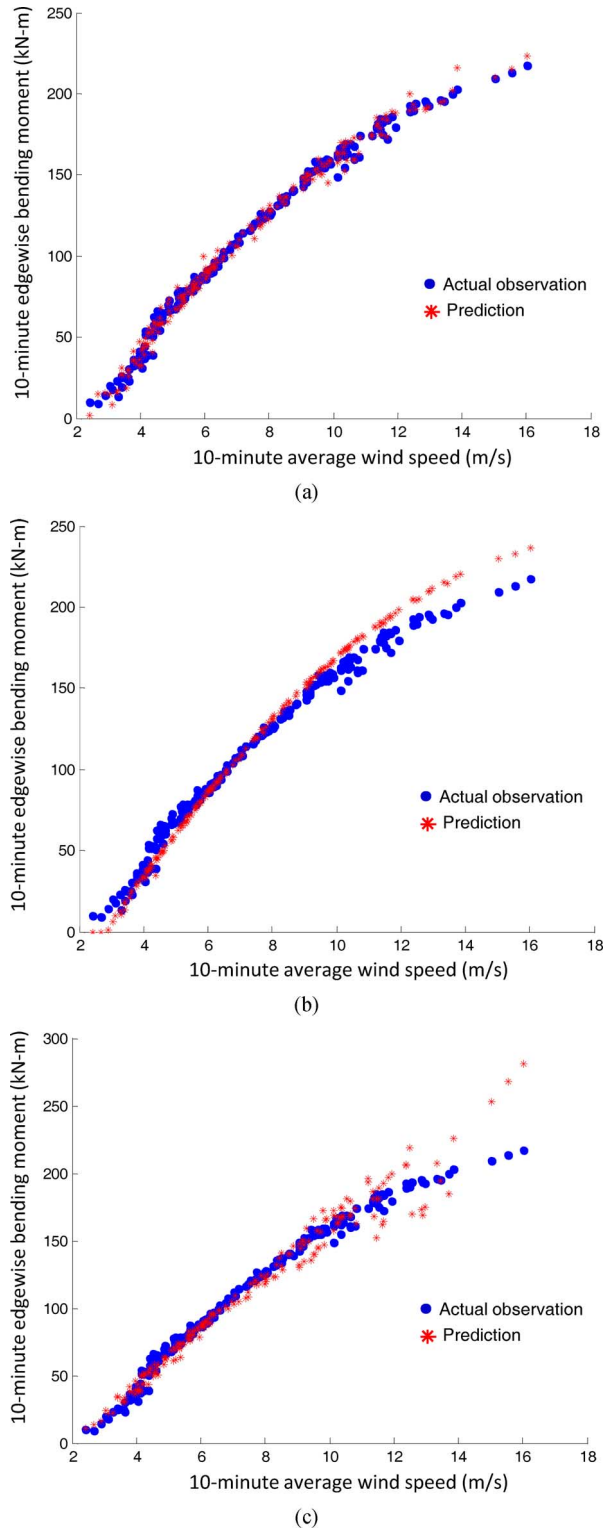

Fig. 3. Prediction results of edgewise bending moment during #1–#200 periods in the testing set (note: the prediction results from the original ARL are similar to those from the sequential ARL). (a) Sequential ARL. (b) Kernel ridge regression. (c) Adaptive RNN.

C. Computational Efficiency

We evaluate the computational efficiency of the proposed approaches and the adaptive RNN in Table IV. The computation time includes both the prediction time as well as the baseline

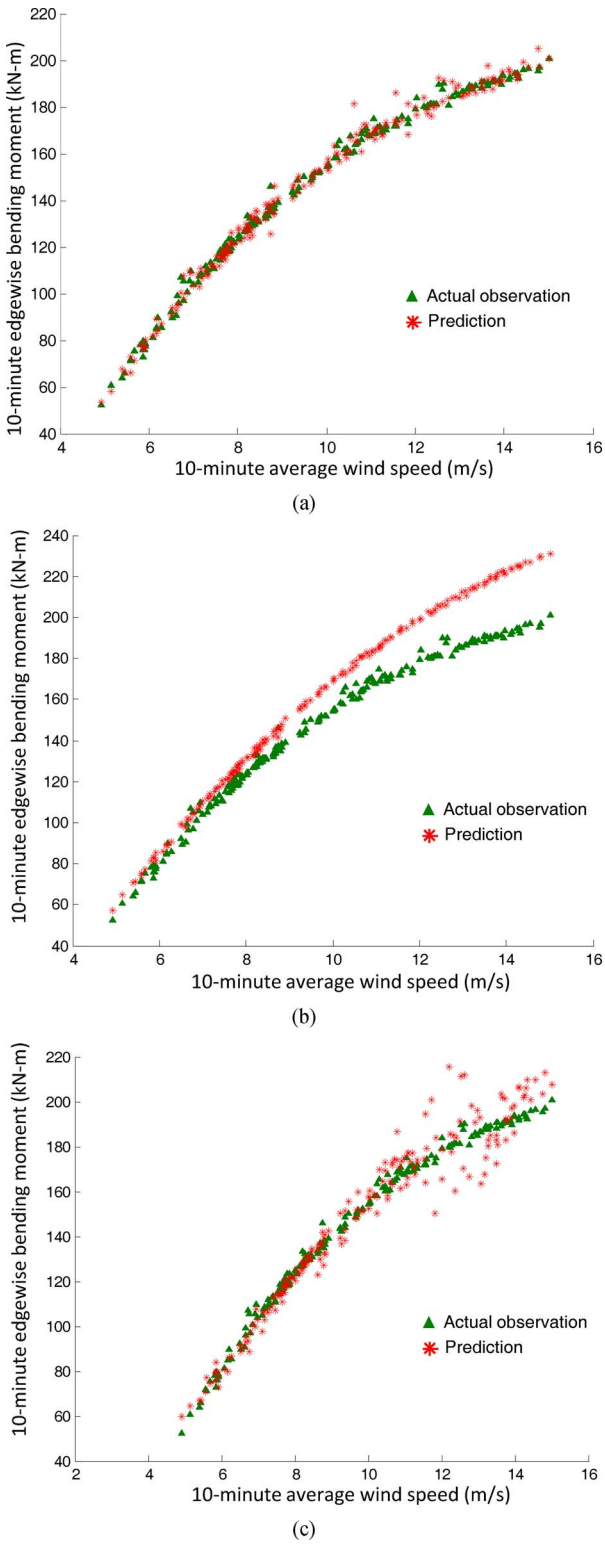

Fig. 4. Prediction results of edgewise bending moment during #601–#800 periods in the testing set (note: the prediction results from the original ARL are similar to those from the sequential ARL). (a) Sequential ARL. (b) Kernel ridge regression. (c) Adaptive RNN.

function (or relevant model parameters) updating time upon each new observation. Therefore, the computation times from the kernel ridge regression, nonadaptive RNN and EWMA are not included in the table because they do not update the relevant

TABLE IV
COMPARISON OF COMPUTATIONAL EFFICIENCY (UNIT: SECONDS)

	Power generation	Edgewise moment	Flapwise moment
Original ARL	96.1	131.7	138.9
Sequential ARL	0.1	0.1	0.1
Adaptive RNN	1085.6	1237.4	1246.7

TABLE V
10-MIN AHEAD PREDICTION RESULTS WITH THE BATCH SEQUENTIAL ARL (MSES)

	$B = 5$	$B = 10$	$B = 15$	$B = 20$
Power	56.0	69.2	79.1	97.9
Edgewise	34.9	40.3	44.5	45.4
Flapwise	1.4	2.1	2.3	3.1

Note: The regularization parameters used for power generation are 6.5, 9.6, 6.2 and 20.0 when the batch size is 5, 10 , 15 and 20, respectively. The regularization parameters used for edgewise moment are 1.0, 1.0, 1.0 and 6.2, whereas the regularization parameters used for flapwise moment are 1.5, 1.3, 19.9 and 1.6, respectively.

model parameters with new observations. The adaptive RNN is much less efficient than the proposed ARL approaches. It appears that the sequential ARL is the most efficient method. The data set used in this study spans around 15 days. We believe that the computational efficiency of the sequential ARL would be much more evident in a larger scale data set collected for a longer period.

D. Batch Sequential Adaptive Regularized Learning

A batch learning approach is additionally considered, which is called "batch sequential ARL" in this study. Recall that the sequential ARL updates the baseline upon every new observation. On the contrary, the batch sequential ARL updates the baseline every B observations.

Table V, which summarizes the 10-min ahead prediction results with different batch sizes, shows slightly higher MSEs than those from the sequential ARL in Table III. Regarding the computational efficiency, the batch sequential ARL uses the computational time similar to the sequential ARL, and even with a larger batch size, the computational time does not decrease. The reason is that unlike the sequential ARL, the batch sequential ARL involves the matrix inversion in its updating process, which is computationally more challenging than solving a linear equation in (13).

E. Application to Fault Detection

The proposed approach can be applicable to detect faulty conditions. As discussed in Section I, the system process could change due to the combination of both external operating conditions and internal degradations. In a typical engineering system, degradation initially evolves slowly but once damage begins, the rate of degradation is noticeably accelerated. We believe that the intrinsic change in a system's baseline during the slow degradation should be treated as "normal" behavior to differentiate it from the changes during the rapid damage progression period. The proposed approach can identify such rapid damage progression. The idea is that in the proposed ARL methods, the

TABLE VI
PREDICTION RESULTS IN NUMERICAL EXAMPLES WITH RAPID BASELINE CHANGE RATES UNDER ABNORMAL CONDITIONS (MSES)

	Original ARL	Sequential ARL
$\delta\beta = -0.003$, $\delta\nu = -0.01$	2.96	3.69
$\delta\beta = -0.05$, $\delta\nu = -0.05$	15.56	17.56

regularization term, γ, is related to the rate of model changes and is determined using the data collected during normal periods. Accordingly, when damage starts to rapidly progress and the process change rate starts to deviate from the slowly changing rate under the normal condition, the predictions from the proposed approach will also deviate from real observations.

Because the wind turbine data in this case study does not include observations under faulty conditions, the numerical example discussed in Section IV is used to illustrate the effectiveness of the proposed method in fault detection. Specifically, we use the data generated in Section IV-A to represent the normal condition, and then generate additional data of size 100 with faster transition rates to simulate an abnormal system condition. Table VI shows that with these rapid rates, the prediction errors increase compared to those in Table I. The high prediction errors reflect the large deviations of the actual observations from the anticipated responses under the normal condition. Therefore, the proposed approach will be capable of detecting abnormalities by tracking prediction errors. Note that the higher MSE in the last row with more rapid transition rates indicates that the proposed approach becomes more sensitive to detect system changes as the difference of rates under normal and abnormal conditions gets larger.

Even though the results indicate the potential strength of the proposed approach in detecting faults in the nonstationary processes, the fault detection methods should differ, depending on the degradation rate. When the degradation process is very slow, fault diagnosis with the baseline updates will be unnecessary, and a nonadaptive baseline can be used. On the other hand, when the degradation occurs relatively fast, the approach discussed in this section would be applicable. Also, as in our case study, when a limited number of operational factors are used as explanatory variables, it will be important to distinguish changes in the operating conditions from the changes in the system's health condition. Devising effective fault detection methods that consider the degradation rate and operating conditions will be studied in our future research.

F. Effects of Other Environmental Factors

In addition to the wind conditions, other environmental factors, including air density and humidity, affect the turbine responses [27]. According to the study by Lee *et al.* [27] where the effects of multiple factors on the power prediction performance are investigated, it turns out that the air density is one of the significant factors affecting the power generation for land-based turbines. The air density, ρ, is a function of temperature, T, and air pressure, a_p, that is, $\rho = a_p/(R \cdot T)$, where T and a_p are expressed in Kelvin and Newton/m^2 and $R = 287(\text{Joule})(\text{kg})^{-1}$ [27]. Because the data set used in this study only includes the temperature measurement, we include the ambient temperature as one of the input variables in addition to the wind velocity. The third and fourth columns of Table VII compare the prediction results without and with the temperature. The result indicates that adding the temperature does not significantly improve the prediction accuracy. In the future, when other environmental measurements become available to us, we will study the effect of each factor, which would help reduce the updating frequency.

TABLE VII
COMPARISON OF 10-MIN AHEAD PREDICTION PERFORMANCE WITHOUT AND WITH TEMPERATURE (MSES)

		Wind velocity	Wind velocity & Temperature
Power	Original ARL	30.3	28.4
	Sequential ARL	48.4	43.8
	Kernel ridge regression	160.7	163.8
Edgewise	Original ARL	12.8	13.1
	Sequential ARL	19.3	16.5
	Kernel ridge regression	110.9	121.8
Flapwise	Original ARL	0.6	0.7
	Sequential ARL	0.8	1.0
	Kernel ridge regression	12.7	14.7

Note: In the fourth column, the regularization parameters used for power generation are 2, 0.7, 2900, respectively. The regularization parameters used for edgewise moment are 19, 10 and 400, whereas the regularization parameters used for flapwise moment are 5, 5.8 and 10, respectively.

VI. SUMMARY

This study proposes regularized learning methods for nonstationary processes based on a time-variant nonparametric regression analysis. We formulate the optimization problem to regulate the change in the baseline function as well as to minimize the empirical estimation errors. Using the insight from the original ARL model, we develop a sequential version where the regression coefficient is updated from the last period's one for each new observation. Our implementation for making short-term predictions suggests that the ARL methods are superior to other benchmark methods. The prediction capability of the sequential ARL is similar to the original ARL, while providing much better computational efficiency compared to other methods. We believe the sequential ARL provides a new data-driven way of learning a nonstationary process.

The case study uses the bending moment and power data to illustrate the nonstationary characteristics of wind turbine systems. In the current wind industry, the sensors that collect bending moment signals are not yet commonly employed. However, with the increasing importance of condition monitoring systems and load-mitigating controls [3], it is expected that various types of sensors will be deployed in turbine subsystems. The presented approach will be generally applicable to various types of sensor measurement. In addition, our case study does not consider the effect of control on the turbine responses such as pitch controls. Depending on the values of the control parameters, each turbine will generate different responses even under the same wind condition. In this case, the control parameters can be included as explanatory variables, so that the input vector includes the information of both wind condition and control parameters.

The next step in our research will advance the prediction method; in this study, the most updated baseline function is used

for future predictions. When the time-varying input-to-output relationship changes slowly, our proposed prediction provides accurate predictions as observed in our case study and numerical examples. When the system behavior changes rapidly, we plan to additionally consider the magnitude and direction of the response change to improve the prediction accuracy. Another research direction is to develop an effective method for removing outliers and imputing the missing data in the nonstationary process. Devising effective fault detection methods that consider the degradation rate and operating conditions will be also studied in our future research. Finally, although the performance of the batch sequential ARL appears slightly worse than that of the sequential ARL in the wind turbine case study, its prediction capability would depend on the dynamic characteristics of the system, and we are interested in investigating its performance in other applications.

REFERENCES

[1] 20% Wind energy by 2030: Increasing wind energy's contribution to U.S. electricity supply U.S. Dept. Energy, Washington, DC, USA, Tech. Rep. DOE/GO-102008-2567, 2008.

[2] E. Byon and Y. Ding, "Season-dependent condition-based maintenance for a wind turbine using a partially observed Markov decision process," *IEEE Trans. Power Syst.*, vol. 25, no. 4, pp. 1823–1834, Nov. 2010.

[3] M. M. Hand and M. J. Balas, "Blade load mitigation control design for a wind turbine operating in the path of vortices," *Wind Energy*, vol. 10, no. 4, pp. 339–355, 2007.

[4] E. Byon, E. Pérez, Y. Ding, and L. Ntaimo, "Simulation of wind farm maintenance operations and maintenance using discrete event system specification," *Simul.*, vol. 87, no. 12, pp. 1093–1117, 2011.

[5] A. Lau and P. McSharry, "Approaches for multi-step density forecasts with application to aggregated wind power," *Ann. Appl. Stat.*, vol. 4, no. 3, pp. 1311–1341, 2010.

[6] S. Al-Yahyai, Y. Charabi, and A. Gastli, "Review of the use of numerical weather prediction (NWP) models for wind energy assessment," *Renewable Sustainable Energy Rev.*, vol. 14, no. 9, pp. 3192–3198, 2010.

[7] Y. Zhang, T. Igarashi, and H. Hu, "Experimental investigations on the performance degradation of a low-reynolds-number airfoil with distributed leading edge roughness," in *Proc. 49th AIAA Aerosp. Sci. Meet.*, Orlando, FL, USA, 2011.

[8] I. Sánchez, "Recursive estimation of dynamic models using Cook's distance, with application to wind energy forecast," *Technometrics*, vol. 48, no. 1, pp. 61–73, 2006.

[9] G. Lee, E. Byon, L. Ntaimo, and Y. Ding, "Bayesian spline method for assessing extreme loads on wind turbines," *Ann. Appl. Stat.*, vol. 7, no. 4, pp. 2034–2061, 2013.

[10] N. Yampikulsakul, E. Byon, S. Huang, S. Sheng, and M. Yu, "Condition monitoring of wind turbine system with nonparametric regression-based analysis," *IEEE Trans. Energy Convers.*, vol. 29, no. 2, pp. 288–299, Jun. 2014.

[11] Y. Choe, E. Byon, and N. Chen, "Importance sampling for reliability evaluation with stochastic simulation models," *Technometrics*, 2015, to be published.

[12] P. Pinson, H. A. Nielsen, H. Madsen, and T. S. Nielsen, "Local linear regression with adaptive orthogonal fitting for the wind power application," *Stat. Comput.*, vol. 18, no. 1, pp. 59–71, 2008.

[13] A. Kusiak, H. Zheng, and Z. Song, "Wind farm power prediction: A data-mining approach," *Wind Energy*, vol. 12, no. 3, pp. 275–293, 2009.

[14] T. G. Barbounis, J. B. Theocharis, M. C. Alexiadis, and P. S. Dokopoulos, "Short-term wind power ensemble prediction based on Gaussian processes and neural networks," *IEEE Trans. Energy Convers.*, vol. 21, no. 1, pp. 273–284, 2006.

[15] G. Sideratos and N. D. Hatziargyriou, "An advanced statistical method for wind power forecasting," *IEEE Trans. Power Syst.*, vol. 22, pp. 258–265, Feb. 2007.

[16] N. Amjadya, F. Keyniaa, and H. Zareipourb, "Short-term wind power forecasting using ridgelet neural network," *Electr. Power Syst. Res.*, vol. 81, pp. 2099–2107, 2011.

[17] M. G. De Giorgi, A. Ficarella, and M. Tarantino, "Error analysis of short term wind power prediction models," *Appl. Energy*, vol. 88, pp. 1298–1311, 2011.

[18] D. Lee and R. Baldick, "Short-term wind power ensemble prediction based on Gaussian processes and neural networks," *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 501–510, Jan. 2014.

[19] C. M. Bishop, *Pattern Recognition and Machine Learning*, 1st ed. New York, NY, USA: Springer, 2006.

[20] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, 2nd ed. New York, NY, USA: Springer, 2009.

[21] N. Kim, Y.-S. Jeong, M.-K. Jeong, and T. Young, "Kernel ridge regression with lagged-dependent variable: Applications to prediction of internal bond strength in a medium density fiberboard process," *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 42, no. 6, pp. 1011–1020, Nov. 2012.

[22] K. D. Brabanter, J. D. Brabanter, J. A. K. Suykens, and B. D. Moor, "Approximate confidence and prediction intervals for least square support vector regression," *IEEE Trans. Neural Networks*, vol. 22, no. 1, pp. 110–120, Jan. 2011.

[23] S. Boyd and L. Vandenberghe, *Convex Optimization*, 7th ed. Cambridge, U.K.: Cambridge Univ. Press, 2009.

[24] *WindData,* 2010. [Online]. Available: http://www.winddata.com

[25] A. C. Tsoi and A. D. Back, "Locally recurrent globally feedforward networks: A critical review of architectures," *IEEE Trans. Neural Networks*, vol. 5, no. 2, pp. 229–239, Mar. 1994.

[26] D. R. Cox, "Prediction by exponentially weighted moving averages and related methods," *J. Roy. Statist. Soc. Ser. B*, pp. 414–422, 1961.

[27] G. Lee, Y. Ding, M. G. Genton, and L. Xie, "Power curve estimation with multivariate environmental factors for inland and offshore wind farms," *J. Amer. Statist. Assoc.*, to be published.

Eunshin Byon (S'09–M'14) received the Ph.D. degree in industrial and systems engineering from Texas A&M University, College Station, TX, USA, in 2010.

She is an Assistant Professor with the Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI, USA. Her research interests include optimizing operations and management of wind power systems, data analytics, quality and reliability engineering, and simulation optimization.

Prof. Byon is is a member of IIE and the Institute for Operations Research and the Management Sciences (INFORMS).

Youngjun Choe (S'13) received the B.S. degree in physics and management science from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2010. He is now working towards the Ph.D. degree at the Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI, USA.

His current research focuses on quality and reliability engineering and stochastic simulations.

Nattavut Yampikulsakul received the B.S. degree in computer science with a minor in economics, and the M.S. degree in operations research from Columbia University, New York, NY, USA, in 2008 and 2010, respectively. He is currently working towards the Ph.D. degree at the Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI, USA.

His research interest includes the fault diagnosis of wind power systems and modeling, analysis and predictions of nonstationary processes.