Forecasting the evolution of nonlinear and nonstationary systems using recurrence-based local Gaussian process models

Satish T. S. Bukkapatnam and Changqing Cheng
Sensor Networks and Complex Systems Monitoring Research Laboratory, Department of Industrial Engineering and Management, Oklahoma State University, Stillwater, Oklahoma 74075, USA

(Received 22 March 2010; revised manuscript received 3 September 2010; published 15 November 2010)

An approach based on combining nonparametric Gaussian process (GP) modeling with certain local topological considerations is presented for prediction (one-step look ahead) of complex physical systems that exhibit nonlinear and nonstationary dynamics. The key idea here is to partition system trajectories into multiple near-stationary segments by aligning the boundaries of the partitions with those of the piecewise affine projections of the underlying dynamic system, and deriving nonparametric prediction models within each segment. Such an alignment is achieved through the consideration of recurrence and other local topological properties of the underlying system. This approach was applied for state and performance forecasting in Lorenz system under different levels of induced noise and nonstationarity, synthetic heart-rate signals, and a real-world time-series from an industrial operation known to exhibit highly nonlinear and nonstationary dynamics. The results show that local Gaussian process can significantly outperform not just classical system identification, neural network and nonparametric models, but also the sequential Bayesian Monte Carlo methods in terms of prediction accuracy and computational speed.

DOI: 10.1103/PhysRevE.82.056206 PACS number(s): 05.45.Tp, 02.50.Ey, 05.10.−a, 05.45.Gg

I. INTRODUCTION

Prediction of the future states and performance from the measured signals is becoming crucial for improving monitoring and control of real-world complex systems, including several biological, physical and engineering systems [1–4]. Due to the recent advancements in sensors, computing and communication technologies, abundant data sources in the form of multidimensional time series are becoming available for analyzing complex systems. Consequently, the impetus has shifted toward harnessing information from these data sources for effective prediction, prognostics, and preventive control of these complex systems. Significant developments have taken place in the application of nonlinear dynamic system theory, and pertinently, recurrence properties of the attractors of nonlinear systems to improve prediction [5–9]. These attractor-based prediction methods use the local evolution patterns of neighbors in the state space, and/or the knowledge of local trajectory divergence rates [7–9]. They are mostly applicable to deterministic nonlinear systems, as well as nonlinear systems with stationary noise and/or simple forms of nonstationarities (e.g., seasonality and such trends in the first moment) [8,10]. Despite these research efforts, effective prediction of the future states remains a challenge because these complex systems exhibit combined nonlinear and nonstationary dynamics [11]. The structure of the nonlinear dynamic relationship among the measured signals and states remains unknown if not indeterminable in most cases. However, most prediction approaches for nonlinear time series use some form of parametric models, which are only effective as the mathematical structures of the models used to capture this relationship. The recent advances in nonparametric modeling approaches offer a unique opportunity to advance nonlinear system prediction under highly nonstationary conditions. Among the nonparametric approaches, Gaussian process (GP) regression [12] is attractive in that the Gaussian properties can be used to simplify the modeling efforts.

However, most GP models assume stationarity of the dynamics, i.e., the covariance structure remains time invariant and is identical at all points in the state space. This severely limits their predictability in many real-world systems which are highly nonlinear (often chaotic) *and* nonstationary. To overcome this limitation, nonstationary covariance functions have been attempted [13,14]. These can only capture the simple forms of nonstationarities (e.g., linear trend and seasonality). Under highly nonlinear and nonstationary conditions, local nonparametric models have been attempted to improve accuracy and computational efficiency. Although some statistical clustering methods and surrogate sample methods have been attempted (see [3] as well as additional details in Sec. II), the pertinent issue of how to segment the state space of a system into different local stationary subsystems remains elusive.

In contrast to previous works, local recurrence [15,16] and such topological properties of nonlinear systems [17] are used to partition the state space into near-stationary segments, and a separate GP model is derived for each segment. The key idea in the present local Gaussian process (LGP) modeling approach is to align the segments with the piecewise affine components of a system's state space. The resulting LGP model was found to outperform several prediction methods, including the classical system identification and forecasting methods, global Gaussian process (GGP) model, and computationally intensive mixtures of Gaussian processes (MGP), recurrent neural network, and sequential Bayesian particle filtering methods in terms of accuracy. Also, it is computationally more efficient than GGP because LGP uses only the samples within a particular segment for modeling and prediction. Consequently, the data size needed for training a GP model is greatly reduced, and the computational efficiency improved. The remainder of this paper is

organized as follows: Sec. II presents a concise background of GP regression modeling; recurrence-based local Gaussian process (LGP) modeling approach is presented in Sec. III; Sec. IV contains the results from the application of the LGP and other models for state and performance prediction (forecasting) in synthetic and real-world nonstationary and nonlinear systems; conclusions are presented in Sec. V.

II. BACKGROUND

A variety of applications of GP for state and performance prediction in domains including geophysics, robotics, human motion tracking and finance have been reported in the literature [1–4]. A GP model seeks to establish a mapping f of the form [12]

$$y = f(x) + \varepsilon \tag{1}$$

between the predictor (output) $y \in \mathbb{R}$ of a complex dynamical system with an input vector $x \in \mathbb{R}^d$, from their historical realizations (also referred to as the training set) $\mathbf{S}=\{(x_i,y_i),i=1,2,\ldots,n\}$.[1] Here, $\varepsilon \sim \boldsymbol{N}(0,\sigma^2_{noise})$ and x may include autoregressive terms (past realizations) of y. Defining $X=[x_1,x_2,\ldots,x_n]^T$ and $Y=[y_1,y_2,\ldots,y_n]^T$, we have

$$Y \sim \boldsymbol{N}(0,K(X,X)+\sigma^2_{noise}I), \tag{2}$$

where $K(X,X)$ is the covariance matrix, whose elements $K_{ij}=k(x_i,x_j)$ are the covariance functions, usually given by a squared exponential form

$$k(x_i,x_j) = \sigma_f^2 \exp\left\{-\frac{1}{2}(x_i-x_j)^T M(x_i-x_j)\right\}. \tag{3}$$

Here, σ_f^2 is the signal variance, $M=\mathrm{diag}(l)^{-2}$, l is a d-dimensional vector capturing the length scales. Roughly, l determines the separation along different dimensions of two input vectors so that the corresponding function values become uncorrelated. Thus, the points that are near each other in the input space will have similar distribution. For a certain input set expressed as an $n^* \times d$ matrix X_*, the conditional distribution of the nominal (noise-free) predictions f_* is given in terms of its first two moments as [12]

$$\bar{f}_* = K(X,X_*)^T[K(X,X)+\sigma^2_{noise}I]^{-1}Y,$$

$$\mathrm{cov}(f_*) = K(X_*,X_*) - K(X,X_*)^T[K(X,X)+\sigma^2_{noise}I]^{-1}K(X,X_*). \tag{4}$$

It may be noted that the diagonal elements of the covariance matrix are the point-wise variances of the predictor. Thus, in contrast to traditional parametric prediction methods, GP models not only provide a point estimate but a complete distribution of the predictor variable.

However, the accuracy of the prediction depends on the hyperparameters $\theta=[\{M\},\sigma_f,\sigma_{noise}]$ of the covariance function [Eq. (3)]. These may be estimated by defining a log-likelihood function

$$\log p(Y|X,\theta^*) = -\frac{1}{2}Y^T(K+\sigma^2_{noise}I)^{-1}Y - \frac{1}{2}\log|K+\sigma^2_{noise}I| - \frac{n}{2}\log(2n) \tag{5}$$

so that

$$\theta^* = \arg\max_{\theta}\{\log[p(Y|X,\theta)]\} \tag{6}$$

The estimation of the optimal hyperparameters θ^* is also referred to as the training of a GP. Typically, the log-likelihood function [in Eq. (5)] tends to be nonconcave with multiple local maxima. Various numerical methods are reported in the literature for GP training process [12].

A main drawback of GP models is that they mostly assume stationarity of dynamics, implying that the covariance structure remains time-invariant, and is identical at all points in the state space. This severely limits their predictability in many real-world systems which are known to be highly nonlinear (often chaotic) and nonstationary. Also, computational efforts toward matrix inversion in Eq. (5) can become prohibitive with large data sets.

Many attempts have been made to address the two issues of computational overhead and limitation of the stationary covariance functions. These methods include local GP with a selected subset of the data points and complex covariance functions [14,17–19], sparse Gaussian process (SPGP) [21], and mixtures of GP (MGP) models [22–25]. However, GP with complex nonstationary covariance functions, as well as MGP models require several additional hyperparameters compared to stationary GP. This can significantly undermine the computational efficiency, especially as the input space dimension increases ($d \geq 3$). MGP models can entail less computational overhead in both modeling (also referred to as inference or training) and prediction steps compared to a global GP. But for highly nonstationary systems the computational costs can be prohibitive [22]. Several SPGP methods have been investigated for reducing computational overhead during predictions with large training datasets. The idea is to select a suitable, small set of *induced samples* that serves as a surrogates to the entire training dataset **S**. Quiñonero-Candela and Rasmussen [21] provided a review of different SPGP models. Although SPGP can reduce the computational overhead, they are still global, i.e., the induced set aims to capture the global information contained in the original training data [20]. Also, choosing the induced sets can be cumbersome, and statistical approaches typically used for this purpose can be computationally intensive. Thus, when the process is highly nonstationary, the SPGP, being global, may not be suitable. Similarly, computational overhead and large data requirements would render MGP prohibitive under highly nonstationary conditions. Also, with purely statistical methods for selection of the neighborhoods, it is likely that some useful information is lost during the MGP localization. Therefore systematic and computationally efficient clustering methods are necessary to improve prediction under nonlinear and nonstationary conditions.

[1]For simplicity of the expressions we employ $y \in \mathbb{R}$.

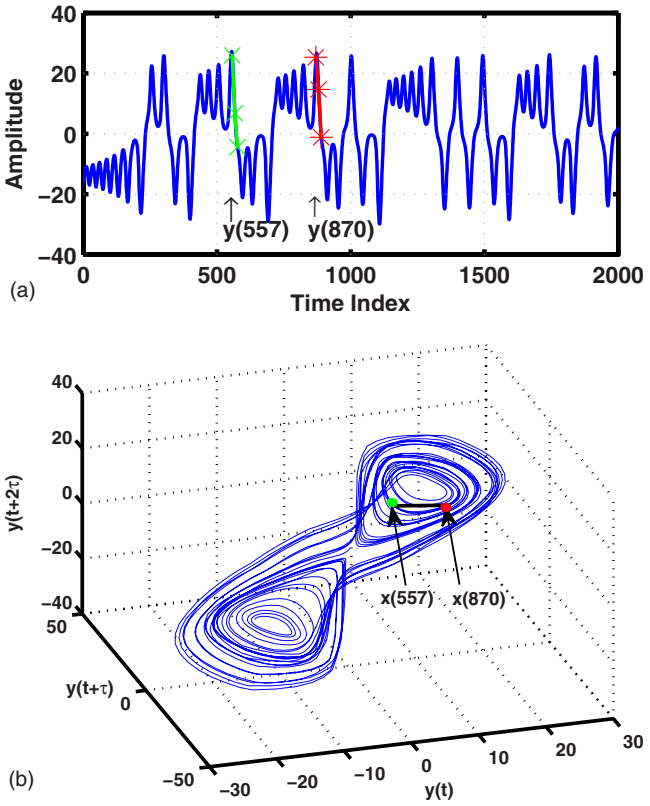

FIG. 1. (Color online) (a) Lorenz time-series and (b) state portrait obtained from delay embedding of the time-series, showing a reconstructed Lorenz attractor embedded in a three-dimensional delay coordinate space [30].

The recurrence-based LGP models presented here can overcome these drawbacks of the current GP methods by using the training data within a systematically derived segment around a prediction point. Consequently, it was found to improve the accuracy and speed of prediction as described in the following section.

III. RECURRENCE BASED LOCAL GAUSSIAN PROCESS (LGP) MODELING

For the prediction of $y_{i+1}=y(t_{i+1})$ of a nonlinear dynamic system from its past realizations $y_1, y_2, \ldots, y_i$, one can invoke the Markov property (Markov order d) such that, if $x_i = [y_i, y_{i-\tau}, \ldots, y_{i-(d-1)\tau}]^T$, where τ is the time delay (or lag) and d is the dimension of the reconstructed state space,[2] we have

$$p(y_{i+1}|y_i, y_{i-1}, \ldots, y_1) \approx p(y_{i+1}|x_i), \tag{7}$$

where x_i, reconstructed from the lags of y_i, captures the topological properties of the observable subset of the original state space of the dynamic system [28].

For example, Fig. 1 shows the Lorenz time series, and the corresponding attractor $\mathbf{A}$ embedded in a $d=3$ dimensional reconstructed space. Figure 2 is a two-dimensional (2D) projection of the reconstructed Lorenz attractor shown in Fig. 1(b). It is evident from Fig. 2 that the geometry of a Lorenz attractor can be decomposed into two near-periodic orbits ($\mathbf{U_1}$ and $\mathbf{U_3}$) about the foci $\bar{x}^1$ and $\bar{x}^3$, respectively, separated by a region ($\mathbf{U_2}$) with a saddle point $\bar{x}^2$. Such an approximation can be used to decompose a complex nonlinear attractor into simple affine segments:

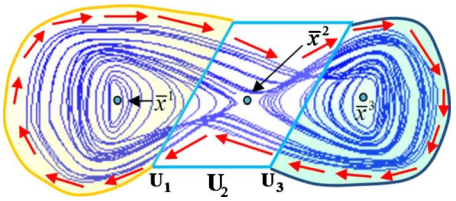

FIG. 2. (Color online) A 2D projection of a reconstructed Lorenz attractor with a schematic illustration of partitioning of the attractor into three affine segments: $\mathbf{U_1}$ and $\mathbf{U_3}$ contain trajectories about foci $\bar{x}^1$ and $\bar{x}^3$, and $\mathbf{U_2}$ about a saddle point $\bar{x}^2$.

Proposition 1 [29]. Dynamics of a nonlinear system in the vicinity of an attractor can be represented in terms of piecewise affine systems linked by switching laws as follows:

$$\dot{x} = F(x) \approx \sum_{\nu=1}^{\Gamma} B_\nu[s(x)] J_\nu (x - \bar{x}^\nu), \tag{8}$$

where $x \in \mathbb{R}^d$ is the state vector, Γ is the total number of affine subsystems, and $\bar{x}^\nu \in \mathbb{R}^d$ are the fixed points associated with each affine subsystem. Constant Jacobian matrix J_ν defines the local linear dynamics, $s(\cdot)$ defines the switching surface, and $B(\cdot)$ is a Boolean function defined so that at any given time only one affine subsystem is active.

Thus, within each piecewise affine segment the trajectories sustain a similar evolution pattern, and have a unique (in practice, similar, as will become evident in the following section) eigen-system as defined by a Jacobian matrix.

Proposition 2. Let the state space be partitioned into Γ affine segments as in Proposition 1, and let $\bar{f}_*^{U_1}$ be the noise-free prediction of f_* at $x_* \in \mathbf{U_1}$ that uses historical realizations $x_i^{U_1} \in \mathbf{U_1}, i=1,2,\ldots,n_1$ for estimation [Eq. (4)], and $\bar{f}_*^{U_{1:\Gamma}}$ be the estimate obtained using $n=n_1+n_2+\cdots+n_\Gamma$ points $x_i \in \mathbf{U_1} \cup \mathbf{U_2} \cup \ldots \cup \mathbf{U_\Gamma}$, then $\forall x_* \in \mathbf{U_1}$ we have

$$\bar{f}_*^{U_{1:\Gamma}} - \bar{f}_*^{U_1} = \Delta, \tag{9}$$

where Δ is the prediction error due to the inclusion of training points outside segment $\mathbf{U_1}$, whose closed-form expression is given in Eq. (A5) of the Appendix, and

$$\|\bar{f}_*^{U_1} - f_*\| \leq \|\bar{f}_*^{U_{1:\Gamma}} - f_*\|. \tag{10}$$

Thus, whenever all the points in the X matrix are chosen from the same segment as x_*, systemic prediction errors are not sustained. A numerical study presented in the following section shows that significant errors could be sustained when the points from other segments lie arbitrarily close to a boundary of the current segment $\mathbf{U_1}$.

[2]The time delay τ is chosen to minimize the mutual information [26] among the components of x_i and the dimension d is determined based on the false nearest neighbors test [27].

In this context, the challenge remains as to how the state space can be partitioned into near-affine segments. The recurrence property of nonlinear systems can be used to partition the state space into these piecewise affine segments. As stated in the Poincaré recurrence theorem [16], for any measure-preserving transformation on an attractor of a dynamic system, the trajectories will eventually reappear at the neighborhood of the former points in the state space. An unthresholded recurrence plot, sometimes referred to as a distance plot, can be used to capture this recurrence pattern in a d dimensional state space [10]. As summarized in Fig. 3(a), it delineates the distance between every two points $x(t_1)$ and $x(t_2)$ in the state space. For instance, color coding at the coordinate locations (557, 870) and (870, 557) in the recurrence plot represents the distance between points $x(557)$ and $x(870)$ in the state space shown in Fig. 1.

It has been recognized that variations in recurrence patterns can be used to detect certain kinds of nonstationarities [8–10]. Under stationary conditions, a recurrence plot shows fairly homogenous patterns, and the quantifiers of the recurrence, such as recurrence rate, statistical distribution of the diagonal patterns remain invariant over the length of the plot [10]. As the system transitions from one near-stationary dynamics to the next, the recurrence patterns and their quantifiers undergo a statistically significant change. Earlier, we had used certain quantitative pattern analysis of the recurrence plot to locate the boundaries between different segments [30]. We had treated the segmentation between two near-stationary segments as the detection of vertical/horizontal edges in a 2D image formed by the recurrence plot. This procedure consisted of (a) applying a common image filter based on a Sobel operator [30] at every point in the recurrence plot to obtain a contrast (binary) image. A contrast image delineates the points (coordinate locations) where significant changes in the recurrence patterns take place, and (b) interrogating at every time index whether the distribution of the contrast image points at that time index can constitute a statistically valid edge (vertical line). A simple threshold criterion was applied to determine the edges that formed the boundaries between segments.

Although Sobel operator-based segmentation was able to detect the statistically significant changes in the recurrence patterns, it is not evident if it marks the boundaries between piecewise affine segments. However, piecewise affine segment partitioning is a precondition for the present approach, as specified in Propositions 1 and 2. We have therefore investigated the tracking of the variations in correlation patterns to provide a more systematic means for state space segmentation. The basic idea is that whenever the system transitions into a different segment, the local eigensystem (as captured by the Jacobian matrix), undergoes a significant change, as can be implied from Proposition 1. Consequently, a correlation index ρ_k (e.g., R^2 statistic [31]) between the adjacent columns k and $k+1$ of the recurrence matrix will exhibit considerable decrement at the segment boundaries [see Fig. 3(b)]. The increments $\Delta\rho_k$ in the correlation index values also tend to attain their local maxima at the boundaries between segments [see Fig. 3(c)]. Evidently, marking the boundaries where the correlation change rates $\Delta\rho_k$ are unusually high can render the resulting segments to be

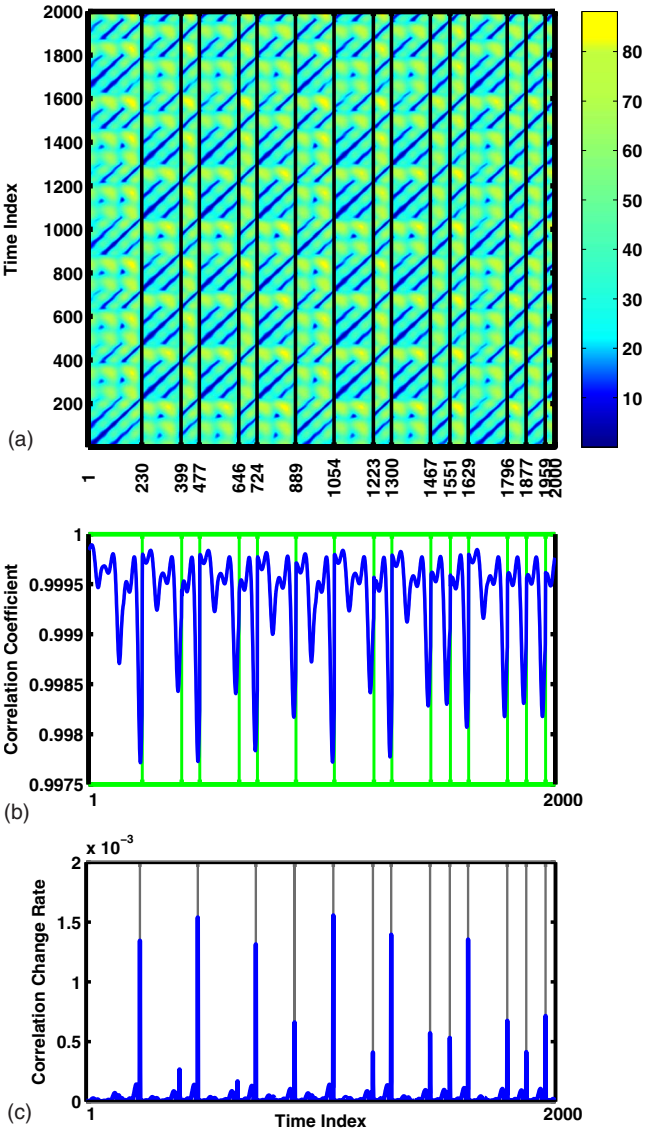

FIG. 3. (Color online) (a) Unthresholded recurrence plot of a Lorenz attractor with black vertical lines marking the true boundaries between two piecewise affine segments; (b) variation of correlation ρ_k between the adjacent columns k and $k+1$ of the recurrence matrix, with the green (light gray) vertical lines marking the boundaries as in (a), indicating that the boundaries between two piecewise affine segments are marked by local minima of the correlation ρ_k; (c) segmentation of the recurrence plot using correlation change rate $\Delta\rho_k$, with the green (light gray) vertical lines representing the boundaries between different segments. The boundaries are obtained by applying the segmentation threshold $\Delta\rho^*$ determined from the distribution of $\Delta\rho_k$ values. The results indicate that the local maxima of $\Delta\rho_k$, that are deemed statistical outliers, serve as effective boundaries between piecewise affine segments. Note that the method also tends to further partition a piecewise stationary segment into multiple such segments. However, this does not have a significant effect on prediction accuracy.

aligned with the piecewise affine and stationary components of the dynamic system. Therefore, we marked a segmentation boundary whenever $\Delta\rho_k$ values exceeded a specified threshold $\Delta\rho^*$. For statistical consistency, we had used the

concept of outlier detection for specifying the segmentation threshold $\Delta\rho^*$. In specific, the values of $\Delta\rho_k$ that exceed 1.5 times the fourth spread (also referred to as upper quartile), *fs* of the specified $\Delta\rho_k$ samples are deemed as statistical outliers [31]. Thus we had used $\Delta\rho^*=1.5\ fs(\Delta\rho_k)$ as the segmentation threshold. Our experimental investigations indicate that the application of this statistical outlier-based application of this statistical outlier-based threshold leads to near-affine segmentation of the state space.

A comparison of the segmentation using the correlation change rate criterion in Fig. 3(c) and the true segmentation of the recurrence plot of the Lorenz system in Fig. 3(a) shows that the such a criterion can correctly partition the state space into piecewise affine segments. It may also be noted that the application of this criterion tends to further partition a piecewise stationary segment into multiple such segments. However this does not have a significant effect on the prediction accuracy. When partitioned thus, as a consequence of Proposition 2, it can be shown that accuracy of predicting the noise-free estimate $\bar{f}_*$ can be improved by using the points within the same segment for estimation, i.e., the columns of the X matrix in Eq. (4) is constituted by points within the same segment.

Based on these findings, one may summarize the procedure for recurrence-based LGP as follows:

(i) Use state space reconstruction method to define the input vector x_i and output y_i;

(ii) Partition the state space using recurrence analysis to obtain various segments $\mathbf{U_1}, \mathbf{U_2}, \ldots, \mathbf{U_\Gamma}$;

(iii) Train the GP model within these segments according to Proposition 2, and obtain the optimized hyperparameters θ for each segment using Eq. (6);

(iv) For prediction at a certain location x_*, identify the corresponding segment and use the optimized LGP hyperparameters to predict f_* using Eq. (4).

As stated in Sec. II, step (iii) of the LGP procedure (i.e., training a GP model within a segment) can pose significant computational challenges. We have used a conjugate gradient method with multiple restarts for training purposes [12]. In traditional line/gradient search methods, the initial solution needs to be chosen correctly in order to avoid being trapped in a local optimum. The use of multiple restarts can alleviate this issue. Our investigations indicate the use of 20 restarts was adequate to obtain near-optimal solution for the hyperparameters θ.

IV. IMPLEMENTATION DETAILS AND RESULTS

The LGP method was compared with other prediction methods including classical time-series analysis (ARMA), GGP, sparse Gaussian process model (SPGP) [20], mixtures of Gaussian processes model (MGP) [32], extended Kalman filter (EKF) [33], recurrent predictive neural network [34], and a Monte Carlo particle filter (PF) model [35], using three benchmarking case studies. Root mean square error (RMSE) and R^2 statistics of one-step look-ahead forecasts are used as the metrics to compare the prediction accuracies of different methods.

Case 1 (Synthetic Lorenz-like system prediction). This numerical study is aimed at verifying the significance of the affine segment-based partitions and the resulting prediction error estimates given in Proposition 2. Toward this end, time-series were generated from solving a Lorenz-like system of piecewise affine differential equations (see Fig. 4 and Ref. [36]). It may be noted that segment boundaries are known *a priori* for the time series generated from solving this system of differential equations. Consequently, the effectiveness of the present approach can be verified using this time series. The system of differential equations was constructed as follows:

$$\dot{x}=\begin{bmatrix}\dot{\mathrm{x}}_1\\ \dot{\mathrm{x}}_2\\ \dot{\mathrm{x}}_3\end{bmatrix}=\begin{cases}\begin{bmatrix}-\alpha & \alpha & 0\\ \beta-\mathrm{x}_{3+} & -1 & -\mathrm{x}_{1+}\\ \mathrm{x}_{2+} & \mathrm{x}_{1+} & -\gamma\end{bmatrix}\begin{bmatrix}\mathrm{x}_1-\mathrm{x}_{1+}\\ \mathrm{x}_2-\mathrm{x}_{2+}\\ \mathrm{x}_3-\mathrm{x}_{3+}\end{bmatrix} & \mathrm{x}_1\tan\theta+\mathrm{x}_2>C\\ \begin{bmatrix}-\alpha & \alpha & 0\\ \beta-\mathrm{x}_{3o} & -1 & -\mathrm{x}_{1o}\\ \mathrm{x}_{2o} & \mathrm{x}_{1o} & -\gamma\end{bmatrix}\begin{bmatrix}\mathrm{x}_1-\mathrm{x}_{1o}\\ \mathrm{x}_2-\mathrm{x}_{2o}\\ \mathrm{x}_3-\mathrm{x}_{3o}\end{bmatrix} & -C\le\mathrm{x}_1\tan\theta+\mathrm{x}_2\le C\\ \begin{bmatrix}-\alpha & \alpha & 0\\ \beta-\mathrm{x}_{3-} & -1 & -\mathrm{x}_{1-}\\ \mathrm{x}_{2-} & x_{1-} & -\gamma\end{bmatrix}\begin{bmatrix}\mathrm{x}_1-\mathrm{x}_{1-}\\ \mathrm{x}_2-\mathrm{x}_{2-}\\ \mathrm{x}_3-\mathrm{x}_{3-}\end{bmatrix} & \mathrm{x}_1\tan\theta+\mathrm{x}_2<-C\end{cases} \tag{11}$$

where $(\mathrm{x}_{1+},\mathrm{x}_{2+},\mathrm{x}_{3+})$, $(\mathrm{x}_{1o},\mathrm{x}_{2o},\mathrm{x}_{3o})$, and $(\mathrm{x}_{1-},\mathrm{x}_{2-},\mathrm{x}_{3-})$ are the fixed points (here, we chose the three fixed points to be $[(-14,-12,0)$, $(0,\ 0,\ 5)$, and $(14,\ 12,\ 0)$, respectively], $\alpha=10$, $\beta=9$, and $\gamma=4$ are the Jacobian matrix parameters, $\theta=\pi/2$ and constant $C=4$ define the location of the switching surface (i.e., the boundary between the affine segments). This system of differential equations is considered to approximate a Lorenz attractor, as is evident from comparing the generated time series and the reconstructed attractor in Fig. 4 with the corresponding plots of Lorenz system shown in Fig. 1.

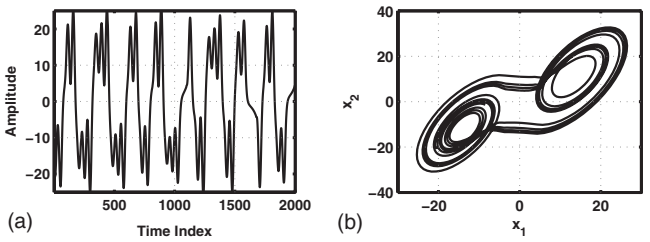

FIG. 4. (a) A 1000 data points long segment of first component of Lorenz-like system time-series, and (b) 2D projection of its corresponding phase portait. Here, the sampling interval is 0.01 time units.

This system largely meets the antecedents of Proposition 2. A 1000 data point long time-series was obtained by uniformly resampling the solution $[\mathsf{x}_1(t),\mathsf{x}_2(t),\mathsf{x}_3(t)]$ of the aforementioned system. The sampling interval of 0.01 time units was such that there were roughly 50 data points per loop (near-orbit about a focus) of the Lorenz-like attractor shown in Fig. 4.

Verification of the Propositions. Let us suppose that a prediction needs to be made at a point $x_* \in \mathbf{U_1}$ as shown in Fig. 5. We chose $n_1=30$ points (shown as green stars in Fig. 5) in the interior of the segment $\mathbf{U_1}$, and $n_2=3$ points close to the boundary (shown as red squares in Fig. 5). Evidently, all points on the reconstructed phase portrait to the left of the boundary line $\mathsf{x}_1 \tan\theta+\mathsf{x}_2<-C$ belong to segment $\mathbf{U_1}$. Furthermore, the number of points n_1 was chosen so that the prediction accuracies remained consistent across multiple replications of this numerical study and did not change significantly with further increase in the sample size.

As summarized in Table I, the prediction is very close to and within 0.01% of the observed realization y_*, implying that the prediction error $\Delta=\bar{f}^{U_1}-y_* \approx 0$ was holding consistently. Next, the $n_2=3$ points near the boundary of segment $\mathbf{U_1}$ were replaced by $n_2=3$ points from segment $\mathbf{U_2}$ (shown as red circles in Fig. 5), just across the segment boundary

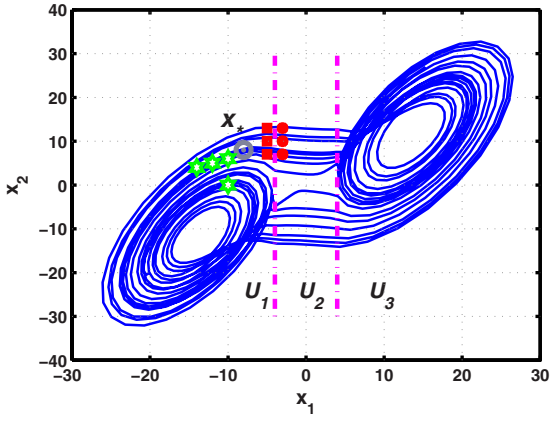

FIG. 5. (Color online) Segmentation of Lorenz-like system attractor with a representative point $x_* \in \mathbf{U_1}$ (shown as a gray annular ring) whose evolution needs to be predicted using samples taken from the interior of the segment $\mathbf{U_1}$ (shown as green stars), at the boundary of $\mathbf{U_1}$ (shown as red squares), and/or samples from $\mathbf{U_2}$ that are just across the boundary from $\mathbf{U_1}$ (shown as red circles).

TABLE I. Prediction error of GP using sample points from different segments, showing that the errors predicted using Eq. (A8) match well with those actually sustained.

	Training set X		
	$x \in \mathbf{U_1}$	$x \in \mathbf{U_1} \cup \mathbf{U_2}$	
Testing point x_*	$\Delta=\bar{f}^{U_{1:2}}-\bar{f}^{U_1}$	$\Delta=\bar{f}^{U_{1:2}}-\bar{f}^{U_1}$	Δ from Eq. (A8)
1	−0.001	−0.34	−0.41
2	0.009	−0.36	−0.32
3	0.001	−0.30	−0.31

from $\mathbf{U_1}$. In this case, we found that a significant prediction error $\Delta=\bar{f}^{U_{1:2}}-\bar{f}^{U_1}$ was sustained, although the new points were located arbitrarily close to the previous ones, but just across the boundary. Here, $\bar{f}^{U_{1:2}}$ represents the prediction obtained using sample points from both segments $\mathbf{U_1}$ and $\mathbf{U_2}$. Since the points from $\mathbf{U_2}$ are arbitrarily close to the boundary, we used the approximation from Eq. (A8) to estimate prediction errors. Evident from Table I is that the prediction errors increase by two orders of magnitude with a slight perturbation of a small subset (<10%) of the sample points. This underscores the sanctity of the affine segment boundary obtained through the recurrence analysis. From a practical standpoint, these results indicate that even under sparse data conditions, a significant improvement in prediction accuracy is possible through the use of points from the same segment and avoiding sample points, however few, from other segments.

Prediction accuracies with Lorenz-like time series. Next, we studied the effects of nonlinearity and nonstationarity on the prediction errors. Here, we used the first component of the 1000 data point long time series. The embedding parameters of $\tau=5$ and $d=3$, obtained based on the first minimum of the mutual information function [26] and the false nearest neighbors test [27], respectively, were used to reconstruct the state space. Under stationary and deterministic conditions, ARMA, GGP and MGP needed 200 data points (i.e., training samples) for model consistency, i.e., autocorrelation functions for ARMA, and hyperparameters for GGP and MGP became fairly insensitive to the composition and size of the training data. In contrast, SPGP needed just 100 points, and LGP a total of about 100 points, to ensure model consistency. The prediction accuracies of all the models were tested at 60 independent points not used for training. The application of LGP partitioned the time series into ten near-uniform segments. A segmentation threshold of $\Delta\rho^*=0.0003$ was obtained from applying the statistical outlier criterion stated in the previous section on the $\Delta\rho_k$ values computed for the first 100 points. The test results, summarized in Table II, indicate that all methods tested, namely, ARMA,[3] GGP, SPGP, MGP, and LGP were effective. Among the methods tested, LGP was found to yield the least RMSE for one-step prediction.

[3]The prediction results from an ARMA models depend heavily on the model order. Here only the results from the best ARMA model are reported.

TABLE II. Comparison of one-step look-ahead predictions for Lorenz-like system using different models showing that all the models tested have adequate prediction accuracy (low RMSE and high R^2) when the time-series is stationary, but LGP sustains significantly lower prediction RMSE and higher R^2 when the time series is nonstationary.

	Stationary series		Nonstationary series	
Method	RMSE	R^2 (%)	RMSE	R^2 (%)
ARMA(4,4)	4.15	92.1	8.13	75.2
GGP	3.68	95.6	9.74	71.6
SPGP	3.68	95.2	10.62	51.6
MGP	2.94	96.6	11.34	55.3
LGP	1.80	97.6	6.78	87.4

Subsequently, we studied the performance of the methods on the Lorenz-like time series randomly chopped and permuted, and then contaminated with additive nonstationary (piece-wise stationary) noise. Here a stationary Gaussian white noise sequence was added to a time series within a specified permuted segment. The variance of the Gaussian noise components for each segment was chosen independently of the others. The signal-to-noise ratio (SNR) of the resulting time-series segments varied between −10 and 40 dB (the noise variance ranged of 2–10), as summarized in Fig. 6(a). We note that the time-series is nonstationary in the second moment, (i.e., variance and hence signal energy) and phase (resulting from random chopping). While much of the literature in nonlinear time-series prediction address scenarios involving nonstationarities in the first and/or the second moments, nonstationarities in many real-world time series, such as heart rate, product flow rate in a production plant, etc., certainly do not appear to lend themselves to variations in the first or second moments, let alone having noticeable, gradual transitions between two wide sense stationary sets [30,37–39].

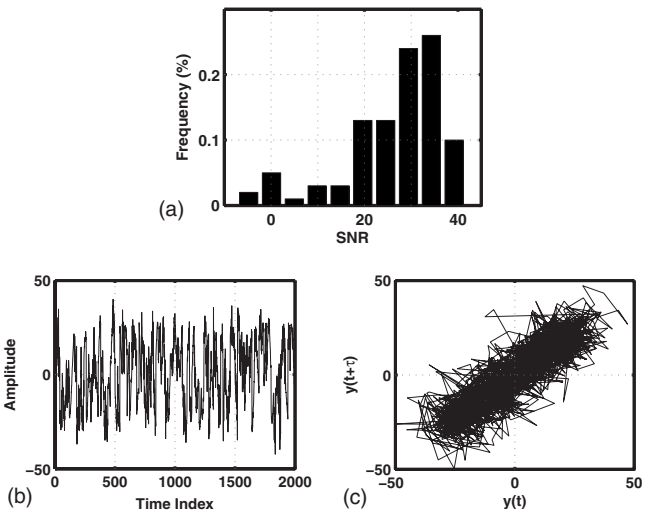

FIG. 6. (a) A histogram showing the distribution of SNR values of the noise added to various segments, (b) Lorenz-like nonstationary system time-series, and (c) a 2D projection of the corresponding phase space. Here the sampling interval is 0.01 time units.

In this work we have taken an approach to generate transients (i.e., nonstationarities), which is more aligned with the present piecewise affine criteria. We assume that a transient phase can be treated as a concatenation of evolutions of a system in the vicinity of different stationary sets (i.e., attractors). In other words, the system evolution switches among different segments of (one or more) attractors. This provides us with some theoretical rationale to formally treat transients, and derive piecewise (nonparametric) models to capture their local evolutions. We note that this construct involving random permutations also makes the resulting time-series a legitimate solution of a time-varying dynamic system where the dynamics switches to and from different (albeit simplistic) vector fields intermittently over certain short, random time segments.

The resulting time series and the 2D projections of the reconstructed attractor are shown in Figs. 6(b) and 6(c). The embedding parameters of $d=3$ and $\tau=2$ were used. The application of LGP resulted in 170 segments corresponding to a segmentation threshold $\Delta\rho^*=0.05$. Out of these segments, the 60 test points were chosen from the last 100 segments. Just as in the previous case, we had trained ARMA, GGP and MGP with 200 samples, and SPSP and LGP with about 100 samples. The prediction accuracies of the methods for one-step look ahead were tested at 60 points. In this case, the prediction RMSE of GGP was significantly (about 50%) higher than for LGP. Remarkably, LGP yields prediction accuracy R^2 of 87.4%, which shows the effectiveness of the method for nonlinear and nonstationary time-series (see Table II).

Prediction accuracies with Lorenz time series. A nonlinear Lorenz system, whose Jacobian matrix is not piecewise constant, was considered for evaluating the LGP approach. We had used a Lorenz system whose structure is similar to that Letellier used to present certain affine decompositions [35]. We had used the values $(\alpha,\beta,\gamma,C)=(10,9,4,4)$ for the Lorenz parameters, and solved the system with an initial condition of $(-20,-20,0)$ to generate a 50 000 data points long time series. The last 1000 data points of the resampled solution of this deterministic and stationary Lorenz system was used to reconstruct the state space with the embedding parameters of $d=3$ and $\tau=5$. The application of LGP yielded 13 segments with a segmentation threshold $\Delta\rho^*=0.0003$. The prediction results obtained from testing the various methods at 60 test points are summarized in Table III. For the stationary series, all the tested methods had $R^2>85\%$. Again nonstationary series were obtained by randomly chopping and permuting the Lorenz time series. The embedding parameters of $d=3$ and $\tau=1$ were used for state space reconstruction. The LGP approach partitioned this time-series into 150 segments. We had used $\Delta\rho^*=0.006$ as the segmentation threshold. Here, LGP yielded $R^2=85.3\%$, compared to 47.6% with SPGP.

It may be noted that SPGP is mostly aimed at improving the computation efficiency, which in turn depends on the induced sample set size. If the induced set size in SPGP is the same as that of the original training data, SPGP will deliver the same performance as GGP [20]. In our case, we

TABLE III. Comparison of one-step look-ahead predictions for Lorenz system using different models showing that all the models tested have adequate prediction accuracy (low RMSE and high R^2) when the time series is stationary, but LGP and MGP have the least RMSE and highest prediction R^2 when time series is nonstationary.

	Stationary series		Nonstationary series	
Method	RMSE	R^2 (%)	RMSE	R^2 (%)
ARMA(7,3)	6.51	90.8	10.98	65.6
GGP	5.98	93.4	7.23	76.4
SPGP	7.73	86.2	13.62	47.6
MGP	3.63	92.4	6.35	83.4
LGP	2.95	96.8	5.48	85.3

chose the induced set size to be half of the training data size used in GGP.

As for MGP, the prediction accuracy depends heavily on the number of clusters [23]. The MGP model partitions the input space into different clusters through a rough k-means clustering technique followed by standard expectation-maximization (EM) procedure [20]. This procedure is known to require large training sets whenever heavy transients are present in the measured time-series. These factors may help in rationalizing the somewhat inconsistent performances of SPGP and MGP for nonstationary time series.

Case 2 (Synthetic nonstationary physiological time-series by Chen *et al.* [40]). The time series (see Fig. 7) mimics the 24 h record of heart rate fluctuations. The time series shows larger variations and volatility during the periods of increased stress and physical activity, and small variations during rest periods. A 500 data points long segment was considered for our analysis. The embedding parameters of $d=6$ and $\tau=1$ were used for state space reconstruction. The application of LGP yielded 45 segments with $\Delta\rho^*=0.05$. It was noted that about 50% of the segments were less than 5 data points long, and were not considered for LGP modeling or prediction. The performance of LGP was compared with that from ARMA(3,2), extended Kalman filter (EKF), particle filter (PF), recurrent predictive neural network (RPNN) [34], GGP, MGP [32] and SPGP regression models [20]. The prediction accuracies of all the methods were tested at 100 test points. While ARMA model assumes linearity and stationarity of the process, EKF, PF, RPNN, and GP consider the underlying nonlinearities. The results indicate that LGP can outperform these approaches as indicated by significant improvements in the prediction R^2 and RMSE (see Table IV). The SPGP model (here, the induced set size is half of the original training data size) showed the worst prediction result, although it should be noted that SPGP only focuses on the computational efficiency not on modeling the nonstationarity. The RMSE of LGP is 50% less than that of ARMA(3,2) model. In fact, ARMA, GPP, and SPGP had $R^2<10\%$, indicating the inability of these methods to predict the heart-rate time series. This result implies that one needs models that explicitly consider the nonlinearity and nonstationarity for predicting such time series.

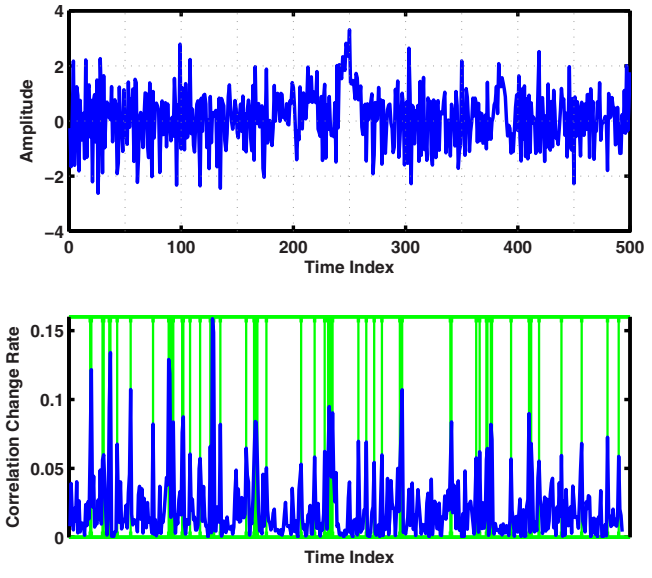

FIG. 7. (Color online) A highly nonlinear and nonstationary synthetic physiological heart-rate signal (top) and the piecewise stationary segment boundaries marked by green (light gray) vertical lines obtained from applying the correlation change rate criterion (bottom). Here the sampling frequency is set to the average heart rate of the measured time-series.

TABLE IV. Comparison of the prediction accuracies of different models for one-step look-ahead forecasting of a synthetic heart-rate time series showing that the prediction accuracies increase from R^2 of less than 10% for classical forecasting approaches to R^2 of 71% for LGP.

Method	RMSE	R^2 (%)	R^2 (bootstrap) (%)
ARMA(3,2)	1.00	2	3.2
EKF	1.03	1.5	1.1
PF	0.61	66	58
RPNN	0.73	49	46
GGP	0.99	7.5	1.1
SPGP	1.04	1.4	3.7
MGP	0.64	61	51
LGP	0.46	71	61

Furthermore, the performance of LGP was shown to be comparable to that of PF and MGP. A PF predictor relaxes the Gaussian assumption that underpins LGP, yet is known to be computationally more intensive and needs large amounts of data for highly nonlinear and nonstationary system prediction. Similarly, MGP involves much high computational overhead than LGP, and it requires elaborate tuning of the clusters to achieve acceptable prediction accuracies. The closeness of one-step predictions from LGP to the corresponding actual observations is evident in Fig. 8. Here, the gray region denotes the 95% confidence interval of prediction results. Notably, 93% of the points tested are within the 95% confidence interval compared to <70% with MGP.

Case 3 (Real time throughput prediction in an automotive

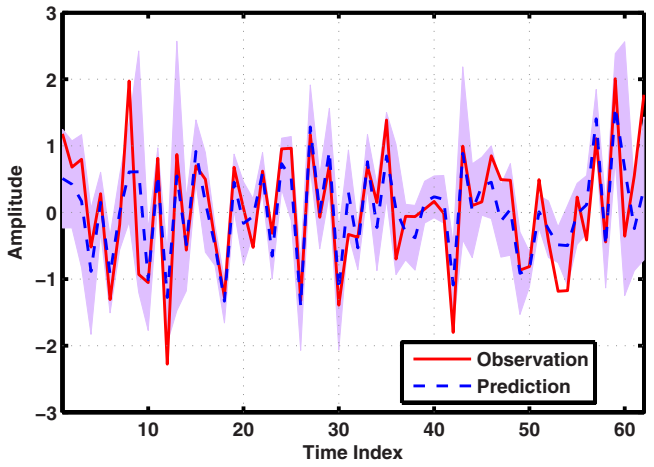

FIG. 8. (Color online) A representative segment of the synthetic physiological heart rate signal (red line) with the corresponding one-step look-ahead prediction from LGP (blue dashed line) and a 95% confidence limit (blue shade) indicating that the LGP model is able to predict the evolution of the heart rate time-series (R^2 =71%), and that 93% of the actual realizations were within the 95% of LGP-predicted confidence limits.

assembly line). The time series in Fig. 9(a) shows the number of items produced (also called the throughput) in an 8 h long shift from a station in a hypothetical automotive assembly line. As the figure indicates, the values fluctuate rapidly and erratically, rendering conventional prediction approaches unwieldy. There were about 18 different time series, each corresponding to the throughput from a station in the assembly line. The embedding parameters of $d=6$ and $\tau=1$ were found to be optimal for all the 18 time series. We note that it is likely just a coincidence that the values of the embedding parameters turned out to be the same as those used in the previous case of synthetic heart-rate data. The application of LGP for one-step look-ahead prediction yielded about 218 segments ($\Delta\rho^*=0.05$) for the 560 data points long time-series, of which the last 90 segments were used to testing the prediction accuracies. It was noted that 80% of the segments were less than 5 data points long, and they were not considered for LGP modeling or prediction. The one-step predictions from LGP model and MGP model relative to the observation values are shown in Figs. 9(c) and 9(b), respectively. Here, the PF model did not converge because of high nonstationarity and data sparsity. Also for similar reasons, GGP's performance was worse than that of the ARMA model. Our investigations also indicate that the prediction accuracies with radial basis function (RBF) models are comparable to those of ARMA [30].

Among the methods tested, LGP and MGP yielded the least prediction errors, as summarized in Fig. 10. Here, the red line in the middle of a box represents the median, the ends of the blue box indicate the lower and upper quartiles of data distributions, and the flat black line-ends indicate the extreme values that lay at the 95% of the total range of the respective values (here RMSE and R^2 statistics). It is noteworthy the median RMSE as well as its spread were the lowest for LGP and MGP compared to other methods. Also,

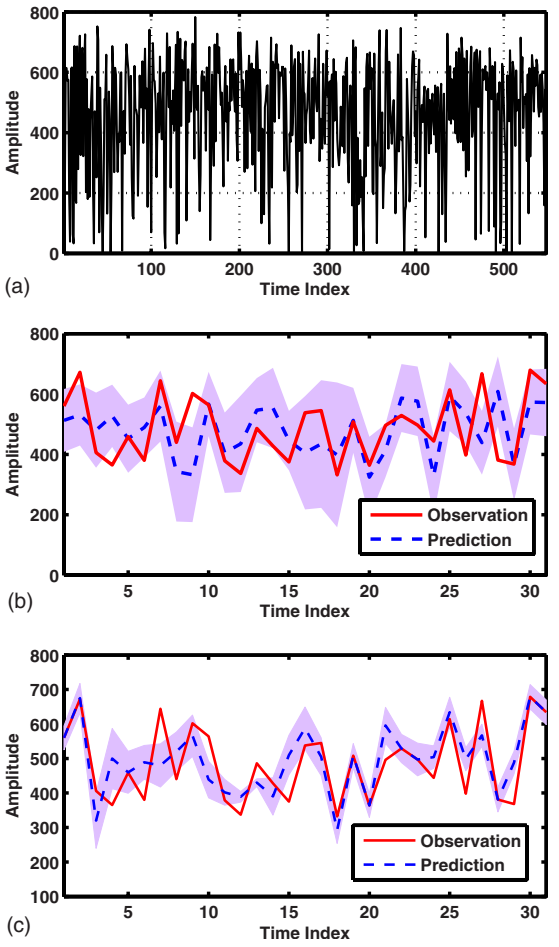

FIG. 9. (Color online) (a) Automobile assembly line throughput time-series; (b) One-step prediction with 95% confidence interval for MGP; (c) One-step prediction with 95% confidence interval for LGP, showing that 84% of the actual realizations lie within the 95% confidence interval compared to 51% for MGP. Here, each time unit indicates a production shift (an 8 h period).

the prediction accuracy was improved on an average by about 50% compared to the next best method among the ones tested (i.e., ARMA). From the comparison of the result by LGP [Fig. 9(c)] and MGP [Fig. 9(b)], we can see that about 84% of the tested points are within the 95% confidence interval using LGP model, while for MGP model, only 51% points lie within the 95% confidence interval. This indicates that whereas both LGP and MGP are fairly accurate for cap-

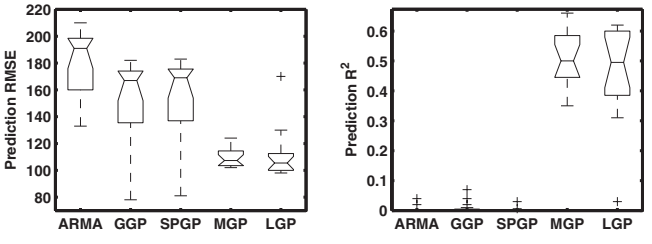

FIG. 10. Comparison of accuracies for prediction of throughputs of assembly line machines using alternative methods indicating that among all the models tested, MGP and LGP sustain the least RMSE and highest prediction R^2 values.

turing the first moment, LGP is superior to MGP in capturing the second moment, i.e., the prediction covariance.

V. CONCLUDING REMARKS

Forecasting future states and performance of nonlinear systems under nonstationarities in higher order statistics (not just the first and/or second moment) remains an open issue. Whenever system dynamics is complex and/or unknown, nonparametric approaches can be attractive, although few of the current nonlinear time-series prediction approaches employ nonparametric models. Among nonparametric approaches, Gaussian process (GP) models offer the advantage of capturing the nonlinearities without explicit specification of the underlying vector fields. Successful application of GP models has been reported various domains, including flexible robotic systems, weather forecasting, and financial markets. The salient processes in these domains are known to exhibit nonlinear dynamics. Consequently, the GP modeling assumption that the residuals as well as the random state vector are Gaussian at all times t will not hold under any nontrivial condition. However, if we relax the Gaussian assumption, one has little recourse than to use sequential Monte Carlo (e.g., MCMC, particle filter, mixture models), and such approaches for prediction. These approaches typically involve significant computational overhead. Comparatively, GP is more computationally efficient. Also, as it turns out, for complex real-world systems, especially those in the context of complex flexible robotics, weather, complex material phase transformation processes, and physiological process, the number of sources of uncertainty (noise) tend to be significantly large, especially when one attempts to capture the dynamics in a finite dimensional state space. In other words, the noise term becomes a superposition of a large number of independent random variables. Under such circumstances the Gaussian assumptions may not be too restrictive, and little advantage can accrue from resorting to elaborate PF and such computationally involved approaches suitable for predicting the evolution of the random state vector.

Also pertinently, much of the prior applications of GP were limited predictions under near-stationary conditions. The extensions of GP to prediction under nonstationary conditions have had limited success. By using certain local topological properties of nonlinear dynamics, we have extended the GP for prediction under nonstationary conditions. The approach is based on the recent results of local affine representation and recurrence properties of nonlinear systems. It allows a new means for forecasting future states of nonlinear (likely chaotic) time series under higher order nonstationarities (i.e., beyond variations in the first two moments).

Our extensive experimental investigations suggest that the recurrence-based LGP can be used to improve prediction accuracies in nonlinear systems that can be partitioned into piecewise affine segments. A numerical study involving a synthetic piecewise affine system and a nonlinear system with different forms of nonstationarity conditions shows that as few as 3 out of 33 samples, located arbitrarily close to, but outside the boundary of an affine segment can increase the prediction errors by about two orders of magnitude compared to the case with all sample points within the segment. This supports our proposition that a significant improvement in prediction accuracies is possible through careful state space segmentation, and concomitant selection of sample points for local nonparametric modeling, afforded by the recurrence-based LGP approach.

Results from the three case studies presented in the foregoing also indicate that the recurrence-based LGP yields better prediction accuracy compared to other conventional and sophisticated models. On an average, LGP reduces the prediction RMSE by about 40% over ARMA and about 17% over EKF. Also evident is that the MGP model showed comparable RMSE and R^2 values for automotive assembly line throughput prediction. But LGP outperforms MGP for physiological heart rate time series. In fact, for prediction of this time series only LGP, followed by PF and MGP yield some reasonable prediction accuracies ($R^2 > 60\%$). All other methods tested do not capture the evolution of the time-series ($R^2 < 10\%$). Furthermore, R^2 with LGP is 10% higher, and RMSE about 30% lower than with MGP. Remarkably, 93% of the actual realizations lie within the LGP-predicted confidence limits compared to $<70\%$ with MGP.

The assembly line data has high noise levels, which can pose additional challenges to recurrence-based state space segmentation. Compared with LGP, SPGP is faster but the prediction accuracies (R^2) are about 20% below that for LGP. Although MGP and LGP are comparable both in terms of prediction accuracy (R^2 and RMSE) as well as computational speed for the manufacturing assembly line data, only 51% of actual realizations lie within MGP-predicted 95% confidence limits. Comparatively, 84% of the realizations were within the LGP-predicted 95% limits. In other words, while both MGP and LGP can yield similar performance for first-moment prediction, LGP yields significantly better second-moment estimates. Additionally, the number of clusters in MGP needs to be elaborately tuned to ensure adequate prediction accuracies. In contrast, consistent criteria, such as the false nearest neighbors, mutual information, and statistical outliers exist for choosing the various LGP parameters, as stated in the foregoing. This can mitigate the need for elaborate parameter tuning in the present recurrence-based LGP approach. Overall, LGP model appears to be best suited for one-step look-ahead prediction in scenarios where the process exhibits highly nonlinear and nonstationary behavior, but the noise levels are relatively low ($<30\%$) compared to the signal energy so that some of the local topological properties can be leverage. In cases where noise levels are much higher, one can use elaborate MGP and more generic PF methods for nonlinear time-series prediction.

It may be noted that the present investigation has focused on evaluating LGP models for one-step look-ahead prediction in multiple application scenarios. Ongoing efforts are focused on adapting this approach for multistep look ahead (often referred to as free running) prediction applications. In this context, our initial investigations toward adapting LGP for free running multistep predictions suggest that the following issues are pertinent: (a) the predictions of both the first and the second moments made at the previous step need to be recursively used to update the RHS of Eq. (4), instead

of just the first-moment information alone, and (b) the prediction accuracies depend on the length of the current near-stationary segment. In order to improve the selection of local models at future times we are investigating a high-level probabilistic model based on discrete state logic (e.g., a Markov chain or a timed automaton) [41]. This model captures the likelihoods of transitions among the near-affine segments.

Furthermore, the two metrics (RMSE and R^2) used for performance comparison in the present study essentially quantify the variation in the residuals, i.e., the difference between the one-step ahead predictions and the actual realizations. Since stability issues do not arise in the one-step prediction with LGP and other methods compared, these metrics are reasonable for quantifying one step look-ahead performances. In multistep prediction scenarios, the interest may be in ensuring that certain topological characteristics of the process dynamics is preserved in the predicted time series, and the metrics need to just quantify how well the topological characteristics of the predicted time series relate to those of the attractor segments where the predictions are made. Coherence and distortion metrics (e.g., see Refs. [42–44]) may be considered to quantify the similarity of the characteristics between the predicted and the measured time series for longer prediction horizons (such as in the free-running cases). Also, we are investigating the comparison of LGP with local nonlinear parametric models, such as optimized radial basis function (RBF) (e.g., [45]) and localized intrinsic mode functions [46] for prediction applications.

ACKNOWLEDGMENTS

The authors wish to thank an anonymous referee for her/his thorough review and suggestions. The authors would also like to acknowledge the support of the National Science Foundation (Grants No. CMMI-0729552 and CMMI-0830023), and the General Motors R&D (Manufacturing Systems Research Laboratory).

APPENDIX

The system dynamics within each segment $\mathbf{U_i}, i=1,2,\ldots,\Gamma$, may be captured using piecewise affine differential equations of the form

$$\dot{x} = J_i(x - \bar{x}^i) + \varepsilon \quad x \in \mathbf{U_i} \tag{A1}$$

The system trajectories from a particular initial condition x_0 may be obtained as

$$x = \bar{x}^i + (x_0 - \bar{x}^i)\exp(J_i t) + \xi \quad x \in \mathbf{U_i}, \tag{A2}$$

where ε and ξ are the noise terms, $\bar{x}^i$ is the fixed point for segment $\mathbf{U_i}$ and x_0 is the initial state for the dynamical system trajectory. For the testing point x_* in segment $\mathbf{U}_1$, if the n training points are all located in segments $\mathbf{U}_1$,

$$\bar{f}^{U_1} = \sum_{j=1}^{n_1} w_j y_j^{U_1} + \sum_{j=n_1+1}^{n} w_j y_j^{U_1} \tag{A3}$$

Here, the RHS is divided into two parts, since we have n_1 training points in the interior of segment $\mathbf{U}_1$, and other $n-n_1$ training points are located near the boundary of segment $\mathbf{U}_1$. If $n-n_1$ training points (plausibly near the boundary) are replaced by points from other segments,

$$\bar{f}^{U_{1:\Gamma}} = \sum_{j=1}^{n_1} \tilde{w}_j y_j^{U_1} + \sum_{j=n_1+1}^{n} \tilde{w}_j y_j^{U_{2:\Gamma}}, \tag{A4}$$

where $y_j^{U_1}$ and $y_j^{U_{2:\Gamma}}$ are the observation values corresponding to the training points inside and outside segment $\mathbf{U}_1$.

Evidently, the prediction error Δ given in Eq. (9) may be expressed

$$\Delta = \bar{f}^{U_{1:\Gamma}} - \bar{f}^{U_1} = \sum_{j=n_1}^{n_1} (\tilde{w}_j y_j^{U_1} - w_j y_j^{U_1}) + \sum_{j=n_1+1}^{n} (\tilde{w}_j y_j^{U_{2:\Gamma}} - w_j y_j^{U_1}) \tag{A5}$$

where w_j and $\tilde{w}_j$ are of the form

$$w_j = [K_{1^*} \ldots K_{n_1^*} \ldots K_{n^*}] D_{\cdot j} = \sum_{i=1}^{n_1} K_{i^*} D_{ij} + \sum_{i=n_1+1}^{n} K_{i^*} D_{ij}. \tag{A6}$$

Here $D_{\cdot j}$ represents the elements in column j of the matrix D, given by

$$D = (K + \sigma_{noise}^2 I)^{-1} \tag{A7}$$

The weight w_j and $\tilde{w}_j$ depends on the covariance matrix, and thus on the hyperparameters. Points in segment $\mathbf{U}_1$ have the same hyperparameters. When the $n-n_1$ points are close to each other across the boundary then $w_j \approx \tilde{w}_j$ since the state space is continuous, and we can approximate Δ as

$$\Delta = \sum_{j=n_1+1}^{n} w_j (y_j^{U_{2:\Gamma}} - y_j^{U_1}) = \sum_{j=n_1+1}^{n} w_j \boldsymbol{\delta}_j \tag{A8}$$

Here,

$$\delta_j = [\bar{x}^i + (x_j - \bar{x}^i) e^{J_i t} + \xi_i]_1 - [\bar{x}^1 + (x_j - \bar{x}^1) e^{J_1 t} + \xi_1]_1 \tag{A9}$$

where $(\cdot)_1$ represents the first element of $(\cdot)$.

If all the training points lie in the same segment $\mathbf{U_1}$ and when the training data set size n is fairly large, RHS of Eq. (A8) will be equal to 0 under deterministic conditions, i.e., $\boldsymbol{\delta}_j \to 0$ as $n \to \infty$, and the prediction error Δ can be considerably reduced.

[1] J. M. Wang, D. J. Fleet, and A. Hertzmann, IEEE Trans. Pattern Anal. Mach. Intell. **30**, 283 (2008).
[2] T. Lang, C. Plagemann, and W. Burgard, *Robotics* (Science and Systems, Atlanta, GA, 2007).
[3] N.-T. Duy and J. Peters, in *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE Service Center, Piscataway, NJ2008).
[4] J. Gao, J. Appl. Probab. **41**, 467 (2004).
[5] C. C. Strelioff and A. W. Hübler, Phys. Rev. Lett. **96**, 044101 (2006).
[6] D. S. Broomhead and G. P. King, Physica D **20**, 217 (1986).
[7] C. M. Danforth and J. A. Yorke, Phys. Rev. Lett. **96**, 144102 (2006).
[8] M. C. Casdagli, Physica D **108**, 12 (1997).
[9] J. B. Gao, Y. Cao, L. Gu, J. G. Harris, and J. C. Principe, Phys. Lett. A **317**, 64 (2003).
[10] N. Marwan, M. Carmen Romano, M. Thiel, and J. Kurths, Phys. Rep. **438**, 237 (2007).
[11] R. Yulmetyev, P. Hänggi, and F. Gafarov, Nonlinear Phenom. Complex Syst. (Dordrecht, Neth.) **5**, 129 (2002).
[12] C. E. Rasmussen and C. K. I. Williams, *Gaussian Process for Machine Learning* (MIT Press, Cambridge, MA, 2006).
[13] M. Gibbs, Ph.D. dissertation, University of Cambridge, 1997.
[14] R. P. Adams and O. Stegle, in *Proceedings of the 25th International Conference on Machine Learning*, Helsinki, Finland (ACM, New York, 2008).
[15] J.-P. Eckmann, S. O. Kamphorst, and D. Ruelle, Europhys. Lett. **4**, 973 (1987).
[16] A. Katok and B. Hasselblatt, *Introduction to the Modern Theory of Dynamical Systems* (Cambridge University Press, Cambridge, England, 1995).
[17] T. P. Minka, Ph.D. dissertation, Massachusetts Institute of Technology, 2001.
[18] C. J. Paciorek and M. J. Schervish, in *Proceedings of the Conference on Neural Information Processing Systems (NIPS)*, 2004 (MIT Press, Cambridge, MA, 2004).
[19] C. E. Rasmussen and Z. Ghahramani, in *The 2001 Neural Information Processing Systems (NIPS) Conference* (MIT Press, Cambridge, MA, 2004).
[20] E. Snelson and Z. Ghahramani, in *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics*, San Juan, Puerto Rico (MIT Press, Cambridge, MA, 2007).
[21] J. Quinonero-Candela and C. E. Rasmussen, J. Mach. Learn. Res. **6**, 1939 (2005).
[22] G. Gregorcic and G. Lightbody, IEEE Trans. Neural Netw. **18**, 1404 (2007).
[23] E. Meeds and S. Osindero, *Advances in Neural Information Processing Systems* (MIT Press, Cambridge, MA, 2006).
[24] J. Q. Shi, R. Murray-Smith, and D. M. Titterington, Int. J. Adapt. Control Signal Process. **17**, 149 (2003).
[25] V. Tresp, in *The 2001 Neural Information Processing Systems (NIPS) Conference* (MIT Press, Cambridge, MA, 2001).
[26] A. M. Fraser and H. L. Swinney, Phys. Rev. A **33**, 1134 (1986).
[27] M. B. Kennel, R. Brown, and H. D. I. Abarbanel, Phys. Rev. A **45**, 3403 (1992).
[28] F. Takens, *Lecture Notes in Mathematics* (Springer, Berlin, 1981), Vol. 898.
[29] G. F. V. Amaral, C. Letellier, and L. A. Aguirre, Chaos **16**, 013115 (2006).
[30] H. Yang, Ph.D. dissertation, Oklahoma State University, 2009.
[31] R. E. Walpole and R. H. Myers, *Probability & Statistics for Engineers & Scientists* (Macmillan, New York, 1989).
[32] S. Calinon, F. Guenter, and A. Billard, IEEE Trans. Syst., Man, Cybern., Part B: Cybern. **37**, 286 (2007).
[33] F. Orderud, in *Proceedings of Scandinavian Conference on Simulation and Modeling* (Tapir Academic Press, Trondhem, 2005).
[34] H. Min, X. Jianhui, X. Shiguo, and Y. Fu-Liang, IEEE Trans. Signal Process. **52**, 3409 (2004).
[35] D. Gencaga, A. Ertuzun, and E. E. Kuruoglu, Digit. Signal Process. **18**, 465 (2008).
[36] C. Letellier, Chaos **17**, 023104 (2007).
[37] M. Shen and F. Shen, in *IEEE Signal Processing Workshop on Higher-Order Statistics*, edited by C. Banff, (IEEE Signal Processing Society, Baniff, Alberta, Canada, 1997).
[38] E. Ghysels and D. R. Osborn, *The Econometric Analysis of Seasonal Time Series* (Cambridge University Press, Cambridge, England, 2001).
[39] J. Xu, L. Duran, and P. Pibarot, IEEE Trans. Biomed. Eng. **47**, 1328 (2000).
[40] Z. Chen, P. C. Ivanov, K. Hu, and H. E. Stanley, Phys. Rev. E **65**, 041107 (2002).
[41] A. Subbu and A. Ray, Appl. Phys. Lett. **92**, 084107 (2008).
[42] A. Pikovsky, M. Rosenblum, and J. Kurths, *Synchronization: A Universal Concept in Nonlinear Sciences* (Cambridge University Press, Cambridge, England, 2001).
[43] J. M. Nichols, M. D. Todd, M. Seaver, and L. N. Virgin, Phys. Rev. E **67**, 016209 (2003).
[44] S. T. S. Bukkapatnam, J. M. Nichols, M. Seaver, S. T. Trickey, and M. Hunter, Struct. Health Monit. **4**, 247 (2005).
[45] A. Braga, A. C. Carvalho, T. Ludermir, M. d. Almeida, and E. Lacerda, *Modeling and Forecasting Financial Data: Techniques of Nonlinear Dynamics* (Kluwer Academic Publishers, Dordrecht, 2002).
[46] R. C. Sharpley and V. Vatchev, Constructive Approx. **24**, 17 (2006).