Data-driven estimation using an Echo-State Neural Network equipped with an Ensemble Kalman Filter

Debdipta Goswami[1], Artur Wolek[2], and Derek A. Paley[3]

Abstract— **This paper considers the problem of data-driven estimation with sparse measurements for a complex nonlinear system. While model-based nonlinear estimation methods are well known, state estimation from partial observations with unmodeled dynamics is less understood. Here we use a method for model-free estimation based on an echo-state network (ESN) where a reasonably accurate set of training data is available during the training period and some sparse measurements are obtained during the testing phase. The measurements are assimilated by an ensemble Kalman filter (EnKF) to improve the predictor's performance when compared to a free-running neural network architecture. The proposed method is applied to three systems: a low-dimensional chaotic system, a high-dimensional chaotic system, and a set of real-time traffic data.**

I. INTRODUCTION

Recently developed machine-learning techniques have become useful for solving a wide variety of problems, e.g., classification, speech recognition [1], and board games [2]. Recurrent neural networks (RNN) have been particularly useful for model-free prediction of dynamical systems. For example, an echo-state network (ESN) [3] can model chaotic systems with great effect [4], [5]. However, these prediction techniques often assume no measurements are available after training and rely instead on a free-running neural network to predict the dynamical system. But in many practical cases, a stream of sensor measurements, even if sparse and/or noisy, may be available. Such applications include fluid flow over an airfoil, atmospheric dynamics, and traffic network data.

In model-based estimation problems, the state estimate is computed in two steps. First, a forecast estimate is obtained by the motion update facilitated by the model. Then, a Bayesian measurement update incorporates the measured quantities to produce the final estimate. For a linear system with Gaussian process and measurement noise, the optimal estimator is given by the celebrated Kalman filter [6], whereas for a nonlinear system, optimal filtering is usually infinite dimensional and requires the solving a stochastic partial differential equation. To mitigate this problem, a variety of suboptimal techniques are usually employed for the state estimation of a nonlinear system, e.g., the extended Kalman filter (EKF) [7], unscented Kalman filter (UKF) [8], ensemble Kalman filter (EnKF), and particle filter. However, these methods need a dynamic model to perform the motion update of the state estimate.

On the other hand, neural-network predictors do not use a dynamic model. Instead, they utilize the measurement and state data for training, and then run freely from an initial condition to predict future states. This prediction requires a reasonably accurate initial condition and does not incorporate any subsequent measurements. The ESN-based method of *reservoir observer* [4] has been developed to utilize measurements for predicting unmeasured variables in the testing phase when all of the states are used during the training phase. The reservoir observer, however, feeds the measured states directly into the ESN and relies on the ESN's structure to assimilate them for prediction of the unmeasured states. This method also relies only on the current measurement, rather than the history of measurements, and does not take measurement noise into account. Recursive training of the ESN output weights has been developed using a Kalman filter [9], [10]. These methods enable the recursive least-square optimization of the ESN output weights as new data comes in, but they require the observation of the complete current state in order to estimate the next state.

This paper develops a data-driven sparse estimation method by combining the strength of a neural network with a nonlinear filtering algorithm. An echo-state network (ESN) is chosen as the recurrent neural engine for modeling the unknown dynamics, because it can be trained quickly with limited computational resources by cutting the computational cost of the backpropagation through time. The ESN adopts an input-output neural network with a randomly generated recurrent reservoir. Linear regression determines the output weights. An ESN with fading memory can universally model nonlinear dynamics [11], [12]. During the training phase, a full measurement of the states is typically utilized as the training data. Once the ESN is trained to reasonably model the dynamics, it is used to generate the motion update of the data-driven estimation. In spatially complex high dimensional systems, the requirement of a prohibitively large reservoir is mitigated by a parallel combination of smaller reservoirs that exploit the local nature of the spatial interactions.

Since the ESN models a nonlinear dynamical system, a nonlinear data-assimilation method is required for the measurement update. While the EKF and UKF perform well in model-based scenarios, the computation of the linearized dynamics is challenging for an ESN. The ensemble Kalman

[1]Debdipta Goswami is a Postdoctoral Research Associate in the Department of Mechanical and Aerospace Engineering, Princeton University, Princeton, NJ 08544. goswamid@princeton.edu

[2]Artur Wolek is an Assistant Professor in the Mechanical Engineering and Engineering Science Department, University of North Carolina at Charlotte, Charlotte, NC 28262. awolek@uncc.edu

[3]Derek A. Paley is the Willis H. Young Jr. Professor of Aerospace Engineering Education in the Department of Aerospace Engineering and the Institute for Systems Research, University of Maryland, College Park, MD 20742. dpaley@umd.edu

filter [13] is thus chosen for the measurement update for its strength in representing the posterior distribution of states by its sample mean and covariance. The incorporation of an ensemble Kalman filter in the feedback loop of an ESN improves measurement assimilation in comparison to reservoir observer [4], because the former accounts for the measurement noise with the help of a traditional Bayesian framework and assimilates a series of measurements over the testing phase, whereas the latter uses the current (noise-free) measurement only [4].

The contributions of this paper are (1) providing a data-driven framework for estimation of a high-dimensional complex nonlinear system when sparse noisy measurements are available during the testing phase along with sufficient data during the training phase; (2) combining the prediction power of a recurrent neural network with the traditional Bayesian measurement update model of an ensemble Kalman filter; (3) improving the estimation accuracy over time for a chaotic nonlinear system relative to prior work; and (4) application of the estimation method to a real set of mobility data in order to predict daily cycles of traffic congestion. The model-free estimation algorithm developed here has wide applications for estimation of complex dynamics from noisy observations when a reliable model is not available.

The paper is organized as follows. Section II provides a brief overview of the echo-state network (ESN). Section III presents the ensemble Kalman filtering algorithm with the ESN-based motion-update model. Section IV illustrates the applications to three different problems: two synthetic data streams generated by chaotic nonlinear systems and one real set of data traffic sensor data. Section V concludes the manuscript and discusses ongoing and future work.

II. Echo-State Networks: A Universal Predictor

Echo-state networks (ESNs) are a special kind of fixed recurrent neural network (RNN) in which a large, random, and fixed RNN is driven by the input signal. The nonlinear response signals thus induced in the neurons are then linearly combined to match a desired output signal. The random, fixed network is called a reservoir and the technique is also known as reservoir computing (RC) [11].

An ESN is composed of three principal components: a linear input layer $\mathbf{u}$ with m input nodes, a recurrent nonlinear reservoir network $\mathbf{r}$ with n neurons, and a linear output layer $\mathbf{y}$ with p output nodes. The reservoir network evolves with the following dynamics [11]

$$\mathbf{r}(t+\Delta t) = (1-\alpha)\mathbf{r}(t) + \alpha\psi(W\mathbf{r}(t) + W_{in}\mathbf{u}(t)), \quad (1)$$

where W is the $n \times n$ reservoir weight matrix, W_{in} is the $n \times m$ input weight matrix, $\mathbf{u}$ is the m-dimensional input signal, and $\mathbf{y}$ is the p-dimensional output signal. The time step Δt is chosen according to the sampling interval of the training data. The parameter $\alpha \in (0, 1]$ is called the leakage rate, which forces the reservoir to evolve more slowly as $\alpha \to 0$. The activation function ψ is usually a sigmoid function, e.g., $\tanh(\cdot)$ or a logistic function. The output is taken as a linear combination of the reservoir states [11], i.e.,

$$\mathbf{y}(t) = W_{out}\mathbf{r}(t), \quad (2)$$

where W_{out} is the $p \times n$ output weight matrix. The weights W_{in} and W are initially randomly drawn and then held fixed. The weight W_{out} is adjusted during the training process. The reservoir weight matrix W is usually kept sparse for computational efficiency.

During the training phase, the ESN is driven by an input sequence $\{\mathbf{u}(t_1), \ldots, \mathbf{u}(t_N)\}$ that yields a sequence of reservoir states $\{\mathbf{r}(t_1), \ldots, \mathbf{r}(t_N)\}$. The reservoir states are stored in a matrix $\mathbf{R} = [\mathbf{r}(t_1), \ldots, \mathbf{r}(t_N)]$. The correct outputs $\{\mathbf{y}(t_1), \ldots, \mathbf{y}(t_N)\}$, which are part of the training data, are also arranged in a matrix $\mathbf{Y} = [\mathbf{y}(t_1), \ldots, \mathbf{y}(t_N)]$. The training is carried out by a linear regression with Tikhonov regularization as follows [3]:

$$W_{out} = (\mathbf{R}\mathbf{R}^T + \beta\mathbf{I})^{-1}\mathbf{R}\mathbf{Y}, \quad (3)$$

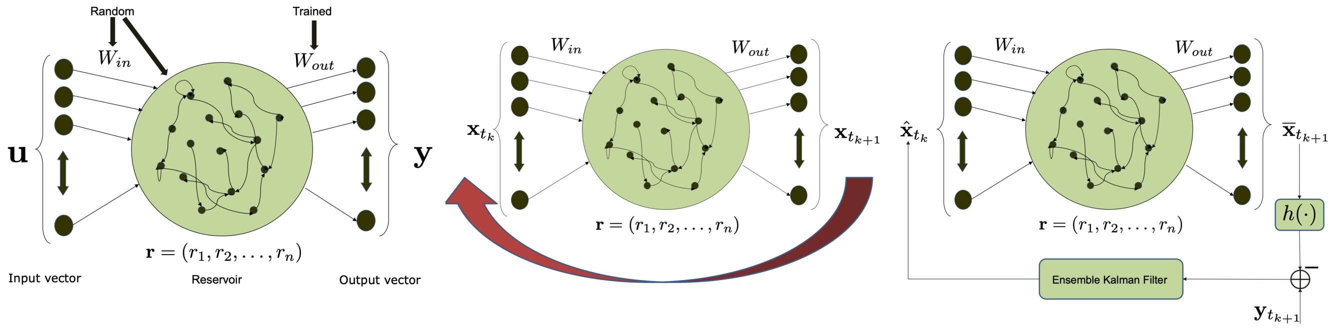
where $\beta > 0$ is a regularization parameter that ensures non-singularity.

Remark 1. For an ESN to be an universal approximator, i.e., to realize every nonlinear operator with bounded memory arbitrarily accurately, it must satisfy the echo state property (ESP) [3], which states that the reservoir will asymptotically wash out any information from the initial conditions. For the $\tanh(\cdot)$ activation function, it is empirically observed that the ESP holds for any input if the spectral radius of W is smaller than unity [3]. To ensure this condition, W is normalized by its spectral radius.

III. Kalman Filtering with Echo-State Networks: A Sparse Estimation Technique

An ESN can be trained to predict a time-series $\{\mathbf{x}_{t_i} \in \mathbb{R}^d : i \in \mathbb{N}\}$ generated by a dynamical system by setting $\mathbf{u}(t)$ and $\mathbf{y}(t)$ as the current and next state value (i.e., $\mathbf{x}_{t_k}$ and $\mathbf{x}_{t_{k+1}}$) respectively. The network is trained for a certain training length N of the time-series data $\{\mathbf{x}_{t_i}, i = 1, \ldots, N\}$, and then can run freely by feeding the output $\mathbf{y}_{t_k}$ back to the input $\mathbf{u}_{t_{k+1}}$ of the reservoir. In this case, both $\mathbf{u}$ and $\mathbf{y}$ have the same dimension d as that of the time-series data. This setup is shown in Fig. 1(b), where a trained ESN is used to predict the next states of a dynamical time series starting from an initial condition.

Although a sufficiently large free-running ESN trained with enough data can predict a dynamical system reasonably well [4], [5], it has some shortcomings. The initial input to the ESN during the free-running phase must match the exact time series data or the initial condition on the trajectory that is predicted. But for a data-driven estimation problem, the state might not be fully observable. Moreover, the free-running ESN does not take into account any change in the availability of possibly sparse observations that are available during the testing phase. A solution to this problem is presented in [4] where an ESN is trained with all state measurements available, and then predicts the states with only a limited subset of them measured. However,



(a) (b) (c)

Fig. 1: Architecture of an EnKF-ESN: (a) the basic ESN, (b) a free-running ESN for time-series prediction, and (c) an ESN with a feedback Kalman filter

this method, called a *reservoir observer*, uses the ESN's internal connections in the testing phase to assimilate the measurement of the current time step only, and does not take measurement noise into consideration.

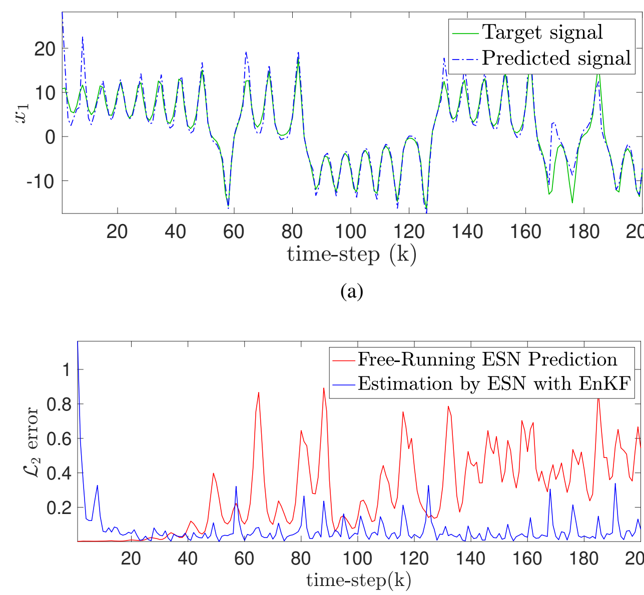
We propose an alternative method for measurement assimilation after training by adding an ensemble Kalman filter (EnKF) [14] in the feedback loop of the ESN (Fig. 1(c)). The EnKF block takes sparse observations and uses the state forecast from the reservoir output to generate a state estimate for feedback to the reservoir input. The ensemble Kalman filter is realized as follows. For time-step $k = 0$, an ensemble $\mathbf{X}_{t_0} = [\mathbf{x}_{t_0}^{(1)}, \ldots, \mathbf{x}_{t_0}^{(M)}]$ is chosen from a Gaussian distribution with an ensemble covariance R_x. Then for $k = 0, 1, \ldots,$ the following steps are computed:

$$\begin{aligned} \mathbf{X}_{t_k} &= [\mathbf{x}_{t_k}^{(1)}, \ldots, \mathbf{x}_{t_k}^{(M)}] \\ \overline{\mathbf{x}}_{t_{k+1}}^{(i)} &= W_{out}\psi(W\mathbf{r}_{t_k} + W_{in}\mathbf{x}_{t_k}^{(i)}), \text{ for } i = 1, \ldots, M \\ \mathbf{X}_{t_{k+1}}^{f} &= [\overline{\mathbf{x}}_{t_{k+1}}^{(1)}, \ldots, \overline{\mathbf{x}}_{t_{k+1}}^{(M)}] \end{aligned} \tag{4}$$

These steps carry out the motion update for the ensemble using the ESN. The superscript (i) denotes the i^{th} ensemble member. The forecast ensemble is collected in the $\mathbf{X}_{t_{k+1}}^{f}$ matrix. Next, the observations are assimilated through an ensemble Kalman filter as follows:

$$\begin{aligned} \mathcal{Y}_{t_{k+1}} &= h(\mathbf{X}_{t_{k+1}}^{f}) \\ P_{xy}(t_{k+1}) &= (\mathbf{X}_{t_{k+1}}^{f} - \overline{\mathbf{X}}_{t_{k+1}}^{f})(\mathcal{Y}_{t_{k+1}} - \overline{\mathcal{Y}}_{t_{k+1}})^T \\ P_{yy}(t_{k+1}) &= (\mathcal{Y}_{t_{k+1}} - \overline{\mathcal{Y}}_{t_{k+1}})(\mathcal{Y}_{t_{k+1}} - \overline{\mathcal{Y}}_{t_{k+1}})^T \\ K_{t_{k+1}} &= P_{xy}(t_{k+1})P_{yy}(t_{k+1})^{-1} \\ \mathbf{X}_{t_{k+1}}^{\hat{}} &= \mathbf{X}_{t_{k+1}}^{f} + K_{t_{k+1}}(\mathbf{Y}_{t_{k+1}} - \mathcal{Y}_{t_{k+1}}) \\ \hat{\mathbf{x}}_{t_{k+1}} &= \frac{1}{M}\sum_{i=1}^{M} \mathbf{x}_{t_{k+1}}^{(i)}, \end{aligned} \tag{5}$$

where $\mathbf{Y}_{t_{k+1}} = [\mathbf{y}_{t_{k+1}}, \ldots, \mathbf{y}_{t_{k+1}}]$ is a matrix constructed by stacking M copies of the true observation. This filtering step assumes Gaussian observation model with measurement map $h : \mathbb{R}^d \to \mathbb{R}^p$ and $\mathbf{y}_{t_k} = h(\mathbf{x}_{t_k}) + \nu_k$ where $\nu_k \sim \mathcal{N}(0, \Sigma_k)$ i. i. d. Gaussian noise. $P_{xy}(t_{k+1})$ denotes the sample cross-covariance between the states and the observation, whereas $P_{yy}(t_{k+1})$ denotes the sample observation covariance. The sample mean is taken as the state estimate $\hat{\mathbf{x}}_{t_{k+1}}$.



(a) (b)

Fig. 2: Estimation of the time-series using an ESN with an EnKF for the Lorenz system: (a) true and estimated signal, (b) error-comparison between a free-running ESN and an ESN with an ensemble Kalman filter (Note the EnKF does not require the initial condition to be known.)

Remark 2. An ESN with an ensemble Kalman filter for the measurement update is particularly useful for estimation problems where the state is fully observable during the training phase but only partial and noisy measurements can be obtained during the testing phase. Some of these applications include prediction of atmospheric quantities [15] and flow estimation over an airfoil [16]. Knowledge of initial condition is not required. It also improves estimation accuracy over reservoir observer [4] by assimilating all the past measurements in a Bayesian framework whereas the latter uses current measurements only.

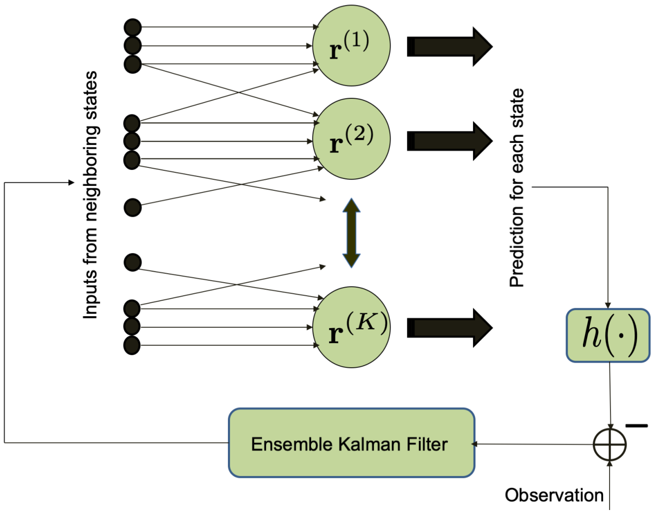
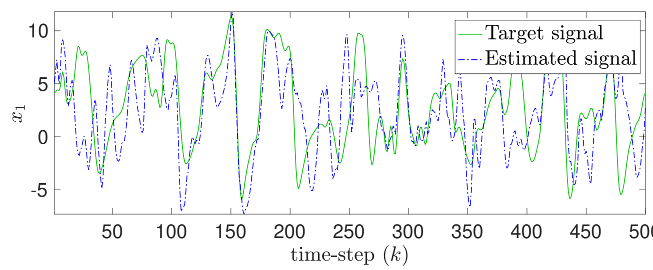
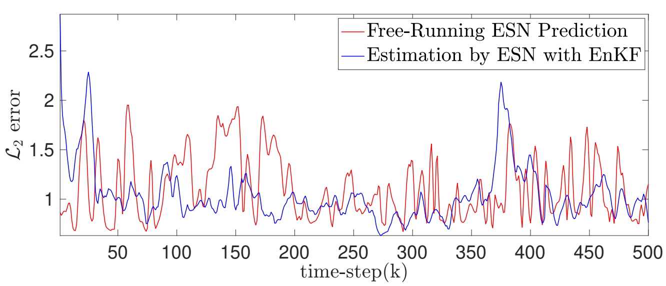


Fig. 3: The parallel ESN scheme with ensemble Kalman filter



(a)



(b)

Fig. 4: Lorenz-96 time series estimation: (a) true and estimated signal for an unmeasured state, (b) $\mathcal{L}_2$ error comparison between free-running parallel ESN with and without an Ensemble Kalman filter

IV. Examples

This section illustrates the ESN-based sparse estimation on three data-assimilation problems. The first two are time series generated by chaotic dynamical systems. The last one is a real-time series of traffic flow data obtained from the Numina sensor nodes [17] installed on the University of Maryland campus.

A. Lorenz System

We tested the ESN with the ensemble Kalman filter to estimate a time series generated by the Lorenz system:

$$
\begin{aligned}
\dot{x}_1 &= \sigma(x_2 - x_1) \qquad (6)\\
\dot{x}_2 &= x_1(\rho - x_3) - x_2
\end{aligned}
$$

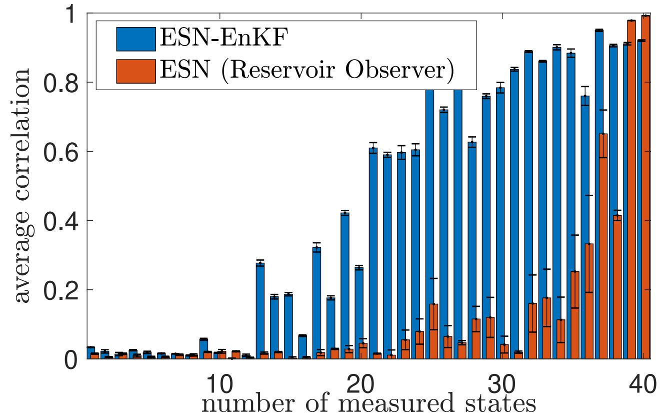


Fig. 5: Correlation between estimated data and the actual data for the Lorenz-96 model with error-bars: estimation by parallel ESNs with an ensemble Kalman filter has a higher average correlation then the prediction with parallel ESNs by a reservoir observer [4]

$$
\dot{x}_3 = x_1 x_2 - \beta x_3,
$$

where $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$ produces chaotic behavior. The time-series of $\mathbf{x}(t_k) = [x_1(t_k), x_2(t_k), x_3(t_k)]$ is available during the training phase with $\Delta t = 0.1$. An ESN with 1000 reservoir nodes is trained with data spanning 1000 time-steps. The reservoir weight matrix is constructed as the adjacency matrix of an Erdös-Rényi graph $G(n,p)$ where $n = 1000$ is the number of the reservoir nodes and $p = 0.01$ denotes the probability that an edge is present, independent of the other edges. Being the adjacency matrix of an undirected graph, W is symmetric. In the testing phase, only x_2 is observed with an i.i.d. additive zero-mean Gaussian noise of covariance 0.01. This measurement is fed into the ensemble Kalman filter with an ensemble size of 100. The reservoir nodes are initialized with random initial conditions and an initial guess of the time series is chosen. The estimated signal is compared with the true signal in Fig 2(a).

Remark 3. The introduction of the ensemble Kalman filter in the ESN feedback loop enables it to accurately estimate the time-series signal even if the initial error is large, for example if the testing phase does not start immediately after the training phase. A free-running ESN can only predict the time series with a sufficiently accurate initial condition.

The comparison of the $\mathcal{L}_2$ error between a free-running ESN predictor and the ESN-EnKF driven by sparse measurements is depicted in Fig. 2(b). Evidently, an ESN with sparse measurements performs significantly better than its free-running counterpart.

B. Lorenz-96 Model

Next, the ESN-EnKF estimation algorithm is tested with the Lorenz-96 model [18], a spatially correlated high-dimensional chaotic system developed by E. N. Lorenz in 1996 to describe the variation of an atmospheric quantity of interest, such as temperature and vorticity, at discrete locations on a periodic lattice representing a latitude circle on the earth. This model has been widely used as a model

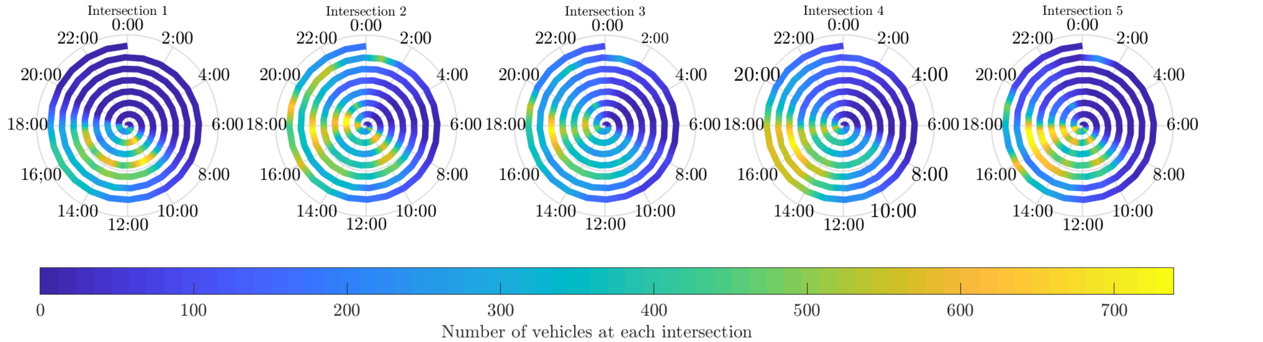


Fig. 6: Traffic congestion pattern of five intersections over a single week, Each revolution denotes a day of the week with times marked as angles; the number of vehicles is denoted by the colormap. The daily pattern of peak congestion between mornings and afternoons is evident.

for atmospheric prediction and the study of spatiotemporal chaos.

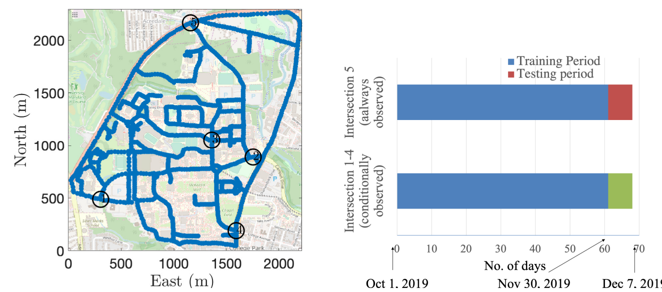
Mathematically, Lorenz-96 is a linear lattice of K variables, where the dynamics of the i^{th} variable are

$$\dot{x}_i = (x_{i+1} - x_{i-2})x_{i-1} - x_i + F, \tag{7}$$

assuming $x_{-1} = x_{K-1}$, $x_0 = x_K$, and $x_{K+1} = x_1$. The parameter F is a forcing constant with $F = 8$ being a common value causing chaotic behavior.

Let $K = 40$ be the total number of lattice points, which can be thought of as the sensors on a latitude circle that measure the atmospheric quantity of interest. For a large number of state variables the size of the ESN reservoir required to predict the system using a single reservoir must also be large. This makes the single reservoir prediction intractable for large values of K. In order to mitigate the problem, the local nature of the interactions among the state variables x_i in (7) is utilized. From (7), x_i depends only on its neighbors x_{i-2} to x_{i+1} as in [5]. A parallel set of ESNs is used, each of which predicts the state-variable x_i. ESN i takes input from the states x_{i-2} to x_{i+1} and produces the prediction for x_i. This scheme is depicted in Fig. 3.

The ESNs are trained for $N = 2000$ time steps. Each of these ESNs can be trained in parallel, thereby reducing the computation time. We have compared the performance of the proposed estimation algorithm with a free-running parallel ESN scheme. Here, a random 50% of the lattice points are assumed measurable during the Kalman filter update. The measurements are corrupted by additive i.i.d. zero mean Gaussian noise with covariance 0.01 and assimilated by an ensemble Kalman filter with an ensemble size of 100. A comparison between the estimated time-series signal and the true data for one unobserved node is shown in Fig. 4(a). The $\mathcal{L}_2$ error comparison between the free-running parallel ESN structure and the ESN-EnKF with sparse measurements is presented in Fig. 4(b). The ESN-EnKF algorithm is further compared with the reservoir observer, a model-free prediction scheme for unmeasured variables [4]. In the reservoir observer, a subset of the state variables is measured and fed into the parallel ESNs at each step, but without the measurement update step used in the ensemble Kalman filter. The average correlation between the true and estimated time series with 20 independent Monte-Carlo trials for both the proposed algorithm and the reservoir observer is depicted in Fig. 5. The proposed algorithm significantly outperforms the reservoir observer, especially when only a moderate number of states are observable.



(a) (b)

Fig. 7: Schematic diagram of traffic data training and testing: (a) University of Maryland road network with Numina sensors, (b) time frame of the data (red intersections are always observed and green intersections are conditionally observed.)

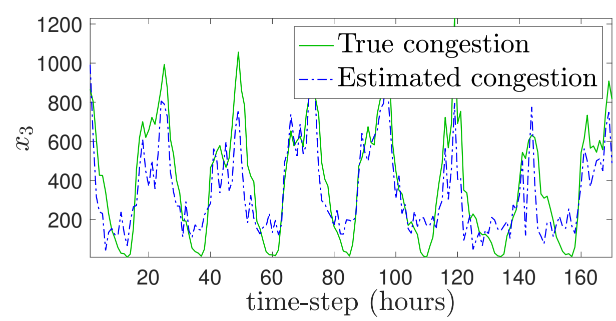


Fig. 8: Estimation of traffic congestion in the third node while only the fifth one is being observed (x_3 denotes the estimated number of vehicles at the 3$^{\text{rd}}$ intersection)

Remark 4. Lorenz-96 is an example where an ESN-based approach is greatly improved by assimilating sparse measurements through an ensemble Kalman filter. This insight has applications in atmospheric and oceanic data assimilation where the sensor measurements are sparse in time and space.

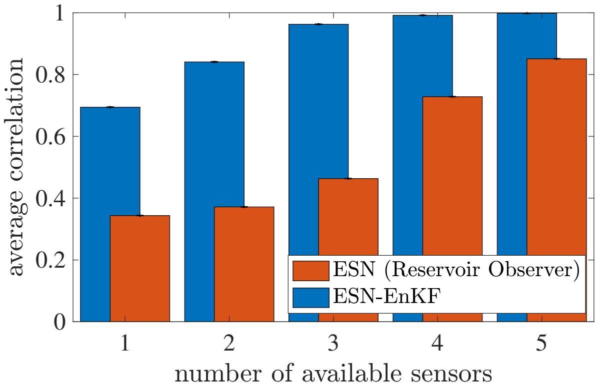


Fig. 9: Correlation between estimated and actual data for traffic congestion: estimation by ESN with an ensemble Kalman filter has a higher average correlation than the prediction with an ESN by a reservoir observer [4]

C. Prediction of Traffic Congestion in a Road Network

The proposed algorithm is now applied to a data set of traffic counts obtained from Numina sensors [17] at five different road intersections on the University of Maryland campus. Fig. 7(a) presents the road network along with the sensor locations. Each sensor counts the number of vehicles, pedestrians, and bicycles at those intersections and records them in a central server. An ESN of 1000 reservoirs are trained from hourly traffic congestion data (total number of vehicles) in all five of these intersections for two months. The training and testing timeline is presented in Fig. 7(b). Since the number of vehicles is non-negative, the activation function ψ is modified to be a rectified $\tanh$ function with the negative part set to zero. The network is then tested for a week with only one sensor active. The ensemble Kalman filter is modified to have positive ensemble members only. The estimates are rounded to the nearest positive integer. Fig. 8 shows the traffic congestion estimator's performance. The algorithm is also compared against the reservoir observer [4] with different numbers of available sensors. The average correlation between the estimated and true traffic congestion time series is computed for 20 independent Monte-Carlo trials and presented in Fig. 9 for both the proposed predictor and the reservoir observer. The Kalman-filter-driven ESN has significantly higher average correlation for partially observable cases.

Remark 5. The measurement noise for the traffic sensors is not Gaussian since the sensors can only report positive integer values, which may account for the relatively large prediction error when the congestion is low.

V. Conclusion

This paper describes a data-driven sparse estimation technique for complex dynamical systems and uses it to estimate the states of three nonlinear systems from time-series data. The method utilizes the echo-state network (ESN) for model identification from the time-series training data and an ensemble Kalman filter for data assimilation in the testing phase. The estimation is carried out in a data-driven way without a dynamic model. The method is applied to a real data set of traffic patterns on the road network of the University of Maryland, College Park campus to predict the traffic congestion at various intersections. The method is also extended to the Lorenz-96 model for atmospheric data assimilation with a parallel-reservoir ESN. In ongoing and future work, a data-driven controller design using the ESN will be investigated.

References

[1] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, Nov 2012.

[2] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[3] H. Jaeger and H. Haas, "Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication," *Science*, vol. 304, no. 5667, pp. 78–80, 2004.

[4] Z. Lu, J. Pathak, B. Hunt, M. Girvan, R. Brockett, and E. Ott, "Reservoir observers: Model-free inference of unmeasured variables in chaotic systems," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 27, no. 4, pp. 1054–1500, 2017.

[5] J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott, "Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach," *Phys. Rev. Lett.*, vol. 120, p. 024102, Jan 2018.

[6] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[7] H. K. Khalil, *Nonlinear Systems*. Upper Saddle River, (N.J.): Prentice Hall, 1996.

[8] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[9] C. Sheng, J. Zhao, H. Leung, and W. Wang, "Extended Kalman Filter based Echo State Network for time series prediction using MapReduce framework," in *2013 IEEE 9th International Conference on Mobile Ad-hoc and Sensor Networks*, 2013, pp. 175–180.

[10] X. Peng, H. Dong, and B. Zhang, "Echo State Network ship motion modeling prediction based on Kalman filter," in *2017 IEEE International Conference on Mechatronics and Automation (ICMA)*, 2017, pp. 95–100.

[11] W. Maass and H. Markram, "On the computational power of circuits of spiking neurons," *Journal of Computer and System Sciences*, vol. 69, no. 4, pp. 593 – 616, 2004.

[12] L. Grigoryeva and J.-P. Ortega, "Echo state networks are universal," *Neural Networks*, vol. 108, pp. 495 – 508, 2018.

[13] T. Bengtsson, C. Snyder, and D. Nychka, "Toward a nonlinear ensemble filter for high-dimensional systems," *Journal of Geophysical Research: Atmospheres*, vol. 108, no. D24, pp. n/a–n/a, 2003, 8775.

[14] P. L. Houtekamer and H. L. Mitchell, "Data assimilation using an ensemble Kalman filter technique," *Monthly Weather Review*, vol. 126, no. 3, pp. 796–811, 1998.

[15] E. Ott, B. R. Hunt, I. Szunyogh, A. V. Zimin, E. J. Kostelich, M. Corazza, E. Kalnay, D. J. Patil, and J. A. Yorke, "A local ensemble Kalman filter for atmospheric data assimilation," *Tellus A*, vol. 56, no. 5, pp. 415–428, 2004.

[16] D. F. Gomez, F. D. Lagor, P. B. Kirk, A. H. Lind, A. R. Jones, and D. A. Paley, "Data-driven estimation of the unsteady flowfield near an actuated airfoil," *Journal of Guidance, Control, and Dynamics*, vol. 42, no. 10, pp. 2279–2287, 2019.

[17] "Numina mobility solutions," https://numina.co, 2019.

[18] E. Lorenz, "Predictability: a problem partly solved," in *Seminar on Predictability, 4-8 September 1995*, vol. 1. Shinfield Park, Reading: ECMWF, 1995, pp. 1–18.