Data-Driven Prediction of Urban Micromobility

A STUDY OF DOCKLESS ELECTRIC SCOOTERS

DEBDIPTA GOSWAMI, ARI RIGGINS, and DEREK A. PALEY

Since the early modern age, the development of various means of transit has contributed to a dramatic improvement in quality of life. However, in past decades (particularly in urban environments), the simultaneous utilization of mobility options has resulted in poor air quality, traffic congestion, and a lack of parking. To solve these problems without imposing severe restrictions on personal mobility, alternatives to cars powered by internal combustion engines are necessary (see "Summary").

One of these alternatives is the electric-powered scooter. Unlike its conventionally powered counterparts, an e-scooter uses a battery and an electrical drivetrain and can accommodate many personal mobility needs within the city. Owing to their small size, e-scooters can significantly reduce traffic congestion and may alleviate the ever-worsening problem of urban air pollution. Shared, dockless e-scooter companies (for example, Lime and Bird) provide a convenient, sustainable mode of transportation to help commuters travel the first or last mile.

Because of their convenience, flexibility, and pleasant riding experience, e-scooters have become very popular in many U.S. cities since 2017. Lime researched transit usage in New York City and argued that the existence of transit deserts [1] and income disparity were two major causes of declining transit ridership [2]. As a partial solution, Lime proposed using micromobility, such as e-scooters, to cover intermediate distances to and from transit stops.

Since the analysis and prediction of e-scooter usage are not widely studied, rental companies record trip data and use empirical methods to distribute the vehicles throughout a city to maximize utilization. Empirical and statistical studies of e-scooter utilization are very limited. According to [3], e-scooter data from Washington, D.C. resemble those of spontaneous bike sharing for leisure activities. In [4], Mathew et al. analyzed the spatiotemporal e-scooter data from Indianapolis and identified the downtown and university campus as two e-scooter traffic hotspots. In Singapore, compact land use, the widespread availability of public transport, and better cycling infrastructure have promoted dockless e-scooters as the last-mile solution [5]. However, dockless e-scooter distributions often favor socioeconomically advantaged neighborhoods [6]–[8].

For the analysis and prediction of dockless e-scooter trips, a machine learning approach is proposed. Recently developed machine learning techniques are useful for solving a wide variety of problems, including the classification of data sets, speech recognition [9], and board games [10]. These techniques use a deep neural network to approximate an unknown function by optimizing the weights of the network layers through backpropagation [11]. Recurrent neural networks (RNNs) are particularly useful for the model-free prediction of time series data. For example, an echo-state network (ESN) [12] can model even a chaotic time series [13], [14]. A newly developed technique [15] uses a fusion of ESN and an ensemble Kalman filter (EnKF) for time series estimation with sparse measurements.

As the recurrent neural engine for modeling the time series, an ESN can be trained quickly with limited computational

Summary

E-scooters are the latest development for urban micromobility solutions. Due to their small sizes and effectiveness as a first-/last-mile solution, dockless e-scooters are increasingly popular in U.S. cities. The lack of widespread statistical studies of e-scooter mobility patterns forces many e-scooter rental companies to resort to empirical methods for deploying the vehicles. This article proposes a data-driven method to predict e-scooter mobility in urban areas. Scooter trip origins and destinations are partitioned by geographical region, and the trip origin volume is predicted as a time series using an echo-state network coupled with an ensemble Kalman filter. A Markov operator prediction technique is used to predict the spatiotemporal distribution of the scooter trip destinations. The proposed method is tested on anonymized data sets from Austin, Texas; Louisville, Kentucky; and Minneapolis, Minnesota to demonstrate its effectiveness.

Digital Object Identifier 10.1109/MCS.2022.3187328
Date of current version: 15 September 2022

1066-033X/22©2022IEEE

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 EnKF is incorporated in the feedback loop of the ESN to improve the measurement assimilation in comparison to a previous observer [13] 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 [13]. This machine learning framework, along with traditional Bayesian filtering, provides a powerful tool to predict the personal transit utilization in today's urban environment.

Shared transit vehicles, such as electric scooters, are usually aggregated into trip-level origin–destination tables between regional partitions—for example, census tracts and/or block groups—and can be modeled as a Markov process. Ulam's method [16] is one way of approximating the Markov state transition matrix from origin–destination data. However, it uses only one pair of origin–destination data snapshots and, thus, suffers from low accuracy. Constrained Ulam dynamic mode decomposition (CU-DMD) [17] utilizes a time series of data snapshots to approximate the Markov matrix. Subsequently, this matrix may be used to predict movement patterns from a given initial distribution. The Markov-based method enables the prediction of trip destinations from a time series of trip origins.

This article presents a data-driven analysis and prediction of e-scooter utilization and trip density in a city by applying an RNN and Markov operator theory. First, a city is divided into a number of geographical regions (for example, census tracts or census block groups). Then, a neural network is trained on a data set of trip origins and destinations over a number of days to predict the volume of trips originating from each region. Finally, a Perron–Frobenius (PF) operator is approximated in a stochastic matrix form to predict the trip destinations. The overall algorithm uses past data to predict trip origin and destination volumes in the future, enabling more effective and equitable operation of dockless scooters.

This article is organized as follows. First, a brief overview of the ESM algorithm with an EnKF (ESN-EnKF) and the PF approximation techniques using CU-DMD is provided. Then, the algorithms for trip origin and destination volumes using ESN-EnKF and CU-DMD are presented. The proposed algorithm is applied to data sets from Austin, Louisville, and Minneapolis to illustrate its performance. Lastly, the article is summarized, and future work is discussed.

PREDICTION ALGORITHMS FOR ORIGIN–DESTINATION DATA

This article presents a data-driven model to predict e-scooter trip origins and destinations with or without the availability of sparse observations after the training period.

Owing to their small size, e-scooters can significantly reduce traffic congestion and may alleviate the ever-worsening problem of urban air pollution.

Origin–destination data are used to train the time series prediction of trip origins via an RNN. A Markov operator-based method is then used to predict the corresponding trip destinations.

Echo-State Network With Ensemble Kalman Filter

The data-driven prediction of time series is of recent interest in the control and machine learning communities. RNNs are particularly useful for such predictions. For example, an ESN [12] can model chaotic systems effectively [13], [14]. The prediction of dockless e-scooter mobility is an important problem in micromobility that can benefit from the utilization of such techniques. This article uses an ESN with an EnKF to predict e-scooter trip origin volumes in a city. The destination prediction is treated in the next section.

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 [18]:

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

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, for example, $\tanh(\cdot)$ or a logistic function. The output is taken as a linear combination of the reservoir states [18], that is,

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

where W_{out} is the $p \times n$ output weight matrix. The input W_{in} and reservoir W weights 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. See "Echo-State Network" for more information on ESNs. The predictive skill of the ESN

Echo-State Network

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 [18]. An ESN consists of 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 in a nonlinear fashion [18]:

$$\mathbf{r}(t+\Delta t) = (1-\alpha)\,\mathbf{r}(t) + \alpha\psi\,(W\mathbf{r}(t) + W_{\text{in}}\,\mathbf{u}(t)), \tag{S1}$$

where $W \in \mathbb{R}^{n\times n}$ is the reservoir weight matrix, $W_{\text{in}} \in \mathbb{R}^{n\times m}$ is the input weight matrix, $\mathbf{u}$ is the m-dimensional input signal, and $\mathbf{y}$ is the p-dimensional output signal. The time step Δt denotes the sampling interval of the time series data. The leakage rate $\alpha \in (0, 1]$ forces the reservoir to evolve more slowly as $\alpha \to 0$. The activation function that incorporates the nonlinearity ψ is usually a sigmoid function, for example, $\tanh(\cdot)$ or a logistic function. The network output is a linear combination of the reservoir states [18]; that is,

$$\mathbf{y}(t) = W_{\text{out}}\,\mathbf{r}(t), \tag{S2}$$

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 Wout 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 implemented by a linear regression with Tikhonov regularization as [12]

$$W_{\text{out}} = (\mathbf{R}\mathbf{R}^T + \beta\mathbf{I})^{-1}\mathbf{R}\mathbf{Y}, \tag{S3}$$

where $\beta > 0$ is a regularization parameter that ensures nonsingularity. For an ESN to be a universal approximator (that is, to realize every nonlinear operator with bounded memory arbitrarily accurately), it must satisfy the echo-state property (ESP) [12], 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 [12]. To ensure this condition is met, the randomly generated W is normalized by its spectral radius.

An ESN is advantageous over other kinds of RNNs when a cheap, fast, and adaptive training is required since its training does not require backpropagation through time. An ESN can be trained to predict a time series $\{\mathbf{x}_{ti} \in \mathbb{R}^d : i \in \mathbb{N}\}$ generated by a dynamical system by setting $\mathbf{u}(t)$ and $\mathbf{y}(t)$ as the current and next state values (that is, $\mathbf{x}_{tk}$ and $\mathbf{x}_{tk+1}$), respectively. The network is trained for a certain training length N of the time series data $\{\mathbf{x}_{ti}, i = 1, \ldots, N\}$ and then can run freely by feeding the output y_{tk} back to the input u_{tk+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 Figure S1(b), where a trained ESN is used to predict the next states of a dynamical time series starting from an initial condition.

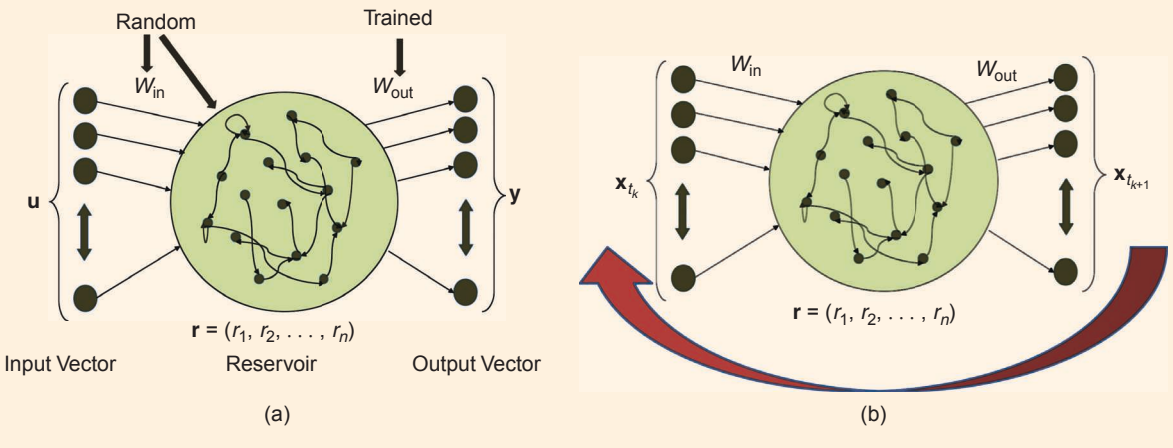

FIGURE S1 The architecture of an echo-state network (ESN) as a time series predictor: (a) the basic ESN and (b) a free-running ESN for time series prediction.

is refined by assimilating any available sparse and/or noisy measurements after the conclusion of the training phase using an EnKF [19], as shown in Figure 1. 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 [15].

The EnKF 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:

$$\begin{aligned} \mathbf{X}_{t_k} &= [\mathbf{x}_{t_k}^{(1)}, \ldots, \mathbf{x}_{t_k}^{(M)}] \\ \bar{\mathbf{x}}_{t_{k+1}}^{(i)} &= W_{\text{out}}\psi(W\mathbf{r}_{t_k} + W_{\text{in}}\mathbf{x}_{t_k}^{(i)}), \quad \text{for } i = 1, \ldots, M \\ \mathbf{X}_{t_{k+1}}^{f} &= [\bar{\mathbf{x}}_{t_{k+1}}^{(1)}, \ldots, \bar{\mathbf{x}}_{t_{k+1}}^{(M)}]. \end{aligned} \tag{3}$$

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

$$\begin{aligned} \mathcal{Y}_{t_{k+1}} &= h(\mathbf{X}_{t_{k+1}}^{f}) \\ P_{xy}(t_{k+1}) &= (\mathbf{X}_{t_{k+1}}^{f} - \bar{\mathbf{X}}_{t_{k+1}}^{f})(\mathcal{Y}_{t_{k+1}} - \bar{\mathcal{Y}}_{t_{k+1}})^T \\ P_{yy}(t_{k+1}) &= (\mathcal{Y}_{t_{k+1}} - \bar{\mathcal{Y}}_{t_{k+1}})(\mathcal{Y}_{t_{k+1}} - \bar{\mathcal{Y}}_{t_{k+1}})^T \\ K_{t_{k+1}} &= P_{xy}(t_{k+1})P_{yy}(t_{k+1})^{-1} \\ \hat{\mathbf{X}}_{t_{k+1}} &= \mathbf{X}_{t_{k+1}}^{f} + K_{t_{k+1}}(\mathbf{Y}_{t_{k+1}} - \mathcal{Y}_{t_{k+1}}) \\ \hat{\mathbf{x}}_{t_{k+1}} &= \frac{1}{M}\sum_{i=1}^{M} \mathbf{x}_{t_{k+1}}^{(i)}, \end{aligned} \tag{4}$$

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. $P_{xy}(t_{k+1})$ denotes the sample cross covariance between the states and the observation, whereas $P_{yy}(t_{k+1})$ denotes the sample observation covariance for the ensemble. The sample mean is taken as the state estimate $\hat{\mathbf{x}}_{t_{k+1}}$. For more information on EnKFs, see "Ensemble Kalman Filter."

Markov Operator Approximation Using Constrained Ulam Dynamic Mode Decomposition

This article constructs a finite-dimensional approximation of the PF transfer operator to analyze e-scooter mobility in a city. This model predicts future movement patterns based on training data. The transfer-operator approximation is a finite-dimensional matrix that characterizes the dynamics of a complex system using a data-driven approach [17]. Assume the movement of each scooter in a population is a stochastic dynamical system that permits modeling the evolution of its probability density in a linear Markov fashion. The PF operator is utilized to characterize the mobility patterns by treating them as the density of states in a dynamical system. However, the PF operator operates on the space of L_1 densities and must be approximated on a finite-dimensional basis. This approximation generates a Markov state transition matrix with the spatial partitions as its states. The approximation is carried out in a data-driven fashion using CU-DMD [17]. CU-DMD combines the data-driven extended dynamic mode decomposition [20], [21] (see "Extended Dynamic Mode Decomposition") algorithm with Ulam's method to better approximate the PF operator through a constrained least-squares optimization.

CU-DMD provides an improvement over Ulam's method [16] for the computation of the Markov state transition matrix. Ulam's method uses a one-pass Monte Carlo approach to numerically estimate the Markov state transition matrix P_τ. Within each partition B_i of the state space, a set of N initial points $x_{i,1}, \ldots, x_{i,N}$ is defined and propagated using the system dynamics to obtain $\phi_f(\tau, x_{i,k})$, $k = 1, \ldots, n$, that is, their final positions. The estimated P_τ is [16]

$$P_{\tau,ij} \approx \frac{\#\{k : x_{i,k} \in B_i, \phi_f(\tau, x_{i,k}) \in B_j\}}{N}. \tag{5}$$

The choice of τ is important and depends on the resolution of the partition $D = B_i$, $i = 1, \ldots, M$ in this method. If the resolution is coarse [that is, there are too few grid cells (M is small), and τ is also small], then many of the test points will not leave their original grid cell B_i, and the estimated P_τ will be close to the identity matrix. In the context of e-scooter mobility, the census block groups or other regional blocks define the partition D, and τ is defined by the time step of the users' location data.

CU-DMD improves the accuracy of Ulam's method using data for multiple time intervals, starting from N initial points within each partition B_i, $i = 1 \ldots, M$. Let $\mathbf{p}_k \in \mathbb{R}^M$ denote the vector of the particle volumes in M partitions in the kth time interval and $\{\mathbf{p}_0, \ldots, \mathbf{p}_{n+1}\}$ be the particle volumes for $n + 2$ subsequent time intervals. The linearity of the PF operator yields $\mathbf{p}_{k+1} = \mathbf{p}_k P_\tau$. Define

$$\Psi_0 = [\mathbf{p}_0^T, \ldots, \mathbf{p}_n^T]^T, \quad \Psi_1 = [\mathbf{p}_1^T, \ldots, \mathbf{p}_{n+1}^T]^T. \tag{6}$$

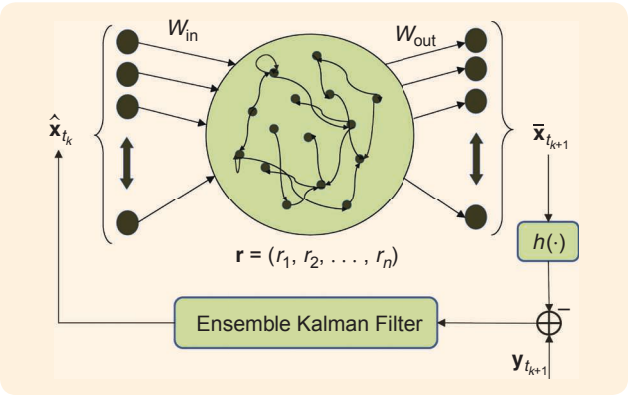

FIGURE 1 The architecture of an echo-state network with an ensemble Kalman filter.

Ensemble Kalman Filter

The ensemble Kalman filter (EnKF) is a Monte Carlo implementation of the traditional Kalman filter where the covariance matrix is replaced with the sample covariance of an ensemble realization of the state. It can be readily extended to nonlinear dynamical systems with a large number of states and/or complex state equations where an extended Kalman filter (EKF) becomes infeasible. An EKF uses an ensemble of state vector realizations at time t_k,

$$\mathbf{X}_{t_k} = [\mathbf{x}_{t_k}^{(1)}, \ldots, \mathbf{x}_{t_k}^{(M)}],$$

which is then used as the samples of the prior distribution at time t_k. However, the ensemble members are not, in general, independent except in the initial ensemble since every EnKF step ties them together. They are deemed to be approximately independent, and all calculations proceed as though they actually are independent. Let the dynamics be

$$\begin{aligned} \mathbf{x}_{t_{k+1}} &= f(\mathbf{x}_{t_k}) + \mathbf{w}_k \\ \mathbf{y}_{t_k} &= h(\mathbf{x}_{t_k}) + \mathbf{v}_k, \end{aligned} \tag{S4}$$

where $\mathbf{w}_k$ and $\mathbf{v}_k$ are independent zero-mean Gaussian noise. For $k = 0$, an ensemble realization $\mathbf{X}_{t_0} = [\mathbf{x}_{t_0}^{(1)}, \ldots, \mathbf{x}_{t_0}^{(M)}]$ is chosen from a Gaussian distribution with an ensemble covariance R_x. For each time step, the forecast step is computed as

$$\begin{aligned} \mathbf{X}_{t_k} &= [\mathbf{x}_{t_k}^{(1)}, \ldots, \mathbf{x}_{t_k}^{(M)}] \\ \tilde{\mathbf{x}}_{t_{k+1}}^{(i)} &= f(\mathbf{x}_{t_k}^{(i)}), \text{ for } i = 1, \ldots, M \\ \mathbf{X}_{t_{k+1}}^{f} &= [\tilde{\mathbf{x}}_{t_{k+1}}^{(1)}, \ldots, \tilde{\mathbf{x}}_{t_{k+1}}^{(M)}]. \end{aligned}$$

Next, the measurement is assimilated using the sample covariances P_{xy} and P_{yy} as

$$\begin{aligned} \mathcal{Y}_{t_{k+1}} &= h(\mathbf{X}_{t_{k+1}}^{f}) \\ P_{xy}(t_{k+1}) &= (\mathbf{X}_{t_{k+1}}^{f} - \bar{\mathbf{X}}_{t_{k+1}}^{f})(\mathcal{Y}_{t_{k+1}} - \bar{\mathcal{Y}}_{t_{k+1}})^T \\ P_{yy}(t_{k+1}) &= (\mathcal{Y}_{t_{k+1}} - \bar{\mathcal{Y}}_{t_{k+1}})(\mathcal{Y}_{t_{k+1}} - \bar{\mathcal{Y}}_{t_{k+1}})^T \\ K_{t_{k+1}} &= P_{xy}(t_{k+1})\, P_{yy}(t_{k+1})^{-1} \\ \hat{\mathbf{X}}_{t_{k+1}} &= \mathbf{X}_{t_{k+1}}^{f} + K_{t_{k+1}}(\mathbf{Y}_{t_{k+1}} - \mathcal{Y}_{t_{k+1}}) \\ \hat{\mathbf{x}}_{t_{k+1}} &= \frac{1}{M}\sum_{i=1}^{M} \mathbf{x}_{t_{k+1}}^{(i)}, \end{aligned} \tag{S5}$$

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. EnKF relies on the Gaussian assumption, although in practice it is used for nonlinear problems where the Gaussian assumption may not be satisfied. Related filters attempting to relax the Gaussian assumption in EnKF while preserving its advantages include filters that fit the state probability density function (PDF) with multiple Gaussian kernels [S1], filters that approximate the state PDF by Gaussian mixtures [S2], [S3], a variant of the particle filter with the computation of particle weights by density estimation, and a variant of the particle filter with a thick-tailed data PDF to alleviate particle filter degeneracy [S4].

REFERENCES

[S1] J. L. Anderson and S. L. Anderson, "A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts," *Monthly Weather Rev.,* vol. 127, no. 12, pp. 2741–2758, Dec. 1999, doi: 10.1175/1520-0493(1999)127<2741:AMCIOT>2.0.CO;2.

[S2] D. Goswami and D. A. Paley, "Non-gaussian estimation and observer-based feedback using the Gaussian Mixture Kalman and Extended Kalman Filters," in *Proc. 2017 Amer. Control Conf. (ACC),* pp. 4550–4555, doi: 10.23919/ACC.2017.7963657.

[S3] D. Goswami and D. A. Paley, "Non-gaussian estimation and dynamic output feedback using the Gaussian Mixture Kalman Filter," *J. Guid., Control, Dyn.*, vol. 44, no. 1, pp. 15–24, 2021, doi: 10.2514/1.G005005.

[S4] P. J. van Leeuwen, "A variance-minimizing filter for large-scale applications," *Monthly Weather Rev.,* vol. 131, no. 9, p. 2071, Jan. 2003, doi: 10.1175/1520-0493(2003)131<2071:AVFFLA>2.0.CO;2.

Therefore, from the Markov relation ($\Psi_1 = \Psi_0 P_\tau$),

$$P_\tau \approx \Psi_1 \Psi_0^\dagger. \tag{7}$$

Since P_τ must satisfy the positivity and stochastic properties of a Markov operator, the unconstrained least-squares solution is modified to a constrained least-squares formulation [17]:

$$\begin{aligned} \underset{P_\tau}{\text{minimize}} \quad & \|\Psi_1 - \Psi_0 P_\tau\|_F \\ \text{subject to} \quad & P_{\tau,ij} \geq 0, i, j \in \{1, \ldots, M\} \\ & \sum_{j=1}^{M} P_{\tau,ij} = 1, i \in \{1, \ldots, M\}. \end{aligned} \tag{8}$$

Because this formulation is a convex quadratic programming problem, it yields a unique minimum and can be solved using gradient-descent or interior-point methods [22].

PREDICTION OF E-SCOOTER TRIPS

This section describes the proposed data-driven method for the prediction of e-scooter trip volumes in a city using the neural network and Markov operator theory described previously.

Trip Origin Prediction Using Echo-State Network With Ensemble Kalman Filter

Consider a data set of scooter trips in which each trip is identified by its start time, origin and destination coordinates, and duration. These data are preprocessed into a series of origin–destination tables, where each entry denotes the number of trips made from one geographical region to another during a specific time interval. The geographical regions and time intervals may be chosen according to the desired spatial and temporal resolution. This article uses census tracts or block groups to partition a city into regions and 24-h time intervals.

Extended Dynamic Mode Decomposition

Extended dynamic mode decomposition (EDMD) [21] is a method to extract the modes of a complex dynamical system by solving a least-squares problem. Consider the autonomous time-invariant ordinary differential equation (ODE)

$$\dot{x} = f(x). \tag{S6}$$

Let $\phi_f : \mathbb{R} \times \mathbb{X} \to \mathbb{X}$ be the flow map of the ODE (S6); that is, $\phi_f(t, x_0)$ is a solution of the ODE (S6) with the initial condition $x(0) = x_0$. EDMD estimates the eigenvalues and eigenfunctions of the Koopman operator [S5], the dual of the Perron–Frobenius (PF) operator, which operates on the space of L^∞ observables. The Koopman semigroup of operator $\mathcal{K}^t : L^\infty(\mathbb{X}) \to L^\infty(\mathbb{X})$ is defined as

$$(\mathcal{K}^t \varphi)(\cdot) = \varphi \circ \phi_f(t, \cdot). \tag{S7}$$

If the time step $t = \tau$ is fixed, then the ODE (S6) becomes an iterative map $x((k+1)\tau) = \phi_f(\tau, x(k\tau))$, and τ can be dropped and $\phi_f(\tau, x) \triangleq F(x)$ defined. The discrete-time dynamics are

$$x_{k+1} = F(x_k). \tag{S8}$$

The time-discretized version of the Koopman operator is $(\mathcal{K}^t \varphi)(\cdot) = \varphi \circ F(\cdot)$. In EDMD, just like Ulam's method, the infinite-dimensional operator $\mathcal{K}^\tau$ is projected onto a finite-dimensional basis in $L^\infty(\mathbb{X})$ to represent it as a matrix K. Let $\{\psi_1, \ldots, \psi_M\}$ be the basis functions, and, as in Ulam's method, define $\pi_M : L^\infty(\mathbb{X}) \to \text{sp}\{\psi_i, \ldots, \psi_M\}$ to project φ onto the span of these basis functions. Then,

$$\varphi(x) = \sum_{i=1}^{M} a_i \psi_i(x) \tag{S9}$$

$$\mathcal{K}^\tau \varphi(x) = \sum_{i=1}^{M} b_i \psi_i(x) + r, \tag{S10}$$

with residue r. Now, observe $\{x_0, \ldots, x_{n+1}\}$ for any $n > 0$, where x_i are from the discretized dynamics (S8). The matrix K can be estimated by the least-squares formulation

$$K = \Psi_{x_0}^{\dagger} \Psi_{x_1}, \tag{S11}$$

where $\Psi_{x_0,ij} = \psi_i(x_j)$ and $\Psi_{x_1,ij} = \psi_i(x_{j+1}), i = 1, \ldots, M$, and $j = 0, \ldots, n$.

In the same light, the weak approximation of the PF operator can be thought of as projecting onto the basis function $\psi_i = 1/m(B_i)\chi_{B_i}$, where

$$\chi_{B_i}(x) = \begin{cases} 1, & \text{if } x \in B_i \\ 0, & \text{otherwise} \end{cases}$$

for partition B_i. Since the basis functions are related to the density of states and cannot be readily observed, the origin–destination data approximate them.

REFERENCE

[S5] B. O. Koopman, "Hamiltonian systems and transformation in Hilbert space," *Proc. Nat. Acad. Sci.*, vol. 17, no. 5, pp. 315–318, 1931, doi: 10.1073/pnas.17.5.315.

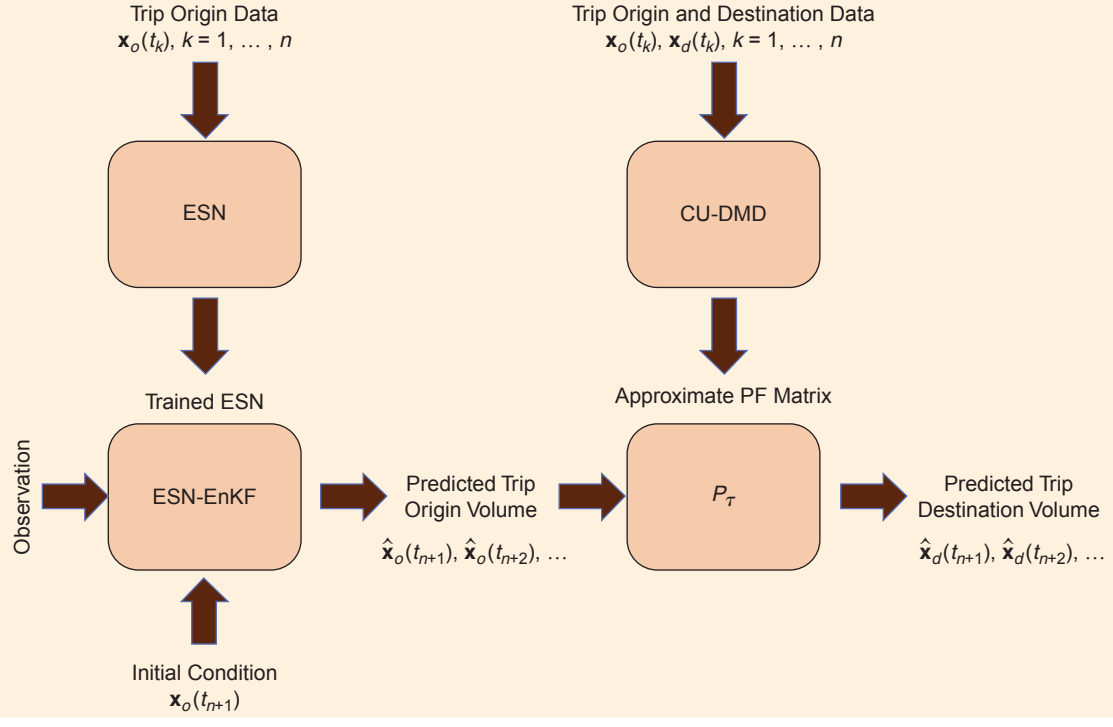

FIGURE 2 A flowchart of the trip origin and destination prediction method.

If the city is divided into M geographical regions, let $\mathbf{x}_o(t_k) \in \mathbb{R}^M$ denote the number of trips originating from M regions on the kth day. The ESN-EnKF is trained on the prior data available to predict the trip origin volumes. The prediction can be carried out in two specific ways. A prediction can be made with a free-running ESN without any Kalman filter, or the multiday prediction accuracy can be improved by observing some of the regions during the testing/prediction phase and using them as observations in the EnKF module.

Trip Destination Prediction Using Markov Operator Theory

A finite-dimensional approximation of the PF operator computed by CU-DMD is used to predict the trip destination

TABLE 1 The echo-state network with ensemble Kalman filter parameters for the e-scooter case studies.

Parameter Name	Symbol	Value
Leaking rate	α	0.8
Activation	$\psi(\cdot)$	$0.5(1+\tanh(\cdot))$
Reservoir size	n	2000
Regularization factor	β	10^{-5}
Initial ensemble covariance	σ_0^2	100
Measurement noise covariance	σ^2	100

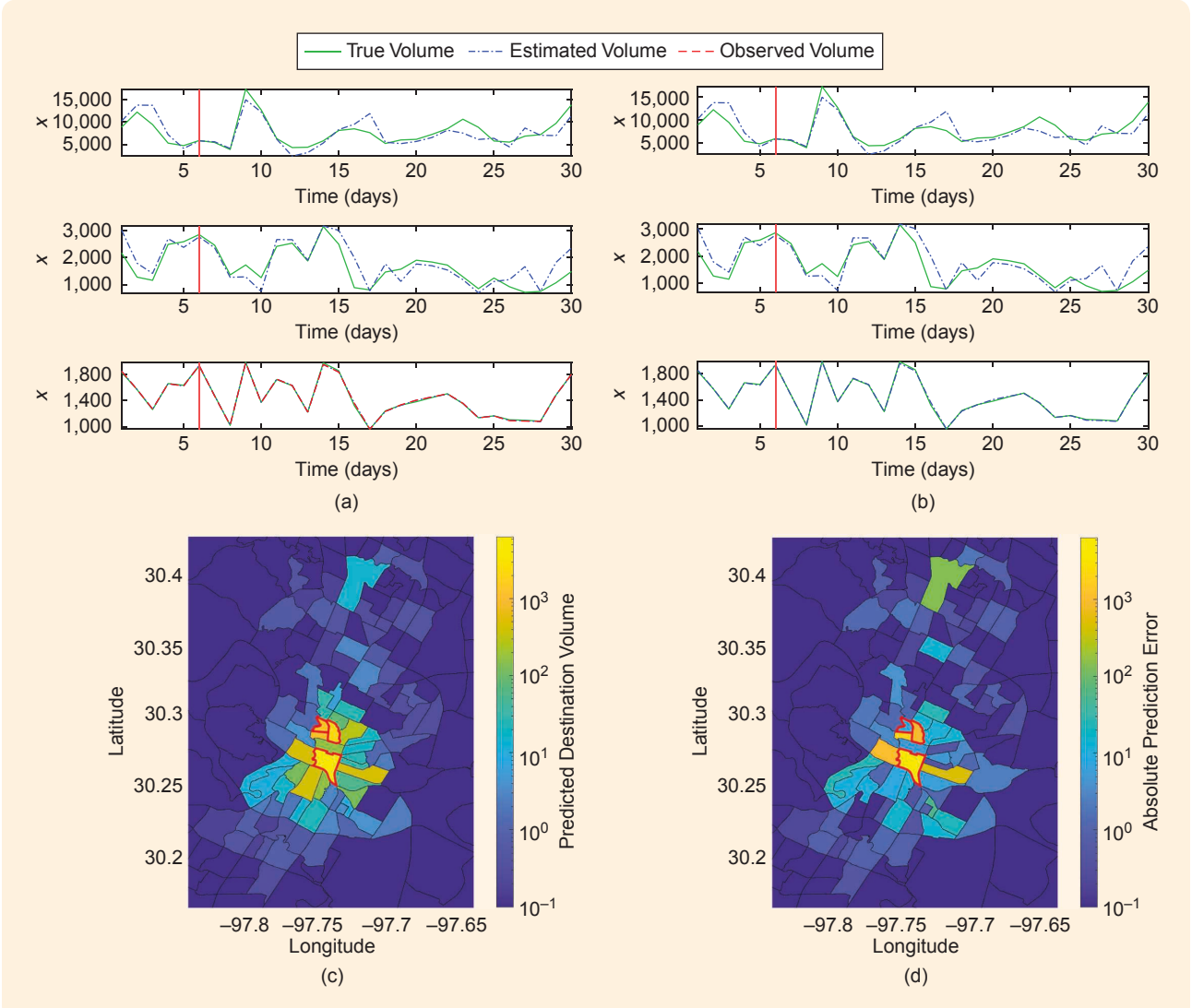

FIGURE 3 The prediction of e-scooter trips in Austin, Texas: the (a) trip origination and (b) trip destination volume prediction for three census tracts, where the trip origination volume for the third tract is measured with added noise; (c) the estimated termination volumes on day 6 of the testing phase; and (d) the absolute error in the termination volume estimation. The three focal census tracts from (a) and (b) are highlighted in red.

volumes. The transfer-operator approximation is a finite-dimensional matrix that characterizes the dynamics of a complex system using a data-driven approach [17]. The dimensions of the matrix approximation are determined by the number of starting/ending regions. By counting the number of trips that transition from one region to another for all pairs of start/end regions and applying the constraint that every trip must end in one of the regions (including possibly the starting one), construct a map P_τ that, once normalized, can be used to predict the future mobility over time span τ. Specifically, P_τ operating on an initial distribution $\mathbf{x}_o(t_k)$ of trips origins at time t_k generates a future distribution $\mathbf{x}_d(t_k)$ of trip destinations.

The origin–destination tables described in the previous section are used to generate the Markov operator, assuming the movement of each individual in a population is a stochastic dynamical system that permits modeling the evolution of its probability density in a linear Markov fashion. The PF operator characterizes the city-level mobility patterns by treating them as the density of states in a dynamical system. To facilitate the use of CU-DMD, the origin–destination trip volume data are preprocessed and sorted into a table for each day. The CU-DMD algorithm is then used to find the Markov state transition matrix P_τ for time interval τ. In the context of e-scooter mobility, the regional blocks define the partition D, and τ is defined by the time step of the location data. Let $\mathbf{x}_o(t_k)$ denote the vector of the trip origin volume in M regions in the kth time instant and $\mathbf{x}_d(t_k)$ be the trip destination volume vector. The Markov assumption on the user mobility yields $\mathbf{x}_d(t_k) = \mathbf{x}_o(t_k)P_\tau$. Define

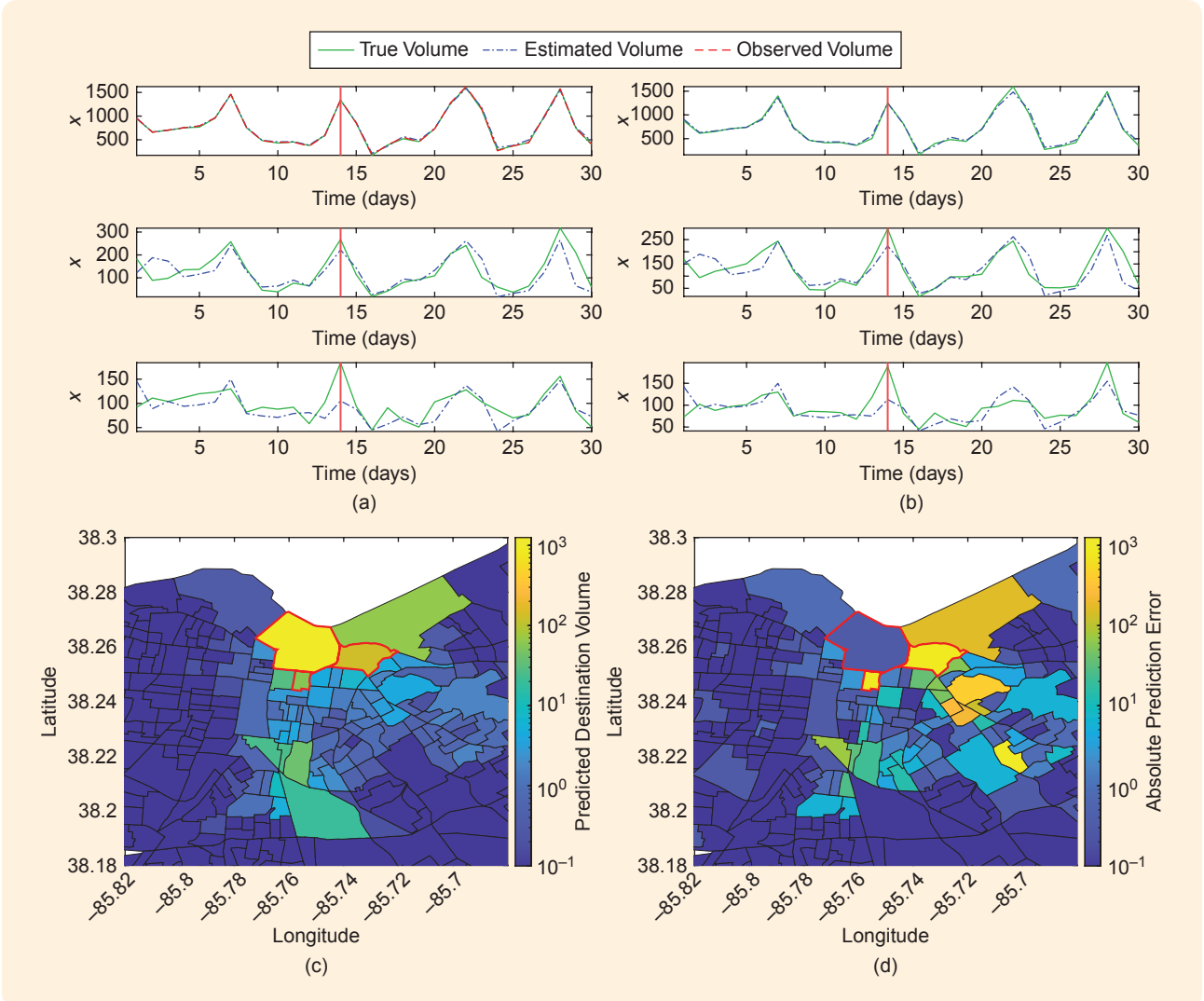

FIGURE 4 The prediction of e-scooter trips in Louisville, Kentucky: the (a) trip origination and (b) trip destination volume prediction for three census block groups, where the trip origination volume for the first block group is measured with added noise; (c) the estimated termination volumes on day 14 of the testing phase; and (d) the absolute error in the termination volume estimation. The three focal census block groups from (a) and (b) are highlighted in red.

$$\Psi_0 = [\mathbf{x}_o(t_1)^T, \ldots, \mathbf{x}_o(t_n)^T]^T$$

and

$$\Psi_1 = [\mathbf{x}_d(t_1)^T, \ldots, [\mathbf{x}_d(t_n)^T]^T.$$

The constrained least-squares problem (8) approximates the PF operator P_τ that is used to predict the mobility/flow of e-scooters at the desired spatial resolution (for example, by census tracts or census block groups). A flowchart of the proposed method to predict scooter mobility is presented in Figure 2.

EXAMPLES: AUSTIN, LOUISVILLE, AND MINNEAPOLIS SCOOTER USAGE

The proposed method is illustrated on publicly available e-scooter data sets from Austin, Texas; Louisville, Kentucky; and Minneapolis, Minnesota from 2018 to 2019. The cities are divided into regional divisions, and the numbers of e-scooter trips originating and terminating in each of them are organized in a origin–destination table for each day. Census tracts are used as the divisions for Austin, whereas census block groups are used for Louisville and Minneapolis, based on the spatial resolution of the data. The parameters used for training the ESN are provided in Table 1. Figure 3 shows the ESN-EnKF prediction framework for e-scooter trips in Austin.

The training data for this example are one year of daily trip origins partitioned by census tracts; the testing phase was one month with sparse measurements from only 30% of the tracts. The observed tracts are chosen randomly with a uniform probability. The estimated volume of trip origins tracks the true volume during the 30-day testing phase, even in census tracts without observations. Both the temporal and spatial results are provided along with the absolute prediction error for a time snapshot.

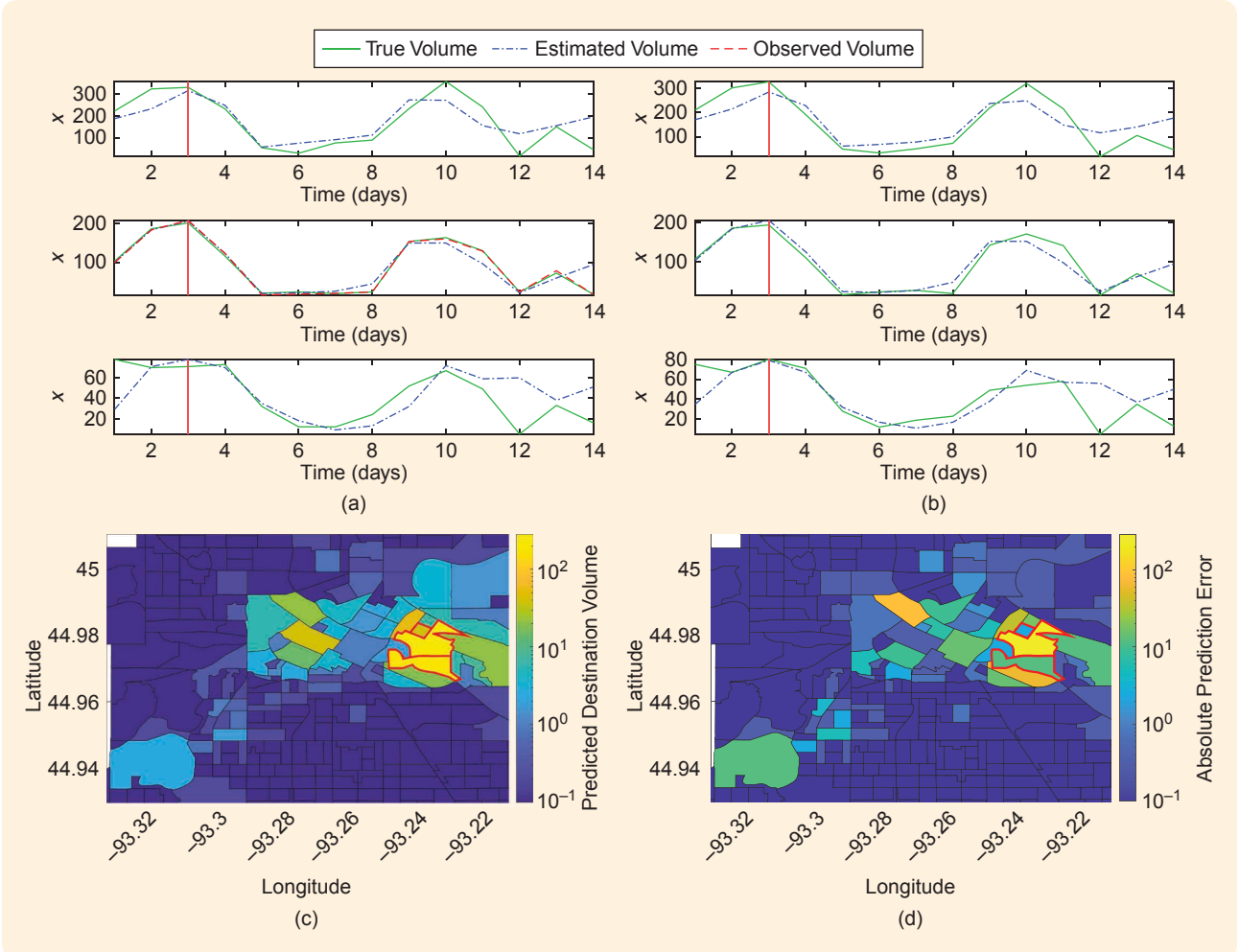

FIGURE 5 The prediction of e-scooter trips in Minneapolis, Minnesota: the (a) trip origination and (b) trip destination volume prediction for three census block groups, where the trip origination volume for the second block group is measured with added noise; (c) the estimated termination volumes on day 3 of the testing phase; and (d) the absolute error in the termination volume estimation. The three focal census block groups from figures (a) and (b) are highlighted in red.

Figures 4 and 5 show the estimation results for Louisville and Minneapolis with census block groups as the regional divisions. Training periods are one year and 130 days for Louisville and Minneapolis, respectively. Data for one entire year from the Minneapolis database were not available. The testing phases last for one month and two weeks, respectively. The other parameters are the same as for the Austin case study.

The ESN-EnKF method is compared to a reservoir observer [13] that uses sparse observations to predict the trip volumes without an EnKF. The reservoir observer uses the true values of the trip volumes in a day and feeds them directly to the ESN to generate a prediction. The results are also compared with a standard autoregressive moving-average (ARMA) regression that does not use any measurement in the testing phase. The Pearson correlation and the normalized root-mean-square error (RMSE) between the original and predicted time series of the trip origin volumes are computed for both the ESN-EnKF and reservoir-observer methods with different numbers of observable census tracts/block groups. The Pearson correlation coefficient is expressed as

$$r(\mathbf{x}, \hat{\mathbf{x}}) = \frac{\sum_{i,t}(\mathbf{x}_{i,t} - \bar{\mathbf{x}})(\hat{\mathbf{x}}_{i,t} - \bar{\hat{\mathbf{x}}})}{\sqrt{\sum_{i,t}(\mathbf{x}_{i,t} - \bar{\mathbf{x}})^2}\sqrt{\sum(\hat{\mathbf{x}}_{i,t} - \bar{\hat{\mathbf{x}}})^2}}, \tag{9}$$

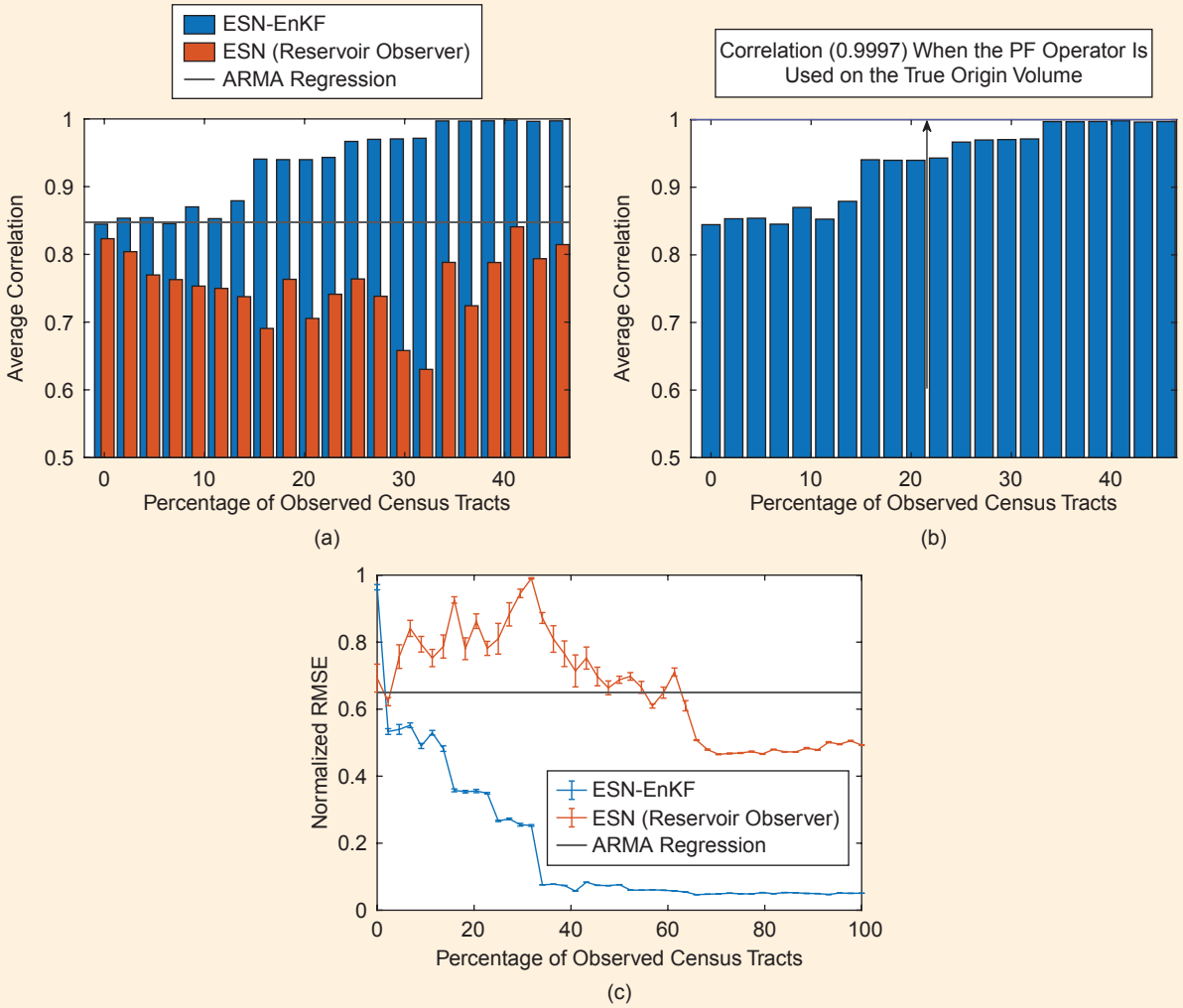

FIGURE 6 (a) The Pearson correlation between the true and estimated trip origin volumes for the Austin data: the echo-state network algorithm with an ensemble Kalman filter (ESN-EnKF) is compared with the reservoir observer and a standard autoregressive moving-average (ARMA) regression with different percentages of observable census tracts. (b) The same result is shown for the trip destination volume with ESN-EnKF only. (The horizontal line shows the Pearson correlation between the estimated and true destination volume when the true origin volume is used by the approximate PF operator.) (c) The normalized root-mean-square error (RMSE) for the destination volume is compared among the ESN-EnKF, reservoir observer, and ARMA regression. PF: Perron–Frobenius.

where $\mathbf{x}_{i,t}$ and $\hat{\mathbf{x}}_{i,t}$ denote the true and estimated scooter traffic volume on the tth day in the ith census tract/block group, respectively. The normalized RMSE is

$$\text{RMSE}(\mathbf{x}, \hat{\mathbf{x}}) = \sqrt{\frac{\sum_{i,t} (\mathbf{x}_{i,t} - \hat{\mathbf{x}}_{i,t})^2}{\sum_{i,t} \mathbf{x}_{i,t}^2}}. \tag{10}$$

The correlation coefficient and the normalized RMSE are also computed for the ARMA regression. True values of the trip origin volumes can be obtained from the observable census tracts/block groups. A comparison of the mean correlation coefficient computed for more than 50 Monte Carlo runs with randomized initialization of the ESN reservoir for the Austin data is shown in Figure 6(a). It is evident that the ESN performs better with the EnKF. The performance does not significantly change after 35% of the census tracts become observable. The same correlation is computed between the true and predicted trip destination volume time series, and the results are shown in Figure 6(b), where the trip destination volumes are predicted using the approximate PF operator from the predicted trip origin volumes. The average normalized RMSEs between the true and predicted destination

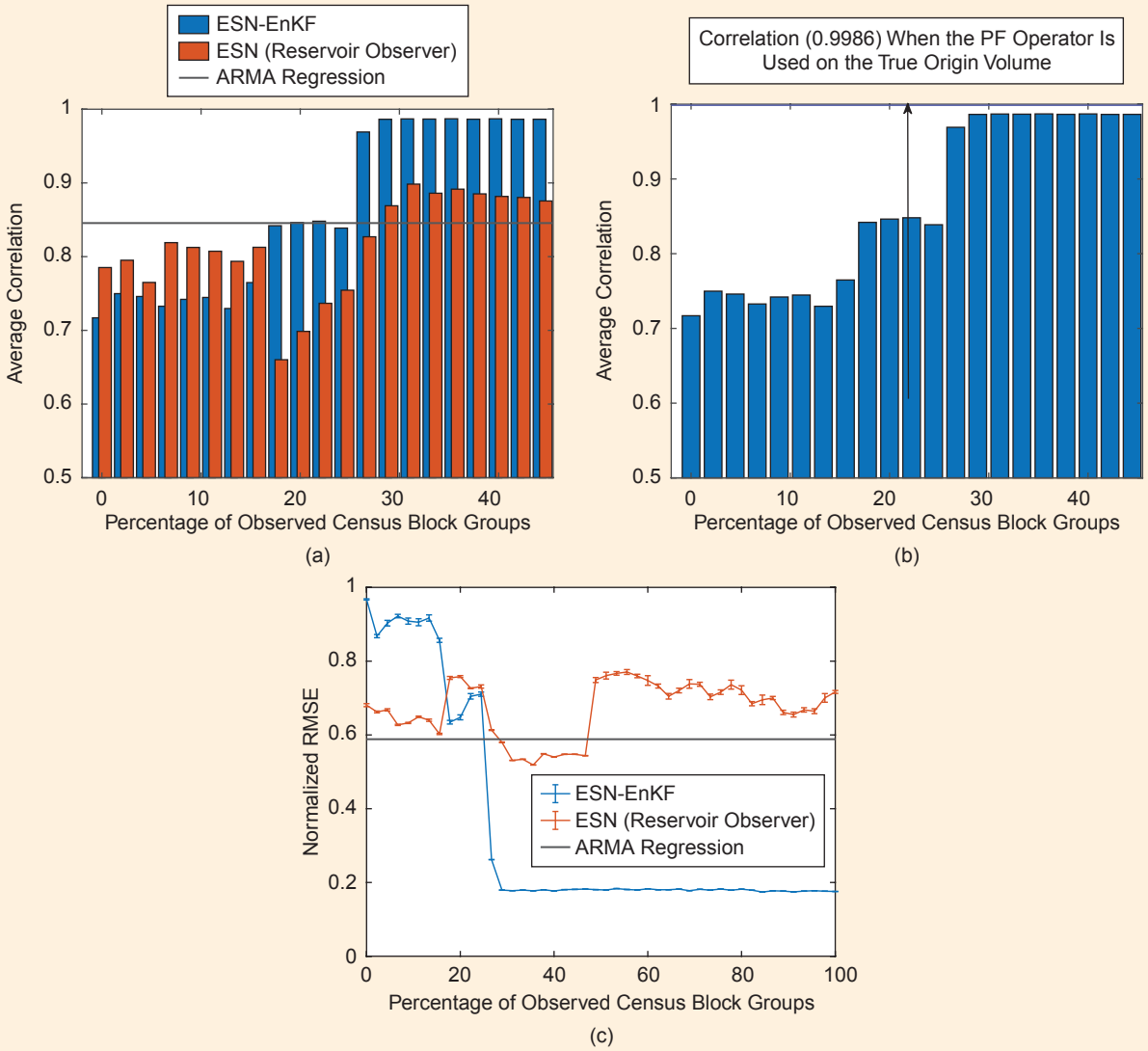

FIGURE 7 (a) The Pearson correlation between the true and estimated trip origin volumes for the Louisville data: The echo-state network algorithm with an ensemble Kalman filter (ESN-EnKF) is compared with the reservoir observer and a standard autoregressive moving-average (ARMA) regression with different percentages of observable census block groups. (b) The same result is shown for the trip destination volume with ESN-EnKF only. (The horizontal line shows the Pearson correlation between the estimated and true destination volume when the true origin volume is used by the approximate PF operator.) (c) The normalized root-mean-square error (RMSE) for the destination volume is compared among the ESN-EnKF, reservoir observer, and ARMA regression. PF: Perron–Frobenius.

volume series computed for more than 50 Monte Carlo runs are shown in Figure 6(c).

The same correlation and normalized RMSE comparisons between the true and predicted origin and destination data for the Louisville and Minneapolis data sets are presented in Figures 7 and 8, respectively. The proposed ESN-EnKF outperforms a standard ARMA regression, in terms of both the correlation coefficient and normalized RMSE, after 20%–25% of the census tracts/block groups become observable. The performance of the ESN-EnKF is significantly better than the ARMA regression for the Minneapolis data set [Figure 8(c)] in terms of the normalized RMSE, where the latter has a near 100% prediction error. A comparison of trip destination volume predictions using the true and predicted trip origin volume data is presented in Figure 9.

CONCLUSION

This article describes a data-driven method to predict e-scooter micromobility trends in cities with an ESN and Markov operator theory. The scooter trip data of a city are preprocessed and partitioned into different geographical regions (census tracts or block groups) according to their origin and destination. The resultant time series of trip origin volumes is used to train a neural network for future prediction. The trip destination volumes are predicted using an approximate PF operator, assuming a Markov relationship between the

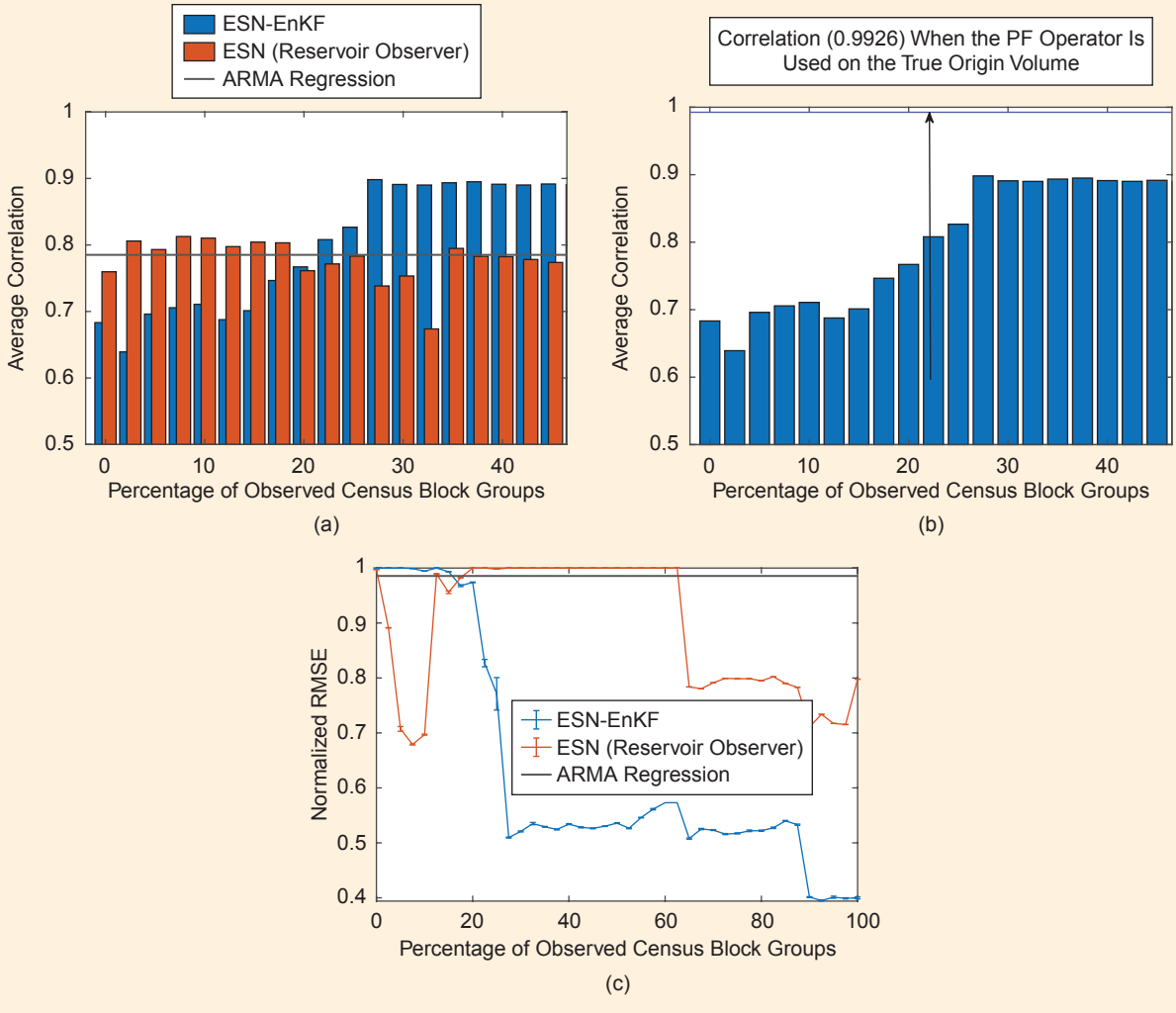

FIGURE 8 (a) The Pearson correlation between the true and estimated trip origin volumes for the Minneapolis data: the echo-state network algorithm with an ensemble Kalman filter (ESN-EnKF) is compared against the reservoir observer and a standard autoregressive moving-average (ARMA) regression with different percentages of observable census block groups. (b) The same result is shown for the trip destination volume with ESN-EnKF only. (The horizontal line shows the Pearson correlation between the estimated and true destination volume when the true origin volume is used by the approximate PF operator.) (c) The normalized root-mean-square error (RMSE) for the destination volume is compared among the ESN-EnKF, reservoir observer, and ARMA regression. PF: Perron–Frobenius.

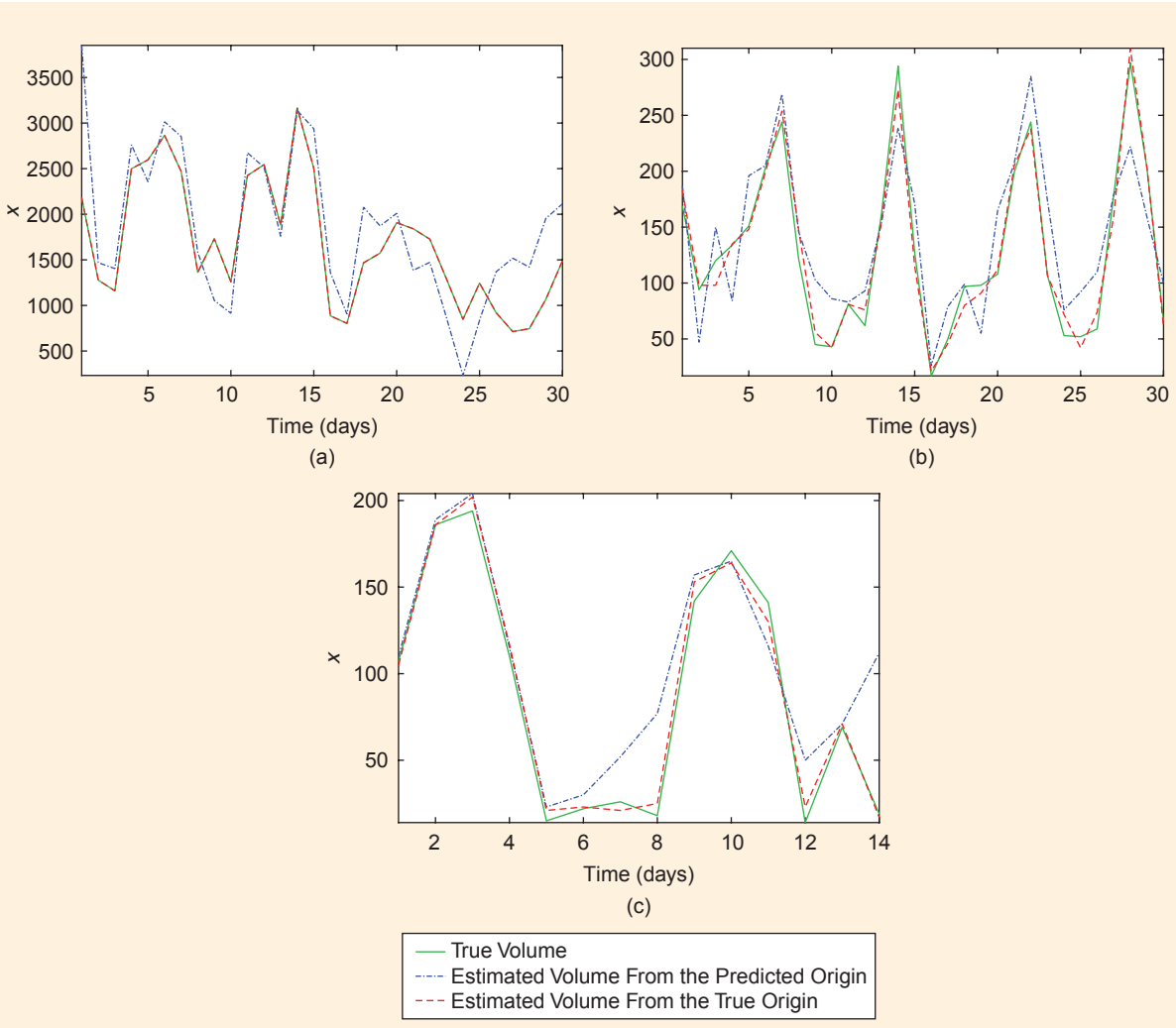

FIGURE 9 A comparison between the predictions of trip destination volume from the true and predicted origin volume data in (a) Austin, (b) Louisville, and (c) Minneapolis.

trip origin and destination. The proposed method is demonstrated over e-scooter data sets from three cities: Austin, Louisville, and Minneapolis. The method is also compared against the reservoir-observer method using the normalized RMSE and Pearson correlation as metrics.

The predicted usage data are useful in determining the optimal distribution of the shared mobility resources. They also may be used to quantify the equity and accessibility of personal mobility options in various neighborhoods of a city. The trip origin–destination prediction framework is not specific to e-scooters and could be applied to other data sets (for example, cell phone location data to predict population movement during pandemic-related travel restrictions).

ACKNOWLEDGMENTS

We thank Dr. Chenfeng Xiang for sharing the e-scooter data sets. This research was supported by a seed grant from the Maryland Transportation Institute.

AUTHOR INFORMATION

Debdipta Goswami (goswamid@princeton.edu) is a postdoctoral research associate in the Department of Mechanical and Aerospace Engineering at Princeton University, Princeton, New Jersey, 08540, USA. He received the Ph.D. degree in electrical and computer engineering from the University of Maryland, United States, in 2020 and the B.E. degree in electronics and telecommunication engineering from Jadavpur University, India, in 2015. His research interests include machine learning, nonlinear estimation, filtering, uncertainty quantification, and an operator theoretic approach to dynamical systems.

Ari Riggins is an undergraduate student in the Department of Computer Science at Princeton University, Princeton, New Jersey, 08540, USA. Her research interests include machine learning and statistics. She participated in the 2020 National Science Foundation Training and Research Experience in Nonlinear Dynamics—Research

Experiences for Undergraduates program at the University of Maryland.

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 at the University of Maryland, College Park, Maryland, 20740, USA. He received the B.S. degree in applied physics from Yale University in 1997 and the Ph.D. degree in mechanical and aerospace engineering from Princeton University in 2007. He teaches introductory dynamics, advanced dynamics, aircraft flight dynamics and control, and nonlinear control. His research interests include dynamics and control, including the cooperative control of autonomous vehicles, adaptive sampling with mobile networks, and spatial modeling of biological groups. He is associate fellow of the American Institute of Aeronautics and Astronautics and a Senior Member of IEEE.

REFERENCES

[1] J. Jiao, "Identifying transit deserts in major Texas cities where the supplies missed the demands," *J. Transport Land Use*, vol. 10, no. 1, pp. 529–540, Jan. 2017, doi: 10.5198/jtlu.2017.899.

[2] "New Lime report details NYC transit equity crisis and micromobility solution." Lime Blog. https://www.li.me/blog/lime-report-nyc-transit-equity-crisis-micromobility-solution (Accessed: Mar. 1, 2019).

[3] G. McKenzie, "Spatiotemporal comparative analysis of scooter-share and bike-share usage patterns in Washington, D.C.," *J. Transport Geography*, vol. 78, pp. 19–28, Jun. 2019, doi: 10.1016/j.jtrangeo.2019.05.007.

[4] J. Mathew, M. Liu, H. Li, S. Seeder, and D. Bullock, "Analysis of e-scooter trips and their temporal usage patterns," *ITE J.*, vol. 89, no. 6, pp. 44–49, Jun. 2019.

[5] Y. Shen, X. Zhang, and J. Zhao, "Understanding the usage of dockless bike sharing in Singapore," *Int. J. Sustain. Transp.*, vol. 12, no. 9, pp. 686–700, 2018, doi: 10.1080/15568318.2018.1429696.

[6] X. Li, Y. Zhang, L. Sun, and Q. Liu, "Free-floating bike sharing in Jiangsu: Users' behaviors and influencing factors," *Energies*, vol. 11, no. 7, p. 1664, Jun. 2018, doi: 10.3390/en11071664.

[7] T. Gu, I. Kim, and G. Currie, "To be or not to be dockless: Empirical analysis of dockless bikeshare development in China," *Transp. Res. A, Policy Pract.*, vol. 119, pp. 122–147, Jan. 2019, doi: 10.1016/j.tra.2018.11.007.

[8] S. J. Mooney *et al.*, "Freedom from the station: Spatial equity in access to dockless bike share," *J. Transport Geography*, vol. 74, pp. 91–96, Jan. 2019, doi: 10.1016/j.jtrangeo.2018.11.009.

[9] G. Hinton *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012, doi: 10.1109/MSP.2012.2205597.

[10] D. Silver *et al.*, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016, doi: 10.1038/nature16961.

[11] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986, doi: 10.1038/323533a0.

[12] 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, doi: 10.1126/science.1091277.

[13] 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 J. Nonlinear Sci.*, vol. 27, no. 4, pp. 1054–1500, 2017, doi: 10.1063/1.4979665.

[14] 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, no. 2, p. 024102, Jan. 2018, doi: 10.1103/PhysRevLett.120.024102.

[15] D. Goswami, A. Wolek, and D. A. Paley, "Data-driven estimation using an Echo-State Neural Network equipped with an Ensemble Kalman Filter," in *Proc. 2021 Amer. Control Conf. (ACC)*, pp. 2543–2548, doi: 10.23919/ACC50511.2021.9483373.

[16] G. Froyland and K. Padberg, "Almost-invariant sets and invariant manifolds — connecting probabilistic and geometric descriptions of coherent structures in flows," *Physica D, Nonlinear Phenomena*, vol. 238, no. 16, pp. 1507–1523, 2009, doi: 10.1016/j.physd.2009.03.002.

[17] D. Goswami, E. Thackray, and D. A. Paley, "Constrained Ulam dynamic mode decomposition: Approximation of Perron-Frobenius operator for deterministic and stochastic systems," *IEEE Control Syst. Lett.*, vol. 2, no. 4, pp. 809–814, 2018, doi: 10.1109/LCSYS.2018.2849552.

[18] W. Maass and H. Markram, "On the computational power of circuits of spiking neurons," *J. Comput. Syst. Sci.*, vol. 69, no. 4, pp. 593–616, 2004, doi: 10.1016/j.jcss.2004.04.001.

[19] P. L. Houtekamer and H. L. Mitchell, "Data assimilation using an ensemble Kalman filter technique," *Monthly Weather Rev.*, vol. 126, no. 3, pp. 796–811, 1998, doi: 10.1175/1520-0493(1998)126<0796:DAUAEK>2.0.CO;2.

[20] M. O. Williams, M. S. Hemati, S. T. M. Dawson, I. G. Kevrekidis, and C. W. Rowley, "Extending data-driven Koopman analysis to actuated systems," *IFAC-PapersOnLine*, vol. 49, no. 18, pp. 704–709, 2016, doi: 10.1016/j.ifacol.2016.10.248.

[21] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data–driven approximation of the Koopman operator: Extending dynamic mode decomposition," *J. Nonlinear Sci.*, vol. 25, no. 6, pp. 1307–1346, Dec. 2015, doi: 10.1007/s00332-015-9258-5.

[22] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, Mar. 2004.