

Traffic signal prediction on transportation networks using spatio-temporal correlations on graphs

Semin Kwak, Nikolas Geroliminis, and Pascal Frossard

Abstract—Multivariate time series forecasting poses challenges as the variables are intertwined in time and space, like in the case of traffic signals. Defining signals on graphs relaxes such complexities by representing the evolution of signals over a space using relevant graph kernels such as the heat diffusion kernel. However, this kernel alone does not fully capture the actual dynamics of the data as it only relies on the graph structure. The gap can be filled by combining the graph kernel representation with data-driven models that utilize historical data. This paper proposes a traffic propagation model that merges multiple heat diffusion kernels into a data-driven prediction model to forecast traffic signals. We optimize the model parameters using Bayesian inference to minimize the prediction errors and, consequently, determine the mixing ratio of the two approaches. Such mixing ratio strongly depends on training data size and data anomalies, which typically correspond to the peak hours for traffic data. The proposed model demonstrates prediction accuracy comparable to that of the state-of-the-art deep neural networks with lower computational effort. It notably achieves excellent performance for long-term prediction through the inheritance of periodicity modeling in data-driven models.

Index Terms—Multivariate time series forecasting, Bayesian inference, heat diffusion model, dynamic linear model.

I. INTRODUCTION

MULTIVARIATE time-series prediction is an important task since many real-life problems can be modeled within this framework, such as weather forecasting [1]–[3], traffic prediction [4]–[19], power consumption forecasting [10], [20], and others [5], [18], [21]–[23]. In transportation sensor networks, output signals from neighboring sensors may be similar or vastly different, as shown in Fig. 1(a) and (b). Therefore, in this example, sensor A’s signal can be utilized to predict sensor B’s as the two signals are well correlated. However, the signal of sensor C is not correlated with that of sensor B, so it may not contribute to the prediction; Sensor C is located after an intersection, and most traffic demands flow in another direction in the intersection, therefore, the sensor rarely suffers congestion. Naturally freeway congestion (expressed with a sharp decrease in the average speed of vehicles) is initiated at a bottleneck location such as an on-ramp merging area with high entrance flow or an incident location. Then, it propagates backwards with a finite speed, which is 3 to 4 times smaller than the speed of traffic. Fig. 1(c) shows an example of congestion propagation in I-280 and I-880 freeways in California. Note that there is a drastic decrease in the speed at a location (sensor B) and a time (around 3 pm) that propagates through the traffic stream (this is called a shockwave). Once demand for travel decreases congestion disappears by following the opposite trend during the offset of congestion with a forward moving wave. Note

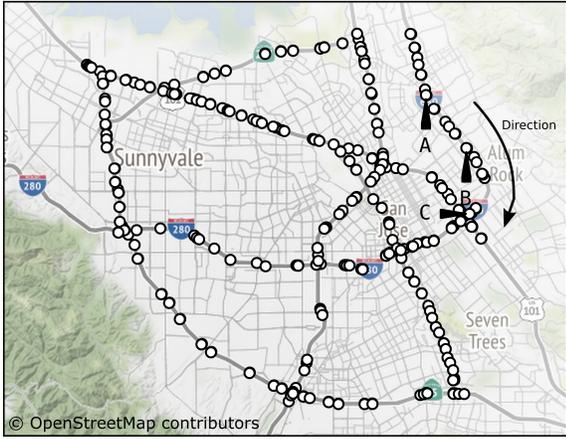
that this propagation speed is not constant and depends on the concentration or density of vehicles (with units of veh/km) on the two sides of the shockwave. There are various theories in transportation science to describe the mechanisms of stop-and-go phenomena inspired by fluid and heat diffusion models (see [24] for an overview).

Due to complex spatio-temporal correlation, the choice of model greatly influences the predictive performance. For small-scale sensor networks, such correlations can be estimated directly from historical data [6]–[10]. The vector Auto Regression (AR) is a representative model for multivariate time series forecasting [8]–[10]. In this model, regression parameters, or correlations between sensors, are estimated solely using historical data. In our previous work [8], we implemented a predictor that explicitly expresses the periodicity of traffic signals with temporally localized vector AR model. However, these data-driven models are not suitable for multivariate time series prediction with a large number of variables because the number of correlations to be estimated increases exponentially compared with the number of sensors, which causes incompleteness of the estimator (or overfitting).

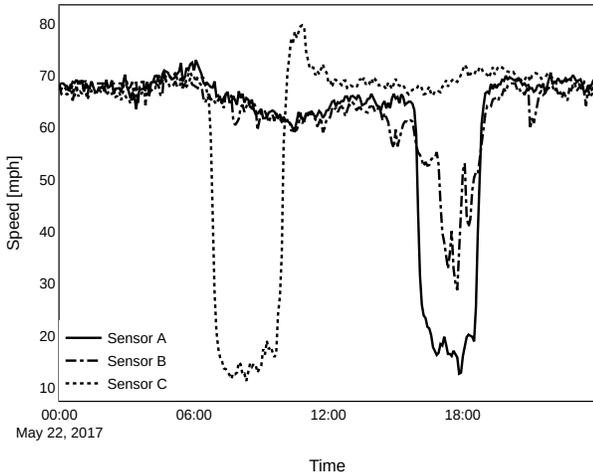
Recently, many studies have prioritized the correlations among sensors by defining signals on graphs [11]–[19]. In particular, in transportation networks, the physical travel distance between sensors is a critical *a priori* information, the closer the sensors are in space, the higher the correlation [25]. Utilizing this information, the authors had extracted the signal’s spatial features through the heat propagation kernel (or convolutional filter) and passed it to temporal blocks for forecasting, such as recurrent neural network (RNN) [11]–[14] and temporal convolutional layer (TCN) [15]–[18]. By introducing this prior information to complex deep neural networks, they achieved state-of-the-art performance in traffic prediction.

However, the two predictors (with and without graphs) each have their own drawbacks. In the former case, to the best of our knowledge, all studies, which currently show the best performance, construct predictors based on deep neural networks. Therefore, these models require expensive tuning processes of many hyperparameters and relatively long training due to numerical optimization processes. In the latter case, on the other hand, it can be inefficient concerning the prediction accuracy, especially for large networks when the structural information becomes important.

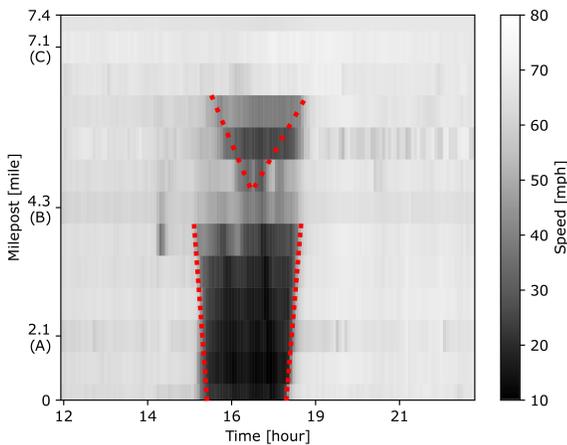
This paper proposes a new model that combines the advantages of different frameworks by implanting the sensors’ structural information into the existing data-driven model [8], inheriting the periodicity modeling for the traffic signal. In most studies, the periodicity of the traffic signal is taken as the input feature of the predictor, such as an encoded vector



(a) Sensor locations of PEMS-BAY network. The distance between two consecutive sensors in a freeway is 0.6 mile in average.



(b) Signals on different sensors



(c) Speed profile for the evening peak over time and space. The day 2017-05-22 (Monday) is selected. The red dashed lines represent the waves that congestion propagates.

Fig. 1. A transportation sensor network in California and signals of three different sensors on the network. Although the sensors B and C are close to each other in distance, two traffic signals from these sensors show very different patterns.

that represents the time of the day or the day of the week, but the study [8] instead induces the periodicity of the signal more clearly by making the model itself different for each time. Each model has a matrix, which should be estimated by historical data, representing the correlation between signals at two consecutive time intervals. As the size of the network is proportional to the size of the matrix, a larger network can lead to overfitting. In this paper, we resolve the overfitting problem by approximating this matrix to the one derived from data-independent graph topological information, therefore, we estimate only the remainder by data. In detail, we transform the graph topological information into heat diffusion kernels, which is introduced in [26], and approximate the matrix to a combination of the heat diffusion kernels. In the process, we introduce some hyper-parameters. For example, one determines which of the prior or historical datasets is more reliable. Most of the existing studies estimate hyper-parameters through exhaustive search as a cross-validation method using a validation set, but we estimate hyper-parameters directly from data by utilizing Bayesian inference [27]. As a result, the estimation process is relatively fast as most parameter estimation is performed by analytic calculations except a few ones requiring a numerical optimization process. Besides, our model is strongly interpretable. For example, through the hyper-parameter, it can be seen that during the peak period, traffic prediction is relatively more dependent on data than structural information compared to the non-peak period. Also, most importantly, predictors based on this model showed comparable performance with a much shorter learning time than state-of-the-art models. Especially, the proposed model shows great long-term prediction performance as the model captures well the periodicity of traffic signals. Since the proposed model requires a minimal number of hyper-parameter tuning, it might be applied to other daily periodic graph signal prediction problems easily (e.g., weather forecasting, daily energy consumption prediction). Here we summarize contributions of the work:

- We propose a novel traffic prediction method that successfully integrate graph structural information to the existing data-driven model [8]. Hyper-parameters are learned directly from data through Bayesian inference rather than by exhaustive search.
- Therefore, the training time required for inference is minimal. The trained model is straightforward to analyze, unlike other deep neural network-based models.
- It shows prediction performance comparable with deep learning methods especially for long-term prediction.

II. DATA MODEL

In this section, we describe a mathematical model that represents a relationship between traffic signals that are different in time. First, we define traffic signals on a graph and introduce an existing prediction model [8] using this signals. Then, we suggest a model extending the previous one that is applicable for large scale networks by exploiting graph information.

TABLE I
THE NOTATIONS AND DEFINITIONS USED IN THIS ARTICLE.

\mathcal{R}^m	m -dimensional Euclidean space
$a, \mathbf{a}, \mathbf{A}$	Scalar, vector, matrix
$\text{diag}(\mathbf{a})$	The diagonal matrix whose diagonal elements are from the vector \mathbf{a}
$\text{diag}(\mathbf{A})$	The vector whose elements are the diagonal components of the matrix \mathbf{A}
\mathbf{I}	Identity matrix
$\mathbf{1}$	All one vector
$e^{\mathbf{A}}$	$\lim_{n \rightarrow \infty} (\mathbf{I} + \frac{1}{n} \mathbf{A})^n = \sum_{n=0}^{\infty} \frac{1}{n!} \mathbf{A}^n$
$[\mathbf{A}]_{i,j}$	The element of i -th row and j -th column of the matrix \mathbf{A}
$[\mathbf{A}]_{i,:}$	The slice of i -th row of the matrix \mathbf{A}
$ \mathbf{A} $	The determinant of the matrix \mathbf{A}
$ \mathcal{S} $	The cardinality of the set \mathcal{S}
$\mathcal{N}(\mu, \sigma^2)$	A Gaussian distribution which has the probability density function $f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)$
$\mathcal{N}(\mu, \Sigma)$	A multivariate Gaussian distribution which has the probability density function $f(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^N \Sigma }} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu)\right)$
$\mathcal{N}(\mathbf{M}, \sigma^2)$	$\prod_{i,j} \mathcal{N}([\mathbf{M}]_{i,j}, \sigma^2)$
$\mathcal{N}(\mathbf{M}, \Sigma)$	$\prod_i \mathcal{N}([\mathbf{M}]_{i,:}, \Sigma)$

A. Graph signal

We start with modeling a transportation network using a graph. We define an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$; \mathcal{V} is a set of nodes where each $v \in \mathcal{V}$ denotes a node (sensor) on the graph; \mathcal{E} is a set of edges where each of the edges connects two nodes. We define a signal on the nodes of the graph with a traffic feature, in this paper, for instance, speed, which is expressed as a vector $\mathbf{x}_t^d \in \mathcal{R}^N$ of a day d and time t , where the constant N is the number of nodes. Therefore, the vector \mathbf{x}_t^d represents a snapshot of speeds at a particular time and day. Especially, we express the day index on the vector representation to exploit the periodicity of traffic signals later.

B. Dynamic linear model (DLM)

In our previous study [8], we defined a state equation of traffic in a small-scale transportation network (a path graph) as temporally localized linear models as follows:

$$\mathbf{x}_{t+1}^d = \mathbf{H}_t \mathbf{x}_t^d + \mathbf{n}_t^d, \forall t \in [0, T-1]. \quad (1)$$

We called this model the Dynamic linear model (DLM). The first time index ($t = 0$) corresponds to the beginning of a day (midnight in our work), and the last index ($t = T-1$) refers to the end of the day. Each entry of the noise vector $\mathbf{n}_t^d \in \mathcal{R}^N$ is assumed to be an independent and identically distributed (i.i.d.) random variable, which follows a Gaussian distribution $\mathcal{N}(0, \alpha_t^{-1})$. Here the precision parameter α_t explains how precisely a data pair $(\mathbf{x}_t^d, \mathbf{x}_{t+1}^d)$ fits to the model. The transition matrix \mathbf{H}_t represents the linear relationship between traffic signals \mathbf{x}_t^d and \mathbf{x}_{t+1}^d .

The most important motivation behind this model is that the propagation of traffic features over time occurs periodically on a daily basis. Consequently, we modeled that the transition

matrix \mathbf{H}_t as a time-variant matrix that contains temporally localized (only between two consecutive traffic features) spatio-temporal correlations of every sensor pair regardless of the day of the week, noting that the transition matrix does not have the day index. In other words, we assumed the correlations are identical both for weekends and weekdays [8].

In the work [8], the transition matrix is estimated by maximizing the likelihood (note that we ignore some parameters such as the regularization parameter and the forgetting factor introduced in the work for the brevity) as follows:

$$\bar{\mathbf{H}}_t = \underset{\mathbf{H}_t}{\text{argmax}} f(\mathbf{X}_{t+1} | \mathbf{X}_t, \mathbf{H}_t, \alpha_t) = \mathbf{X}_{t+1} \mathbf{X}_t^T (\mathbf{X}_t \mathbf{X}_t^T)^{-1}, \quad (2)$$

where the collection of the m -past signals $\mathbf{X}_t = (\mathbf{x}_t^0 \ \mathbf{x}_t^1 \ \dots \ \mathbf{x}_t^{m-1})$. Therefore, the optimal transition matrix is solely determined by the historical data \mathbf{X}_t and \mathbf{X}_{t+1} . From Eq. (2) we see that the matrix $\mathbf{X}_t \mathbf{X}_t^T$ can be an ill-conditioned matrix when N is large. In other words, the transition matrix $\bar{\mathbf{H}}_t$ can be overfitted by data. In the following subsection, we suggest a method to avoid this problem by utilizing graph topological information.

C. DLM with graph topological information

In this subsection, we suggest a way to avoid the overfitting problem approximating the transition matrix to a heat diffusion matrix. To achieve this goal, we first define a weight matrix that contains all edge weights between node v_i and v_j using a Gaussian kernel weighting function with a threshold constant κ :

$$[\mathbf{W}]_{i,j} = \begin{cases} e^{-\frac{\text{dist}^2(i,j)}{\sigma^2}}, & \text{if } \text{dist}(i,j) \leq \kappa \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

The function $\text{dist}(i,j)$ denotes the shortest travel distance on \mathcal{G} between the node v_i and v_j :

$$\text{dist}(i,j) = \min\{\text{dist}(v_i \rightarrow v_j), \text{dist}(v_j \rightarrow v_i)\}, \quad (4)$$

where the function $\text{dist}(v_i \rightarrow v_j)$ represents the shortest travel distance from node v_i to node v_j . As the graph \mathcal{G} is undirected, the weight matrix is a symmetric matrix, i.e., $\mathbf{W}^T = \mathbf{W}$.

The constants σ and κ are the kernel width and the distance threshold. If the kernel width is large, the correlation of a pair of nodes is strong (close to one) even though the shortest travel distance between the two nodes is large. On the other hand, the smaller the threshold is, the sparser the weight matrix is.

The graph heat diffusion model [26] explains how each vertex propagates its heat to its neighbors on the graph over time. As congestion evolves from one location to its neighbor over time, we can express the change of traffic features by the heat diffusion model, especially for short-term traffic changes since the total traffic volume of a network is well preserved for the short-term in general.

The kernel on graphs that supports the heat diffusion model is introduced by [26]:

$$\mathbf{H}^{\mathcal{G}}(\tau) = e^{-\tau \mathbf{L}(\mathcal{G})}, \quad (5)$$

where the constant τ denotes the diffusion period and the matrix $\mathbf{L}(\mathcal{G})$ is the Laplacian of a graph \mathcal{G} . The matrix is defined as

$$\mathbf{L}(\mathcal{G}) = \text{diag}(\mathbf{W}\mathbf{1}) - \mathbf{W}. \quad (6)$$

By definition, two extreme heat diffusion kernels of a connected graph \mathcal{G} are:

$$\mathbf{H}^{\mathcal{G}}(\tau) = \begin{cases} \mathbf{I}, & \text{when } \tau \rightarrow 0, \\ \frac{1}{N}\mathbf{1}\mathbf{1}^T, & \text{when } \tau \rightarrow \infty, \end{cases} \quad (7)$$

where $\mathbf{1}$ is the vector whose elements are all one.

Therefore, with the heat diffusion kernel, we can describe the diffusion of a traffic signal through the graph \mathcal{G} as follows:

$$\tilde{\mathbf{x}}_{t+1}^d(\tau) = \mathbf{H}^{\mathcal{G}}(\tau)\mathbf{x}_t^d. \quad (8)$$

We call the vector $\tilde{\mathbf{x}}_{t+1}^d(\tau)$ the internally diffused signals from \mathbf{x}_t^d by the diffusion period τ on the graph \mathcal{G} over one incremental time step.

Here, we define a convex combination of the heat diffusion kernels of K different predetermined diffusion periods with a set $\mathcal{T} = \{\tau^{(0)}, \tau^{(1)}, \dots, \tau^{(K-1)}\}^1$ as

$$\mathbf{H}^{\mathcal{G}}(\mathcal{T}) = \sum_{\tau \in \mathcal{T}} \pi^{(\tau)} \mathbf{H}^{\mathcal{G}}(\tau), \quad (9)$$

where $\sum_{\tau \in \mathcal{T}} \pi^{(\tau)} = 1$. The mixture retains the property that the total input volume is preserved through the diffusion process as shown in Appendix A, i.e., $\mathbf{1}^T \mathbf{H}^{\mathcal{G}}(\mathcal{T}) \mathbf{x}_t^d = \mathbf{1}^T \mathbf{x}_t^d$.

We embed heat diffusion kernels into DLM to exploit topological information of the transportation network. The key idea is to express the transition matrix as a small variant from a mixture of diffusion kernels. We decompose the transition matrix into the time-variant internal diffusion and residual as follows:

$$\mathbf{H}_t = \mathbf{H}_t^{\mathcal{G}}(\mathcal{T}) + \text{residual} \quad (10)$$

so that the internal diffusion matrix $\mathbf{H}_t^{\mathcal{G}}(\mathcal{T})$ preserves the total traffic volume over time, i.e., $\mathbf{1}^T \mathbf{H}_t^{\mathcal{G}}(\mathcal{T}) \mathbf{x}_t^d = \mathbf{1}^T \mathbf{x}_t^d$. Here, the time dependent internal transition matrix can be safely defined as in Eq. (9) by substituting the time-invariant parameter $\pi^{(\tau)}$ for the time-variant one $\pi_t^{(\tau)}$ because of the volume conservation property. The internal diffusion matrix represents how the current signal \mathbf{x}_t^d diffuses through the transportation network (endogenous) whereas the residual represents how much the traffic situation is getting better or worse in the next time step based on the current signal (exogenous).

With this interpretation, we model the prior distribution of the transition matrix as:

$$f(\mathbf{H}_t | \gamma_t, \Pi_t, \mathcal{G}) = \mathcal{N}(\mathbf{H}_t^{\mathcal{G}}(\mathcal{T}), \gamma_t^{-1}), \quad (11)$$

where the precision parameter γ_t represents how precisely the diffusion matrix explains the transition matrix and $\Pi_t = \{\pi_t^{(\tau)} | \tau \in \mathcal{T}\}$.

The decomposition allows us to utilize data more efficiently during the estimation process later. In Eq. (1), the transition matrix is a variable to be estimated from the data. Since the dimension of this matrix is N^2 , an increase in the number of

¹We predetermine the set \mathcal{T} with two diffusion periods τ_0 and τ_∞ that correspond to each extreme case in Eq. (7), respectively. In practice, we set τ_0 as the biggest one that satisfies $\|\mathbf{H}^{\mathcal{G}}(\tau) - \mathbf{I}\|_2 < \epsilon$ and τ_∞ as the smallest one that satisfies $\|\mathbf{H}^{\mathcal{G}}(\tau) - \frac{1}{N}\mathbf{1}\mathbf{1}^T\|_2 < \epsilon$ with a predefined set $\tau \in \text{linspace}(-10, 10, 0.1)$, where the set contains evenly spaced (0.1) numbers from -10 to 10 . After that, we define $\mathcal{T} = \text{logspace}(\tau_0, \tau_\infty, K)$, where the function returns K evenly spaced numbers on a log scale from τ_0 to τ_∞ .

Algorithm 1 Inference of parameters

```

1: function INFERENCE( $\mathbf{W}, K, \mathbf{X}_{1:T}$ )
2:   Set  $\mathcal{T} = \text{logspace}(\tau_0, \tau_\infty, K)$ 
3:   Define  $\mathbf{L}(\mathcal{G})$  by Eq. (6)
4:   Define the function  $\mathbf{H}^{\mathcal{G}}(\tau) = e^{-\tau\mathbf{L}(\mathcal{G})}$ 
5:   for  $t \in [0, T - 2]$  do
6:     Infer  $\hat{\alpha}_t, \hat{\gamma}_t$  and  $\hat{\Pi}_t$  by solving (25)
7:     Infer  $\hat{\mathbf{H}}_t$  by Eq. (19)
8:   end for
9:   return  $\hat{\mathbf{H}}_t, \forall t$ 
10: end function

```

sensors causes the estimation of more elements, which results in an overfitting problem. This is the biggest impediment to extending DLM to large networks. Still, if the structural information is set as *a priori* through Eq. (11), the problem can be effectively avoided even if the number of sensors increases. Assuming the graph \mathcal{G} and the period set \mathcal{T} are predefined, the internal diffusion matrix only depends on the parameters $\pi_t^{(\tau)}$. By setting the number of diffusion periods to be much smaller than that of sensors i.e., $|\mathcal{T}| \ll N$, we can describe the major part of the transition matrix by the internal diffusion matrix with a few parameters when the sampling interval (the time difference of two consecutive time indices) is relatively short, with likely preservation of the traffic volumes, i.e., $\mathbf{1}^T \mathbf{x}_{t+1}^d \approx \mathbf{1}^T \mathbf{x}_t^d$. Consequently, we only need to exploit data to infer the parameters $\pi_t^{(\tau)}$ and the residual part whose norm is small with the decomposition.

III. PREDICTION AND INFERENCE

This section describes how to estimate modeling parameters and predict graph signals by using the model. Both the estimation and the prediction were performed by maximizing the posterior distribution of each variable. Especially for hyperparameters, we utilize Bayesian inference to estimate them instead of exhaustive search.

A. Inference of the transition matrix

We infer the transition matrix by maximizing its posterior distribution:

$$\hat{\mathbf{H}}_t = \underset{\mathbf{H}_t}{\text{argmax}} f(\mathbf{H}_t | \mathbf{X}_t, \mathbf{X}_{t+1}, \alpha_t, \gamma_t, \Pi_t, \mathcal{G}), \quad (12)$$

which is proportional to the product of the prior and the likelihood by Bayes' rule:

$$\text{Posterior dist.} \propto f(\mathbf{H}_t | \gamma_t, \Pi_t, \mathcal{G}) f(\mathbf{X}_{t+1} | \mathbf{X}_t, \mathbf{H}_t, \alpha_t). \quad (13)$$

Maximizing the posterior distribution can be interpreted as balancing between the prior and likelihood of the transition matrix. For example, if there is no topological information about sensors, the transition matrix should be inferred by considering the training dataset only. In this case, we can set the prior distribution as a uniform distribution, meaning that there is no strong preference for a particular value of the

transition matrix; the most probable transition matrix becomes the maximum likelihood solution, which is Eq. (2):

$$\begin{aligned} \hat{\mathbf{H}}_t | \text{No topological info.} &:= \bar{\mathbf{H}}_t \\ &= \underset{\mathbf{H}_t}{\operatorname{argmax}} f(\mathbf{X}_{t+1} | \mathbf{X}_t, \mathbf{H}_t, \alpha_t) \\ &= \mathbf{X}_{t+1} \mathbf{X}_t^T (\mathbf{X}_t \mathbf{X}_t^T)^{-1}. \end{aligned} \quad (14)$$

On the other hand, if we do not have any measurements, the most probable transition matrix should be the maximizer of the prior distribution:

$$\hat{\mathbf{H}}_t | \text{No measurements} = \underset{\mathbf{H}_t}{\operatorname{argmax}} f(\mathbf{H}_t | \gamma_t, \Pi_t, \mathcal{G}) = \mathbf{H}_t^{\mathcal{G}}(\mathcal{T}). \quad (15)$$

Since we use both prior and data measurements, the actual optimal transition matrix becomes a combination of these two. According to the dynamic linear model, the likelihood

$$f(\mathbf{X}_{t+1} | \mathbf{H}_t, \mathbf{X}_t, \alpha_t) \propto e^{-\frac{1}{2} \operatorname{tr}\{\alpha_t (\mathbf{X}_{t+1} - \mathbf{H}_t \mathbf{X}_t) (\mathbf{X}_{t+1} - \mathbf{H}_t \mathbf{X}_t)^T\}} \quad (16)$$

and the prior

$$f(\mathbf{H}_t | \gamma_t, \Pi_t, \mathcal{G}) \propto e^{-\frac{1}{2} \operatorname{tr}\{\gamma_t (\mathbf{H}_t - \mathbf{H}_t^{\mathcal{G}}(\mathcal{T})) (\mathbf{H}_t - \mathbf{H}_t^{\mathcal{G}}(\mathcal{T}))^T\}}. \quad (17)$$

Therefore, by Eq. (13), (16) and (17),

$$\begin{aligned} f(\mathbf{H}_t | \mathbf{X}_{t+1}, \mathbf{X}_t, \alpha_t, \gamma_t, \Pi_t, \mathcal{G}) &\propto e^{-\frac{1}{2} \alpha_t \operatorname{tr}\{(\mathbf{X}_{t+1} - \mathbf{H}_t \mathbf{X}_t) (\mathbf{X}_{t+1} - \mathbf{H}_t \mathbf{X}_t)^T\}} \\ &\quad \cdot e^{-\frac{1}{2} \gamma_t \operatorname{tr}\{(\mathbf{H}_t - \mathbf{H}_t^{\mathcal{G}}(\mathcal{T})) (\mathbf{H}_t - \mathbf{H}_t^{\mathcal{G}}(\mathcal{T}))^T\}} \\ &\propto e^{-\frac{1}{2} \operatorname{tr}\{(\mathbf{H}_t - \hat{\mathbf{H}}_t) (\alpha_t \mathbf{X}_t \mathbf{X}_t^T + \gamma_t \mathbf{I}) (\mathbf{H}_t - \hat{\mathbf{H}}_t)^T\}}, \end{aligned} \quad (18)$$

where

$$\begin{aligned} \hat{\mathbf{H}}_t &= (\bar{\mathbf{H}}_t \alpha_t \mathbf{U}_t \mathbf{\Lambda}_t + \mathbf{H}_t^{\mathcal{G}}(\mathcal{T}) \gamma_t \mathbf{U}_t) (\alpha_t \mathbf{\Lambda}_t + \gamma_t \mathbf{I})^{-1} \mathbf{U}_t^T \\ &= \bar{\mathbf{H}}_t \alpha_t \mathbf{U}_t \mathbf{\Lambda}_t (\alpha_t \mathbf{\Lambda}_t + \gamma_t \mathbf{I})^{-1} \mathbf{U}_t^T \\ &\quad + \mathbf{H}_t^{\mathcal{G}}(\mathcal{T}) \gamma_t \mathbf{U}_t (\alpha_t \mathbf{\Lambda}_t + \gamma_t \mathbf{I})^{-1} \mathbf{U}_t^T, \end{aligned} \quad (19)$$

with the eigendecomposition of $\mathbf{X}_t \mathbf{X}_t^T = \mathbf{U}_t \mathbf{\Lambda}_t \mathbf{U}_t^T$. Therefore, $f(\mathbf{H}_t | \mathbf{X}_{t+1}, \mathbf{X}_t, \alpha_t, \gamma_t, \Pi_t, \mathcal{G})$ is a multivariate Gaussian distribution with mean $\hat{\mathbf{H}}_t$ and the covariance of each row; $(\alpha_t \mathbf{X}_t \mathbf{X}_t^T + \gamma_t \mathbf{I})^{-1}$.

Here, we measure how much each part contributes to the transition matrix

$$c_t^{\text{data}} = \frac{w_t^{\text{data}}}{w_t^{\text{data}} + w_t^{\text{prior}}}, \quad c_t^{\text{prior}} = \frac{w_t^{\text{prior}}}{w_t^{\text{data}} + w_t^{\text{prior}}} \quad (20)$$

by defining the weight of each part

$$\begin{aligned} w_t^{\text{data}} &= \left\| \alpha_t \mathbf{U}_t \mathbf{\Lambda}_t (\alpha_t \mathbf{\Lambda}_t + \gamma_t \mathbf{I})^{-1} \mathbf{U}_t^T \right\|_F, \\ w_t^{\text{prior}} &= \left\| \gamma_t \mathbf{U}_t (\alpha_t \mathbf{\Lambda}_t + \gamma_t \mathbf{I})^{-1} \mathbf{U}_t^T \right\|_F. \end{aligned} \quad (21)$$

Note that these weights depend on the precision parameters α_t and γ_t . If the data precision parameter α_t is relatively large compared to γ_t , then $c_t^{\text{data}} > c_t^{\text{prior}}$, meaning that the contribution of data measurements is larger than that of the prior information.

Algorithm 2 Prediction of traffic features (h -steps ahead)

```

function PREDICTION( $\mathbf{x}_t^d, h, \hat{\mathbf{H}}_t, \dots, \hat{\mathbf{H}}_{t+h-1}$ )
  Set  $\mathbf{p} = \mathbf{x}_t^d$ 
  for  $i \in [0, h - 1]$  do
    Set  $\mathbf{p} = \hat{\mathbf{H}}_{t+i} \mathbf{p}$ 
  end for
   $\mathbf{x}_{t+h|t} = \mathbf{p}$ 
  return  $\mathbf{x}_{t+h|t}$ 
end function

```

B. Inference of other parameters

For the next step, we infer parameters α_t , γ_t , and Π_t . Similar to inferring the most probable transition matrix, we infer the most probable α_t , γ_t , and Π_t by maximizing the following posterior distribution:

$$\hat{\alpha}_t, \hat{\gamma}_t, \hat{\Pi}_t = \underset{\alpha_t, \gamma_t, \Pi_t}{\operatorname{argmax}} f(\alpha_t, \gamma_t, \Pi_t | \mathbf{X}_{t+1}, \mathbf{X}_t). \quad (22)$$

Setting the prior distribution $f(\alpha_t, \gamma_t, \Pi_t)$ as a uniform distribution based on the assumption that there is no preference for a certain value for these parameters before inferring, the objective changes to maximize *evidence* $f(\mathbf{X}_{t+1} | \mathbf{X}_t, \alpha_t, \gamma_t, \Pi_t)$ [27] since

$$\begin{aligned} f(\alpha_t, \gamma_t, \Pi_t | \mathbf{X}_{t+1}, \mathbf{X}_t) &\propto f(\mathbf{X}_{t+1} | \mathbf{X}_t, \alpha_t, \gamma_t, \Pi_t) f(\alpha_t, \gamma_t, \Pi_t) \\ &\propto f(\mathbf{X}_{t+1} | \mathbf{X}_t, \alpha_t, \gamma_t, \Pi_t). \end{aligned} \quad (23)$$

In Appendix B, we show that the evidence is

$$\begin{aligned} f(\mathbf{X}_{t+1} | \mathbf{X}_t, \alpha_t, \gamma_t, \Pi_t) &= \int f(\mathbf{X}_{t+1} | \mathbf{X}_t, \mathbf{H}_t, \alpha_t) f(\mathbf{H}_t | \gamma_t, \Pi_t) d\mathbf{H}_t \\ &= \mathcal{N}(\mathbf{H}_t^{\mathcal{G}}(\mathcal{T}) \mathbf{X}_t, \alpha_t^{-1} \mathbf{I} + \gamma_t^{-1} \mathbf{X}_t^T \mathbf{X}_t). \end{aligned} \quad (24)$$

Therefore, we infer the most probable hyper-parameters by maximizing the log-evidence with a quasi-newton method (L-BFGS-B [28]):

$$\begin{aligned} &\underset{\alpha_t, \gamma_t, \Pi_t}{\operatorname{maximize}} \quad \log \mathcal{N}(\mathbf{H}_t^{\mathcal{G}}(\mathcal{T}) \mathbf{X}_t, \alpha_t^{-1} \mathbf{I} + \gamma_t^{-1} \mathbf{X}_t^T \mathbf{X}_t) \\ &\text{subject to} \quad 0 \leq \pi_t^{(\tau)} \leq 1 \quad \forall \tau \in \mathcal{T}, \quad 0 < \alpha_t, \quad 0 < \gamma_t, \\ &\quad \sum_{\tau \in \mathcal{T}} \pi_t^{(\tau)} = 1. \end{aligned} \quad (25)$$

Algorithm 1 summarizes the inference processes.

We emphasize that parameter inference through evidence maximization prevents overfitting of the transition matrix to either data measurements or prior information. In Eq. (24) we calculate the evidence by marginalizing the transition matrix. In other words, we set the transition matrix as a *random variable* instead of fixing it as a representative value, e.g., maximum likelihood estimator. Noting that these parameters determine the contributions of measurements and priors when the transition matrix is estimated in Eq. (19), the marginalization process automatically penalizes the transition matrix to avoid the extreme cases [27].



Fig. 2. Transportation sensor networks (District 7 area in California) that are used for evaluating the proposed method.

C. Prediction of traffic features

Prediction of traffic features is performed by extracting and exploiting as much information as possible from measurements and prior knowledge. Mathematically, we can express a traffic signal that we want to predict as a random variable since the signal defined in the future is entirely unknown. In this paper, therefore, we try to infer the probability density function of the signal \mathbf{x}_{t+h}^d

$$f(\mathbf{x}_{t+h}^d | \mathbf{x}_t^d, \mathbf{x}_{t-1}^d \cdots, \mathcal{G}), \quad (26)$$

where the time indices t and $t+h$ represent respectively the current time and the future time index (h -steps ahead) that we want to predict. In the expression, the probability density function is conditioned by the signals $\{\mathbf{x}_t^d, \mathbf{x}_{t-1}^d, \cdots\}$ and the graph \mathcal{G} that represents a set of measurements and prior structural information, respectively.

In reality, it is common to limit the number of measurements to a fixed-sized one in a training set. In addition to the training set that contains measurements apart from the day to be predicted, it is crucial to keep measurements just before t , as the temporal correlation is strong when the time difference is small. As a result, we estimate the density function that is conditioned by a training set, the p -most recent measurements, and the graph \mathcal{G} :

$$f(\mathbf{x}_{t+h}^d | \mathbf{x}_t^d, \mathbf{x}_{t-1}^d, \cdots, \mathbf{x}_{t-(p-1)}^d, \mathbf{X}_{0:T-1}, \mathcal{G}), \quad (27)$$

where the training set $\mathbf{X}_{0:T-1}$ contains signals from $t=0$ to $t=T-1$ of multiple days $d \in [0, m-1]$. The dynamic linear model further simplifies the distribution (27) as follows

$$f(\mathbf{x}_{t+h}^d | \mathbf{x}_t^d, \mathbf{X}_{t:t+h}, \mathcal{G}) \quad (28)$$

because of the temporal locality of the model.

We define a predictor $\mathbf{x}_{t+h|t}^d$ at the time step t for the horizon h as the maximizer of the probability density function

$$\mathbf{x}_{t+h|t}^d := \underset{\mathbf{x}_{t+h}^d}{\operatorname{argmax}} f(\mathbf{x}_{t+h}^d | \mathbf{x}_t^d, \mathbf{X}_{t:t+h}, \mathcal{G}). \quad (29)$$

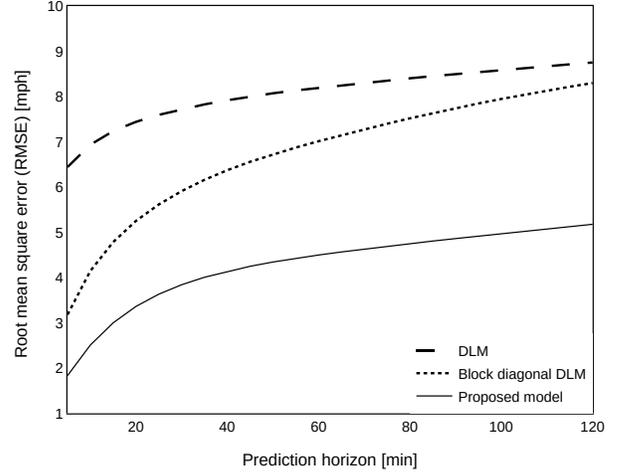


Fig. 3. Prediction accuracy (RMSE) for the three different models on the PEMS-BAY dataset. Each model represents respectively a single DLM (without topological information), separate multiple DLMs for each freeway, and the proposed model (a DLM with topological information).

In other words, we define the predictor $\mathbf{x}_{t+h|t}^d$ as the most probable \mathbf{x}_{t+h}^d based on the current measurement vector \mathbf{x}_t^d , the training set $\mathbf{X}_{t:t+h}$, and the graph \mathcal{G} .

Proposition 1. $f(\mathbf{x}_{t+h}^d | \mathbf{x}_t^d, \mathbf{X}_{t:t+h}, \mathcal{G})$ is a Gaussian distribution that has the mean vector $\hat{\mathbf{H}}_{t+h-1} \cdots \hat{\mathbf{H}}_t \mathbf{x}_t^d$ assuming $f(\mathbf{H}_t | \mathbf{X}_t, \mathbf{X}_{t+1}, \alpha_t, \gamma_t, \Pi_t, \mathcal{G}) = \delta(\mathbf{H}_t - \hat{\mathbf{H}}_t)$, where the Dirac delta function $\delta(x) = 1$ when $x = 0$ and $\delta(x) = 0$, otherwise. The most probable transition $\hat{\mathbf{H}}_t$ is the maximizer of the posterior distribution $f(\mathbf{H}_t | \cdot)$.

Proof. See Appendix C. \square

Since the mean value of a Gaussian distribution maximizes the distribution, the optimal predictor is

$$\mathbf{x}_{t+h|t}^d = \hat{\mathbf{H}}_{t+h-1} \cdots \hat{\mathbf{H}}_t \mathbf{x}_t^d := \hat{\mathbf{H}}_{t+h-1 \leftarrow t} \mathbf{x}_t^d. \quad (30)$$

Therefore, the most probable signal \mathbf{x}_{t+h}^d is the successive propagation of the current measurement vector \mathbf{x}_t^d through the most probable transition matrices that coincides with a straightforward computation with Eq. (1) ignoring the noise term. Therefore, the prediction for any horizon is just a matrix multiplication. Algorithm 2 summarizes this.

IV. EXPERIMENTS

A. Settings

The proposed method was evaluated on different transportation networks. Figure 2 shows the networks (\mathcal{G}_1 and \mathcal{G}_2) consisting of respectively 288 and 357 sensors with multiple freeways that are connected through ramps. They experience significant levels of congestion in the morning and evening peaks at various locations. These networks connect many origins and destinations with complex demand profiles, creating propagation of congestion that is different in duration, size, and time of occurrence. The PEMS-BAY dataset was also used as a benchmark to compare with other state-of-the-art models [11], [17]. This data set consists of data measured

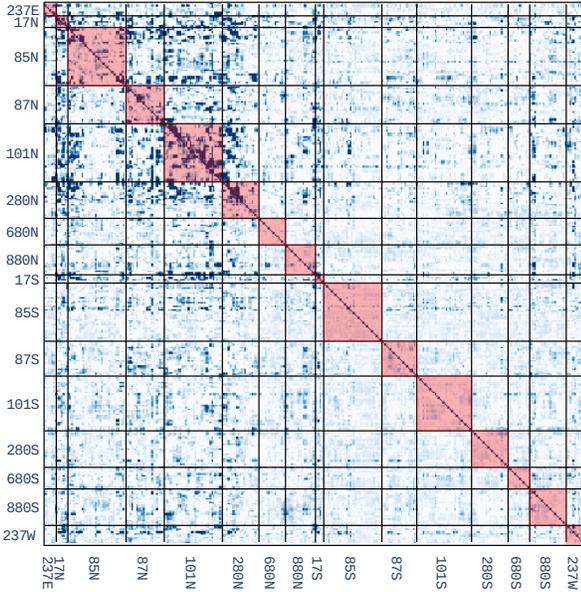


Fig. 4. The heatmap of the elements in an estimated transition matrix \mathbf{H}_t of the proposed model. Darker colors represent larger absolute values. The sensors are grouped by freeways and ordered from upstream to downstream within each freeway. Each axis shows the name of the freeways. The sensors' correlations within the same freeway are represented as red-shaded areas (block-diagonal elements of the matrix). The separate multiple DLMs only use block diagonal elements in the matrix.

from 325 sensors (Fig. 1(a)) on the freeways of San Francisco Bay area. The training and test dataset were constructed in the same way as [11], [17] to achieve a fair comparison.

The sampling interval of each dataset is 5 minutes by default, and in the following subsection, it is downsampled to 10 and 15 minutes, respectively, for a specific experiment. Both datasets of networks \mathcal{G}_1 and \mathcal{G}_2 contain 209 days of speed data, and each of those is divided into a training set and a test set at an 8:2 ratio by default. Another ratio is applied in Section IV-B for a specific experiment.

We used the root mean square error (RMSE) as an error metric to measure the accuracy of prediction since the solution in Eq. (30) is also the optimal under the minimum mean squares error (MMSE) sense [8]. The RMSE of a method with the prediction horizon h is defined as

$$\text{RMSE}(h, \text{method}) = \sqrt{\text{mean}(\mathbf{x}_{t+h|t}^{\text{method}} - \mathbf{x}_{t+h})^2}, \quad (31)$$

where the mean value is evaluated over all t in the test set.

For prediction horizons, we set from 5 minutes to 120 minutes every 5 minutes. In our previous work [8], on a freeway with a total length of about 60 miles (similar to the longest path of the networks considered here), the actual travel time is about 70 minutes under usual congestion. In the most severe congestion, the maximum travel time is about 100 minutes, and accordingly, we set the maximum prediction horizon to 120 minutes.

All datasets were normalized using the mean and standard deviation of each sensor in the training set. For a reference, we defined a baseline method that predicts future traffic features

assuming that the current traffic does not change over time, i.e., $\mathbf{x}_{x+h|t}^{\text{baseline}} = \mathbf{x}_t$.

B. Analysis of network prior

In this section, we show how network prior information contributes to predictive performance. Our model generalizes the DLM [8] to extend the model for a more extensive sensor network using the sensor's topology structure. When the DLM is simply used in an extensive network without topology structure information, an overfitting problem can occur. We introduce the three following setups to evaluate how well the proposed model utilizes the topology structure avoiding the overfitting problem,

- 1) a single DLM for the entire sensor network (without topological information),
- 2) separate DLMs ($K = 5$) for each freeway (block-diagonal DLM),
- 3) and the proposed model that is a single DLM ($K = 5$) with topological information.

As shown in Fig. 3, the proposed model shows the best performance, followed by block-diagonal DLM and single DLM without topological information. The proposed model induces the sensor's topological information through heat diffusion kernels to give weights to each element of this transition matrix and focus on estimating more essential components, resulting in it as a sparse matrix, as shown in Fig. 4. As a result, it shows excellent performance in long-term prediction by effectively estimating off-diagonal elements (correlation between signals of sensors installed on different freeways) while avoiding the overfitting problem. In the case of the model with a single DLM, all elements of this matrix are estimated using historical data, while in the case of the model with separate DLMs, only the block diagonal elements are estimated (red shaded area). Therefore, since the former one needs to estimate a much larger number of elements from the data than the latter, an overfitting problem may occur. In contrast, in the separate DLMs, the historical data cannot be fully utilized due to the lack of association between sensors belonging to different freeways. In particular, this insufficiency causes degradation of long-term predictions as congestion propagates slowly from one freeway to others.

The low prediction error is obtained only when the topological information is optimally implanted into the DLM. Bayesian inference in our model is the key component to support this process, as it optimally estimates various parameters that characterize the mixing ratio between data and prior, which respectively correspond to DLM and topological information. We set up the following experiment to find test the effectiveness of this estimation method:

- 1) the model with measurements (Eq. (2)),
- 2) the model with topological information (Eq. (15)),
- 3) and the model with both topological information and measurements (Eq. (19)).

For all the above models, we set three different cases that are characterized by different sizes of the training sets with the same test set.

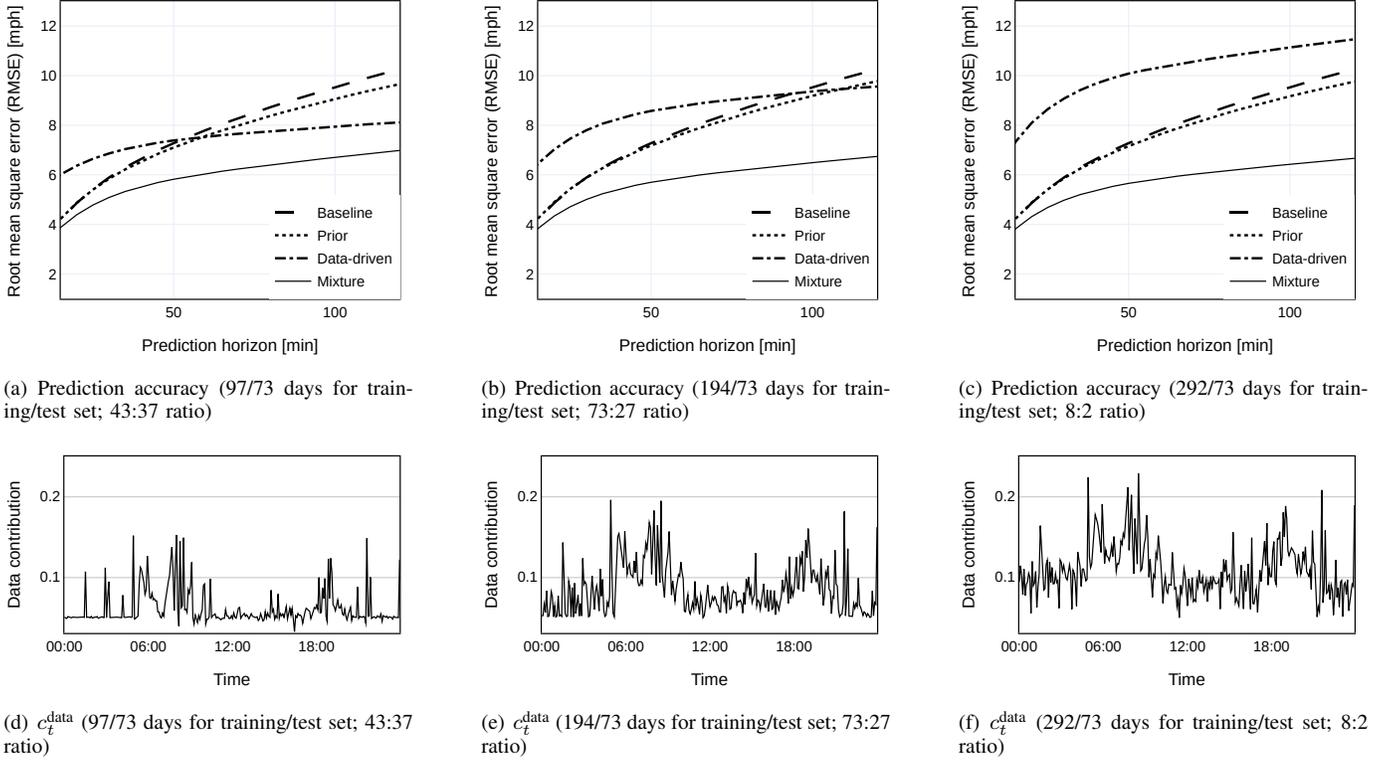


Fig. 5. Accuracy of the prediction and the data contribution for different training-test set ratio. The baseline method predicts future traffic features assuming that the current traffic does not change over time, i.e., $\mathbf{x}_{x+h|t}^{\text{baseline}} = \mathbf{x}_t$.

Figures 5(a)-(c) show the prediction accuracy of each case. Interestingly, the model with measurements produced smaller errors when the size of the training set is smaller. The reason is that each training set period is close to that of the test set with respect to time, which means larger training sets contain measurement that are far from those in the test sets. This may distort the inference process as traffic measurements have seasonal patterns. On the other hand, the model using only the topological information showed poor performance in predicting the far future because mixture kernels do not represent well the change in traffic conditions due to the volume preservation characteristic. The model with both topological information and measurements showed the best performance and similar outputs regardless of the size of the training set. It shows that Bayesian inference estimates parameters α_t and γ_t in Eq. (19) optimally, extracting maximal information both from data and prior.

Figures 5(d)-(f) show the data contribution which is defined in Eq. (20) of the mixture model. As the size of the training set increases, the data contribution increases since the larger training set can generalize measurements more easily. Another important aspect from the results is that the data contribution increases during peak periods such as morning and evening peaks since the traffic volume is most likely not preserved during these periods (therefore, it is difficult to explain it only with diffusion processes).

C. Analysis of different diffusion periods

We evaluated the proposed method with different diffusion processes (short, long, and mixture of both) in order to examine how the model of Eq. (9) performs in different settings. The transition matrix $\hat{\mathbf{H}}_t$ was set from Eq. (19) with three different diffusion priors:

- 1) $\mathbf{H}_t^{\mathcal{G}}(\mathcal{T}) = \mathbf{H}^{\mathcal{G}}(\tau_0) = \lim_{\tau \rightarrow 0} e^{-\tau \mathbf{L}(\mathcal{G})}$ (short diffusion kernel; identity mapping),
- 2) $\mathbf{H}_t^{\mathcal{G}}(\mathcal{T}) = \mathbf{H}^{\mathcal{G}}(\tau_\infty) = \lim_{\tau \rightarrow \infty} e^{-\tau \mathbf{L}(\mathcal{G})}$ (long diffusion kernel; averaging),
- 3) and $\mathbf{H}_t^{\mathcal{G}}(\mathcal{T}) = \pi_t^{(\tau_0)} \mathbf{H}^{\mathcal{G}}(\tau_0) + \pi_t^{(\tau_\infty)} \mathbf{H}^{\mathcal{G}}(\tau_\infty)$ (mixture of short and long diffusion kernels).

We also set three different cases that are characterized by different sampling intervals (T_s), 5, 10, and 15 minutes. The sampling interval indicates the time duration that corresponds to the one-time incremental (the difference between $t+1$ and t). The sampling interval is related to the diffusion period τ as a diffusion kernel expresses how traffic signals diffuse through a graph within a sampling interval.

Figures 6(a)-(c) show the prediction accuracy of each diffusion prior on the transportation network \mathcal{G}_1 with the three different sampling intervals. The predictor with the long diffusion process showed relatively poor performance compared to the baseline method for small prediction horizons, but it was improved when prediction horizons become larger. On the other hand, the one with the short diffusion process showed relatively good performance compared to the baseline method for all prediction horizons; however, it had insufficient

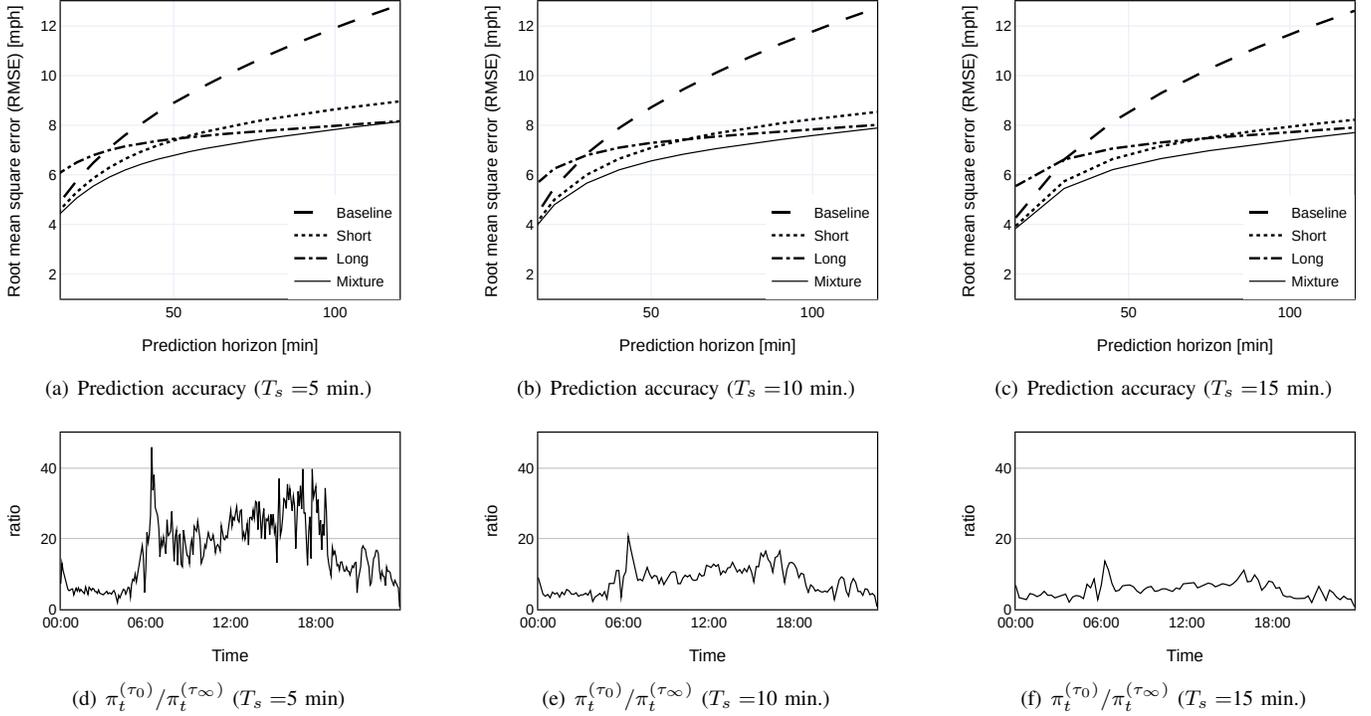


Fig. 6. Accuracy of the prediction and ratio of the short and long diffusion processes for the same test set with different time intervals. The baseline method predicts future traffic features assuming that the current traffic does not change over time, i.e., $\mathbf{x}_{x+h|t}^{\text{baseline}} = \mathbf{x}_t$.

performance for large prediction horizons compared to the one with the long diffusion process. The mixture model takes advantage of the two extreme cases, significantly improving the performance for both small and large prediction horizons. Specifically, around 50 minutes prediction horizon in Fig. 6(a), the performance of the mixture model is noticeably better than the others, meaning that a mixture of poor predictors can produce a good performance.

We emphasize that the distribution of the diffusion processes (Π_t) was determined optimally by Bayesian inference. Figures 6(d)-(f) show the ratio of the coefficients $\pi_t^{(\tau_0)}$ and $\pi_t^{(\tau_\infty)}$ in the mixture model that corresponds to the short and long diffusion processes, respectively. Although the short diffusion process dominates the whole process, as shown in the figures, the small portion of the long diffusion process contributes to the improvement. More importantly, the ratio becomes smaller when the sampling interval increases. It shows that Bayesian inference performs well in optimally determining parameters, since the performance of the mixture model stays similar when the sampling interval is changed.

We also emphasize that the ratio depends on time. For example, during the early morning, the diffusion kernel with long diffusion period (τ_∞) contributes more to the prediction performance although short diffusion (identity mapping) seems to be a more reasonable choice as there are few changes in traffic during that time. However, if the signal values are relatively uniform (in the case of a traffic signal at early morning), taking an average can remove noise while minimizing signal distortion as $\mathbf{x}_{t+1} \approx \mathbf{x}_t$ (identity) $\approx \frac{1}{N} \mathbf{1} \mathbf{1}^T \mathbf{x}_t$ (averaging; robust to noise).

D. Comparison with state-of-the-art technologies

We compare the proposed method with other methods using a benchmark dataset: PEMS-BAY dataset [11]. For a fair comparison, we use the same settings which are defined in [11] (also same in [19])². The models used for the comparison are as follows.

1) *FC-LSTM (Fully Connected Long Short-Term Memory)*: This model has been used as a representative reference for time-sequence modeling in deep learning [29]. In general, the LSTM module extracts correlations of signals farther apart in time than the RNN structure. However, this model's disadvantage is that spatial correlations can only be expected to learn directly from data as there is no separate module for extracting spatial relationships of signals. The RMSE score for PEMS-BAY dataset is retrieved from [11].

2) *STGCN*: Yu *et al.* [15] extracted spatial features with Graph Convolutional Neural Network (CNN) utilizing spectral graph convolution in graph theory. After that, they attached Gated CNN block to extract temporal features.

3) *DCRNN (Diffusion Convolution Recurrent Neural Network)*: Li *et al.* [11] constructed a successful predictor by extracting the signal's spatial features from the underlying graph structure by diffusion convolutional layers. Compared to STGCN, they designed the filter in the spatial domain directly rather than the graph spectral domain. The authors combine this diffusion module to Gated Recurrent Unit (GRU) which is a Recurrent Neural Network (RNN) variant.

4) *Graph WaveNet*: Xu *et al.* [19] improved DCRNN by using dilated 1D convolution (also called WaveNet) to

²Our code is available at: <https://github.com/semink/lslml/>

TABLE II
RMSE OF DIFFERENT METHODS
FOR PEMS-BAY DATASET.

Horizon	15 min	30 min	60 min
FC-LSTM [29]	4.19	4.55	4.96
DCRNN [11]	2.95	3.97	4.74
STGCN [21]	2.96	4.27	5.69
Graph WaveNet [19]	2.74	3.70	4.52
ST-MetaNet [30]	2.90	4.02	5.06
Proposed	2.90	3.77	4.44

TABLE III
COMPUTATION COSTS FOR TRAINING ON THE PEMS-BAY DATASET

Model	Training(s)
DCRNN [11]	750 (per epoch)
Graph WaveNet [19]	580 (per epoch)
Proposed	760 (total)

extract temporal features in terms of computation time and performance.

5) *ST-MetaNet*: Pan *et al.* [30] introduced graph attention network to extract spatial features. They utilize RNN architecture to extract temporal features.

Table II shows the RMSE of each model and our proposed method. We confirm that the performance of the proposed method reaches that of state-of-the-art methods based on a complex deep learning architecture. It even performs better for long-term prediction as we model based on DLM that explicitly expresses the daily periodicity of traffic signals. For example, the RMSEs of our proposed method for 90 and 120 min horizons are respectively 4.70 and 5.26, while these are 5.26 and 6.02 with the pre-trained DCRNN model.³

Our proposed method requires lower computational effort compared to the others. Also, it infers the majority of the parameters (N^2) analytically by Eq. (19). The method only requires numerical computation when it solves the optimization problem (25) to infer $K + 2$ parameters, which has $O(K^2)$ complexity, where K^2 is noticeably smaller than N^2 . Note that the hyperparameters are optimally estimated by solving the optimization problem (25) rather than the cross-validation method. As hyperparameter tuning is an expensive task, it can be a major advantage of the proposed method.

On the other hand, all state-of-the-art methods require heavy numerical computations to train a large number of parameters as they are based on deep-neural-net architectures. Our method successfully infers all parameters at the time scale of minutes with CPU computations, which is noticeably shorter than other DNN based methods with GPU computations as shown in Table III (note that the DNN based methods required from 50 epochs to 100 epochs to converge).

Another advantage of our model compared to the deep-learning-based architectures is that only a small number of parameters need to be decided heuristically. This can provide easy scalability to apply our model to other traffic datasets

³As GraphWaveNet predicts all the horizons at once (not recursive), we could not use the pre-trained model for the longer horizons. As a result, we choose DCRNN which shows the second-best result on 60 min horizon.

or datasets with similar properties to traffic data (daily periodicity). For example, in our model, the parameters to be determined before training are the threshold constant κ , the kernel width σ to build a proper graph, and the number of diffusion processes K to determine how many diffusion processes should be mixed. We empirically choose the constants κ and σ such that the corresponding graph \mathcal{G} is a k -vertex-connected graph with a small number k . For the number of diffusion processes K , we set $K = 5$ for the PEMS-BAY dataset but the prediction performance is not sensitive to the parameter (± 0.01 minutes changes of the RMSE score from $K = 3$ to $K = 7$).

V. CONCLUSION

In this paper, we proposed a method for predicting traffic signals in transportation sensor networks. We successfully integrated topological information of the sensor network into a data-driven model by assuming that the parameters in the model are supported by the mixture of diffusion kernels with uncertainties. We exploited the Bayesian inference to optimally determine the parameters that characterize the distribution of diffusion processes and the importance of measurements against prior information. The importance varies with time, and we discover that the data are relatively more important, especially for the peak period. Most importantly, the proposed method reached accurate prediction at the level of state-of-the-art methods with less computational effort. It particularly shows excellent performance in long-term predictions by exploiting DLM’s periodicity modeling. Our method can be applicable for predicting graph signals exhibiting daily patterns such as weather or energy consumption. For future works, we may improve the short-term prediction performance if we give more valuable prior information (e.g., graph structure more suitable for prediction; currently, it only depends on topology), or if it is possible to derive all inference processes (especially the marginalization steps in Eq. (40) and (35)) with a non-linear model overcoming the limitation of linear models.

REFERENCES

- [1] R. Wan, S. Mei, J. Wang, M. Liu, and F. Yang, “Multivariate temporal convolutional network: A deep neural networks approach for multivariate time series forecasting,” *Electronics*, vol. 8, no. 8, p. 876, 2019.
- [2] M. Das and S. K. Ghosh, “Sembnet: A semantic bayesian network for multivariate prediction of meteorological time series data,” *Pattern Recognition Letters*, vol. 93, pp. 192–201, 2017.
- [3] T. Ouyang, X. Zha, and L. Qin, “A combined multivariate model for wind power prediction,” *Energy Conversion and Management*, vol. 144, pp. 361–373, 2017.
- [4] K. Wang, K. Li, L. Zhou, Y. Hu, Z. Cheng, J. Liu, and C. Chen, “Multiple convolutional neural networks for multivariate time series prediction,” *Neurocomputing*, vol. 360, pp. 107–119, 2019.

- [5] S. Huang, D. Wang, X. Wu, and A. Tang, “Dsanet: Dual self-attention network for multivariate time series forecasting,” in *Proceedings of the 28th ACM international conference on information and knowledge management*, 2019, pp. 2129–2132.
- [6] S. R. Chandra and H. Al-Deek, “Predictions of freeway traffic speeds and volumes using vector autoregressive models,” *Journal of Intelligent Transportation Systems*, vol. 13, no. 2, pp. 53–72, 2009.
- [7] T. Mai, B. Ghosh, and S. Wilson, “Multivariate short-term traffic flow forecasting using bayesian vector autoregressive moving average model,” Tech. Rep., 2012.
- [8] S. Kwak and N. Geroliminis, “Travel time prediction for congested freeways with a dynamic linear model,” *IEEE Transactions on Intelligent Transportation Systems*, 2020.
- [9] L. Cavalcante, R. J. Bessa, M. Reis, and J. Browell, “Lasso vector autoregression structures for very short-term wind power forecasting,” *Wind Energy*, vol. 20, no. 4, pp. 657–675, 2017.
- [10] W. B. Nicholson, I. Wilms, J. Bien, and D. S. Matteson, “High dimensional forecasting via interpretable vector autoregression,” *Journal of Machine Learning Research*, vol. 21, no. 166, pp. 1–52, 2020.
- [11] Y. Li, R. Yu, C. Shahabi, and Y. Liu, “Diffusion convolutional recurrent neural network: Data-driven traffic forecasting,” in *International Conference on Learning Representations*, 2018.
- [12] Z. Cui, K. Henrickson, R. Ke, and Y. Wang, “Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting,” *IEEE Transactions on Intelligent Transportation Systems*, 2019.
- [13] C. Chen, K. Li, S. G. Teo, X. Zou, K. Wang, J. Wang, and Z. Zeng, “Gated residual recurrent graph neural networks for traffic prediction,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 485–492.
- [14] C. Zhang, J. James, and Y. Liu, “Spatial-temporal graph attention networks: A deep learning approach for traffic forecasting,” *IEEE Access*, vol. 7, pp. 166 246–166 256, 2019.
- [15] B. Yu, H. Yin, and Z. Zhu, “Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting,” in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 2018, pp. 3634–3640.
- [16] L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li, “T-gcn: A temporal graph convolutional network for traffic prediction,” *IEEE Transactions on Intelligent Transportation Systems*, 2019.
- [17] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, “Graph wavenet for deep spatial-temporal graph modeling,” in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, AAAI Press, 2019, pp. 1907–1913.
- [18] Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang, and C. Zhang, “Connecting the dots: Multivariate time series forecasting with graph neural networks,” in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 753–763.
- [19] B. Xu, H. Shen, Q. Cao, Y. Qiu, and X. Cheng, “Graph wavelet neural network,” in *International Conference on Learning Representations*, 2018.
- [20] S. Du, T. Li, Y. Yang, and S.-J. Horng, “Multivariate time series forecasting via attention-based encoder-decoder framework,” *Neurocomputing*, vol. 388, pp. 269–279, 2020.
- [21] H. Yao, F. Wu, J. Ke, X. Tang, Y. Jia, S. Lu, P. Gong, J. Ye, and Z. Li, “Deep multi-view spatial-temporal network for taxi demand prediction,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018.
- [22] L. Munkhdalai, T. Munkhdalai, K. H. Park, T. Amariyabagal, E. Batbaatar, H. W. Park, and K. H. Ryu, “An end-to-end adaptive input selection with dynamic weights for forecasting multivariate time series,” *IEEE Access*, vol. 7, pp. 99 099–99 114, 2019.
- [23] J. Du Preez and S. F. Witt, “Univariate versus multivariate time series forecasting: An application to international tourism demand,” *International Journal of Forecasting*, vol. 19, no. 3, pp. 435–451, 2003.
- [24] D. Helbing, “Traffic and related self-driven many-particle systems,” *Reviews of modern physics*, vol. 73, no. 4, p. 1067, 2001.
- [25] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, *et al.*, “Relational inductive biases, deep learning, and graph networks,” *arXiv preprint arXiv:1806.01261*, 2018.
- [26] R. I. Kondor and J. Lafferty, “Diffusion kernels on graphs and other discrete structures,” in *Proceedings of the 19th international conference on machine learning*, vol. 2002, 2002, pp. 315–22.
- [27] D. J. MacKay, “Bayesian interpolation,” *Neural computation*, vol. 4, no. 3, pp. 415–447, 1992.
- [28] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, “A limited memory algorithm for bound constrained optimization,” *SIAM Journal on scientific computing*, vol. 16, no. 5, pp. 1190–1208, 1995.
- [29] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [30] Z. Pan, Y. Liang, W. Wang, Y. Yu, Y. Zheng, and J. Zhang, “Urban traffic prediction from spatio-temporal data using deep meta learning,” in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 1720–1730.

APPENDIX

A. Volume conservation of mixture of heat diffusion

By definition (in Eq. (6)), the graph Laplacian $L(\mathcal{G})$ has an eigenvector $\frac{1}{\sqrt{N}}\mathbf{1}$ with the corresponding eigenvalue 0. Let an

eigen-decomposition of the matrix be

$$\mathbf{L}(\mathcal{G}) = \mathbf{V}\mathbf{D}\mathbf{V}^T, \quad (32)$$

where the orthonormal matrix \mathbf{V} and the diagonal matrix \mathbf{D} contain eigenvectors and corresponding eigenvalues, respectively. Since the orthonormal matrix \mathbf{V} contains the eigenvector $\frac{1}{\sqrt{N}}\mathbf{1}$,

$$\begin{aligned} \mathbf{1}^T \tilde{\mathbf{x}}_{t+1}^d(\tau) &\stackrel{(8)}{=} \mathbf{1}^T \mathbf{H}^{\mathcal{G}}(\tau) \mathbf{x}_t^d \\ &\stackrel{(5)}{=} \mathbf{1}^T e^{-\tau \mathbf{L}(\mathcal{G})} \mathbf{x}_t^d = \mathbf{1}^T \mathbf{V} e^{-\tau \mathbf{D}} \mathbf{V}^T \mathbf{x}_t^d \\ &= \frac{N}{\sqrt{N}} \frac{1}{\sqrt{N}} \mathbf{1}^T \mathbf{x}_t^d = \mathbf{1}^T \mathbf{x}_t^d. \end{aligned} \quad (33)$$

Therefore,

$$\begin{aligned} \mathbf{1}^T \tilde{\mathbf{x}}_{t+1}^d(\mathcal{T}) &= \mathbf{1}^T \mathbf{H}^{\mathcal{G}}(\mathcal{T}) \mathbf{x}_t^d = \mathbf{1}^T \left(\sum_{\tau} \pi(\tau) \mathbf{H}^{\mathcal{G}}(\tau) \right) \mathbf{x}_t^d \\ &= \sum_{\tau} \pi(\tau) \mathbf{1}^T \mathbf{H}^{\mathcal{G}}(\tau) \mathbf{x}_t^d = \sum_{\tau} \pi(\tau) \mathbf{1}^T \mathbf{x}_t^d \\ &= \mathbf{1}^T \mathbf{x}_t^d \sum_{\tau} \pi(\tau) = \mathbf{1}^T \mathbf{x}_t^d. \end{aligned} \quad (34)$$

B. Evidence

$$\begin{aligned} f(\mathbf{X}_{t+1} | \mathbf{X}_t, \alpha_t, \Pi_t) &= \int f(\mathbf{X}_{t+1} | \mathbf{X}_t, \mathbf{H}_t, \alpha_t) f(\mathbf{H}_t | \Pi_t) d\mathbf{H}_t \\ &\propto \int e^{-\frac{1}{2} \alpha_t \text{tr}\{(\mathbf{X}_{t+1} - \mathbf{H}_t \mathbf{X}_t)(\mathbf{X}_{t+1} - \mathbf{H}_t \mathbf{X}_t)^T\}} \\ &\quad \cdot e^{-\frac{1}{2} \gamma_t \text{tr}\{(\mathbf{H}_t - \mathbf{H}_t^{\mathcal{G}}(\mathcal{T}))(\mathbf{H}_t - \mathbf{H}_t^{\mathcal{G}}(\mathcal{T}))^T\}} d\mathbf{H}_t \\ &\propto e^{-\frac{1}{2} \alpha_t (\mathbf{X}_{t+1} (\mathbf{I} - \alpha_t \mathbf{X}_t^T \Sigma_t \mathbf{X}_t) \mathbf{X}_{t+1}^T - 2\gamma_t \mathbf{H}_t^{\mathcal{G}}(\mathcal{T}) \Sigma_t \mathbf{X}_t \mathbf{X}_{t+1}^T)} \\ &\quad \cdot \int (2\pi)^{-\frac{N^2}{2}} |\Sigma_t|^{-\frac{N}{2}} e^{-\frac{1}{2} \text{tr}\{(\mathbf{H}_t - \hat{\mathbf{H}}_t) \Sigma_t^{-1} (\mathbf{H}_t - \hat{\mathbf{H}}_t)^T\}} d\mathbf{H}_t \\ &\propto e^{-\frac{1}{2} \alpha_t (\mathbf{X}_{t+1} (\mathbf{I} - \alpha_t \mathbf{X}_t^T \Sigma_t \mathbf{X}_t) \mathbf{X}_{t+1}^T - 2\gamma_t \mathbf{H}_t^{\mathcal{G}}(\mathcal{T}) \Sigma_t \mathbf{X}_t \mathbf{X}_{t+1}^T)} \\ &\propto e^{-\frac{1}{2} \text{tr}\{\alpha_t (\mathbf{X}_{t+1} - \mathbf{H}_t^{\mathcal{G}}(\mathcal{T}) \mathbf{X}_t) (\mathbf{I} + \alpha_t \gamma_t^{-1} \mathbf{X}_t^T \mathbf{X}_t)^{-1} (\mathbf{X}_{t+1} - \mathbf{H}_t^{\mathcal{G}}(\mathcal{T}) \mathbf{X}_t)^T\}}, \end{aligned} \quad (35)$$

where $\Sigma_t^{-1} = \alpha_t \mathbf{X}_t \mathbf{X}_t^T + \gamma_t \mathbf{I}$.

C. Posterior of \mathbf{x}_{t+h}

When $h = 1$,

$$\begin{aligned} f(\mathbf{x}_{t+1} | \mathbf{x}_t, \mathbf{X}_{t+1}, \mathbf{X}_t) &= \int f(\mathbf{x}_{t+1} | \mathbf{x}_t, \mathbf{H}_t, \alpha_t) f(\mathbf{H}_t | \mathbf{X}_{t+1}, \mathbf{X}_t, \alpha_t, \gamma_t, \Pi_t, \mathcal{G}) d\mathbf{H}_t \\ &= f(\mathbf{x}_{t+1} | \mathbf{x}_t, \hat{\mathbf{H}}_t, \alpha_t) = \mathcal{N}(\hat{\mathbf{H}}_t \mathbf{x}_t, \alpha_t^{-1} \mathbf{I}). \end{aligned} \quad (36)$$

Assume the statement is true for $h = l - 1$ so that

$$f(\mathbf{x}_{t+l-1} | \mathbf{x}_t, \mathbf{X}_{t:t+l-1}) = \mathcal{N}(\hat{\mathbf{H}}_{t+l-2 \leftarrow t} \mathbf{x}_t, \mathbf{R}_{t+l-2}), \quad (37)$$

where $\hat{\mathbf{H}}_{t+l-2 \leftarrow t} = \hat{\mathbf{H}}_{t+l-2} \hat{\mathbf{H}}_{t+l-3} \cdots \hat{\mathbf{H}}_t$. By the chain rule,

$$\begin{aligned} f(\mathbf{x}_{t+l} | \mathbf{x}_t, \mathbf{X}_{t:t+l}) &= \int f(\mathbf{x}_{t+l} | \mathbf{x}_{t+l-1}, \mathbf{X}_{t+l-1}) f(\mathbf{x}_{t+l-1} | \mathbf{x}_t, \mathbf{X}_{t:t+l-1}) d\mathbf{x}_{t+l-1}. \end{aligned} \quad (38)$$

Since

$$\begin{aligned} f(\mathbf{x}_{t+l} | \mathbf{x}_{t+l-1}, \mathbf{X}_{t+l}, \mathbf{X}_{t+l-1}) &\stackrel{(36)}{=} \mathcal{N}(\hat{\mathbf{H}}_{t+l-1} \mathbf{x}_{t+l-1}, \alpha_{t+l-1}^{-1} \mathbf{I}), \\ f(\mathbf{x}_{t+l-1} | \mathbf{x}_t, \mathbf{X}_{t+l-1}, \dots, \mathbf{X}_t) &\stackrel{(37)}{=} \mathcal{N}(\hat{\mathbf{H}}_{t+l-2 \leftarrow t} \mathbf{x}_t, \mathbf{R}_{t+l-2}), \end{aligned} \quad (39)$$

$$\begin{aligned} f(\mathbf{x}_{t+l} | \mathbf{x}_t, \mathbf{X}_{t+l}, \dots, \mathbf{X}_t) &= \int \mathcal{N}(\hat{\mathbf{H}}_{t+l-1} \mathbf{x}_{t+l-1}, \alpha_{t+l-1}^{-1} \mathbf{I}) \\ &\quad \cdot \mathcal{N}(\hat{\mathbf{H}}_{t+l-2 \leftarrow t} \mathbf{x}_t, \mathbf{R}_{t+l-2}) d\mathbf{x}_{t+l-1} \\ &\propto \int \exp\left(-\frac{1}{2} \left\{ \alpha_{t+l-1} (\mathbf{x}_{t+l} - \hat{\mathbf{H}}_{t+l-1} \mathbf{x}_{t+l-1})^T \right. \right. \\ &\quad \cdot (\mathbf{x}_{t+l} - \hat{\mathbf{H}}_{t+l-1} \mathbf{x}_{t+l-1}) \\ &\quad \left. \left. + (\mathbf{x}_{t+l-1} - \hat{\mathbf{H}}_{t+l-2 \leftarrow t} \mathbf{x}_t)^T \mathbf{R}_{t+l-2} \right. \right. \\ &\quad \left. \left. \cdot (\mathbf{x}_{t+l-1} - \hat{\mathbf{H}}_{t+l-2 \leftarrow t} \mathbf{x}_t) \right\}\right) d\mathbf{x}_{t+l-1} \\ &\propto \exp\left(-\frac{1}{2} (\alpha_{t+l-1} \mathbf{x}_{t+l}^T \mathbf{x}_{t+l} \right. \\ &\quad \left. - (\alpha_{t+l-1} \hat{\mathbf{H}}_{t+l-1}^T \mathbf{x}_{t+l} + \mathbf{R}_{t+l-2}^{-1} \hat{\mathbf{H}}_{t+l-2 \leftarrow t} \mathbf{x}_t)^T \right. \\ &\quad \cdot (\alpha_{t+l-1} \hat{\mathbf{H}}_{t+l-1}^T \hat{\mathbf{H}}_{t+l-1} + \mathbf{R}_{t+l-2}^{-1})^{-1} \\ &\quad \left. \cdot (\alpha_{t+l-1} \hat{\mathbf{H}}_{t+l-1}^T \mathbf{x}_{t+l} + \mathbf{R}_{t+l-2}^{-1} \hat{\mathbf{H}}_{t+l-2 \leftarrow t} \mathbf{x}_t)\right) \\ &\propto \exp\left(-\frac{1}{2} \alpha_{t+l-1} \right. \\ &\quad \cdot (\mathbf{x}_{t+l}^T (\mathbf{I} - \alpha_{t+l-1} \hat{\mathbf{H}}_{t+l-1} (\alpha_{t+l-1} \hat{\mathbf{H}}_{t+l-1}^T \hat{\mathbf{H}}_{t+l-1} \\ &\quad \left. + \mathbf{R}_{t+l-2}^{-1})^{-1} \hat{\mathbf{H}}_{t+l-1}^T) \mathbf{x}_{t+l} \right. \\ &\quad \left. - 2 \mathbf{x}_{t+l}^T \hat{\mathbf{H}}_{t+l-1} (\alpha_{t+l-1} \hat{\mathbf{H}}_{t+l-1}^T \hat{\mathbf{H}}_{t+l-1} + \mathbf{R}_{t+l-2}^{-1})^{-1} \right. \\ &\quad \left. \cdot \mathbf{R}_{t+l-2}^{-1} \hat{\mathbf{H}}_{t+l-2 \leftarrow t} \mathbf{x}_t\right). \end{aligned} \quad (40)$$

Applying matrix inversion lemma, Eq. (40) becomes

$$\begin{aligned} &\exp\left(-\frac{1}{2} \alpha_{t+l-1} \right. \\ &\quad \cdot (\mathbf{x}_{t+l}^T (\mathbf{I} + \alpha_{t+l-1} \hat{\mathbf{H}}_{t+l-1} \mathbf{R}_{t+l-2} \hat{\mathbf{H}}_{t+l-1}^T)^{-1} \mathbf{x}_{t+l} \\ &\quad \left. - 2 \mathbf{x}_{t+l}^T (\mathbf{I} + \alpha_{t+l-1} \hat{\mathbf{H}}_{t+l-1} \mathbf{R}_{t+l-2} \hat{\mathbf{H}}_{t+l-1}^T)^{-1} \right. \\ &\quad \left. \cdot \hat{\mathbf{H}}_{t+l-1} \hat{\mathbf{H}}_{t+l-2 \leftarrow t} \mathbf{x}_t\right) \\ &\propto \exp\left(-\frac{1}{2} (\mathbf{x}_{t+l} - \hat{\mathbf{H}}_{t+l-1} \hat{\mathbf{H}}_{t+l-2 \leftarrow t} \mathbf{x}_t)^T \right. \\ &\quad \left. \cdot \mathbf{R}_{t+l-1}^{-1} (\mathbf{x}_{t+l} - \hat{\mathbf{H}}_{t+l-1} \hat{\mathbf{H}}_{t+l-2 \leftarrow t} \mathbf{x}_t)\right), \end{aligned} \quad (41)$$

where $\mathbf{R}_{t+l-1} = \alpha_{t+l-1}^{-1} \mathbf{I} + \hat{\mathbf{H}}_{t+l-1} \mathbf{R}_{t+l-2} \hat{\mathbf{H}}_{t+l-1}^T$ and by definition $\hat{\mathbf{H}}_{t+l-1 \leftarrow t} = \hat{\mathbf{H}}_{t+l-1} \hat{\mathbf{H}}_{t+l-2 \leftarrow t}$, so

$$f(\mathbf{x}_{t+l} | \mathbf{x}_t, \mathbf{X}_{t+l}, \dots, \mathbf{x}_t) = \mathcal{N}(\hat{\mathbf{H}}_{t+l-1 \leftarrow t} \mathbf{x}_t, \mathbf{R}_{t+l-1}). \quad (42)$$

Finally $\mathbf{x}_{t+h|t} = \hat{\mathbf{H}}_{t+h-1} \cdots \hat{\mathbf{H}}_t \mathbf{x}_t$.