2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

# Learning the dependency structure of highway networks for traffic forecast

Samitha Samaranayake , Sébastien Blandin , and Alexandre Bayen

***Abstract*— Forecasting road traffic conditions requires an accurate knowledge of the spatio-temporal dependencies of traffic flow in transportation networks. In this article, a Bayesian network framework is introduced to model the correlation structure of highway networks in the context of traffic forecast. We formulate the dependency learning problem as an optimization problem and propose an efficient algorithm to identify the *inclusion-optimal* dependency structure of the network given historical observations. The optimal dependency structure learned by the proposed algorithm is evaluated on benchmark tests to show its robustness to measurement uncertainties and on field data from the *Mobile Millennium* traffic estimation system to show its applicability in an operational setting.**

## I. INTRODUCTION

The increasing volume of urban traffic in the recent decades has led to the growth of massive recurrent congestion phenomena whose cost is estimated to be 87.3 billion dollars per year in the United States [24]. In order to reduce the direct costs associated with traffic congestion as well as the negative externalities it creates, accurate modeling and controls tools are required. The development of more accurate traffic models, real-time congestion estimates, forecast and routing algorithms can enable traffic managers to attain higher efficiency on the existing road infrastructure.

Traffic modeling is a well established discipline which dates back to classical macroscopic traffic models [9], [19], [22]. Macroscopic traffic models consider the dynamics of traffic flow using hydrodynamics theory. Classical estimation techniques [6], [16], have been shown to work efficiently with macroscopic traffic models [27], [29]. In particular, the recent growth of the number of smartphones equipped with sensing devices has provided the traffic estimation community with a large number of possible measurement sources [11], [28]. However, a significant portion of the research work has been considering the estimation problem for current traffic conditions (nowcast), and the forecast problem has been left relatively untouched.

Traffic forecast is an important problem, because accurate forecasts are a mandatory requirement for optimal routing on road networks, which is a major component of congestion mitigation. A crucial requirement for accurate traffic forecasting is the knowledge of how traffic states at different times and locations depend on each other. Most physical principles assumed by classical traffic models can be seen as defining information propagation rules. For scalar macroscopic models, the mass conservation principle yields a *partial differential equation* (PDE) which can be directly solved by *wavefront tracking* methods [1], or numerically by *Godunov* schemes [8]. Both of these methods provide an encoding of the spatio-temporal dependencies of the physical system. Unlike in the estimation problem, in which the model errors can be corrected via observations, the forecast problem relies heavily on an accurate model as the forecast error grows with the length of the forecast horizon [15]. Thus, knowing the exact dependency structure of the network is crucial in forecasting traffic conditions.

In this article, we propose to represent the evolution of traffic on the road network as a Bayesian network [13], [14], [21] and to learn its time-space dependencies using a structure learning algorithm. Bayesian networks provide an efficient framework to represent and work with dependencies in distributed systems, in particular to address fundamental problems such as learning and inference. This statistical framework allows us to learn the conditional dependencies that best fit the data and avoid the need for detailed assumptions regarding the physical system, which are often unknown or inaccurate. Thus, the proposed model is more flexible and allows for a variety of traffic phenomena to be accounted for, unlike some classical models. Once the dependency structure of the network is identified, it can be used to forecast the evolution of traffic conditions.

Several approaches can be considered for the dependency structure learning problem. In this article we propose to learn the physical dependency structure due to information propagation, and not the statistical dependency structure due to similar and concurrent traffic episodes without a cause-consequence relation. i.e. learning the dynamics of the system as opposed to pattern matching. We assume that the traffic state evolution is a Markov process and thus can be modeled by a Bayesian network with specific properties [14], [21]. In this setting, the structure learning algorithm identifies the most likely set of dependencies characterizing traffic flow.

The Bayesian network framework has been used for traffic applications, in [12], [18], for the estimation problem using different statistical methods, and in [26] for the forecast problem, but in all of these cases, the dependency structure is assumed to be known. To the best of our knowledge, there is no significant previous work on Bayesian network structure learning for traffic flow. In a wider context, the structure learning problem for Bayesian networks has been studied in [2], [3], [7], [23]. In this work we propose to adapt

S. Samaranayake is a Ph.D. student, Systems Engineering, Department of Civil and Environmental Engineering, University of California, Berkeley, CA 94720-1764 USA (e-mail: samitha@berkeley.edu). Corresponding author.

S. Blandin is a Ph.D. student, Systems Engineering, Department of Civil and Environmental Engineering, University of California, Berkeley, CA 94720-1764 USA (e-mail: blandin@berkeley.edu).

A. Bayen is an Associate Professor, Department of Electrical Engineering and Computer Sciences, and Department of Civil and Environmental Engineering, University of California Berkeley, CA 94720-1764 USA (e-mail: bayen@berkeley.edu).

the *greedy-equivalence search* (GES) algorithm developed in [3] to our specific problem. We show that reasonable traffic assumptions yield a much simpler version of the original algorithm. In particular we show that the dependency structure of the Bayesian network represents a causality relation of traffic flow.

The contributions of this article are as follows:

- A modeling framework for traffic flow based on Bayesian networks.
- A tractable approach to solve the structure learning problem for road networks in the context of traffic forecast. An adaptation of the GES algorithm [3] specific to the topological properties of the proposed Bayesian network.
- An implementation and evaluation of the method on field data collected in the San Francisco Bay Area.

The rest of the article is organized as follows: in Section II we recall the classical framework of Bayesian networks and introduce notations. Section III presents our Bayesian model of traffic and formulates the structure learning problem as an optimization problem. In Section IV we present and analyze experimental results obtained with real data from the *Mobile Millennium* traffic estimation system. Finally, Section V draws some conclusions and discusses further extensions to this work and related research topics.

## II. Bayesian networks

A Bayesian network is a directed acyclic graph in which each vertex represents a random variable and an edge between two vertices implies a dependence between the two random variables, where the strength of the influences is represented by conditional probabilities [21]. We consider a directed acyclic graph $\mathcal{G}(\mathcal{V},\mathcal{E})$ where $\mathcal{V} = \{\mathcal{V}_{n,t} | (n,t) \in \{1,\dots,N\} \times \{1,\dots,T\}\}$ is a set of vertices and $\mathcal{E}$ is a set of directed edges [14]. Each vertex $\mathcal{V}_{n,t}$ in the graph is naturally associated with a random variable representing the traffic velocity at a spatio-temporal location indexed by $n$ for the spatial dimension and by $t$ for the temporal dimension. A typical Bayesian network is illustrated in Figure 1. The graph

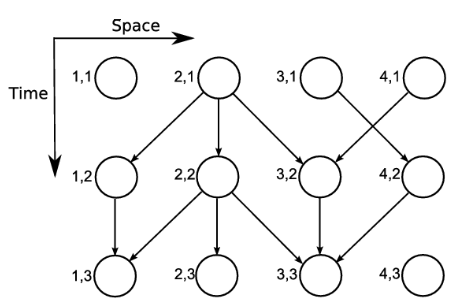


Fig. 1. **Bayesian network for vehicular traffic**: Traffic dependencies are modeled by a Bayesian network in which each vertex represents the distribution of traffic velocity at a given spatio-temporal location. The horizontal dimension corresponds to space and the vertical dimension corresponds to time.

structure encodes the *Markov conditions* [21] which state that each vertex $\mathcal{V}_{n,t}$ is independent of its non-descendants given its parents. The probability of a set of values $\mathcal{D} = (\nu_{1,1},\dots,\nu_{N,T})$ for a graph $\mathcal{G}$ with a parameter set $\Theta$ reads:

$$p(\mathcal{D}|\Theta,\mathcal{G}) = \prod_{n=1}^{N}\prod_{t=1}^{T} p(\nu_{n,t}|\pi_{n,t},\Theta),$$

where $\pi_{n,t}$ denotes the set of parents of the vertex $\mathcal{V}_{n,t}$, and $\mathcal{D}$ the data (observations). Realizations of random variables are denoted by lower-case letters. The term $\Theta$ denotes the parameters of the joint distribution of the graph. The *log-likelihood* for a dataset $\mathcal{D}$ on the graph $\mathcal{G}$ is given by:

$$l(\mathcal{D};\Theta,\mathcal{G}) = \sum_{n=1}^{N}\sum_{t=1}^{T} \log p(\nu_{n,t}|\pi_{n,t},\Theta). \tag{1}$$

In a typical learning framework, in which the conditional distributions are more often available, equation (1) conveniently expresses the likelihood of the data in terms of local conditional probabilities. If joint distributions are available, this expression can be transformed to:

$$l(\mathcal{D};\Theta,\mathcal{G}) = \sum_{n=1}^{N}\sum_{t=1}^{T} \log p(\nu_{n,t},\pi_{n,t}|\Theta) - \log p(\pi_{n,t}|\Theta),$$

where the first term is assumed to be known as the joint distribution of a vertex and its parents. The second term can be computed by marginalizing out the vertex $\mathcal{V}_{n,t}$ [14]. As illustrated in Figure 1, Bayesian networks are an efficient tool to represent complex dependency relations between random variables. However, Bayesian network theory depends on strong assumptions regarding the generative distributions being modeled, such as the symmetry, decomposition and intersection properties described in [21]. For example, the generative distribution of $\mathcal{G}$ is required to satisfy the *decomposition* axiom, which states that if $\forall Y \in \mathcal{Y}$, $X$ is independent of $Y$, then $X$ is independent of $\mathcal{Y}$ (with $X$, $Y$ being two nodes of the network and $\mathcal{Y}$ a set of nodes). This property is satisfied by the Gaussian distribution. However, it is not satisfied by most probability distributions (see [21] for an example and [17] for the general case). This motivates the choice of the Gaussian distribution as the generative distribution for our problem in the following section.

## III. Problem Statement

### A. *Traffic modeling*

We propose using a Bayesian network to model traffic flow by discretizing space and time into intervals of respective size $\Delta x$ and $\Delta t$. Each vertex of the Bayesian network has a spatio-temporal index $(n,t) \in \{1,\dots,N\} \times \{1,\dots,T\}$, and thus the Bayesian network has a grid form (Figure 1). The horizontal dimension represents space and the vertical dimension represents time. Each vertex $\mathcal{V}_{n,t}$ is associated with a probability distribution which represents the traffic velocity at the corresponding spatio-temporal location. A directed edge between two vertices implies a dependence between the corresponding velocity distributions. Our model makes the following assumptions.

*Assumption 1:* Traffic state evolution can be represented by a Markov process.

In particular the dependency structure learned only depends on the dynamics of the traffic state. The parameters of the dependency structure are the parameters of the state transition matrix of a Markov process. While this assumption ignores the recurrence of traffic conditions based on the time of day and day of week, the proposed framework can be easily extended to incorporate this information by conditioning on the expected traffic velocity or learning separate models for each congestion state.

*Assumption 2:* Traffic state evolution can be modeled as a linear function of past observations.

For tractability, we also assume that the velocity distributions of interest belong to the Gaussian family. In particular the Gaussian family is closed under the conditional operator, and conditional Gaussian distributions have expressions analytical in the parameters of the original distributions [5]. By using Gaussian distributions to model spatio-temporal traffic state dependencies, we assume that locally, traffic state evolution can be considered to be linear in the observations. According to the triangular model of traffic [4], [20], this is true within a single traffic phase (free flow or congestion).

The nature of traffic flow leads us to impose the following set of constraints $\mathcal{C}$ on the structure of a graphical representation of a traffic model.

$$\mathcal{C} = \begin{cases} \forall(\nu_{n,t}, \nu_{n',t}) \in \mathcal{V}^2,\ (\nu_{n,t}, \nu_{n',t}) \notin \mathcal{E} \\ \forall(\nu_{n,t}, \nu_{n',t'}) \in \mathcal{V}^2 \text{ s.t. } t < t', (\nu_{n',t'}, \nu_{n,t}) \notin \mathcal{E} \end{cases} \tag{2}$$

The first line in equation (2) states that there is no direct dependency relationship between the velocity on the highway at two different locations at the same time step. The second line in equation (2) states that the traffic state at a given time period can only impact the traffic state at a future time period, i.e. information does not propagate backwards in time.

*Remark 1:* Any graph $\mathcal{G}$ generated under the structure constraint $\mathcal{C}$ expressed by (2) is acyclic.

This feature is of crucial importance for the tractability of the structure learning algorithm described in the following section.

### *B. Structure learning*

Structure learning is a natural problem in Bayesian network theory [13], [14]. Given a set of vertices $\mathcal{V}$ and a set of constraints $\mathcal{C}$ on the Bayesian network, the structure learning problem consists of finding an optimal set of edges $\mathcal{E}$ and an optimal set of distribution parameters $\Theta$ for the application of interest. A standard approach to this problem is the maximization of a *score function*. In general, for correctness and computational tractability, a score function should satisfy the following properties:

- The *decomposability* property states that the score can be decomposed in a sum of local scores. This allows for efficient comparison of structures by local comparisons.
- The *asymptotic consistency* property states that in the limit of a large number of samples, the score function prefers the model with the fewest number of parameters. A more formal definition can be found in [3].
- The *local consistency* of the score guarantees that a locally greedy structure search nudges the model in an optimal direction. If two vertices $\mathcal{V}_j$ and $\mathcal{V}_i$ are not independent from each other given the parents of $\mathcal{V}_i$ in the generative distribution, then adding the edge $\mathcal{V}_j \to \mathcal{V}_i$ increases the value of the score function. If they are independent given the parents of $\mathcal{V}_i$, adding the edge $\mathcal{V}_j \to \mathcal{V}_i$ decreases the score function.

In our algorithm, we use the *Bayesian Information Criterion* (BIC) score [10], [25], which satisfies all of the conditions listed above and is one of the most commonly used score functions. The BIC score is given by:

$$S_{BIC}(\mathcal{D}, \mathcal{G}, \Theta) = \log P(\mathcal{D}|\hat{\Theta}, \mathcal{G}) - \frac{d}{2}\log m + \mathcal{O}(1) \tag{3}$$

where $\mathcal{D}$ is the data, $\hat{\Theta}$ is the maximum likelihood distribution parameters for $\mathcal{D}$, $d$ is the number of edges, and $m$ is the sample size per vertex. The BIC score is an approximation of the *Bayes-Dirichlet* score [25], which expresses the posterior probability of the network parameters $\Theta$ given the data $\mathcal{D}$.

Our structure learning problem can be formulated as an optimization problem. Given a set of vertices $\mathcal{V}$, a set of constraints $\mathcal{C}$ (2) on a network structure, a score function $\mathcal{S}_{BIC}$ (3), and a dataset $\mathcal{D}$, we consider the following problem:

$$\begin{aligned} \max_{\Theta, \mathcal{E}} \quad & \mathcal{S}_{BIC}(\mathcal{D}, \mathcal{G}(\mathcal{V}, \mathcal{E}), \Theta) \\ \text{subject to:} \quad & \mathcal{G}(\mathcal{V}, \mathcal{E}) \text{ satisfies } \mathcal{C} \end{aligned} \tag{4}$$

Solving this problem gives us the structure of the Bayesian network most likely to explain the observed data given the modeling constraints. In the following section we describe an algorithm for solving this optimization problem efficiently. The solution of this optimization problem consists of the set of edges and the distribution parameters which are the closest to the generative distribution in the *BIC* score sense.

### *C. Learning algorithm*

In general, finding the optimal Bayesian network structure given a dataset $\mathcal{D}$ requires searching through the entire *directed acyclic graph* (DAG) space and selecting the graph $\mathcal{G}$ with the highest score function value $S(\mathcal{D}, \mathcal{G}, \times)$. This is clearly intractable given that the size of the DAG space is super-exponential in the number of vertices.

*Definition 1:* We define a valid *topological ordering* to be an ordering where the vertices are sorted by ascending time index.

Given the constraint set $\mathcal{C}$ (2) and the resulting property stated in Remark 1, we can reduce the DAG space to a set of topological orderings of the vertices that satisfy the constraints.

*Lemma 1:* All topological orderings are equivalent. As stated in the constraint set $\mathcal{C}$, there are no dependencies between vertices with the same time index, thus all topological orderings allow for the same set of dependencies.

As a result, the DAG space can be limited to a single topological ordering of the vertices.

The restricted DAG space given the constraint set $\mathcal{C}$ is however exponential in the number of vertices in the network. While the general structure learning problem is this setting is hard, more tractable solutions exist for finding an *inclusion-optimal* model. An inclusion-optimal model is a Bayesian network that contains the generative distribution and has no subgraph with the same set of independence relations[1]. In general, the inclusion-optimal structure must be found by searching over all topological orderings. However, as explained above, given the constraints of our problem it is sufficient to consider one arbitrary valid topological ordering.

**Algorithm 1** *Greedy equivalence search algorithm for traffic modeling*. Forward phase: edge additions which maximize the increase of the score function. Backward phase: Edge removals which do not decrease the score function.

Define the edges $\mathcal{E} = \emptyset$
Define topological ordering satisfying the constraints $\mathcal{C}$
*Forward phase*
**for** each vertex $\mathcal{V}_{n,t}$ **do**
  **for** each candidate parent $\tilde{\pi}_{n,t}$ **do**
    Compute the score of the graph with the additional edge $(\mathcal{V}_{n,t}, \tilde{\pi}_{n,t})$
    Keep the local structure maximizing the score
  **end for**
**end for**
*Backward phase*
**for** each vertex $\mathcal{V}_{n,t}$ **do**
  **for** each parent $\pi_{n,t}$ **do**
    Compute the score of the graph with the edge $(\mathcal{V}_{n,t}, \pi_{n,t})$ removed
    Keep the local structure maximizing the score
  **end for**
**end for**

We adapt the *greedy equivalence search* (GES) algorithm[2] from [3] that finds an inclusion-optimal Bayesian network to solve our structure learning problem. The pseudo-code for the modified version of this greedy search algorithm is given in Algorithm 1. The algorithm consists of two phases, a *forward phase* with edge additions and a *backward phase* of edge removals. Edge additions and removals are performed only when they increase the score function. The set of *candidate parents* of a vertex is a parameter of the algorithm which denotes the set of possible parents considered in the forward phase of the algorithm. In Section IV (Figure 3) we analyze the sensitivity of the algorithm to this parameter.

The simplified version of the greedy equivalent search algorithm from [3] is used to solve the optimization problem (4).

[1] This is not necessarily optimal in the general sense as there might be a graph with fewer edges that also contains the generative distribution.

[2] In addition to the properties listed in Section III-B this algorithm also requires that $\mathcal{G}$ satisfies the path property [3] which expresses that a dependency in $\mathcal{G}$ can be characterized by the existence of a path between vertices.

The search algorithm given in Algorithm 1 has a manageable complexity that is linear in its inputs. The complexity is equal to the product of the sample size $|\mathcal{D}|$, the number of vertices in the Bayesian network $|\mathcal{V}|$, and the size of the candidate parents set. It should be noted that this structure learning procedure easily lends itself to be distributed over a parallel computing framework, due to the *local consistency* property of the score function.

## IV. Results

In this section, we present results on sensitivity and robustness to noise with a benchmark dataset. We also study the velocity forecast accuracy and its impact on route choice with experimental traffic data from the *Mobile Millennium* traffic estimation system.

### A. Benchmark tests

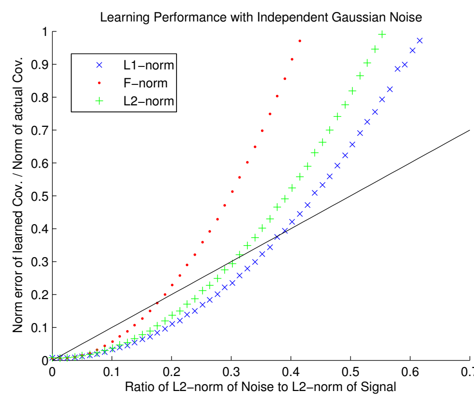


Fig. 2. **Stability of the structure learning algorithm**: The relative error in the covariance matrix of the joint distribution increases with respect to the relative $L^2$ norm of the noise with respect to the data. The algorithm is able to reconstruct the covariance structure reasonably well for relative $L^2$ norm of the noise lower than $0.2$.

We study the performance of the structure learning algorithm for two criteria; first the robustness to noise in the training set, second the impact of the optimal number of parents on the forecast accuracy with synthetic data.

In order to study the robustness of our structure learning algorithm, we consider a dataset generated from a probability distribution corresponding to the Bayesian network detailed in Figure 1. We add independent and identically distributed Gaussian noise to each observation in the network. This is assumed to model a real life situation, in which the data does not correspond exactly to a given generative distribution, but is subject to sensing, processing, modeling errors. The results from Figure 2 show that the structure learning algorithm is stable (error in the structure of the joint covariance matrix learned is lower than the error in the signal) up to reasonably high relative values of noise to signal ratio.

We also analyze the sensitivity of the forecast accuracy to the number of parents using synthetic data from the graph from Figure 1. The complexity of the algorithm increases linearly with the size of the candidate parents. Therefore, it is important to understand the trade-off between accuracy and

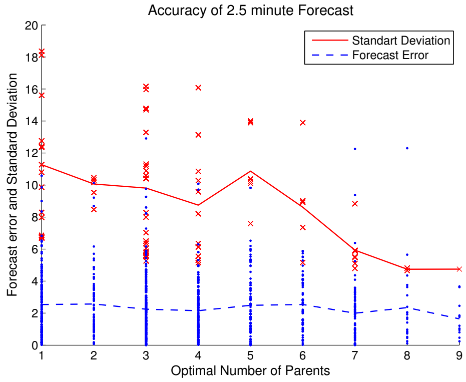


Fig. 3. **Dependency to optimal number of parents**: Traffic state forecast at locations which depend on a high number of parents have the same forecast mean error in average (dashed blue line) but a smaller standard deviation (continuous red line).

computation time. The results from Figure 3 show that the accuracy of the forecast mean is not significantly impacted by the optimal number of parents. However, the forecast standard deviation error decreases as a function of the number of parents. More information leads to a tighter estimate, which is what would be expected from a statistical estimate that is unbiased and consistent. This is of particular importance for applications which focus on distributed quantities such as travel time. Forecast results with real data from the *Mobile Millennium* system are presented in the following section.

## B. Forecast accuracy

The algorithm is tested in a operational setting with experimental data from the *Mobile Millennium* system on a 10-mile section of I-880 in the Bay Area, California (Figure 4). Experimental data consists of traffic velocity estimate at a $1/4$ mile, 30-second resolution. We learn the optimal structure of the Bayesian network on data corresponding to 2 hours of the morning rush on February 1st, 2010. The forecast accuracy of the algorithm is assessed by comparing the forecast given by our method on February 2nd, 2010 with the output of the *Mobile Millennium* system over the same time period. The random variables corresponding to

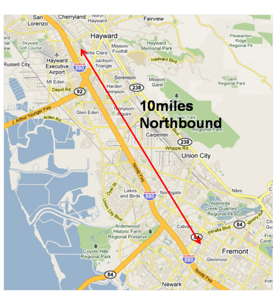


Fig. 4. **Experimental data location**: We consider the *Mobile Millennium* highway model output on a 10-mile long stretch of I-880 in the Bay Area.

each vertex in the Bayesian network represent a mean traffic velocity for a 2.5 minute interval over a $1/4$ mile road segment. The Markov structure has a spatio-temporal extent of 10 miles and 30 minutes, i.e. in the forecast phase, knowledge of traffic state up to 30 minutes before and 5 miles upstream and downstream can be considered to compute the forecast. Results show that the performance of the algorithm does not depend on the sample size for a vertex sample size of at least 100 in the training set, which given the dataset available corresponds to about 1 hour of data.

A 10 minutes velocity profile forecast is presented in Figure 5. Typical features of traffic such as queue locations are correctly forecast. However, the forecast cannot capture the full variability of traffic phenomena observed during this time. We believe this is in part due to the modeling of non-linear traffic dynamics using a linear dependency relationship as stated in Assumption 2.

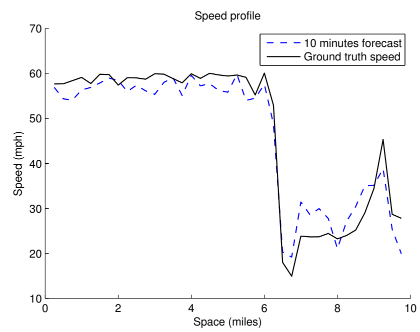


Fig. 5. **10 minutes velocity profile forecast**: The velocity Bayesian forecast (dashed blue line) captures the space variation of ground-truth traffic velocity (solid black line). The forecast tends to have lower variability than the ground truth.

## C. A tool for real-time routing

The forecast of traffic conditions has a great value for real-time traffic routing applications. However, most of the existing routing engines neither account for current traffic conditions nor incorporate forecast based on current traffic conditions. Common routing estimates are either based on purely static quantities or on historical estimates of the traffic conditions. In Figure 6, we consider a 2.5-mile segment of I-880 in the Bay Area, California, and compute a travel time forecast at a 10 minute horizon by two different means. The first one assumes that traffic conditions are stationary and forecast the exact current conditions, whereas the second forecast is given by our Bayesian algorithm for traffic states aggregated at a 2.5 minute, $1/4$ mile resolution. We convert this velocity forecast to a travel time forecast.

As seen in Figure 6 the accuracy of the naive approach of equating future traffic conditions to the current traffic conditions is reasonable when traffic conditions do not change significantly. However, over the entire period of interest, our Bayesian travel time forecast method performs better. This is also related to the result presented in Figure 3. The travel time with the simplistic method can be quite good on average, but since it does not use the conditional knowledge of a large set of vertices, its variance is larger. This has a negative impact on the travel time accuracy for which errors are accumulated over the path.

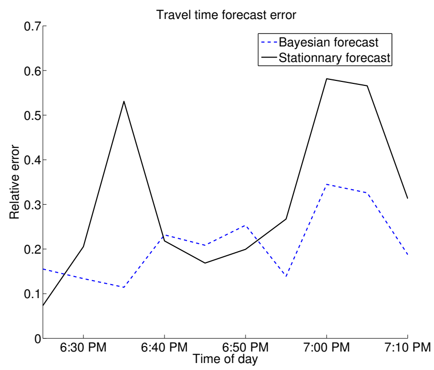


Fig. 6. **10 minutes forecast with Bayesian network**: Forecast travel time relative error computed from Bayesian velocity forecast (solid blue line) and from current conditions (dashed red line). The travel time forecast relying on the stationarity of traffic is not always accurate and in average does not perform as well as the Bayesian forecast method that we develop.

## V. CONCLUSION

We considered the problem of learning a Bayesian network structure for accurate traffic forecast in a highway network. Under some reasonable physical and statistical assumptions, we implemented a tractable structure learning algorithm which provides an inclusion-optimal solution to the problem. Robustness analysis with respect to noise in the learning set was conducted with synthetic data, and forecast accuracy was assessed using real-time traffic estimates from the *Mobile Millennium* system. Experimental results show that this approach accurately identifies the changes in traffic state for short forecast horizons.

As alluded to previously, the accuracy of the proposed method is limited by attempting to forecast a non-linear system using a linear Gaussian model. While the system is non-linear, empirical data and discretization methods for the PDE based solutions such as the *Godunov* scheme [8] show that the dynamics is linear when conditioned on the density of vehicles on the road. This implies that our model could be extended to cluster the data by traffic state and solve the structure learning problem for each cluster individually. The forecast can then be obtained based on which cluster is most likely given the data. This is a topic we are currently exploring and expect to improve the forecast accuracy with.

## REFERENCES


[1] A. BRESSAN. *Hyperbolic systems of conservation laws: the one-dimensional Cauchy problem*. Oxford University Press, Oxford, UK, 2000.

[2] D. CHICKERING. Learning equivalence classes of Bayesian network structures. *The Journal of Machine Learning Research*, 2:445–498, 2002.

[3] D. CHICKERING and C. MEEK. Finding optimal Bayesian networks. In *Proceedings of the Eighteenth Annual Conference on Uncertainty in Artificial Intelligence*, pages 94–102, San Francisco, CA, 2002. Morgan Kaufmann.

[4] C. DAGANZO. The cell transmission model: a dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B*, 28(4):269–287, 1994.

[5] R. DURRETT. *Essentials of stochastic processes*. Springer, New York, USA, 1999.

[6] G. EVENSEN. *Data assimilation: The ensemble Kalman filter*. Springer, New-York, NY USA, 2009.

[7] N. FRIEDMAN, K. MURPHY, and S. RUSSELL. Learning the structure of dynamic probabilistic networks. In *Proceedings of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence*, pages 139–14, San Francisco, CA, 1998. Morgan Kaufmann.

[8] S. GODUNOV. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. *Matematicheskii Sbornik*, 89(3):271–306, 1959.

[9] B. GREENSHIELDS. A study of traffic capacity. *Proceedings of the Highway Research Board*, 14(1):448–477, 1935.

[10] D. HAUGHTON. On the choice of a model to fit data from an exponential family. *The Annals of Statistics*, 16(1):342–355, 1988.

[11] J.-C. HERRERA, D. WORK, X. BAN, R. HERRING, Q. JACOBSON, and A. BAYEN. Evaluation of traffic data obtained via gps-enabled mobile phones: the mobile century field experiment. *Transportation Research Part C*, 18:568–583, 2009.

[12] R. HERRING, A. HOFLEITNER, S. AMIN, T. ABOU NASR, A. ABDEL KHALEK, P. ABBEEL, and A. BAYEN. Using mobile phones to forecast arterial traffic through statistical learning. In *89th Transportation Research Board Annual Meeting, Washington D.C., January 10-14*, 2010.

[13] F. JENSEN and T. NIELSEN. *Bayesian networks and decision graphs*. Springer, New-York, NY, USA, 2007.

[14] M. JORDAN. *Learning in graphical models*. MIT Press Cambridge, MA, USA, 1999.

[15] J. KAIPIO and E. SOMERSALO. *Statistical and computational inverse problems*. Springer, New-York, NY, USA, 2005.

[16] R. KALMAN. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.

[17] C. KANG and J. TIAN. Markov properties for linear causal models with correlated errors. *The Journal of Machine Learning Research*, 10:41–70, 2009.

[18] J. KWON and K. MURPHY. Modeling freeway traffic with coupled HMMs. 2000.

[19] M. LIGHTHILL and G. WHITHAM. On kinematic waves II a theory of traffic flow on long crowded roads. *Proceedings of the Royal Society of London*, 229(1178):317–345, 1956.

[20] G. NEWELL. A simplified theory of kinematic waves in highway traffic, II: Queueing at freeway bottlenecks. *Transportation research Part B*, 27(4):289–303, 1993.

[21] J. PEARL. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 1988.

[22] P. RICHARDS. Shock waves on the highway. *Operations Research*, 4(1):42–51, 1956.

[23] M. SCHMIDT, A. NICULESCU-MIZIL, and K. MURPHY. Learning graphical model structure using L1-regularization paths. In *Proceedings of the 22nd national conference on Artificial intelligence-Volume 2*, pages 1278–1283. AAAI Press, 2007.

[24] D. SCHRANK and T.LOMAX. The 2009 urban mobility report. *Texas Transportation Institute*, College Station, TX, 2009.

[25] G. SCHWARZ. Estimating the dimension of a model. *The annals of statistics*, pages 461–464, 1978.

[26] S. SUN, C. ZHANG, and G. YU. A Bayesian network approach to traffic flow forecasting. *IEEE Transactions on Intelligent Transportation Systems*, 7(1):124–132, 2006.

[27] Y. WANG and M. PAPAGEORGIOU. Real-time freeway traffic state estimation based on extended Kalman filter: a general approach. *Transportation Research Part B*, 39(2):141–167, 2005.

[28] D. WORK and A. BAYEN. Impacts of the mobile internet on transportation cyberphysical systems: traffic monitoring using smartphones. In *National Workshop for Research on High-Confidence Transportation Cyber-Physical Systems: Automotive, Aviation and Rail*, Washington, D.C., 2008.

[29] D. WORK, S. BLANDIN, O.-P. TOSSAVAINEN, B. PICCOLI, and A. BAYEN. A traffic model for velocity data assimilation. *AMRX Applied Mathematics Research eXpress*, 2010.