

Hidden Markov Flow Network Model: A Generative Model for Dynamic Flow on a Network

Satoshi Koide, Hiroshi Ohno, Ryuta Terashima, Thanomsak Ajjanapanya, and Itti Rittaporn

1

Abstract—In this paper, we propose a generative model that describes the dynamics of flow on a network — the hidden Markov flow network (HMFN) model, which is inspired by the gravity model in traffic engineering. Each node in the network has a dynamic hidden state and the flow observed on links depends on the states of the nodes being connected. For model inference, a collapsed Gibbs sampling algorithm is also proposed. Lastly, the model is applied to synthetic data and real human mobility network generated by GPS data from taxis in Bangkok. The synthetic data example shows that the reconstruction accuracy of the proposed method outperforms compared with the k -means method and the hidden Markov model, which do not consider the network interaction. The results of human mobility data show that the HMFN model can be used for spatio-temporal anomaly detection and prediction of future flow patterns.

Index Terms—Generative model, dynamics of flow network, bayesian inference, spatio-temporal pattern mining.

I. INTRODUCTION

In this paper, we consider the dynamics of flow on a network that is typical of human mobility/vehicle traffic patterns. According to Gonzalez *et al.* [1], macroscopic regularity exists in seemingly random mobility patterns. They have shown that extracting hidden patterns from a large collection of human mobility data is worthwhile. In particular, if more detailed patterns can be found, they can be useful in such machine learning tasks as classification, prediction, and anomaly detection in a network. Our purpose in the present research is to develop a generative model that reveals the statistical structure of network flow dynamics.

Human mobility patterns are often represented as an origin-destination matrix (OD matrix) in which the (i, j) element is the number of people (or vehicles) moving from location i to location j . Traditionally, time dependent OD matrices were estimated by methods such as data assimilation using observed traffic counts on roads [2], [3]. Recently, however, because more and more GPS and communication devices are being equipped on vehicles, we can obtain a large, high-resolution dynamic OD matrix which reflects spatio-temporal mobility patterns. By using tools like traffic simulation with observed OD matrix, it would be expected that enough result is obtained. On the other hand, those approaches do not clarify the latent structure behind the OD

matrix. Here the term “latent structure” means non-stationary and unobserved factors such as concerts, football games, and sudden rain. It is natural to consider that these latent factors cause co-occurrence of mobility demand and shift of the statistical structure of an OD matrix.

In this paper, we treat the flow dynamics of a given network. In the case of a dynamic OD matrix, nodes represent locations and links count the number of trips between two locations. We consider the situation in which the observed flow is sparse (most matrix elements are zero) and has a high spatio-temporal resolution (i.e. the matrix is high dimensional).

It is difficult to understand such data by direct visualization. Therefore, some unsupervised learning method is needed; however, basic clustering methods or pattern extraction methods such as k -means or principal component analysis would be difficult to apply for the following reasons. Firstly, natural flow patterns are expected to be continuous with respect to time. To obtain robustness against sparseness, we should assume that the flow patterns depend on time. The second point is that the non-stationary events discussed above occur independently at a number of geographic points. This fact makes the problem complicated. Even if we assume that the latent variables of each region take only binary values (for example, whether an event is held or not), the number of possible combinations increases exponentially and the latent structure behind the network flow becomes intractable. One approach to handling this curse of dimensionality is to apply the traditional unsupervised learning method to node-local flow, i.e. using flow around the target node only. However, in such cases the number of combinations can potentially increase and a robust result would not be obtained. This consideration leads to the hidden Markov network flow (HMFN) model that we present in this paper.

In order to deal with an essentially exponential number of latent structures, we assume that the observed flow on a link depends on the hidden states of both its beginning and terminating nodes. Rather than regarding the flow dynamics as one huge state space model of the entire network, we consider it as multiple locally interconnected state space models on the network. The reason for this assumption comes from the *gravity model*, which is often used in traffic engineering and trade economics [4]. The gravity model says that the flow on a link depends on the “mass” of its connecting nodes. The purpose of the gravity model is to detect unknown flow on a link based on entropy maximization [4]. Although the setting of the gravity model is different from that of our problem, the assumption on link flow is common and would be reasonable.

The contribution of the present study is threefold. Firstly, we introduce HMFN model, a novel generative model that describes dynamic flow pattern on a network. Secondly, we

Manuscript received February 20, 2014; revised May 14, 2014.

Satoshi Koide, Hiroshi Ohno, and Ryuta Terashima are with Toyota Central R&D Laboratories, Nagakute, Aichi 480-1192, Japan (e-mail: koide@mosk.tytlabs.co.jp, oono-h@mosk.tytlabs.co.jp, ryuta@mosk.tytlabs.co.jp).

Thanomsak Ajjanapanya and Itti Rittaporn are with Toyota Tsusho Electronics (Thailand) Co., Ltd., Pathumwan, Bangkok 10330, Thailand (e-mail: thanomsak@ttet.co.th, itti@ttet.co.th).

propose an inference method for the HMFN model. Finally, we apply the proposed model to real dataset and confirm that it describes human mobility pattern well.

The paper is organized as follows. In Section II, we introduce notation and describe the HMFN model. In Section III, an algorithm for inference by collapsed Gibbs sampler is presented. In Section IV, numerical results for artificial data and real human mobility data from taxi probes in Bangkok are shown. Finally, we present a discussion and conclusion in Section V and VI.

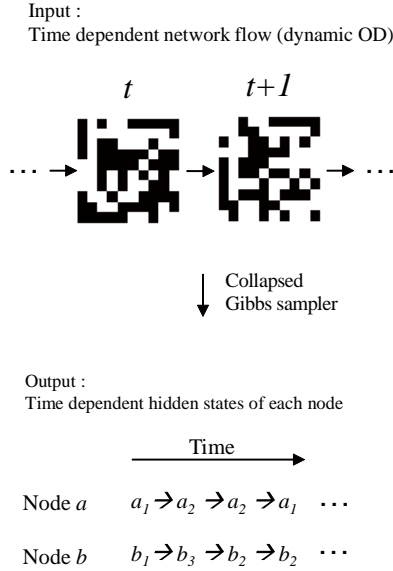


Fig. 1. Input and output of the HMFN model. The model discovers hidden dynamics of each node by using given dynamic network flow shown as matrix in this diagram.

II. HIDDEN MARKOV FLOW NETWORK MODEL

The HMFN model is a generative model on a given directed graph $G = (V; E)$ that describes the stochastic dynamics in the set of nodes V has its own hidden state dynamics. For each link in the set of links E , observations are emitted according to the hidden state of nodes. $E_{out}(i)$ and $E_{in}(i)$ are the set of out- and in-links to node i , respectively. If self link $e = i \rightarrow i$ exists, we define $e \in E_{out}(i)$ and $e \notin E_{in}(i)$. The input and output of the model is shown in Fig. 1. The main purpose of the model is to discover time series of hidden states in each node when the dynamic network flow dataset is given.

Before describing the model, we introduce the notation: x_e^t is observed flow on link $e \in E$ at time t ($1 \leq t \leq T$). $\mathbf{x}_e^{1:T}$ is the observed time series on link e and $\mathbf{x}^{1:T}$ denotes all observations. Z_i is a set of hidden states in node $i \in V$. Notice that, in general, $|Z_i|$ is not necessarily equal to $|Z_j|$ when $i \neq j$. $z_i^t \in Z_i$ is the hidden state of node i at time t . $\mathbf{z}_i^{1:T}$ is the time series of hidden states in node i and $\mathbf{z}^{1:T}$ denotes all hidden states. π_i is the initial distribution of the state of node i . $b(e)$ and $t(e)$ denote the beginning and terminating nodes of link e respectively.

The HMFN assumes the following generative process for flow dynamics $\mathbf{x}^{1:T}$:

- For each node $i \in V$:
 - Draw $|Z_i| \times |Z_j|$ transition matrix P_i . Each row of P_i obeys Dirichlet(α_i).
- For each link $e \in E$, repeat the following:
 - For each $z_1 \in Z_{b(e)}$ and $z_2 \in Z_{t(e)}$:
 - $\lambda_e^{z_1 \rightarrow z_2} \sim \text{Gamma}(\theta; k)$
- For $t = 1 \dots T$, repeat followings:
 - For each node $i \in V$:
 - If $t = 1$, $z_i^t \sim \text{Categorical}(\pi_i)$
 - Otherwise $z_i^t \sim \text{Categorical}(P_{i, z_i^{t-1}})$
 - For each link $e \in E$:
 - $x_e^t \sim \text{Poisson}(\lambda_e^{z_{b(e)}^t \rightarrow z_{t(e)}^t})$

The graphical representation of the HMFN for small network is represented in Fig. 2. As mentioned in the previous section, x_e^t depends on the states of $b(e)$ and $t(e)$ at time t . $\mathbf{z}_i^{1:T}$ obey a Markov chain with transition matrix P_i . Here $P_{i,z}$ denotes the transition probability for the next state when the current state is z in node $i \in V$. We assume the initial state distribution π_i of node i follows a uniform distribution in this study although it can be also regarded as a sample from a Dirichlet distribution. Notice that, although $\mathbf{z}_i^{1:T}$ and $\mathbf{z}_j^{1:T}$ (for $i \neq j$) are generated independently, they depend on each other once we observe the flow between i and j . The hyperparameters of the model are θ , k , and $\alpha = \{\alpha_i\}$. While we used the Poisson distribution and gamma distribution in the above formulation, the model can be well formulated for an arbitrary type of the emission distribution. In the extreme case, different types of data can be assigned to each link (eg. count data for link 1, normally distributed data for link 2, ...). In particular, we emphasize that the collapsed Gibbs sampler described in the following section can also be established if a distribution in the exponential family and its conjugate are used.

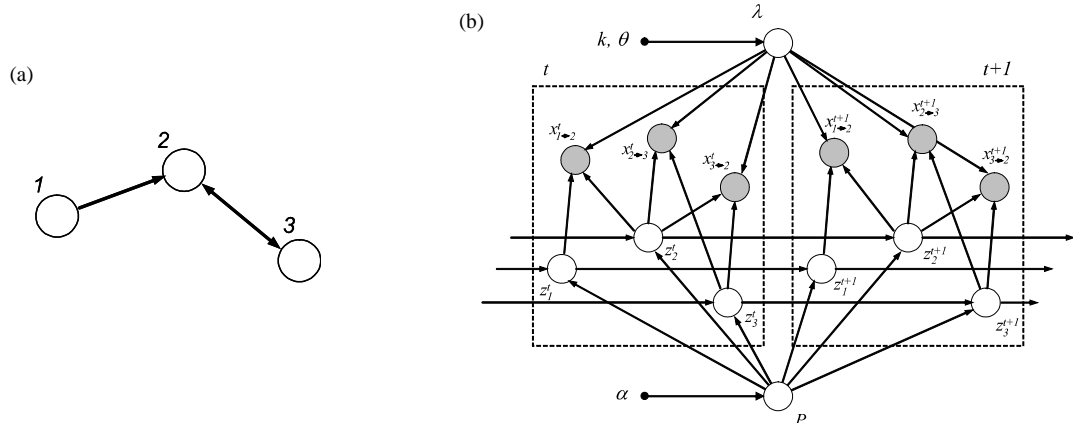


Fig. 2. (a) Example of small directed network with three links. (b) Graphical representation of HMFN for the left small network. Dashed boxes show time slice and small black points show hyperparameters. White and gray circles represent unobserved and observed random variables respectively.

A. Joint Distribution of HMFN

According to the generative process given above, we describe the joint distribution of all random variables in the HMFN model. Firstly, probabilistic density function (pdf) of transition matrix $\mathbf{P} = \{P_i\}$ is the product of Dirichlet distributions:

$$p(\mathbf{P}|\boldsymbol{\alpha}) = \prod_{i \in V} \prod_{z \in Z_i} p(P_{iz}|\boldsymbol{\alpha}_i). \quad (1)$$

Since emission parameters $\boldsymbol{\lambda} = \{\lambda_e^{z_1 \rightarrow z_2}\}$ are also drawn independently, the pdf is the product of the gamma distribution:

$$p(\boldsymbol{\lambda}|\theta, k) = \prod_{e \in E} \prod_{z_1 \in Z_{b(e)}} \prod_{z_2 \in Z_{t(e)}} p(\lambda_e^{z_1 \rightarrow z_2}|\theta, k). \quad (2)$$

The Markov chain of hidden state $z_i^{1:T}$ in node $i \in V$ is described by the following:

$$p(z_i^{1:T}|P_i) = \frac{1}{|Z_i|} \prod_{t=2}^T p(z_i^t|z_i^{t-1}). \quad (3)$$

Here $1/|Z_i|$ corresponds to the initial distribution. Since $z_i^{1:T}$ are drawn independently for each node i ,

$$p(\mathbf{z}^{1:T}|\mathbf{P}) = \prod_{i \in V} p(z_i^{1:T}|P_i). \quad (4)$$

Finally, since the pdf of link flow (referred to as the emission density) is independent when all hidden states $z^{1:T}$ and emission parameters $\boldsymbol{\lambda}$ are given, it is written as follows:

$$p(\mathbf{x}^{1:T}|\boldsymbol{\lambda}, \mathbf{z}^{1:T}) = \prod_{t=1}^T \prod_{e \in E} p(x_e^t|\lambda_e^{z_{b(e)}^t \rightarrow z_{t(e)}^t}). \quad (5)$$

This indicates that the flow x_e^t on link e at time t is drawn from a mixture distribution with component size $|Z_{b(e)}| \cdot |Z_{t(e)}|$ depending only on its beginning and terminating nodes. As a consequence, we obtain the joint density of HMFN by multiplying the densities above:

$$p(\mathbf{x}^{1:T}, \mathbf{z}^{1:T}, \boldsymbol{\lambda}, \mathbf{P}|\theta, k, \boldsymbol{\alpha}) = p(\mathbf{x}^{1:T}|\boldsymbol{\lambda}, \mathbf{z}^{1:T})p(\boldsymbol{\lambda}|\theta, k)p(\mathbf{z}^{1:T}|\mathbf{P})p(\mathbf{P}|\boldsymbol{\alpha}). \quad (6)$$

The HMFN is one of a *Bayesian hierarchical model*. As shown in next section, because the conditional dependencies have been carefully designed, the model can be solved by Gibbs sampler. Here we note that the number of possible hidden states at one time slice is $\prod_{i \in V} |Z_i|$ which increases exponentially with the number of nodes in the network.

B. Interpretation of the Model

The HMFN model can be understood as an unsupervised learning method for dynamic relational data. The model has hidden states that obeys the Markov chain for each node. The hidden state $z_i^t \in Z_i$ is a latent class of the node. A link $e = i \rightarrow j$ has states (z_i^t, z_j^t) at the beginning and terminating nodes for each time. This tuple (z_i^t, z_j^t) can be regarded as a latent class (or feature) of the link e . Furthermore, this method can be regarded as a kind of the multi-clustering method for the flow on network.

C. Related Works

By inference using HMFN, the dynamics of network flow

is compressed into a latent vector of dimension equal to the number of nodes for each time slice. Ide *et al.* proposed an eigenvector-based dimensional reduction method in the anomaly detection context [5]. They assumed a symmetric matrix form and used its principal eigenvector as the feature vector. Time dependency was not considered explicitly. In our case, we emphasize that HMFN does not assume symmetry of the flow. The relationship between their approach and the HMFN model resembles to that of latent semantic indexing (LSI) [6] and latent Dirichlet allocation (LDA) [7]. While LSI is based on singular value decomposition, LDA is a generative approach.

Some generative models have been defined on networks. The stochastic block model (SBM) [8] and its extensions [9], [10] are also generative models for the relational data. Although HMFN is also a model for the relational data, the purpose is completely different. While the aim of SBM is clustering interacting objects (nodes) into groups with similar flow pattern, our purpose is not. HMFN does not share the hidden states, that is, the meaning of Z_i and Z_j ($i \neq j$) is different. It can be intuitively explained by following example: $Z_i = \{\text{morning, afternoon, night, night with a football game}\}$, $Z_j = \{\text{workday, weekend}\}$. For SBM, on the other hand, Z_i 's are common for all $i \in V$ and it is typically represented as $Z_i = \{\text{social group 1, social group 2, social group 3}\}$ ($\forall i \in V$).

Hidden random Markov field (HRMF) [11], which is used in image processing, can be regarded as a latent variable model on a network. As with the HMFN, HRMF has a latent variable on nodes. The difference appears in the location of observations and the type of interaction. In HRMF, observations occur at nodes and latent variables interact directly. On the other hand, in HMFN, flows are observed on links and latent variables interact indirectly through the flow observation.

As a probabilistic model, the HMFN model can be regarded as an extension of the hidden Markov model (HMM). HMFN is modeled as HMMs are connected by a network. In fact, when the network has only one node and a self link, HMFN is equivalent to standard HMM. In this point of view, HMFN extends the topology of the network. Therefore, we believe that the model is general and as powerful as the hidden Markov model (HMM) and its extensions. Additionally, we point that, if the network has only two nodes and one link, the HMFN is equivalent to the factorial hidden Markov model (FHMM) [12] with two streams.

III. INFERENCE AND PARAMETER ESTIMATION

A. Collapsed Gibbs Sampler for HMFN

In this section, we describe an algorithm that enables the inference for HMFN. The purpose is to obtain a posterior distribution of hidden variables. We use gamma and Dirichlet prior distributions, which are conjugate priors of the Poisson and categorical distributions, respectively. This fact makes the joint distribution (6) partially integrable and leads to a *collapsed Gibbs sampler* [13] for HMFN. In general, the autocorrelation of samples by the collapsed Gibbs sampler gets smaller than that of standard Gibbs sampler.

For the derivation of the collapsed Gibbs sampler, the conditional probability $p(z_i^t | x^{1:T}, z_{-i}^t)$ should be considered. Here we use the notation z_{-i}^t as the set of all hidden variables except for z_i^t . For simplicity, we assume that P_i obeys the *symmetric Dirichlet distribution*, i.e. all elements in α_i take the same scalar value $\alpha_i > 0$. Taking into account the marginalization of transition matrices \mathbf{P} , we can calculate as follows:

$$\begin{aligned} & p(z_i^t, \lambda | x^{1:T}, z_{-i}^t) \\ & \propto \frac{(n_{i, -t}^{z_i^t \rightarrow z_i^{t+1}} + \alpha_i)}{N_{i, -t}^{z_i^t} + \alpha_i |Z_i|} \cdot (n_{i, -t}^{z_i^{t-1} \rightarrow z_i^t} + \alpha_i) \\ & \times \prod_{e \in E_{\text{out}}(i)} p(x_e^t | \lambda_e^{z_i^t \rightarrow z_{t(e)}^t}) \prod_{e \in E_{\text{in}}(i)} p(x_e^t | \lambda_e^{z_{t(e)}^t \rightarrow z_i^t}) \\ & \times p(\lambda | x^{1:T}, z_{-i}^t, \theta, k) \end{aligned} \quad (7)$$

where $n_{i, -t}^{z_1 \rightarrow z_2}$ is the count of transitions $z_1 \rightarrow z_2$ in the Markov chain at node $i \in V$ except for the transition at time t , $N_{i, -t}^{z_i^t}$ is the normalizer defined by

$$N_{i, -t}^{z_i^t} = \sum_{w \in Z_i} n_{i, -t}^{z_i^t \rightarrow w}, \quad (8)$$

and $p(\lambda | x^{1:T}, z_{-i}^t, \theta, k)$ is the posterior distribution of emission parameters. We can integrate out $\lambda_e^{z_i^t \rightarrow z_j^t}$ by posterior distribution $H_{e, -t}^{z_i^t \rightarrow z_j^t}(\lambda)$ which is defined by a posterior distribution for given data $D = \{x_e^{t'} | t' \neq t, z_i^{t'} = z_i^t, z_j^{t'} = z_j^t\}$. If D is the empty set, then $H_{e, -t}^{z_i^t \rightarrow z_j^t}(\lambda)$ is the prior distribution. Integrating the emission density by the posterior leads the *posterior predictive*:

$$p(x_e^t | x_{e, -t}^t, z^{1:T}) = \int p(x_e^t | \lambda) dH_{e, -t}^{z_i^t \rightarrow z_{t(e)}^t}(\lambda). \quad (9)$$

We emphasize that $H_{e, -t}^{z_i^t \rightarrow z_j^t}(\cdot)$ can be calculated analytically in the conjugate prior case, and the posterior predictive distribution (9) is also analytically integrable. In the Poisson distributed flow and gamma prior case, $H_{e, -t}^{z_i^t \rightarrow z_j^t}(\cdot)$ is also the gamma distribution and the posterior predictive distribution is a negative binomial distribution. We also note that $H_{e, -t}^{z_i^t \rightarrow z_j^t}(\lambda)$ only depends on the sufficient statistics of D . Maintaining sufficient statistics for each mixture component makes the calculation of (9) fast.

Finally, we obtain the “collapsed posterior distribution of z_i^t ” as follows:

$$\begin{aligned} p(z_i^t | x^{1:T}, z_{-i}^t) & \propto \frac{(n_{i, -t}^{z_i^t \rightarrow z_i^{t+1}} + \alpha_i)}{N_{i, -t}^{z_i^t} + \alpha_i |Z_i|} \cdot (n_{i, -t}^{z_i^{t-1} \rightarrow z_i^t} + \alpha_i) \\ & \times \prod_{e \in E_{\text{out}}(i)} \int p(x_e^t | \lambda) dH_{e, -t}^{z_i^t \rightarrow z_{t(e)}^t}(\lambda) \\ & \times \prod_{e \in E_{\text{in}}(i)} \int p(x_e^t | \lambda) dH_{e, -t}^{z_{t(e)}^t \rightarrow z_i^t}(\lambda). \end{aligned}$$

We can sequentially sample the posterior latent variables $z^{1:T}$ from (10). Algorithm 1 represents the collapsed Gibbs sampling procedure for HMFN.

Algorithm 1 Collapsed Gibbs sampler for HMFN

Input: Directed graph $G(V; E)$; Data $x^{1:T}$ on E ;

Hyper-parameters θ , k and α

Output: MCMC sequence of $z^{1:T}$

Initialize $z^{1:T}$ randomly.

repeat

for $t = 1$ **to** T **do**

for all $i \in V$ **do**

 Calculate sufficient statistics of posterior of \mathbf{P} ,
 and λ for given $x^{1:T}$ and $z^{1:T}$.

 Update z_i^t by sampling from (10).

end for

end for

yield $z^{1:T}$

until forever

B. Hyperparameter Optimization

In the Gibbs sampler described in the previous section, the hyperparameters are fixed. In this section, we describe the hyperparameter optimization. We repeatedly maximize $p(k, \theta, \alpha | x^{1:T}, z^{1:T})$ after the scans of Gibbs sampling. Here we assume that hyperparameters k , θ and α obeys given prior distributions.

The posterior distribution of α_i is given by $p(\alpha_i | z_i^{1:T}) \propto p(z_i^{1:T} | \alpha_i) p(\alpha_i)$ where $z_i^{1:T}$ is a snapshot of the hidden states of the Gibbs sampler. $p(z_i^{1:T} | \alpha_i)$ is the product of Dirichlet multinomial distribution:

$$\begin{aligned} p(z_i^{1:T} | \alpha_i) & = \int p(z_i^{1:T} | P_i) p(P_i | \alpha_i) dP_i = \\ & \frac{1}{|Z_i|} \prod_{z \in Z_i} \frac{\Gamma(\alpha_i | Z_i)}{\Gamma(N_i^z + \alpha_i | Z_i)} \left\{ \prod_{w \in Z_i} \frac{\Gamma(n_i^{z \rightarrow w} + \alpha_i)}{\Gamma(\alpha_i)} \right\}, \end{aligned} \quad (10)$$

where $n_i^{z \rightarrow w}$ is the count of the state transition from z to w in $z_i^{1:T}$, and $N_i^z = \sum_{w \in Z_i} n_i^{z \rightarrow w}$. Therefore the maximization of $\log p(\alpha_i | z_i^{1:T})$ with respect to α_i is achieved by the Newton–Raphson method or a quasi-Newton method such as the BFGS method. The gradient involves the digamma function, therefore it is easily calculated. The posterior of k and θ , which are the hyperparameters of the emission distribution, are given by the following:

$$p(k, \theta | x^{1:T}, z^{1:T}) \propto p(x^{1:T} | z^{1:T}, k, \theta) p(k) p(\theta)$$

Because $p(x^{1:T} | z^{1:T}, k, \theta)$ can be analytically tractable, k and θ can be also optimized numerically.

C. Model selection

In HMFN, sets of hidden states Z_i are defined for each node $i \in V$. For model selection, we have to determine the size of Z_i . The combination of the sizes of the hidden sets is quite large, model selection by an information criterion is not realistic. In order to determine the size of Z_i , it would be useful to adopt a nonparametric Bayesian approach. The model selection for HMFN is realized by using potentially infinite size of Z_i for each node and determining their sizes by the data. The infinite HMM (iHMM) [14] is a model that is a nonparametric Bayesian approach for HMM. Hidden

dynamics in HMFN can be extended by a method similar to the iHMM. In this study, we employed state sets with common fixed size K , i.e. $|Z_i| = K$ ($\forall i \in V$). Practically this would be no problem because the hidden states degenerate and sparse results are obtained at some nodes. The value of K can be chose by evaluating likelihood of test data.

IV. NUMERICAL RESULTS

We show two numerical examples. We first apply HMFN to synthetic data in order to evaluate the performance of the HMFN and the collapsed Gibbs sampler. Then a second example shows an application to real data, specifically to taxi demand data in Bangkok covering four months. Although hyperparameters should be estimated in principle, we use predetermined values in this study.

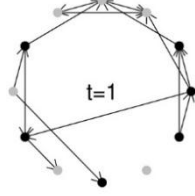


Fig. 3. Network of the test problem ($|V|=12$). Gray and black points indicate the value of the hidden variable (0 and 1, respectively). The arrows indicate that the observed flow x_e is greater than 0.

A. Synthetic Data Examples

1) *Dataset*: Here we examine the performance of HMFN and the collapsed Gibbs sampler for synthetic data. We used a fully connected network with 12 nodes. Nodes are placed on a circle (Fig. 3) and the distance between two nodes is defined by distance on the circle, with the distance between two neighboring nodes normalized to 1. Hidden variables z_i^t take binary values for each node, i.e. $|Z_i| = 2$. Their transition matrix was uniformly defined as follows:

$$P_i = \begin{bmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{bmatrix} \quad \forall i \in V. \quad (11)$$

The Poisson distribution was used as the emission distribution. Let $\lambda_e^{z_{b(e)} \rightarrow z_{t(e)}}$ be the mean parameter of the Poisson distribution when the states at the beginning and terminating nodes of $e \in E$ are $z_{b(e)}$ and $z_{t(e)}$ respectively. We prepare two synthetic data that have different emission densities as follows:

(Problem 1)

$$\begin{bmatrix} \lambda_e^{0 \rightarrow 0} & \lambda_e^{0 \rightarrow 1} \\ \lambda_e^{1 \rightarrow 0} & \lambda_e^{1 \rightarrow 1} \end{bmatrix} = \begin{bmatrix} 0.5 & 1 \\ 1 & 2 \end{bmatrix} \cdot \exp\left(-\frac{d}{2}\right), \quad (12)$$

(Problem 2)

$$\begin{bmatrix} \lambda_e^{0 \rightarrow 0} & \lambda_e^{0 \rightarrow 1} \\ \lambda_e^{1 \rightarrow 0} & \lambda_e^{1 \rightarrow 1} \end{bmatrix} = \begin{bmatrix} 1 & 0.5 \\ 1 & 2 \end{bmatrix} \cdot \exp\left(-\frac{d}{2}\right), \quad (13)$$

where d is the circular distance between $b(e)$ and $t(e)$. The exponential terms in (12) and (13) are *traffic resistance*, which represents that the observed amount of flow on a link decreases as the distance increases. Fig. 3 shows one time slice of the data. We synthesized the network flow dynamics for $T = 1000$, and applied the collapsed Gibbs sampler. For the hyperparameters, $\theta = 1, k = 1$ for the gamma prior, and $\alpha_i = 2$ ($\forall i \in V$) for the symmetric Dirichlet prior were used.

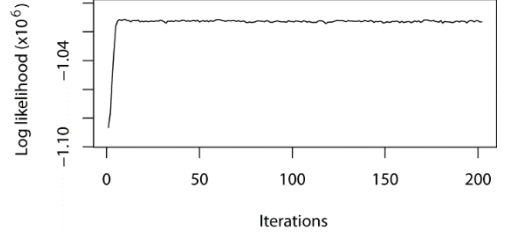


Fig. 4. Likelihood convergence of HMFN for problem 1.

Fig. 4 shows the likelihood convergence for a sample from problem 1. This figure indicates that 200 iterations are enough for convergence.

2) *Results*: To evaluate the performance, we use the reconstruction accuracy and the adjusted Rand index (ARI) [15] of hidden variables as the measurements. For calculation of these measurements, the hidden variables at the final Gibbs step were used.

For comparison, we also applied k -means and HMM to the same data. Since these methods are not defined for network flow, we estimate independently for each node $i \in V$ with *node-local* feature $\{x_e^t | e \in E_{\text{out}}(i) \cup E_{\text{in}}(i)\}$. For the inference of HMM, collapsed Gibbs sampling and the same hyperparameters were used. For HMM and HMFN, we considered respectively the following two models: (1) $|Z_i| = 2$ (true model), (2) $|Z_i| = 3$ (larger model). We synthesized ten network flow dynamics and calculated the average and standard deviation of the performance measures for each test problem.

Table I shows the reconstruction accuracy and ARI of the hidden variables for k -means, HMM, and HMFN. k -means has a relatively low accuracy for each problem. HMM with true hidden variable sizes (HMM(2)) has almost the same performance as HMFN for problem 1. However, the inference does not work well for problem 2.

TABLE I: COMPARISON OF THE RECONSTRUCTION ACCURACY AND THE ADJUSTED RAND INDEX

		k-means(2)	HMM(2)	HMM(3)	HMFN(2)	HMFN(3)
Reconstruction Accuracy (%)	Problem 1	77.34 \pm 0.76	91.18 \pm 0.58	86.40 \pm 2.24	91.86 \pm 0.40	91.40 \pm 0.86
	Problem 2	66.52 \pm 0.91	80.01 \pm 0.99	NA*	90.46 \pm 0.56	90.05 \pm 0.94
Adjusted Rand Index	Problem 1	0.294 \pm 0.014	0.679 \pm 0.012	0.413 \pm 0.018	0.699 \pm 0.016	0.694 \pm 0.020
	Problem 2	0.110 \pm 0.008	0.375 \pm 0.028	0.328 \pm 0.019	0.655 \pm 0.019	0.649 \pm 0.019

*NA means that no meaningful reconstruction is obtained

The reason is explained as follows. In problem 2, if the hidden state of the terminating node is 0, then the emission

density of the link does not depend on the state of the beginning node. This makes problem 2 more noisy than

problem 1. In such a case, information about the other nodes becomes more important for obtaining a more robust result. Because the HMFN takes into account the information of all nonconnecting links and other nodes implicitly, robust results can be obtained when interaction between nodes exists, as this example. On the other hand, the node-local approaches (k -means and HMM) can be regarded as a “mean field” approach which ignores the interaction.

Furthermore, we would like to emphasize that in HMFN(3), whose size of hidden variables for each node is larger than that of the true model, the reconstruction accuracy is not worse while HMM(3) is worse than HMM(2). In HMFN(3), we obtained a sparse result, which indicates that one of the three hidden states has collapsed after the iterations. Note that because we are using a collapsed Gibbs sampler, the high reconstruction accuracy in HMFN inference indicates that the mixture components are also estimated highly accurately.

TABLE II: TEST DATA LOG LIKELIHOOD FOR SYNTHETIC DATA

	Poisson	HMFN(2)	HMFN(3)
Problem 1	-103362	-100350	-100632
Problem 2	-103584	-102378	-102703

Table II shows the log likelihood of the test data. “Poisson” means a model with no hidden variable (equivalent to HMFN(1)) as follows:

$$\begin{aligned} x_e^t &\sim \text{Poisson}(\lambda_e) & \forall e \in E, 1 \leq t \leq T \\ \lambda_e &\sim \text{Gamma}(k, \theta) & \forall e \in E \end{aligned}$$

The result indicates that HMFN(2) is the best model among three models for both problems. Because the product of probabilities of HMM over nodes does not constitute a proper probabilistic model (the links are doubly counted), we did not compare the test likelihood with HMM(m).

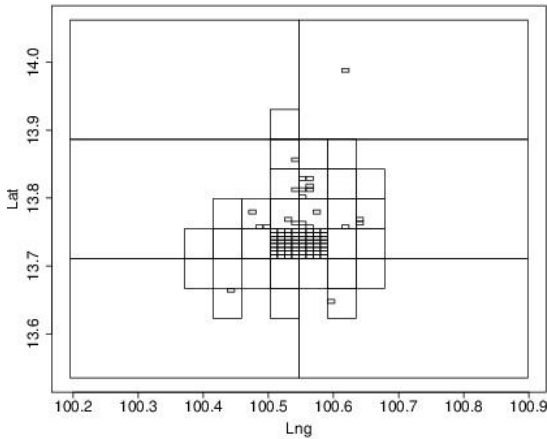


Fig. 5. 120 regions (nodes) used in this study.

B. Application to Real Human Mobility: Taxi Data in Bangkok

1) *Dataset*: GPS data from 9 000 taxis were collected in Bangkok for a long time period. It was converted into the origin-destination demand dataset. Here origin means the zone where the customer gets on the taxi and destination does where the customer gets off. Coordinates should be

discretized into zones. For discretization, we used GeoHash (<http://geohash.org/>) that converts longitude/latitude into a hash string which corresponds to a rectangle zone. In this study we used the 120 zones as shown in Fig. 5. The resolutions of the zones were determined by hand. They are high resolution if the location is near by the center of the city or the taxi demand level is high. The size of smallest zone is about $1\text{km} \times 600\text{m}$. These regions are treated as nodes in HMFN. For flow outside of the network, we define a special boundary node (not shown in Fig. 5).

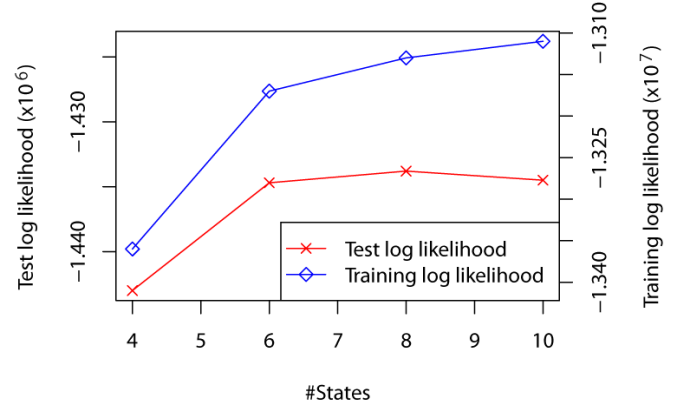


Fig. 6. The test likelihood and the training likelihood of real human mobility data.

We counted up the customer’s trips and obtained an hourly dynamic OD matrix. The (i, j) element of the matrix at time t represents the number of trips in which a customer gets on the taxi at zone i and time t and gets off at zone j . We used 128 days (3 072 hours; $T = 3072$) data from 3/Sep/2012. The zero values occupy more than 90% of the observed OD matrix.

2) *HMFN setting and learning*: The Poisson distribution was used for the emission distribution. In order to determine the size of hidden states, we evaluated the test log likelihood as shown in Fig. 6 (last 14 days were used for test data). We assumed that $|Z_i|$ takes common value K for all $i \in V$. The test likelihood saturates at $K = 6$ while the training likelihood increases monotonically. Therefore, we set the number of hidden states to six. 24 hidden states were used for the boundary node. Here we note that 100 iterations were enough to converge.

3) *Visualizing extracted flow pattern*: By the inference, we obtained time series of hidden states $\mathbf{z}^{1:T}$. In order to check if the result is proper, we would like to visualize the typical flow pattern for each node and each state. For this purpose, we define k -th *eigenflow* $\mathbf{v}_i(k)$ of node $i \in V$, which is defined as

$$\mathbf{v}_i(k) = \{E[X_e^t | Z_i^t = k] \mid e \in E_{\text{out}}(i) \cup E_{\text{in}}(i)\} \quad (14)$$

By definition, the k -th eigenflow represents the expected value of the flow for each link around the node i when $z_i^t = k$. Intuitively, eigenflow $\mathbf{v}_i(k)$ can be interpreted as the center of cluster of k -th state. Solving for a closed form of (14) is intractable, but we can approximate this expectation using Gibbs sampling.

4) *Extracted patterns*: Several interesting patterns were found in the result. Fig. 7 shows the eigenflow of a zone where many nightclubs exist. The columns represent different hidden states ($z_i = 1, \dots, 6$) and two rows show

out-flows and in-flows, respectively. Vertical and horizontal positions of each plot represent latitude and longitude, respectively. The width of the line indicates the size of the eigenflow of the corresponding link. The second column (state 2) and sixth column (state 6) are very characteristic. Actually, the state 2, which has large in-flow values, tends to

appear around 22:00. The state 6 appears after 2:00 (midnight) and is observed only when the state 2 is observed immediately before. We can guess that many people come to this zone around midnights, and leave there before morning. This pattern is observed on mainly Friday and Saturday nights, but sometimes appears on the other days suddenly.

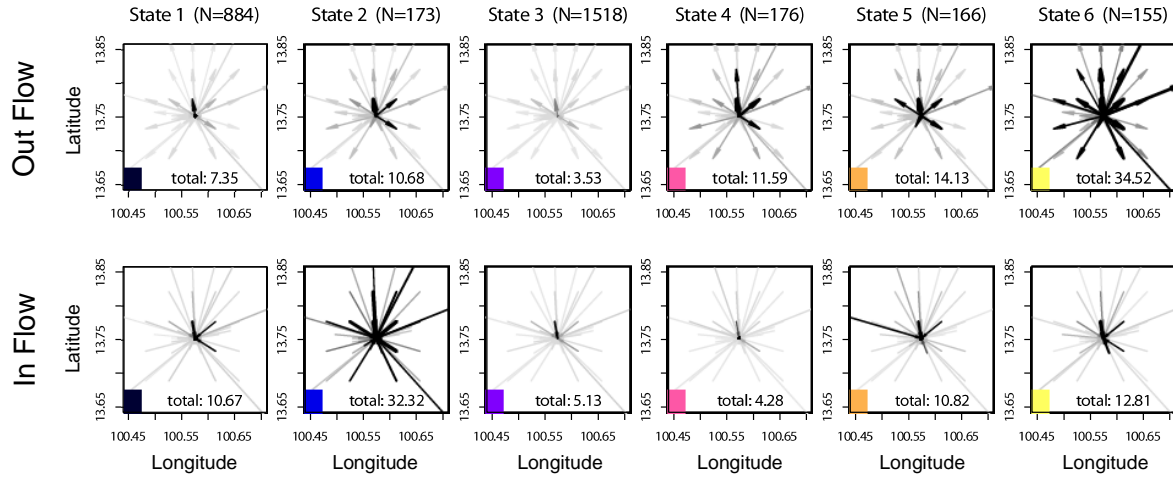


Fig. 7. Eigenflows of “nightclub” zone in Bangkok (geohash: w4rw0x). N in the parenthesis shows the number of times that the corresponding state is assigned. The number in the bottom of the plot means the total amount of eigenflow.

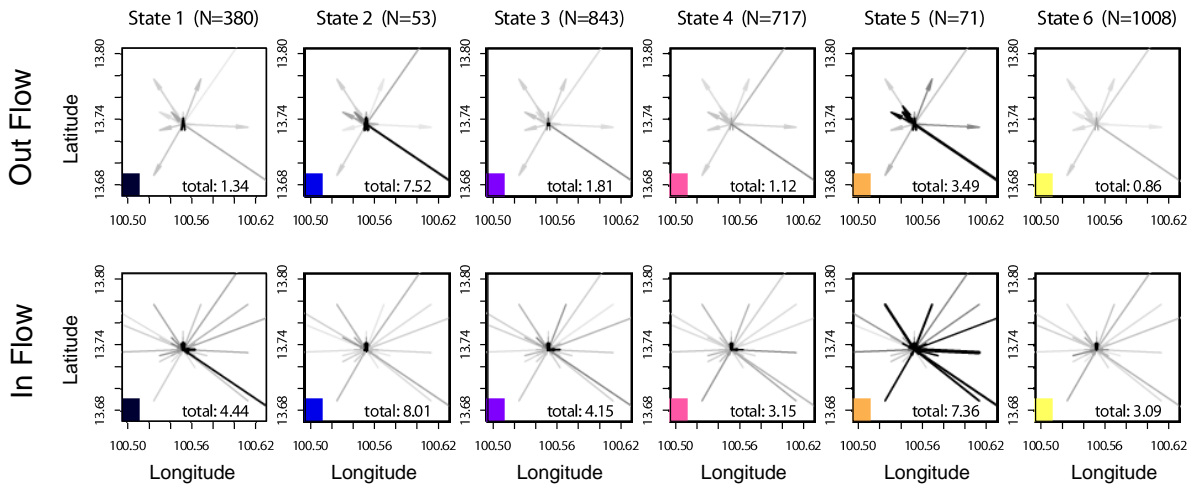


Fig. 8. Eigenflows of “church” zone in Bangkok (geohash: w4rw0h).

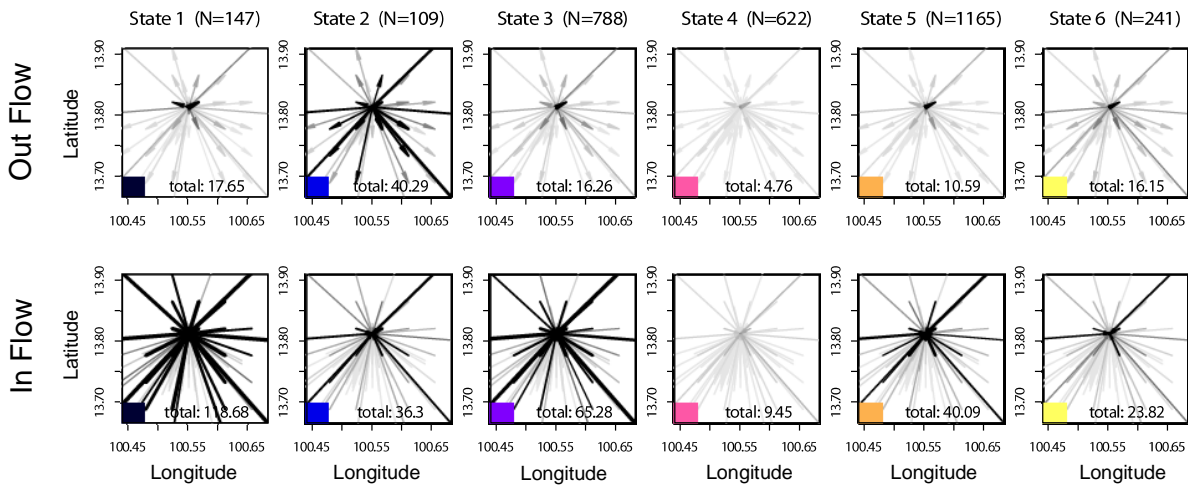


Fig. 9. Eigenflows of “north bus terminal” zone in Bangkok (geohash: w4rw84).

The second example is shown in Fig. 8. The fifth state is quite different from other states. While the total quantity of

in-flow of the state 5 is as many as state 2 (7.36 and 8.01 respectively), the flow of the state 5 is more distributed. This

indicates that taxis travel longer distance in the state 5. The second point is that the state 5 is assigned only on every Sunday morning and Christmas. We note that there are churches in this place. We can guess that the state 5 would be caused by people coming to church for the mass.

The last example shows “seasonal change” pattern. Fig. 9 show the eigenflow at north bus terminal for long distance bus toward north of Thailand. Fig. 9 shows that the daily state share in this place. A peak of the state 1 is observed around day 116 (darkest color). This term corresponds to the new year vacation and observed peak would be homecoming visit. The state 2, which increase after the new year, corresponds to a pattern in which people leave the bus terminal for their home. This kind of special patterns during the new year vacation are observed in many zones. This fact indicates that the traffic pattern is quite different during new year vacation.

In other places, a lot of interesting patterns are extracted. They are classified into following:

- event patterns (eg. at stadiums, event halls, night clubs, festivals,...)
- periodic patterns (eg. commuting, a weekend market, the mass at church,...)
- seasonal patterns (eg. new year vacation)

We have shown that several kinds of patterns are extracted without other features. These results indicate that the proposed model explains the real human mobility pattern well and that applications such as event detection and prediction are possible.

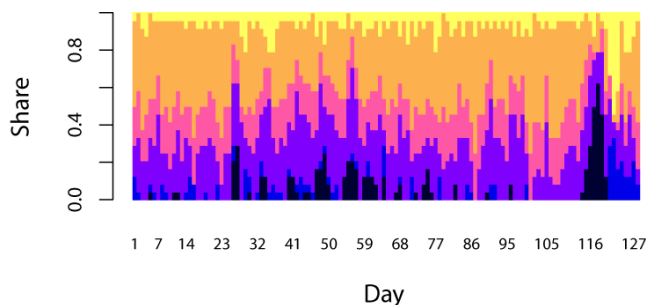


Fig. 10. Daily state share at “north bus terminal”. Colors corresponds to the states shown in Fig. 9.

V. DISCUSSION

One of the direct applications of the HMFN model is to detect the novelty/anomaly of flow pattern in the network. As shown in Fig. 10, the distribution of latent state changes when a novel event occurs. This kind of novelties can be detected by statistical method that monitors the probability of observed patterns. HMFN model can capture the interactions of nodes so that it is expected to work for complex flow patterns. The development of such application is future work.

The proposed collapsed Gibbs sampler converges fast in our examples, but other approach for the inference would be considered. While variational Bayesian inference is one of them, we would like to point the possibility of distributed learning when the network structure is sparse. The probabilistic dependency of the HMFN is local, therefore we can adopt the learning algorithm for each divided network (boundary should be adjusted appropriately). Model selection for HMFN would be difficult task, we have pointed out that

the nonparametric Bayesian extension by a method similar to infinite HMM [14] can defeat this problem.

VI. CONCLUSION

In this paper we have proposed the HMFN model, a generative stochastic model that describes the dynamics of flow on a network. HMFN is based on simple assumptions for hidden states of the nodes and observed flow on the links. We have also proposed a collapsed Gibbs sampler that enables inference of the model. The numerical result shows that the proposed inference algorithm converges quickly.

The synthetic data example have revealed that node-local classification approaches (k -means and HMM) fail if strong interaction between nodes exists. On the other hand, the proposed method enables robust learning. We have applied the model for the taxi demand data in Bangkok that reflects real human mobility. The results for the real data indicate that our model explains the real human mobility patterns well and that its application to, for example, anomaly/novelty detection or flow pattern prediction is possible.

As pointed out in [7], the advantage of generative probabilistic models is their modularity and extensibility. In HMFN case, many kinds of extensions including nonparametric Bayesian extension, Markov chain of order m , and flow of natural language (communication in SNS), can be considered; here we have shown only the simplest formulation.

REFERENCES

- [1] M. Gonzalez, C. Hidalgo, and A. Barabasi, “Understanding individual human mobility patterns,” *Nature*, vol. 453, pp. 779-782, 2008.
- [2] K. Ashok and Ben-Akiva, “Dynamic origin-destination matrix estimation and prediction for realtime traffic management systems,” *Transportation and Traffic Theory*, C. F. Daganzo, Ed. Elsevier, 1993.
- [3] J. Barcelo, L. Montero, L. Marques, and C. Carmona, “Travel time forecasting and dynamic origin-destination estimation for freeways based on bluetooth traffic monitoring,” *Journal of the Transportation Research Board*, vol. 2175, pp.19-27, 2010.
- [4] A. G. Wilson, “A statistical theory of spatial distribution models”, *Transportation Research*, vol. 1, pp. 253-269, 1967.
- [5] T. Ide and H. Kashima, “Eigenspace-based anomaly detection in computer systems,” in *Proc. 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD04)*, ACM Press, 2004.
- [6] S. Deerwester, S. Dumais, T. Landauer, G. Furnas, and R. Harshman, “Indexing by latent semantic analysis,” *Journal of the American Society of Information Science*, vol. 41, pp. 391-407, 1990.
- [7] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation,” *Journal of Machine Learning Research*, vol. 3, pp. 993-1022, 2003.
- [8] K. Nowicki and T. Snijders, “Estimation and prediction for stochastic blockstructures,” *Journal of the American Statistical Association*, vol.96, pp. 1077-1087, 2001.
- [9] C. Kemp, J. Tenenbaum, T. L. Griffiths, T. Yamada, and N. Ueda, “Learning systems of concepts with an infinite relational model,” in *Proc. the 21st National Conference on Artificial Intelligence*, 2006.
- [10] K. Ishiguro, T. Iwata, N. Ueda, and J. Tenenbaum, “Dynamic infinite relational model for time-varying relational data analysis,” in *Proc. Advances in Neural Information Processing Systems*, 2010.
- [11] H. Künsch, S. Geman, and A. Kehagias, “Hidden Markov random fields,” *The annals of Applied Probability*, vol. 5, pp. 572-602, 1995.
- [12] Z. Ghahramani and M. I. Jordan, “Factorial hidden Markov models,” *Machine Learning*, vol. 29, pp. 245-275, 1997.
- [13] J. S. Liu, “The collapsed Gibbs sampler in bayesian computations with applications to a gene regulation problem,” *Journal of the American Statistical Association*, vol. 89, pp. 958-966, 1994.
- [14] M. J. Beal, Z. Ghahramani, and C. E. Rasmussen, “The infinite hidden Markov model,” in *Proc. Advances in Neural Information Processing Systems*, 2002.

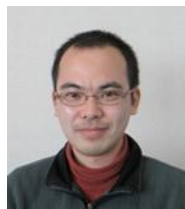
- [15] L. Hubert and P. Arabie, "Comparing partitions," *Journal of Classification*, vol. 2, pp. 193-218, 1985.



Satoshi Koide received his B.S. and M.S. degrees from the University of Osaka, Japan, in 2004 and 2006 respectively. In 2006, he joined Toyota Central R&D Laboratories, Aichi. His research interests include statistical machine learning, optimization and their application to ITS.



Hiroshi Ohno received the B.Eng. and M.Eng. degrees from Gifu University, Japan, in 1984 and 1986, respectively, and the Ph.D. degree from Nagoya University, Japan, in 1999. His current interests include human behavior modeling, statistical machine learning, and materials informatics.



Ryuta Terashima received the B.S. degree from Aichi University of Education, Japan, in 1992, and received Dr. of Engineering from Nagoya Institute of Technology, Japan, in 2010. In 1992, he joined Toyota Central R&D Laboratories, Aichi. His research interests include speech recognition, spoken dialog systems, and modeling of human behavior for advanced safety systems. He is a member of ACM and IPSJ.



Thanomsak Ajjanapanya received his B.Eng, M.Eng from the Thammasat University in 2004, 2006 respectively. He joined Toyota Tsusho Electronics in 2006 and in charge of software development division.



Itti Rittaporn received his B.Eng, M.Eng, and Ph.D from the University of Tokyo in 1983, 1985 and 1988 respectively. He joined Toyota Tsusho Electronics in 2006 and in charge of the development ITS related technology business.