TransInfo
Tier I University Transportation Center

FINAL REPORT

Variational Inference for Agent-Based Models with Applications to Achieve Fuel Economy

Date of report: October 2017

Wen, Dong, PhD, Assistant Professor, University at Buffalo
Chunming, Qiao, PhD, Professor, University at Buffalo

Prepared by: Wen Dong
Department of Computer Science and Engineering, University at Buffalo
Davis Hall #306
Buffalo, NY 14260

Prepared for:
Transportation Informatics Tier I University Transportation Center
204 Ketter Hall
University at Buffalo
Buffalo, NY 14260

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.

4. Title and Subtitle	**5. Report Date**
Variational Inference for Agent-Based Models with Applications to Achieve Fuel Economy	10/29/2017
	6. Performing Organization Code
7. Author(s) Wen Dong and Chunming Qiao	**8. Performing Organization Report No.**
9. Performing Organization Name and Address Department of Computer Science and Engineering, University at Buffalo	**10. Work Unit No. (TRAIS**
	11. Contract or Grant No. DTRT13-G-UTC48
12. Sponsoring Agency Name and Address US Department of Transportation Office of the UTC Program, RDT-30 1200 New Jersey Ave., SE Washington, DC 20590	**13. Type of Report and Period Covered** Final June 2016 – June 2017
	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

In this project, we developed optimization and sampling based inference algorithms to track and predict real-time traffic dynamics in a city-scale transportation network from an agent-based transportation model and isolated observations of the trajectories of several hundred probe vehicles. We demonstrated the value of combining simulation modeling and big data in delivering travel information to drivers and promoting efficient driving through real world road networks and tracking data from mobile phones. This project integrates machine learning, big data, sensor networks, and agent-based transportation modeling to prototype an algorithm that combines the power of a model-driven approach with the power of big data, and promotes responsible driving by showing how different agent trips are associated with different travel time and fuel consumption.

17. Key Words	**18. Distribution Statement**
Machine learning, signal processing, transportation network dynamics	No restrictions. This document is available from the National Technical Information Service, Springfield, VA 22161

19. Security Classif. (of this report)	**20. Security Classif. (of this page)**	**21. No. of Pages**	**22. Price**
Unclassified	Unclassified	12	

Variational Inference for Agent-Based Models with Applications to Achieve Fuel Economy

Wen Dong, Chunming Qiao
Department of Computer Science and Engineering
University at Buffalo
Buffalo, NY 14260

Abstract

In this project, we developed optimization and sampling based inference algorithms to track and predict real-time traffic dynamics in a city-scale transportation network from an agent-based transportation model and isolated observations of the trajectories of several hundred probe vehicles. We demonstrated the value of combining simulation modeling and big data in delivering travel information to drivers and promoting efficient driving through real world road networks and tracking data from mobile phones.

This project integrates machine learning, big data, sensor networks, and agent-based transportation modeling to prototype an algorithm that combines the power of a model-driven approach with the power of big data, and promotes responsible driving by showing how different agent trips are associated with different travel time and fuel consumption.

Acknowledgements

We would like to thank our PhD students Fan Yang, Le Fang and Tong Guan for implementing the algorithms, and benchmarking them with synthetic and real world data.

Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the U.S. Department of Transportation's University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.

Table of Contents

PROBLEM 1

APPROACH AND METHODOLOGY 4

FINDINGS: DOCUMENTATION OF DATA GATHERED,
ANALYSES PERFORMED, RESULTS ACHIEVED 7

CONCULSIONS 10

RECOMMENDATIONS 13

REFERENCES 22

APPENDICES 23

1 Problem – predicting transportation network dynamic from sparse mobile phone data

With the Internet of Things, data are becoming abundant for transportation researchers, but a computational platform is still missing that effectively combines transportation models with noisy sensor network data to make people's travel more efficient with information. In this project, we propose to apply the variational inference method that we developed earlier [1] to drive an agent-based transportation simulation in conjunction with observations about the traffic dynamic in a road network captured by sensor networks, and to use the statistics extracted from the simulation runs (parking space availability, fuel economy, travel time, flow) to help drivers to plan trips with the best fuel and time economy through an interactive driving planner. We will prototype and validate this approach on UB's North Campus.

Vehicle telematics units and smart phones afford us the opportunity to sense road networks in real time through millions of observed vehicle locations. About 42% of newly-sold vehicles in the U.S. today (and a projected 80% by 2018) are able to communicate vehicle state through a telematics unit, and about 57% of the population are connected with the Internet through smart phones. Data sets that track vehicles and people are therefore increasingly available for transportation research, such as the vehicle-tracking records from public transportation authorities[1,2], taxicab tracking from private taxicab companies[3,4,5], call-detail records from telecommunication service providers[6], and data from the joint efforts of government, companies, researchers, and volunteers[7]. Efforts have been made to extract an O-D matrix [2][3], to map an O-D flow to links [4][5][6][7], to visualize city-wide population density dynamics [8], to detect drivers' locations, trips, types of trips [9][10] and mode of trips (private or public vehicle) [11], and to detect poverty from trip statistics [12].

However, in order to track citywide road network dynamics directly from observed vehicle locations, we have to estimate traffic states at times and locations when no observed vehicles pass by; previous methods cannot solve this problem. According to Vlahogianni [19], the challenge in short-term traffic forecasting is not only to predict but also to explain phenomena at the city network level — to fuse new data sources such as those from telematics units and to easily incorporate the effects of non-recurrent conditions. A vector ARIMA model [12] and more generally a state space model [14] can capture the relationship between the flows on neighboring links; however, the assumption of continuous observations about link state is not satisfied when we track road network dynamics directly from the observed vehicle locations. A Bayesian network [15][16] can fuse heterogeneous information such as events, weather, and accidents through traffic cameras and vehicle tracking and so predict the formation and dissolution of traffic jams, but a large training data set is required in order to cover all scenarios that affect traffic-jam dynamics.

---

1 Niagara Frontiers Transportation Authority Developer Tools: http://metro.nfta.com/Contact/Developers.aspx

2 MTA Bus Time: http://bustime.mta.info/wiki/Developers/Index

3 NYCT&L Taxi Data: https://uofi.app.box.com/NYCtaxidata

4 Cabspotting: http://cabspotting.org/

5 mPat sample data: http://cloud.siat.ac.cn/mpat/

6 Data for Development of Ivory Coast and Senegal: http://www.netmob.org/

7 Mobile Millennium: http://traffic.berkeley.edu/

The same is true about other non-parametric machine learning approaches, such as support vector regression, neural networks, and the nearest-neighbors method [17]. Multi-agent modeling [18] can explain phenomena at the city network level in terms of how individual drivers plan their trips, and can easily incorporate the effects of non-recurrent conditions; however, it is nontrivial to combine multi-agent models and data streams to allow short-term traffic forecasting.

In this project, we prototyped a method that identifies an agent-based transportation simulator as a stochastic process. In this simulator, drivers take trips and links respond to traffic demands probabilistically — the system changes its state according to a sequence of events that identify the probabilistic interactions between drivers and road segments. Given a simulator identified as a stochastic process and the trajectories of the observed vehicles, we can search in the probability space of the link state (speed, flow, or capacity) trajectories and individual trips that best match our observations about the tracked vehicles. Starting from the number and behavior of tracked vehicles in a road link, we can determine the total number of vehicles in the link by scaling and estimating traffic conditions. If we trace the origins and destinations of the estimated number of vehicles through the behavior of the simulator and fill any gaps with prior individual travel behaviors, we can extract information about the traffic at other road links. If we then iterate estimations between the traffic at links and the trip choices of simulated vehicles, we improve our estimation of both.

The challenge in making probabilistic inferences about road networks is that we have to deal with an exploding state space: in a simple task of tracking the binary states (free vs. jammed) of 50 links, we must cope with 2^{50} combinatorial states because the links interact with one another. To make inferences about citywide road networks from the trajectories of observed vehicles, we have to cope with the state trajectories of both thousands of links and a number of simulated vehicles that is at least 1% of the total vehicles running in the real-world system. To cope with the exploding state space, we use mean field approximation: the probabilistic evolution of a link state or a vehicle trajectory is determined according to the mean field average effect of the probability evolutions of the states of other links and the trajectories of other vehicles. The variational framework to make inferences about stochastic processes was developed in the field of machine learning as minimizing Bethe variational principle, expectation propagation, and loopy belief propagation [20][21].

2 Approach and Methodology – integrating simulation modeling and sparse data with approximate inference

Through the proposed project, we intend to bridge the fields of agent-based modeling and machine learning, and enable researchers to combine the power of simulation modeling with the power of big data to help people. The key realization behind the proposed work is that an agent-based transportation simulator generates different sample paths with different probabilities — it therefore defines a stochastic process with a probability measure assigned to the space of the sample paths that describe the interactions between vehicles and roads. In this stochastic process, the system state as a function of time is composed of the states of the simulated vehicles and the states of the links. The state of a simulated vehicle is composed of its current location and its trip plan. The state of a road segment is composed of the number of vehicles on this segment and whether the road is in a free or congested state. This stochastic process is driven by a number of events, such as $p \circ l_i \rightarrow p \circ l_j$ — a vehicle p leaves link/facility l_i and enters link/facility l_j. These events change the world state, and happen with event rates that are functions of the current world state. A sample path of the stochastic process is defined by a sequence of events and the corresponding times when those events happened. From the sequence of events and times, we can unambiguously recover the system state as a function of time. An agent-based simulator thus iteratively samples the next event according to event rates then changes the world state according to the sampled state starting from the initial state, until the required amount of simulated time has passed. Mathematically speaking, the dynamics of a transportation network defined by a set of events $p \circ l_i \rightarrow p \circ l_j$ is a Markov process induced by an agent-based model.

Algorithm: Markov process induced by a multi-agent model

Input: initial world state $x(t=0)$, events $v = 1, \ldots, V$ each happening with rate $h_k(x_t, c_k) = c_k g(x_t)$, and change world state $x_{t^-} \rightarrow x_t$.
Output: a series of times when productions are triggered, the IDs of the triggered productions, and the corresponding states brought about by the triggered productions $\{t_i, v_i, x(t_i): i\}$ where $0 = t_0 < t_1 < \cdots < t_n < t_{n+1} = T$, $v_i \in \{1, \ldots, V\}$, $x(t_i^-) \xrightarrow{v_i} x(t_i)$ and $x(t)$ is the right limit and the time series $x(t)$ is left continuous.
Procedure:
Basis set current time to $t_0 = 0$, set the current state to $x(t_0)$, repeat the following step until the current time $t_{i+1} > T$.
Induction sample the next reaction time $\tau \sim$ Exponential $(\sum_k h_k(x(t_i), c_k))$, sample the next reaction $v_i + 1 \sim h_k / \sum_k h_k$, set current time to $t_{i+1} = t_i + \tau$, and update world state $x(t_i + 1)$ according to production v_{i+1}.

Such a model defines a stochastic process. The probability for this sequence of events $(t_i, v_i, x(t_i): i)$ to happen is $P\,(v, x) = \prod_i h_{v_i}(x_t) \cdot \exp\left(-\sum_i h_{v_i}(x_t) \cdot (t_{i+1} - t_i)\right)$ [22].

With this stochastic process view of an agent-based simulator we can search in the probability space of all link responses to travel demands and all vehicle trips, and so make inferences about the system evolution that maximize the likelihood of our observations about the probe vehicles. These inferences follow a forward-backward schema. Suppose we have observed a tracked vehicle in state S_{t_1} at time t_1 and in state S_{t_2} at time t_2, and we want to know the probability distribution of the state of this vehicle S_t for $t_1 \leq t \leq t_2$. We first iteratively update the probability distribution of the state S_t for t from t_1 to t_2 in the forward step, according to how this vehicle moves and starting from S_{t_1}. We then iteratively update our previous estimation of S_t for t from t_2 to t_1 in the backward step, starting from vehicle state S_{t_2}. After the forward step and the backward step, the probability distribution of vehicle state S_t for $t_1 \leq t \leq t_2$ is conditioned on both its state S_{t_1} and its state S_{t_2}.

Let $(X_t, Y_t\colon t)$ be a discrete-time state-space model (Kalman filter and hidden Markov model) with hidden states X_t and observations Y_t, identified by a transition probability $P(X_{t+1}|X_t)$ and an observation model $P(Y_t|X_t)$. The forward-backward algorithm to make inferences about hidden states X_t from observations Y_t is comprised of a forward/filtering sweep to compute the forward statistics $\alpha(X_t) = P(X_t\,|Y_1, \dots, Y_T)$ and a backward/smoothing sweep to estimate the one-slice statistics $\gamma(X_t) = P(X_t|Y_1, \dots, Y_T)$. From the forward statistics and the one-slice statistics we can extract the backward statistics $\beta(X_t) = \gamma(X_t)/\alpha(X_t)$ and the two-slice statistics $\xi(X_t, X_{t+1}) = \alpha(X_t)P(Y_{t+1}, X_{t+1}|X_t)\beta(X_{t+1})$.

The challenge with making inferences about the system dynamics of a transportation network is that we have to search in a formidable state space — $X_t = \left(X_t^{(1)}, X_t^{(2)}, \cdots, X_t^{(M)}\right)$, where the superscripts $1, \cdots, M$ represent the states of the links or the states of the agent vehicles. We therefore estimate the state distributions of links and vehicles in an amiable state space with mean field approximation:

Minimize over $\xi_t^{(m)}\left(x_{t-1}^{(m)}, x_t^{(m)}, v_{t-1}\right)$:

$$\sum_{t,x_{t-1,t}} \prod_m \xi_t^{(m)} \log \frac{\prod_m \xi_t^{(m)}}{P(x_t, v_{t-1}|x_{t-1})} - \sum_{t,x_t} \prod_m \gamma_t^{(m)} \log \prod_m \gamma_t^{(m)}$$

Subject to:

$$\sum_{x_{t-1}^{(m)}, v_{t-1}} \xi_t^{(m)}\left(x_{t-1}^{(m)}, x_t^{(m)}, v_{t-1}\right) = \gamma_t^{(m)}\left(x_t^{(m)}\right),$$
$$\sum_{x_t^{(m)}, v_{t-1}} \xi_t^{(m)}\left(x_{t-1}^{(m)}, x_t^{(m)}, v_{t-1}\right) = \gamma_{t-1}^{(m)}\left(x_{t-1}^{(m)}\right),$$
$$\sum_{x_t^{(m)}} \gamma_t^{(m)}\left(x_t^{(m)}\right) = 1.$$

Taking the derivative of the expression involving Lagrange multipliers over $\xi_t(x_{t-1}, x_t, v_t)$ and $\gamma_t^{(m)}\left(x_t^{(m)}\right)$, we see that $\alpha_t^{(m)}\left(x_t^{(m)}\right) = \exp\left(\sum_{(i)} \alpha_{t,i}^{(m)} \cdot 1\left(x_t^{(m)} = i\right)\right)$ is associated with the marginalized forward probabilities, $\beta_t^{(m)}\left(x_t^{(m)}\right) = \exp\left(\sum_{(i)} \beta_{t,i}^{(m)} \cdot 1\left(x_t^{(m)} = i\right)\right)$ is associated with the marginalized backward probabilities, with $\gamma_t^{(m)}\left(x_t^{(m)}\right) = \alpha_t^{(m)}\left(x_t^{(m)}\right)\beta_t^{(m)}\left(x_t^{(m)}\right)$. The dual optimization problem is to find the marginal forward statistics $\alpha_t^{(m)}\left(x_t^{(m)}\right)$ and the marginal backward statistics $\beta_t^{(m)}\left(x_t^{(m)}\right)$ to maximize the approximate partition function, and the solution is the fixed point of of the two-slice statistics, where normalization constant $Z_t = P\left(y_t \middle| y_{1,\cdots,t-1}\right)$:

The solution to the above Bethe variational principle through Legendre-Fenchel transform is one in which the agent vehicles and links evolve their states marginally according to the average effects of the other vehicles and links [1]. As such, instead of searching the joint probability space involving $\prod_m |X_t^{(m)}|$ states per time step, we search the marginal probability spaces of $(X_1^{(m)}, \cdots, X_T^{(m)})$ each involving $|X_t^{(m)}|$ states.

We also formulated the multi-agent discrete event decision process (MDEDP) to model the decentralized self-interested decision-making of a large population from incomplete information in a complex system. An MDEDP is a stochastic process defined by a series of variables: the state of the agents, $x_t = x_t[1], \cdots, x_t[M]$; control variables representing the action taken by the agents, $a_t = a_t[1], \cdots, a_t[M]$; events v_t that change agent state from x_{t-1} to $x_t = x_{t-1} + \Delta_{v_t}$; observations about agent states, $y_t = y_t[1], \cdots, y_t[M]$; the expected reward for individual agents; the observation model, where observations on agent states are conducted independently and are shared by all agents; policy, how agents stochastically set action variables; and the state transition model, how agent states and action variables jointly determine event rates, where the indicator function $\delta_{x_{t+1}, x_t + \Delta_{v_{t+1}}}$ is 1 if the current state is $x_t = x_{t-1} + \Delta_{v_t}$ and 0 otherwise. Our goal is to maximize the expected future reward of all agents from the observation history $\{y_t: t = 1, \ldots, T\}$ in the MDEDP defined by the probability measure $p(a_{0:T}, v_{1:T}, x_{0:T}, y_{1:T})$ through identifying a_T.

$$\arg\max_{a_0} \mathbb{E}_{x_{0:\infty}, a_{0:\infty}, v_{0:\infty} | y_{-\infty:0}} \left(\sum_{t=0}^{\infty} \gamma_t \sum_{m=1}^{M} r_t[m]\right)$$
$$p(a_{0:T}, v_{1:T}, x_{0:T}, y_{1:T}) = \prod_{t=0}^{T-1} p(a_t, x_{t+1}, y_{t+1}, v_{t+1} | x_t) p(x_0)$$
$$p(a_t, x_{t+1}, y_{t+1}, v_{t+1} | x_t) = p(a_t | x_t) p(v_{t+1} | x_t, a_t) \delta_{x_{t+1}, x_t + \Delta_{v_{t+1}}} p(y_{t+1} | x_{t+1})$$
$$p(v_{t+1} | x_t, a_t; \theta) = \begin{cases} 1 - \sum_k \tau \cdot \mathrm{h_k}(\mathrm{x_t}, \mathrm{a_t}), & \mathrm{v_{t+1}} = \emptyset \\ \tau \cdot \mathrm{h_k}(\mathrm{x_t}, \mathrm{a_t}), & \mathrm{v_{t+1}} = \mathrm{k} \end{cases}$$
$$p(y_t | x_t) = \prod_{m=1}^{M} p\left(y_{t[m]} \middle| x_{t[m]}\right)$$
$$p(a_t | x_t; \pi) = \prod_{m=1}^{M} p(a_t[m] | x_t; \pi[m])$$
$$r[m](x, a) = \mathbb{E}\left[\mathcal{R}[m] \middle| x[m], a[m]\right]$$

Each agent represents one vehicle. We model road traffic dynamics through a single type of event, $p_i \circ l_j \xrightarrow{a_j} p_i \circ l_k$, a vehicle i moving from link/building j to link/building k at rate $c_{j,k} \cdot a_j$, changing the location of the vehicle from $X_t^{(p_i)} = l_j$ to $X_{t+1}^{(p_i)} = l_k$. Here event rate is defined as the probability for the event to happen per unit of time, as time falls to 0.

We model road traffic dynamics through a single type of event, $p_i \circ l_j \rightarrow p_i \circ l_k$, --- a vehicle i moving from link/building j to link/building k with rate constant c_{l_j, l_k}, changing the location of the vehicle from $X_t^{(p_i)} = l_j$ to $X_{t+1}^{(p_i)} = l_k$, changing the number of vehicles on link l_j from $X_t^{(l_j)} = x_t^{(l_j)}$ to $X_{t+1}^{(l_j)} = x_t^{(l_j)} - 1$, and changing the number of vehicles on link l_k from $X_t^{(l_k)} = x_t^{(l_k)}$ to $X_{t+1}^{(l_k)} = x_t^{(l_k)} + 1$. According to this model, a vehicle stays at link/building j for an average duration

$1/\sum_k c_{l_j,l_k}$ and upon exiting chooses a downstream link/building with a probability proportional to the rate constant $c_{l_j,l_k}/\sum_{k'} c_{l_j,l_{k'}}$.

We assume that the probe vehicles are chosen randomly from the system. Let x_{ttl} be the total number of vehicles in the system and y_{ttl} be the total number of observed vehicles. The probability of observing $y_t^{(l_j)}$ probe vehicles at location j conditioned on there being $x_t^{(l_j)}$ vehicles in total is $p\left(y_t^{(l_j)} \middle| x_t^{(l_j)}\right) = \binom{x_t^{(l_j)}}{y_t^{(l_j)}}\binom{x_{ttl}-x_t^{(l_j)}}{y_{ttl}-y_t^{(l_j)}} / \binom{x_{ttl}}{y_{ttl}}$. Here we use ``n choose k'' notation. When the total number of vehicles in the system is large, the percentage of probe vehicles at a link/building is roughly the same percentage of probe vehicles in the system.

We define four possible events in the system: vehicle leaving a building, vehicle entering a link, vehicle leaving a link, and vehicle entering a building. From these four events, we can construct a state transition matrix to represent vehicle dynamics.

3. Findings

We compare the performance of the proposed particle filter algorithm against other algorithms on three data sets of human mobility: SynthTown, Berlin and Dakar.

The SynthTown data set is comprised of a synthesized network of one home location, one work location, and 23 single-direction road links to characterize the trips of 2000 synthesized inhabitants going to work in the morning and retuning home in the afternoon. The graphical illustration is shown in Figure. The prediction problem is to estimate the vehicle counts at home, at work, and at links 1-23 in the present time, 10 minutes later, and 60 minutes later from observations of the 200 ``probe'' inhabitants collected at link 1 and link 20. These 200 probe inhabitants volunteer to share their locations every minute. Simple as it seems, this problem requires a statistical inference algorithm to ``understand'' several concepts in order to achieve successful tracking and forecasting. For example, the algorithm should successively add the estimated vehicle count at link 1 to home and subtract the estimated vehicle count at link 20 from work. In addition, the estimated vehicle counts at link 21-23 should sequentially follow the estimated vehicle count at link 20 and be followed by the estimated vehicle count at link 1.

The Berlin and Dakar data are much larger. They show the capability of the proposed algorithms to work with more complex dynamics and larger data sets. The Berlin data set is comprised of a network of 24,000 single-direction road links derived from Open Street Map and the trips of 9,000 synthesized vehicles representing the travel behaviors of one million vehicles. The trips in the Berlin set were carefully validated with survey and sensor network data, and provide the ground truth for evaluating algorithms in a semi-realistic configuration. We aggregate the 24,000 road links into 1539 clusters with a walktrap algorithm to make the problem small enough for benchmarking. The Dakar data set is comprised of a network of 8,000 single-direction road links derived from Open Street Map and 12,000 real-world vehicle trips derived from the Data for Development call detail records.

We use two metrics to evaluate the performance of our model: coefficient of determination (R^2) and mean squared error (MSE). We use R^2 to evaluate the goodness of fit between a time series of the estimated vehicle counts at a location and the ground truth. Let f_t be the estimated vehicle count at time t, y_t the ground truth and $\bar{y}$ the average of y_t. We define $R^2 = 1 - \sum_t (f_t - y_t)^2 / \sum_t (y_t - \bar{y}_t)^2$. A higher R^2 indicates a better fit between the estimated time series and the ground truth, with $R^2 = 1$ indicating a perfect fit and $R^2 < 0$ a fit worse than using the average.

We use MSE to measure the average squared error difference between the estimated vehicle counts at all locations at a time t and the ground truth. A lower MSE represents a more precise prediction. Let $f^{(i)}$ be the estimated vehicle count at location i and $y^{(i)}$ the ground truth. We define $MSE = \frac{1}{n} \sum_{(i=1)}^{n} \left(\left(y^{(i)} \right) - f^{(i)} \right)^2$. A lower MSE indicates an estimation closer to the ground truth.

We apply the variational inference (VI), deep neural network (DNN), recurrent neural network (RNN), and extended Kalman filter (EKF) to solve the vehicle tracking and prediction problems on the SynthTown, Berlin, and Dakar datasets. In the following, we first inspect how those four models track and predict traffic dynamics at different times in a day as well as different locations in detail on the SynthTown data set, and then compare the summary performance statistics on all data sets.

Figure shows how VI, DNN, RNN and EKF predict the numbers of vehicles at different locations of SynthTown one hour ahead of time throughout a day from observations of probe vehicles (10% of the total) at link 1 and link 20 only. The x-axis indicates the hours of a day, the y-axis shows the numbers of vehicles at different locations --- home, work and road segments marked on the left, and the ground truth (GT) serves as the frame of reference.

All four algorithms perform well, indicating that they all get the structure in the dynamics. In fact, there is little uncertainty about the traffic dynamics at SynthTown if the numbers of vehicles on link 1 and 20 can be monitored, albeit with noise. RNN underperforms the other three algorithms because learning the structure of a dynamical system requires a huge training data set. VI estimation agrees with GT and it is better than DNN and RNN estimations, this is because VI explicitly leverages the problem specific structure, i.e., road topology, while DNN and RNN need to learn it implicitly and gradually. VI is better than EKF estimation, because VI can work with arbitrary probability distributions while EKF assumes Gaussianity. EKF and DNN agrees well with GT at locations with a lot of people (home and work), and less well at locations with a few people. It shows VI can adopt dynamic changes better.

Figure compares the summary MSE and R^2 performance statistics of the four models in vehicle tracking, i.e., estimating the numbers of vehicles up to now, short term prediction (10 minutes) and long term prediction (1 hour) on all data sets. The Dakar dataset is too large for DNN, RNN and EKF, which indicates the better scalability of VI. The comparison leads us to the same conclusions as the detailed comparison on the SynthTown data. Specifically, VI has the lowest MSE across different times of a day, which is followed by DNN, EKF, and RNN in order (top row, lower is better); VI has the highest R^2 across different locations, which is followed by DNN, EKF, and RNN (bottom row, higher is better). First, VI outperforms RNN and DNN because it can explicitly leverage the problem specific structure such as road topology. Second, VI outperforms EKF because it can work with arbitrary probability distributions and sometimes Gaussian assumption is not a good approximation for the real world applications. This comparison also points to new development of neural network architectures that are either regularized by event-based structures of a complex system or can learn such structures explicitly.

3. Findings

In this project, in order to leverage the data generated by the Internet of things and crowdsourcing applications, we developed a viarational inference algorithm with stochastic kinetic model algorithm to continuously track the dynamics of large systems. In addition, we use the stochastic kinetic model to represent dynamic transition probabilities and reduce their dimensionality. Large scale experiments show that our proposed algorithm can accurately track and predict city-scale traffic dynamics and outperform existing algorithms based on deep learning and kalman filter algorithms.

References

[1] W. Dong, Interpreting big data with agent-based modeling. To appear: Proceedings of 2015 Winter Simulation Conference.

[2] M. Nanni, R. Trasarti, B. Furletti, L. Gabrielli, P. Mede, J. Bruijn, E. Romph, and G. Bruil, “Mp4-a project: mobility planning for africa,” in D4d challenge@ 3rd conf. on the analysis of mobile phone datasets (netmob 2013), 2013.

[3] J. Steenbruggen, M. T. Borzacchiello, P. Nijkamp, and H. Scholten, “Mobile phone data from gsm networks for traffic parameter and urban spatial pattern assessment: a review of applications and opportunities,” Geojournal, vol. 78, iss. 2, pp. 223-243, 2013.

[4] J. L. Toole, S. Colak, F. Alhasoun, A. Evsukoff, and M. C. Gonzalez, “The path most travelled: mining road usage patterns from massive call data,” Arxiv preprint arxiv:1403.0636, 2014.

[5] P. Wang, T. Hunter, A. M. Bayen, K. Schechtner, and M. C. González, “Understanding road usage patterns in urban areas,” Scientific reports, vol. 2, 2012.

[6] F. Calabrese, F. C. Pereira, G. Di Lorenzo, L. Liu, and C. Ratti, “The geography of taste: analyzing cell-phone mobility and social events,” in Pervasive computing, Springer, 2010, pp. 22-37.

[7] D. Quercia, N. Lathia, F. Calabrese, G. Di Lorenzo, and J. Crowcroft, “Recommending social events from mobile phone location data,” in Data mining (icdm), 2010 ieee 10th international conference on, 2010, pp. 971-976.

[8] S. Isaacman, R. Becker, R. Cáceres, S. Kobourov, M. Martonosi, J. Rowland, and A. Varshavsky, “Identifying important places in people’s lives from cellular network data,” in Pervasive computing, Springer, 2011, pp. 133-151.

[9] S. Jiang, G. A. Fiore, Y. Yang, J. Ferreira Jr, E. Frazzoli, and M. C. González, “A review of urban computing for mobile phone traces: current methods, challenges and opportunities,” in Proceedings of the 2nd acm sigkdd international workshop on urban computing, 2013, p. 2.

[10] F. Calabrese, L. Ferrari, and V. D. Blondel, “Urban sensing using mobile phone network data: a survey of research,” Acm computing surveys (csur), vol. 47, iss. 2, p. 25, 2014.

[11] M. Berlingerio, F. Calabrese, G. Di Lorenzo, R. Nair, F. Pinelli, and M. L. Sbodio, “Allaboard: a system for exploring urban mobility and optimizing public transport using cellphone data,” in Machine learning and knowledge discovery in databases, Springer, 2013, pp. 663-666.

[12] Neeti Pokhriyal, Wen Dong, and Venugopal Govindaraju. Virtual Networks and Poverty Analysis in Senegal. In NetMob 2015. MIT Media Lab, MA, USA, 2015 (**National Statistics award of Data for Development challenge**)

[13] M. Ben-Akiva, M. Bierlaire, H. Koutsopoulos, and R. Mishalani, “Dynamit: a simulation-based system for traffic prediction,” in Daccors short term forecasting workshop, the Netherlands, 1998.

[14] Y. Wang, M. Papageorgiou, and A. Messmer, “Real-time freeway traffic state estimation based on extended kalman filter: adaptive capabilities and real data testing,” Transportation research part a: policy and practice, vol. 42, iss. 10, pp. 1340-1358, 2008.

[15] E. J. Horvitz, J. Apacible, R. Sarin, and L. Liao, “Prediction, expectation, and surprise: methods, designs, and study of a deployed traffic forecasting service,” Arxiv preprint arxiv:1207.1352, 2012.

[16] A. Krause, E. Horvitz, A. Kansal, and F. Zhao, “Toward community sensing,” in Proceedings of the 7th international conference on information processing in sensor networks, 2008, pp. 481-492.

[17] H. Yin, S_C. Wong, J. Xu, and C. Wong, “Urban traffic flow prediction using a fuzzy-neural approach,” Transportation research part c: emerging technologies, vol. 10, iss. 2, pp. 85-98, 2002.

[18] MATSim development team (ed.), Matsim-t: aims, approach and implementation, ivt, eth zürich, zürich., 2007.

[19] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, “Short-term traffic forecasting: where we are and where we’re going,” Transportation research part c: emerging technologies, vol. 43, pp. 3-19, 2014.

[20] T. Heskes and O. Zoeter, “Expectation propagation for approximate inference in dynamic bayesian networks,” in Proc. of uai, 2002, pp. 216-223.

[21] M. J. Wainwright and M. I. Jordan, “Graphical models, exponential families, and variational inference,” Foundations and trends\textregistered in machine learning, vol. 1, iss. 1-2, pp. 1-305, 2008.

[22] D. J. Wilkinson, Stochastic modeling for systems biology, CRC press, 2011.

[23] S. Kullback, Information theory and statistics, Courier Corporation, 1968.

[24] L. Guo, H. Shan, and A.W. Sadek. "A novel agent-based transportation model of a university campus with application to quantifying the environmental cost of parking search." Transportation Research Part A: Policy and Practice, vol. 50, pp. 86-104, 2013.