A Sensor Network Architecture for Urban Traffic State Estimation with Mixed Eulerian/Lagrangian Sensing Based on Distributed Computing

Edward Canepa, Enas Odat, Ahmad Dehwah, Mustafa Mousa, Jiming Jiang, and Christian Claudel

King Abdullah University of Science and Technology, Thuwal 23955, Saudi Arabia
firstname.lastname@kaust.edu.sa

Abstract. This article describes a new approach to urban traffic flow sensing using decentralized traffic state estimation. Traffic sensor data is generated both by fixed traffic flow sensor nodes and by probe vehicles equipped with a short range transceiver. The data generated by these sensors is sent to a local coordinator node, that poses the problem of estimating the local state of traffic as a mixed integer linear program (MILP). The resulting optimization program is then solved by the nodes in a distributed manner, using branch-and-bound methods. An optimal amount of noise is then added to the maps before dissemination to a central database. Unlike existing probe-based traffic monitoring systems, this system does not transmit user generated location tracks nor any user presence information to a centralized server, effectively preventing privacy attacks. A simulation of the system performance on computer-generated traffic data shows that the system can be implemented with currently available technology.

1 Introduction

Traffic congestion is an increasing concern in large urban areas of the world, and is expected to become worse as global traffic demand increases. While traffic control methods such as ramp metering, adaptive speed limits and demand response could solve the problem to a certain extent, such methods require as an input accurate traffic density, velocity and flow estimates.

In the recent years, probe vehicles (*i.e.* vehicles containing speed and/or position sensors) have emerged as a possible solution to the traffic monitoring problem. Probe sensing offers the potential for low cost sensing (in contrast to expensive fixed traffic sensor networks), in particular when sensing relies on existing devices (for instance smartphones), see for instance [24]. Nevertheless, all current probe-based traffic monitoring systems require users to send their location data to a centralized server, which carries high risks of user privacy intrusion whenever the location data servers are attacked. It should be noted that even anonymous location tracks can yield substantial information on users [17],

E. Maehle et al. (Eds.): ARCS 2014, LNCS 8350, pp. 147–158, 2014.

which can be correlated with social network data to identify user identity based on their tracks. Such privacy risks are one of the main reasons preventing the large-scale implementation of cheap transceivers and positioning devices on all vehicles (despite the considerable societal benefits), specially since the recent PRISM revelations.

Several attempts to address the user privacy issues in probe-based traffic monitoring systems have been made [18]. All techniques either modify the sampling characteristics [14] (locations of samples, sampling rate) or attempt to obfuscate the real data trace by either removing data points or adding fake data points (or noise). A spatial sampling method called *virtual trip lines* (VTLs) is proposed in [15], to prevent users from sending their data whenever they are close to locations that could help identify them (home, workplace). However, this method is not applicable for traffic monitoring in urban environments since most urban areas are either workplaces or accommodations. Another obfuscation method is shown in [21], but the same article shows that generating fake data to hide real location tracks is challenging, even with aggregated statistical data.

The above privacy issues can only addressed at the system level if the system estimates the traffic flow conditions in a decentralized manner, since a central server receiving user data (even temporarily) would be a primary target for a privacy attack. In this article, we propose a new heterogenous sensor network architecture for traffic flow sensing in which user-generated data is processed by the nodes of the sensor network, possibly together with data generated by existing traffic sensors, to generate traffic estimates directly. By construction of this system, no information related to the presence of any user located outside of the radio range of a cluster (of configurable size) can be inferred.

This article is organized as follows. We present the sensing paradigm in section 2, including the distributed computing aspect of the system. We then study the privacy properties of the resulting system in section 3, with an analysis of the possible privacy attack scenarios. We then present in section 5 an ongoing wireless sensor network implementation (with currently no distributed computing algorithms implemented), with an associated simulation of the performance of the system.

2 Sensing Paradigm

2.1 Current Architecture of Probe-Based Traffic Sensing Systems

Probe-based traffic sensing systems follow typical sensor network architectures, in which data generated by sensors is sent to a centralized server for processing or display [22]. Traffic speed and/or density maps are the end product for the user, and the basis of all other location-based services such as travel time estimation or optimal routing. The architecture of such systems is illustrated in Figure 1.

One of the major drawbacks of such systems is the fact that the ID proxy server holds privacy sensitive information regarding the users. Privacy of users is at risk even when data is anonymized [17], therefore even the input database of the system can contain privacy sensitive data. While some systems [22] attempt

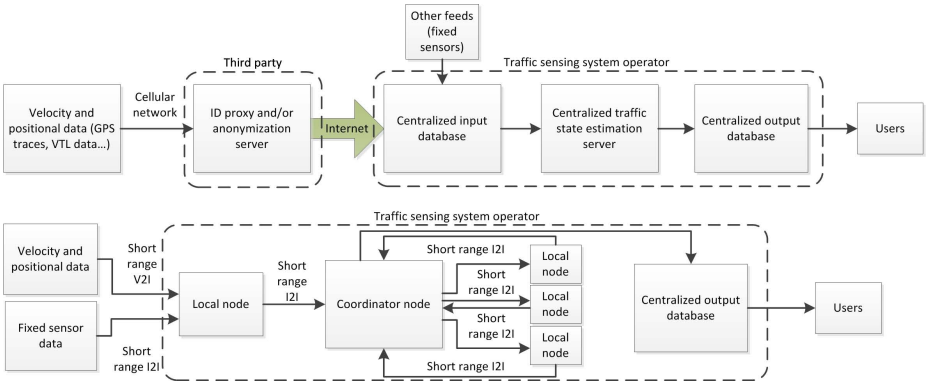

Fig. 1. Traffic monitoring systems architectures
Up: current systems. The data generated by users is sent to a proxy server for obfuscation/anonymization . The resulting data is then sent to an input database that also collects data from the existing fixed sensor infrastructure (if it exists). The resulting data is the fused with traffic flow models (a process sometimes referred to as *data assimilation*) to generate traffic maps. The resulting estimates are sent to an output database, which is queried by the users.
Down: proposed system. In the proposed system, traffic estimation is integrated to the wireless sensor network, which computes the traffic maps in each cluster using distributed computing. The resulting traffic maps are then forwarded to an output database.

to solve the privacy problem using data obfuscation or specific spatial sampling strategies [15], it is important to note that none of these strategies can guarantee that user privacy is preserved in all situations. In particular, no sampling strategy can prevent the identification of the approximate path whenever only one user is present in a given geographical area (the extent of which depends on the vehicle speed and the sampling rate).

Since user location and velocity information is required by the model to build the traffic maps, and since a centralized server handing user data can always be a target of attacks, the above privacy issues can be solved only if the information about the user location and velocity is used locally. This implies that the traffic state estimation process, which consists in fusing traffic flow data with traffic flow models, can only be done locally (for instance by the sensor nodes themselves). In this approach, privacy-critical information (location tracks) are not sent to a centralized location, and remain in a small (configurable) area around the probe vehicle.

2.2 Proposed System Architecture

Our proposed system consists in an heterogenous wireless sensor network, connected to a centralized output database. The database itself can directly be queried by the clients, or feed other on-demand location based services such as optimal routing or travel time estimation.

Fixed Sensor Nodes

The sensor nodes play three roles: communication (in the wireless sensor network), computation (distributed traffic inference, vehicle positioning, fixed sensor data processing) and sensing. Two types of sensing approaches exist: Eulerian (fixed) or Lagrangian (mobile) sensing. *Eulerian sensing* nodes consist in fixed traffic flow sensors, for instance inductive loop detectors [25], magnetometers or traffic cameras. The remaining nodes are called *Lagrangian sensing* nodes, and collect traffic data from users in their vicinity using a short range transceiver. All nodes are forming a wireless mesh network. The output database is in wireless range of the rest of the network. As in any wireless sensor network multiple databases/gateways can be used to reduce network load.

Principle of Operation

The network of fixed nodes is divided into clusters. In a given cluster, the nodes form a subnetwork (for traffic estimation purposes) to compute the local traffic conditions. Clusters can communicate between each other, though the only data sent by a cluster to another is anonymized traffic maps. A local coordinator node is chosen in each subnetwork.

Probe vehicles broadcast their location and/or speed information to surrounding Lagrangian sensing nodes, which temporarily store this data as well as network connectivity data (RSSI, CRC). All location (if any), speed and connectivity data is forwarded to the local coordinator node. If no positional information is available to probe vehicles, the coordinator node estimates the corresponding vehicle positions in the road network using inputs from surrounding nodes. Vehicle mapping can be done through a variety of methods, for example using RSSI data.

In addition to the data transmitted from local Lagrangian sensing nodes, the coordinator receives traffic data generated by Eulerian sensor nodes in the subnetwork.

Since traffic data is sparse and of different nature, reconstructing the state of traffic everywhere requires the combination of available data with traffic flow models, a process sometimes referred to as *data assimilation*. In our present case, we consider all incoming traffic data during the time window $[t - \Delta t, t]$ to estimate the traffic state at time t. The data assimilation method used in this article is outlined in the subsequent sections.

While no user information is directly present in this traffic map, it may nonetheless reveal user presence in some circumstances. To make the system completely privacy preserving, one needs to obfuscate the presence of users in the resulting maps (a problem that all traffic monitoring systems have, irrespective of their internal mechanisms). Such methods are detailed in section 3.

The anonymized density maps are then forwarded by other clusters to a central database for dissemination to the users, using multi-hop communication.

3 User Privacy Analysis

3.1 Threat Model

In this article, we assume that attackers can compromise any part of the system, that is, any individual node, any local coordinator, and any output database.

3.2 Properties of the System

By construction, no vehicle track information can be obtained beyond the radio range of the cluster in which the vehicle lies. Thus, an eavesdropper can "track" a vehicle's position only if he/she can listen to all clusters in the path of the vehicle. While such a distributed attack is theoretically possible, it is very costly and impractical, requiring the deployment of listening nodes in all clusters (which have independent encryption keys).

3.3 Privacy Attacks and Countermeasures

Compromising a Local Coordinator. Since all information in a cluster is handled by the local coordinator, the worst-case attack is to compromise it to obtain the position or velocities of all vehicles in the cluster (though these positions remain anonymous). Other attacks would result in partial knowledge of the position of vehicles in the radio range of the attacker.

This type of attack can be countered in two ways. First, the size of a cluster can be made arbitrarily small, to minimize the extent of the privacy intrusion. There is a tradeoff though, as smaller size will yield less accurate results for the estimation process due to the uncertainty in the estimated boundary conditions.

Another strategy is to change the coordinator node in the cluster periodically, using a scheduler or according to other constraints such as energy or bandwidth. Thus, an attacker compromising a coordinator would have limited knowledge (in time) of the presence of vehicles. Since it cannot be inferred which node will be a coordinator in advance (for instance if the scheduling is random), an attacker would have to physically compromise all nodes in a cluster to guarantee an access to the vehicle positions.

Possible Attacks. Based on these results, an attacker that wants to reidentify the track of a given vehicle has to compromise either all nodes in the path of the vehicle, or all coordinators in all clusters in the path of the vehicle. Thus, the system is only vulnerable to distributed privacy attacks (distributed eavesdropping).

Given the cost of such an attack (installing transceivers around all nodes, and breaking the encryption keys of all clusters), it is probably easier for an attacker to implement its own monitoring network to listen to vehicle communications directly. The system would be vulnerable to this type of attack, though an additional wireless sensor network deployed in a city would probably be detected sooner or later through its radio emissions.

Compromising an Output Database. By compromising an output database (there may be one or many output databases for the complete network), an attacker can only gain access to anonymized traffic maps (since these are the only information sent to the databases), which are also public.

While no track information from a cluster is not propagated beyond its radio range, traffic maps are propagated beyond each cluster to reach a gateway. Thus, the privacy of the user is maintained only if the problem of reconstructing trajectories from traffic maps (speed and/or density maps) does not yield a unique solution. Different anonymization strategies are possible to increase the number of solutions to the previous problem. One of the possible strategies could be the use of k-anonymity techniques [19] to determine the optimal level of noise to apply, in order to guarantee that a user is indistinguishable from others.

Note that while inferring vehicle positions from traffic maps is theoretically possible from any traffic map, it is difficult for two main reasons: low accuracy of current traffic systems, and security through obscurity from traffic providers. A wireless sensor network should not rely on security through obscurity as it is relatively easy to access one node (the code is identical in all nodes) and decompile its code.

4 Distributed Computing for Traffic State Estimation

The data assimilation scheme is based on the seminal *Lighthill Whitham Richards* [20] (LWR) traffic flow model, a first order scalar conservation law, with triangular flux function. It is here based on a decomposition of the solutions using the inf-morphism property of the solutions to the *Hamilton Jacobi* equation from which the LWR model is derived [8,9]. Using this decomposition, we write the problem of estimating traffic density on a section of road as a mixed integer linear program (MILP) [6]. The solution to the MILP correspond to a vector of current traffic densities, which can be interpreted as a traffic density map.

4.1 Input Data

Specifically, on each segment of road, the input data can take any of the following forms.

Definition 41 ***[Affine initial, boundary and internal conditions]*** *Let us define* $\mathbb{K} = \{0, \ldots, k_{\max}\}$, $\mathbb{N} = \{0, \ldots, n_{\max}\}$ *and* $\mathbb{M} = \{0, \ldots, m_{\max}\}$. *For all* $k \in \mathbb{K}$, $n \in \mathbb{N}$ *and* $m \in \mathbb{M}$, *we define the following functions, respectively called initial, upstream, downstream (boundary) and internal conditions:*

$$M_k(t,x) = \begin{cases} -\sum_{i=0}^{k-1} \rho(i)X \\ -\rho(k)(x-kX) \text{ if } t=0 \\ \text{and } x \in [kX,(k+1)X] \\ +\infty \qquad\qquad \text{otherwise} \end{cases} \qquad \gamma_n(t,x) = \begin{cases} \sum_{i=0}^{n-1} q_{\text{in}}(i)T \\ +q_{\text{in}}(n)(t-nT) \text{ if } x=\xi \\ \text{and } t \in [nT,(n+1)T] \\ +\infty \qquad\qquad \text{otherwise} \end{cases}$$

$$\beta_n(t,x) = \begin{cases} \sum_{i=0}^{n-1} q_{\text{out}}(i)T \\ +q_{\text{out}}(n)(t-nT) \\ -\sum_{k=0}^{k_{max}} \rho(k)X \quad \text{if } x=\chi \\ \text{and } t \in [nT,(n+1)T] \\ +\infty \qquad\qquad \text{otherwise} \end{cases} \quad \mu_m(t,x) = \begin{cases} L_m + r_m(t-t_{\min}(m)) \\ (\text{if } x = x_{\min}(m) \\ +\frac{x_{\max}(m)-x_{\min}(m)}{t_{\max}(m)-t_{\min}(m)}(t-t_{\min}(m)) \\ \text{and } t \in [t_{\min}(m),t_{\max}(m)]) \\ +\infty \qquad\qquad \text{otherwise} \end{cases}$$

The LWR model [20] is encoded by the following Hamilton-Jacobi [9] partial differential equation:

$$\frac{\partial \mathbf{M}(t,x)}{\partial t} - \psi\left(-\frac{\partial \mathbf{M}(t,x)}{\partial x}\right) = 0 \tag{1}$$

The function $\psi(\cdot)$ defined in equation (1) is the *Hamiltonian*. The B-J/F [4,13] solutions to equation (1) are fully characterized by a *Lax-Hopf* formula [3,8], which was initially derived using the control framework of viability theory [2]. We assume that the Hamiltonian is piecewise affine and continuous [12]:

$$\psi(\rho) = \begin{cases} v_f \rho & : \rho \in [0,k_c] \\ w(\rho-\kappa) & : \rho \in [k_c,\kappa] \end{cases} \tag{2}$$

4.2 Traffic State Estimation Using Mixed Integer Linear Programming

We consider a set of block boundary conditions 41, with unknown coefficients. Let us call V the vector space of unknown coefficients. Our measurement data (from the data set) constraints the possible values of these coefficients. Such constraints are called *data constraints*. Similarly, the PDE model also constraints the possible values of the unknown coefficients. Such constraints are called *model constraints*. An important and nontrivial result of [10] is that all these constraints are explicit. The extensive list of all constraints can be found in [6,7], though we do not write them in this article for compactness. The main result is the following:

Fact 42 ***[Mixed integer linear inequality property]*** *The model constraints [7] are mixed integer linear in the variables* $\rho(1),\rho(2),\ldots,\rho(k_{\max})$, $q_{\text{in}}(1),\ldots,q_{\text{in}}(n_{\max}), q_{\text{out}}(1),\ldots,q_{\text{out}}(n_{\max})$, $L_1,\ldots,L_{m_{\max}}$ *and* $r_1,\ldots,r_{m_{\max}}$.

The proof of this proposition is available in [7].

Similarly, the unknown coefficients of the initial, boundary and internal conditions have to satisfy data constraints to be compatible with the observations. The data constraints express the fact that the true values of the initial, boundary and internal conditions coefficients should be within the bounds of the sensor measurement errors (which are known).

Hypothesis 43 ***[Data constraints]*** *In the remainder of our article, we assume that the data constraints are linear in the unknown coefficients of the initial, boundary and internal conditions.*

Different important and practical choices of error models that yield linear data constraints are available in [6]. Among all possible choices, the L_1 norm of the initial (or final) densities is a good candidate to obtain a sparse density map.

In the remainder of this article, we define y as the decision variable of the problem, containing the continuous variables $\rho(1), \rho(2), \ldots, \rho(k_{\max})$, $q_{\text{in}}(1), \ldots, q_{\text{in}}(n_{\max}), q_{\text{out}}(1), \ldots, q_{\text{out}}(n_{\max})$, $L_1, \ldots, L_{m_{\max}}$ and $r_1, \ldots, r_{m_{\max}}$, with additional integer variables representing continuity constraints.

Using the above equations, the set of possible traffic scenarios compatible with the data and the model can be written as $\{y | Ay \leq b \text{ and} Cy \leq d\}$. To select a solution among all possible choices, we choose a linear function of y, which can represent for instance the minimal travel time or the maximal average density at the current time. We can also look for sparse solutions by minimizing the L_1 norm of y. All of these examples boil down (modulo additional slack variables) to *Mixed Integer Linear Programs* (MILPs):

$$\begin{array}{l} \text{Min. } c^T y \\ \text{s. t. } \begin{cases} Ay \leq b \\ Cy \leq d \end{cases} \end{array} \tag{3}$$

We refer the reader to [11] for examples of choices of linear objectives relevant to traffic state estimation.

4.3 Distributed Computing Principle

MILPs can be parallelized [1,16] using parallel branch and bound methods. In the present case, the coordinator will coordinate the computation of the solution to the MILP, sending branches to explore to other nodes in the cluster, under a tree topology. See [1] for an example of implementation of a parallel MILP solver. Note that the attribution of tasks is *dynamic*. Once a node has found a better optimal solution, it will broadcast its results (multi-hop communication will be used if the nodes are not all in radio range) to the remaining nodes so only branches with possibly optimal candidates can be explored.

Once the MILP is solved (if it is not solved by the deadline, then the most optimal current solution can be used in lieu of the optimal solution), the coordinator node "anonymizes" the map by adding an optimal amount of noise (if the map "reveals" the location of an user) and then sends the resulting traffic map to an output database (through other clusters). The general principle is illustrated in Figure 2.

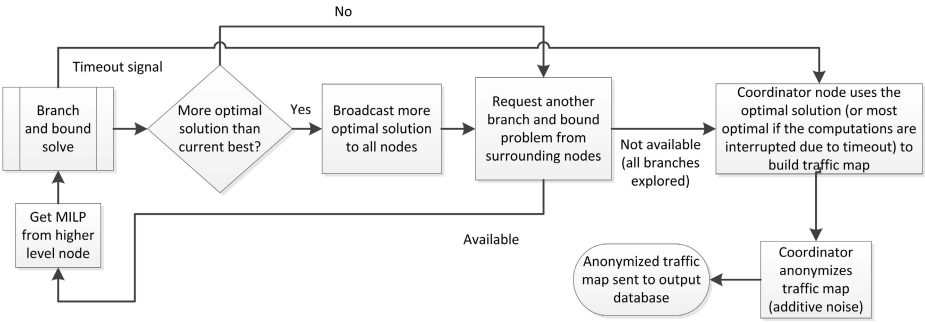

Fig. 2. Distributed computing process used by the proposed system

5 Implementation

We now present an ongoing implementation of a distributed-computing based traffic sensing system, as well as a simulation of the performance of the deployed system.

5.1 Computational Platform

In order to minimize power consumption while allowing distributed computing to be performed, we designed a new hardware platform around a 32-bit `ARM Cortex M4` normally operating at 168 MHz. The platform draws its energy jointly from a solar panel and a rechargeable $Li - FePO_4$ battery. It is designed to be OTA (over-the-air programming) capable. The current implementation of this platform is illustrated in Figure 3.

The STM32F407 microcontroller (MCU) includes a 1 Mbyte Flash memory and 196 KBytes of data RAM. It supports up to seventeen timers, 24 channels for analog to digital conversion and two 12-bit DACs. With embedded real-time memory accelerator, multi-AHB bus matrix and two dual-port DMA controllers, a maximal performance of 1.25DMIPS/MHz (Dhrystone million instructions per second per MHz) can be achieved.

5.2 Fixed Eulerian and Lagrangian Sensor Nodes

For this application, we also developed fixed traffic flow sensors that can sense both traffic and urban flash flooding (a secondary application of this system, which is out of the scope of this article). Each node is capable of monitoring traffic flow on two adjacent traffic lanes, as well as detect the presence and accurately measure the level of water in case of flooding. Measurements rely on two arrays of remote temperature (Melexis MLX90614) sensors (using passive infrared detection), as well as one ultrasonic rangefinder (MaxBotix MB7066), as illustrated in Figure 3. All sensors are digitally connected (SMBus and serial respectively) to a fixed transceiver node (described above) which generates traffic measurement data. .

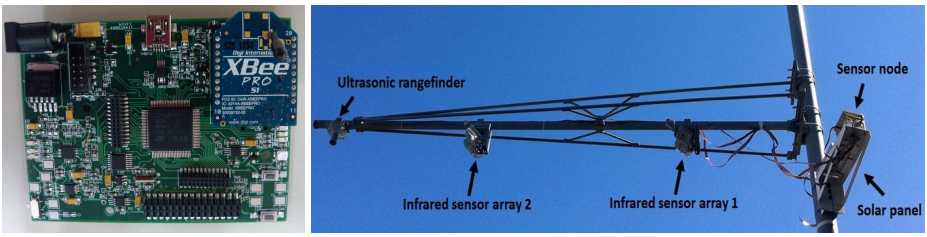

Fig. 3. Fixed sensing nodes
This figure shows the common computational platform used in the Lagrangian and Eulerian sensor nodes (left), as well as an Eulerian sensor node deployed on KAUST campus (right).

5.3 Mobile Transceivers

Mobile transceivers equipping vehicles are a key component of the proposed system, and will initially consist in dedicated low-cost modules, though they can piggyback on future V2V systems [5].

5.4 Simulated Performance of the System

Since the system is not fully functional yet (due to porting an OS and developing libraries for the OTA and for the MILP solver), we simulate the performance of an actual system using traffic data generated by the `PTV VISSIM`[23] microsimulator. We simulate a small road network consisting of 10 roads. Owing to the fact that the boundary conditions between links are known in some instances (for instance when the traffic light in a section is red), we decompose the traffic estimation problem into smaller scale subproblems involving 4 roads only, as illustrated in Figure 4. We consider a time horizon of 5 seconds and two internal conditions (obtained from vehicle position data), two boundary conditions and one initial condition (which can be the previous estimate) per road. On our 5 minutes of simulated data, The computational time required to solve it on an `Imac` with an `Intel i5`@2.5GHz varies between 25 ms and 65 ms, which translates into between 1 second to 3 seconds on our prototype experimental platform (using the `Coremark` benchmark, and assuming a similar computational efficiency between both platforms). On the simulated examples, the MILPs have between 44 and 49 variables, and between 177 and 196 constraints.

Since there exists an overhead for transmitting data (during the branch and bound process) and for the traffic sensing activities themselves, we expect that 1-2 nodes would be required to reliably estimate the traffic on these four roads, or equivalently that 4 nodes would be required for the complete set of 10 roads, which covers an area of 0.15 km^2, making this system an inexpensive traffic sensing solution.

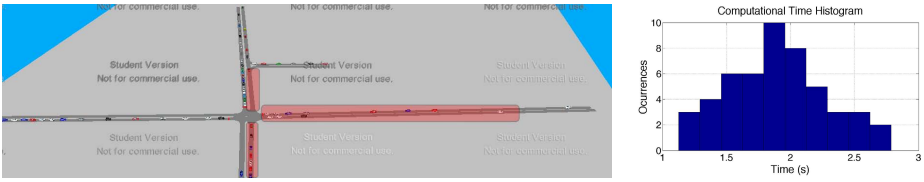

Fig. 4. Screenshot of the simulated transportation network
Left: we consider a subnetwork of four roads (in red), and compute 50 consecutive traffic estimation problems (5s time horizon).
Right: distribution of simulated computational times on the `ARM`-based platform.

6 Conclusions

This article presents a new wireless sensor network architecture for estimating traffic conditions in an urban environment based on distributed computing. Provided that the traffic estimation is distributed among a set of local nodes, we show that no user track information is sent beyond the radio range of this cluster, thereby preventing inference attacks on user location tracks. An ongoing implementation is briefly discussed, as well as a simulation of the system's performance. Future work will deal with the implementation of this system on the new ARM-based computational platform developed in our lab.

Acknowledgments. We would like to thank Guodong Li (KAUST) for his CAD design of the traffic sensor system, Sergio Favela (MS, KAUST) for his help writing the embedded code and Ehsan Wariach (PhD, U. Groningen) for his help on Eulerian sensor data processing.

References

1. Alonso, J., Schmidt, H., Alexandrov, V.N.: Parallel branch and bound algorithms for integer and mixed integer linear programming problems under PVM. In: Bubak, M., Waśniewski, J., Dongarra, J. (eds.) PVM/MPI 1997. LNCS, vol. 1332, pp. 313–320. Springer, Heidelberg (1997)
2. Aubin, J.-P.: Viability Theory. Systems and Control: Foundations and Applications. Birkhäuser, Boston (1991)
3. Aubin, J.-P., Bayen, A.M., Saint-Pierre, P.: Dirichlet problems for some Hamilton-Jacobi equations with inequality constraints. SIAM Journal on Control and Optimization 47(5), 2348–2380 (2008)
4. Barron, E.N., Jensen, R.: Semicontinuous viscosity solutions for Hamilton-Jacobi equations with convex Hamiltonians. Communications in Partial Differential Equations 15, 1713–1742 (1990)
5. Biswas, S., Tatchikou, R., Dion, F.: Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety. IEEE Communications Magazine 44(1), 74–82 (2006)
6. Canepa, E.S., Claudel, C.G.: Exact solutions to traffic density estimation problems involving the LWR traffic flow model using MILPs. In: Proceedings of the 15th IEEE ITSC Conference, Anchorage, AK (September 2012)

7. Canepa, E.S., Claudel, C.G.: Spoofing Cyber Attack Detection in Probe-based Traffic Monitoring Systems using MILP. In: Proceedings of IEEE ICNC, San Diego, CA (January 2013)
8. Claudel, C.G., Bayen, A.M.: Lax-Hopf based incorporation of internal boundary conditions into Hamilton-Jacobi equation. Part I: theory. IEEE Transactions on Automatic Control 55(5), 1142–1157 (2010), doi:10.1109/TAC.2010.2041976.
9. Claudel, C.G., Bayen, A.M.: Lax-Hopf based incorporation of internal boundary conditions into Hamilton-Jacobi equation. Part II: Computational methods. IEEE Transactions on Automatic Control 55(5), 1158–1174 (2010), doi:10.1109/TAC.2010.2045439.
10. Claudel, C.G., Bayen, A.: Convex formulations of data assimilation problems for a class of Hamilton-Jacobi equations. SIAM Journal on Control and Optimization 49, 383–402 (2011)
11. Claudel, C.G., Chamoin, T., Bayen, A.M.: Solutions to estimation problems for Hamilton-Jacobi equations using Linear Programming. In: Submitted to IEEE Transactions on Control Sytems Technology (2010)
12. Daganzo, C.F.: A variational formulation of kinematic waves: basic theory and complex boundary conditions. Transporation Research B 39B(2), 187–196 (2005)
13. Frankowska, H.: Lower semicontinuous solutions of Hamilton-Jacobi-Bellman equations. SIAM Journal of Control and Optimization 31(1), 257–272 (1993)
14. Gruteser, M., Grunwald, D.: Anonymous usage of location-based services through spatial and temporal cloaking. In: Proceedings of the 1st International Conference on Mobile Systems, Applications and Services, pp. 31–42. ACM (2003)
15. Hoh, B., Gruteser, M., Herring, R., Ban, J., Work, D., Herrera, J.C., Bayen, A.M., Annavaram, M., Jacobson, Q.: Virtual trip lines for distributed privacy-preserving traffic monitoring. In: MobiSys 2008, Breckenridge, CO (2008) (to appear)
16. Kitakami, H., Hara, H., Yamanaka, H., Miyazaki, T.: Performance evaluation for parallel mixed-integer linear programming system. Optimization Methods and Software 3(4), 257–272 (1994)
17. Krumm, J.: Inference attacks on location tracks. In: LaMarca, A., Langheinrich, M., Truong, K.N. (eds.) Pervasive 2007. LNCS, vol. 4480, pp. 127–143. Springer, Heidelberg (2007)
18. Krumm, J.: A survey of computational location privacy. Personal and Ubiquitous Computing 13(6), 391–399 (2009)
19. Le Ny, J., Pappas, G.: Privacy-preserving release of aggregate dynamic models. In: Proceedings of the 2nd ACM International Conference on High Confidence Networked Systems, pp. 49–56. ACM (2013)
20. Lighthill, M.J., Whitham, G.B.: 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)
21. Peddinti, S.T., Saxena, N., Birmingham, A.L.: On the limitations of query obfuscation techniques for location privacy. In: International Conference on Ubiquitous Computing (2011)
22. Work, D., Blandin, S., Tossavainen, O., Piccoli, B., Bayen, A.: A distributed highway velocity model for traffic state reconstruction. Applied Research Mathematics eXpress (ARMX) 1, 1–35 (2010)
23. `http://www.vissim.de/`
24. `http://traffic.berkeley.edu/`
25. `http://pems.eecs.berkeley.edu`