Urban Traffic State Estimation for Signal Control Using Mixed Data Sources and the Extended Kalman Filter

Simon Box (corresponding author)
Transportation Research Group
University of Southampton, UK
S.Box@soton.ac.uk

Ian Snell
Siemens plc
Sopers Lane
Poole, UK

Ben Waterson
Transportation Research Group
University of Southampton, UK

R. Eddie Wilson
Transportation Research Group
University of Southampton, UK

4253 words + 8 figures + 0 tables

March 25, 2014
ABSTRACT
This paper describes a methodology for fusing data from multiple sensors, including wireless devices and inductive loops, to make an estimation of the instantaneous state of an urban traffic network. An extended Kalman filter is employed along with a state evolution model to make estimates of the state in a discretized network. The instantaneous state is an estimate of the current distribution of vehicles in the network and their instantaneous speeds. Microsimulation tests were used to evaluate the performance of the state estimation on a small urban network. These results indicate low error between the estimated state and the known ground truth.
INTRODUCTION
The number of wireless devices in the transport network is growing rapidly. This includes smartphones carried by drivers and passengers, in-car blue-tooth systems (for example in the radio) and, increasingly in-car WiFi. The adoption of WiFi systems for vehicles by vehicle manufacturers is due to the demand for “infotainment”, that is the provision information and entertainment into vehicles via the internet (Macario et al. [1]). However the data that could be made available from such systems are also of significant benefit to ITS (Intelligent Transportation Systems) applications. In Europe three major studies have recently examined the benefits of vehicle to infrastructure (V2I) and vehicle to vehicle (V2V) WiFi communications (Kompfner, COOPERS, SAFESPOT [2, 3, 4]). Furthermore common European protocols have been set for this type of communication (IEEE 802.11p).

It is clear that the future trend is towards a large number of different types of wireless devices in the road traffic network. The data that may be available from these wireless devices carries valuable information that can be exploited by Urban Traffic Control (UTC) systems. The challenge becomes to combine these data sources and existing traffic data sources e.g. inductive loops (Sreedevi [5]), microwave detectors (Wood et al. [6]) and cameras (Qadri and Asif [7]) to estimate a single coherent image of the state of the network.

The potential value of wireless device data in UTC has been investigated by simulating prototype junction systems that employ simulated wireless data in their control algorithms (Waterson and Box, Box and Waterson, Box and Waterson [8, 9, 10]). This research indicates that signal controllers that employ wireless data can significantly outperform existing control systems (e.g. MOVA V. Vincent and Pierce [11]) both in terms of delay and equitability. In this case the variance of the distribution of journey times across the junction is taken as a measure of equitability. In particular Box and Waterson [9] and Box and Waterson [10] show that the fidelity of wireless data supports the use of machine learning signal control algorithms, which exhibit high performance. In particular the machine learning control algorithms require an instantaneous estimate of the network state giving the current positions and speeds of vehicles.

This paper presents a methodology for estimating a single coherent image of the state of the network that can support the machine learning algorithms for urban signalized junction control described in Box and Waterson, Box and Waterson [8, 9, 10]. The proposed methodology discretizes the road network into small areas at the lane level. Metrics defining the state of the network (e.g. average speed $\bar{V}$, number of vehicles $N$) are associated with each area and estimated from multiple information sources using an Extended Kalman Filter (EKF), which employs a state evolution model.

Simulation test results are presented that indicate the performance of the proposed methodology for different sensor mixes and on two different networks, a simple straight road (Figure 4) and a small urban network containing three signalised junctions (Figure 1).

STATE ESTIMATION BACKGROUND
Estimation of the traffic state from sensor data is a task that is central to all aspects of traffic control. It is employed in recording the state for future analysis, providing traffic information and journey time predictions, making strategic control decisions and also making signal control (traffic light) decisions.

As such a large amount of work has been done on traffic state estimation with many techniques being proposed, the vast majority of proposed techniques are been based either on some
variant of the Kalman Filter (Kalman and Bucy (12)) or similar Bayesian recursive filtering techniques. The attraction of these techniques is the ability to combine multiple sources of sensor data and to support the sensor data with traffic flow models to give both an estimation of the state and a measure of the confidence in this estimate.

Initial work in this area has focused on state estimation for highways. Wang and Papageorgiou (13) present a method for applying the Extended Kalman Filter (EKF) (employing a Macroscopic traffic flow model) for estimating average speeds and densities of highways. Tests of this approach using real world data indicated favourable performance. Wang et al. (14). Tampere and Immers (15) also applied the EKF, but employing a Cell Transition Model, to predict highway travel times. Hegyi et al. (16) compared the performance of a number of filters for highway traffic state estimation, including the Unscented Kalman Filter (UKF), the EKF and joint filters. This work also gave some indication of the numbers of sensors required for good performance.

In all the work reported above “sensors” means inductive loops, or other census type detectors counting vehicles. However the Kalman Filter framework is ideal for combining these sensor measurements with data from probe sensors on-board vehicles, which are likely to become commonplace in the near future (see Section 2). Nanthawichit et al. (17) presents a technique for combining conventional loop detector data with GPS data from on-board vehicle probes, within the Kalman Filter framework. Again this was for traffic state estimation on a highway.

Traffic state estimation in the urban environment presents a significantly greater challenge than the highway environment due to the frequency with which vehicles are changing from one road to another and the transience of vehicle speeds, with vehicles often getting caught in queues and waiting at red traffic lights. van Hinsbergen et al. (18) propose a method employing Bayesian Neural Networks for prediction of travel times between points in the urban environments. While travel times are useful for analysis of the networks performance, more instantaneous data on the positions and speeds of vehicles is required for signal control applications. Kong et al. (19) present a method based on the Kalman Filter and evidence theory to fuse data from GPS probes and inductive loop data for estimation of traffic states in the Urban environment. This technique requires careful placement of inductive loops to suit the evidence theory model.

The approach to traffic state estimation in the urban environment that is described in this paper is closest to the Wang and Papageorgiou (13) approach for highway state estimation in so far as it uses the Extended Kalman Filter (EKF) and a “state evolution model” which is similar in concept to the macroscopic flow models employed in Wang and Papageorgiou, Wang et al. (13, 14). The main difference is that the model is estimating instantaneous flows on comparatively small areas of the urban network, down to the lane level. It also handles the flow between areas taking into account both signalized and non-signalised junctions.

The aim of this state estimation approach is to estimate metrics such as instantaneous number of vehicles and average speed in small geographical areas, into which the urban road network is discretized. By design this approach produces data which are suitable for input into the “machine learning” traffic light junction control algorithms that are described in Box and Waterson, Box and Waterson (9, 10).

It is also a goal of this approach that the algorithm structure enables a parallel computing approach so that on large networks the state in local areas can be computed by individual CPU’s making this approach scalable. Finally it is also a goal that the treatment of sensor sources in this approach is general and can incorporate data from any new census (e.g. loops, bluetooth) or probe (e.g. GPS, WiFi, cellphone) data sources, without an ad-hoc redesign.
FIGURE 1 A four junction network with three signalized junctions (the corners of the triangle) state areas are numbered

METHODOLOGY
The sensors that provide data for UTC systems can be classified into two types: those which collect census data, that is vehicles are detected passing a specific point in space; and those which collect probe data, that is tracking the position and speed of individual vehicles.

Trying to combine multiple independent sources of census and probe data, which are measuring different things in different ways, can present some challenges. For example, not all of the data sources are available all of the time (latency); data from different sources may be contradictory; some vehicles may contain multiple wireless devices, some none (penetration).

The proposed methodology to meet these challenges is to apply an Extended Kalman Filter (EKF) (Zarchan and Musoff (20)) as described in this section.

Definition of State
Within the EKF framework we assume that no single source of information is providing the truth of the state on the road but is instead providing us with evidence of a state, which must be defined.
To define the state we discretize the network up into small areas, an example is shown in Figure 1. Each area has one or more metrics associated with it. In the example presented here we assume two metrics: mean vehicle speed, averaged across all vehicles in the area at time $t$ ($\bar{V}_t$) and number of vehicles in the area at time $t$ ($N_t$). The size of areas is something which can be set up in the design of the network state and tuned to provide the required level of complexity in information.

**State Evolution Model**

When dynamically assessing the state of the network it is possible to make reasonable predictions of how the state will evolve over the very short term, even in the absence of any information from sensors. This can be useful, especially during short periods of high sensor latency. An example of a simple state evolution model is presented below.

\[ Q_t = \begin{cases} 0 & \text{at a red light} \\ \frac{N_t \bar{V}_t}{l} & \text{otherwise} \end{cases} \]  \hspace{1cm} (1)

Where $l$ is the total length of all lanes in the area.

**FIGURE 2** State evolution model predict the flow of vehicles between neighbouring areas

Each area in the network is considered individually along with its upstream neighbour (Figure 2). The out-flow of an area at time $t$ ($Q_t$) is estimated from $\bar{V}_t$ and $N_t$ within the area using (1), except for the special case where end of the area corresponds with a junction stop line and the light is currently red, where $Q_t = 0$ (1).

\[ N_{A,t+1} = N_{A,t} + Q_{B,t} \delta t - Q_{A,t} \delta t \] \hspace{1cm} (2)

\[ \bar{V}_{A,t+1} = \bar{V}_{A,t} \] \hspace{1cm} (3)

Where $\delta t$ is the time step between $t$ and $t+1$. In the event that area A has more than one upstream neighbour (Figure 3), for example, at a junction, then the model is adjusted as in equation (4).

\[ N_{A,t+1} = N_{A,t} + Q_{B,t} \delta t + Q_{C,t} \delta t - Q_{A,t} \delta t \] \hspace{1cm} (4)

**FIGURE 3** State evolution model: multiple upstream neighbours are possible e.g. at junctions

The model estimates the state in area A at time $t + 1$ as
In the case where an upstream area feeds more than one downstream area the flow is multiplied by an appropriate flow split coefficient, which can be learned from historical data.

\[ N_{A,t+1} = N_{A,t} + Q_{B,t}S\delta t - Q_{A,t}\delta t \]  

(5)

Where \( S \) is the flow split coefficient.

**Prediction Step**

Considering a single area \( A \) we define the state

\[ X_t = \begin{bmatrix} N_{A,t} \\ \bar{V}_{A,t} \end{bmatrix} \]  

(6)

At time \( t + 1 \) the state evolution model is used to make a prediction of \( X_{t+1} \).

\[ X^{-}_{t+1} = f(X_t) \]  

(7)

Where the superscript \((-\)) indicates that this is the prediction. Larger regions containing multiple areas can also be dealt with using this technique; however by considering single areas like this the computational task can be parallelised and distributed meaning it can be deployed on networks of arbitrary size.

A covariance matrix describing the Gaussian uncertainty in \( X^{-}_{t+1} \) is given by

\[ P^{-}_{t+1} = FP_tF^T + U \]  

(8)

Where \( F \) is the matrix of first order partial derivatives (Jacobian) for the prediction of state function in (7). In this example \( F \) is given by (9) below. \( U \) is a covariance matrix for the uncertainty in the state evolution model. This can be estimated, for example using a micro-simulation model.

\[ F = \begin{bmatrix} \frac{\partial N^{-}_{A,t+1}}{\partial N_{A,t}} & \frac{\partial N^{-}_{A,t+1}}{\partial \bar{V}_{A,t}} \\ \frac{\partial \bar{V}_{A,t+1}}{\partial N_{A,t}} & \frac{\partial \bar{V}_{A,t+1}}{\partial \bar{V}_{A,t}} \end{bmatrix} = \begin{bmatrix} 1 - \frac{\bar{V}_{A,t}\delta t}{l} & \frac{N_{A,t}\delta t}{l} \\ 0 & \frac{l}{1} \end{bmatrix} \]  

(9)

**Sensor Model**

The goal of the sensor model is to estimate the sensor signals that will be received given the predicted state \( X^{-}_{t+1} \). The specific sensor model employed depends on how many sensors, which collect census data are in the area of interest and how many types of wireless probe sensors are currently in the network. In general for a census sensor \( C_1 \), The expected number of counts registered on the sensor for time interval \( \delta t \). Is modelled as

\[ N^{C_1} = \frac{N^{-}_{A,t+1}\bar{V}_{A,t+1}\delta t}{l} \]  

(10)

For a wireless probe sensor type \( W_1 \), The expected number of detections in area \( A \) is modelled as

\[ N^{W_1} = N^{-}_{A,t+1}\phi^{W_1} \]  

(11)

Where \( \phi^{W_1} \) is the penetration rate for \( W_1 \), which is the fraction of vehicles in the network that carry sensor type \( W_1 \). For some sensors, for example mobile phones, \( \phi^{W_1} \) could be greater than 1.
If the wireless probe sensor $W_1$ can report vehicle speed then the mean speed averaged across all $W_1$ sensors detected in area $A$ is modelled as

$$\bar{V}_{W_1} = \bar{V}_{A,t+1}$$

The same approach in (12) is used for census detectors that measure speed, for example inductive loop pairs.

**Update Step**

Take the example where area $A$ contains an inductive loop sensor $C_1$. And the system currently detects two types of wireless probe data: $W_1$, which provides speed data and $W_2$ which does not. The measurement vector $Z$ is given by

$$Z = [N_{C_1}, N_{W_1}, \bar{V}_{W_1}, N_{W_2}]$$

(13)

$y$ is the difference between the actual sensor measurements and the expected measurements from the sensor model ($h$) described above.

$$y = Z - h(X_{t+1})$$

(14)

$y$ is used to apply a correction to the predicted state and covariance

$$X_{t+1} = X_{t+1}^r + K y$$

(15)

$$P_{t+1} = (I - KH)P_{t+1}^r$$

(16)

Where $H$ is the Jacobian matrix for the sensor model $h$ and $K$ is the Kalman gain matrix calculated according to the EKF equations (Zarchan and Musoff (20)) using

$$K = P_{t+1}^rH^T(HP_{t+1}^rH^T + R)^{-1}$$

(17)

Where $R$ is a covariance matrix giving the Gaussian uncertainty in the measurement data, this can be estimated from the rated performance of the sensors.

**SIMULATION TESTS**

The above methodology was tested in Microsimulation using Sias Paramics software. Two network models were constructed, a simple model of a straight road as shown in Figure 4 and a model of a small urban network with three signalised junctions as shown in Figure 1. According to the methodology the networks in each model were discretized into areas, these areas are indicated on Figure 1 and Figure 4.

The precise positions and speeds of all vehicles in the model are known at any one time, thus there is a well defined ground truth for the network state. During simulation tests the EKF was run with a 10 second time step and the ground truth was recorded at 10 second intervals to allow an error between the estimated state and the ground truth to be calculated.

Three types of sensor were simulated in the tests: Inductive loop sensors (the locations of these are marked by an ‘L’ in Figure 1 and 4); WiFi probe sensors assuming a positioning accuracy of $1\sigma = 10\ m$ and mobile phone based probes assuming a positioning accuracy of $1\sigma = 20\ m$. Different levels of penetration for the two probe sensors were investigated as described below. Loop sensors were simulated using the in built functionality of Sias Paramics software. probe sensors were simulated by perturbing the actual positions of vehicles using Gaussian noise following the technique described in Waterson and Box (8).
Benchmark

A lower benchmark was required against which to measure the performance of the method described above. The benchmark used was a simple averaging over the sensor data collected for each area as described below. Assuming the area contained an inductive loop sensor $C_1$ and if we call the two probe sensor types $W_1$ and $W_2$ then the set of count measurements at each time step would be $[N^{C_1}, N^{W_1}, N^{W_2}]$. The average $\bar{N}$ would be calculated as

$$\bar{N} = \frac{N^{C_1} + N^{W_1} + N^{W_2}}{3}. \quad (18)$$

Similarly the average over the mean speed measurements would be

$$\bar{V} = \frac{\bar{V}^{C_1} + \bar{V}^{W_1} + \bar{V}^{W_2}}{3}. \quad (19)$$

Model 1: Simple straight road

![Road layout for the simple straight road model. The shading on the figure indicates the 10 state areas](image)

Figure 4. Shows a schematic of the Simple road model. Vehicles enter the model on the left and right and are spawned stochastically. Overall the average flow rate of vehicles into the model is 10 Vehicles per minute in each direction.

Tests were carried out where the instantaneous state was estimated using the simple averaging (benchmark) method and the EKF. In the first test the penetration rate of WiFi probes was 50% and the penetration rate of mobile phones was 20%. In the second test the penetration rates were respectively 80% and 50%.

The RMS error (averaged across all areas) between the estimated state and the ground truth was calculated at each time step of the test. The results of these calculations are shown in Figure 5 and Figure 6.

Figure 5 shows the RMS error in the estimated number of vehicles in each area. This shows that the largest error occurred when the state was estimated by the benchmark simple averaging technique. This error was much reduced when using the EKF, with the error in the count of vehicles in each area being around 1 or below.

In the second test where higher penetration rates were simulated it can be seen that the richer data source leads to an improvement in performance with the error reducing further.

Figure 6 shows the RMS error in the estimated average speed of vehicles in each area. This shows a significant improvement in performance between the state estimated by simple averaging and the state estimated using the EKF. Again a further performance improvement is seen with the higher penetration rate of probe sensors.
FIGURE 5 RMS error averaged over all areas for the number of vehicles in each area at each time step. Three methods are used: a simple average across sensor readings and the EKF with different levels of sensor penetration.

Model 2: Small urban network
The straight road model is a suitable initial test but if this approach is going to be useful for signal control it has to work on an urban network with signalized junctions like the one in Figure 1. Additional tests were carried out on this model assuming penetration rates of 50% and 20% respectively for WiFi and mobile phone sensors. Also these tests were much longer, running for 40 minutes.

Model 1 contained no junctions, but model two contains three signalized junctions. The state evolution model described in Section 4.2 can model stopping at red lights using equation (1). In the first test carried out on Model 2 this stopping functionality was not employed, in the second test stopping at red lights was modelled in the state evolution model.

Figure 7 shows the RMS error in the number of vehicles estimated in each area for the duration of the two tests. This shows the error in the number of vehicles starting off low and initially increasing. This is because the simulation starts with no vehicles on the road and it takes a minute or two to fill up. The same effect was observed in the previous tests in Figure 5. The error then settles on an average of about 1 vehicle per area (also as seen in Figure 5) but with this longer test the error reduces over time as the covariance matrices in the EKF become tuned. Eventually the error settles on an average value of around 0.6 vehicles per area. The difference between the two plots highlights that there is only a small advantage modelling the stopping of vehicles in the state evolution model. Averaged over the whole test the count error is 0.03 vehicles lower when vehicle stopping is modelled.

Figure 8 shows the RMS error in the estimated average speed of vehicles in each area. Here we can see that the error in speed is larger than what was observed in the straight road test (Model 1). This is because the speed of the vehicles in the Model 1 tests was reasonably constant allowing the state evolution model to capture what the vehicles are doing easily. In the Model 2 tests the speed
is much more variable with the vehicles stopping at red lights. Therefore we can conclude that improvements to the state evolution model could lead to gains in performance in the estimation of speed here. Again a small improvement was seen by modelling the stopping of vehicles at junctions. Averaged over the whole test the speed error is $0.2 \, \text{m/s}$ lower when vehicle stopping is modelled.

CONCLUSIONS
This paper has presented a methodology for estimating the state on an urban road network, that is specifically designed for UTC applications. The methodology can employ existing UTC sensors, for example inductive loops, microwave sensors and cameras. However it can also employ multiple types of wireless device sensors. By design the Extended Kalman Filter approach proposed is flexible to varying rates of penetration and latency in these sensors.

Results from simulation experiments have demonstrated that the EKF method produces better estimates than a basic averaging across sensors and that feeding the EKF with better sensor data can improve performance still further. Tests on an urban network containing junctions indicated that there was no degradation in the performance of the method when it came to estimating the number of vehicles. However performance was reduced on the urban model when estimating the average speed of vehicles. This leads to the conclusion that a more sophisticated state evolution model could yield performance improvements in average speed estimation.
FIGURE 7 RMS error averaged over all areas for the number of vehicles in each area at each time step. For a long (40 minute) test on the triangular junction model.

FIGURE 8 RMS error averaged over all areas for the mean speed of vehicles in each area at each time step. For a long (40 minute) test on the triangular junction model.
REFERENCES


