AVOIDING EXPLICIT MAP-MATCHING IN VEHICLE LOCATION

Peter Lamb
Sylvie Thiébaux

CSIRO Mathematical and Information Sciences
GPO Box 664, Canberra ACT 2601, Australia
Phone: +61 2 6216 7000   Fax: +61 2 6216 7111
E-mail: Peter.Lamb@cmis.csiro.au, Sylvie.Thiebaux@cmis.csiro.au

SUMMARY

In this paper we combine Kalman filters and Markov models to solve the problem of locating a vehicle travelling on a road network. The location system incorporates information from a noisy vehicle positioning system, a simple model of vehicle dynamics and driver behaviour, and a representation of the road network. The Markov model is used to handle the topological aspects of the problem, maintaining a set of hypotheses for the segment on which the vehicle is travelling and their respective probabilities. The Kalman filters handle the metric aspects, providing estimates of the vehicle location on each of the hypothesized road segments. The two are closely coupled, with the statistics from the Kalman filters used to update the Markov belief state at each time step, and the Markov model providing a probability distribution over the Kalman filters. This differs from conventional vehicle location schemes by not performing any explicit map-matching step, and has advantages in both robustness and flexibility.

INTRODUCTION

Determining the accurate location of a vehicle relative to the physical environment (eg. roadway, intersections, services) is an important part of a range of ITS services, for example in-vehicle navigation, fleet management and infrastructure maintenance.
The first step in vehicle location is determining the vehicle's position in some convenient coordinate system. Unfortunately, most of the available positioning technologies have limitations either in accuracy of the absolute position (eg. GPS with Selective Availability), accumulated error (eg. dead-reckoning systems such as odometry), or availability (eg. GPS with or without differential correction).
To overcome these limitations, vehicle location systems typically combine vehicle positioning information (possibly from both an absolute positioning system like GPS and from a dead-reckoning system) with information from a digital map, making use of the fact that the vehicle is travelling on the road network. This map-matching step takes the position in two dimensions and any information that is available regarding the error distribution of the position to determine the most probable location of the vehicle on the road network (see, for example, Zhao (1)).
This two-step approach has a number of drawbacks:
a) it may not make sufficient use of the history of the vehicle's trajectory, and permit sudden switching of mapped location between unconnected road segments (eg. with simple projection of the location onto road segments used in the map-matching),

b) it may be too dependent on the history of the vehicle's trajectory (eg. map-matching with a dead-reckoning positioning system which at some point incorrectly assigns a vehicle to an incorrect link may have difficulty recovering from the error), and

c) it is difficult to incorporate other non-position information into the algorithm (eg. it is known that the vehicle is in an airport car park, but it is not known which one).

We propose here a location procedure where explicit map-matching is avoided and the incorporation of the map information is distributed between Kalman filters (eg. Bar-Shalom and Fortmann (2)) used to estimate the vehicle's trajectory on the roadway, and a Markov model of the vehicle's history of progression through the road network. Markov modelling for locating a vehicle on a grid is used for location systems in robotics (3, 4, 5).

The Markov model is used to account for the progression of the vehicle from road segment to road segment, and permits a number of concurrent hypotheses to be maintained about the vehicle's current road segment. The Kalman filter(s) model the distance that the vehicle has travelled on the hypothesised road segment(s), using inputs from either (or both) absolute positioning systems or dead-reckoning systems, and statistical information about the vehicle dynamics.

The Kalman filters can provide information to the Markov model regarding the likelihood of the trajectory in a filter relative to filters modelling the vehicle's trajectory on other hypothesised road segments, and the position on the road segment estimated in the Kalman filter can be used by the Markov model to determine the probability that the vehicle has reached an intersection and moved onto another road segment. Moving through an intersection necessitates the initiation of new Kalman filters for the connected road segments.
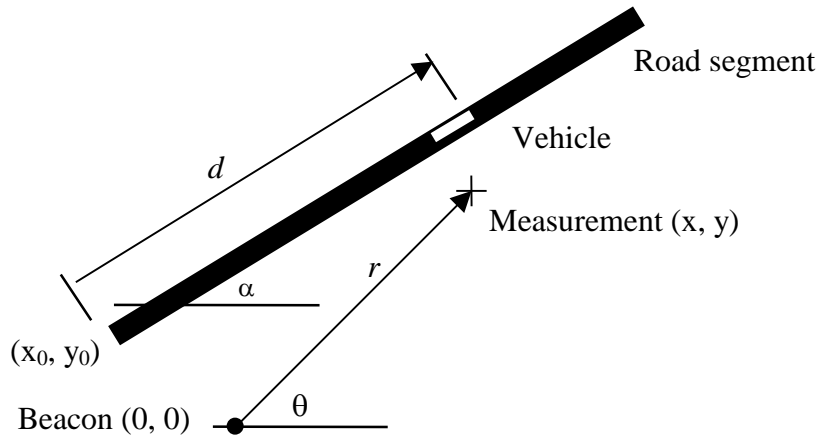

Figure 1: Vehicle on a road segment

In this paper we will examine the application of this technique to vehicle location using a roadside beacon location system. The next section examines the use of a Kalman filter in a single spatial dimension to estimate the position of the vehicle on a road segment, and following that, the use of a Markov model to maintain the set of hypothesised road segments that the vehicle may be travelling on. Some experimental results on simulated and field data are then examined and finally we present our conclusions on the technique and future work.

1-DIMENSIONAL KALMAN FILTER

In its simplest form, a Kalman filter provides an optimal estimate of the state $\mathbf{x}(t)$ for a discrete linear system:

$$\mathbf{x}(t+1) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) + \mathbf{v}(t) \qquad (1)$$

Where $\mathbf{A}(t)$ is the state transition matrix, $\mathbf{u}(t)$ are the control inputs to the system, and $\mathbf{B}(t)$ is the input gain matrix, and there is Gaussian noise $\mathbf{v}(t)$ introduced into the system. The Kalman filter is provided with measurements $\mathbf{z}(t)$:

$$\mathbf{z}(t) = \mathbf{C}(t)\mathbf{x}(t) + \mathbf{w}(t) \qquad (2)$$

Where $\mathbf{C}(t)$ is the measurement matrix, and $\mathbf{w}(t)$ is the Gaussian measurement noise. In our case we model one spatial dimension, the distance from a reference point on the road segment, d, and the state of the system is a third-order model: $\mathbf{x} = \left[d, \dot{d}, \ddot{d}\right]^{\mathrm{T}}$.

In this application, the driver's input to the vehicle is not known, so $\mathbf{u}(t)$ is taken to be zero, and the driver's behaviour is modelled by the process noise $\mathbf{v}(t) = \left[0, 0, \dddot{d}\Delta t\right]^{\mathrm{T}}$, where $\dddot{d} \sim N(0, \sigma_j)$ and Δt is the time interval for the discrete time system. A suitable value for σ_j can be taken by experiment, or from published data for representative vehicle behaviour. For our work we derived σ_j from a city drive cycle designed for a vehicle fuel consumption standard (6).

The system measurement model is linear, so the measurements from the beacon system (r, θ) are reduced to (x, y), and translated to a coordinate system with its origin at the reference point for the road segment (x_0, y_0). This results in the measurement matrix:

$$\mathbf{C}(t) = \begin{bmatrix} \cos\alpha & 0 & 0 \\ \sin\alpha & 0 & 0 \end{bmatrix}$$

The system transition matrix simply reflects the integration of the state over the time step:

$$\mathbf{A}(t) = \begin{bmatrix} 1 & \Delta t & \dfrac{\Delta t^2}{2} \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix}$$

Finally, for the operation of the Kalman filter, the covariance matrices $\mathbf{R}(t)$ and $\mathbf{Q}(t)$ for $\mathbf{v}(t)$ and $\mathbf{w}(t)$ respectively are required:

$$\mathbf{R}(t) = \begin{bmatrix} r^2\sigma_\theta^2 \sin^2\theta + \sigma_r^2\cos^2\theta & \cos\theta\sin\theta(\sigma_r^2 - r^2\sigma_\theta^2) \\ \cos\theta\sin\theta(\sigma_r^2 - r^2\sigma_\theta^2) & r^2\sigma_\theta^2\cos^2\theta + \sigma_r^2\sin^2\theta \end{bmatrix} \quad \mathbf{Q}(t) = \sigma_j^2 \begin{bmatrix} \dfrac{\Delta t^6}{36} & \dfrac{\Delta t^5}{12} & \dfrac{\Delta t^4}{6} \\ \dfrac{\Delta t^5}{12} & \dfrac{\Delta t^4}{4} & \dfrac{\Delta t^3}{2} \\ \dfrac{\Delta t^4}{6} & \dfrac{\Delta t^3}{2} & \Delta t^2 \end{bmatrix}$$

The formulation for $\mathbf{R}(t)$ assumes that σ_θ is small, so that $\sin\theta \approx \theta$.

At each step of the Kalman filter, the filter first uses the state equation and the filter posterior state estimate $\hat{\mathbf{x}}(t)$ to calculate a prior estimate of the state $\hat{\mathbf{x}}^-(t+1)$. After the measurement $\mathbf{z}(t+1)$ is made, the filter innovation, $\mathbf{r}(t+1) = \mathbf{z}(t+1) - \mathbf{C}(t+1)\hat{\mathbf{x}}^-(t+1)$ is calculated. This represents the difference between the prior prediction of the

measurement and the actual measurement made at time *t* and is related to the new information made available to the filter by the measurement. The innovation is used by the filter to correct the prior prediction. The innovation is a zero-mean Gaussian variable, with the useful property that for an observation history $Z(t) = \mathbf{z}(1)\ldots\mathbf{z}(t)$, the probability density of an observation at time $t+1$ is equal to the probability density of the of the innovation, $p(\mathbf{z}(t+1)|Z(t)) = p(\mathbf{r}(t+1))$, and the covariance matrix $\mathbf{S}(t)$ for $\mathbf{r}(t)$ is readily estimated in the filter (see, for example, (3)).

This probability relationship is required for the update of the belief state in the Markov model discussed in the next section.

MARKOV MODELS FOR MAP-MATCHING

The 1-dimensional Kalman filter will track a vehicle on a straight segment of roadway of indefinite length. The process of introducing new segments to the set of segments being modelled, and eliminating segments which are no longer reasonable models of the vehicle behaviour is handled here by a Markov model. The belief state of the Markov model consists of a set of Kalman filters for the current set of hypothesized road segments, and a probability distribution over the filters. Curved roads may be accommodated as piece-wise linear approximations and handled as a special case of the operations for intersections to be discussed here.

The Markov model maintains multiple hypotheses as to the most likely segments for the vehicle, modifies the set of most likely segments, and initiates 1-D Kalman filters to track the vehicle based on these hypotheses. Thus, the maintenance of the segment and filter set are modelled as state changes in a partially observable Markov process. For a network with *k* segments, at time step *t*, there can be at most k^t Kalman filters in the belief state of the Markov model, each one representing a path (possibly not physically feasible) through the network. We denote the probability distribution over the filter set at time *t* in the Markov model as $\Pr_t$ and the probability of a particular filter *f* in the belief state as $\Pr_t(f)$.
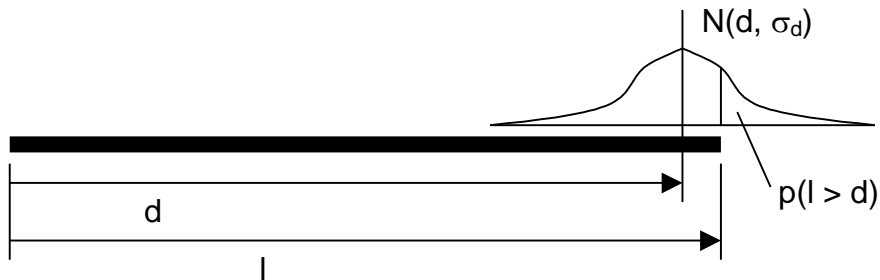

Figure 2: Probability that a vehicle has passed the end of a link

At time *t*, the vehicle being tracked by *f* may either continue along the segment ϕ that it is currently traversing, or it may turn onto a new segment. If segment ϕ has length *l* and the position estimate for the vehicle is *d* (the first component of $\hat{\mathbf{x}}$ for the filter), then the probability that the vehicle remains on ϕ is $p(0 < d < l)$, and the probability that the vehicle has turned onto another segment (Figure 2) at one extremity is $p(d > l)/n_1$ (for each of the n_1 connected segments), and at the other extremity, $p(d < 0)/n_2$ (for each of

its n_2 connected segments). Since d is Gaussian, and its standard deviation is estimated by the Kalman filter, these probabilities can be readily calculated. Indeed, the covariance matrix for $\hat{\mathbf{x}}$, $\mathbf{P}$ is estimated in the filter process, and $\mathbf{P}_{00} = \sigma_d^2$, so estimates of the parameters for the density function for d, $N(d, \sigma_d)$ are available from the filter.

Prior knowledge of the probability of making turns can be readily incorporated into these transition probabilities. The possibility of the transition onto a new link, or of continuing on the same link, constitute events.

Following the occurrence of an event, the Kalman filter set may need to be adjusted. For an event representing the vehicle continuing on ϕ, no action is required. For the event of a vehicle turning onto a new segment ρ, a new filter g with measurement equation

$\mathbf{z}(t) = \mathbf{C}_\rho(t)\, \mathbf{x}_g(t) + \mathbf{w}(t)$

must be initiated, with its state vector $\mathbf{x}_g(t)$ initialised by transforming $\mathbf{x}_f(t)$ into ρ's coordinate system.

The probability of an event from filter f to filter f', in the time step from t to $t+1$ is denoted by $T_t(f, f')$. In the case of the event representing the vehicle continuing on the same segment, f and f' represent the same filter.

For example, the probability of an event from filter f on ϕ to filter g on ρ, $T_t(f, g)$ (assuming wlog. that segment ρ is at the $d > l$ extremity of ϕ) is $p(d > l) / n_1$. Similarly, the probability of the event of the vehicle remaining modelled by filter f, $T_t(f, f)$ is $p(0 < d < l)$.

As was seen in the previous section, the probability of the observation history in f' at $t+1$ is simply the innovation at $t+1$, $p(\mathbf{r}_f(t+1))$. The belief state of the Markov model can then be updated using Bayes's rule (K is a normalising factor):

$$\mathrm{Pr}_{t+1}(f') = K \times p(\mathbf{r}_f(t+1)) \sum_{f \in \mathrm{Pr}_t} T_t(f, f')\, \mathrm{Pr}_t(f)$$

As was noted earlier, the number of filters in the full belief state grows exponentially with time and this is true even if only the segments adjacent to a filter's current segment are examined for transfer events. Hence, a process of elimination of filters from the belief state is required to keep the computational cost bounded. Almost all of the filters in the full belief state have extremely low probability. To keep computation manageable, two parameters are used to control the growth of the belief state set.

First, there is a threshold on $T_t(f, f')$, so that if $T_t(f, f') < p_{thresh}$, the transfer event is ignored. Second, only the N most probable filters are retained at each time step. The robustness and accuracy of location depends on these two parameters, but fortunately, we have observed that the algorithm itself operates satisfactorily over a wide range of settings for the parameters.

The location which is returned by the Markov model is the location contained in the most probable filter in the belief set. The more usual approach of using the probability-weighted location across all filters in the Markov model is not used, because that weighted location will typically not lie on the road network.

EXPERIMENTS

We have tested this approach to vehicle location on both simulated and field data. An example of the location algorithm operating on field data from an experimental beacon-based vehicle location system can be seen in Figure 3. The beacon is located at (0,0) at the intersection of Bunda and Ainslie.
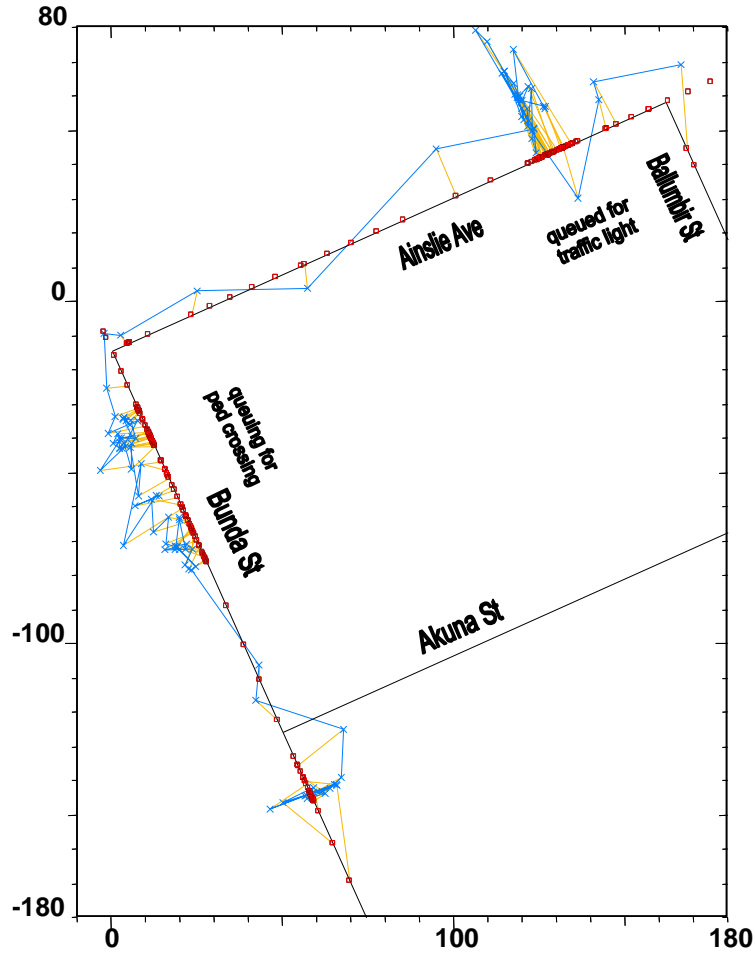
Figure 3: Field trial

Successive measured locations are denoted by crosses, and joined by blue lines, and they are connected to the corresponding location estimate (denoted by a red square) by an orange line. Location estimates without a corresponding measurement are interpolated by the Kalman filter(s). The error parameters for the data are: $\sigma_r = 5.9$m, $\sigma_\theta = 11.6°$.

The robustness of the location can be seen near the corner of Bunda and Akuna, where a data point lies much closer to Akuna, though the vehicle never enters Akuna, and where the filter correctly predicts the path of the vehicle. At the corner of Bunda and Akuna, the filter initially follows an incorrect path continuing along Akuna, because there is a data point that lies on that path. However, at the first data point which provides clear evidence that the vehicle has turned, the vehicle is tracked correctly down Ainslie Avenue, even though there are a large number of missed measurements along that road segment.

As the vehicle approaches the corner of Ainslie and Ballumbir, there are a sequence of missing measurements as the beacon's range limits are stretched and there is interference to the path between the beacon and the vehicle being obscured. At each time step with a missing measurement, the $\mathbf{P}_{00} = \sigma_d^2$ measure increases, reflecting the ever-increasing uncertainty of the vehicle's location. This keeps the probability that the vehicle has switched from Ainslie to Ballumbir below that of the vehicle maintaining its

track along Ainslie, and with only one filter in the Markov belief set, the event is ignored until a measurement takes place which forces the transition. During this time the respective probabilities of the two choices lie in the range of around 0.3–0.7, the locations would be better described as undetermined between the two choices. Figure 3 was produced with $p_{thresh} = 0.01$, and with only 1 filter being retained in the belief state. The quality of the location improves slightly if the size of the retained belief state in the Markov model is increased to 5 filters.
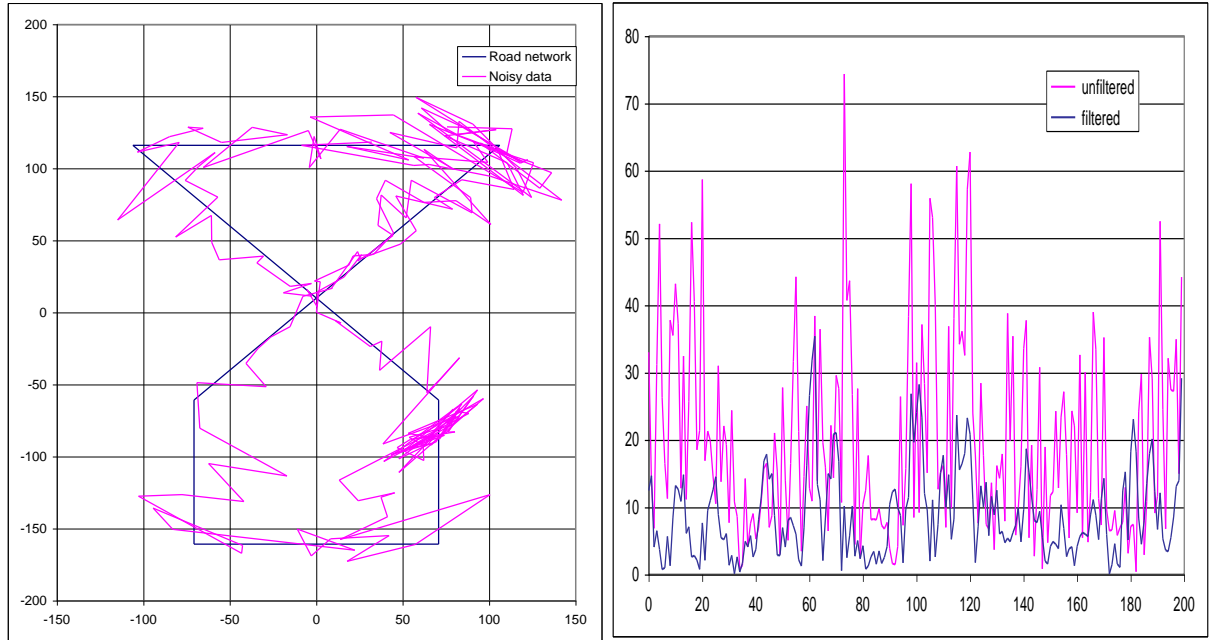

Figure 4: Synthetic network; raw data and error comparison

The synthetic network and data in Figure 4 permit an accurate comparison of the vehicle's estimated position with the known vehicle location. The vehicle moves around the synthetic network with its speed controlled by the same drive cycle data used to estimate the Kalman filter's driver behaviour parameter σ_j (6). The vehicle makes a random turn choice at the intersection in the middle of the network. The location beacon is near this intersection at (0,0). The error in the location system is modelled with the same parameters as for the field test above, i.e. $\sigma_r = 5.9m$, $\sigma_\theta = 11.6°$. This network is more demanding of the location system, especially in treating the acute-angled corners at the top of the network, where there are also large errors in the data, due to the relatively long range from the beacon. For this reason, a minimum of 10 filters must be retained in the Markov model's belief state, and the event probability threshold needed to be set at a very low value to ensure that events were generated for locations "around the corner", especially at the two acute angled corners. The filtering improved the mean location error from 20.0m to 9.0m, and the RMS error from 24.7m to 11.2m. The error comparison for the raw (violet) and filtered (blue) data is shown in Figure 4 along with the raw location data for the run. Occasionally an incorrect choice is made by the Markov model as a corner is approached, but the choice is corrected within a few (2-3) seconds of the turn being made.

7

CONCLUSIONS AND FUTURE WORK

In this approach, the problem is divided into two parts, the problem of estimating the vehicle's position on a roadway segment given a hypothesized segment; and the maintenance of a set of reasonable hypothesized segments. The metric component of the location problem is handled by Kalman filters, which provide a least-squares optimal estimate of the vehicle's position on a road segment, given a model of the roadway and a rudimentary model of the driver's behaviour. The Markov process handles the topological aspects of the location problem by providing a mechanism for expanding the hypothesized set of road segments, assigning them probabilities and for removing the least probable segments. In this way, the complexity of the Markov process and the complexity of the filter processing is kept bounded, and the system delivers an optimal estimate of the vehicle's position (and speed and acceleration) as long as the Markov process selects the correct segments. The Markov model's computational cost *vs.* robustness and accuracy can be traded off explicitly by the size of the retained belief state and p_{thresh}, but its performance does not appear to be very sensitive to small changes in these settings.

This work has two other major influences beyond the literature already cited on Markov models in robot location.

Ackerson and Fu (7) consider the use of Markov models in the estimation of state where the characteristics of the noise in the system changes with time, while in our case the main problem is the detection of changes in the measurement matrix.

The idea of the use of a 1-D Kalman filter for vehicle position estimation is also not new. Scott and Drane (8) use the Kalman filter to discriminate between the location hypotheses generated by map-matching, using $p(\mathbf{r}(t+1))$ as we have here. Where more than one hypothesis is plausible (where the error is large, or near intersections), then the Kalman filter is split to carry forward multiple hypotheses. An important difference between this approach and ours is the lack of explicit map-matching in our approach, and the fact that our approach is simpler and conceptually more robust, and can be parameterised easily.

Our plans for future work include experimentation with combined beacon/GPS measurements, and the use of multiple-model Kalman filters (adding constant-position and constant-velocity filters to the constant-acceleration filter used here). This approach can assist in identifying the behaviour of the vehicle, particularly in deciding whether it has stopped. In the longer term we are interested in application of the approach to in-vehicle navigation, and to priority signalling for traffic control.

REFERENCES

1. Y. Zhao, *Vehicle Location and Navigation Systems*. Artech House, 1997.
2. Y. Bar-Shalom and T. E. Fortman, *Tracking and Data Association*, Academic Press, 1988.
3. A. R. Cassandra, L. Kaelbling and J. A. Kurien, "Acting under Uncertainty: Discrete Bayesian models for mobile-robot navigation", in *Proc. IEEE/RJS Conference on Intelligent Robots and Systems*, 1996.
4. D. Fox, W. Burgard, S. Thrun and A. B. Cremers, "Position estimation for mobile robots in dynamic environments", in *Proc. AAAI-98*, pp 983-988, 1995.
5. R. Simmons and S. Koenig, "Probabilistic robot navigation in partially observable environments", in *Proc. IJCAI-95*, pp 1080-1087, 1995.

6. Standards Association of Australia, *Methods of test for fuel consumption of motor vehicles designed to comply with Design Rules 37 and 40*, Publication AS 2877-1986, 1986.

7. G. Ackerson and K. S. Fu, "On state estimation in switching environments", *IEEE Trans. On Automatic Control*, 15(1), 1970.

8. C. A. Scott and C. R. Drane, "Characterising empirical information sources concerning motor vehicles", in *Proc. 2nd World Congress on Intelligent Transportation Systems*, pp 573-579, 1995.