We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS BOOK CITATION INDEX INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

1

Kalman Filter Applications for Traffic Management

Constantinos Antoniou[1], Moshe Ben-Akiva[2] and Haris N. Koutsopoulos[3]
[1]*National Technical University of Athens,*
[2]*Massachusetts Institute of Technology,*
[3]*Royal Institute of Technology,*
[1]*Greece*
[2]*U.S.A.*
[3]*Sweden*

1. Introduction

Traffic congestion is a major problem in urban areas that has a significant adverse economic impact through deterioration of mobility, safety and air quality. As a result, the importance of better management of the road network to efficiently utilize existing capacity is increasing. To that end, many urban areas build and operate modern Traffic Management Centers (TMCs), which perform several functions, including collection and warehousing of real-time traffic data, as well as utilization of this data for various dynamic traffic control and route guidance applications. In order to be effective, these applications —which include Advanced Traveler Information Systems (ATIS) and Advanced Traffic Management Systems (ATMS)— require traffic models that provide, in real-time, estimation and prediction of traffic conditions.

The complexity of transportation systems often dictates the use of detailed simulation-based Dynamic Traffic Assignment (DTA) models (Ben-Akiva et al., 1991, 2002, Mahmassani, 2001) for this purpose. Dynamic Traffic Assignment (DTA) systems support both planning and real–time applications. Planning applications may include the off–line evaluation of incident management strategies, the evaluation of alternative traffic signal and ramp meter operation strategies and the generation of evacuation and rescue plans for emergencies (e.g. natural disasters) that could affect the traffic network. Real–time applications make use of the traffic prediction capabilities of DTA systems and may include on–line evaluation of guidance and control strategies, real–time incident management and control, support of real–time emergency response efforts and optimization of the operation of TMCs through the provision of real–time predictions.

Real–time DTA systems typically comprise two main functions: traffic state estimation, and traffic prediction (Ben-Akiva et al., 2002). An overview of the state-of-the-art Dynamic Traffic Assignment framework is shown in Fig. 1. DTA functionality is supported by two main modules: a demand simulator and a supply simulator. The demand simulator fuses surveillance information with historical information for the estimation and prediction of the evolving demand patterns. This is achieved through a combination of aggregate predictive

Source: Kalman Filter, Book edited by: Vedran Kordić,
ISBN 978-953-307-094-0, pp. 390, May 2010, INTECH, Croatia, downloaded from SCIYO.COM

models and disaggregate behavioral models (Antoniou et al., 1997). The supply simulator is usually based on high–level (mesoscopic or macroscopic) models that represent traffic dynamics using speed-density relationships, kinematic representation of traffic elements of queueing theory, etc. A detailed treatment of the demand-supply interactions within a state-of-the-art DTA system can be found in Ben-Akiva et al. (2002).

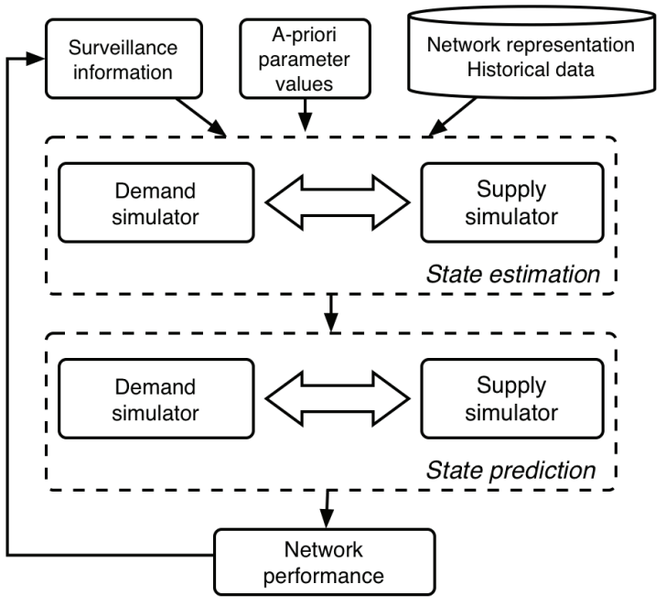


Fig. 1. Dynamic traffic assignment framework overview

In the current DTA framework, only the OD flows are calibrated on-line: one of the key components of dynamic traffic assignment is the Origin-Destination (OD) estimation and prediction process (Ashok & Ben-Akiva, 1993, Ashok, 1996, Ashok & Ben-Akiva, 2000, 2002). OD estimation combines historical and real-time information to obtain dynamic demand matrices. However, a number of other parameters are used by models in the demand simulator and the supply simulator. On the supply side parameters include speed–density relationship parameters and output capacities of network links and intersections. On the demand side, additional parameters (besides the OD flows) include behavioral model parameters.

In most cases, the approach to the problem of calibration of these parameters has been to perform off–line calibration of the simulation models using a database of historic information. The calibrated parameter values are then used in the on–line simulations. The calibrated model parameters therefore represent average conditions over the period represented in the data. Models that were calibrated this way may produce satisfactory results in off–line evaluation studies, which are concerned with the expected performance of various traffic management strategies.

However, this may not be the case in real-time applications, which are concerned with the system performance on the given day. If the model calibrated off-line is used without adjustment, the system is not sensitive to the variability of the traffic conditions between days, which are the result of variations in the parameters of the system, such as weather and surface conditions. Such variations may cause traffic conditions to differ significantly from the average values. Thus, the predictive power of the simulation model may be significantly reduced.

Speed–density relationships may depend on location–specific parameters, such as type of facility, number of lanes, lane width, slope, surroundings, as well as exhibit temporal variations, i.e. they may vary by season, day of the week, or even time of day, reflecting different driving behaviors (e.g. experienced drivers during commute periods). Off-line calibration could, in principle, deal with these situations, through the generation of a historical database of different speed-density relationships, categorized by the conditions. Based on the prevailing conditions, the "appropriate" relationship could then be retrieved and used. However, traffic dynamics also depend on factors that cannot always be anticipated or observed, such as weather conditions, incidents, unscheduled maintenance work, traffic mix. Even when these factors can be predicted, it would be impractical to calibrate traffic dynamics models for each possible scenario. Minor incidents (such as a car slowing down in the break–down lane) that are not reported or captured otherwise in the system may also impact the traffic dynamics.

The output capacity of the network links and intersections is another example. Average values could in general be obtained during an off-line calibration phase. However, capacities are affected by several phenomena (including weather and lighting conditions, traffic composition, etc.) and may therefore change as prevailing conditions change.

To overcome this problem, real-time data can be used to re-calibrate and adjust the model parameters on-line, so that prevailing traffic conditions can be captured more accurately. The wealth of information included in the off-line values can be incorporated into this process by using them as a priori estimates.

The remainder of this chapter is organized as follows. Section 2 presents a review of relevant literature, Section 3 presents a formulation of the on–line calibration problem as a state–space model, and Section 4 presents applicable solution approaches. Section 5 presents an application of the methodology to a network in Southampton, U.K., and Section 6 concludes the chapter with a summary and directions for further research.

2. Literature review

The topic of on-line calibration of traffic simulation models has received only limited attention in the literature. This section presents a review of prior on–line calibration research. System–level approaches are presented first, followed by research focused on individual components.

2.1 System–level approaches

Doan et al. (1999) outline a framework for periodic adjustments to a traffic management simulation model in order to maintain an internal representation of the traffic network consistent with that of the actual network. The authors categorize the error sources as demand estimation, path estimation, traffic propagation, internal traffic model structure, and on–line data observation and propose a system of on–line and off–line adjustment modules. A similar approach is proposed in Peeta and Bulusu (1999), where consistency is sought in terms of minimizing the deviations of the predicted time–dependent path flows from the corresponding actual flows. He et al. (1999) develop a combined off–line and on–line calibration process to adjust the analytical dynamic traffic model's output to be consistent with real–world traffic conditions by periodically detecting inconsistencies between model outputs andreal–world data, and actuating the correction model to correct the errors.

2.2 Supply parameters

van Arem and van der Vlist (1992) developed an on–line procedure for the estimation of current capacity at a motorway cross–section. The procedure is based on the combination of an on–line estimation of a "current" fundamental diagram with a maximum occupancy that may be achieved under free–flow conditions. The capacity is estimated by substituting the current maximum occupancy into the current fundamental diagram.

Tavana and Mahmassani (2000) use transfer function methods (bivariate time series models) to estimate dynamic speed–density relations from typical detector data. The parameters are estimated using the past history of speed–density data; no predetermined parameters or shape for the model are assumed. The method is based on time series analysis, using density as a leading indicator. Hyunh et al. (2000) extend the work of Tavana and Mahmassani (2000) by incorporating the transfer function model into a DTA simulation–based framework. Furthermore, the estimation of speeds using the transfer function model is implemented as an adaptive process, where the model parameters are updated on–line based on the prevailing traffic conditions. Qin and Mahmassani (2004) evaluate the same model with actual sensor data from several links of the Irvine, CA, network. In this chapter, determination of system input and output is derived from the higher–order continuum model. From the numerical results, the performance and the robustness of the transfer function model is in general found to be superior to the static model.

Van Lint et al. (2002) develop a state-space formulation of the travel time prediction problem and use it to derive a recurrent state-space neural network (SSNN) topology that captures the highly non-linear characteristics of the freeway travel time prediction problem. Van Lint et al. (2005) extend the model with preprocessing strategies based on imputation in order to achieve accuracy and robustness with respect to missing or corrupt data. Liu et al. (2006) present two distinct ways of using Extended Kalman Filters to address the problem of short-term urban arterial travel time prediction. Van Lint (2006) proposes a delayed EKF method for the online incremental training of a data driven travel time prediction model (a state-space neural network) for the prediction of travel times.

Antoniou et al. (2005) formulate the problem of on-line calibration of the speed-density relationship as a flexible state-space model and present applicable solution approaches. Three of the solution approaches [Extended Kalman Filter (EKF), Iterated EKF, and Unscented Kalman Filter (UKF)] are implemented and an application of the methodology with freeway sensor data from two networks in Europe and the U.S. is presented. The EKF provides the most straightforward solution to this problem, and indeed achieves considerable improvements in estimation and prediction accuracy. The benefits obtained from the —more computationally expensive— Iterated EKF algorithm are shown. An innovative solution technique (the UKF) is also presented.

Wang and Papageorgiou (2005) present a general approach to the real-time estimation of the complete traffic state in freeway stretches. They use a stochastic macroscopic traffic flow model, and formulate it as a state-space model, which they solve using an Extended Kalman Filter. The formulation allows dynamic tracking of time-varying model parameters by including them as state variables to be estimated. A random walk is used as the transition equations for the model parameters. A detailed case study of this methodology is presented in Wang et al. (2007).

Boel and Mihaylova (2006) present a stochastic model of freeway traffic suitable for on-line estimation. The model is estimated using a recursive filter based on Monte Carlo techniques (called also particle filters). Ben Aissa et al. (2006) use sequential Monte-Carlo or particle filter methods for the estimation and prediction of travel time.

2.3 Behavioral parameters

Peeta and Yu (2006) propose a behavior–based consistency–seeking modeling approach to bridge the functional gaps between route choice models and dynamic traffic assignment models vis–a–vis predicting the time-dependent network traffic flow patterns. The approach consistently addresses day–to–day learning and within–day dynamics using a single hybrid probabilistic–possibilistic behavioral model (Peeta & Yu, 2004, 2005) through intuitive if–then rules that are based on the findings of past studies in the literature. The approach avoids rigid assumptions on driver behavioral tendencies and a priori knowledge of driver behavior class fractions, and enables the classification of information characteristics and the consistent modeling of information effects. The proposed approach uses currently available data and achieves computational tractability by obviating a search procedure to predict the dynamically evolving traffic flow pattern.

2.4 Demand parameters

Ashok and Ben-Akiva (Ashok & Ben-Akiva, 1993, Ashok, 1996, Ashok & Ben-Akiva 2000, 2002) formulate the real–time OD estimation and prediction problem as a state–space model and solve it using a Kalman Filtering algorithm. One interesting characteristic of this approach is the use of deviations of OD flows (instead of the OD flows themselves) as variables. The use of deviations incorporates the wealth of structural information about spatial and temporal relationships between OD flows contained in the historical estimates into the OD estimation framework. The real–time OD estimation and prediction framework has been implemented in the DynaMIT DTA system (Antoniou et al., 1997, Ben-Akiva et al., 2002). An efficient solution algorithm for the OD estimation problem has been presented by Bierlaire and Crittin (2004).

Zhou and Mahmassani (2004) develop a similar Kalman-filter based adaptive OD estimation and prediction procedure using a polynomial trend filter to recursively capture demand deviations from a priori demand estimates.

2.5 Conclusion

The problem of on–line calibration of DTA systems has received some attention in the literature. Most existing methodologies, however, impose serious constraints and make restrictive assumptions. In particular, the components of a DTA system are considered in a sequential approach and iterative/heuristic approaches are proposed to estimate the appropriate parameters on–line.

Individual approaches for the on–line calibration of subsets of the parameters have also been developed. Such approaches update only a subset of the parameters in a DTA system. Therefore, all error or uncertainty is attributed to one source, which is unrealistic. Instead, an approach is needed that jointly estimates demand and supply parameters simultaneously and captures the complex demand and supply interactions (Ben-Akiva et al., 2002), thus ensuring consistency between the estimated parameters.

3. State–space formulation

A classical technique for dealing with dynamic systems is state–space modeling. In this section, the on–line calibration problem is formulated as a state–space model, comprising:

- Transition equations that capture the evolution of the state vector over time, and

- Measurement equations that capture the mapping of the state vector on the measurements.

Given that state–space models have been extensively studied and efficient algorithms have been developed to solve them, this formulation will lead us naturally to Section 4 where solution approaches are discussed.

The first step in developing a state–space model is to define the state vector. In this context, the parameters and inputs that need to be calibrated define the state. Measurement and transition equations are developed next, followed by a reformulation of the problem in terms of deviations.

3.1 State vector

The concept of the state (or state vector) is fundamental in the description of a state–space model. The state vector $\mathbf{x}_h$ is defined as the minimal set of data that is sufficient to uniquely describe the dynamic behavior of the system at time interval h (the assumption of a discrete, stochastic, dynamic system is made). The state vector includes the parameters $\boldsymbol{\pi}_h$ that need to be calibrated during time interval h. The main parameters for the on–line calibration problem for a DTA system include:

- OD flows,
- Behavioral model parameters, such as route, departure and mode choice model parameters,
- Speed–density relationship parameters, and
- Segment capacities.

It should be noted, however, that the approach is general and can easily incorporate a different set of parameters.

3.2 Measurement equations

Available information is associated with the unknown parameter values through measurement equations. A priori values of the model parameters provide direct measurements of the unknown parameters. Surveillance information, on the other hand, can be used to formulate indirect measurement equations, where the output of the simulator model $\boldsymbol{S}$ (when the unknown set of parameter values is used as input) would match the surveillance information.

By definition, a direct measurement provides a preliminary estimate of a parameter. Within the context of on–line calibration, preliminary estimates of the parameters are provided by the off–line calibration. Therefore, the vector of off–line calibrated parameter values $\boldsymbol{\pi}_h^{\alpha}$ can be used as an a priori estimate of the true parameter vector $\boldsymbol{\pi}_h$. The a priori values of the input parameters can be expressed as a function of the "true" parameters:

$$\pi_h^a = \pi_h + \mathbf{v}'_h \tag{1}$$

where $\mathbf{v}'_h$ is a vector of random error terms.

Direct measurements of some OD flows could also be available from advanced surveillance technologies, such as Automated Vehicle Identification (AVI) systems or probe vehicles. Such technologies allow the tracking of equipped vehicles as they move through the network, thus obtaining detailed surveillance information (based on a sample of the population). When the vehicles can be detected close to their origin and their destination, it

is possible to infer direct measurements of OD flows (Antoniou et al., 2004). Such information could easily be incorporated as additional direct measurements.
Practically any type of available traffic measurements can be used as indirect measurement equations, linking the observed traffic measurements with their simulated counterparts when a particular set of parameters is used as input. In the general case, modeled trips last longer than one interval. Therefore, simulated trajectories of vehicles are impacted by the traffic conditions during previous intervals (and consequently by the model parameters used during these intervals). The simulated traffic measurements during time interval h can therefore be represented as:

$$\mathbf{M}_h^s = S(\pi_h, \pi_{h-1}, ..., \pi_{h-p}) = S(\Pi_h) \tag{2}$$

where $\boldsymbol{S}$ is a mapping of the input parameters onto the measurements (representing the simulation model), p is the number of intervals required for the longest trip in the network, and $\Pi_h=\pi_h, \pi_{h-1},\ldots, \pi_{h-p}$ is an augmented vector of parameters.
The relationship between the observed and the simulated measurements can then be written as follows:

$$\mathbf{M}_h^o = \mathbf{M}_h^s + \upsilon'_h \tag{3}$$

where $\boldsymbol{\upsilon}'_h=\boldsymbol{\upsilon}_h^f+\boldsymbol{\upsilon}_h^s+\boldsymbol{\upsilon}_h^m$ is a compound observation error comprising three error sources:

- $\boldsymbol{\upsilon}_h^f$ captures structural errors (due to the inexactness of the simulation models),
- $\boldsymbol{\upsilon}_h^s$ captures simulation errors (e.g. sampling and numerical errors), and
- $\boldsymbol{\upsilon}_h^m$ captures measurement errors.

As it is not possible to distinguish between these three error components, however, they will be treated together. Furthermore, it is assumed that $\boldsymbol{\upsilon}'_h$ is independent from the error vector $\mathbf{v}'_h$ introduced in Equation 1.

3.3 Transition equations

Transition equations capture the evolution of the state vector over time. A typical formulation for the transition equation relates the state during a given interval to a series of states from previous intervals. A general formulation of such a transition equation would be:

$$\pi_{h+1} = T(\pi_h, \pi_{h-1}, ..., \pi_{h-p}) + \eta'_h \tag{4}$$

where $\boldsymbol{T}$ is a function capturing the dependence of the parameter vector $\boldsymbol{\pi}_{h+1}$ during interval $h+1$ on the values of the parameter vector during the past several intervals, p is the number of past parameter vectors that are considered, and $\boldsymbol{\eta}'_h$ is a vector of random error terms.
A common approach to the representation of transition equations is the use of autoregressive processes. Expressed as an autoregressive function, the transition equation can be written in matrix form as follows:

$$\pi_{h+1} = \sum_{q=h-p}^{h} F_q^{h+1} \pi_q + \eta''_h \tag{5}$$

The three components of the state vector (OD flows, speed–density relationship parameters, capacities) represent distinct aspects of the transportation problem and have different

characteristics. Therefore, each of these may evolve over time according to a distinct autoregressive process. This can easily be handled by writing a separate transition equation like the one presented in Equation 5 for each such autoregressive process.

3.4 The idea of deviations

Suppose that the model parameters and inputs have been estimated from historical data for several previous days or months. These already estimated (demand and supply) parameters embody a wealth of information about the relationships that affect trip making and traffic dynamics, as well as their temporal and spatial evolution. It is desirable to incorporate as much historical information into the formulation as possible. The most straightforward way to achieve this is to use deviations of the model parameters from best available estimates instead of the actual parameters themselves as state variables. Thus, the model formulation would indirectly take into account all the available a priori structural information. The use of deviations has been proposed by Ashok and Ben-Akiva (1993) for the OD estimation and prediction problem.

Using deviations also has other benefits. Traffic flow variables have skewed distributions (unlike the normal distribution which is symmetric). On the other hand, the corresponding deviations of these variables from available estimates would have symmetric distributions and hence are more amenable to approximation by a normal distribution. A normal distribution for the model variables is a useful property for the available statistical tools such as the Kalman Filter extensions used in this research.

The state vector can therefore be expressed as deviations from best historical values: $\Delta\pi_h=\pi_h-\pi_h^H$. The transition equation can easily be reformulated with respect to the new state vector as:

$$\begin{aligned} \pi_{h+1}-\pi_{h+1}^H &= \sum_{q=h-p}^{h} F_q^{h+1}(\pi_q-\pi_q^H)+\eta_h \Rightarrow \\ \Delta\pi_{h+1} &= \sum_{q=h-p}^{h} F_q^{h+1}\cdot\Delta\pi_q+\eta_h \end{aligned} \tag{6}$$

Similarly, the direct measurement equation can be written in deviations' form as:

$$\begin{aligned} \pi_h^a-\pi_h^H &= \pi_h-\pi_h^H+\mathbf{v}_h \Rightarrow \\ \Delta\pi_h^a &= \Delta\pi_h+\mathbf{v}_h \end{aligned} \tag{7}$$

It should be noted that π_h^α and π_h^H capture essentially the same thing: an available estimate of the state vector. However, there are subtle differences and —in the interest of generality— a distinction is made. For example, the a priori parameters π_h^α may correspond to the parameters obtained from the off–line calibration, while the historical parameters π_h^H may refer to the latest available estimates (e.g. values obtained from the same interval the previous day).

Finally, the indirect measurement equation can be rewritten as:

$$\begin{aligned} \mathbf{M}_h-\mathbf{M}_h^H &= S(\pi_h)-\mathbf{M}_h^H+\upsilon_h \Rightarrow \\ \Delta\mathbf{M}_h &= S(\pi_h^H+\Delta\pi_h)-\mathbf{M}_h^H+\upsilon_h \end{aligned} \tag{8}$$

where the historical traffic measurement vector $\mathbf{M}_h^H$ can be obtained from an evaluation of the historical parameters π_h:

$$\mathbf{M}_h^H = S(\pi_h) \tag{9}$$

3.5 The model at a glance

The on–line calibration algorithm has been expressed in deviations' form (where Equation 6 is the transition equation and Equations 7 and 8 are the measurement equations). The complete state–space model is shown below for clarity:

$$\Delta\pi_{h+1} = \sum_{q=h-p}^{h} F_q^{h+1} \cdot \Delta\pi_q + \eta_h$$

$$\Delta\pi_h^a = \Delta\pi_h + \mathbf{v}_h \tag{10}$$

$$\Delta\mathbf{M}_h = S(\pi_h^H + \Delta\pi_h) - \mathbf{M}_h^H + \upsilon_h$$

Before moving to the presentation of applicable solution approaches (Section 4), it is useful to express the model in the following form:

$$\mathbf{x}_{h+1} = \mathbf{f}(\mathbf{x}_h) + \mathbf{w}_h \tag{11}$$

$$\mathbf{y}_h = \mathbf{h}(\mathbf{x}_h) + \mathbf{u}_h \tag{12}$$

where Equation 11 is the transition equation and Equation 12 is the measurement equation. This form is obtained directly from Equations 10 if we denote

$$\mathbf{x}_h = \Delta\pi_h$$

$$\mathbf{y}_h = \begin{bmatrix} \Delta\pi_h^a \\ \Delta\mathbf{M}_h \end{bmatrix}$$

$$\mathbf{f}(\mathbf{x}_h) = \sum_{q=h-p}^{h} F_q^{h+1}\mathbf{x}_q$$

$$\mathbf{h}(\mathbf{x}_h) = \begin{bmatrix} \Delta\pi_h \\ S(\pi_h^H + \Delta\pi_h) - \mathbf{M}_h^H \end{bmatrix}$$

$$\mathbf{u}_h = \begin{bmatrix} \mathbf{v}_h \\ \upsilon_h \end{bmatrix}$$

Furthermore, the following assumptions are made on the error vectors $\mathbf{w}_h$ and $\mathbf{u}_h$:

1. $E[\mathbf{w}_h] = 0$

2. $E[\mathbf{w}_h\ \mathbf{w'}_m]=\mathbf{Q}_h\delta_{hm}$ where δ_{hm} is the Kronecker delta, i.e. $\delta_{hm}=1$ if $h=m$ and 0 otherwise $\forall h,m$, and $\mathbf{Q}_h$ is a variance–covariance matrix.
3. $E[\mathbf{u}_h]=0$
4. $E[\mathbf{u}_h\ \mathbf{u'}_m]=\mathbf{R}_h\delta_{hm}$ where δ_{hm} is the Kronecker delta, i.e. $\delta_{hm}=1$ if $h=m$ and 0 otherwise $\forall h,m$, and $\mathbf{R}_h$ is a variance–covariance matrix.
5. $E[\mathbf{u}_h\ \mathbf{w'}_m]=0\ \forall h,m$, i.e. the errors of the transition and measurement equations are uncorrelated.

These assumptions allow for the derivation of the Kalman Filter–based solution approaches. The assumption of no serial correlation for the transition equation can be defended because the unobserved factors that could be correlated over time are captured by the historical matrix $\boldsymbol{\pi}^{\alpha}{}_h$ (or $\boldsymbol{\pi}^{H}{}_h$). In some situations (e.g. incidents), however, this assumption might break down. A violation of this assumption, however, can be easily taken care of by using a variant of the estimation algorithm. (An algorithm to handle correlated errors in the transition or measurement equations can be found, for example, in Chui and Chen, 1999). The assumption of no serial correlation for the measurement equation can be defended using a similar argument. However, this assumption might also break down if, for example, a specific detector consistently under–estimates or over–estimates a link volume on a particular day. Again, it is easy to relax this assumption and use a variant of the estimation algorithm.

3.6 An alternative formulation

The on-line calibration problem can be formulated as a minimization problem where the objective function aims to jointly minimize the following components:

- $\boldsymbol{\varepsilon}^{o}{}_h$: deviation of simulated traffic conditions $\mathbf{M}^{s}{}_h$ from the respective observed measurements $\mathbf{M}^{o}{}_h$, and
- $\boldsymbol{\varepsilon}^{\alpha}{}_h$: deviation of a set of parameters and inputs $\boldsymbol{\pi}_h$ (over which the optimization is performed) from their a priori values $\boldsymbol{\pi}^{\alpha}{}_h$.

The objective function could then be expressed as:

$$\pi_h \min[N_1(\varepsilon_h^o) + N_2(\varepsilon_h^a)] \tag{13}$$

where $\mathbf{N}_i(.)$ are appropriate functions measuring the magnitude of the errors. For example, $\mathbf{N}_i(.)$ may be the Euclidian norm.

Substituting the expressions for the error terms from Equations 1 and 3, the objective function can be restated as:

$$\pi_h \min[N_1(\mathbf{M}_h^o - \mathbf{M}_h^s) + N_2(\pi_h^a - \pi_h)] \tag{14}$$

The above formulation can be made operational in a number of different ways, depending on the assumptions regarding the nature of the various error terms and the functional forms of $\mathbf{N}_i(.)$. The various formulations may lead to different solution approaches with different convergence and computational properties. For example, if $\boldsymbol{\varepsilon}_{\alpha}$ and $\boldsymbol{\varepsilon}_{o}$ are assumed to be normally distributed the formulation reduces to the following generalized least squares (GLS) problem:

$$\begin{aligned}\pi_h \min[&(\mathbf{M}_h^o - \mathbf{M}_h^s)'\mathbf{W}^{-1}(\mathbf{M}_h^o - \mathbf{M}_h^s) + \\ &+(\pi_h^a - \pi_h)'\mathbf{V}^{-1}(\pi_h^a - \pi_h)]\end{aligned} \tag{15}$$

where **W** and **V** are the variance–covariance matrices of the measurements and a priori values, respectively. The solution $\boldsymbol{\pi}^*_h$ to this optimization problem can then be obtained from:

$$\pi_h^* = \pi_h \operatorname{argmin}[(\mathbf{M}_h^o - \mathbf{M}_h^s)'\mathbf{W}^{-1}(\mathbf{M}_h^o - \mathbf{M}_h^s) + \\ +(\pi_h^a - \pi_h)'\mathbf{V}^{-1}(\pi_h^a - \pi_h)] \tag{16}$$

In an on–line application, however, this formulation would be impractical since the problem needs to be solved at every time interval, with all the information on previous time intervals (because of the temporal correlations between the errors). Stating this problem as a state–space model, which can then be solved efficiently using recursive methods such as Kalman Filtering techniques, overcomes this difficulty.

The on–line calibration approach can also be solved using other algorithms for non–linear systems of equations. A particularly suitable algorithm has been recently developed (Crittin, 2003, Crittin & Bierlaire, 2003) as a generalization of secant methods. The proposed class of methods calibrates a linear model based on several previous iterates. The difference with existing approaches is that the linear model to interpolate the function is not imposed. Instead, the linear model, which is as close as possible to the nonlinear function (in the least–squares sense), is identified.

4. Solution approaches

The Kalman Filter is the optimal minimum mean square error (MMSE) estimator for linear state–space models (Kalman, 1960). However, the on–line calibration approach is non–linear (due to the indirect measurement equation). Since many other interesting problems are non–linear, solutions for non–linear models have been sought, leading to the development of modified Kalman Filter methodologies. The most straightforward extension is the Extended Kalman Filter (EKF), in which optimal quantities are approximated via first order Taylor series expansion (linearization) of the appropriate equations (Kalman, 1960, Gelb, 1974). A special case of the EKF with very favorable computational properties is the Limiting Extended Kalman Filter (LimEKF; Limiting Kalman Filters are discussed e.g. in Chui & Chen, 1999). The Unscented Kalman Filter (UKF) (Julier et al., 1995; Julier & Uhlmann, 1997; Wan et al., 2000; Wan & van der Merwe, 2000; van der Merwe et al., 2000) is an alternative filter. The main difference between the EKF and UKF lies in the representation of the random variables for propagation through the system dynamics.

4.1 Extended Kalman filter

The Extended Kalman Filter employs a linearization of the non–linear relationship to approximate the measurement equation with a first–order Taylor expansion:

$$\mathbf{H}_h = \left.\frac{\partial \mathbf{h}(\mathbf{x}^*)}{\partial \mathbf{x}^*}\right|_{\mathbf{x}^* = \mathbf{X}_{h|h-1}} \tag{17}$$

In words, the Extended Kalman Filter main steps are as follows. Suppose that a starting estimate of the state $\mathbf{X}_0$ is available (Equation 18), along with an estimate of the initial state variance–covariance matrix $\mathbf{P}_0$ (Equation 19). A time update phase makes a prediction of the state (Equation 20) and its covariance matrix (Equation 21) for the next time interval.

The measurement update phase incorporates the new information about the measurement vector $\mathbf{Y}_h$ and uses it to correct the prediction of the state made during the time update. The measurement matrix $\mathbf{H}_h$ is obtained through an intermediate linearization step (Equation 22). Instrumental in this process is the Kalman gain $\mathbf{G}_h$, which is computed as per Equation 23. The state can then be updated (corrected) using Equation 24. Similarly, the state covariance is updated using Equation 25.

Algorithm 1. Extended Kalman Filter
Initialization

$$\mathbf{X}_{0|0} = \mathbf{X}_0 \tag{18}$$

$$\mathbf{P}_{0|0} = \mathbf{P}_0 \tag{19}$$

for *h=1* to *N* **do**
Time update

$$\mathbf{X}_{h|h-1} = \mathbf{F}_{h-1}\mathbf{X}_{h-1|h-1} \tag{20}$$

$$\mathbf{P}_{h|h-1} = \mathbf{F}_{h-1}\mathbf{P}_{h-1|h-1}\mathbf{F}_{h-1}^T + \mathbf{Q}_h \tag{21}$$

Linearization

$$\mathbf{H}_h = \left.\frac{\partial \mathbf{h}(\mathbf{x}^*)}{\partial \mathbf{x}^*}\right|_{\mathbf{x}^*=\mathbf{X}_{h|h-1}} \tag{22}$$

Measurement update

$$\mathbf{G}_h = \mathbf{P}_{h|h-1}\mathbf{H}_h^T\left(\mathbf{H}_h\mathbf{P}_{h|h-1}\mathbf{H}_h^T + \mathbf{R}_h\right)^{-1} \tag{23}$$

$$\mathbf{X}_{h|h} = \mathbf{X}_{h|h-1} + \mathbf{G}_h\left[\mathbf{Y}_h - \mathbf{h}\left(\mathbf{X}_{h|h-1}\right)\right] \tag{24}$$

$$\mathbf{P}_{h|h} = \mathbf{P}_{h|h-1} - \mathbf{G}_h\mathbf{H}_h\mathbf{P}_{h|h-1} \tag{25}$$

end for

Further information on the Extended Kalman Filter can be found in many texts, including for example Sorenson (1985) and Chui and Chen (1999).
The on–line calibration approach presented in previous sections does not —in general— have an analytical representation. Therefore, in order to perform the linearization step (Equation 22) it is necessary to use numerical derivatives. Assuming the use of central derivatives, it is necessary to evaluate the function *2n* times, where *n* is the dimension of the state vector. (If forward derivatives are used, then this number drops to *n+1* evaluations.) Each such evaluation implies one run of the simulator. Therefore, it becomes apparent that this process of linearization dominates the computational complexity of the algorithm.

4.2 Limiting extended Kalman filter

In this section, a special case of the Extended Kalman Filter is presented that significantly improves the computational performance of the algorithm. As mentioned in Section 4.1, the most computationally intensive step in the EKF algorithm is the linearization of the measurement equation (Equation 22), as it requires the use of numerical derivatives. Another costly operation is the inversion required for the computation of the Kalman gain (Equation 23). In real–time applications, it may be possible to replace the Kalman Gain matrix $\boldsymbol{G}_h$ by a constant gain matrix considerably decreasing the computation time. The limiting Kalman Filter will be defined by replacing $\boldsymbol{G}_h$ with its "limit" $\boldsymbol{G}$, called the limiting (or stable) Kalman gain matrix (Chui & Chen, 1999). The main steps of the Limiting Extended Kalman Filter (LimEKF) algorithm are presented in Algorithm 2. The differences from the EKF algorithm are limited to the computation of the numerical derivative (which is not computed on–line in the LimEKF) and the use of the limiting Kalman gain $\boldsymbol{G}$ for every iteration (Equations 30 and 31).

The limiting Kalman gain matrix can be computed off–line. The simplest way would be to express the limiting Kalman gain matrix as the average of a number of available Kalman gain matrices:

$$\mathbf{G} = \frac{\sum_{m=1:M} \mathbf{G}_m}{M} \tag{32}$$

where $\mathbf{G}_m$ is the Kalman gain obtained from EKF during interval m and M is the total number of available Kalman gain matrices. Several strategies can be developed to improve the quality of the limiting Kalman gain. For example, the EKF could be run off–line, with each run producing a new Kalman gain matrix. These Kalman gain matrices could then be used to update the *limiting* Kalman gain matrix. Another strategy would be to consider only the last few Kalman gain matrices, i.e. use a type of moving average. Weighted averages (e.g. using lower weights for "older" gain matrices) can also be considered.

Algorithm 2. Limiting Extended Kalman Filter

Generation of *limiting* Kalman gain matrix **G** and **H**

Initialization

$$\mathbf{X}_{0|0} = \mathbf{X}_0 \tag{26}$$

$$\mathbf{P}_{0|0} = \mathbf{P}_0 \tag{27}$$

for h=*1* to N **do**

Time update

$$\mathbf{X}_{h|h-1} = \mathbf{F}_{h-1}\mathbf{X}_{h-1|h-1} \tag{28}$$

$$\mathbf{P}_{h|h-1} = \mathbf{F}_{h-1}\mathbf{P}_{h-1|h-1}\mathbf{F}_{h-1}^{T} + \mathbf{Q}_h \tag{29}$$

Measurement update

$$\mathbf{X}_{h|h} = \mathbf{X}_{h|h-1} + \mathbf{G}\left[\mathbf{Y}_h - \mathbf{h}\left(\mathbf{X}_{h|h-1}\right)\right] \tag{30}$$

$$\mathbf{P}_{h|h} = \mathbf{P}_{h|h-1} - \mathbf{GHP}_{h|h-1} \tag{31}$$

end for

The main component of the Kalman gain matrix is the derivative $\mathbf{H}_h$ of the measurement equation. This is directly required in Equation 31. Using the same principle as above, it is possible to replace the time–dependent matrix $\mathbf{H}_h$ with the average $\mathbf{H}$ of a number of available matrices:

$$\mathbf{H} = \frac{\sum_{m=1:M} \mathbf{H}_m}{M} \tag{33}$$

where $\mathbf{H}_m$ is the matrix obtained from EKF during interval m and M is the total number of available matrices. The resulting matrix $\mathbf{H}$ can be then used to update the state covariance in Equation 31.

4.3 Unscented Kalman filter

The Unscented Kalman Filter (UKF) (Julier et al., 1995; Julier and Uhlmann, 1997; Wan et al., 2000; Wan and van der Merwe, 2000; van der Merwe et al., 2000) is an alternative filter for dynamic state–space models. The UKF uses a deterministic sampling approach (Unscented Transformation, UT) to represent a random variable using a number of deterministically selected sample points (often called *sigma points*). These points capture the mean and covariance of the random variable and, when propagated through the *true* nonlinear system, capture the posterior mean and covariance accurately to the second order (Taylor series expansion).

The Unscented Transformation is based on the intuitive expectation that "with a fixed number of parameters it should be easier to approximate a Gaussian distribution than it is to approximate an arbitrary nonlinear function/transformation" (Julier & Uhlmann, 1996). Following this intuition, one would seek to find a parameterization that would capture the mean and covariance information while at the same time permitting the direct propagation of the information through an arbitrary set of nonlinear equations. This can be accomplished by generating a discrete distribution having the same first and second (and possibly higher) moments, where each point in the discrete approximation can be directly transformed. The mean and covariance of the transformed ensemble can then be computed as the estimate of the nonlinear transformation of the original distribution.

Given an n–dimensional distribution with covariance $\mathbf{P}$, it is possible to generate $O(n)$ points with the same sample covariance from the columns (or rows) of the matrices $\pm\sqrt{n\mathbf{P}}$ (the positive and negative roots). This set of points has a zero mean. However, simply adding the mean $\mathbf{X}$ of the original distribution to each of the points yields a symmetric set of $2n$ points with the desired mean and covariance. Because the set is symmetric its odd central moments are zero, so its first three moments match the original Gaussian distribution.

The main steps of the Unscented Transformation (UT) for calculating the statistics of a random variable that undergoes a nonlinear transformation (e.g. $\mathbf{y_h} = f(\mathbf{x_h})$) are presented in Algorithm 3 (Julier & Uhlmann, 1997). Let the n-dimensional random variable $\mathbf{x}_h$ with covariance matrix $\mathbf{P}_{x,h}$ denote the state for time interval h. Since this algorithm also considers the covariance of the measurement vector $\mathbf{P}_{y,h}$ during interval h and the covariance of the state and measurement vectors $\mathbf{P}_{xy,h}$, the covariance of the state vector will be denoted

$\mathbf{P}_{x,h}= \mathbf{P}_h$ in order to avoid confusion.
To calculate the statistics of **y**, a matrix *X* is generated using *2n+1* weighted *sigma points*. $\kappa \in R$ is a scaling parameter and $\left(\sqrt{(n+\kappa)\mathbf{P}_{x,h}}\right)_i$ is the *i*th row or column of the matrix square root of (n+κ) $\mathbf{P}_{x,h}$. A Cholesky decomposition (Golub & van Loan, 1996) can be used for this step. The value of the scaling parameter κ has a direct effect on the scaling of the points and is an input to the algorithm. The constant α determines the spread of the sigma points around $\bar{\mathbf{x}}$ and is usually set to $0.0001 \leq \alpha \leq 1$. *b* is used to incorporate prior knowledge of the distribution of **X** (for Gaussian distributions, *b*=2 is optimal). The weights are not time–dependent and do not need to be recomputed for every time interval.

Algorithm 3. Unscented Transformation
Generation of *sigma points*

$$X_{0,h} = \mathbf{x_h} \tag{34}$$

for *i=1* to *n* **do**

$$X_{i,h} = \mathbf{x_h} + \left(\sqrt{(n+\kappa)\mathbf{P}_{x,h}}\right)_i \tag{35}$$

end for
for *i=n+1* to *2n* **do**

$$X_{i,h} = \mathbf{x_h} - \left(\sqrt{(n+\kappa)\mathbf{P}_{x,h}}\right)_i \tag{36}$$

end for
Generation of weights

$$W_0^m = \kappa / (n+\kappa) \tag{37}$$

$$W_0^c = \kappa / (n+\kappa) + (1 - a^2 + b) \tag{38}$$

for *i=1* to *2n* **do**

$$W_i^m = W_i^c = 1 / \left[2(n+\kappa)\right] \tag{39}$$

end for

The main steps of the UKF are presented in Algorithm 4 (van der Merwe et al., 2000). The initialization step uses the Unscented Transformation (Algorithm 3) to generate the *2n+1 sigma points* and appropriate weights for the mean and covariance computations. A time and measurement update step is repeated for each run of the algorithm.
The first step in the time update phase is the propagation of the *sigma points* through the transition equation (Equation 40). The prior estimate of the state vector is computed as a weighted sum of the propagated sigma points (Equation 41). A similar approach is used for the prior estimate of the state covariance (Equation 42). The true measurement equation is used to transform the sigma points into a vector of respective measurements (Equation 43). The measurement vector is computed as a weighted sum of the generated measurements (Equation 44).

The computation of the Kalman gain (and consequently the "correction" phase of the filtering) is based on the covariance of the measurement vector (Equation 45) and the covariance of the state and measurement vectors (Equation 46). These are computed using the weights (that were obtained from the Unscented Transformation during the initialization step) and the deviations of the sigma points from their means.
The Kalman gain is then computed from these covariance matrices (Equation 47). Equation 48 introduces the measurement vector $\mathbf{y}_h$ and uses the Kalman gain to correct the state estimate $\mathbf{x}_h$. The state covariance is updated using Equation 49.

Algorithm 4. Unscented Kalman Filter
for *h=1* to *N* **do**
Generate *sigma points* and weights using the Unscented Transformation (Algorithm 3)
Time update

$$X_{h|h-1} = \mathbf{f}(X_{h-1}) \tag{40}$$

$$\mathbf{x}_{h|h-1} = \sum_{i=0}^{2n} W_i^m X_{i,h|h-1} \tag{41}$$

$$\mathbf{P}_{x,h|h-1} = \sum_{i=0}^{2n} W_i^c \left(X_{i,h|h-1} - \mathbf{x}_{h|h-1} \right) \times \tag{42}$$

$$\times \left(X_{i,h|h-1} - \mathbf{x}_{h|h-1} \right)^T + \mathbf{Q}_h$$

$$Y_{i,h|h-1} = \mathbf{h}\left(X_{i,h|h-1} \right) \tag{43}$$

$$\mathbf{y}_{h|h-1} = \sum_{i=0}^{2n} W_i^m Y_{i,h|h-1} \tag{44}$$

Measurement update

$$\mathbf{P}_{y,h} = \sum_{i=0}^{2n} W_i^c (Y_{i,h|h-1} - \mathbf{y}_{h|h-1}) \times \tag{45}$$

$$\times (Y_{i,h|h-1} - \mathbf{y}_{h|h-1})^T + \mathbf{R}_h$$

$$\mathbf{P}_{xy,h} = \sum_{i=0}^{2n} W_i^c (X_{i,h|h-1} - \mathbf{x}_{h|h-1}) \times \tag{46}$$

$$\times (Y_{i,h|h-1} - \mathbf{y}_{h|h-1})^T$$

$$\mathbf{G}_h = \mathbf{P}_{xy,h} \mathbf{P}_{y,h}^{-1} \tag{47}$$

$$\mathbf{x}_h = \mathbf{x}_{h|h-1} + \mathbf{G}_h \left(\mathbf{y}_h - \mathbf{y}_{h|h-1} \right) \tag{48}$$

$$\mathbf{P}_{x,h} = \mathbf{P}_{x,h|h-1} - \mathbf{G}_h \mathbf{P}_{y,h} \mathbf{G}_h^T \tag{49}$$

end for

5. Application

The objective of this application is to demonstrate the performance of the joint on–line estimation of demand and supply parameters for a DTA system. The situation when only the demand parameters are calibrated on–line is used as the base. Since this reference problem is the usual OD estimation problem, GLS or Kalman Filter algorithms can be applied. When both the demand and supply parameters are jointly updated on–line, however, the problem cannot be represented analytically and the algorithms presented in Section 4 can be used. The on–line calibration has been implemented and demonstrated as it applies to the DynaMIT–R DTA system. Three algorithms have been implemented (EKF, LimEKF and UKF) and their performance for a freeway network in Southampton, UK, is presented.

DynaMIT–R is a state–of–the–art DTA system. The high–level framework of DynaMIT–R has been presented in Fig. 1. The key to the functionality of DynaMIT–R is its detailed network representation, coupled with models of traveler behavior. Through an effective integration of historical databases with real-time inputs, DynaMIT–R is designed to efficiently achieve:

- Real time estimation of network conditions.
- Rolling horizon predictions of network conditions in response to alternative traffic control measures and information dissemination strategies.
- Generation of traffic information and route guidance to steer drivers towards optimal decisions.

To sustain users' acceptance and achieve reliable predictions and credible guidance, DynaMIT–R incorporates unbiasedness and consistency into its core operations. Unbiasedness guarantees that the information provided to travelers is based on the best available knowledge of current and anticipated network conditions. Consistency ensures that DynaMIT–R's predictions of expected network conditions match what drivers would experience on the network. DynaMIT–R has the ability to trade-off level of detail (or resolution) and computational practicability, without compromising the integrity of its output. A more detailed description of DynaMIT-R can be found in Ben-Akiva et al. (2002).

The network includes a 35km long part of freeway (M27) from Southampton, U.K. The network starts to the west of the city of Southampton, then goes around it, and continues eastbound towards Portsmouth. The network also includes seven off–ramps and eight on–ramps. A schematic representation is shown in Fig. 2, which indicates the ten sensors, which provide traffic information (counts, speeds and occupancies). Traffic is loaded onto the network via twenty origin-destination pairs. The peak period for this direction is the afternoon/evening. Weekday data (speeds and densities) from the first two weeks of September 2001 have been used. Fig. 3 shows the speed and density distribution at a mainline sensor for these days.

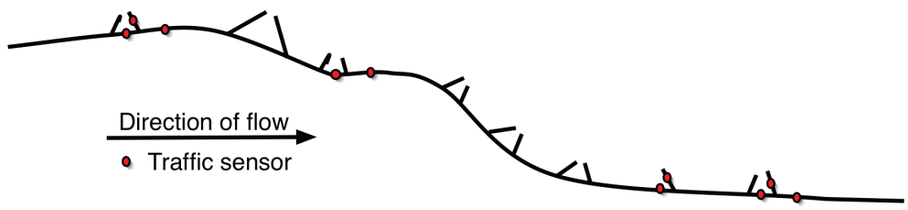


Fig. 2. Schematic of the study network

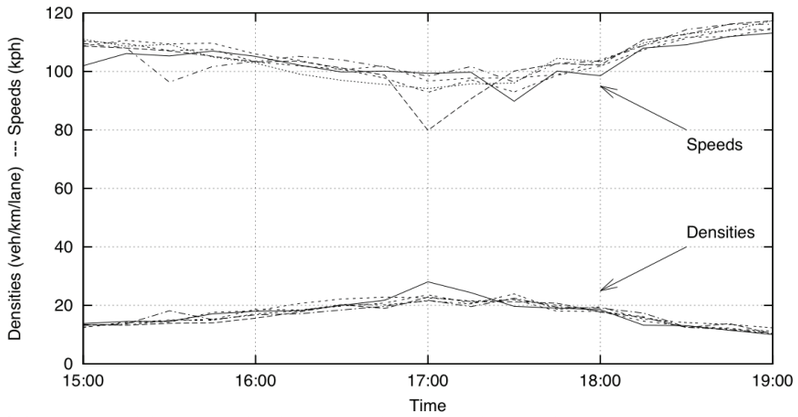


Fig. 3. Speeds/densities (days with dry weather conditions)

5.1 Methodology

Since an off–line calibration was not available for this network, the first step in this case study was to perform an off–line calibration. Data from five weekdays (without adverse weather or incidents) during the first two weeks of September 2001 were used and a sequential off–line calibration approach was followed. Supply parameters were first calibrated. Speed–density relationship parameters were obtained by fitting speed and density data to the appropriate functional form. Speed–density relationship parameters were obtained by using non–linear regression to fit speed and density data to the speed–density relationship used by DynaMIT–R:

$$u = u_f \left[1 - \left(\frac{\max(0, K - K_{min})}{K_{jam}} \right)^{\beta} \right]^{\alpha} \tag{50}$$

where u denotes the speed, u_f is the free flow speed, K is the density, K_{min} is the minimum density, K_{jam} is the *jam* density and α and β are model parameters. The 45 segments of the network were split into three groups with homogeneous characteristics. The mainline segments were classified into two types, while the ramp segments were grouped together.

Segment capacities were estimated using Highway Capacity Manual guidelines (HCM, 2000). Capacity computations are usually based on appropriate guidelines (e.g. the Highway Capacity Manual, HCM, 2000, for the United States). Although the study network is in the United Kingdom, no equivalent national guidelines are available for the United Kingdom and therefore the Highway Capacity Manual guidelines were used. Analysis and comparison of the estimated capacities against observed counts were performed, to ensure

that the capacity values did not result in counterintuitive results (such as observed sensor counts exceeding the segment capacity).

Then, using the calibrated parameters as inputs, time–dependent OD flows were estimated. A sequential OD estimation approach (Balakrishna, 2002) was applied on five weekdays. A static seed matrix was used to initialize the process. For the first day, the estimated OD flows from each interval were used as historical estimates for the next interval. The estimated flows for each day were then used as historical flows for the next day. An ordinary least squares (OLS) approach was used for the first two days. At the end of the second day, measurement error covariances were estimates from the residuals of the fitted sensor counts and OD flows from their observed or historical values. This allowed for the use of a generalized least squares (GLS) approach for the remaining days. Estimated OD flows across time intervals were used to estimate autoregressive factors for the transition equation. The planning version of the DynaMIT system (DynaMIT–P) was used in this process. The Normalized Root Mean Square error (RMSN) statistic for the fit-to-counts was equal to 0.1232. The total fit of the speeds, as quantified by the Normalized Root Mean Square error (RMSN) statistic, was equal to 0.1102. The RMSN statistic was computed using the following formula (Ashok & Ben-Akiva, 2000, 2002):

$$RMSN = \frac{\sqrt{N \sum_N (y - \hat{y})^2}}{\sum_N y}$$

where N is the number of observations, y denotes an observation and $\hat{y}$ is the corresponding estimated (predicted) value.

One of the main outputs of DTA systems is traffic information and guidance, usually in the form of travel times. Speeds are the closest surveillance measurement and there are ways to compute travel times from speeds. Furthermore, given a properly calibrated traffic estimation and prediction system it is possible to obtain (simulated) travel times directly. On the other hand, the most ubiquitous traffic measurement is traffic counts. Therefore, the two first measures of effectiveness are based on fit of estimated (predicted) speeds and counts with observed values, quantified using the normalized root mean square error (RMSN).

The computational performance of the algorithms is another important consideration. In particular, given the *on–line* nature of the application, it is important to understand the computational complexity of each algorithm. Given a simulation–based DTA system, the function evaluations (required by the solution approaches) are by far the most computational intensive task, since each evaluation implies one run of the simulator. In the subsequent discussion, the number of function evaluations are used as a measure of effectiveness for each algorithm.

The state vector for the on–line calibration consists of OD flows, segment capacities and speed–density relationship parameters. (Route choice parameters are not included due to the nature of the network used for the case study.) The total dimension is 80, broken down in:

- 20 OD flows
- 45 segment capacities
- 15 speed–density relationship parameters: a speed–density relationship (and therefore 5 parameters: free flow speed, minimum and jam density and exponents α and β) has been defined for each of the three segment types.

The duration of the estimation and prediction intervals was set to fifteen minutes. OD estimation requires that the count measurements are aggregated to the duration of the interval (i.e. fifteen minutes in this case study). To maintain consistency between the various algorithms, this level of aggregation has been maintained for the counts in the on–line calibration approach. Furthermore, minute–by–minute speed and density surveillance information was used.

Therefore, the measurement vector for each fifteen–minute interval consisted of 390 elements:

- 15–minute count measurements: 10 count measurements in total
- Minute–by–minute speed measurements: 150 speed measurements
- Minute–by–minute density measurements: 150 density measurements
- *A priori* state vector: 80 elements comprising 20 OD flows, 45 capacities, and 15 speed–density relationship parameters (3 groups of 5 parameters each).

The transition fractions that were estimated during the off–line calibration were used for the OD flows. The degree of the autoregressive process for the OD flows was found to be equal to two. For the supply parameters, a degree of one was used for the autoregressive process. Furthermore, variance/covariance matrices were estimated based on the output of the off–line calibration (Ashok, 1996, Balakrishna et al., 2005).

The period of analysis comprises six intervals of fifteen minutes (i.e. from 16:15 to 17:45). A warm–up period of 75 minutes (from 15:00 to 16:15) is used to ensure that the network is adequately loaded.

5.2 Results

Tables 1 and 2 summarize the obtained fit for estimated and predicted speeds and counts. The *base* row provides the performance when only demand parameters are estimated on–line. The next three rows show the results obtained when demand and supply parameters are jointly estimated (each row corresponds to one of the considered algorithms). RMSN values are provided, as well as percent improvement over the base case; in particular, the results for the cases where demand and supply parameters are estimated jointly are shown as percent improvement over the *base* case (i.e. when only demand parameters are estimated on–line).

	Estimated		One-step pred.		Two-step pred.		Three-step pred.	
Algorithm	RMSN	(%)	RMSN	(%)	RMSN	(%)	RMSN	(%)
Base	0.1286	---	0.1540	---	0.1666	---	0.1905	---
EKF	0.1039	(19.2%)	0.1318	(14.4%)	0.1550	(7.0%)	0.2008	(-5.4%)
LimEKF	0.1091	(15.1%)	0.1321	(14.2%)	0.1702	(-2.2%)	0.2036	(-6.9%)
UKF	0.1293	(-0.6%)	0.1505	(2.3%)	0.1756	(-5.4%)	0.2221	(-16.6%)

Table 1. Case study results (counts)

These results indicate that the joint calibration of demand and supply parameters can improve the ability of the system to accurately estimate and predict the traffic conditions. The EKF algorithm exhibits the best performance with considerable improvements in estimation and prediction accuracy (except for three–step predicted counts).

	Estimated		One-step pred.		Two-step pred.		Three-step pred.	
Algorithm	RMSN	(%)	RMSN	(%)	RMSN	(%)	RMSN	(%)
Base	0.1266	---	0.1494	---	0.1448	---	0.1494	---
EKF	0.1107	(12.6%)	0.1209	(19.1%)	0.1303	(10.0%)	0.1331	(10.9%)
LimEKF	0.1121	(11.5%)	0.1249	(16.4%)	0.1346	(7.0%)	0.1362	(8.8%)
UKF	0.1156	(8.7%)	0.1249	(16.4%)	0.1315	(9.2%)	0.1346	(9.9%)

Table 2. Case study results (speeds)

A small decrease in performance (compared to the EKF) —but still an improvement— is obtained when the LimEKF algorithm is used. It is interesting to note that while the LimEKF algorithm has order(s) of magnitude lower computational complexity (than the EKF or UKF algorithms), it still provides a significant improvement over the base case. Improvements of more than 10% are obtained in estimated and one–step predicted speeds and counts. Furthermore, the LimEKF algorithm provides a 7% improvement in two–step predicted speeds and close to a 9% improvement in three–step predicted speeds. However, there is a small deterioration (-2.2%) in the two–step predicted counts and almost a 7% deterioration in three–step predicted counts.

The UKF algorithm seems to have the least desirable performance in this application. While in general this algorithm provides improvement over the base case, its two–step and three–step predicted counts deteriorate (-5.4% and -16.6). Furthermore, (with the exception of two–step and three–step predicted speeds) this algorithm is generally outperformed by the LimEKF, which has vastly better computation properties.

One observation that is applicable to all algorithms is that the approach shows a superior performance in prediction of speeds over prediction of counts. The source of this phenomenon may be traced to the surveillance data that have been used for this case study. In particular, sensor counts have been aggregated in 15-minute intervals, while minute–by–minute speeds and densities have been used (resulting in a larger number of observations and hence a larger "weight" on these measurements). Counts have been aggregated to 15–minute intervals in order to be as compatible and comparable as possible with the base case where only OD flows are estimated on–line. For more details on the OD estimation procedure used in DynaMIT, cf. Antoniou et al. (1997).

The third measure of effectiveness (besides the fit of speed and counts) is the computational complexity of each algorithm. As discussed above, the computational performance of the on–line calibration approach for a simulation–based DTA system is determined largely by the number of function evaluations required. The EKF and the UKF algorithms require $2n$ and $2n+1$ function evaluations (and thus runs of the simulator) respectively (where n is the dimension of the state vector). Therefore, the two algorithms have similar computational requirements (converging as the dimension of the problem increases).

The LimEKF algorithm, on the other hand, requires a single function evaluation irrespective of the dimension of the problem. Therefore, the computational performance of this algorithm is vastly superior to that of the other two algorithms (EKF and UKF). The *constant* computational requirements of the LimEKF algorithm (i.e. the fact that a single function evaluation is required irrespective of the application) makes it easy to obtain some further insight into the scalability of the approach to larger networks.

For example, assuming that an estimation interval of fifteen minutes is used, this approach can be used for any application (i.e. combination of DTA system and network) that allows for one function evaluation (for the on–line calibration) and another run of the simulator (for the prediction of the state using the on–line calibrated parameters within that time–frame). A related observation is that using the LimEKF and the on–line calibration approach to jointly estimate all inputs and model parameters is computationally equivalent to the base case (i.e. only using OD estimation to calibrate OD flows on–line).

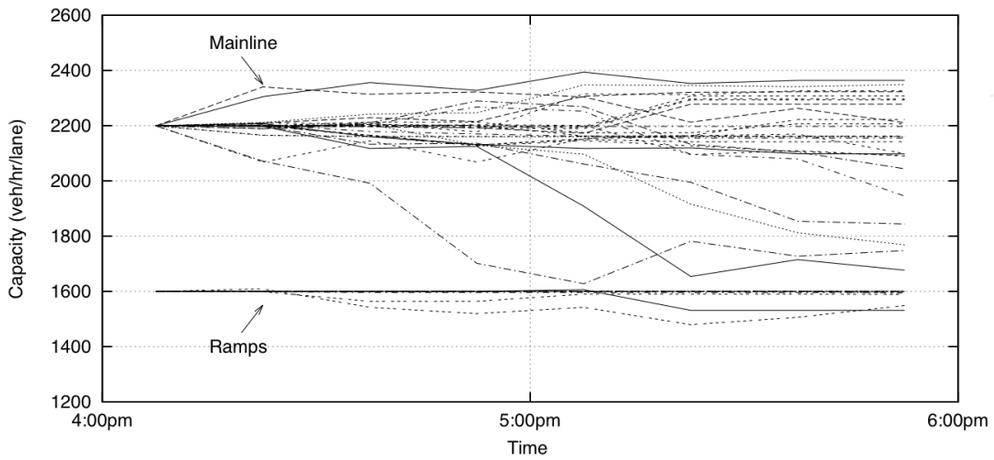


Fig. 4. Estimated capacities (EKF)

The impact of the on–line calibration on the parameters is discussed next, using the results from the EKF algorithm. Fig. 4 shows the estimated capacities for all segments over time. The capacity in several mainline segments has been increased (from the original 2200 vehicles per hour per lane), sometimes close to 2400 vehicles per hour per lane. A capacity of 2400 veh/hr/lane for a motorway such as M27 is reasonable. The capacity for a few mainline segments, which include weaving and/or merging, is reduced to less than 1800 vehicles per hour per lane. The location of these four segments with reduced on–line calibrated capacity is indicated in Fig. 5. The distribution of the other segment capacities is fairly uniform around the off–line calibrated mean capacity (obtained through the general guidelines of the HCM, which could not capture the subtle –geometrical or other– variations among the segments). On–line calibration, on the other hand, is able to estimate individual capacities for each segment, based on the traffic dynamics of these segments, thus capturing these variations. Ramp capacity is generally stable, with the exception of three (out of sixteen in total) ramp segments, in which small decreases (of less than 125 vehicles per hour per lane) are observed.

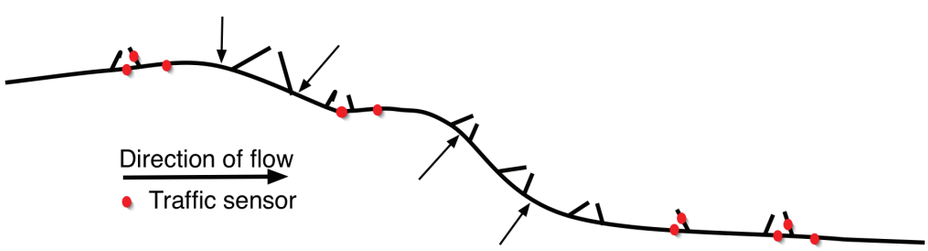
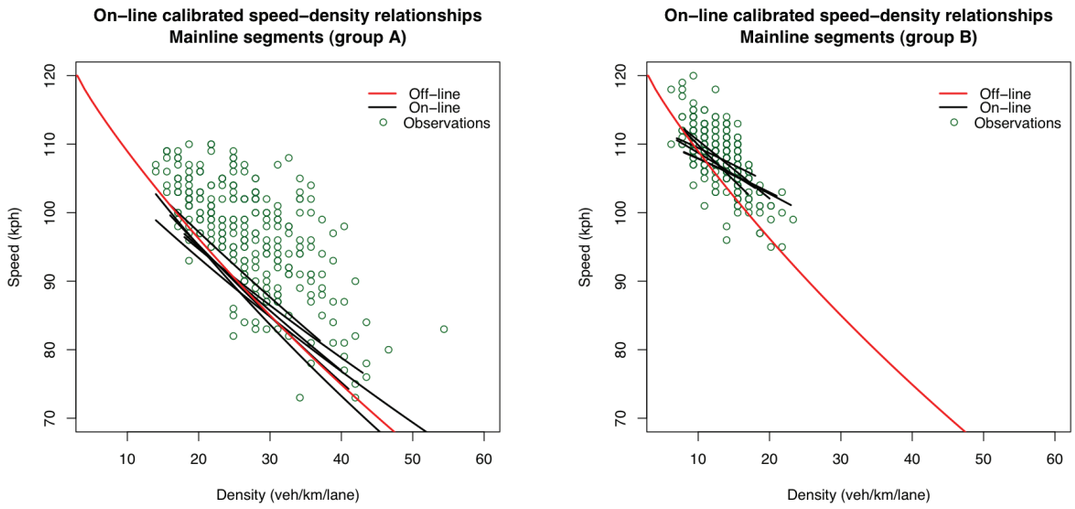


Fig. 5. Mainline segments with lower capacities (due to weaving and merging)

Fig. 6 and 7 show the evolution of the on–line calibrated speed–density relationships over time for mainline segments and ramp segments respectively. Each curve in these figures corresponds to a time interval, for which the on–line calibration was performed. One general observation is that on the day, for which the on–line calibration took place, the observed speeds and densities were higher than those observed during the off-line calibration (as indicated by the off–line calibrated speed–density relationship, also shown in these figures). The on–line calibrated speed–density relationships have changed in a way that captures this behavioral shift, indicating that the on–line calibration performs as intended. This shift is clearer for ramp segments (Fig. 7) and the second mainline segment group (Fig. 6(b)), while smaller changes (still in the right direction) are obtained for the other group of mainline segments (Fig. 6(a)).



(a) Group A (b) Group B

Fig. 6. On–line calibrated speed–density relationships (Mainline segments – EKF)

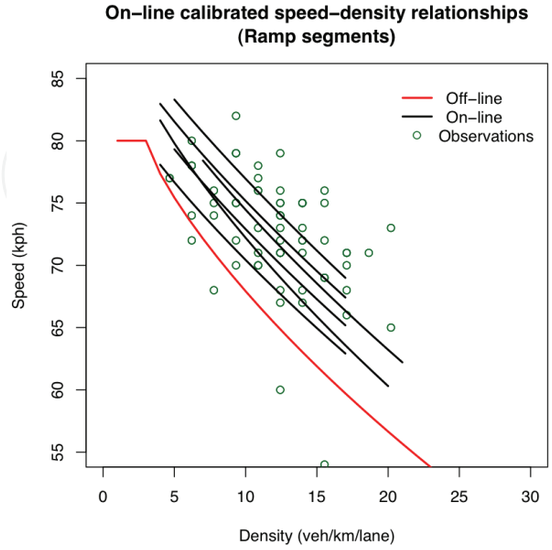


Fig. 7. On–line calibrated speed–density relationships (Ramp segments – EKF)

6. Conclusion

An on–line calibration approach for dynamic traffic assignment systems has been developed. The approach is general and flexible and makes no assumptions on the type of the DTA system, the models or the data that it can handle. Therefore, it is applicable to a wide variety of tools including simulation–based and analytical, as well as microscopic and macroscopic models.

The objective of the on–line calibration approach is to introduce a systematic procedure that will use the available data to *steer* the model parameters to values closer to the *realized* ones. The output of the on–line calibration is therefore a set of parameter values that —when used as input for traffic estimation and prediction— minimizes the discrepancy between the simulated (estimated and predicted) and the observed traffic conditions. The scope of the on–line calibration is neither to duplicate nor to substitute for the off–line calibration process. Instead, the two processes are complementary and synergistic in nature.

The on–line calibration problem is formulated as a state–space model. State–space models have been extensively studied and efficient algorithms have been developed, such as the Kalman Filter for linear models. Because of the non–linear nature of the on–line calibration formulation, modified Kalman Filter methodologies have been presented. The most straightforward extension is the Extended Kalman Filter (EKF), in which optimal quantities are approximated via first order Taylor series expansion (linearization) of the appropriate equations. The Limiting EKF is a variation of the EKF that eliminates the need to perform the most computationally intensive steps of the algorithm on–line. The use of the Limiting EKF provides dramatic improvements in terms of computational performance. The Unscented Kalman Filter (UKF) is an alternative filter that uses a deterministic sampling approach. The computational complexity of the UKF is of the same order as that of the EKF.

Empirical results suggest that joint on–line calibration of demand and supply parameters can improve estimation and prediction accuracy of a DTA system. While the results obtained from this real network application are promising, they should be validated in further empirical studies. In particular, the scalability of the approach to larger, more complex networks needs to be investigated.

The results also suggest that —in this application— the EKF has more desirable properties than the UKF (which may be expected to have superior performance over the EKF), while the UKF seems to perform better in terms of speeds than in terms of counts. Other researchers have also encountered situations where the UKF does not outperform the EKF, e.g. LaViola, J. J., Jr. (2003) and van Rhijn et al. (2005).

The Limiting EKF provides accuracy comparable to that of the best algorithm (EKF), while providing order(s) of magnitude improvement in computational performance. Furthermore, the LimEKF algorithm is that it requires a single function evaluation irrespective of the dimension of the state vector (while the computational complexity of the EKF and UKF algorithms increases proportionally with the state dimension). This property makes this an attractive algorithm for large–scale applications.

7. References

Antoniou, C., Ben-Akiva, M., Bierlaire, M., and Mishalani, R. (1997). Demand simulation for dynamic traffic assignment. In Proceedings of the 8th IFAC Symposium on Transportation Systems, Chania, Greece.

Antoniou, C., Ben-Akiva, M., and Koutsopoulos, H. N. (2004). Incorporating automated vehicle identification into origin–destination estimation. Transportation Research Record: Journal of the Transportation Research Board, 1882:37–44. Washington, D.C.

Antoniou, C., Ben-Akiva, M., and Koutsopoulos, H. N. (2005). On–line calibration of traffic prediction models. Transportation Research Record: Journal of the Transportation Research Board, 1934:235–245.

Ashok, K. (1996). Estimation and Prediction of time-dependent Origin-Destination Flows. PhD thesis, Massachusetts Institute of Technology.

Ashok, K. and Ben-Akiva, M. (1993). Transportation and Traffic Theory, chapter Daganzo, C., editor, Dynamic O-D matrix estimation and prediction for real-time traffic management systems, pages 465–484. Elsevier Science Publishing.

Ashok, K. and Ben-Akiva, M. (2000). Alternative approaches for real-time estimation and prediction of time-dependent origin–destination flows. Transportation Science, 34(1):21–36.

Ashok, K. and Ben-Akiva, M. (2002). Estimation and prediction of time-dependent origin-destination flows with a stochastic mapping to path flows and link flows. Transportation Science, 36(2):184–198.

Balakrishna, R. (2002). Calibration of the demand simulator in a dynamic traffic assignment system. Master's thesis, Massachusetts Institute of Technology.

Balakrishna, R., Koutsopoulos, H. N., and Ben-Akiva, M. (2005). H. S. Mahmassani, ed., 16th International Symposium on Transportation and Traffic Theory, chapter Calibration and validation of Dynamic Traffic Assignment systems, pages 407–426. University of Maryland, College Park. ISBN: 0-08-044680-9.

Ben Aissa, A., Sau, K., El Faouzi, N.-E., and de Moyzon, O. (2006). Sequential monte-carlo traffic estimation for intelligent transportation systems: Motorway travel time prediction application. In Proceedings of 2nd ISTS Symposium. Lausanne, Switzerland.

Ben-Akiva, M., Bierlaire, M., Koutsopoulos, H. N., and Mishalani, R. (2002). Gendreau, M. and Marcotte, P., editors, Transportation and network analysis: current trends, chapter Real-time simulation of traffic demand-supply interactions within DynaMIT, pages 19–36. Kluwer Academic Publishers. Miscellenea in honor of Michael Florian.

Ben-Akiva, M., DePalma, A., and Kaysi, I. (1991). Dynamic network models and driver information systems. Transportation Research A, 25(5):251–266.

Bierlaire, M. and Crittin, F. (2004). An efficient algorithm for real–time estimation and prediction of dynamic OD tables. Operations Research, 52(1):116–127.

Boel, R. and Mihaylova, L. (2006). A compositional stochastic model for real time freeway traffic simulation. Transportation Research Part B, 40:319–334.

Chui, C. K. and Chen, G. (1999). Kalman Filtering with Real-Time Applications. Springer-Verlag.

Crittin, F. (2003). New algorithmic methods for real–time transportation problems. PhD thesis, Ecole Polytechnique Federale de Lausanne.

Crittin, F. and Bierlaire, M. (2003). A generalization of secant methods for solving non-linear systems of equations. In Proceeding of the 3rd Swiss Transportation Research Conference, Ascona, Switzerland. http://www.strc.ch/Paper/crttin 6.pdf.

Doan, D. L., Ziliaskopoulos, A., and Mahmassani, H. (1999). On-line monitoring system for real-time traffic management applications. Transportation Research Record, 1678:142–149.

Gelb, A., editor (1974). Applied Optimal Estimation. M.I.T. Press.

Golub, G. H. and van Loan, C. F. (1996). Matrix Computations. Johns Hopkins Series in the Mathematical Sciences. Johns Hopkins University Press, 3rd edition.

HCM (2000). Highway Capacity Manual. Transportation Research Board, Washington, D.C.

He, R., Miaou, S., Ran, B., and Lan, C. (1999). Developing an on-line calibration process for an analytical dynamic traffic assignment model. In Proceedings of the 78th Annual Meeting of the Transportation Research Board, Washington, D.C.

Huynh, N., Mahmassani, H., and Tavana, H. (2002). Adaptive speed estimation using transfer function models for real-time dynamic traffic assignment operation. In Proceedings of the 81st Annual Meeting of the Transportation Research Board.

Julier, S. and Uhlmann, J. (1997). A new extension of the Kalman filter to nonlinear systems. In Proceedings of Aerosense: The 11th International Symposium of Aerospace/Defense Sensing, Simulation and Controls, Orlando, FL., volume Multi Sensor Fusion, Tracking and Resource Management II.

Julier, S. and Uhlmann, J. (1996). A general method for approximating nonlinear transformations of probability distributions. Technical report, Robotics Research Group, Department of Engineering Science, University of Oxford, Oxford, OX1 3PJ United Kingdom.

Julier, S. J., Uhlmann, J. K., and Durrant-Whyte, H. (1995). A new approach for filtering nonlinear systems. In Proceedings of the American Control Conference, pages 1628–1632.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. Journal of Basic Engineering (ASME), 82D:35–45.

LaViola, J. J., Jr. (2003). A comparison of unscented and extended Kalman filtering for estimating quaternion motion. In Proceedings of the 2003 American Control Conference, pages 2435–2440.

Liu, H., van Lint, H., van Zuylen, H., and Zhang, K. (2006). Two distinct ways of using Kalman filters to predict urban arterial travel time. In Proceedings of the IEEE Intelligent Transport Systems Conference (ITSC), pages 845–850, Toronto, Canada.

Mahmassani, H. S. (2001). Dynamic network traffic assignment and simulation methodology for advanced system management applications. Networks and Spatial Economics, 1(3):267–292.

Peeta, S. and Bulusu, S. (1999). Generalized singular value decomposition approach for consistent on–line dynamic traffic assignment. Transportation Research Record, 1667:77–87.

Peeta, S. and Yu, J. W. (2004). Adaptability of a hybrid route choice model to incorporating driver behavior dynamics under information provision. IEEE Transactions on Systems, Man and Cybernetics — Part A: Systems and Humans, 34(2):243–256.

Peeta, S. and Yu, J. W. (2005). A hybrid model for driver route choice incorporating en–route attributes and real–time information effects. Networks and Spatial Economics, 5(1):21–40.

Peeta, S. and Yu, J. W. (2006). Behavior–based consistency–seeking models as deployment alternatives to dynamic traffic assignment models. Transportation Research Part C, 14:114–138.

Qin, X. and Mahmassani, H. (2004). Adaptive calibration of dynamic speed-density relations for online network traffic estimation and prediction applications. In Proceedings of the 83rd Annual Meeting of the Transportation Research Board, Washington, D.C.

Sorenson, H. W., editor (1985). Kalman Filtering: Theory and Application. IEEE Press, New York.

Tavana, H. and Mahmassani, H. (2000). Estimation and application of dynamic speed-density relations by using transfer function models. Transportation Research Record, 1710:47–57.

van Arem, B. and van der Vlist, M. J. M. (1992). An on–line procedure for estimating current capacity. Technical Report INRO-VVG 1991-17, TNO Institute of Spatial Organization, Delft.

van der Merwe, R., Doucet, A., de Freitas, N., and Wan, E. (2000). The unscented particle filter. Technical Report CUED/F-INFENG/TR 380, Cambridge University Engineering Department.

van Lint, H., Hoogendoorn, S. P., and van Zuylen, H. J. (2002). State space neural networks for freeway travel time prediction. Artificial Neural networks - ICANN 2002 Lecture Notes in Computer Science, 2415:1043–1048.

van Lint, J., Hoogendoorn, S. P., and van Zuylen, H. J. (2005). Accurate freeway travel time prediction with state–space neural networks under missing data. Transportation Research Part C–Emerging Technologies, 13(5–6):347–369.

van Lint, J. W. C. (2006). Incremental and online learning through extended Kalman filtering with constraint weights for freeway travel time prediction. In Proceedings of the IEEE Intelligent Transport Systems Conference (ITSC), pages 1041–1046, Toronto, Canada.

van Rhijn, A., van Liere, R., and Mulder, J. (2005). An analysis of orientation prediction and filtering methods for VR/AR. In Proceedings of the IEEE Virtual Reality Conference 2005, pages 67–74.

Wan, E. and van der Merwe, R. (2000). The unscented Kalman filter for nonlinear estimation. In Proceedings of IEEE Symposium (AS-SPCC). Lake Louise, Alberta, Canada.

Wan, E., van der Merwe, R., and Nelson, A. T. (2000). Dual estimation and the unscented transformation. In Solla, S. A., Leen, T. K., and Mueller, K.-R., editors, Advances in Neural Information Processing Systems 12, pages 666–672. MIT Press.

Wang, Y. and Papageorgiou, M. (2005). Real–time freeway traffic state estimation based on extended Kalman filter: A general approach. Transportation Research Part B, 39:141–167.

Wang, Y., Papageorgiou, M., and Messmer, A. (2007). Real–time freeway traffic state estimation based on extended Kalman filter: A case study. Transportation Science, 41:167–181.

Zhou, X. and Mahmassani, H. S. (2004). A structural state space model for real–time origin–destination demand estimation and prediction in a day–to–day updating framework. In Proceedings of the 83rd Annual Meeting of the Transportation Research Board, Washington, D.C.

Kalman Filter

Edited by Vedran Kordic

ISBN 978-953-307-094-0
Hard cover, 390 pages
Publisher InTech
Published online 01, May, 2010
Published in print edition May, 2010

The Kalman filter has been successfully employed in diverse areas of study over the last 50 years and the chapters in this book review its recent applications. The editors hope the selected works will be useful to readers, contributing to future developments and improvements of this filtering technique. The aim of this book is to provide an overview of recent developments in Kalman filter theory and their applications in engineering and science. The book is divided into 20 chapters corresponding to recent advances in the filed.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen