Proceedings of the 2007 IEEE
Intelligent Transportation Systems Conference
Seattle, WA, USA, Sept. 30 - Oct. 3, 2007



# An efficient non-linear Kalman filtering algorithm using simultaneous perturbation and applications in traffic estimation and prediction

Constantinos Antoniou, *Member, IEEE*, Haris N. Koutsopoulos, and George Yannis

*Abstract*— **The Extended Kalman Filter, a well-established and straightforward extension of the Kalman filter, requires a computationally intensive linearization step. In this paper, the use of the simultaneous perturbation is proposed for the computation of the gradient in a far more efficient way than the usual numerical derivatives. The resulting algorithm is applied to the problem of on-line calibration of traffic dynamics models and empirical results are presented. The use of the simultaneous perturbation gradient approximation provides significant improvement over the base case, and comparable results to those obtained by the more computationally intensive finite difference gradient approximation.**

## I. Introduction

The original Kalman filter theory applies to linear systems. However, since many 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. The Iterated EKF (IEKF) method attempts to improve upon EKF, by using the current estimate of the state vector to linearize the measurement equation in an iterative mode.

The different methods have different performance characteristics in terms of computational effort and accuracy of the results [1, 2]. The linearization step of the gradient computation usually dominates the computational performance of these algorithms. In this paper, the use of the simultaneous perturbation [3] for the computation of the gradient is considered and compared to the commonly used numerical derivative using finite differences.

Simultaneous perturbation has been considered as a component in optimization frameworks. For example, simultaneous perturbation was combined with dynamic tunneling for training single hidden layer feedforward networks [4]. In another example, Alessandri and Parisini [5] present a tuning methodology for complex large-scale models, based on the suitable use of neural networks and specific stochastic-approximation algorithms. Spall [6] presents a resampling-based method for computing the information matrix efficiently, which relies on an efficient technique for estimating the Hessian matrix, introduced as part of the adaptive ("second-order") form of the simultaneous perturbation stochastic approximation (SPSA) optimization algorithm. Chan et al. [7] address the optimisation of particle filtering (or Sequential Monte Carlo, SMC) methods using stochastic approximation. In particular, the Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm is used for the optimisation of an average cost function.

The remainder of this paper is structured as follows. Section II introduces the linear Kalman filter framework while Section III discusses non-linear extensions. Section IV outlines the gradient approximation approach that is combined with the EKF in Section V. Section VI presents an application of the resulting algorithm in a network in Irvine, CA, and Section VII concludes the paper.

## II. Linear Kalman Filter Framework

The Kalman Filter (Algorithm 1) is the optimal minimum mean square error (MMSE) estimator for linear state-space models [1]. While the model formulation of the on-line calibration is not linear (due to the indirect measurement equation), it is still useful to review the basic Kalman Filtering algorithm, since modified Kalman Filter methodologies have been developed for non-linear models.

The Kalman filter provides a recursive solution to the linear optimal filtering problem defined by the following equations

$$\mathbf{X}_{h+1} = \mathbf{F}_h \mathbf{X}_h + \mathbf{w}_h \tag{1}$$

$$\mathbf{Y}_h = \mathbf{H}_h \mathbf{X}_h + \mathbf{u}_h \tag{2}$$

where $\mathbf{X}_h$ is the vector of state variables at interval $h$, $\mathbf{Y}_h$ is the vector of measurements observed during interval $h$, $\mathbf{w}_h$ is assumed to be a vector of zero mean, normal and uncorrelated errors with covariance matrix $\mathbf{Q}_h$ and $\mathbf{u}_h$ is

Manuscript received May 15, 2007. This work was supported in part by the EU Marie Curie project MIRG-CT-2005-036590.

C. Antoniou is with the National Technical University of Athens, 5 Iroon Polytechniou Street, GR-15773 Zografou (phone: +30-210-7721380, fax: +30-210- 7721454, email: antoniou@central.ntua.gr).

H. N. Koutsopoulos is with Northeastern University, 437 Snell Engineering Center, 360 Huntington Ave. Boston, MA 02115 (email: haris@coe.neu.edu).

G. Yannis is with the National Technical University of Athens, 5 Iroon Polytechniou Street, GR-15773 Zografou
(email: geyannis@central.ntua.gr).

assumed to be a vector of zero mean, normal and uncorrelated errors with covariance matrix $\mathbf{R}_h$.

A common way to look at the recursive nature of the Kalman Filter is as a "prediction-correction" approach, due to the two steps of time and measurement update. In words, the main steps of the Kalman Filter are as follows. Suppose that a starting estimate of the state $\mathbf{X}_0$ is available (3), along with an estimate of the initial state variance-covariance matrix $\mathbf{P}_0$ (4). A time update phase makes a *prediction* of the state (5) and its covariance matrix (6) for the next time interval.

**Algorithm 1** Kalman Filter

Initialization

$$\mathbf{X}_{0|0} = \mathbf{X}_0 \quad (3)$$
$$\mathbf{P}_{0|0} = \mathbf{P}_0 \quad (4)$$

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

Time update

$$\mathbf{X}_{h|h-1} = \mathbf{F}_{h-1}\mathbf{X}_{h-1|h-1} \quad (5)$$
$$\mathbf{P}_{h|h-1} = \mathbf{F}_{h-1}\mathbf{P}_{h-1|h-1}\mathbf{F}_{h-1}^{T} + \mathbf{Q}_h \quad (6)$$

Measurement update

$$\mathbf{G}_h = \mathbf{P}_{h|h-1}\mathbf{H}_h^{T}\left(\mathbf{H}_h\mathbf{P}_{h|h-1}\mathbf{H}_h^{T} + \mathbf{R}_h\right)^{-1} \quad (7)$$
$$\mathbf{X}_{h|h} = \mathbf{X}_{h|h-1} + \mathbf{G}_h\left(\mathbf{Y}_h - \mathbf{H}_h\mathbf{X}_{h|h-1}\right) \quad (8)$$
$$\mathbf{P}_{h|h} = \mathbf{P}_{h|h-1} - \mathbf{G}_h\mathbf{H}_h\mathbf{P}_{h|h-1} \quad (9)$$

**end for**

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. Instrumental in this process is the *Kalman gain* $\mathbf{G}_h$, which is computed as per (7). The state can then be updated (*corrected*) using (8). Similarly, the state covariance is updated using (9).

Further information on the Kalman Filter can be found in many texts, including for example [2], [8], and [9].

## III. Non-linear Kalman filter algorithms

Non-linear models are very common. In the remainder of this section the Extended and Iterated Extended Kalman Filter algorithms are outlined as they apply to the following non-linear state-space model:

$$\mathbf{X}_{h+1} = \mathbf{F}_h\mathbf{X}_h + \mathbf{w}_h \quad (10)$$

$$\mathbf{Y}_h = \mathrm{h}(\mathbf{X}_h) + \mathbf{u}_h \quad (11)$$

The transition equation (10) is assumed to be linear (for example, an autoregressive process). The function **h**(.), however, is more general and therefore is not assumed to be linear.

### A. 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:

$$\mathbf{H}_h = \left.\frac{\partial \mathrm{h}(\mathbf{x}^*)}{\partial \mathbf{x}^*}\right|_{x^* = X_{h|h-1}} \quad (12)$$

The main steps of the Extended Kalman Filter are shown in Algorithm 2. Besides the addition of the linearization step in (17), this algorithm is a straightforward extension of the linear Kalman filter.

**Algorithm 2** Extended Kalman Filter

Initialization

$$\mathbf{X}_{0|0} = \mathbf{X}_0 \quad (13)$$
$$\mathbf{P}_{0|0} = \mathbf{P}_0 \quad (14)$$

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

Time update

$$\mathbf{X}_{h|h-1} = \mathbf{F}_{h-1}\mathbf{X}_{h-1|h-1} \quad (15)$$
$$\mathbf{P}_{h|h-1} = \mathbf{F}_{h-1}\mathbf{P}_{h-1|h-1}\mathbf{F}_{h-1}^{T} + \mathbf{Q}_h \quad (16)$$

Linearization

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

Measurement update

$$\mathbf{G}_h = \mathbf{P}_{h|h-1}\mathbf{H}_h^{T}\left(\mathbf{H}_h\mathbf{P}_{h|h-1}\mathbf{H}_h^{T} + \mathbf{R}_h\right)^{-1} \quad (18)$$
$$\mathbf{X}_{h|h} = \mathbf{X}_{h|h-1} + \mathbf{G}_h\left[\mathbf{Y}_h - \mathbf{h}\left(\mathbf{X}_{h|h-1}\right)\right] \quad (19)$$
$$\mathbf{P}_{h|h} = \mathbf{P}_{h|h-1} - \mathbf{G}_h\mathbf{H}_h\mathbf{P}_{h|h-1} \quad (20)$$

**end for**

Further information on the Extended Kalman Filter can be found in many texts, including for example [8], and [9].

### B. Iterated Extended Kalman Filter

The Iterated Extended Kalman Filter is an extension of the EKF that can provide superior performance. The update step of the EKF involves the linearization of the measurement equation about the present best estimate of the state vector **X**, i.e., $\mathbf{X}_{h|h-1}$, the calculation of the Kalman gain and the update of the state vector and its covariance. However, once this step is completed, a presumably superior estimate $\mathbf{X}_{h|h}$ is available which could then be used to linearize the measurement equation (17) and repeat the update step (18-20). At the end of this iteration an even better estimate of the state vector is available, which can in turn be used to linearize the measurement equation. These iterations could be repeated as many times as deemed necessary. Each of these iterations comprises (17-20).

## IV. Gradient approximation approaches

A key step of the EKF and IEKF is the linearization, i.e. the calculation of the gradient. In cases where analytical

relationships exist, this can be performed analytically (e.g. [14,18]. However, in cases that the measurement equation cannot be represented analytically (e.g. [10, 11]) alternative approaches are required for the calculation of numerical derivatives. Considering the applications in transportation modeling in which such approaches can be applied [10-14], the properties of the gradient approximation approaches include robustness to noisy functions and ability to handle large-scale problems.

The need for solving multivariate optimization problems is pervasive in engineering and stochastic approximations are one of the techniques that have received considerable attention. The Finite Difference Stochastic approximation (FDSA) method computes a search direction by comparing the objective function after perturbing each variable individually [3]. While FDSA is an option with proven performance, it is associated with considerable computational complexity: each gradient calculation involves at least *n*+1 function evaluations, where *n* is the number of parameters to be calibrated.

Spall [3,15] proposed an algorithm for the solution of the problem of minimizing an objective function based on a stochastic approximation of the gradient. Simultaneous Perturbation Stochastic Approximation (SPSA) provides significant computational improvements over FDSA by perturbing all variables at once. SPSA requires only two computations of the objective function at a given iterate for estimating the gradient vector, irrespective of the number *n* of parameters to be calibrated.

Although FDSA is expected to give a more accurate and reliable estimate of the gradient vector at each iteration, the associated computational cost is high. Studies have shown that SPSA and FDSA require a comparable number of iterations to reach the global optimum. The per-iteration computational savings of SPSA thus result in a significantly more efficient algorithm for large-scale applications. Spall shows, through several standard stochastic problems, that SPSA outperforms FDSA in terms of overall convergence speed. The performance of SPSA for the calibration of large-scale traffic simulation models has also been demonstrated in Balakrishna et al. [12,13,16], who also include more details about the algorithm.

## V. SP-EKF: A COMBINATION OF EKF AND SP

In this paper, the use of the gradient approximation of the SPSA algorithm is proposed as an alternative for the linearization step of the EKF. The SPSA algorithm iterates from an initial guess of the optimal $\theta$, using a gradient approximation that depends on the highly efficient "simultaneous perturbation" (SP). The innovation of SP is that it does not require each of the n elements to be perturbed individually, but instead all elements are perturbed simultaneously. As a result, only two function evaluations are required (compared to 2n when a finite difference gradient approximation is used). The perturbation is based on a randomly generated p-dimensional random perturbation vector $\Delta$ , where each of the *n* components of $\Delta$ are independently generated from a zero-mean probability distribution satisfying the conditions in Spall [3]. A practical and suitable approach is the use of a Bernoulli ±1 distribution with probability of ½ for each ±1 outcome. Spall [17] notes that uniform and normal random variables are not allowed for the elements of $\Delta_k$ by the SPSA regularity conditions (as they have infinite inverse moments).

The simultaneous perturbation approximation to the unknown gradient $\hat{g}\left(\hat{\theta}\right)$ is obtained from:

$$\hat{g}\left(\hat{\theta}\right)=\frac{y\left(\hat{\theta}+c\,\Delta\right)-y\left(\hat{\theta}-c\,\Delta\right)}{2c}\begin{bmatrix}\Delta_1^{-1}\\ \Delta_2^{-1}\\ \vdots\\ \Delta_p^{-1}\end{bmatrix} \tag{21}$$

where y(.) are measurements of the loss function based on the perturbation around the current $\hat{\theta}$ , $\Delta_i$ is the *i*th component of the $\Delta$ vector, which may be ± 1 random variables as discussed above, and c is an appropriately selected small positive number. The common numerator in all n components of $\hat{g}\left(\hat{\theta}\right)$ reflects the simultaneous perturbation of all components in $\hat{\theta}$ (in contrast to the component-by-component perturbations in the standard finite-difference approximation.)

The expected value of $\hat{g}\left(\hat{\theta}\right)$ is a biased estimator of the true gradient $g\left(\theta\right)$, i.e. $E\left[\hat{g}\left(\hat{\theta}\right)\right]=g(\theta)+bias$. The bias is small and proportional to $c^2$.

To summarize, in the proposed algorithm the linearization step of the EKF (17), would be replaced by (21).

## VI. APPLICATION

The objective of this application is to demonstrate the feasibility of this approach the context of the on-line calibration of traffic dynamics models. An example of such models is the estimation and prediction of speed based on density measurements. Rather than relying on a single off-line calibrated relation, in this application the model parameters are dynamically re-estimated in each time interval so that the model output would better match the prevailing traffic condition.

### *A. On-line calibration of traffic dynamics models*

The on-line calibration problem is formulated as a state-space model, comprising measurement and transition equations [18]. The transition equation for the state vector can be written in the general form of an autoregressive process. The speed-density relationship provides a (likely

non-linear) measurement equation. However, the problem is under-defined, as the dimension of the state vector (densities plus model parameters) is larger than the number of equations (speed measurements). To overcome this issue (and also introduce additional information into the formulation), it is possible to augment the formulation with two additional sets of measurement equations. First, off-line calibrated values of the parameters of the speed-density relationship can be used as a priori estimates of the actual parameters values. In addition, density measurements (obtained, for example, indirectly from occupancy reported by sensors) can be incorporated directly.

*1) Measurement Equations*

The speed-density relationship provides the first measurement equation, which can be written in the following general form:

$$\mathbf{u}_h^m = \mathrm{h}(\mathbf{\Pi}_h, \mathbf{K}_h) + \mathbf{v}_h \quad (22)$$

where $h$ is the interval of interest, $\mathbf{h}(.)$ is the mapping of densities into speeds, $\mathbf{\Pi}_h$ is the vector of parameters to be calibrated, $\mathbf{K}_h$ is the vector of densities that are used as state variables, and $\mathbf{v}_h$ is a vector of Gaussian, zero-mean, uncorrelated errors. An example of a functional form for the speed-density relationship is:

$$u = u_f \left[ 1 - \left( \frac{\max(0, K - K_{\min})}{K_{jam}} \right)^{\beta} \right]^{\alpha} \quad (23)$$

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. Further examples of speed-density relationships can be found in e.g. [19] and [20].

Density measurements (obtained, for example, indirectly from occupancy reported by sensors) can also be used as a priori estimates of the density state vector:

$$\mathbf{K}_h^m = \mathbf{K}_h + \zeta_h \quad (24)$$

where $\mathbf{K}_h^m$ is the vector of measurements and $\zeta_h$ is a vector of Gaussian, zero-mean, uncorrelated errors. Off-line calibrated values of the parameters of the speed-density relationship can be used as a priori estimates of the actual parameter values:

$$\mathbf{\Pi}_h^m = \mathbf{\Pi}_h + \boldsymbol{\psi}_h \quad (25)$$

where $\boldsymbol{\psi}_h$ is a vector of Gaussian, zero-mean, uncorrelated errors.

*2) Transition Equations*

The transition equation can be written in the general form:

$$\mathbf{X}_{h+1} = \mathrm{f}(\mathbf{X}_h) + \mathbf{w}_h \quad (26)$$

The state vector $\mathbf{X}_h$ comprises the model parameters and densities to be estimated.

The transition equation for the parameters can be represented as an autoregressive process:

$$\mathbf{\Pi}_{h+1} = \sum_{p=h-r}^{h} \mathbf{F}_h^p \mathbf{\Pi}_h + \varphi_h \quad (27)$$

where $r$ is the degree of the autoregressive process and $\mathbf{F}_h^p$ is a matrix of autoregressive factors capturing the contribution of the parameter vector in interval $p$ on the parameter vector in interval $h$. The autoregressive degree, the transition matrices $\mathbf{F}_h^p$, and the error terms $\varphi_h$ are estimated off-line. A detailed description of this process can be found in numerous references (e.g. [21] pp. 56-58 and [22] pp. 54-60) and is outside the scope of this paper. For the special case of $r = 0$, this transition equation reduces to a random walk: $\mathbf{\Pi}_{h+1} = \mathbf{\Pi}_h + \varphi_h$, which is what has been assumed –without loss of generality- in the application in this paper.

Density measurements are usually available in intervals finer than the estimation interval. For example, measurements may be available every 30 seconds or every minute, while estimation of the model parameters may be performed every 5, 10 or 15 minutes. One way to express the transition equation of the density measurements is to construct a finer autoregressive process, in which each element of the state vector depends on the preceding elements. For each element in the density measurement state vector the transition equation could be:

$$k_{t+1} = \sum_{s=t-l}^{t} f_{t+1}^s k_s + \tau_t \quad (28)$$

where $t$ is the measurement time, $l$ is the degree of the autoregressive process, $f_{t+1}^s$ is the autoregressive factor mapping the contribution of $k_s$ on $k_{t+1}$, and $\tau_t$ is a Gaussian, zero-mean error, uncorrelated with the other error terms.

*B. Data and Methodology*

Several days of sensor data from freeway I-405 in Irvine, CA, have been used. Validation of the algorithms involved off-line calibration of the approach with several days of data and subsequent application of the models for different days (not used in the off-line calibration). Morning period (04:00am to 10:00am) data have been used, since this period includes the peak flow for this sensor). Speed and density data are available in 30-second intervals.

The chosen speed-density relationship has been used for the estimation of both off-line and on-line calibrated speeds. First, a priori estimates of the parameters (along with densities) were used for the estimation and prediction of off-line calibrated speeds. A priori estimates of the parameter values are obtained by fitting the speed-density relationship to an initial set of data (three days) using non-linear least squares and are then held constant for the entire period. The accuracy of these off-line calibrated speeds was used as a

baseline reference, from which the benefits obtained from the on-line calibration are measured.

Data from a different day (not used in the off-line calibration) was then used for the on-line calibration of the speed-density relationship parameters for each data-set, using the Extended Kalman Filter modifications presented above, where the finite difference and stochastic perturbation linearization steps have been used. The estimated parameters were then used for the prediction of the parameter values using the transition equation (27). The estimated and predicted speeds were obtained by substituting in (23) the on-line calibrated (estimated/predicted) parameters for each time and the corresponding densities.

The parameter vector would then become

$$\Pi_h = \begin{bmatrix} \alpha_h & \beta_h & K_{\min,h} & K_{jam,h} & u_{f,h} \end{bmatrix}.$$

The application of the EKF and the IEKF requires the differentiation of (23). However, due to the maximizing function, (23) is not continuously differentiable. In order to overcome this issue, the numerical differentiation is applied each time to the regime, in which the evaluation point belongs.

The performance of each algorithm is assessed using the normalized root mean square error (RMSN) of the speeds (used, for example, in [23, 24]):

$$RMSN = \frac{\sqrt{N \sum_N (u - \hat{u})^2}}{\sum_N u} \tag{29}$$

where N is the number of measurements and û denotes estimated (respectively predicted) speeds.

### C. Results

Table I presents the estimation and prediction results obtained by the various alternative algorithms (1Pred denotes one-step ahead results, while 2Pred denotes two-step ahead results). For each case, RMSN values are presented, along with percent improvement relative to the base case (in parentheses below the RMSN values; the offline case is used as the reference case). Offline corresponds to the results obtained when the off-line computed parameter values are used, which is used as the base case. FD-EKF denotes the EKF algorithm when the finite difference method is used for the computation of the gradient in the linearization step, while in the SP-EKF the simultaneous perturbation algorithm has been used for the computation of the gradient in the linearization step. Due to the different scale of the state variables, the magnitude of the elements of vector $\Delta$ has been normalized in the SP gradient approximation step. Iterated EKF has also been run for both gradient computation approaches. The results for FD-IEKF and SP-IEKF presented in Table I correspond to 4 iterations of the IEKF algorithm.

The first observation is that all on-line calibrated sets of parameters result in significant improvements over the base case. As expected, the use of the FD results in larger improvements than the use of the SP for the gradient approximation step for the estimated speeds. Predicted speeds, however, are fit better by the SP-generated speeds. This is a property that should be further investigated empirically and theoretically. The use of the iterated algorithms improves the estimation and prediction of the speeds.

TABLE I
ESTIMATION AND PREDICTION RESULTS

| | Est | 1Pred | 2Pred |
|---|---|---|---|
| Offline | 0.0709 | 0.0733 | 0.0755 |
| FD-EKF | 0.0588 (17%) | 0.0712 (3%) | 0.0713 (6%) |
| SP-EKF | 0.0611 (14%) | 0.0669 (9%) | 0.0676 (10%) |
| FD-IEKF | 0.0573 (19%) | 0.0675 (8%) | 0.0688 (9%) |
| SP-IEKF | 0.0574 (19%) | 0.0671 (8%) | 0.0680 (10%) |

An interesting observation is that the two gradient approximations perform similarly. Considering the vastly different computational requirements of the two approaches, it becomes apparent that the combination of the SP gradient approximation with the IEKF algorithm results in superior accuracy than the FD-EKF (or equivalent accuracy with the FD-IEKF) but with significantly lower computational requirements. Note that each iteration of the FD-EKF requires 2n function evaluations (where n is the number of parameters), while SP-EKF requires only two irrespective of the number of parameters to be estimated. In this particular application, where the number of parameters to be estimated for each interval is 35 (30 densities plus 5 parameters of the speed-density relationship), each FD gradient approximation run requires 70 function evaluations while each SP gradient approximation run only 2 function evaluations. This impact becomes even more pronounced as the cost of running each function evaluation increases, e.g. in the case of simulation functions.

## VII. CONCLUSION

In this paper, the use of the simultaneous perturbation is proposed for the computation of the gradient in a far more efficient way than the usual numerical derivatives. The resulting algorithm is applied to the problem of on-line calibration of traffic dynamics models and empirical results are presented. The use of the SP gradient approximation provides comparable results to those obtained by the more computationally intensive FD algorithm. After four iterations of the iterated EKF algorithm, the SP gradient approximation algorithm provides equivalent results with those obtained with the FD gradient approximation (after the same number of iterations).

The approach should be further studied in terms of its theoretical properties and validated. Furthermore, issues like the scalability, the convergence, the robustness to noise, the performance when the prediction horizon is extended, need to be further investigated empirically.

## ACKNOWLEDGMENT

The authors would like to thank Prof. Wilfred Recker from University of California, Irvine, who provided the data for the case study.

## REFERENCES

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

[2] A. Gelb, editor. Applied Optimal Estimation. M.I.T. Press, 1974.

[3] Spall, J. C. (1992). Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. IEEE Transactions on Automatic Control, 37(3):332–341.

[4] P. Thangavel and T. Kathirvalavakumar. Training feedforward networks using simultaneous perturbation with dynamic tunneling. Neurocomputing 48 (2002) 691–704.

[5] A. Alessandri and T. Parisini. Nonlinear Modeling of Complex Large-Scale Plants Using Neural Networks and Stochastic Approximation. IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans, Vol. 27, No. 6, November 1997

[6] Spall, J. C. (2005). Monte Carlo Computation of the Fisher Information Matrix in Nonstandard Settings. Journal of Computational & Graphical Statistics, Volume 14, Number 4, December 2005, pp. 889-909.

[7] Chan, B.L., Doucet, A., and Tadic, V.B. (2003), "Optimisation of Particle Filters Using Simultaneous Perturbation Stochastic Approximation," Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, Hong Kong, April 2003, vol. VI, pp. 681-684.

[8] H. W. Sorenson, editor. Kalman Filtering: Theory and Application. IEEE Press, New York, 1985.

[9] C. K. Chui and G. Chen. Kalman Filtering with Real-Time Applications. Springer-Verlag, 1999.

[10] C. Antoniou, Online calibration for dynamic traffic assignment, PhD Thesis, Department of Civil and Environmental Engineering, Massachusetts Institute of Technology, September 2004.

[11] C. Antoniou, M. Ben-Akiva, and H. N. Koutsopoulos. Non–linear Kalman Filtering Algorithms for On–line Calibration of Dynamic Traffic Assignment Models. Proceedings of the 9th IEEE Intelligent Transport Systems Conference (ITSC), pp. 833-838, Toronto, Canada, September, 2006.

[12] Balakrishna, R., M. Ben-Akiva and H.N. Koutsopoulos. Time-Dependent Origin-Destination Estimation without Assignment Matrices. Second International Symposium on Transport Simulation, Lausanne, 2006.

[13] Balakrishna, R., M. Ben-Akiva and H.N. Koutsopoulos. Offline Calibration of Dynamic Traffic Assignment: Simultaneous Demand-Supply Estimation. Transportation Research Record: Journal of the Transportation Research Board (forthcoming), 2007.

[14] Y. Wang, M. Papageorgiou, and A. Messmer, "Real–time freeway traffic state estimation based on extended Kalman filter: A case study," Transportation Science, vol. 41, pp. 167–181, May 2007.

[15] Spall, J. C. (1999). Stochastic optimization, stochastic approximation and simulated annealing. In Webster, J. G., editor, Wiley Encyclopedia of Electrical and Electronics Engineering, pages 529–542. Wiley-Interscience.

[16] R. Balakrishna, C. Antoniou, M. Ben-Akiva, H. N. Koutsopoulos and Y. Wen. Calibration of Microscopic Traffic Simulation Models: Methods and Application. Transportation Research Record: Journal of the Transportation Research Board (forthcoming), 2007.

[17] Spall J.C. Implementation of the Simultaneous Perturbation Algorithm for Stochastic Approximation, IEEE Transactions on Aerospace and Electronic Systems, Vol. 34(3), 1998.

[18] C. Antoniou, M. Ben-Akiva, and H. N. Koutsopoulos. On-Line Calibration of Traffic Prediction Models. Transportation Research Record 1934, pp. 235-245, 2005.

[19] H. Greenberg. An analysis of traffic flow. Operations research, 7:79-85, 1959.

[20] G. F. Newell. Nonlinear effects in the dynamics of car following. Operations Research, 9:209-229, 1961.

[21] K. Ashok. Estimation and Prediction of Time-Dependent Origin-Destination Flows. PhD Thesis. Department of Civil and Environmental Engineering, Massachusetts Institute of Technology, Cambridge, MA, 1996.

[22] R. Balakrishna. Calibration of the Demand Simulator in a Dynamic Traffic Assignment System. MS Thesis. Department of Civil and Environmental Engineering, Massachusetts Institute of Technology, Cambridge, MA, 2002.

[23] K. Ashok, and M. Ben-Akiva. Alternative Approaches for Real-Time Estimation and Prediction of Time-Dependent Origin-Destination Flows. Transportation Science, Vol. 34, No. 1, INFORMS, 2000.

[24] K. Ashok, and M. Ben-Akiva. Estimation and Prediction of Time-Dependent Origin-Destination Flows with a Stochastic Mapping to Path Flows and Link Flows. Transportation Science, Vol. 36. No. 2, INFORMS, 2002.