A Stochastic Approach to Online Vehicle State and Parameter Estimation, with Application to Inertia Estimation for Rollover Prevention and Battery Charge/Health Estimation

Prepared by

The Pennsylvania State University

The Pennsylvania State University ❖ University of Maryland
University of Virginia ❖ Virginia Polytechnic Institute and State University ❖ West Virginia University

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
MAUTC-2010-02		

4. Title and Subtitle	5. Report Date
A Stochastic Approach to Online Vehicle State and Parameter Estimation, with Application to Inertia Estimation for Rollover Prevention and Battery Charge/Health Estimation	August 31, 2013
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Hosam K. Fathy, Corina Sandu, and Avinash Unnikrishnan	LTI 2013-15

9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
The Pennsylvania State University, The Thomas D. Larson Pennsylvania Transportation Institute, 201 Transportation Research Building, University Park, PA 16802 Virginia Tech, Blacksburg, Virginia West Virginia University, Morgantown, West Virginia	**11. Contract or Grant No.** DTRT07-G-0003

12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
US Department of Transportation Research & Innovative Technology Admin UTC Program, RDT-30 1200 New Jersey Ave., SE Washington, DC 20590	Final April 1, 2011 – January 31, 2013
	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

This report summarizes research conducted at Penn State, Virginia Tech, and West Virginia University on the development of algorithms based on the generalized polynomial chaos (gpc) expansion for the online estimation of automotive and transportation system parameters from experimental data. The authors used gpc estimation for determining the state of charge and state of health of a lithium-ion battery in real time, and also for the determination of road-vehicle inertial parameters in real time. The overarching goals in these two applications are to improve vehicle safety through better battery diagnostics and better inertia estimate-based rollover prevention. The authors performed experimental validation studies for both of these applications. Finally, the authors began exploring the application of gpc-based estimation for stochastic traffic flow models.

17. Key Words	18. Distribution Statement
Generalized polynomial chaos, vehicle inertia estimation, battery SOC/SOH estimation, stochastic traffic modeling	No restrictions. This document is available from the National Technical Information Service, Springfield, VA 22161

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	80	

TECHNICAL DOCUMENTATION

The material presented in this report is a summary of the following publications by the report's authors. These publications were funded, at least in part, by this MAUTC grant.

1. Aabhas Sharma, 2012, "Combined State of Charge and State of Health Estimation for Lithium-Ion Batteries," M.S. Thesis, Penn State University (supervised by H. K. Fathy).
2. Saeid Bashash and Hosam K. Fathy, "Battery State-of-Health Estimation Using Polynomial Chaos Theory," in preparation for conference submission.
3. Jeremy Kolansky and Corina Sandu, "Generalized Polynomial Chaos-Based Extended Kalman Filter: Improvement and Expansion," in preparation for conference submission.
4. Jeremy Kolansky, Corina Sandu, Theunis Botha, and Schalk Els, "Real-Time Vehicle Parameter Estimation," in preparation for conference submission.

Additional material in preparation for publication by author Avinash Urikrishnan is also included in this report.

Acknowledgments and Disclaimer

The authors thank the Mid-Atlantic Universities Transportation Center (MAUTC) for support of this work. The contents of this report reflect the views of the authors, who are responsible for the facts and accuracy of the information presented herein. This document is disseminated under the sponsorship of the U.S. Department of Transportation's University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.

This Page Intentionally Left Blank

TABLE OF CONTENTS

LIST OF FIGURES

Page

LIST OF FIGURES (CONTINUED)

Page

LIST OF TABLES

Page

PART 1: GPC ESTIMATION USING EXTENDED KALMAN FILTERING

PROBLEM

ABSTRACT

The generalized polynomial chaos (gPC) method for propagating uncertain parameters through dynamical systems (previously developed at Virginia Tech) has been shown to be very computationally efficient. This method seems also to be ideal for real-time parameter estimation when merged with the Extended Kalman Filter (EKF). The resulting technique is shown in the present report for systems in state-space representations, and then expanded to systems in regressions formulations. Due to the way the filter interacts with the polynomial chaos expansions, the covariance matrix is forced to zero in finite time. This problem shows itself as an inability to perform state estimations and causes the parameters to converge to incorrect values for state space systems. In order to address this issue, improvements to the method are implemented and the updated method is applied to both state space and regression systems. The resultant technique shows high accuracy of both state and parameter estimations.

INTRODUCTION

The Generalized Polynomial Chaos Extended Kalman Filter (gPC-EKF), developed by Blanchard [1, 2], is capable of performing parameter estimation in nonlinear systems [3-5]. When the Extended Kalman Filter (EKF) is altered by the Generalized Polynomial Chaos (gPC) expansions, a major flaw in the filter appears which drives the covariance matrix to zero. This prevents state estimation and signal filtering and causes the parameter values to converge to incorrect values. Improvements are implemented that enable the standard state estimation capabilities and improve the parameter convergence.

The gPC-EKF is expanded to regression systems, where it performs as a computationally efficient, derivative free, iterative, real-time non-linear, recursive regression estimator. The gPC-EKF is applied to a non-linear vehicle model and compared to the recursive least squares (RLS) method.

APPROACH

GENERALIZED POLYNOMIAL CHAOS

Generalized Polynomial Chaos is a method that propagates uncertain parameters through the dynamics of a system. The collocation approach for gPC resembles a Monte Carlo approach, where the system is iterated at each time step. However, unlike Monte Carlo, the iteration process is used to approximate integration over the parameter space. This integration process is used to create a continuous map between the parameter space and the system dynamics.

A deterministic system can be described in a state space formulation. This system is a function of time (t)*, the state variables* (x) *and velocities* (v)*, and the system parameters* (p)*. The system can also be affected by process noise (*w*).*

$$\dot{\boldsymbol{x}}_{\boldsymbol{k}} = \boldsymbol{v}_{\boldsymbol{k}} \tag{1}$$

$$\dot{v}_k = F_k(t, x, v; p) + w \tag{2}$$

The gPC method expands each of the parameters in a power series [6, 7]:

$$\dot{\boldsymbol{x}}_{\boldsymbol{k}}^{\boldsymbol{i}} = \boldsymbol{v}_{\boldsymbol{k}}^{\boldsymbol{i}} \tag{3}$$

$$\sum_{\boldsymbol{i=1}}^{\boldsymbol{S}} \dot{\boldsymbol{v}}_{\boldsymbol{k}}^{\boldsymbol{i}}\, \Psi^{\mathbf{i}}(\boldsymbol{\xi}) = \boldsymbol{F}_{\boldsymbol{k}}\left(\boldsymbol{t}, \sum_{\boldsymbol{m=1}}^{\boldsymbol{S}} \boldsymbol{x}^{\boldsymbol{m}}\, \Psi^{\mathbf{m}}(\boldsymbol{\xi}), \sum_{\boldsymbol{m=1}}^{\boldsymbol{S}} \boldsymbol{v}^{\boldsymbol{m}}\, \Psi^{\mathbf{m}}(\boldsymbol{\xi}); \sum_{\boldsymbol{m=1}}^{\boldsymbol{S}} \boldsymbol{p}^{\boldsymbol{m}}\right. \tag{4}$$

The indexes are defined as: $\boldsymbol{i}$ *or* $\boldsymbol{m}$ *are the power series term indexes and* $\boldsymbol{k}$ *is the discreet time index. Each parameter is described by S power series coefficients. These coefficients multiply a set of orthogonal or orthonormal basis functions. The basis functions are chosen to represent different parameter distributions. For the gPC-EKF the most useful is the Legendre polynomials, as they describe a uniform parameter distribution and are more appropriate for implementation with the EKF. The set of Jacobi polynomials for a uniform beta distribution return the same results as the Legendre polynomials.*

These basis functions $(\boldsymbol{\psi})$ *are functions of an n dimensional random variable* $(\boldsymbol{\xi})$*. This random variable has an element for each uncertain parameter, and is used to span the space of the basis functions. Each uncertain parameter* $(\boldsymbol{d})$ *has a distribution:*

$$\boldsymbol{p}_{\boldsymbol{d}} = \sum_{\boldsymbol{i}} \boldsymbol{p}_{\boldsymbol{d}}^{\boldsymbol{i}}\, \boldsymbol{\psi}_{\boldsymbol{d}}^{\boldsymbol{i}}(\xi_{\boldsymbol{d}}) \tag{5}$$

This gives the states a multivariable distribution, because uncertain parameters cause the states to become uncertain:

$$\boldsymbol{x}_{\boldsymbol{n}} = \sum_{\boldsymbol{i}} \boldsymbol{x}_{\boldsymbol{n}}^{\boldsymbol{i}}\, \boldsymbol{\Psi}^{\boldsymbol{i}}(\boldsymbol{\xi}) \tag{6}$$

The multidimensional basis function is created through a tensor product of the single dimensional basis functions:

$$\Psi^j(\xi_1 \dots \xi_d) = \prod_{k=1}^{d} \psi_k^i(\xi_k), \quad (7)$$

$$j = 1, 2, \dots S, \quad i = 1, 2, \dots PO$$

S is calculated from the polynomial order (PO) and the number of uncertain parameters (d) as:

$$S = \frac{(d + PO)!}{d!\, PO!} \quad (8)$$

For the collocation technique the power series coefficients are solved through an iteration process. A matrix of collocation points ($\boldsymbol{\mu}$) are chosen with the $\boldsymbol{i_{th}}$ iteration as the vector $\boldsymbol{\mu^i}$:

$$\boldsymbol{\mu^i} = \left[\mu_1^i \dots \mu_d^i\right]^T \textbf{ for all } 1 \le i \le Q \quad (9)$$

The system dynamics are re-written as:

$$\dot{x}^i = v^i \quad (10)$$

$$\sum_{j=1}^{S} \dot{v}^j \, \Psi^j(\boldsymbol{\mu^i}) = F\left(t, \sum_{m=1}^{S} x^m \, \Psi^m(\boldsymbol{\mu^i}), \sum_{m=1}^{S} v^m \, \Psi^m(\boldsymbol{\mu^i}); \sum_{m=1}^{S} p^m \right. \quad (11)$$

$$1 \le i \le Q$$

The set of values of $\Psi^j(\boldsymbol{\mu^i})$ can be written in matrix form as $A_{i,j} = \Psi^j(\boldsymbol{\mu^i})$, $1 \le j \le s$, $1 \le i \le Q$. The system dynamics equations are written in their final form:

$$\sum_{j=1}^{S} \dot{v}^j \, \Psi^j(\boldsymbol{\mu^i}) = F\left(t, \sum_{m=1}^{S} A_{i,j} \, x^m, \sum_{m=1}^{S} A_{i,j} \, v^m; \sum_{m=1}^{S} A_{i,j} \, p^m \right), \quad (12)$$

$$1 \le i \le Q$$

The system equations for the iteration are written as:

$$\boldsymbol{X^i(t) = \sum_{j=1}^{Q} A_{i,j}\, x^j(t)} \tag{13}$$

Then equation (12) becomes:

$$\boldsymbol{\dot{V}^i(t) = F(t, Y^i, V^i : P^i), \qquad 1 \le i \le Q} \tag{14}$$

The power series coefficients are recovered using the collocation matrix:

$$\boldsymbol{x^i(T) = \sum_{j=1}^{Q} (A^{\#})_{i,j}\, X^j(T)} \tag{15}$$

GPC-EKF

The EKF equations are defined as [8]:

$$\boldsymbol{K_k = M_k\, H_k^T \left(H_k\, M_k\, H_k^T + R_k\right)^{-1}} \tag{16}$$

$$\boldsymbol{M_k = \Phi_k\, P_{k-1}\, \Phi_k + Q_k} \tag{17}$$

$$\boldsymbol{P_k = (I - K_k\, H_k)\, M_k} \tag{18}$$

K *is the Kalman update coefficient,* ***P*** *is the forecast covariance matrix,* ***M*** *is the updated covariance matrix,* ***R*** *is the noise covariance matrix,* ***Q*** *is the process noise covariance matrix,* ***H*** *is the linearized observation matrix, and* ***Φ*** *is the linearized functional form matrix.*

The Kalman state update is expressed as:

$$\boldsymbol{x_k^u = x_k^f + K_k\, (z_k - H_k\, x_k^f)} \tag{19}$$

$\boldsymbol{x_k^u}$ *is the updated state vector at time* $\boldsymbol{k}$*,* $\boldsymbol{x_k^f}$ *is the initial state estimate, and* $\boldsymbol{z_k}$ *is the sensor measurement.*

These equations use the system's functional form to create the covariance matrix. The gPC gives a trivial method of extracting the covariance matrix, however:

$$\boldsymbol{cov}(\boldsymbol{x}_{d,k}, \boldsymbol{x}_{j,k}) = \sum_{i=2}^{S} \boldsymbol{x}_{d,k}^{i}\, \boldsymbol{x}_{j,k}^{i} \qquad (20)$$

The parameters are formally added to the state vector:

$$\dot{\boldsymbol{x}}_{\boldsymbol{n}} = \begin{bmatrix} \dot{\boldsymbol{x}}_{\boldsymbol{n}} \\ \dot{\boldsymbol{p}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{w} \\ \mathbf{0} \end{bmatrix} \qquad (21)$$

And the state update equations are used to update the states and the parameters. The Kalman update equations are expanded to the gPC system where they modify the power series coefficients.

$$\boldsymbol{x}_{\boldsymbol{k}}^{\boldsymbol{u,i}} = \boldsymbol{x}_{\boldsymbol{k}}^{\boldsymbol{f,i}} + \boldsymbol{K}_{\boldsymbol{k}} \left(\boldsymbol{z}_{\boldsymbol{k}} \boldsymbol{\delta}(\boldsymbol{i} - \mathbf{1}) - \boldsymbol{H}_{\boldsymbol{k}}\, \boldsymbol{x}_{\boldsymbol{k}}^{\boldsymbol{f,i}} \right) \qquad (22)$$

$$\boldsymbol{K}_{\boldsymbol{k}} = \boldsymbol{cov}\, \boldsymbol{H}^{\boldsymbol{T}} \left(\boldsymbol{R}_{\boldsymbol{k}} + \boldsymbol{H}\, \boldsymbol{cov}\, \boldsymbol{H}^{\boldsymbol{T}} \right)^{-1} \qquad (23)$$

The measured signal z_k is expanded in the same power series as the states and parameters, but the measured signal is assumed to have no uncertainty and expands to only the first time in the power series. The rest of the coefficients are zero. In gPC the first term in the power series expansion is the mean and the rest of the terms sum to the variance. The initial creation of this filter is found in Blanchard's paper [2]. δ is the delta function.

METHODOLOGY

GPC-EKF IMPROVEMENTS FOR STATE SPACE SYSTEMS

The problem in the previous section is that the measured signal is expanded to the power series without any higher-order terms. The system drives the first-order terms in the expansions to the measured signal while driving the higher-order terms to zero. Since the covariance matrix is calculated from only the higher-order terms, this drives the covariance to zero. Blanchard initially diagnosed this problem but attributed it to the power series truncation [9].

For steady state estimation and filtering, a small diagonal matrix is added to the covariance matrix:

$$\boldsymbol{cov}(\boldsymbol{x_k}, \boldsymbol{x_k}) \tag{24}$$
$$= \begin{bmatrix} \sum_{i=2}^{S} \boldsymbol{x_{1,k}^i}\, \boldsymbol{x_{1,k}^i} & \cdots & \sum_{i=2}^{S} \boldsymbol{x_{1,k}^i}\, \boldsymbol{x_{n,k}^i} \\ \vdots & \ddots & \vdots \\ \sum_{i=2}^{S} \boldsymbol{x_{n,k}^i}\, \boldsymbol{x_{1,k}^i} & \cdots & \sum_{i=2}^{S} \boldsymbol{x_{n,k}^i}\, \boldsymbol{x_{n,k}^i} \end{bmatrix}$$
$$+ \boldsymbol{c} \begin{bmatrix} \mathbf{1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{1} \end{bmatrix}$$

This introduces a new problem, because the added covariance information is independent of the actual system dynamics and parameters. If this added matrix is too large, it will de couple the observed states, the unobserved states and the parameter estimation process. The parameter $\boldsymbol{c} \sim \mathbf{1e} - \mathbf{4}$.

Therefore, at each step forward the parameter values have their distributions expanded. This distribution should remain small, so as not to enable too fast of a Kalman update coefficient, but large enough to be more significant than the added diagonal matrix.

$$\boldsymbol{X_K^j} = \boldsymbol{X_k^j} + \begin{bmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \boldsymbol{c_1} * \boldsymbol{p_1^2} * \boldsymbol{\mu_1^i} \\ \vdots \\ \boldsymbol{c_d} * \boldsymbol{p_d^2} * \boldsymbol{\mu_d^i} \end{bmatrix} \tag{25}$$

The parameters $\boldsymbol{c_1} \ldots \boldsymbol{c_d}$ *are coefficients for each individual parameter to be estimated. These values are also small, but must be tuned for the specific parameters.*

Starting the system with a distribution in the states creates a significant amount of information for the covariance matrix. This information will converge to zero, and so it does not impact the steady state solution of the filter, but significantly improves convergence for incorrect initial state estimates. For the first integration, a random noise vector is fed to each state:

$$\boldsymbol{X_0^j} = \boldsymbol{X_0^j} + \begin{bmatrix} \boldsymbol{N_1} \\ \vdots \\ \boldsymbol{N_n} \\ \boldsymbol{c_1} * \boldsymbol{p_1^2} * \boldsymbol{\mu_1^i} \\ \vdots \\ \boldsymbol{c_d} * \boldsymbol{p_d^2} * \boldsymbol{\mu_d^i} \end{bmatrix} \tag{26}$$

The added noise $\boldsymbol{N}$ *changes value for each index* $\boldsymbol{j}$ *of the iteration process. It can be colored or white noise.*

GPC-EKF IMPROVEMENTS FOR REGRESSION SYSTEMS

For regression systems there is no state vector, but because the system is fed through the gPC-EKF, a state vector is created. This state vector is a combination of the regression equations and parameter values. The modifications for improving the state estimations are no longer needed. The Kalman update equations for the states are actually set to zero, such that:

$$\boldsymbol{K} = \left[\boldsymbol{0_1} \dots \boldsymbol{0_n}, \boldsymbol{k_{p1}} \dots \boldsymbol{k_{pn}}\right]^{\boldsymbol{T}} \tag{27}$$

There should be no alteration of the regression equations from the filter. The additions to the parameter distributions are helpful. The system is far more sensitive to this addition, however, and the additions should be very small.

GPC-EKF APPLIED TO A STATE SPACE SYSTEM

For a mass spring system, with state space formulation defined as:

$$\begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{v}} \\ \dot{\boldsymbol{m}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{v} \\ -\dfrac{\boldsymbol{k}}{\boldsymbol{m}} * \boldsymbol{x} \\ \boldsymbol{0} \end{bmatrix} + \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{\sin(t)} \\ \boldsymbol{0} \end{bmatrix} \tag{28}$$

The mass spring system is simulated and the parameters and states are estimated through the original gPC-EKF method. The first simulation performed is with the mass parameter perturbed but with the states of the filter set to the same initial conditions as the actual system.

Table 1 *lists the values for each of the filter's tuneable coefficients. Figure 1 shows the unmodified filter performing the state estimation of the system's position, Figure 2 shows the filter's estimation of the system's velocity, and Figure 3 shows the mass estimation.*

TABLE 1 PARAMETER VALUES FOR UNMODIFIED GPC-EKF SYSTEM

Poly Order	2
Q	6
Time Step (S)	0.01
$\boldsymbol{c}$	0
$\boldsymbol{c_1}$	0
$\boldsymbol{x_0}$	0
$\boldsymbol{v_0}$	0
$\boldsymbol{m^1}$	200
$\boldsymbol{m^2}$	30
$\boldsymbol{N_1}$	0
$\boldsymbol{N_2}$	0

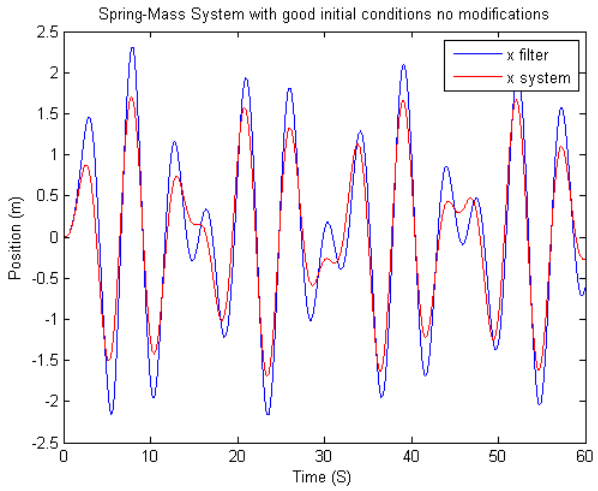

FIGURE 1 POSITION ESTIMATION FOR UNMODIFIED GPC-EKF

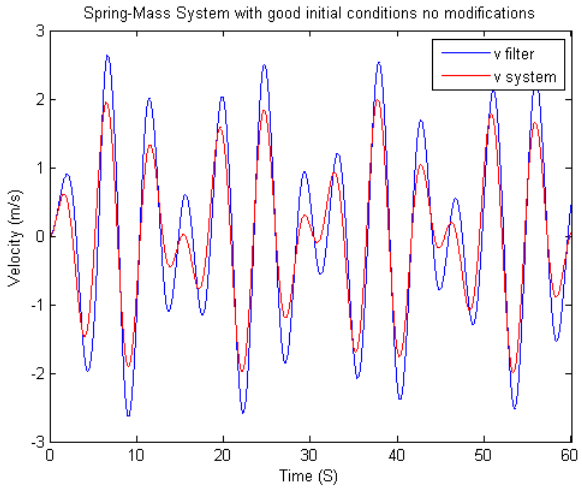

FIGURE 2 VELOCITY ESTIMATION FOR UNMODIFIED GPC-EKF

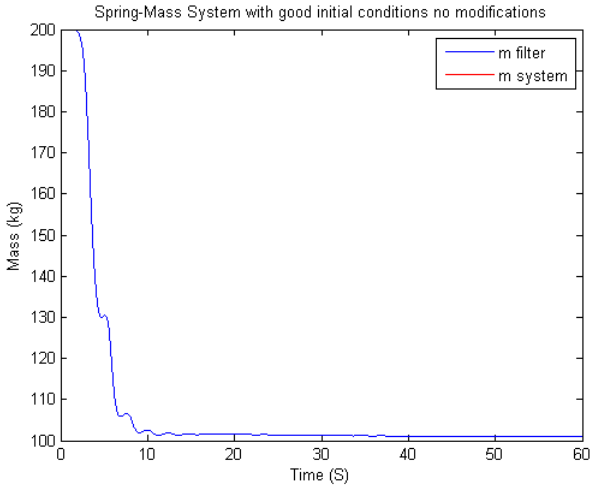

FIGURE 3 MASS ESTIMATION FOR UNMODIFIED GPC-EKF

This simulation demonstrates the key problem with the gPC-EKF. The covariance stays significant enough for the parameter value to almost converge, but the states show permanent tracking error. This is because the covariance has trivial information about the states. For initial errors in the states the unmodified gPC-EKF will perform poorly. It may stay stable, but will not be able to estimate any parameter values.

RESULTS OF THE GPC-EKF WITH MODIFICATIONS

For the same system run through the modified filter, the initial conditions can be off by a significant amount. Table 2 details the modified filter's coefficients.

TABLE 2 PARAMETER VALUES FOR GPC-EKF WITH MODIFICATIONS

Poly Order	2
Q	4
Time Step (S)	0.01
$\boldsymbol{c}$	0.001
$\boldsymbol{c_1}$	0.01
$\boldsymbol{x_0}$	15
$\boldsymbol{v_0}$	15
$\boldsymbol{m^1}$	1000
$\boldsymbol{m^2}$	50
$\boldsymbol{N_1}$	[-3,3]
$\boldsymbol{N_2}$	[-3,3]

Figure 4 shows the modified filter's estimation of the position, Figure 5 shows the velocity estimation, and Figure 6 shows the mass estimation.

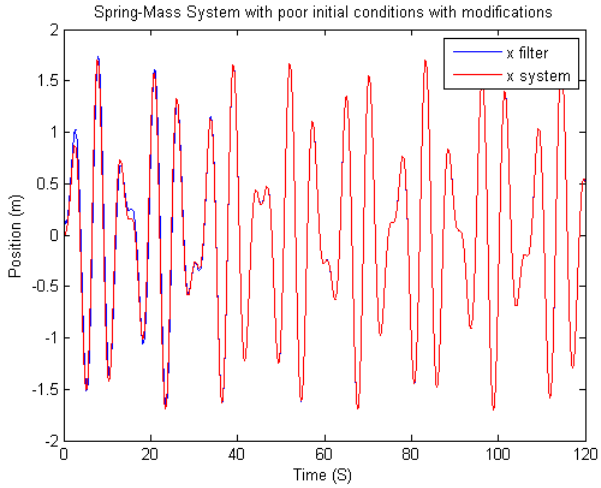

FIGURE 4 POSITION ESTIMATION FOR MODIFIED GPC-EKF

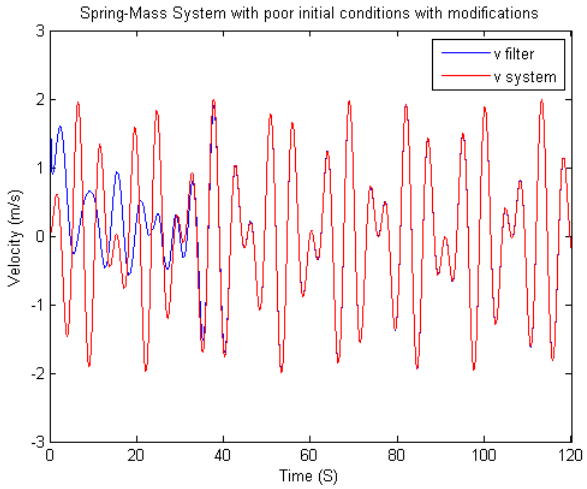

FIGURE 5 VELOCITY ESTIMATION FOR MODIFIED GPC-EKF

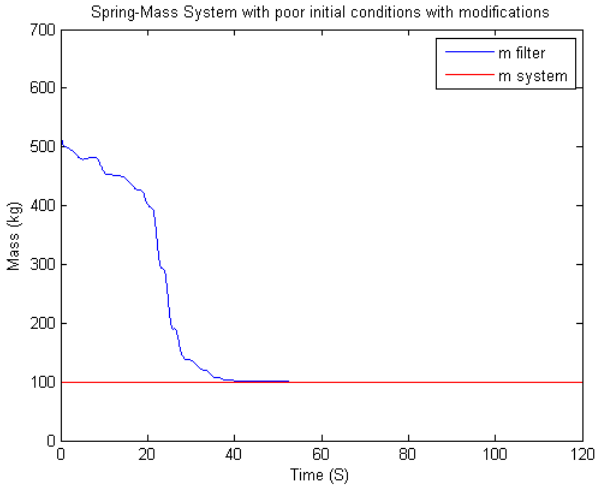

FIGURE 6 MASS ESTIMATION FOR MODIFIED GPC-EKF

The modified filter is able to handle significant errors in the initial states and parameters. This method performs significantly better than the unmodified filter.

GPC-EKF APPLIED TO REGRESSION SYSTEM

The state vector, including uncertain parameters, for the vehicle model is:

$$\boldsymbol{x} = \left[\ddot{\boldsymbol{Z}}, \ddot{\boldsymbol{\theta}}, \ddot{\boldsymbol{\phi}}; \boldsymbol{p}\right]^{\boldsymbol{T}} \qquad (29)$$

This is a vector of the vertical acceleration ($\ddot{\boldsymbol{Z}}$), pitch acceleration ($\ddot{\boldsymbol{\theta}}$), roll acceleration ($\ddot{\boldsymbol{\phi}}$), and the parameters ($\boldsymbol{p}$):

$$\boldsymbol{p} = \left[\boldsymbol{m}, \boldsymbol{J_\theta}, \boldsymbol{J_\phi}, \boldsymbol{h}\right] \tag{30}$$

The parameters of the system are mass, pitch inertia, roll inertia, and center of gravity height. The regression system is defined as:

$$\ddot{\boldsymbol{Z}} = \frac{\boldsymbol{1}}{\boldsymbol{m}} \sum_{\boldsymbol{i=fl,fr,rl,rr}} \boldsymbol{SF_i} \tag{31}$$

$$\ddot{\boldsymbol{\phi}} = \frac{\boldsymbol{1}}{\boldsymbol{J_\phi}} \left[\boldsymbol{l}\left(\boldsymbol{SF_{fl}} + \boldsymbol{SF_{rl}}\right) - \boldsymbol{r}\left(\boldsymbol{SF_{fr}} + \boldsymbol{SF_{rr}}\right) + \boldsymbol{mhA_y}\right] \tag{32}$$

$$\ddot{\boldsymbol{\theta}} = \frac{\boldsymbol{1}}{\boldsymbol{J_\theta}} \left[-\boldsymbol{a}\left(\boldsymbol{SF_{fl}} + \boldsymbol{SF_{fr}}\right) + \boldsymbol{b}\left(\boldsymbol{SF_{rl}} + \boldsymbol{SF_{rr}}\right) + \boldsymbol{mhA_x}\right] \tag{33}$$

$\boldsymbol{SF_i}$ is the force in the suspension for the front left (fl), front right (fr), rear left (rl) and rear right (rr) respectively. $\boldsymbol{A_y}$ is the lateral acceleration and $\boldsymbol{A_x}$ is the longitudinal acceleration of the vehicle. The geometric parameters ($\boldsymbol{l}, \boldsymbol{r}, \boldsymbol{a}, \boldsymbol{b}$) of the vehicle define the distance from the CG to the left side of the vehicle ($\boldsymbol{l}$), to the right side of the vehicle ($\boldsymbol{r}$), the distance to the front axle from the CG ($\boldsymbol{a}$), and the distance to the rear axle from the CG ($\boldsymbol{b}$).

The parameter estimation is performed for the vehicle model with no sensor noise first. Table 3 defines the filter's coefficients for the regression.

TABLE 3 PARAMETER VALUES FOR GPC-EKF FOR REGRESSION SYSTEM

Poly Order	2
Q	4
Time Step (S)	0.01
$\boldsymbol{c}$	0
$\boldsymbol{c_1}$	1/2000
$\boldsymbol{c_2}$	1/2000
$\boldsymbol{c_3}$	1/1000
$\boldsymbol{c_4}$	1/2000
R	0.01

Figure 7 shows the CG height estimation for both the RLS and gPC-EKF methods. The RLS method performs a better estimation.

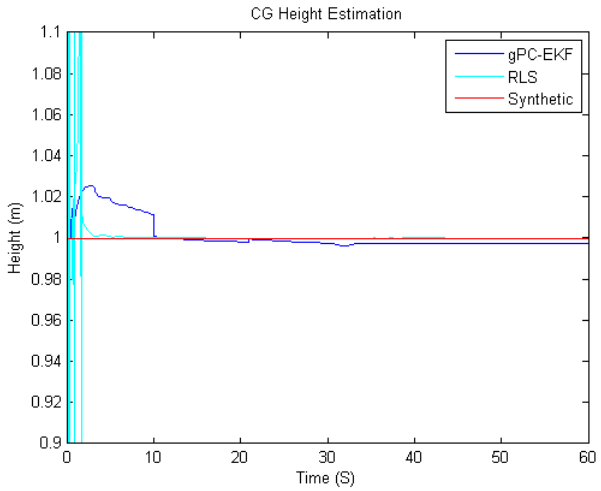

FIGURE 7 HEIGHT ESTIMATION

Figure 8 shows the mass estimation. The RLS algorithm again performs a better estimation of the system parameter. The gPC-EKF does start with a significantly worse initial estimate.

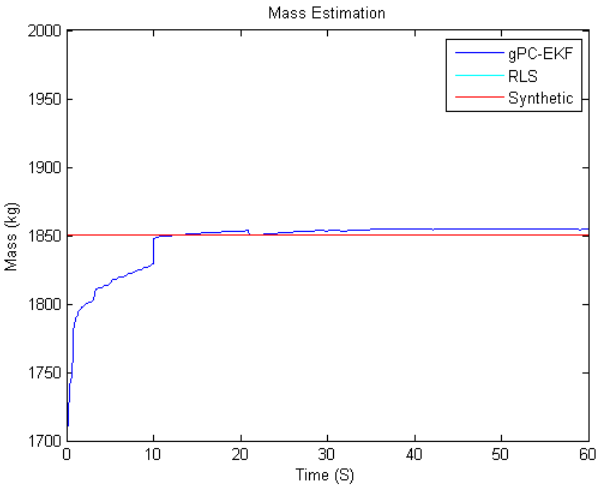

FIGURE 8 MASS ESTIMATION

Figure 9 shows the pitch inertia estimation. The RLS algorithm performs the estimation better than the gPC-EKF, with the RLS estimate overlapping the system's actual value.

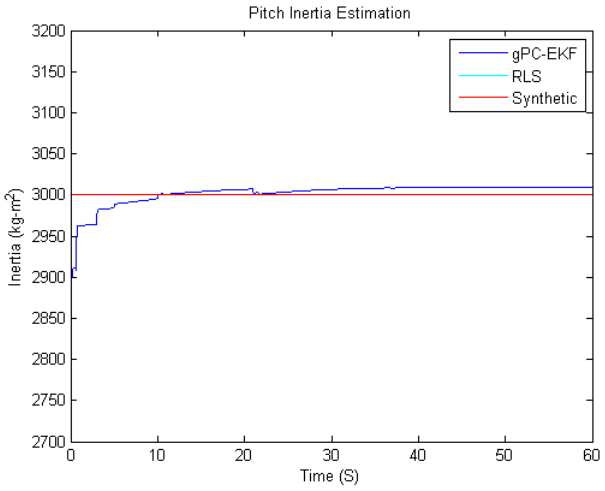

FIGURE 9 PITCH INERTIA ESTIMATION

Figure 10 shows the roll inertia estimation. The RLS algorithm does not perform as well as the gPC-EKF estimate here, and ends with a non-trivial biasing of the parameter estimate.

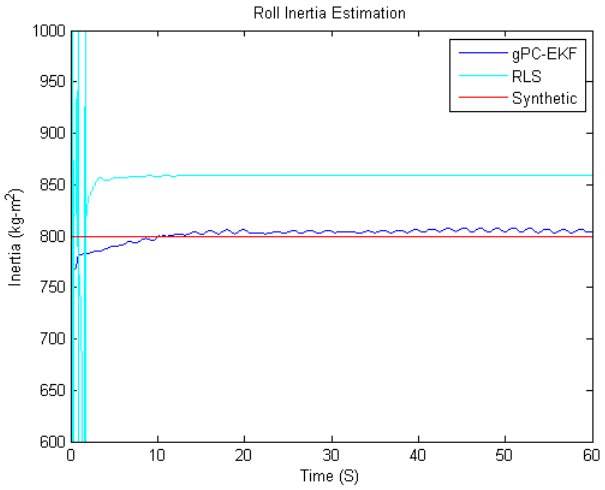

FIGURE 10 ROLL INERTIA ESTIMATION

The recursive least squares algorithm [10] starts out with good initial estimates for the parameter values but still has drastic variations in the beginning, and some of the parameters converge to incorrect values. The gPC-EKF is able to simultaneously estimate all four parameters without any serious variations, even though it starts with poor initial conditions. Overall, the RLS algorithm performs a better estimate than the gPC-EKF for this system without noise.

The same simulation is performed but with a white noise process added to the synthetic sensors. The noise is a $\mathbf{0.1}$ *variance white noise process.*

Table **4** *defines the gPC-EKF tuneable coefficients.*

TABLE 4 PARAMETER VALUES FOR GPC-EKF REGRESSION SYSTEM WITH NOISE

Poly Order	2
Q	4
Time Step (S)	0.01
$\boldsymbol{c}$	0
$\boldsymbol{c_1}$	1/2000
$\boldsymbol{c_2}$	1/2000
$\boldsymbol{c_3}$	1/1000
$\boldsymbol{c_4}$	1/2000
R	0.1

Figure 11 shows the CG height estimations. Note that only a small amount of noise causes the RLS algorithm to converge to an incorrect value. The gPC-EKF, however, is able to perform the estimation.

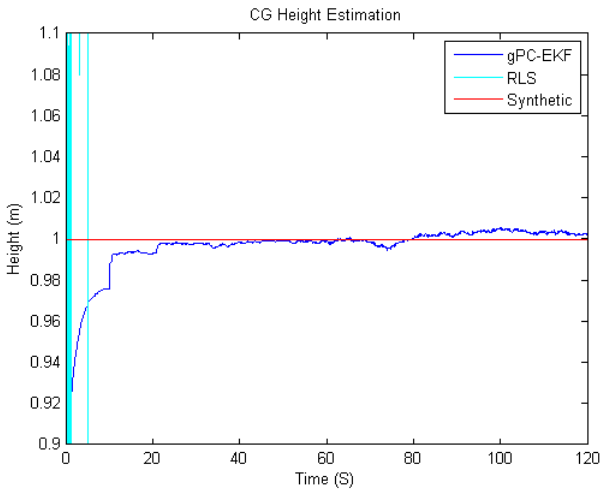

FIGURE 11 HEIGHT ESTIMATION WITH SENSOR NOISE

Figure 12 shows the mass estimation. The RLS algorithm is able to get close to the correct value, but does not perform as well as it did without the added noise. The gPC-EKF method performs a better estimation than the RLS.

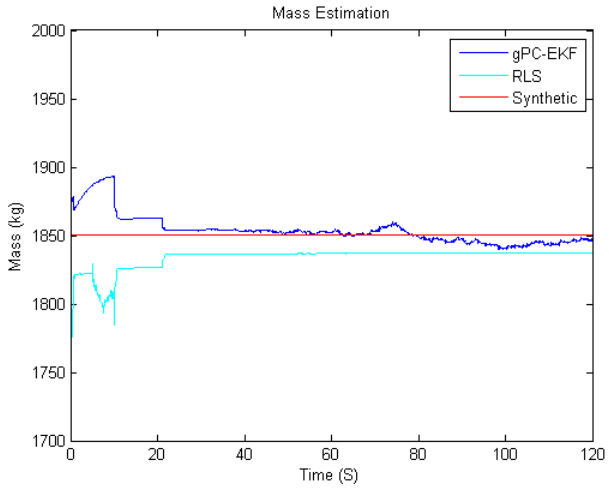

FIGURE 12 MASS ESTIMATION WITH SENSOR NOISE

Figure 13 shows the pitch inertia estimation. Both the RLS and the gPC-EKF methods are able to perform the parameter estimation here with acceptable accuracy.

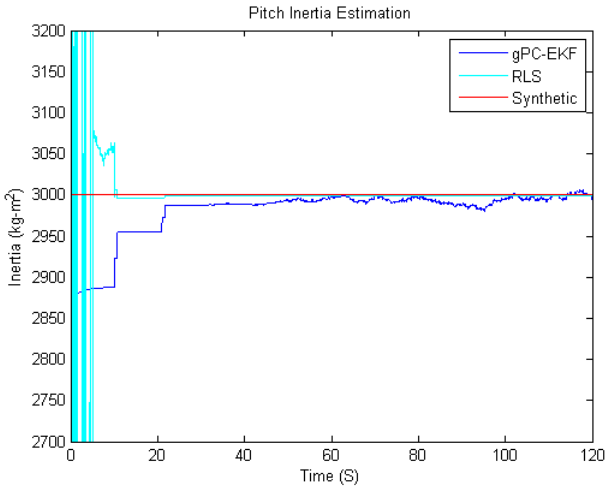

FIGURE 13 PITCH INERTIA ESTIMATION WITH SENSOR NOISE

Figure 14 shows the roll inertia estimation. Here the RLS algorithm does not converge to the actual value and instead converges to a value with significant error. The gPC-EKF performs the estimation with similar accuracy to the system without sensor noise.

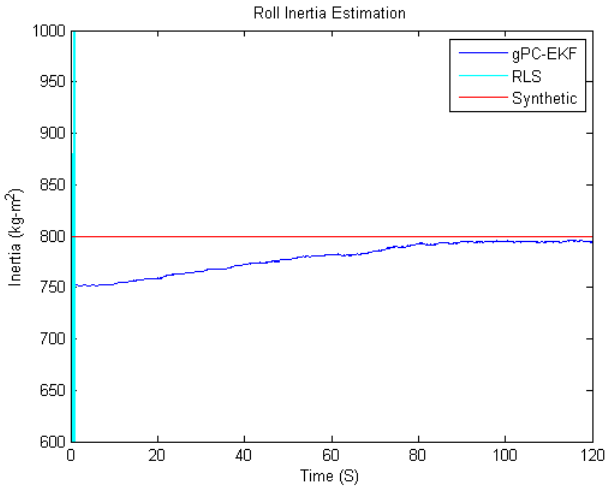

FIGURE 14 ROLL INERTIA ESTIMATION WITH SENSOR NOISE

The gPC-EKF has many different values that can be tuned to ensure good parameter estimations. As the sensor noise values increase there are many coefficients that can be tuned to improve the filter's convergence. In this case a slight modification of the noise covariance matrix enables the parameter estimation with noisy data. The RLS filter cannot handle the added noise and has both the CG height and the roll inertia values converge to incorrect values.

CONCLUSIONS AND RECOMMENDATIONS

The improvements to the gPC-EKF significantly improve the filter's ability to converge in both states and parameters for state space systems. The filter is able to perform the standard state estimations of the EKF once the modifications are in place.

For regression systems, the gPC-EKF shows a significant improvement over the RLS algorithm. The gPC-EKF is capable of performing the parameter estimations even when there is noise added to the measurements and the initial conditions are poor.

The gPC-EKF for these examples runs about three times faster than real-time on a first-generation laptop Intel i5 processor. The regression system handles up to about seven parameters for real-time estimations. The state space filter depends much more heavily on the method of forward integration, the functions, and the time-steps needed for the integrations. In general, for real-time applications, the number of parameters should remain small (1-3).

REFERENCE LIST

[1] E. D. Blanchard, A. Sandu, and C. Sandu, "Polynomial Chaos-Based Parameter Estimation Methods Applied to Vehicle System," *IMechE,* vol. 223, 2009.

[2] E. D. Blanchard, A. Sandu, and C. Sandu, "PSM: A Polynomial Chaos-Based Kalman Filter Approach for Parameter Estimation of Mechanical Systems," *Journal of Dynamic Systems, Measurement, and Control,* vol. 132, 2010.

[3] D. Xiu, "Efficient collocational approach for parametric uncertainty analysis," *Communications in Computational Physics,* vol. 2, pp. 293-309, 2007.

[4] D. Xiu, "Fast numerical methods for stochastic computations: a review," *Communications in Computational Physics,* vol. 5, pp. 242-272, 2009.

[5] D. Xiu and J. S. Hesthaven, "High-Order Collocation Methods for Differential Equations with Random Inputs," *SIAM Journal of Scientific Computing,* vol. 27, pp. 118-1139, 2005.

[6] A. Sandu, C. Sandu, and M. Ahmadian, "Modeling Multibody Systems With Uncertainties. Part I: Theoretical And Computational Aspects," *Multibody Syst Dyn,* 2006.

[7] C. Sandu, A. Sandu, and M. Ahmadian, "Modeling multibody systems with uncertainties. Part II: Numerical applications," *Multibody Syst Dyn,* vol. 15, 2006.

[8] P. Zarchan and H. Musoff, *Fundamentals of Kalman Filtering: A practical Approach* vol. 232: American Institute of Aeronautics and Astronautics, Inc., 2009.

[9] E. D. Blanchard, "Polynomial Chaos Approaches to Parameter Estimation and Control Design for Mechanical Systems with Uncertain Parameters," 2010.

[10] L. LJung, *System Identification: Theory for the user*: Prentice Hall PTR, 1999.

PART 2: REAL-TIME VEHICLE PARAMETER ESTIMATION

PROBLEM

ABSTRACT

Parameter estimation for vehicle systems is in general a challenging topic from both sensor instrumentation and modeling perspectives. Modeling vehicle systems is a rather complex process, especially considering the numerous unknown effects on the system, such as, for example, aerodynamic effects, road grade and bank angles, roll and pitch kinematics, and suspension nonlinearities. This study develops a method that is able to estimate several vehicle parameters with high accuracy for regular driving behavior. The parameter estimations are performed using the polynomial chaos-based extended Kalman filter. This method is a computationally efficient, derivative free, iterative, non-linear regression technique which is able to estimate multiple parameters in real time. This report presents the results obtained for estimating the location of the CG of the vehicle in all three directions, an estimate of the coefficient of aerodynamic drag, as well as the sprung mass of the vehicle using the proposed technique. Real test data have been used for validation purposes.

INTRODUCTION

Vehicle rollover events are highly nonlinear. These events are difficult to prevent and even harder to detect. Accurate knowledge of the vehicle parameters is critical to vehicle stability.

This study presents a method that is capable of extracting multiple vehicle parameters in real time. The vehicle parameters that are estimated are the vehicle's center of gravity (CG) in the horizontal plane, the mass of the vehicle, the coefficient of aerodynamic drag, and the distances from the CG to the roll and pitch centers. The method is applied to a vehicle performing regular driving in two different environments: a rural road and an urban area. The parameters that can be validated are the vehicle's mass, and lateral and longitudinal CG positions. These parameters are estimated to high accuracy.

The goal of this study was to describe a parameter estimation method that can be implemented on a vehicle in real time for regular driving purposes. There should be no limiting assumptions, and the system should be capable of running for extended periods of time.

The vehicle parameter estimations are performed by the generalized polynomial chaos extended Kalman filter for regression systems. The generalized polynomial chaos technique is a computationally efficient method for propagation of uncertain parameters through dynamical systems [1, 2]. The gPC-EKF is a regression technique that preserves the structure of the nonlinear equations, unlike recursive least squares. The preservation of the structure enables a better estimate of the parameter values.

The gPC-EKF for regression systems was developed simultaneously with this research. The method is detailed in [3].

VEHICLE MODEL

Modeling of vehicle systems is complicated. There are many effects that cannot be accounted for (aerodynamic, road excitation, roll/pitch center mechanics, etc.) which are important effects. For parameter estimation, the more accurate the model and the more knowledge about the system inputs, the better the parameter estimation schemes operate. Model-based parameter estimation that includes forward integrations is generally not applicable due to errors in the modeling of the vehicle.

The model proposed here is a regression system that does not require forward integration. This model is based off of a load transfer model (LTM). The model uses the force balance between suspension struts to predict the parameter values. The standard LTM is a model that attempts to balance lateral or longitudinal acceleration forces with the transfer of forces at the suspension struts. This model is a fairly general model which is capable of dealing with most of the effects that a vehicle experiences during regular driving. Wong [4] derives some of the basics of the model, and here the generalized model is designed and expanded to account for some of the aerodynamic effects.

Load Transfer Model

During dynamic maneuvers the changes in forces at each corner of a vehicle can be described as load or weight transfer from one section of the vehicle to another [4, 5]. This occurs purely because of the external forces applied to the vehicle, and is not an actual transfer of mass.

For a vehicle that is cornering, the load transfer from one side of the vehicle to another is defined as:

$$\Delta W = \frac{m\, A_z\, A_x\, h_r}{B} \qquad (34)$$

Where m *is the mass of the vehicle,* A_z *is the vertical acceleration that the vehicle feels,* A_x *is the centrifugal acceleration in g's,* h_r *is the distance that the center of gravity is above the roll center of the vehicle, and* B *is the track width of the vehicle. This is calculated as the moment about the roll center of the vehicle. A similar moment calculation about the pitch center can be performed to estimate the weight transfer during acceleration or braking. A diagram of this model can be seen in Figure 15.*

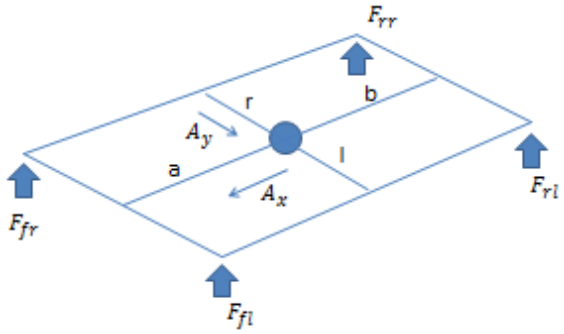

FIGURE 15 VEHICLE MODEL DIAGRAM

When each of these effects is summed individually at the suspension points of the vehicle, the load at each corner is:

$$F_{fl} = \frac{mA_z}{BL}\left[br - A_y bh_r - A_x rh_p\right] - \frac{rh_p V_x^2 C}{BL} \quad (35)$$

$$F_{fr} = \frac{mA_z}{BL}\left[bl + A_y bh_r - A_x lh_p\right] - \frac{lh_p V_x^2 C}{BL} \quad (36)$$

$$F_{rl} = \frac{mA_z}{BL}\left[ar - A_y ah_r + A_x rh_p\right] + \frac{rh_p V_x^2 C}{BL} \quad (37)$$

$$F_{rr} = \frac{mA_z}{BL}\left[al + A_y ah_r + A_x lh_p\right] + \frac{lh_p V_x^2 C}{BL} \quad (38)$$

The terms a, b, l, r, L, h_p *are the distance from the front axle to the CG, distance from the rear axle to the CG, distance to the CG from the left track, distance to the CG from the right track, wheelbase of the vehicle, and the height of the CG above the pitch center of the vehicle.* F_i *is the force in the suspension for the front left (fl), front right (fr), rear left (rl), and rear right (rr), respectively. Here,* A_y *is the lateral acceleration at the CG, and* A_x *is the longitudinal acceleration at the CG.* A_y *and* A_x *are measured in g's,* A_z *is measured in* $\frac{m}{s}$.

The term C is what incorporates the aerodynamic drag. In general this term should include more effects than just the vehicle's forward velocity, such as wake, drafting, and wind effects. These effects are much more complicated to model and are therefore assumed to be negligible enough (which is to say that the estimation term will evaluate a "good enough" value for the parameter estimate).

The effects of aerodynamic forces are significant enough that they cannot be neglected. For an SUV with a value for $C_d A \approx 1$*, the aerodynamic drag force applied to a vehicle travelling at* $110\ Km/H$ *is approximately 575 N.*

Modified Load Transfer Model (MLTM)

The load transfer model can be applied to the vehicle using linear accelerometers at each suspension strut, rather than an estimate of the vertical acceleration at the CG of the vehicle. In practice this is a more reasonable method of estimation, as the acceleration is generally not known at the center of gravity. For an unknown center of gravity, the measured CG vertical acceleration can be approximated as:

$$\ddot{\boldsymbol{Z}} = \frac{\mathbf{1}}{\mathbf{4}}\left[\boldsymbol{A_{z,fl}} + \boldsymbol{A_{z,fr}} + \boldsymbol{A_{z,rl}} + \boldsymbol{A_{z,rr}} + \mathbf{2}(\boldsymbol{r} - \boldsymbol{l}) * \ddot{\boldsymbol{\phi}} + \mathbf{2}(\boldsymbol{b} - \boldsymbol{a})\ddot{\boldsymbol{\theta}}\right] \quad (39)$$

This model returns an estimate of the mass at each corner of the vehicle rather than an estimate of the total mass of the vehicle. These estimations are then used for estimating the total mass and the center of gravity of the vehicle.

The load transfer equations are transformed:

$$\boldsymbol{F_{fl}} = \boldsymbol{m_{fl}A_{z,fl}} - \frac{\bar{\boldsymbol{m}}\ddot{\boldsymbol{Z}}}{\boldsymbol{BL}}\boldsymbol{A_y bh_r} - \frac{\bar{\boldsymbol{m}}\ddot{\boldsymbol{Z}}}{\boldsymbol{BL}}\boldsymbol{A_x rh_p} - \frac{\boldsymbol{rh_p V_x^2 C}}{\boldsymbol{BL}} \quad (40)$$

$$\boldsymbol{F_{fr}} = \boldsymbol{m_{fr}A_{z,fr}} + \frac{\bar{\boldsymbol{m}}\ddot{\boldsymbol{Z}}}{\boldsymbol{BL}}\boldsymbol{A_y bh_r} - \frac{\bar{\boldsymbol{m}}\ddot{\boldsymbol{Z}}}{\boldsymbol{BL}}\boldsymbol{A_x lh_p} - \frac{\boldsymbol{lh_p V_x^2 C}}{\boldsymbol{BL}} \quad (41)$$

$$\boldsymbol{F_{rl}} = \boldsymbol{m_{rl}A_{z,rl}} - \frac{\bar{\boldsymbol{m}}\ddot{\boldsymbol{Z}}}{\boldsymbol{BL}}\boldsymbol{A_y ah_r} + \frac{\bar{\boldsymbol{m}}\ddot{\boldsymbol{Z}}}{\boldsymbol{BL}}\boldsymbol{A_x rh_p} + \frac{\boldsymbol{rh_p V_x^2 C}}{\boldsymbol{BL}} \quad (42)$$

$$\boldsymbol{F_{rr}} = \boldsymbol{m_{rr}A_{z,rr}} + \frac{\bar{\boldsymbol{m}}\ddot{\boldsymbol{Z}}}{\boldsymbol{BL}}\boldsymbol{A_y ah_r} + \frac{\bar{\boldsymbol{m}}\ddot{\boldsymbol{Z}}}{\boldsymbol{BL}}\boldsymbol{A_x lh_p} + \frac{\boldsymbol{lh_p V_x^2 C}}{\boldsymbol{BL}} \quad (43)$$

$$\boldsymbol{F_{bounce}} = \boldsymbol{F_{fl}} + \boldsymbol{F_{fr}} + \boldsymbol{F_{rl}} + \boldsymbol{F_{rr}} = \boldsymbol{m_{fl}A_{z,fl}} + \boldsymbol{m_{fr}A_{z,fr}} + \boldsymbol{m_{rl}A_{z,rl}} + \boldsymbol{m_{rr}A_{z,rr}} \quad (44)$$

With the total bounce of the vehicle:

$$\boldsymbol{F_{bounce}} = \boldsymbol{m_{fl}A_{z,fl}} + \boldsymbol{m_{fr}A_{z,fr}} + \boldsymbol{m_{rl}A_{z,rl}} + \boldsymbol{m_{rr}A_{z,rr}} \quad (45)$$

Here the estimates for $\bar{\boldsymbol{m}}$ are the sum of the estimates for $\boldsymbol{m_{fl}}, \boldsymbol{m_{fr}}, \boldsymbol{m_{rl}}, \boldsymbol{m_{rr}}$, where each of these is the mass estimate at each corner of the vehicle. Due to the number of uncertain parameters it is

not possible to perform the gPC-EKF estimation on this with the desire of a real-time estimation. This model has too many uncertain parameters.

The lateral and longitudinal positions can be calculated through basic geometry of the masses at each of the suspension struts:

$$\boldsymbol{a} = (\boldsymbol{m_{rl}} + \boldsymbol{m_{rr}})/\bar{\boldsymbol{m}} \quad (46)$$

$$\boldsymbol{l} = (\boldsymbol{m_{fr}} + \boldsymbol{m_{rr}})/\bar{\boldsymbol{m}} \quad (47)$$

With these substitutions the MLTM becomes:

$$\boldsymbol{F_{fl}} = \boldsymbol{m_{fl}}\,\boldsymbol{A_{z,fl}} - \frac{(\boldsymbol{m_{fl}} + \boldsymbol{m_{fr}})\ddot{\boldsymbol{Z}}}{\boldsymbol{B}}\boldsymbol{A_y h_r} - \frac{(\boldsymbol{m_{fl}} + \boldsymbol{m_{rl}})\ddot{\boldsymbol{Z}}}{\boldsymbol{L}}\boldsymbol{A_x h_p} - \frac{\boldsymbol{h_p V_x^2 C}}{\boldsymbol{L}}\left(\frac{\boldsymbol{m_{fl}} + \boldsymbol{m_{rl}}}{\boldsymbol{m_{fl}} + \boldsymbol{m_{fr}} + \boldsymbol{m_{rl}} + \boldsymbol{m_{rr}}}\right) \quad (48)$$

$$\boldsymbol{F_{fr}} = \boldsymbol{m_{fr}}\,\boldsymbol{A_{z,fr}} + \frac{(\boldsymbol{m_{fl}} + \boldsymbol{m_{fr}})\ddot{\boldsymbol{Z}}}{\boldsymbol{B}}\boldsymbol{A_y h_r} - \frac{(\boldsymbol{m_{fr}} + \boldsymbol{m_{rr}})\ddot{\boldsymbol{Z}}}{\boldsymbol{L}}\boldsymbol{A_x h_p} - \frac{\boldsymbol{h_p V_x^2 C}}{\boldsymbol{L}}\left(\frac{\boldsymbol{m_{fr}} + \boldsymbol{m_{rr}}}{\boldsymbol{m_{fl}} + \boldsymbol{m_{fr}} + \boldsymbol{m_{rl}} + \boldsymbol{m_{rr}}}\right) \quad (49)$$

$$\boldsymbol{F_{rl}} = \boldsymbol{m_{rl}}\,\boldsymbol{A_{z,rl}} - \frac{(\boldsymbol{m_{rl}} + \boldsymbol{m_{rr}})\ddot{\boldsymbol{Z}}}{\boldsymbol{B}}\boldsymbol{A_y h_r} + \frac{(\boldsymbol{m_{fl}} + \boldsymbol{m_{rl}})\ddot{\boldsymbol{Z}}}{\boldsymbol{L}}\boldsymbol{A_x h_p} + \frac{\boldsymbol{h_p V_x^2 C}}{\boldsymbol{L}}\left(\frac{\boldsymbol{m_{fl}} + \boldsymbol{m_{rl}}}{\boldsymbol{m_{fl}} + \boldsymbol{m_{fr}} + \boldsymbol{m_{rl}} + \boldsymbol{m_{rr}}}\right) \quad (50)$$

$$\boldsymbol{F_{rr}} = \boldsymbol{m_{rr}}\,\boldsymbol{A_{z,rr}} + \frac{(\boldsymbol{m_{rl}} + \boldsymbol{m_{rr}})\ddot{\boldsymbol{Z}}}{\boldsymbol{B}}\boldsymbol{A_y h_r} + \frac{(\boldsymbol{m_{fr}} + \boldsymbol{m_{rr}})\ddot{\boldsymbol{Z}}}{\boldsymbol{L}}\boldsymbol{A_x h_p} + \frac{\boldsymbol{h_p V_x^2 C}}{\boldsymbol{L}}\left(\frac{\boldsymbol{m_{fr}} + \boldsymbol{m_{rr}}}{\boldsymbol{m_{fl}} + \boldsymbol{m_{fr}} + \boldsymbol{m_{rl}} + \boldsymbol{m_{rr}}}\right) \quad (51)$$

$$\boldsymbol{F_{bounce}} = \boldsymbol{m_{fl}}\boldsymbol{A_{z,fl}} + \boldsymbol{m_{fr}}\boldsymbol{A_{z,fr}} + \boldsymbol{m_{rl}}\boldsymbol{A_{z,rl}} + \boldsymbol{m_{rr}}\boldsymbol{A_{z,rr}} \quad (52)$$

With the uncertain parameters:

$$\boldsymbol{p} = \left[\boldsymbol{m_{fl}}, \boldsymbol{m_{fr}}, \boldsymbol{m_{rl}}, \boldsymbol{m_{rr}}, \boldsymbol{h_r}, \boldsymbol{h_p}, \boldsymbol{C}\right]^{\boldsymbol{T}} \qquad (53)$$

METHODOLOGY

EXPERIMENTAL VEHICLE

The experimental data is collected on a Land Rover Defender 110 off-road vehicle. This specific experimental vehicle is fitted with a prototype Hydro-pneumatic spring-damper suspension system developed by the Vehicle Dynamics Group of the University of Pretoria. Each strut is equipped with a pressure transducer which measures the pressure above each strut piston. The pressure is used to determine the combined damping and spring forces at each strut and thus accounts for all forces, except for friction, in the strut. Each strut is also equipped with a linear potentiometer displacement transducer which is used to obtain each strut's displacement. Two accelerometers are mounted vertically on each strut, a 10g accelerometer on the unsprung mass and a 4g accelerometer on the sprung mass. A 4g tri-axial accelerometer is also placed at approximately the center of mass of the vehicle. The vehicle is also equipped with a Racelogic VBox 3 GPS receiver. The GPS receiver is used to accurately determine the vehicle speed and provides additional information such as latitude, longitude, heading and height above sea level. The GPS information is logged separately but synchronized by means of a trigger signal as well as the vehicle speed which is recorded on both data acquisition systems. All vehicle mass and inertia properties were obtained experimentally [6]. Figure 16 shows the instrumented land rover that was used for the experiments.

FIGURE 16 INSTRUMENTED LAND ROVER

VEHICLE DATA SETS

Two different data sets were used in evaluating the parameter estimation. The first data set was obtained on a winding rural road with many filled and unfilled potholes. The road is therefore very uneven and bumpy. The test was conducted at normal driving speeds with traffic, thus a non-constant speed profile is obtained with typical braking and acceleration pattern of an everyday driver. The uneven road surface and speed profile in conjunction with the road cornering, embankment and grade results in

considerable excitation to the vehicle. The second data set was obtained in an urban environment with a much smoother road surface and very little cornering. Most of the road excitation is as a result of the road grade. Thus, the urban road has a lot less road excitation and more excitation in the form of braking and accelerating.

VEHICLE PARAMETER ESTIMATION TECHNIQUE

The generalized polynomial chaos extended Kalman filter (gPC-EKF) is applied to the vehicle for the parameter estimation. The gPC-EKF for regression systems is a real-time, nonlinear, derivative free, iterative regression method.

For the estimation method to apply, a state vector of the regression equations and parameter is created. For the LTM the state vector is:

$$\boldsymbol{x} = \left[\boldsymbol{F_{fl}}, \boldsymbol{F_{fr}}, \boldsymbol{F_{rl}}, \boldsymbol{F_{rr}}, \boldsymbol{F_{bounce}}, \boldsymbol{m}, \boldsymbol{a}, \boldsymbol{l}, \boldsymbol{h_r}, \boldsymbol{h_p}, \boldsymbol{C}\right]^{\boldsymbol{T}} \quad (54)$$

The details of the estimation technique are included in [3].

FINDINGS

VEHICLE PARAMETER ESTIMATION

The parameter estimation technique is applied to both the LTM and MLTM for comparison. For the Urban data set the mass estimation can be found in Figure 17, the lateral position estimation is performed in Figure 18, and the longitudinal position estimate is demonstrated in Figure 19.

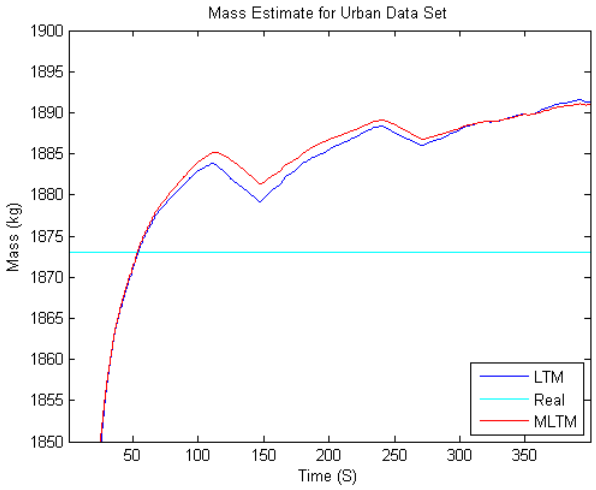

FIGURE 17 MASS ESTIMATION FOR URBAN DATA SET

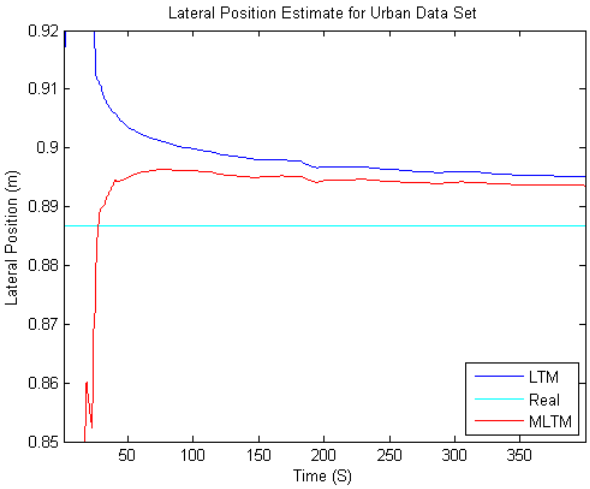

FIGURE 18 LATERAL POSITION ESTIMATION FOR URBAN DATA SET

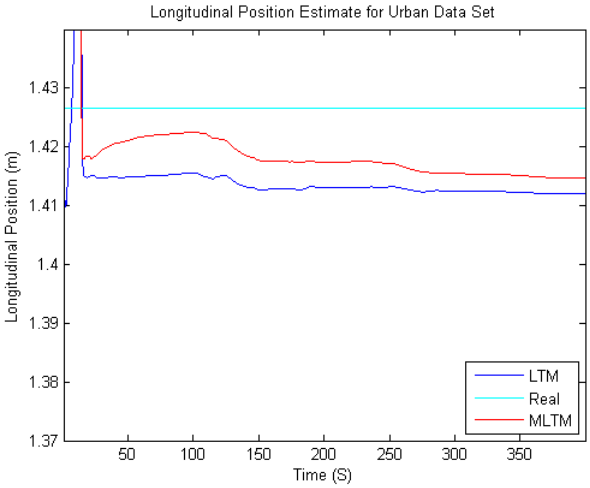

FIGURE 19 LONGITUDINAL POSITION ESTIMATE FOR URBAN DATA SET

The error analysis of the parameter estimations performed for the Urban data set are shown in Table 5.

TABLE 5 PARAMETER ESTIMATION RESULTS FOR THE URBAN DATA SET

Urban Data Set	*Mass*	*Longitudinal Position*	*Lateral Position*
Estimated Value (LTM)	*1891*	*1.412*	*0.895*
Error (LTM)	*0.96%*	*1.02%*	*0.93%*
Estimated Value (MLTM)	*1891*	*1.415*	*0.8934*
Error (MLTM)	*0.96%*	*0.81%*	*0.75%*

The parameter estimations for the Rural data set are performed. The mass estimation is performed in Figure 20.

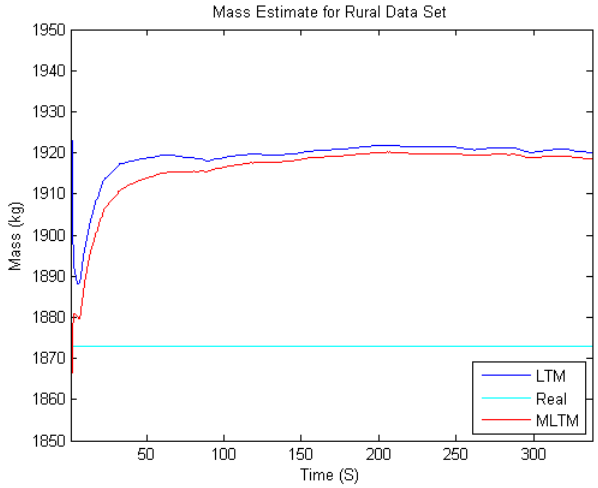

FIGURE 20 MASS ESTIMATE FOR RURAL DATA SET

Figure 21 shows the parameter estimation for the lateral position of the vehicle's CG.

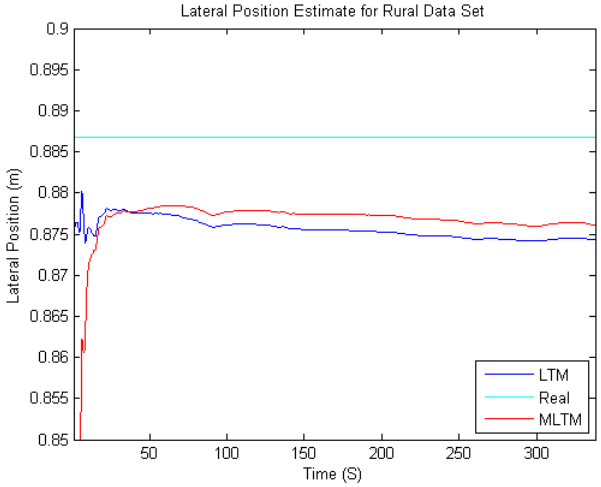

FIGURE 21 LATERAL POSITION ESTIMATE FOR RURAL DATA SET

Figure 22 shows the longitudinal position estimate for the vehicle's CG.

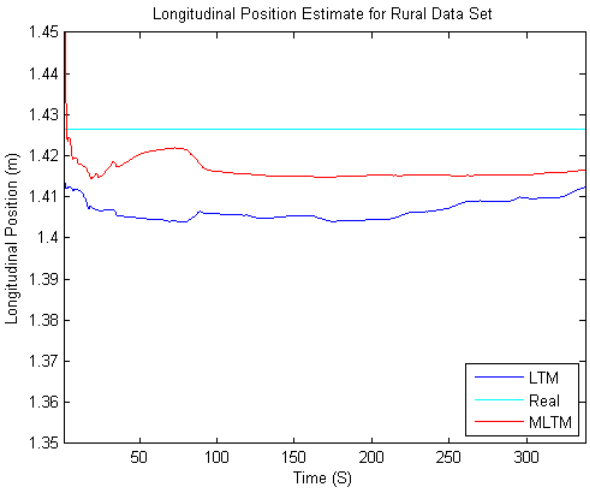

FIGURE 22 LONGITUDINAL POSITION ESTIMATE FOR RURAL DATA SET

The final parameter values and errors for the Rural data set are shown in Table 6.

TABLE 6 PARAMETER ESTIMATION RESULTS FOR THE RURAL DATA SET

Rural Data Set	*Mass*	*Longitudinal Position*	*Lateral Position*
Estimated Value (LTM)	*1920*	*1.412*	*0.8743*
Error (LTM)	*2.57%*	*1.02%*	*1.4%*
Estimated Value (MLTM)	*1919*	*1.417*	*0.8761*
Error (MLTM)	*2.46%*	*0.67%*	*1.2%*

CONCLUSIONS AND RECOMMENDATIONS

Both methods show exceptional ability to estimate the parameters in real-time within error bounds of 0.6% and 2.6%. The reason for creating the MLTM is that it has a significantly reduced need for prior knowledge about the CG location. The LTM requires that sensors be placed at the CG of the vehicle. Obviously, if that is already known, then this whole scheme is unnecessary. The MLTM is designed to remove that requirement through the ability to estimate the CG accelerations through the accelerometers placed at the four corners of the vehicle.

The errors in the parameters are primarily caused from instrumentation. The accuracy of the parameter estimate is directly proportional to the accuracy of the sensors. Slight misalignments of the sensors or the filtering method for the data play a large role in the accuracy of the parameter estimate. The values for the actual values are approximate, as they are measured before the tests are conducted.

The difference in mass between the Rural and the Urban tests can is most likely associated with the consumption of fuel.

The values shown are the parameters that can be validated. There are also estimates for the vehicle's aerodynamic drag, and distances from the CG to the roll and pitch centers of the vehicle. These parameter estimates are similar for both models. The value for the aerodynamic drag is incorrect compared to the experimentally measured value. The aerodynamic effects do not sum at the CG of the vehicle, instead they interact with the vehicle in a way that is consistent with the topology of the vehicle's surface. The value for the coefficient of drag is therefore a representative value of the aerodynamic effects at the CG of the vehicle with respect to the load transfer mechanics and not a good estimate of the actual drag effects.

Because the CG of the vehicle is just about at the half-width, half-length of the vehicle the two methods show comparable results. If the CG of the vehicle were different these methods would show more significant differences. The MLTM method shows a general improvement over the LTM, as it was expected.

REFERENCES

[1] *D. Xiu, "Efficient collocational approach for parametric uncertainty analysis," Communications in Computational Physics, vol. 2, pp. 293-309, 2007.*

[2] *D. Xiu, "Fast numerical methods for stochastic computations: a review," Communications in Computational Physics, vol. 5, pp. 242-272, 2009.*

[3] *J. Kolansky and C. Sandu, "Generalized Polynomial Chaos Extended Kalman Filter: Improvement and Expansion," Proceedings of IDETC/CIE 2013, 2013.*

[4] *J. Y. Wong, Theory of Ground Vehicles: Wiley, 2008.*

[5] *P. Currier, "A Method for Modeling and Prediction of Ground Vehicle Dynamics and Stability in Autonomous Systems," Ph.D, Dissertation, Virginia Tech, 2011.*

[6] *P. E. Uys, P. S. Els, M. J. Thoresson, K. G. Voigt, and W. C. Combrinck, "Experimental determination of moments of inertia for an off-road vehicle in a regular engineering laboratory," International Journal of Mechanical Engineering Education, vol. 34, p. 291, 2006.*

PART 3: GPC-BASED BATTERY CHARGE AND HEALTH ESTIMATION

PROBLEM

Abstract

In this effort, we use the generalized Polynomial Chaos theory (gPC) for the real-time state and parameter estimation of electrochemical batteries. We use an equivalent circuit battery model, comprising two states and five parameters, and then formulate the online parameter estimation problem using battery current and voltage measurements. Using a combination of the conventional recursive gradient-based search algorithm and gPC framework, we propose a novel battery parameter estimation strategy capable of estimating both battery state-of-charge (SOC) and parameters related to battery health, e.g., battery charge capacity, internal resistance, and relaxation time constant. Using a combination of experimental tests and numerical simulations, we examine and demonstrate the effectiveness of the proposed battery estimation method.

1-Introduction

This study examines the problem of estimating the states and parameters of electrochemical batteries for real-time battery management and control applications. Simultaneous battery state-of-charge (SOC) and state-of-health (SOH) estimation has recently gained significant attention in the scientific community due to the rapid growth of battery-powered systems, such as potable electronics, electric vehicles, and stationary grid storage systems, and the need for scientific analysis and design. This study attempts to address the battery SOC/SOH estimation problem by combining the generalized polynomial chaos (gPC) theory with the recursive parameter estimation technique.

The current battery estimation literature embodies a large number of studies on the SOC estimation problem. Methods such as mixed coulomb-counting with voltage correction [1, 2] and Kalman filtering [3-5] are among the most widely-used methods on this front. The battery SOH estimation problem is, however, a less mature area, mainly because of the generic definition of SOH which can result in various interpretations. In most studies, SOH is related to the degradation of specific parameters, such as battery internal resistance, or electrochemical reaction rates, etc. Methods such as, sub-space parameter estimation [6], particle filtering [7], Lyapunov-based adaptive estimation [8], and nonlinear PDE-based least squares estimation [9] have been proposed, and in some cases experimentally implemented.

Most of the current battery health estimation methods suffer at least from one of the following weaknesses: (i) Using linear circuit models for batteries which are essentially nonlinear, (ii) choosing the battery health-related parameters arbitrarily, and not including the battery charge capacity in the list of unknown parameters to be estimated, and (iii) deriving the estimation scheme based on a set of specific charge/discharge cycles, rather than assuming a broader range of cycling conditions.

In this study, we address the above problems by simultaneously estimating battery SOC, charge capacity, internal resistance, and the relaxation time constant using a second order equivalent circuit model, the

generalized polynomial chaos (gPC) theory, and the recursive parameter estimation technique. We also account for the nonlinearity of battery open-circuit voltage through a quasi-linearization method. The results show the proposed scheme provides an accurate estimation of the battery parameters and states in both short-term SOC estimation and long-term health estimation and monitoring applications, under arbitrary cycling conditions.

The remainder of the presentation is organized as follows: In section 2, a mathematical description of the model is provided. Section 3, formulates the battery estimation problem and uses the gPC and recursive estimation methods to obtain the parameter update laws. Section 4 evaluates the developed estimator for an experimentally characterized $LiFePO_4$ battery model in a plug-in hybrid electric vehicle (PHEV) application. Finally, Section 5 summarizes the study's key conclusions.

APPROACH

2- System Model

The estimation method proposed in this strategy is based on a second-order equivalent circuit battery model shown in Figure 23*. The circuit consists of a SOC-dependent voltage source, so-called the open circuit voltage, V_{oc}, in series with a resistor, R, and a parallel resistor/capacitor pair, R_r and C_r, representing the battery relaxation dynamics. When a positive current is applied to the battery (as shown in the picture) the open-circuit voltage increases as a result of charge accumulation in the cell. In contrary, when we discharge the cell, the open-circuit voltage drops due to the loss of stored charge. The mathematical description of this system can be written as:*

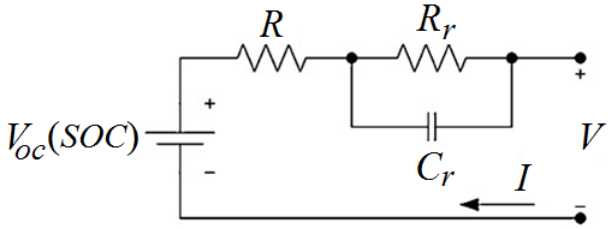

FIGURE 23 A SECOND-ORDER EQUIVALENT CIRCUIT BATTERY MODEL

$$\begin{cases} \dot{\boldsymbol{SOC}}(\boldsymbol{t}) = \frac{\mathbf{1}}{\boldsymbol{Q}}\boldsymbol{I}(\boldsymbol{t}) \\ \dot{\boldsymbol{V}}_{\boldsymbol{r}}(\boldsymbol{t}) = -\frac{\mathbf{1}}{\boldsymbol{R_r C_r}}\boldsymbol{V_r}(\boldsymbol{t}) + \frac{\mathbf{1}}{\boldsymbol{C_r}}\boldsymbol{I}(\boldsymbol{t}) \end{cases} \quad (55)$$

$$\boldsymbol{V}(\boldsymbol{t}) = \boldsymbol{V_{oc}}\big(\boldsymbol{SOC}(\boldsymbol{t})\big) + \boldsymbol{V_r}(\boldsymbol{t}) + \boldsymbol{RI}(\boldsymbol{t}) \quad (56)$$

where Q is the battery charge capacity, I is the applied (charging) current, V_r is the voltage drop across the resistor/capacitor pair, and V is the battery terminal (output) voltage.

The overarching goal of this study is to find a solution for estimating the parameters of the battery model, i.e., Q, R, C_r , and R_r, as well as the battery SOC trajectory from poor but reasonable initial

guesses. For the SOC estimation, we add its initial value, SOC(0), to the list of unknown parameters, and adjust the SOC estimate by estimating its initial value. The initial value of the relaxation voltage, i.e., $V_r(0)$, is assumed to be zero in this study. This is a reasonable assumption since the magnitude of relaxation voltage is small (for most real battery chemistries), and drops to zero exponentially, when the battery is put to rest.

METHODOLOGY

3- Battery Parameter and State Estimation

The objective of the battery estimation problem here is to minimize the integral of the squared output error between the model and the actual system, over the unknown model parameters:

$$\min_{Q,R,C_r,R_r,SOC(0)} \quad J(t) = \frac{1}{2}\int_0^t \big(V(\tau) - V_m(\tau)\big)^2 d\tau \tag{57}$$

where V and V_m are the model and the measured battery voltages, respectively.

To facilitate the estimator design, as will be thoroughly explained in Section 3.2, we transfer most of the system parameters from the state equations to the output equation through a change of variables, and linearize the output equation around the current SOC value. The modified model based on which the estimator is designed is given by:

$$\begin{cases} \dot{x}_1(t) = I(t), & x_1(0) = 0 \\ \dot{x}_2(t) = -\sigma x_2(t) + I(t), & x_2(0) = 0 \end{cases} \tag{58}$$

$$V(t) = V_0(t) + \alpha C_0(t)\big(x_{10} + x_1(t)\big) + \beta x_2(t) + RI(t) \tag{59}$$

where

$$x_1(t) = Q \cdot \big(SOC(t) - SOC(0)\big) \tag{60}$$

$$x_2(t) = C_r V_r(t) \tag{61}$$

$$V_0(t) = V_{oc}\big(SOC(t)\big) - \left.\frac{\partial V_{oc}(SOC)}{\partial SOC}\right|_{@SOC=SOC(t)} SOC(t) \tag{62}$$

$$C_0(t) = \left.\frac{\partial V_{oc}(SOC)}{\partial SOC}\right|_{@SOC=SOC(t)} \tag{63}$$

$$x_{10} = Q \cdot SOC(0) \tag{64}$$

$$\alpha = \frac{1}{Q}, \qquad \beta = \frac{1}{C_r}, \qquad \sigma = \frac{1}{R_r C_r} \tag{65}$$

Eq. (58) shows that the modified state equations contain only one unknown parameter, i.e., σ. In this study, we treat the state equation containing the unknown parameter as a stochastic differential equation, and use the gPC theory to solve it. This process is discussed in detail in the next section.

3.1. Approximation of Battery Relaxation Dynamic Using gPC

Generalized polynomial chaos theory is a powerful method for solving stochastic differential equations [10, 11]. The application of gPC in online parameter estimation has been investigated in different research studies [12-14]. In this section, we apply the gPC theory to the battery estimation problem under study. The details of the gPC-based recursive parameter estimation method can be found in [14].

To solve the differential equation containing parameter σ, we treat this parameter as a uniformly distributed stochastic variable between two known upper and lower bounds, i.e., σ_{min} and σ_{max}, and then apply gPC theory on the second state equation in Eq. (58) to transfer σ to the output equation for recursive estimation. To simplify the process further, we map σ to another uniformly distributed stochastic variable $\boldsymbol{\xi} \in [-\mathbf{1}\ \ \mathbf{1}]$*:*

$$\boldsymbol{\sigma} = \boldsymbol{\sigma_0} + \boldsymbol{\xi}\boldsymbol{\sigma_1} \tag{66}$$

where

$$\boldsymbol{\sigma_0} = \frac{\mathbf{1}}{\mathbf{2}}(\boldsymbol{\sigma_{max}} + \boldsymbol{\sigma_{min}}) \tag{67}$$

$$\boldsymbol{\sigma_1} = \frac{\mathbf{1}}{\mathbf{2}}(\boldsymbol{\sigma_{max}} - \boldsymbol{\sigma_{min}}) \tag{68}$$

Following the gPC framework, we can write the solution of the stochastic differential equation as a function of the stochastic variable and time, and approximate it by a finite sum expansion as follows:

$$\boldsymbol{x_2(t)} = \boldsymbol{f(\xi, t)} \approx \sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{\varphi_i(\xi) q_i(t)} \tag{69}$$

where $\boldsymbol{\varphi_i}$*'s are a set of polynomial chaos functions, satisfying an orthogonality condition given by:*

$$\int_{\mathbf{-1}}^{\mathbf{1}} \boldsymbol{\varphi_i(\xi)\varphi_j(\xi)d\xi} = \mathbf{0}, \qquad \boldsymbol{if\ i \neq j} \tag{70}$$

Under certain assumptions [10], the approximation becomes exact as $\boldsymbol{N} \to \infty$*.*

Taking the time-derivative of Eq. (69) and replacing it (along with Eq. (66)) in Eq. (58) results in:

$$\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{\varphi_i(\xi)\dot{q}_i(t)} = -(\boldsymbol{\sigma_0} + \boldsymbol{\xi}\boldsymbol{\sigma_1}) \sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{\varphi_i(\xi) q_i(t)} + \boldsymbol{I(t)} \tag{71}$$

To proceed with the above equation, two methods from the gPC literature can be used: (i) The Galerkin projection method [10], and (ii) the collocation method [15]. For linear differential equations, the Glerkin approach has been widely used due to its simplicity and effectiveness. To apply the Galerkin projection method to Eq. (71), we first multiply the two sides of Eq. (71) by $\boldsymbol{\varphi_j(\xi)}$ *and then integrate over* $\boldsymbol{\xi}$*. This results in:*

$$\dot{q}_i(t)\int_{-1}^{1}\varphi_i^2(\xi)d\xi = -\sigma_0 q_i(t)\int_{-1}^{1}\varphi_i^2(\xi)d\xi - \sigma_1\sum_{j=1}^{N}q_j(t)\int_{-1}^{1}\xi\varphi_i(\xi)\varphi_j(\xi)d\xi + I(t)\int_{-1}^{1}\varphi_i(\xi)d\xi \quad (72)$$

These equations can be written in the matrix form as:

$$\dot{q}(t) = Aq(t) + BI(t) \quad (73)$$

where

$$q(t) = [q_1(t), q_2(t), \dots, q_N(t)]^T \quad (74)$$

$$A \in \mathbb{R}^{N\times N}, \qquad A_{ij} = -\sigma_0\delta_{ij} - \sigma_1\frac{\int_{-1}^{1}\xi\varphi_i(\xi)\varphi_j(\xi)d\xi}{\int_{-1}^{1}\varphi_i^2(\xi)d\xi} \quad (75)$$

$$B \in \mathbb{R}^{N\times 1}, \qquad B_i = \frac{\int_{-1}^{1}\varphi_i(\xi)d\xi}{\int_{-1}^{1}\varphi_i^2(\xi)d\xi} \quad (76)$$

where δ_{ij} is the Kronecker delta function, i.e., $\delta_{ij} = 1$, if $i = j$, and $\delta_{ij} = 0$ otherwise.

Eq. (73) is a deterministic differential equation which can be computed numerically in the real-time by choosing an orthogonal polynomial chaos basis and applying the current measurement signal. In the next section, we combine Eq. (73) with the original system model, i.e., Eq. (58), and derive a set of parameter update laws for the system unknown parameters using a recursive parameter estimation technique.

3.2. Recursive Battery Parameter Estimation

Using the gPC expansion of the battery relaxation dynamics, we can recast the system states as:

$$\begin{cases}\dot{x}_1(t) = I(t), & x_1(0) = 0\\ \dot{q}(t) = Aq(t) + BI(t), & q(0) = 0\end{cases} \quad (77)$$

Moreover, by replacing Eq. (69) into Eq. (59), we can rewrite the output equation as follows:

$$V(t) \approx V_0(t) + \alpha C_0(t)\big(x_{10} + x_1(t)\big) + \beta\sum_{i=1}^{N}\varphi_i(\xi)q_i(t) + RI(t) \quad (78)$$

Since our estimation problem uses an integral cost function (given by Eq. (57)) which can always grow with time and become numerically unbounded, we embed a forgetting factor term $\lambda \in (0\ 1]$ to give more emphasis to the most recent measurements:

$$\min_{\theta=[\alpha,R,\beta,\xi,x_{10}]} J(t) = \frac{1}{2}\int_0^t \lambda^{(t-\tau)}\left(V(\theta,\tau) - V_m(\tau)\right)^2 d\tau \tag{79}$$

One solution to the above estimation problem is to use a gradient descent-based parameter update law for the unknown parameters as follows:

$$\dot{\hat{\theta}}(t) = -\Gamma \frac{\partial J(t)}{\partial \theta}\Big|_{\theta=\hat{\theta}(t)} \tag{80}$$

where Γ is a diagonal estimator gain matrix, determining the convergence rate of the parameter estimates.

Applying Eq. (80) to Eq. (79) and using Eqs. (77) and (78) results in the following parameter update laws:

$$\begin{aligned}
\dot{\hat{\alpha}}(t) = -\Gamma_\alpha \Bigg(& \int_0^t \left(\lambda^{(t-\tau)}\left(V_0(\tau) - V_m(\tau)\right)C_0(\tau)x_1(\tau)\right)d\tau \\
& + \hat{x}_{10}(t)\int_0^t \left(\lambda^{(t-\tau)}\left(V_0(\tau) - V_m(\tau)\right)C_0(\tau)\right)d\tau \\
& + \hat{\alpha}(t)\int_0^t \left(\lambda^{(t-\tau)}C_0^2(\tau)x_1^2(\tau)\right)d\tau \\
& + 2\hat{\alpha}(t)\hat{x}_{10}(t)\int_0^t \left(\lambda^{(t-\tau)}C_0^2(\tau)x_1(\tau)\right)d\tau \\
& + \hat{\alpha}(t)x_{10}^2(t)\int_0^t \left(\lambda^{(t-\tau)}C_0^2(\tau)\right)d\tau \\
& + \sum_{i=1}^N \left\{\hat{\beta}(t)\varphi_i\left(\hat{\xi}(t)\right)\int_0^t \left(\lambda^{(t-\tau)}q_i(\tau)C_0(\tau)x_1(\tau)\right)d\tau\right\} \\
& + \sum_{i=1}^N \left\{\hat{\beta}(t)\hat{x}_{10}(t)\varphi_i\left(\hat{\xi}(t)\right)\int_0^t \left(\lambda^{(t-\tau)}q_i(\tau)C_0(\tau)\right)d\tau\right\} \\
& + \hat{R}(t)\int_0^t \left(\lambda^{(t-\tau)}I(\tau)C_0(\tau)x_1(\tau)\right)d\tau \\
& + \hat{R}(t)\hat{x}_{10}(t)\int_0^t \left(\lambda^{(t-\tau)}I(\tau)C_0(\tau)\right)d\tau\Bigg)
\end{aligned} \tag{81}$$

$$\begin{aligned}
\dot{\hat{R}}(t) = -\Gamma_r \Bigg(& \int_0^t \left(\lambda^{(t-\tau)}\left(V_0(\tau) - V_m(\tau)\right)I(\tau)\right)d\tau + \hat{\alpha}(t)\hat{x}_{10}(t)\int_0^t \left(\lambda^{(t-\tau)}C_0(\tau)I(\tau)\right)d\tau \\
& + \hat{\alpha}(t)\int_0^t \left(\lambda^{(t-\tau)}C_0(\tau)x_1(\tau)I(\tau)\right)d\tau \\
& + \sum_{i=1}^N \left\{\hat{\beta}(t)\varphi_i\left(\hat{\xi}(t)\right)\int_0^t \left(\lambda^{(t-\tau)}q_i(\tau)I(\tau)\right)d\tau\right\} \\
& + \hat{R}(t)\int_0^t \left(\lambda^{(t-\tau)}I^2(\tau)\right)d\tau\Bigg)
\end{aligned} \tag{82}$$

$$\dot{\hat{\boldsymbol{\beta}}}(t) = -\boldsymbol{\Gamma}_{\boldsymbol{\beta}}\Bigg(\sum_{i=1}^{N}\left\{\boldsymbol{\varphi}_i\left(\hat{\boldsymbol{\xi}}(t)\right)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\boldsymbol{q}_i(\tau)\left(\boldsymbol{V}_0(\tau)-\boldsymbol{V}_m(\tau)\right)\right)d\tau\right\}$$
$$+\sum_{i=1}^{N}\left\{\hat{\boldsymbol{\alpha}}(t)\boldsymbol{\varphi}_i\left(\hat{\boldsymbol{\xi}}(t)\right)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\boldsymbol{q}_i(\tau)\boldsymbol{C}_0(\tau)\boldsymbol{x}_1(\tau)\right)d\tau\right\}$$
$$+\sum_{i=1}^{N}\left\{\hat{\boldsymbol{\alpha}}(t)\hat{\boldsymbol{x}}_{10}(t)\boldsymbol{\varphi}_i\left(\hat{\boldsymbol{\xi}}(t)\right)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\boldsymbol{q}_i(\tau)\boldsymbol{C}_0(\tau)\right)d\tau\right\} \qquad (83)$$
$$+\sum_{i=1}^{N}\sum_{j=1}^{N}\left\{\hat{\boldsymbol{\beta}}(t)\boldsymbol{\varphi}_i\left(\hat{\boldsymbol{\xi}}(t)\right)\boldsymbol{\varphi}_j\left(\hat{\boldsymbol{\xi}}(t)\right)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\boldsymbol{q}_i(\tau)\boldsymbol{q}_j(\tau)\right)d\tau\right\}$$
$$+\sum_{i=1}^{N}\left\{\hat{\boldsymbol{R}}(t)\boldsymbol{\varphi}_i\left(\hat{\boldsymbol{\xi}}(t)\right)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\boldsymbol{q}_i(\tau)\boldsymbol{I}(\tau)\right)d\tau\right\}\Bigg)$$

$$\dot{\hat{\boldsymbol{\xi}}}(t) = -\boldsymbol{\Gamma}_{\boldsymbol{\xi}}\Bigg(\sum_{i=1}^{N}\left\{\hat{\boldsymbol{\beta}}(t)\boldsymbol{\varphi}_i'\left(\hat{\boldsymbol{\xi}}(t)\right)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\boldsymbol{q}_i(\tau)\left(\boldsymbol{V}_0(\tau)-\boldsymbol{V}_m(\tau)\right)\right)d\tau\right\}$$
$$+\sum_{i=1}^{N}\left\{\hat{\boldsymbol{\alpha}}(t)\hat{\boldsymbol{\beta}}(t)\boldsymbol{\varphi}_i'\left(\hat{\boldsymbol{\xi}}(t)\right)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\boldsymbol{q}_i(\tau)\boldsymbol{C}_0(\tau)\boldsymbol{x}_1(\tau)\right)d\tau\right\}$$
$$+\sum_{i=1}^{N}\left\{\hat{\boldsymbol{\alpha}}(t)\hat{\boldsymbol{x}}_{10}(t)\hat{\boldsymbol{\beta}}(t)\boldsymbol{\varphi}_i'\left(\hat{\boldsymbol{\xi}}(t)\right)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\boldsymbol{q}_i(\tau)\boldsymbol{C}_0(\tau)\right)d\tau\right\} \qquad (84)$$
$$+\sum_{i=1}^{N}\sum_{j=1}^{N}\left\{\hat{\boldsymbol{\beta}}^2(t)\boldsymbol{\varphi}_i\left(\hat{\boldsymbol{\xi}}(t)\right)\boldsymbol{\varphi}_j'\left(\hat{\boldsymbol{\xi}}(t)\right)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\boldsymbol{q}_i(\tau)\boldsymbol{q}_j(\tau)\right)d\tau\right\}$$
$$+\sum_{i=1}^{N}\left\{\hat{\boldsymbol{R}}(t)\hat{\boldsymbol{\beta}}(t)\boldsymbol{\varphi}_i'\left(\hat{\boldsymbol{\xi}}(t)\right)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\boldsymbol{q}_i(\tau)\boldsymbol{I}(\tau)\right)d\tau\right\}\Bigg)$$

where $\boldsymbol{\varphi}_i'\left(\hat{\boldsymbol{\xi}}(t)\right) = \frac{\partial\boldsymbol{\varphi}_i(\xi)}{\partial\xi}\Big|_{\xi=\hat{\boldsymbol{\xi}}(t)}$.

$$\dot{\hat{\boldsymbol{x}}}_{10}(t) = -\boldsymbol{\Gamma}_{x_{10}}\Bigg(\hat{\boldsymbol{\alpha}}(t)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\left(\boldsymbol{V}_0(\tau)-\boldsymbol{V}_m(\tau)\right)\boldsymbol{C}_0(\tau)(\tau)\right)d\tau$$
$$+\ \hat{\boldsymbol{\alpha}}^2(t)\hat{\boldsymbol{x}}_{10}(t)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\boldsymbol{C}_0^2(\tau)\right)d\tau+\ \hat{\boldsymbol{\alpha}}^2(t)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\boldsymbol{C}_0^2(\tau)\boldsymbol{x}_1(\tau)\right)d\tau$$
$$+\sum_{i=1}^{N}\left\{\hat{\boldsymbol{\alpha}}(t)\hat{\boldsymbol{\beta}}(t)\boldsymbol{\varphi}_i\left(\hat{\boldsymbol{\xi}}(t)\right)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\boldsymbol{q}_i(\tau)\boldsymbol{C}_0(\tau)\right)d\tau\right\} \qquad (85)$$
$$+\hat{\boldsymbol{\alpha}}(t)\hat{\boldsymbol{R}}(t)\int_0^t\left(\boldsymbol{\lambda}^{(t-\tau)}\boldsymbol{I}(\tau)\boldsymbol{C}_0(\tau)\right)d\tau\Bigg)$$

From Eqs. (27)-(31) and using Eqs. (6), (10) and (11), the original battery parameter and SOC estimates are obtained as:

$$\widehat{\boldsymbol{Q}}(\boldsymbol{t}) = \frac{\mathbf{1}}{\widehat{\boldsymbol{\alpha}}(\boldsymbol{t})}, \qquad \widehat{\boldsymbol{C}}_{\boldsymbol{r}}(\boldsymbol{t}) = \frac{\mathbf{1}}{\widehat{\boldsymbol{\beta}}(\boldsymbol{t})}, \qquad \widehat{\boldsymbol{R}}_{\boldsymbol{r}} = \frac{\widehat{\boldsymbol{\beta}}(\boldsymbol{t})}{\boldsymbol{\sigma_0} + \boldsymbol{\sigma_1}\widehat{\boldsymbol{\xi}}(\boldsymbol{t})} \tag{86}$$

$$\boldsymbol{S\widehat{O}C}(\boldsymbol{t}) = \widehat{\boldsymbol{\alpha}}(\boldsymbol{t})\big(\boldsymbol{x_1}(\boldsymbol{t}) + \widehat{\boldsymbol{x}}_{\mathbf{10}}(\boldsymbol{t})\big) \tag{87}$$

Note that the estimate of battery resistance, R, is directly computed from Eq. (82).

Remark 1: In the parameter update laws, $V_0(t)$ and $C_0(t)$ are obtained by linearizing $V_{oc}(SOC)$ around the most recent estimate of battery SOC. This approximation can be initially inaccurate since the SOC estimate can be significantly far from the actual battery SOC. However, as the SOC estimate converges to the actual SOC, this approximation becomes more accurate, leading to reliable steady-state parameter estimation.

Remark 2: All of the integral coefficients in the parameter update laws can be computed recursively, thereby, making the proposed estimation scheme a viable approach for real-time applications.

FINDINGS

4- Numerical Evaluation of the gPC-based Battery Estimator

In this section, we evaluate the developed estimation method using numerical simulations. We first obtain a set of realistic battery parameter values using experimental data, to develop a representative battery model. We then pretend that the parameters of this model are not accurately known, and try to estimate them using the developed estimator. To apply a realistic current profile, we use a previously-developed plug-in hybrid electric vehicle (PHEV) model to obtain battery current variation under a Federal drive cycle simulation.

4.1. Development of a Representative Battery Model

In this section, we use experimental battery data to obtain a set of realistic parameter values and open-circuit voltage function for the battery model described in Section 2. Using a battery-in-the loop (BIL) test bed shown in **Figure 24***, we cycle an A123 Systems $LiFePO_4$ battery cell to obtain its open circuit voltage as a function of SOC. The main component of the BIL test bed include a pair of Sorensen power supply and programmable electronic load for battery cycling, and a DS1103 dSPACE board for controlling the process in the Matlab/Simulink environment.*

The nominal charge capacity and voltage of the battery are 2 Amp-hours and 3.3 V, respectively. **Figure 25***(a) shows the battery cell's voltage variation between 2.2-3.8 V as a function SOC at the constant rate of 0.1C ($\pm$0.2A). Note that we have assumed this voltage range corresponds to the SOC range of 0-100%, since the operation of the battery cell below 2.2 V and above 3.8 V can result in severe damage to the*

battery. We further limit the model to operate between 10-90% of the full SOC range to avoid steep changes in the voltage due to practical considerations. **Figure 24***(b) shows the charge/discharge voltage curves between 10-90% SOC range, and a 6th order polynomial fit to the average charge/discharge data. We use this polynomial as a representative V_{oc} curve in the equivalent circuit model.*

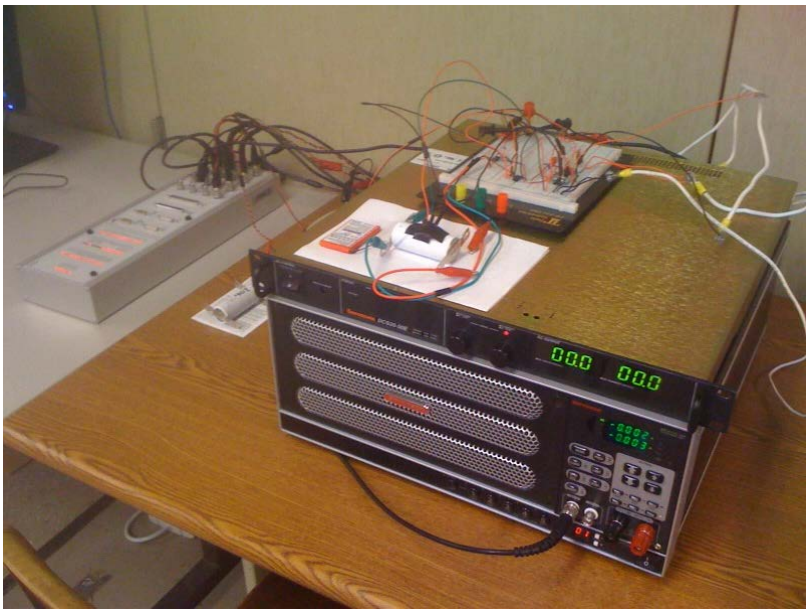

FIGURE 24 BATTERY-IN-THE-LOOP TEST BED COMPRISING A PAIR OF SORENSON POWER SUPPLY AND PROGRAMMABLE ELECTRONIC LOAD, AND A DS1103 DSPACE CONTROL BOARD.

To obtain a set of representative parameter values for the battery model, we apply a pulse current profile as shown in Figure 26*(a) to the battery cell resting at 3.2 V. The resultant voltage response, as well as the response of a hand-tuned model are shown in* Figure 26*(b). As seen from the figure, the battery model and the actual battery cell respond very similarly to the applied current signal. We use the hand-tuned battery model as a representative model to evaluate the estimator.*

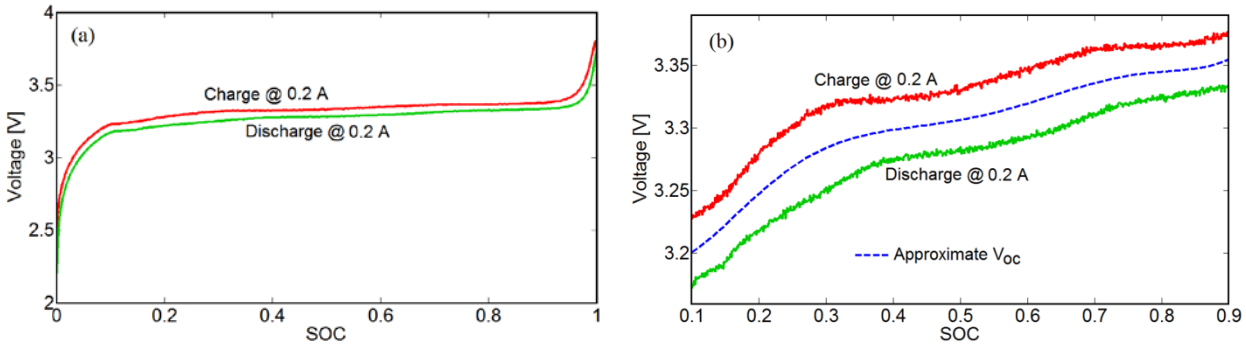

FIGURE 25 LOW-CURRENT BATTERY VOLTAGE AS A FUNCTION OF SOC: (A) FULL SOC RANGE, AND (B) 10-90% SOC RANGE WITH A 6TH ORDER POLYNOMIAL APPROXIMATION REPRESENTING THE OPEN-CIRCUIT VOLTAGE.

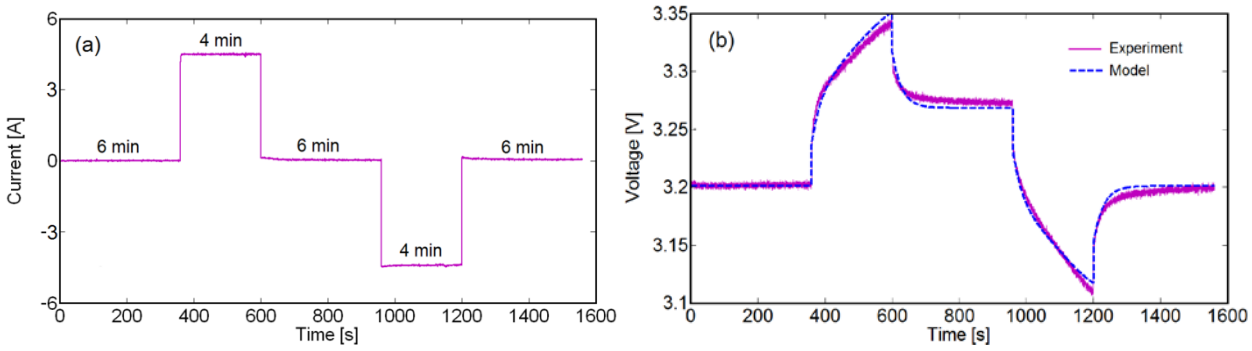

FIGURE 26 BATTERY MODEL EVALUATION: (A) APPLIED CURRENT, (B) THE ACTUAL BATTERY CELL, AND THE HAND-TUNED MODEL VOLTAGE RESPONSES.

4.2. Generating Current Profiles

In this section, we use a previously-developed PHEV power management system model [16] to generate a battery current profile under realistic driving conditions. In the previous study, the authors have used stochastic dynamic programming to develop an optimal control policy for a power split PHEV model similar in configuration and dynamics to Toyota Prius. The power management system has been optimized for a stochastic representation of real-world driving conditions, resulting in a control policy that blends energy from fuel and the battery. The details of the system dynamics and control development can be found in [16]. In this study, we only provide a simulation of the PHEV model with 16 kWh battery.

Figure 27 *shows the FTP-72 (Federal Test Procedure) drive cycle, and the resultant battery current trajectory obtained from simulating the PHEV model. The current trajectory is dominantly negative, which results in the depletion of the battery. However, due to regenerative breaking, we can also see positive spikes in the current, leading to temporary battery charging.*

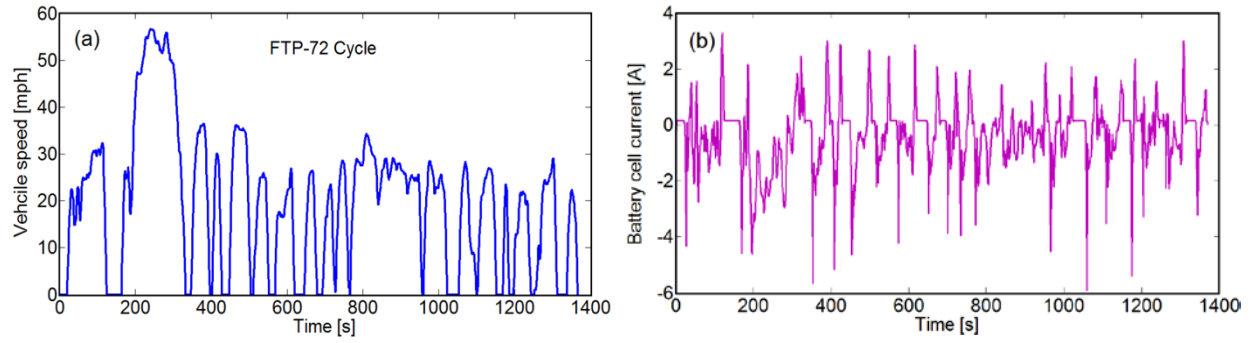

FIGURE 27 PHEV MODEL SIMULATION: (A) FTP-72 CYCLE, AND (B) CELL-LEVEL BATTERY CURRENT.

To enable longer simulation times for the battery estimator, we use the obtained current profile and generate a full charge/discharge cycle by applying the same current trajectory back-to-back until the battery SOC reaches its lower limit. Then we apply a constant positive current to charge the battery until it reaches the upper SOC limit. This cycle can be repeated for as many times as the battery estimator

simulation time requires. Figure 28 *shows the obtained current trajectory for nearly three consecutive charge/discharge cycles between 20-80% SOC limits.*

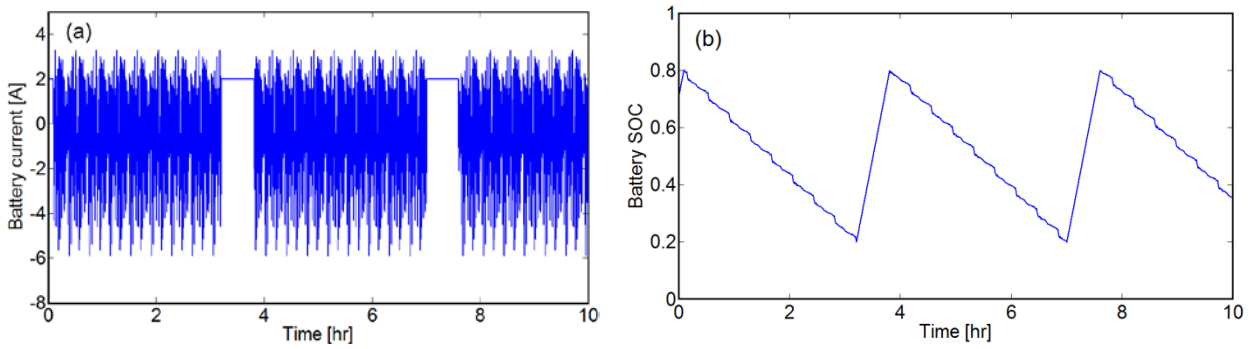

FIGURE 28 BATTERY CYCLING BETWEEN 20-80% SOC USING REPEATED DRIVE CYCLE-BASED DISCHARGING AND CONSTANT-CURRENT CHARGING AT 2 AMPERES: (A) CURRENT PROFILE AND (B) SOC TRAJECTORY FOR 10 HOURS.

4.3. Battery Estimator Simulation

In this section, we use the representative battery model and the current profile obtained in the previous sections to evaluate the developed battery state/parameter estimation method. Figure 29 *shows the block diagram of the estimation strategy. The current is simultaneously applied to both the representative battery model and the estimator. The estimator parameters are initialized to be 10-20% different from those of the representative model. The voltage obtained from the representative model is also fed to the estimator as the measured voltage. To create a more realistic simulation, we also add two random Gaussian noise signals to the current and voltage measurements. The noise standard deviations are set to 10 mA for the current signal and 2 mV for the voltage signal. These values are set at roughly twice the actual noise standard deviations measured from the BIL test bed, to account for additional noise contributions in real-world environments, such as moving vehicles.*

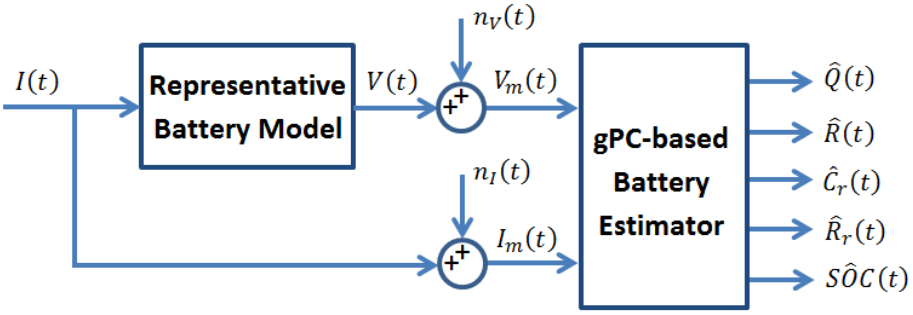

FIGURE 29 BATTERY ESTIMATOR SIMULATION DIAGRAM

The parameter values used in the representative battery model and the initial guesses for the estimator are summarized in **Table 7**. *We use the same V_{oc} function for both the model and the estimator. To simulate the model and the estimator, we discretize them using zero-order hold method and set the time step to 1 s and the forgetting factor value to 0.99999. Based on this λ value, the weight of a*

measurement from 20 hours ago is half of that of the current measurement in the estimator cost function.

TABLE 7 LIST OF PARAMETER VALUES FOR THE MODEL AND THE ESTIMATOR.

Parameter [unit]	*Value used in the model*	*Initial guess for the estimator*
Q [A.hr]	*2*	*2.2*
R [mΩ]	*8*	*10*
C_r [F]	*2700*	*3000*
R_r [mΩ]	*11*	*10*
SOC(0)	*0.7*	*0.5*

In the simulations of this section, we use the Legendre polynomials as the polynomial chaos basis for the estimator. To demonstrate the impacts of the gPC expansion order, we simulate the estimator separately for 2 and 3 gPC expansions terms, i.e., N = 2 and N = 3. The first three terms of the Legendre polynomials are given by:

$$\boldsymbol{\varphi_1(\xi) = 1}, \quad \boldsymbol{\varphi_2(\xi) = \xi}, \quad \boldsymbol{\varphi_3(\xi) = \frac{1}{2}\left(3\xi^2 - 1\right)} \tag{88}$$

The inner product of two Legendre polynomials is given by:

$$\int_{-1}^{1} \boldsymbol{\varphi_i(\xi)\varphi_j(\xi)d\xi = \frac{2}{2i+1}\delta_{ij}} \tag{89}$$

The estimator gain values can affect the stability of the estimator. Large estimator gain values can result in instability, and small gain values can slow down the estimation convergence rate. We choose the gain values through a trial-and-error process (see **Table 8***). Note that we have deliberately chosen the estimator gain Γ_{x10} to be large, since it directly impacts the convergence rate of SOC estimate, to which the convergence of other parameters is tightly related. Moreover, most battery management systems require a fast and accurate estimation of SOC for the safe and efficient operation of the battery.*

TABLE 8 ESTIMATOR GAIN VALUES.

Estimator gain	*Value*
Γ_α	2×10^{-14}
Γ_R	10^{-8}
Γ_β	10^{-10}
Γ_ξ	10^{-5}
Γ_{x10}	100

The battery parameter estimation results for parameters Q, R, C_r and R_r are shown in Figure 30 *for a 10-day time window. The estimator is turned off for the first day, to show the transient part of the estimation more clearly. As seen from the results, the estimator with 2 gPC expansions terms estimates parameters Q and R quite well, but fails to accurately estimate the battery relaxation parameters, C_r and R_r. The estimator with 3 gPC expansion terms, however, converges to a close neighborhood of the actual model parameters for all four parameters. Including further expansion terms in the gPC approximation may result in marginal improvements in the estimator's accuracy, but at the cost of significant increase in the numerical and computational complexity.*

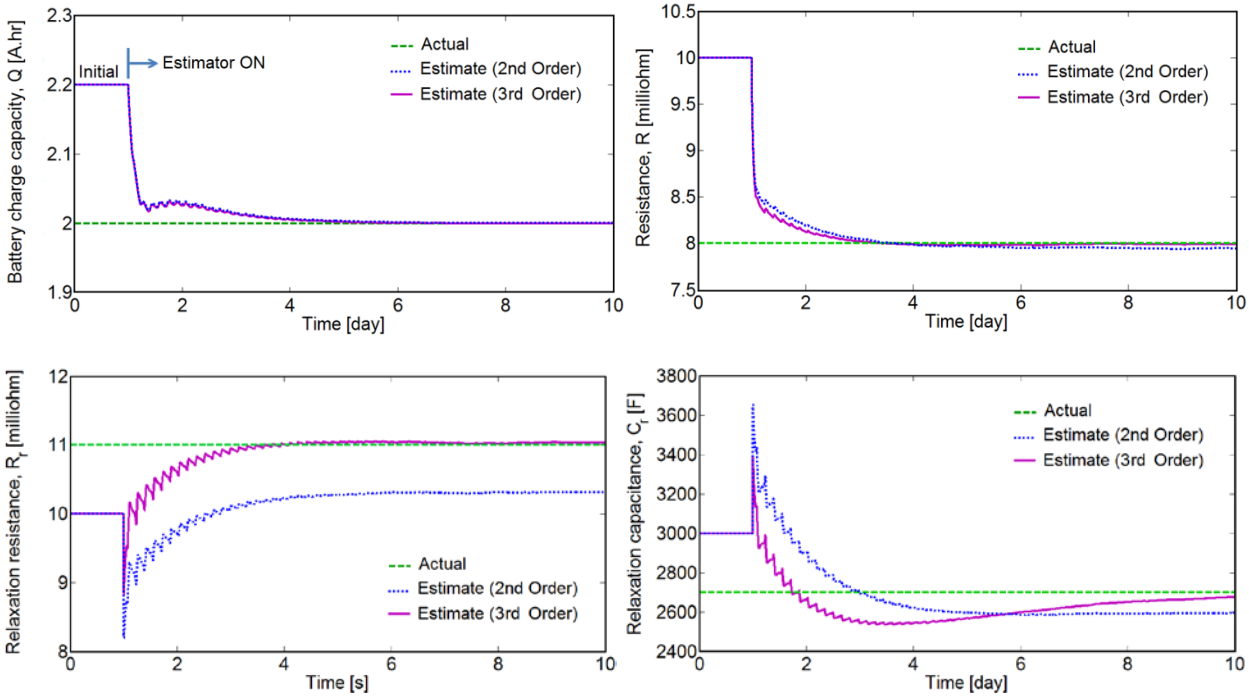

FIGURE 30 BATTERY PARAMETER ESTIMATION USING 2ND- AND 3RD-ORDER GPC-BASED RECURSIVE ESTIMATOR.

The SOC estimation and output voltage absolute error trajectories for the 3rd-order estimator are shown in **Figure 31**. *We can see that the SOC estimation error has a sudden drop initially, and then converges to zero at nearly the same rate as the other parameters. The voltage error plot shows that the estimator has been able to suppress the output error to nearly the noise level over time.*

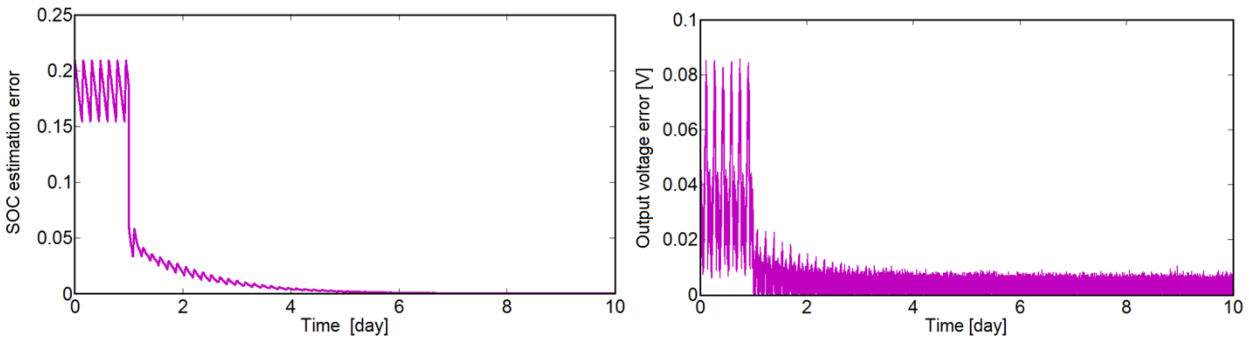

FIGURE 31 BATTERY SOC ESTIMATION AND OUTPUT VOLTAGE ERROR TRAJECTORIES.

4.4. Long-term Battery Health Estimation

In this section, we evaluate the estimator's capability in estimating battery state-of-health in the long term. To carry out this simulation, we use a previously developed battery degradation model [17], where the rate at which the charge capacity of the battery degrades is related to the battery current and voltage through a nonlinear map. This map is obtained by cycling the A123 System's $LiFePO_4$ battery cells at various charge/discharge cycles for several months and measuring the charge capacity of the cells at certain time intervals. The obtained experimental data have then been used to construct a surface response function, as shown in **Figure 32***. The details of the model can be found in [17]. Here, we use this model to evaluate the long-term performance of the proposed estimator, noting that this model applies to the battery charge capacity only, and the other parameters are kept constant in the simulation.*

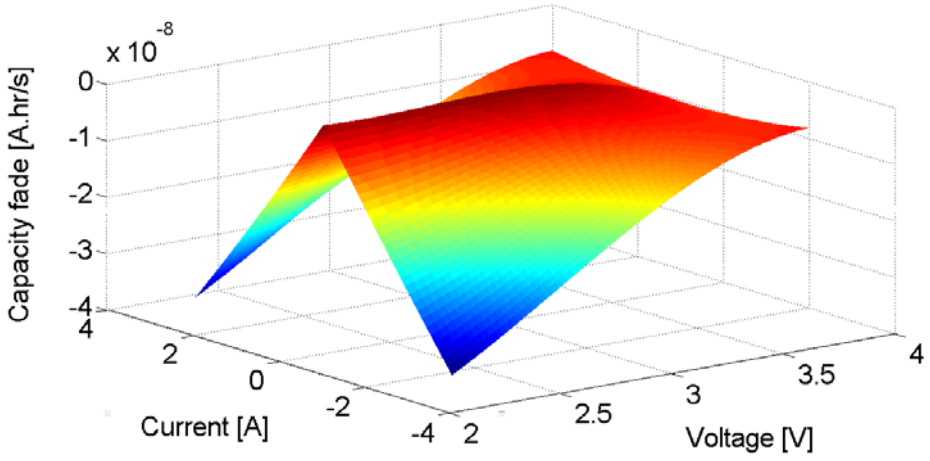

FIGURE 32 A123 SYSTEMS LIFEPO4 BATTERY DEGRADATION MAP [17].

Figure 33 shows the long-term battery state-of-health estimation results for a 100-day time window. The same initial values as those of the first simulation are used here for the estimator. The parameter update laws are turned off for the first 10 days to show the convergence of the parameters more clearly when

the estimator is turned on. As we can see from the charge capacity and resistance estimation plots, the estimator is capable of both converging to the actual parameter values and also tracking them despite their possible variations over time. This simulation further endorses the effectiveness of the proposed battery estimation scheme for utilization in real-world applications, such as electric vehicles.

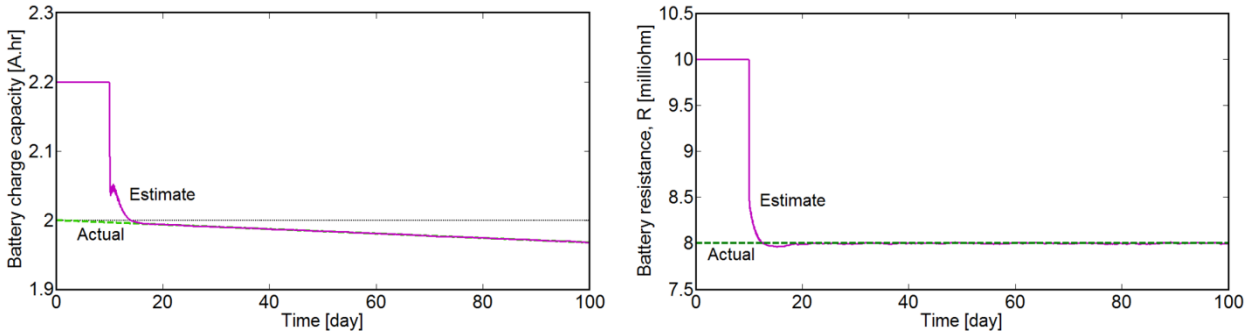

FIGURE 33 LONG-TERM BATTERY STATE-OF-HEALTH ESTIMATION SIMULATION.

Future work will focus on applying the proposed estimation method to real battery cells under various charging and discharging conditions. Investigating the effects of temperature variation on the estimation accuracy, and including it in the estimator for possible improvements, will also be considered.

CONCLUSIONS AND RECOMMENDATIONS

In this research, we developed a battery parameter and state estimation framework using the generalized polynomial chaos (gPC) theory and the gradient-based recursive estimation technique. We used an equivalent-circuit battery model with a first-order relaxation term as a representative battery model for estimator design. We then used a change of variables and the gPC theory to move all of the system parameters to the output equation. We formulated a parameter estimation problem based on the integral of the output error squares, and used the gradient descent-based recursive estimation method to obtain a set of parameter update laws for the estimator. Simulation results show that the proposed estimator, with at least three gPC expansion terms, can converge to the actual system parameters. Moreover, it is able to track the battery capacity fade as the battery undergoes gradual degradation over time due to cycling.

REFERENCES

[1] *F. Codeca, S. M. Savaresi, and G. Rizzoni, "On battery state of charge estimation: A new mixed algorithm," IEEE International Conference on Control Application, pp. 102-107, Sept. 2008.*

[2] *M. Verbrugge and E. Tate, "Adaptive state of charge algorithm for nickel metal hydride batteries including hysteresis phenomena," Journal of Power Sources, 126, pp. 236-249, 2004.*

[3] *G. L. Pett, "Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 3. State and parameter estimation," Journal of Power Sources, 134, pp. 277-92, 2004.*

[4] *D. V. Do, C. Forgez, K. El-Kadri-Benkara, and G. Friedrich, "Impedance observer for a Li-ion battery using Kalman filter," IEEE Transactions on Vehicular Technology, 58, pp. 3930-3937, 2009.*

[5] D. D. Domenico, A. Stefanopoulou, and G. Fiengo, "Lithium-ion battery state of charge and critical surface charge estimation using an electrochemical model-based extended Kalman filter," Journal of Dynamic Systems, Measurement, and Control, 132, pp. 061302, 2010.

[6] N. A. Chaturvedi, R. Klein, J. Christensen, J. Ahmed, and A. Kojic, "Algorithms for advanced battery-management systems," IEEE Control Systems Magazine, 30, pp. 49-68, 2010.

[7] C. R. Gould, C. M. Bingham, D. A. Stone, and P. Bentley, "New battery model and state-of-health determination through subspace parameter estimation and state-observer techniques," IEEE Transactions on Vehicular Technology, 58, pp. 3905-3916, 2009.

[8] Y. H. Chiang, W. Y. Sean, and J. C. Ke, "Online estimation of internal resistance and open-circuit voltage of lithium-ion batteries in electric vehicles," Journal of Power Sources, 196, pp. 3921-3932, 2011.

[9] S. J. Moura, N. A. Chaturvedi, and M. Krstić, "PDE estimation techniques for advanced battery management systems – Part II SOH identification," 2012 American Control Conference, Montreal, June 2012.

[10] D. Xiu and G. E. Karniadakis, "The Wiener-Askey polynomial chaos for stochastic differential equations," SIAM Journal on Scientific Computing, 24, pp. 619-644, 2002.

[11] R. G. Ghanem and P. D. Spanos. Stochastic finite elements: A spectral approach, New York, Springer-Verlag, 1991.

[12] E. D. Blanchard, A. Sandu, and C. Sandu, "A polynomial chaos-based Kalman filter approach for parameter estimation of mechanical systems," Journal of Dynamic Systems, Measurement and Control, 132, 061404 (18 pp.), 2010.

[13] A. Smith, A. Monti and F. Ponci, "Indirect measurements via polynomial chaos observer," IEEE Transactions on Instrumentation and Measurement, 56, pp. 743-752, 2007.

[14] B. L. Pence, H. K. Fathy, and J. L. Stein, "Recursive maximum likelihood parameter estimation for state space systems using polynomial chaos theory," Automatica, 47, pp. 2420-2424, 2011.

[15] D. Xiu, "Efficient collocation approach for parametric uncertainty analysis," Communications in Computational Physics, 2, pp. 293-309, 2007.

[16] S. J. Moura, H. K. Fathy, D. S. Callaway, and J. L. Stein, "A stochastic optimal control approach for power management in plug-in hybrid electric vehicles," IEEE Transactions on Control Systems Technology, 99, pp. 1-11, 2010.

[17] J. C. Forman, "Minimally invasive characterization of lithium iron phosphate battery electrochemical and health models using Fisher information and optimal experimental design," Ph.D. Dissertation, Department of Mechanical Engineering, University of Michigan, 2012.

PART 4: STOCHASTIC TRAFFIC FLOW MODELING

PROBLEM

Introduction and Overview

Macroscopic traffic flow models can be used to study the relationships between macroscopic traffic flow parameters such as flow, density, and speed by assuming that the flow of traffic can be compared to the flow of continuum fluids. The most famous continuum traffic flow model, known as the LWR model, was developed concurrently by Lighthill and Whitham (1955) and Richards (1956). The LWR model is a first-order model and assumes that the number of vehicles between any two points is conserved if there are no sources or sinks. If $\boldsymbol{q}(\boldsymbol{x},\boldsymbol{t})$ denotes the flow at any point $\boldsymbol{x}$ and time $\boldsymbol{t}$ and $\boldsymbol{k}(\boldsymbol{x},\boldsymbol{t})$ denotes the density at any point $\boldsymbol{x}$ and time $\boldsymbol{t}$, the conservation of flow can be represented by the following equation:

$$\frac{\partial \boldsymbol{k}}{\partial \boldsymbol{t}} + \frac{\partial \boldsymbol{q}}{\partial \boldsymbol{x}} = \boldsymbol{0} \qquad (90)$$

In addition to the conservation equation, an equilibrium relationship is assumed, $\boldsymbol{q}(\boldsymbol{x},\boldsymbol{t}) = \boldsymbol{Q}(\boldsymbol{k}(\boldsymbol{x},\boldsymbol{t}))$ between the flow rate and the density and boundary conditions will need to be satisfied. Some of the popular equilibrium relationships are based on the Greenshield, Greenberg, and Underwood models (May, 1990). According to the equilibrium relationships, the flow at any point on the road is a function of the concentration or density at that point. Changes in flow are instantaneously propagated in the traffic stream as kinematic waves.

If $\boldsymbol{C}$ is a curve connecting two points $(\boldsymbol{t_1},\boldsymbol{x_1})$ and $(\boldsymbol{t_2},\boldsymbol{x_2})$ and if $\boldsymbol{N}(\boldsymbol{t_1},\boldsymbol{x_1})$ and $\boldsymbol{N}(\boldsymbol{t_2},\boldsymbol{x_2})$ denote the cumulative number of vehicles observed at the two points, then the conservation equation can also be written in the integral form as:

$$\boldsymbol{N}(\boldsymbol{t_2},\boldsymbol{x_2}) - \boldsymbol{N}(\boldsymbol{t_1},\boldsymbol{x_1}) = \int_{\boldsymbol{C}} \boldsymbol{q}\boldsymbol{dt} - \boldsymbol{k}\boldsymbol{dx} \qquad (91)$$

One way to easily solve the LWR model for homogenous freeway segments is to determine curves along which the density $\boldsymbol{k}(\boldsymbol{x},\boldsymbol{t})$ is constant. These curves are called characteristic curves. The conservation law can be rewritten as

$$\frac{\partial \boldsymbol{k}}{\partial \boldsymbol{t}} + \frac{\boldsymbol{dq}}{\boldsymbol{dk}}\frac{\partial \boldsymbol{k}}{\partial \boldsymbol{x}} = \boldsymbol{0} \qquad (92)$$

For the above model, the characteristic curves are straight lines. Consider an observer traveling in a straight line such that

$$\frac{\boldsymbol{dx}}{\boldsymbol{dt}} = \frac{\boldsymbol{dq}}{\boldsymbol{dk}} \qquad (93)$$

The above expression implies that the observer travels at a speed equal to the speed of the wave.

$$\frac{\boldsymbol{dk}}{\boldsymbol{dt}} = \frac{\partial \boldsymbol{k}}{\partial \boldsymbol{t}} + \frac{\partial \boldsymbol{k}}{\partial \boldsymbol{x}}\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{t}} = \frac{\partial \boldsymbol{k}}{\partial \boldsymbol{t}} + \frac{\boldsymbol{dq}}{\boldsymbol{dk}}\frac{\partial \boldsymbol{k}}{\partial \boldsymbol{x}} = \boldsymbol{0} \qquad (94)$$

Thus along the characteristic line, the observer will not observe a change in density. So if the initial density is known (from the boundary conditions), then the concentration on any point on the line can be determined. The LWR model has been applied to evaluate several applications in relation to traffic dynamics. The LWR model is accurate in modeling several observed traffic flow phenomena, such as shockwaves. However, the LWR models have several deficiencies also. The LWR model is unable to model differences in driver behavior, especially in uncongested, light-traffic conditions. As the system evolution is described by one parameter density $\boldsymbol{k}(\boldsymbol{x},\boldsymbol{t})$*, it cannot differentiate between fast-moving and slow-moving traffic. The model assumes instantaneous deceleration or acceleration when a vehicle reaches a shockwave. The shockwaves do not have any width or time. The LWR model does not account for driver anticipation and reaction time. Drivers have the ability to look ahead and adjust their driving behavior even before the shockwaves reach their vehicle. The LWR model is unable to model this phenomenon. The flow density relationship may have oscillations and there might not exist a perfect equilibrium flow density relationship.*

In order to address the issue of driver reaction and anticipation, higher-order models were developed. Lighthill and Whitham (1955) proposed the following second-order model:

$$\frac{\partial q}{\partial t} + \boldsymbol{C}\frac{\partial q}{\partial x} + \boldsymbol{T}\frac{\partial^2 q}{\partial t^2} - \boldsymbol{D}\frac{\partial^2 q}{\partial x^2} = \mathbf{0} \qquad (95)$$

In the above equation, $\boldsymbol{C}$ *is the speed of disturbance shock waves,* $\boldsymbol{T}$ *is the reaction time, and* $\boldsymbol{D}$ *the diffusion coefficient, which represents how vehicles can react to changes in traffic conditions that are not local.*

Payne (1971) developed a second-order traffic flow model from car-following theory, which combines the following momentum equation with the conservation equation:

$$\frac{\partial v}{\partial t} + \frac{v\partial v}{\partial x} = -\frac{v - v^*(k)}{T} - \frac{v^*(k)}{2Tk}\frac{\partial k}{\partial x} \qquad (96)$$

In the above equation, $\boldsymbol{v}^*(\boldsymbol{k})$ *denotes the equilibrium speed flow relationship. The assumption in the Payne model is that the speed at a location is affected with a lag* $\boldsymbol{T}$*. The second term on the left-hand side captures the change of speed due to the convection effect, the first term on the right-hand side captures the driver adaptation or adjustment to equilibrium speed, and the second term on the right-hand side captures the driver anticipation, which depends on the concentration gradient. One of the main issues with the above model is the presence of density in the denominator. This makes it difficult to solve using the finite difference method, especially for small densities. Papageorgiou (1983) accounted for this computational instability at small densities by adding a* $\boldsymbol{K}$ *to the density in the denominator of the anticipation term. Ross (1988) developed a simple variant of the model based on the assumption that drivers always desire to travel at the free-flow speed:*

$$\frac{\partial v}{\partial t} + \frac{v\partial v}{\partial x} = \frac{v_f - v}{T} \qquad (97)$$

where $\boldsymbol{v_f}$ *is the free-flow speed. Kuhn (1985) further modified Payne's (1971) momentum equation by the addition of a new anticipation term.*

$$\frac{\partial v}{\partial t} + \frac{v \partial v}{\partial x} = \frac{v^*(k) - v}{T} - C_0^2 \frac{\partial k}{\partial x} + \nu \frac{\partial^2 u}{\partial x^2} \quad (98)$$

In the above equation, ν corresponds to the viscosity coefficient and the C_0 term relates to the elasticity of car following behavior. Michaelopoulos (1993) developed a new higher-order formulation to account for the impact of geometric aspects of roadway systems on traffic flow and traffic friction caused by interaction between ramp flows and mainline traffic.

$$\frac{du}{dt} = \frac{\Phi}{T}\left[u_f(x) - u\right] - \mu k^{\epsilon} g - \nu k^{\beta} \frac{\partial k}{\partial x} \quad (99)$$

In the above equation, Φ is a flag which indicates if there is a change in free-flow speed in the current location when compared to upstream location and the relaxation time T is assumed to vary with congestion. The second term in the right-hand side corresponds to the friction term, while the third term is the anticipation term.

Daganzo (1995) critically reviewed all second-order traffic flow models and showed how the above models resulted in negative flow and speed. Zhang (1998) developed a new non-equilibrium based theory of traffic flow that addresses the above issue of wrong direction travel. Zhang's model is based on the assumption that drivers adjust their speeds based on traffic conditions ahead of them with a time delay and the existence of a unique speed density relationship.

$$\frac{\partial v}{\partial t} + \frac{v \partial v}{\partial x} = \frac{v^*(k) - v}{T} - k\, v^{*\prime}(k) \frac{\partial k}{\partial x} \quad (100)$$

Aw and Rascle (1999) developed a new momentum equation which also addresses the deficiency identified by Daganzo (1995) based on a pressure law.

$$\frac{\partial [v + p(k)]}{\partial t} + v \frac{\partial [v + p(k)]}{\partial x} = 0 \quad (101)$$

In the above equation, $p(k) = p^{\gamma}$ with $\gamma > 0$. To summarize, despite a significant amount of work being done in using variants of the LWR and second-order models in capturing traffic flow, there exist numerous deficiencies. One of the critical deficiencies is the inability to model the various uncertainties in driving behavior and equilibrium speed flow relationships.

APPROACH

Uncertainty in Traffic Flow Models

Models used for evaluating, planning, designing, and operating transportation systems are plagued with uncertainty from numerous sources. The uncertainty in transportation systems modeling can be categorized into two types – input parameter uncertainty and functional uncertainty. Uncertainty in input parameters can be from two sources. The first is from our inability to conduct extensive data

collection to completely characterize the input parameter. The second source of uncertainty is because certain traffic and transportation parameters are naturally uncertain, as they are affected by a number of factors – human decision making, weather, etc. – all of which cannot be characterized precisely. Functional uncertainty arises from the inability of the models to capture all factors associated with the transportation system. For example, the previous section has clearly highlighted the issues associated with LWR model and higher models and their inability to capture certain critical phenomena associated with traffic flow systems.

Uncertainty in transportation systems has been well researched in the transportation planning and network analysis domain. The three types of uncertainties discussed above have been defined as specification errors, calibration errors, and forecasted input parameter errors in the planning context (Lowe and Richards, 1983). Mahmassani (1984) identified five types of uncertainty in the transportation decision-making context – extreme random shocks and unexpected events due to technology shifts, changes in travel patterns due to socio-political causes, errors in forecasted demands and link flows, fuzziness in performance metrics characterizing transportation systems, and variation in people's choices in decision making. Several models have been developed to characterize the uncertainty and negate its impact in the transportation network analysis and planning community. Asakura and Kashiwadani (1991), Clark and Watling (2005), and Shao et al. (2005) studied the impact of short-term demand uncertainty on traffic network models. Waller et al. (2001), Ukkusuri and Waller (2006), and Duthie et al. (2011) developed methods to account for the impacts of long-term demand uncertainty on transportation network performance. Du and Nischolson (1997), Chen et al. (2002), Lo and Tung (2003), Boyles and Waller (2010), and Unnikrishnan and Waller (2011) developed models to account for the impact of day-to-day capacity uncertainty and degradation on traffic network models. A majority of the above studies focused on accounting for uncertainty and resilient transportation decision making in a planning context. Almost all the models used have been for a static analysis, where a cost function captures the impact of flow on travel costs and system performance. To date, work on characterizing the impact of uncertainty on performance of dynamic flow models has been limited.

Jo and Lou (2001) developed a nonlinear stochastic model that can use Greenshield's, Greenberg's, or Underwood's speed density relationship. The nonlinear models are based on the LWR model. Nonlinear terms are decomposed using Adomian's polynomial approach. Then the deterministic model is transferred to a stochastic case using Ito's equation. Wang et al. (2005, 2007) used a stochastic traffic flow model embedded in an extended Kalman filtering framework to estimate the traffic states on a freeway segment. The stochasticity is modeled by adding an error noise term in the speed and flow equation in the space-time discretized form. The purpose of adding the noise term was to account for modeling inaccuracies. Khoshyaran and Lebacque (2009) developed a stochastic variant of a generic second-order model, which is the LWR model combined with a driver-dependent fundamental diagram. In the stochastic variant the driver attribute is assumed to be random whose dynamics are described by a Brownian process.

Boel and Mihaylova (2006) extended the Cell Transmission Model by modeling the sending and receiving functions as random variables. The uncertainty arises because the authors assume that the location and speed of the vehicles in a cell at a given time are random. Sumalee et al. (2011) developed a stochastic

variant of the modified cell transmission model proposed by Munoz et al. (2003). In the new stochastic cell transmission model, all parameters associated with the sending and receiving functions and the freeway travel demand are assumed to be random variables. Jabari and Liu (2011) model the uncertainty due to driver gap choice in a new stochastic traffic flow model. The stochastic traffic flow model developed is similar to that of the CTM in the sense that it operates on discrete space. However, the time is considered to be continuous. The uncertainty in driver gap choice is modeled by assuming the time headways to be random.

Several authors have noted the uncertainty arising in the speed-flow relationship (Wang et al., 2011). Kerner (1998, 1999a, 1999b), based on empirical data, postulated that the fundamental speed-flow relationship does not hold under congested conditions. Ngoduy (2009) argued that the scattering in flow-density relationship is caused by variations in driving behavior and developed a multi-class, first order model with random capacities. Li et al. (2011) developed a variant of the first-order LWR model where the free-flow speed in Greenshield's model is assumed to be random and varying with density.

This work focuses on the impact of the uncertainty in the equilibrium speed-flow relationship on traffic flow parameters in the first-order LWR model. The two equilibrium relationships considered are Greenshield's and Underwood's speed density models. Free-flow speed is assumed to be uncertain and to follow a random probability distribution. The uncertainty is captured using Monte Carlo simulation and finite difference methods are used to solve the resulting stochastic differential equation. In this report, different ways of Monte Carlo sampling are also studied to understand its impact on the resulting traffic flow parameters.

METHODOLOGY

Uncertainty in LWR Model - Computational Study

In this research, the first-order LWR equation is applied to model the flow of traffic on a freeway segment with constant demand. We consider a freeway segment which is 5 miles long. The freeway segment has a constant demand of 2,400 vehicles per hour. We use two equilibrium speed density relationships – Greenshield's model and Underwood's model. Using Greenshield's model, two uncertain parameters are considered: free-flow speed and jam density. With Underwood's model, two uncertain parameters are considered: free-flow speed and critical density. Due to short-term operational uncertainty, the free-flow speed and jam/critical density are assumed to vary from day-to-day depending on factors like weather. The mean free-flow speed is assumed to be 75 mph and the mean jam density is assumed to be 250 vehicles per mile. The mean critical density is assumed to be 50 vehicles per mile. The disturbance in the free-flow speed and jam/critical density is assumed to follow a normal distribution, with zero mean and known variance. Monte Carlo simulation is used to simulate the realization of free-flow speed and jam density. Each realization of the Monte Carlo simulation is assumed to correspond to one specific day during peak hour in the free-flow segment. The velocity profile is obtained using a forward difference method for 300 seconds, using a time interval of 1 second and a cell length of 0.1 mile. This discretization will satisfy the Courant-Friedrichs-Lewy condition for all potential values of the

free-flow speed. The freeway is assumed to be empty initially. The evolution of velocity is studied at four locations – at 0.5 miles, 1 mile, 2.5 miles, and 5 miles.

FINDINGS

Greenshield's Model – Mean Value

Figures 34 through 37 display the variation in velocity with time at the four locations. The deterministic LWR model is solved assuming the mean free-flow speed and the mean jam density. As expected, since the initial densities are assumed to be zero (there are no vehicles on the freeway at time zero), the decrease is more gradual as we travel further down the freeway.

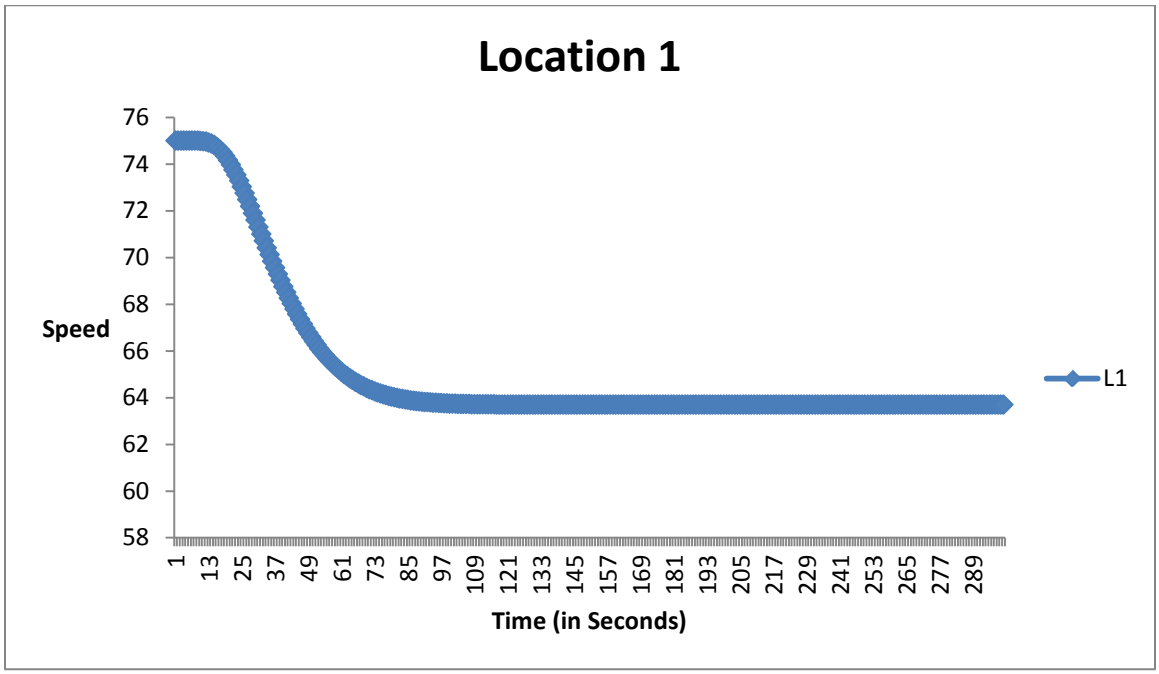

FIGURE 34 LOCATION 1 VARIATION OF VELOCITY PROFILE OVER TIME

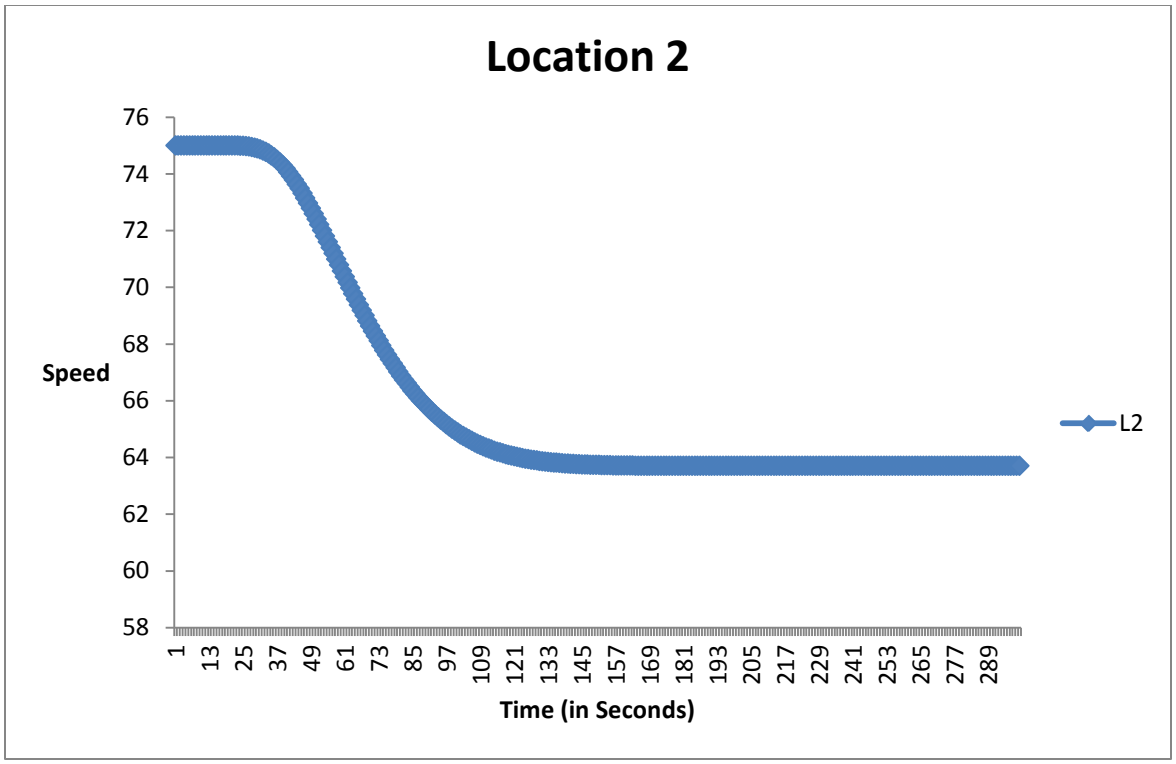

FIGURE 35 LOCATION 2 VARIATION OF VELOCITY PROFILE OVER TIME

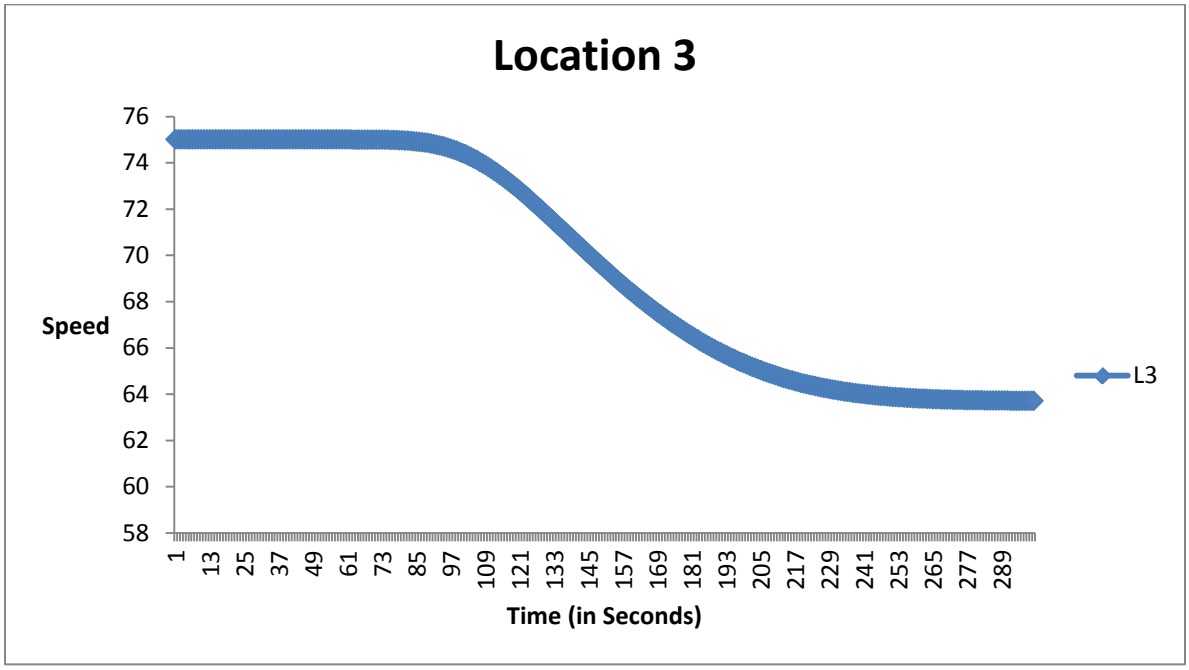

FIGURE 36 LOCATION 3 VARIATION OF VELOCITY PROFILE OVER TIME

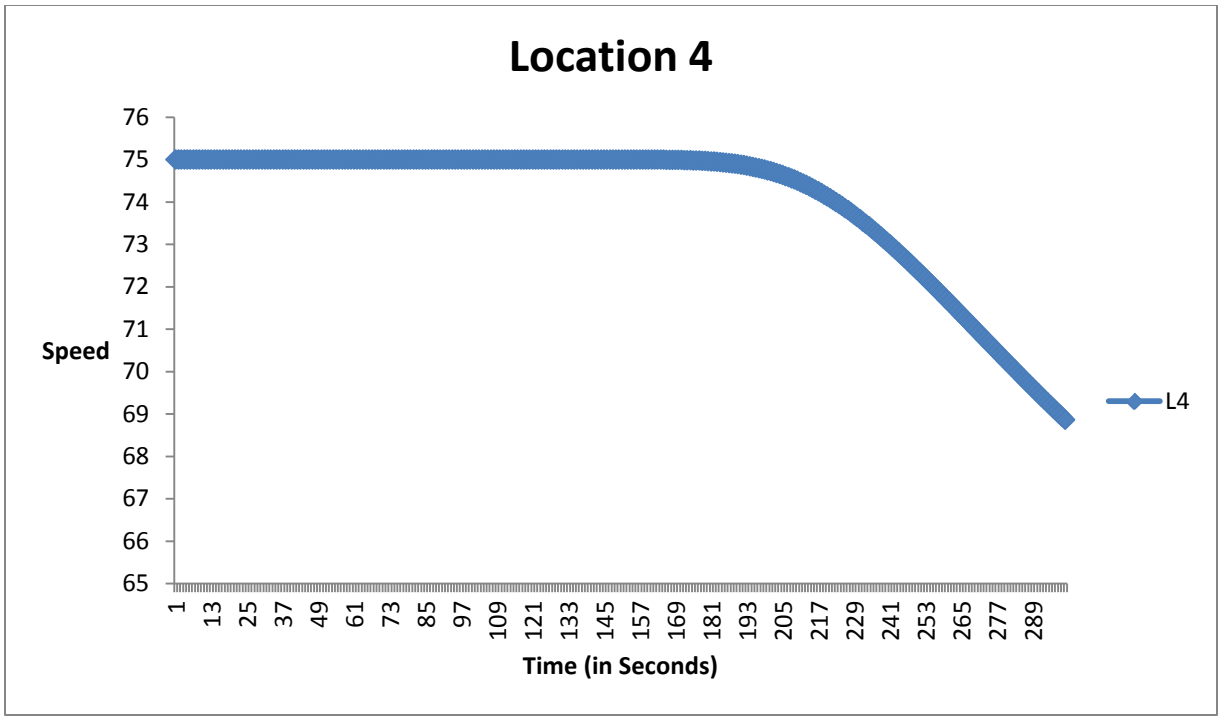

FIGURE 37 LOCATION 4 VARIATION OF VELOCITY PROFILE OVER TIME

Greenshield's Model – Variation of Velocity profile with uncertainty

Figures 38 through 41 illustrate the variation of velocity profiles with uncertainty in free-flow speed and jam density. The disturbance in free-flow speed and jam density is assumed to have a normal distribution, with zero mean and standard deviation of 3. Five free-flow speed and jam density scenarios are simulated. The velocity profiles of the difference scenarios are found to have the same shape with different magnitudes. At locations 1, 2, and 3, the reduction in speed occurs at the same time for all scenarios. At location 4 the speed reduction appears to happen at different time intervals for different scenarios.

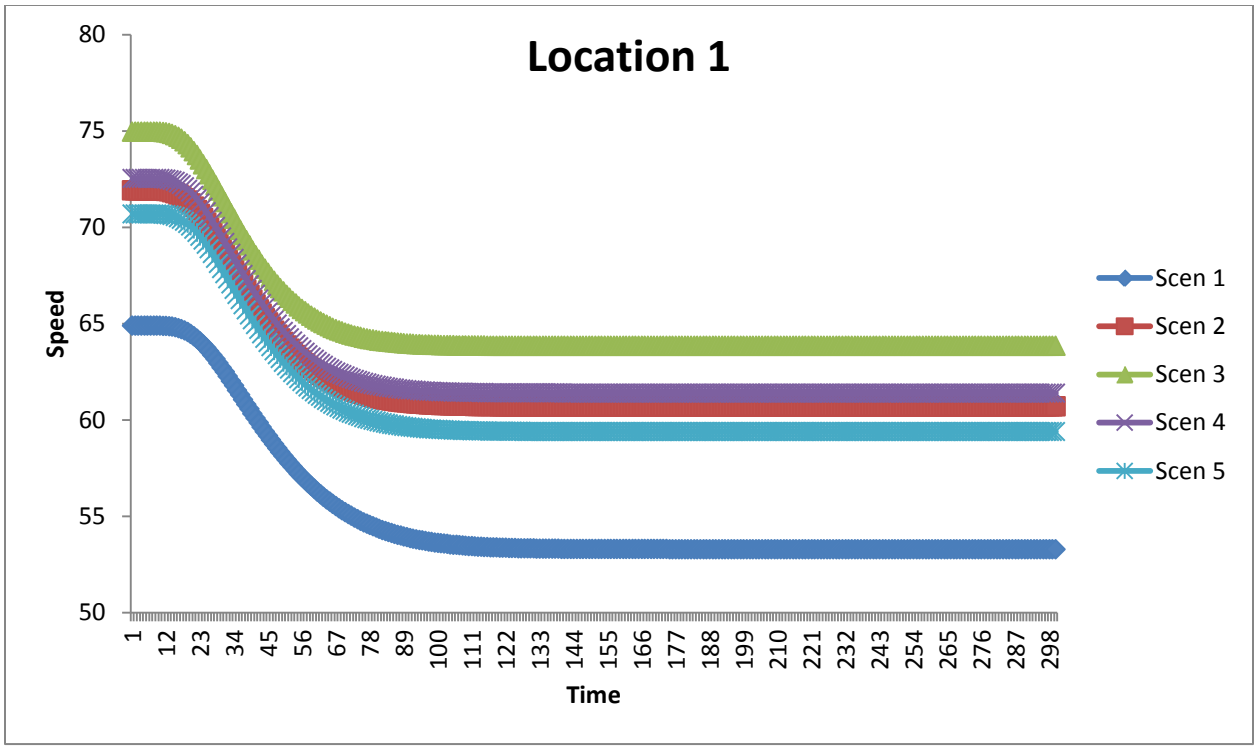

FIGURE 38 LOCATION 1 - VELOCITY PROFILE FOR 5 MONTE CARLO REALIZATIONS

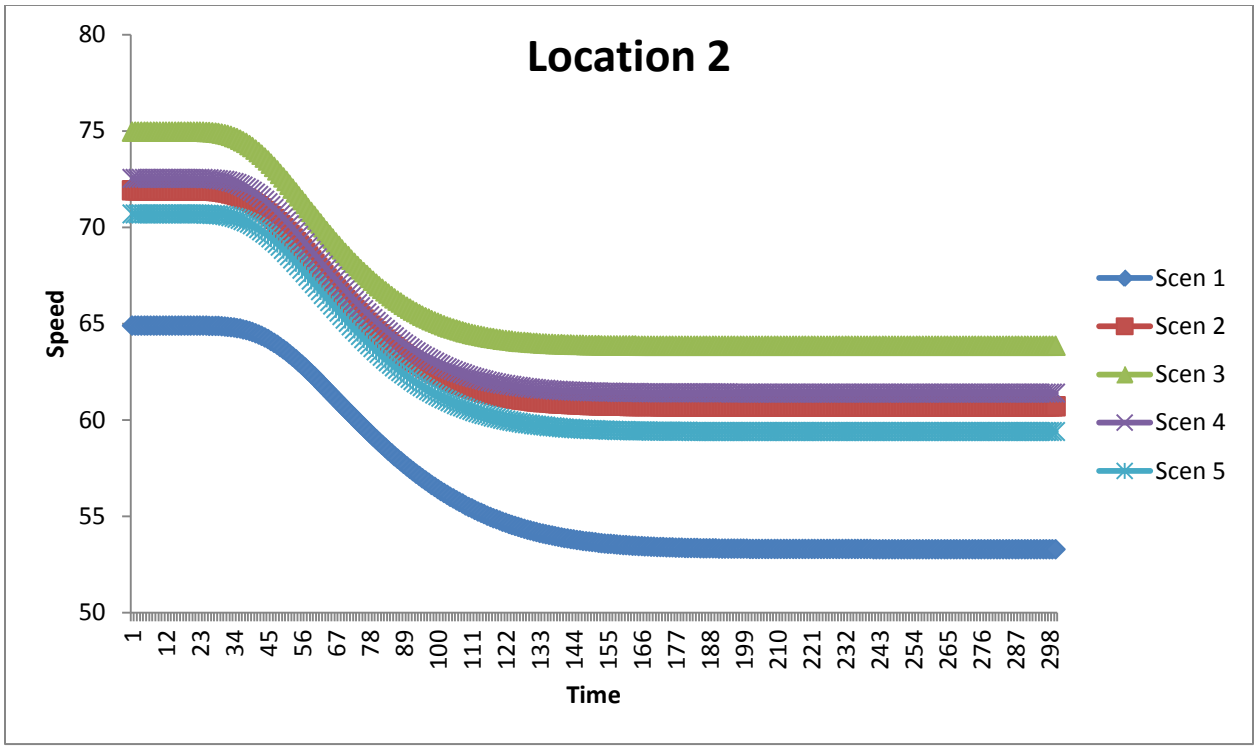

FIGURE 39 LOCATION 2 - VELOCITY PROFILE FOR 5 MONTE CARLO REALIZATIONS

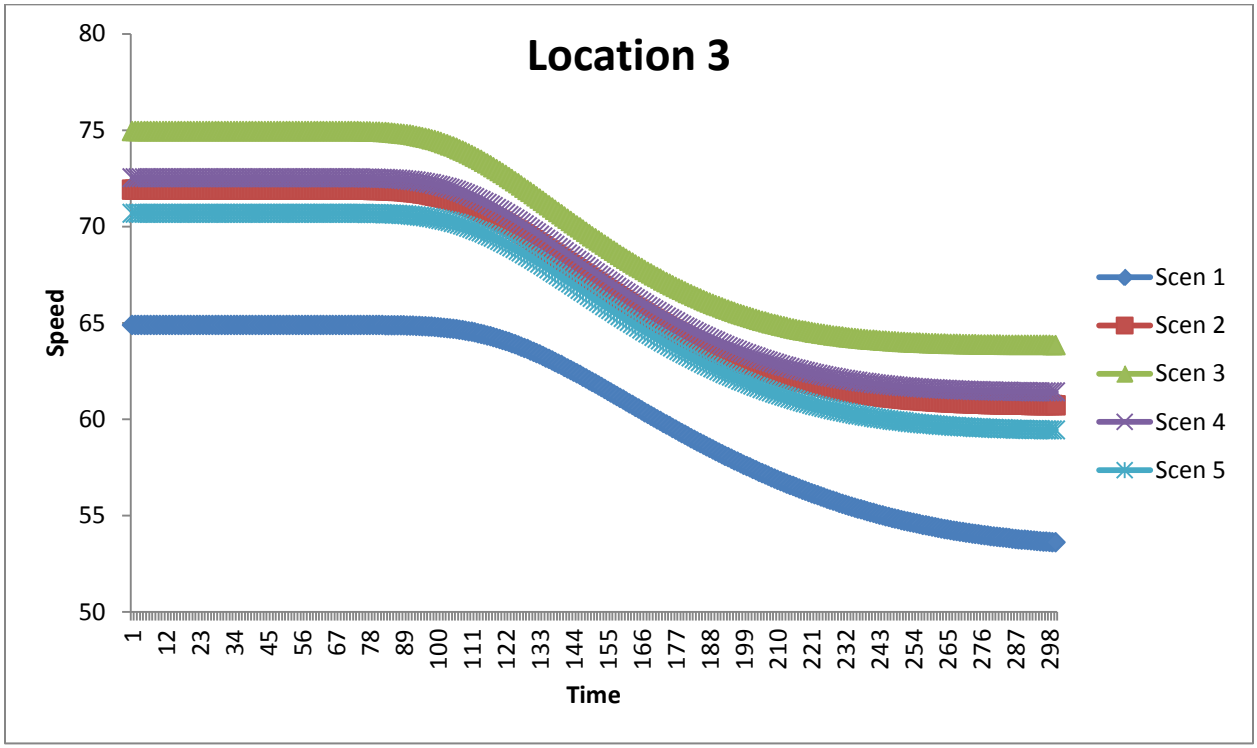

FIGURE 40 LOCATION 3 - VELOCITY PROFILE FOR 5 MONTE CARLO REALIZATIONS

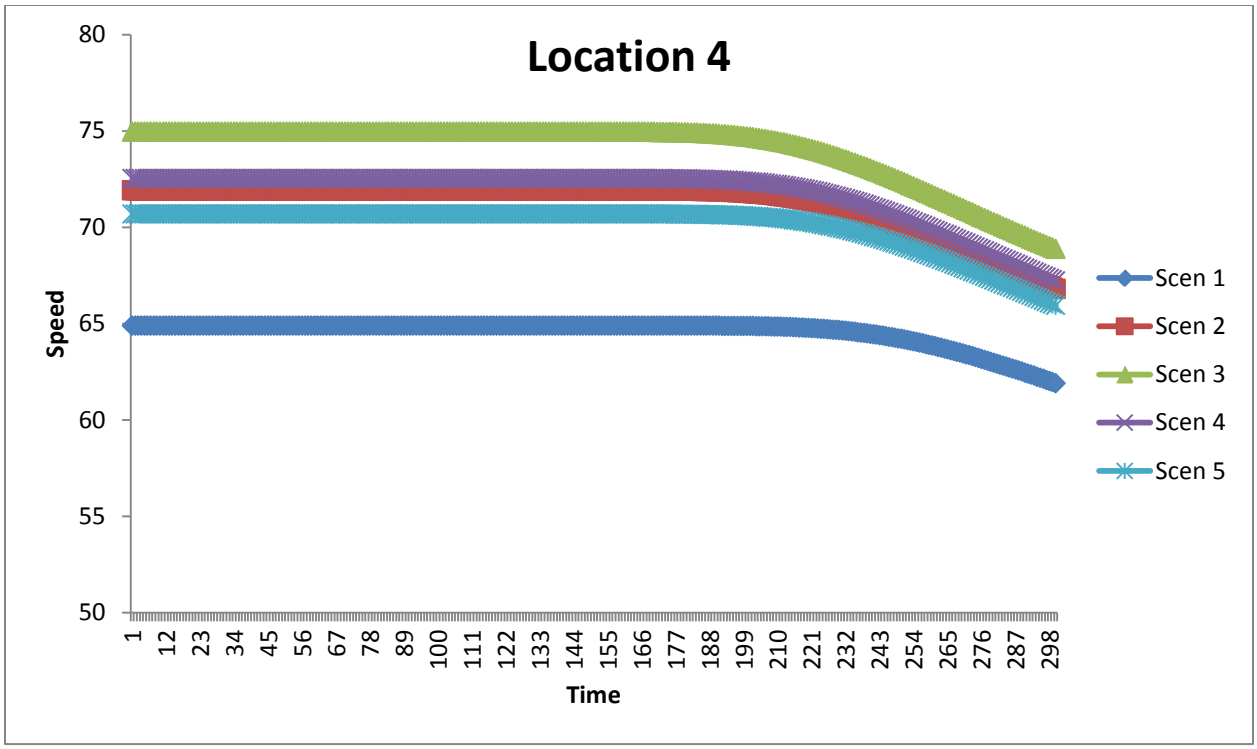

FIGURE 41 LOCATION 4 - VELOCITY PROFILE FOR 5 MONTE CARLO SIMULATIONS

Greenshield's Model – Variation of Error with Monte Carlo Sample Size

Tables 9 and 10 show the variation of errors and means at the four locations with increase in sample size at the first time interval and the last time interval. The standard deviation of the disturbance in free-flow speed and jam density is set equal to 3. As expected, the errors decrease with increase in Monte Carlo sample size. The least errors are found at 100 realizations. The errors are found to be less than 0.5 for 100 realizations for all cases. The errors are marginally higher for the last time interval when compared to the first time interval.

TABLE 9 VARIATION OF ERROR/MEAN WITH SAMPLE SIZE (T =1 SECOND)

	L1	*L2*	*L3*	*L4*
Scen	*Error*			
5	*1.67759*	*1.67759*	*1.67759*	*1.67759*
10	*1.0298*	*1.0298*	*1.0298*	*1.0298*
25	*0.71124*	*0.71124*	*0.71124*	*0.71124*
50	*0.44395*	*0.44395*	*0.44395*	*0.44395*
75	*0.34823*	*0.34823*	*0.34823*	*0.34823*
100	*0.2885*	*0.2885*	*0.2885*	*0.2885*
	Mean			
5	*70.995*	*70.995*	*70.995*	*70.995*
10	*71.896*	*71.896*	*71.896*	*71.896*
25	*74.36*	*74.36*	*74.36*	*74.36*
50	*74.718*	*74.718*	*74.718*	*74.718*
75	*74.924*	*74.924*	*74.924*	*74.924*
100	*74.862*	*74.862*	*74.862*	*74.862*

TABLE 10 VARIATION OF ERROR/MEAN WITH SAMPLE SIZE (T =300 SECOND)

	L1	*L2*	*L3*	*L4*
Scen	*Error*			
5	*1.76354*	*1.76354*	*1.7084*	*1.16177*
10	*1.07973*	*1.07973*	*1.05193*	*0.70968*
25	*0.73608*	*0.73608*	*0.72638*	*0.49234*
50	*0.46204*	*0.46204*	*0.45758*	*0.30902*
75	*0.36303*	*0.36303*	*0.36002*	*0.24215*
100	*0.30096*	*0.30096*	*0.29864*	*0.20022*
	Mean			
5	*59.723*	*59.723*	*59.806*	*66.146*
10	*60.528*	*60.528*	*60.591*	*66.749*
25	*63.046*	*63.046*	*63.077*	*68.442*
50	*63.433*	*63.433*	*63.455*	*68.693*
75	*63.633*	*63.633*	*63.652*	*68.832*
100	*63.552*	*63.552*	*63.571*	*68.784*

Underwood's Model – Mean Value

Figures 42 through 45 display the variation in velocity with time at the four locations. The deterministic LWR model is solved assuming the mean free-flow speed and the mean jam density. The variation in velocity profile under the Underwood model appears to be marginally different from that of Greenshield's model. The Underwood model is defined by two parameters – the free-flow speed and the density at which flow is maximum. The mean critical density at which the flow is maximum is assumed to be 50 vehicles per mile. At location 4 the speed profile is found to be similar to that of Greenshield's model.

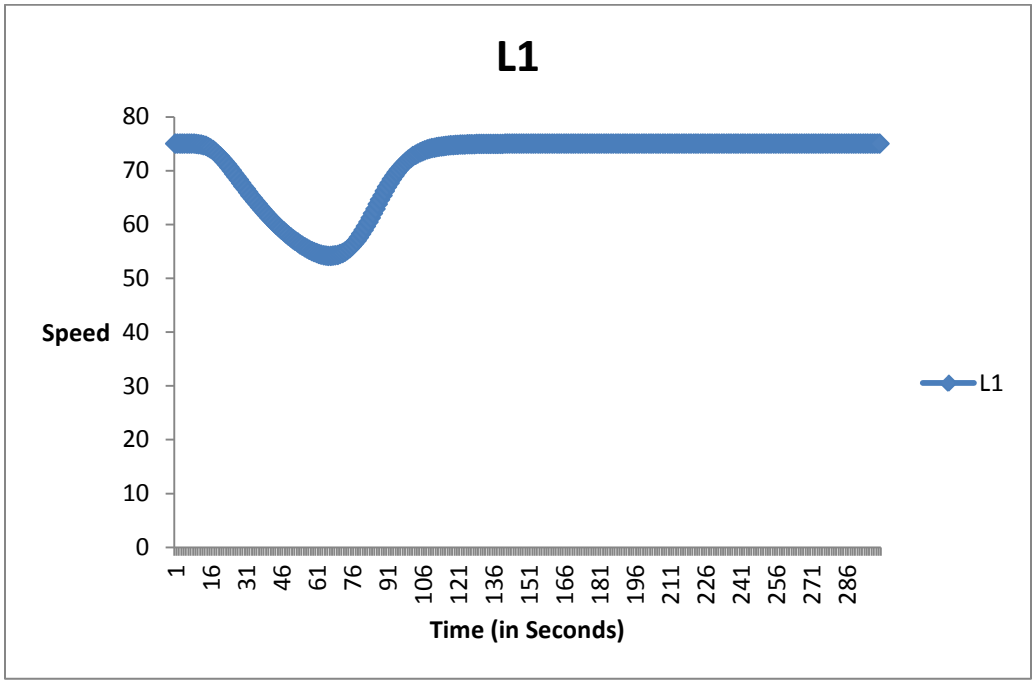

FIGURE 42 LOCATION 1 VARIATION OF VELOCITY PROFILE OVER TIME

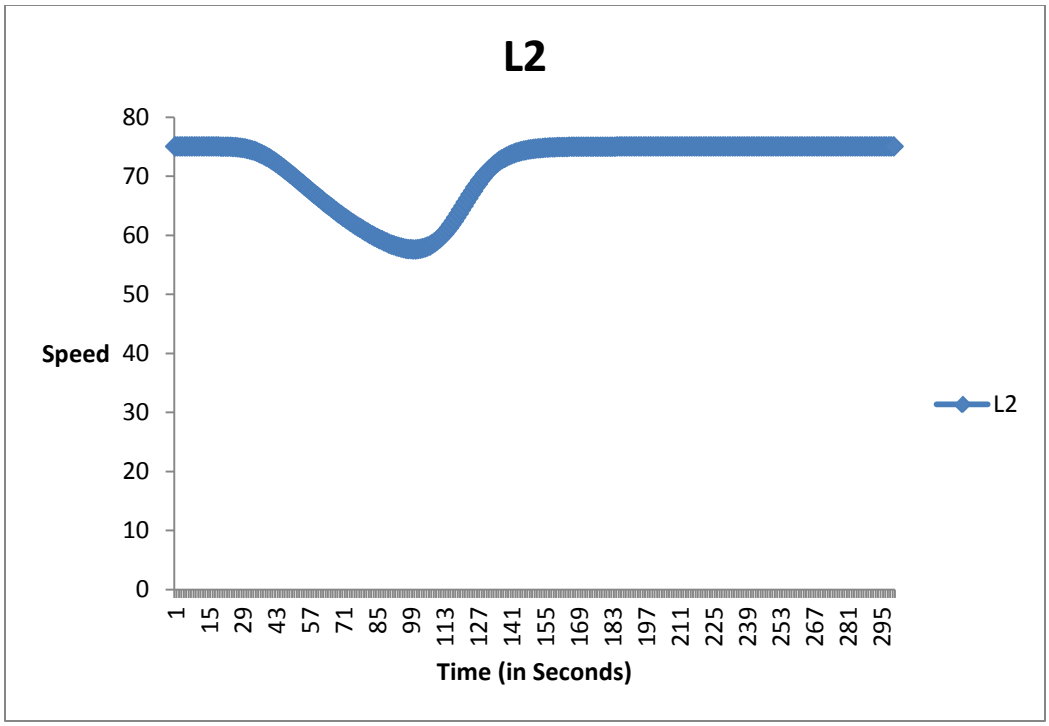

FIGURE 43 LOCATION 2 VARIATION OF VELOCITY PROFILE OVER TIME

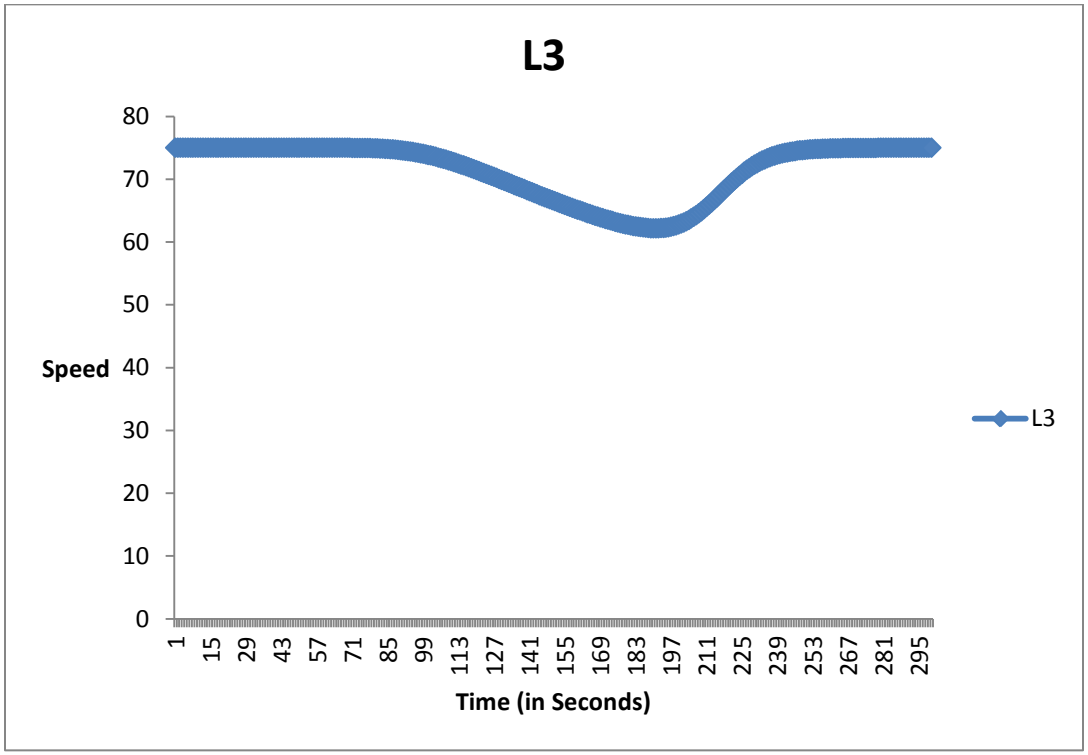

FIGURE 44 LOCATION 3 VARIATION OF VELOCITY PROFILE OVER TIME

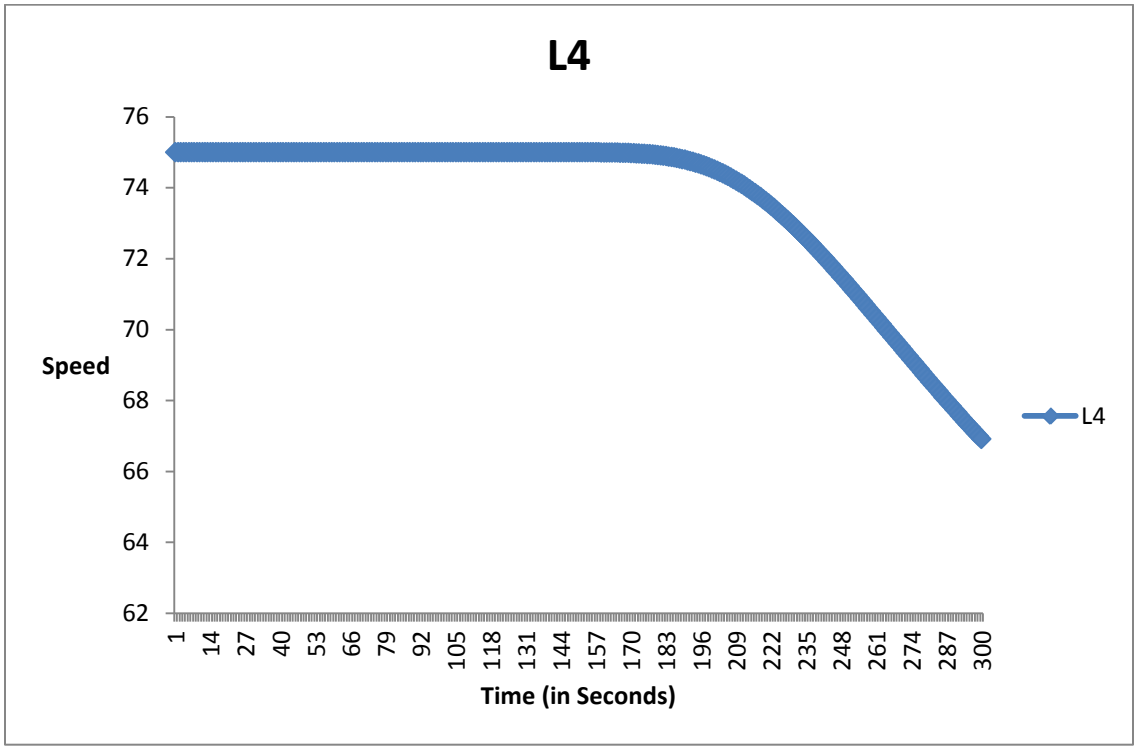

FIGURE 45 LOCATION 4 VARIATION OF VELOCITY PROFILE OVER TIME

Underwood's Model – Variation of Velocity Profile with Uncertainty

Figures 46 through 49 illustrate the variation of velocity profiles with uncertainty in free-flow speed and jam density. The disturbance in free-flow speed and critical density is assumed to have a normal distribution, with zero mean and standard deviation of 3. Five free-flow speed and jam density scenarios are simulated. The speed profiles at all four locations appear to be similar to each other. The speed profile at location 4 and location 1 with Underwood's model appears to be similar to that of the Greenshield model. As expected, the point at which the speed reduces increases with increase in distance from the beginning of the freeway.

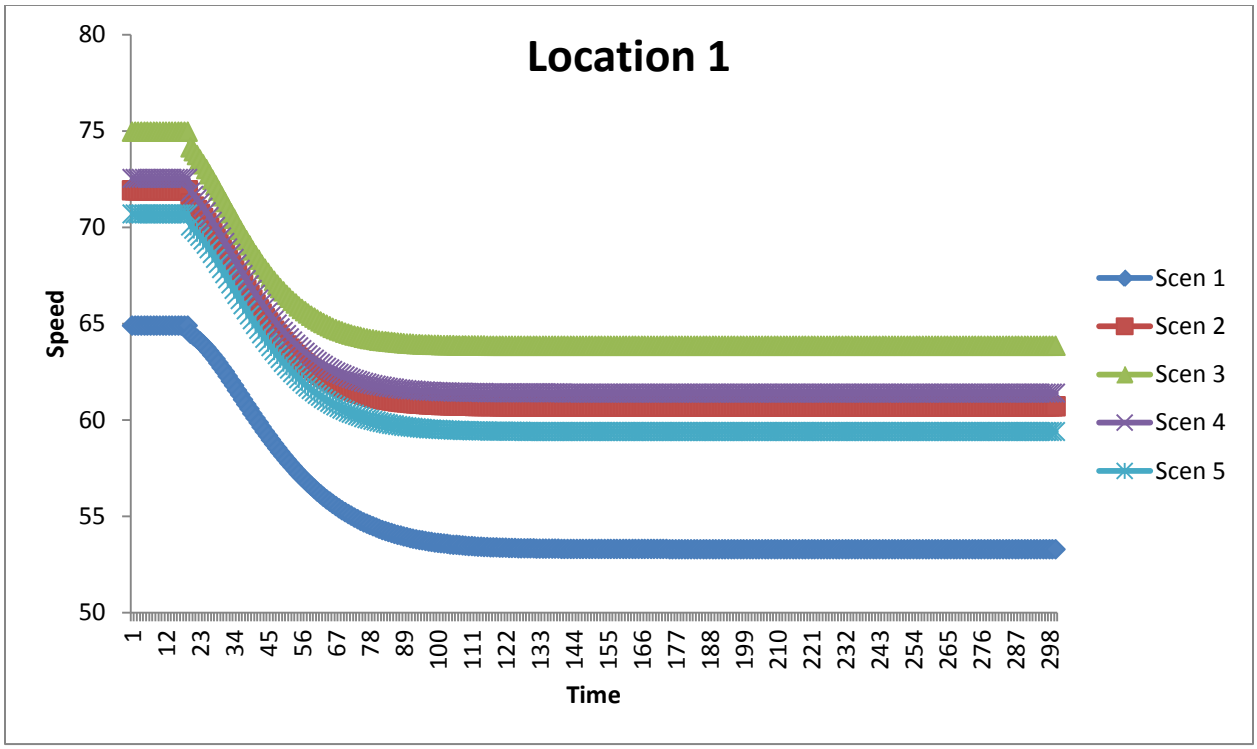

FIGURE 46 LOCATION 1 - VELOCITY PROFILE FOR 5 MONTE CARLO REALIZATIONS

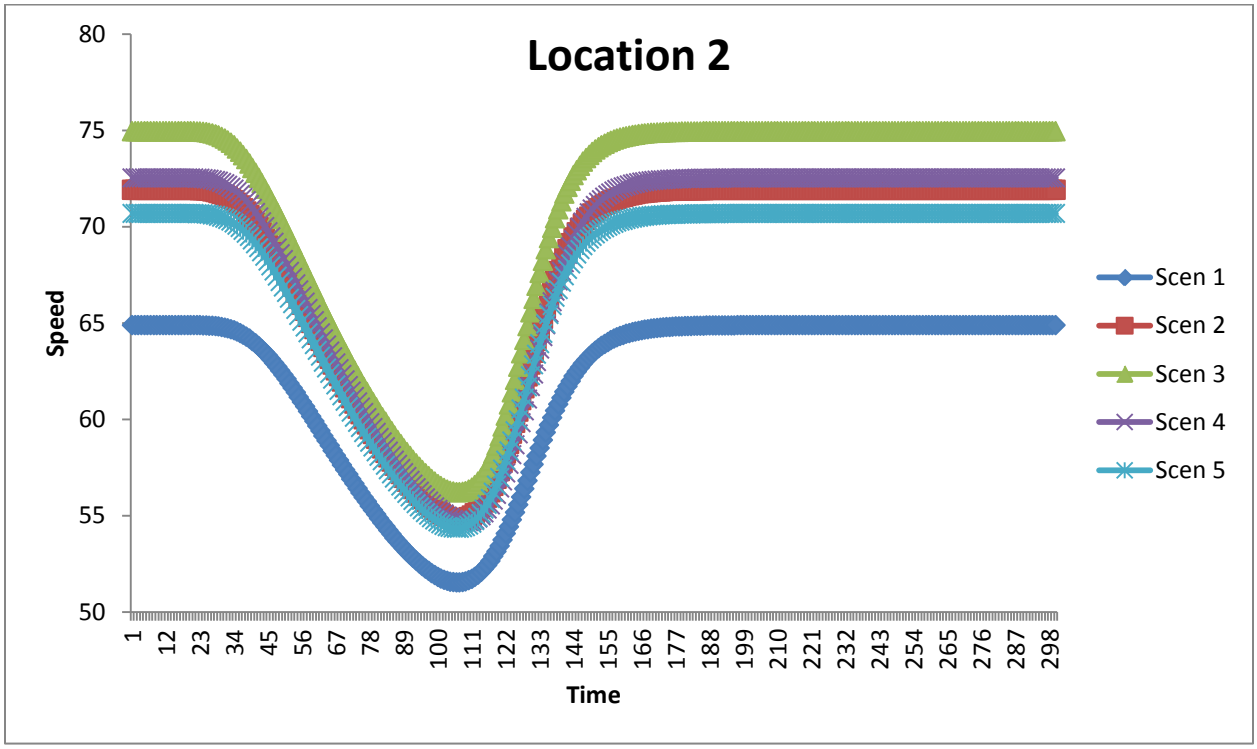

FIGURE 47 LOCATION 2 - VELOCITY PROFILE FOR 5 MONTE CARLO REALIZATIONS

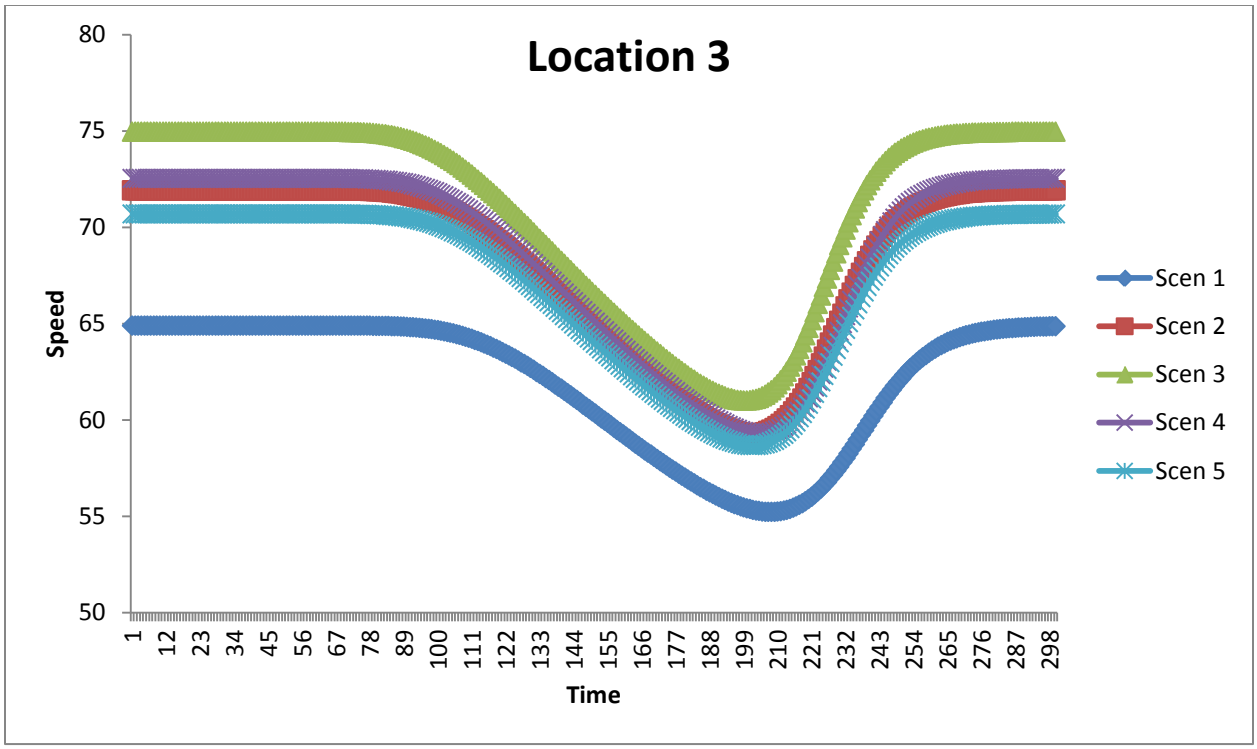

FIGURE 48 LOCATION 3 - VELOCITY PROFILE FOR 5 MONTE CARLO REALIZATIONS

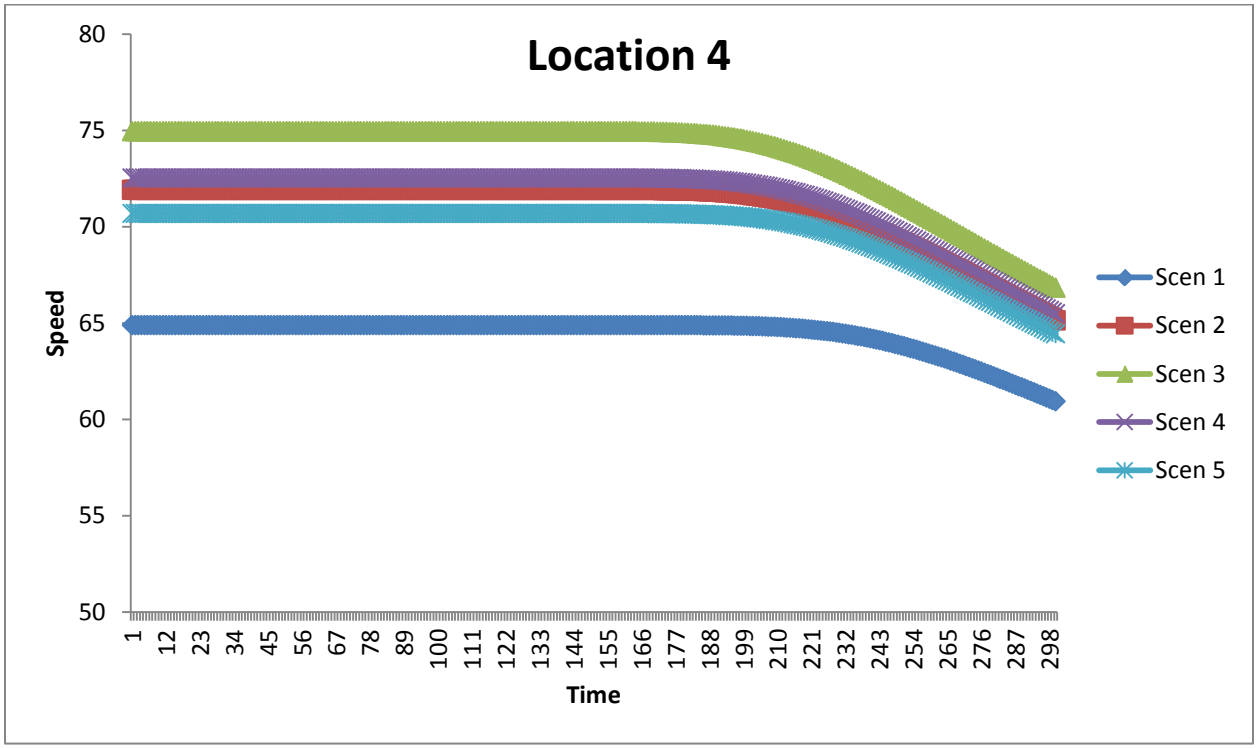

FIGURE 49 LOCATION 4 - VELOCITY PROFILE FOR 5 MONTE CARLO REALIZATIONS

Underwood's Model – Variation of Error with Monte Carlo Sample Size

Tables 11 and 12 show the variation of errors and means at the four locations, with increases in sample size at the first time interval and the last time interval. The standard deviation of the disturbance in free-flow speed and jam density is set equal to 3. As expected the errors decrease with increase in Monte Carlo sample size. The least errors are found at 100 realizations. The errors are found to be less than 0.5 for even 50 realizations for all cases. When compared with Greenshield's model, the errors are found to be relatively stable for t=1 second and t = 300 seconds. At location 4, the errors are found to be higher at t=1 second when compared to t=300 seconds.

TABLE 11 VARIATION OF ERROR/MEAN WITH SAMPLE SIZE (T =1 SECOND)

	L1	L2	L3	L4
Scen	Error			
5	1.6776	1.6776	1.6776	1.6776
10	1.0298	1.0298	1.0298	1.0298
25	0.71124	0.71124	0.71124	0.71124
50	0.44395	0.44395	0.44395	0.44395
75	0.34823	0.34823	0.34823	0.34823
100	0.2885	0.2885	0.2885	0.2885
	Mean			
5	70.995	70.995	70.995	70.995
10	71.896	71.896	71.896	71.896
25	74.36	74.36	74.36	74.36
50	74.718	74.718	74.718	74.718
75	74.924	74.924	74.924	74.924
100	74.862	74.862	74.862	74.862

TABLE 12 VARIATION OF ERROR/MEAN WITH SAMPLE SIZE (T =300 SECOND)

	L1	*L2*	*L3*	*L4*
Scen	*Error*			
5	*1.6776*	*1.6776*	*1.6826*	*0.98648*
10	*1.0298*	*1.0298*	*1.0321*	*0.60376*
25	*0.71124*	*0.71124*	*0.71209*	*0.40362*
50	*0.44395*	*0.44395*	*0.44429*	*0.249*
75	*0.34823*	*0.34823*	*0.34844*	*0.19453*
100	*0.2885*	*0.2885*	*0.28865*	*0.16174*
	Mean			
5	*70.995*	*70.995*	*70.985*	*64.567*
10	*71.896*	*71.896*	*71.89*	*65.14*
25	*74.36*	*74.36*	*74.357*	*66.539*
50	*74.718*	*74.718*	*74.716*	*66.747*
75	*74.924*	*74.924*	*74.922*	*66.868*
100	*74.862*	*74.862*	*74.86*	*66.846*

The preceding numerical analysis demonstrates that with increase in sample sizes, the errors in velocity profiles will decrease. However, for large freeway segments with more complicated second-order models it might not be possible to arrive at relatively low errors in a reasonable computational time due to the time it takes to run each scenario. Therefore, there is a need to develop more efficient sampling strategies that arrive at low errors at lower samples. In this research effort, we study the efficacy of using antithetic variance reduction technique on the errors of the velocity profiles.

Antithetic variance reduction technique

The antithetic variance reduction technique was implemented for the same number of scenarios and the performance compared with the simple Monte Carlo simulation. The antithetic technique was found to outperform other, more complex Monte Carlo simulation techniques for transportation planning applications (Duthie et al., 2011). In the numerical study, performance of antithetic sampling over Monte Carlo sampling was studied by comparing the errors at the four locations at time = 150 seconds and time = 300 seconds for both the Greenshield and Underwood equilibrium speed-flow relationships (see Tables 15-15). A higher standard deviation of 6 and 10 were assumed for the uncertain parameters. The mean was assumed to be the same as the previous study.

TABLE 13 ERRORS AT T = 150 SECONDS, STD DEV = 6, GREENSHIELD'S

	L1	*L2*	*L3*	*L4*
Scen	*Antithetic*			
5	*0.06187*	*0.21699*	*0.03935*	*0.01701*
10	*0.0361*	*0.12835*	*0.0245*	*0.01007*
25	*0.0172*	*0.05151*	*0.01457*	*0.00395*
50	*0.01053*	*0.0295*	*0.00869*	*0.00219*
75	*0.00787*	*0.01982*	*0.00695*	*0.00146*
100	*0.0077*	*0.01674*	*0.00688*	*0.00118*
	Monte Carlo			
5	*3.542*	*3.2282*	*2.6389*	*3.3551*
10	*2.169*	*2.0092*	*1.5969*	*2.0595*
25	*1.4745*	*1.4197*	*1.089*	*1.4215*
50	*0.92421*	*0.89941*	*0.68102*	*0.88709*
75	*0.72618*	*0.70951*	*0.53202*	*0.69587*
100	*0.60207*	*0.58925*	*0.43902*	*0.57657*

TABLE 14 ERRORS AT T = 300 SECONDS, STD DEV = 6, GREENSHIELD'S

	L1	*L2*	*L3*	*L4*
Scen	*Antithetic*			
5	*0.08903*	*0.08803*	*0.31473*	*0.15577*
10	*0.05231*	*0.05171*	*0.18709*	*0.09209*
25	*0.02275*	*0.02256*	*0.07412*	*0.04217*
50	*0.0135*	*0.01341*	*0.0422*	*0.02566*
75	*0.00971*	*0.00966*	*0.02822*	*0.01856*
100	*0.00912*	*0.00908*	*0.02381*	*0.01697*
	Monte Carlo			
5	*3.5718*	*3.5709*	*3.1285*	*2.4896*
10	*2.1823*	*2.1819*	*1.9658*	*1.497*
25	*1.4786*	*1.4785*	*1.4095*	*1.027*
50	*0.92591*	*0.92586*	*0.89643*	*0.64417*
75	*0.72719*	*0.72716*	*0.70848*	*0.50218*
100	*0.60277*	*0.60276*	*0.58909*	*0.41316*

TABLE 15 ERRORS AT T = 150 SECONDS, STD DEV = 10, GREENSHIELD'S

	L1	*L2*	*L3*	*L4*
Scen	*Antithetic*			
5	*0.13412*	*0.84575*	*0.04148*	*0.13501*
10	*0.08435*	*0.50013*	*0.02836*	*0.08044*
25	*0.04236*	*0.20383*	*0.03128*	*0.03118*
50	*0.02293*	*0.11947*	*0.01847*	*0.01722*
75	*0.01798*	*0.08078*	*0.01587*	*0.01143*
100	*0.0171*	*0.0693*	*0.01576*	*0.00917*
	Monte Carlo			
5	*5.6284*	*4.8425*	*4.7677*	*5.5919*
10	*3.5053*	*3.0739*	*2.845*	*3.4326*
25	*2.4243*	*2.2798*	*1.9026*	*2.3639*
50	*1.5257*	*1.4614*	*1.1854*	*1.4744*
75	*1.2022*	*1.1585*	*0.92113*	*1.157*
100	*0.99832*	*0.96453*	*0.75653*	*0.959*

TABLE 16 ERRORS AT T = 300 SECONDS, STD DEV = 10, GREENSHIELD'S

	L1	*L2*	*L3*	*L4*
Scen	*Antithetic*			
5	*0.37005*	*0.08092*	*1.1126*	*0.21216*
10	*0.21864*	*0.04633*	*0.65825*	*0.12601*
25	*0.09056*	*0.04175*	*0.27126*	*0.07522*
50	*0.05254*	*0.02938*	*0.16121*	*0.04703*
75	*0.03676*	*0.02281*	*0.10948*	*0.03674*
100	*0.03366*	*0.0235*	*0.0944*	*0.03416*
	Monte Carlo			
5	*6.1807*	*5.8631*	*4.5659*	*4.6793*
10	*3.7481*	*3.6169*	*2.928*	*2.7605*
25	*2.4981*	*2.4587*	*2.2392*	*1.8446*
50	*1.5559*	*1.5398*	*1.4461*	*1.1516*
75	*1.2199*	*1.2107*	*1.1493*	*0.89164*
100	*1.0103*	*1.0041*	*0.95843*	*0.72834*

The errors reveal that antithetic sampling significantly outperforms simple Monte Carlo sampling for almost all scenarios. In fact, the antithetic sampling produces fewer errors (10% of that of simple Monte

Carlo sampling at 100) at five scenarios when compared to Monte Carlo sampling at 100. Therefore, it is computationally more efficient to use antithetic sampling when compared to simple Monte Carlo sampling, especially if the length of the freeway and the time frame of analysis increase. Figure 50 demonstrates visually the difference in errors between antithetic sampling at 5 and simple Monte Carlo sampling at 100 samples. Therefore, antithetic sampling delivers significant computational savings.

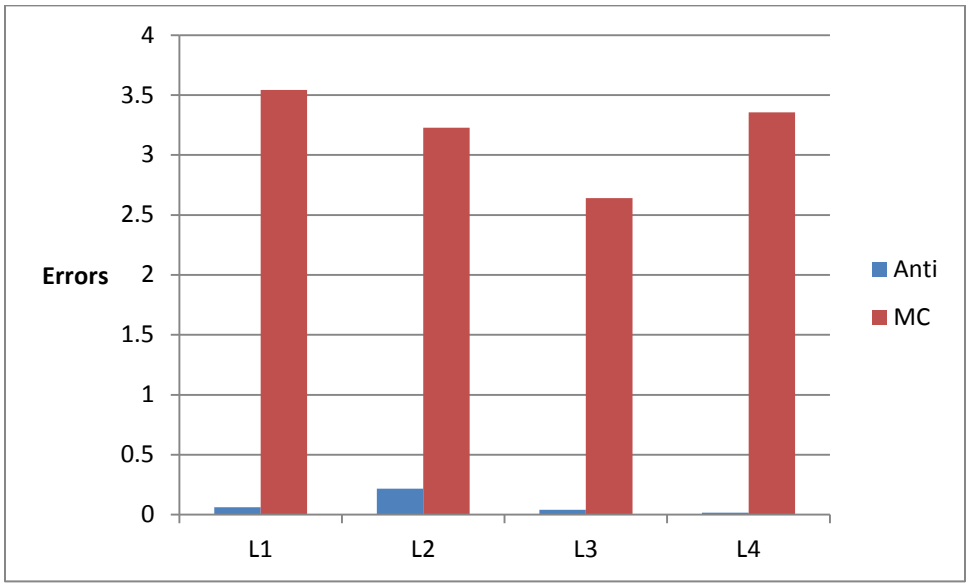

FIGURE 50 REPRESENTATIVE COMPARISON OF ERRORS OF ANTITHETIC SAMPLING AT 5 SCENARIOS VS MONTE CARLO SAMPLING AT 100

When Underwood's model was used for the equilibrium speed-flow relationship, similar results were obtained, thus demonstrating the efficacy of antithetic sampling over simple Monte Carlo sampling and the robustness of the variance reduction techniques. Note that in Tables 17 through 20, many entries are 0, which denotes that the errors are negligible.

TABLE 17 ERRORS AT T = 150 SECONDS, STD DEV = 6, UNDERWOOD'S MODEL

	L1	*L2*	*L3*	*L4*
Scen	*Antithetic*			
5	*0.00233*	*0.10861*	*0.45739*	*0.03618*
10	*0.00403*	*0.2166*	*0.27167*	*0.02143*
25	*0.01262*	*0.58771*	*0.11313*	*0.0084*
50	*0.007*	*0.32879*	*0.06636*	*0.0047*
75	*0.00734*	*0.34081*	*0.04527*	*0.00313*
100	*0.31112*	*0.37797*	*0.03917*	*0.00253*
	Monte Carlo			
5	*3.355*	*3.4095*	*2.2756*	*3.3549*
10	*2.0549*	*2.226*	*1.3636*	*2.0595*
25	*1.4238*	*1.6157*	*0.86274*	*1.4202*
50	*1.001*	*1.0047*	*0.52877*	*0.8861*
75	*0.76464*	*0.76585*	*0.41048*	*0.69516*
100	*0.62283*	*0.63644*	*0.34014*	*0.57604*

TABLE 18 ERRORS AT T = 300 SECONDS, STD DEV = 6, UNDERWOOD'S MODEL

	L1	*L2*	*L3*	*L4*
Scen	*Antithetic*			
5	*Negligible*	*Negligible*	*0.11039*	*0.21008*
10	*Negligible*	*Negligible*	*0.06499*	*0.12472*
25	*Negligible*	*Negligible*	*0.02555*	*0.11656*
50	*Negligible*	*Negligible*	*0.01365*	*0.06926*
75	*Negligible*	*Negligible*	*0.00903*	*0.04828*
100	*Negligible*	*Negligible*	*0.20231*	*0.05393*
	Monte Carlo			
5	*3.3551*	*3.3551*	*3.4759*	*2.209*
10	*2.0596*	*2.0596*	*2.1106*	*1.3268*
25	*1.4225*	*1.4225*	*1.4388*	*0.85254*
50	*0.88789*	*0.88789*	*0.92388*	*0.52749*
75	*0.69646*	*0.69646*	*0.71942*	*0.41008*
100	*0.577*	*0.57699*	*0.59229*	*0.34513*

TABLE 19 ERRORS AT T = 150 SECONDS, STD DEV = 10, UNDERWOOD'S MODEL

	L1	L2	L3	L4
Scen	Error			
5	0.04163	0.89674	1.3832	0.25169
10	0.03246	0.81229	0.84239	0.14966
25	1.375	1.72	0.34606	0.05821
50	0.73914	0.9868	0.20175	0.03239
75	0.81129	0.86972	0.13714	0.02152
100	0.84513	0.80948	0.12308	0.0173
	Mean			
5	5.6217	4.5454	4.3391	5.5918
10	3.3307	3.6643	2.5544	3.4325
25	2.3887	2.7427	1.5251	2.3569
50	1.5035	1.5943	0.93118	1.469
75	1.1768	1.2777	0.71501	1.1532
100	0.98715	1.0924	0.59656	0.9563

TABLE 20 ERRORS AT T = 300 SECONDS, STD DEV = 10, UNDERWOOD'S MODEL

	L1	L2	L3	L4
Scen	Error			
5	Negligible	Negligible	0.58901	1.981
10	Negligible	Negligible	0.33842	1.086
25	Negligible	Negligible	0.58216	0.66116
50	Negligible	Negligible	0.3121	0.37102
75	0.00026	0.00229	0.51523	0.2518
100	0.40334	0.38823	0.59568	0.22678
	Mean			
	5.5919	5.5921	6.2338	4.3689
	3.4327	3.4328	3.623	2.5631
25	2.3709	2.3709	2.4699	1.6123
50	1.5726	1.5609	1.3969	0.9908
75	1.2183	1.2115	1.1251	0.78331
100	1.0002	0.99555	0.93991	0.65955

CONCLUSIONS AND RECOMMENDATIONS

Traffic flows on freeways are affected by a number of behavioral and environmental parameters that are uncertain and difficult to quantify. However, existing traffic flow models are not adequate to account for and capture the impact of the uncertainties. For example, significant day-to-day variations in free-flow speed, jam density, and critical density can exist based on weather conditions. Several past empirical studies have also noted the uncertainties in the equilibrium speed-flow relationship. In this project, the impact of uncertainties of free-flow speed, jam density, and critical density was studied for the basic LWR model using Greenshield's and Underwood's equilibrium relationships. The numerical results demonstrate that the velocity profiles can be significantly different depending on the actual values of the parameters. Therefore, there is a need to develop methodologies that account for this uncertainty. Monte Carlo based simulations were used to solve the stochastic differential equations. The errors were found to reduce with increase in sample size. Antithetic variance reduction technique was implemented to obtain lower errors at lower samples. The results demonstrate that using intelligent variance reduction techniques like antithetic sampling will result in significant computational savings.

In terms of directions of future research, the results show the importance of accounting for uncertainty in traffic flow models. Therefore, there is a need to develop more accurate methods for solving traffic flow theory models under uncertainty. One potential venue could be to study the advances made in modeling and solving stochastic differential equations and apply that to traffic flow theory models. There is also a need to collect data to calibrate the uncertainties in parameters used in the traffic flow models.

REFERENCES

Asakura, Y., and Kashiwadani, M., (1991), Road network reliability caused by daily fluctuations of traffic flow. *Proceedings of the 19th PTRC Summer Annual Meeting*, pp. 73-84.

Aw, A., and Rascle, M., (1999), Resurrection of "second order" models of traffic flow, *SIAM Journal of Applied Mathematics*, Volume 60, 281-345.

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

Boyles, S.D., and Waller, S.T., (2011), Optimal Information Location for Adaptive Routing, *Networks and Spatial Economics*, Volume 11, Issue 2, 233- 254.

Clark, S., and Watling, D., (2005), Modelling network travel time reliability under stochastic demand, *Transportation Research Part B*, Volume 39, 119-140.

Chen, A., Yang, H., Lo, H. K., and Tang, W.H., (2002), Capacity reliability of a road network: an assessment methodology and numerical results, *Transportation Research Part B*, Volume 36, 225-252.

Daganzo, C.F., (1995), Requiem for second-order fluid approximations of traffic flow, *Transportation Research Part B*, Volume 29, 277-286.

Du, Z., and Nicholson, A., (1997), Degradable transportation systems: Sensitivity and reliability analysis, *Transportation Research Part B*, Volume 31, 225-237.

Duthie, J.C., Unnikrishnan, A., and Waller, S.T., (2011), Influence of Demand Uncertainty and Correlations on Traffic Predictions and Decisions, *Computer Aided Civil and Infrastructure Engineering*, Volume 26(1), 16-29.

Jabbari, S. E., and Liu, H. X., (2012), A stochastic model of traffic flow: Theoretical foundations, *Transportation Research Part B*, Volume 46, 156 - 174.

Jo, Y.-J., and Cho, S.-C., (2001), Modeling of Nonlinear Stochastic Dynamic Traffic Flow, *Journal of Transportation Research Record*, Volume 1771, 83-88.

Kerner, B. S., (1998), Experimental Features of Self-Organization in Traffic Flow, *Physical Review Letters*, 81, 3797-3400.

Kerner, B. S., (1999a), The physics of traffic, *Physics World Magazine*, 12, 25-30.

Kerner, B. S., (1999b), Congested Traffic Flow: Observations and Theory, *Transportation Research Record*, Volume 1678, 160-167.

Khoshyaran, M. M., Lebacque, J-P., (2009), A Stochastic Macroscopic Traffic Model Devoid of Diffusion, *Traffic and Granular Flow*, 139-150.

Kuhne, R.D., (1984), Macroscopic freeway model for dense traffic - stop-start waves and incident detection. *Ninth International Symposium on Transportation and Traffic Theory*, 21-42.

Li, J., Chen, Q.-Y., Wang, H., and Ni, D., (2011), Analysis of LWR model with fundamental diagram subject to uncertainties, Transportmetrica, Volume 8 (12), 1-19.

Lighthill, M. H., and Whitham, G. B., (1955), On kinematic waves: II. A theory of traffic flow on long crowded roads. *Proceedings of The Royal Society of London Ser*. A 229, 317-345.

Lo , H.K, and Tung, Y. K., (2003), Network with degradable links: capacity analysis and design, *Transportation Research Part B*, Volume 37,345-363.

Lowe, S., and Richards, M.G., (1983), Systematic sensitivity testing in project appraisal- a method of identifying the uncertainty of transportation forecasts, In *Proceedings of the 10th Transportation Planning Colloquium*, Delft, Netherlands, 3-12.

Mahmassani, H.S., (1984), Uncertainty in transportation systems evaluation: Issues and approaches, *Transportation Planning and Technology*, Volume 9, 1-12.

Michalopoulos, R., Yi, P., and Lyrintzis, A.S., (1993), Continuum modeling of traffic dynamics for congested freeways, *Transportation Research Part B*, Volume 23, 386-389.

May, A.D., (1990), *Traffic Flow Fundamentals*, Prentice Hall, Englewood Cliffs, New Jersey.

Muñoz, L., Sun, X., Horowitz, R., and Alvarez, L., (2003), Traffic density estimation with the cell transmission model. In *Proceedings of the American Control Conference*, Denver, Colorado, USA, 3750–3755.

Ngoduy, D., (2009), Multiclass First-Order Modelling of Traffic Networks using Stochastic Fundamental Diagrams, *Transportmetrica*, 1-15.

Payne, H. J., (1971), Models of freeway traffic and control. In G. A. Bekey (Ed.), *Mathematical Models of Public Systems*, Volume 1 (1), 5l-61.

Papageorgiou, M., (1983), A hierarchical control system for freeway traffic, *Transportation Research Part B*, Volume 17, 251-261.

Richards, P.I., (1956), Shock waves on the highway, Operations Research, 4, 42–51.

Ross, P., (1988), Traffic dynamics, *Transportation Research Part B*, Volume 22, 421-435.

Shao, H., Lam, W. H. K., and Lam, M. L., (2005), A reliability based stochastic traffic assignment model for network with multiple user classes under uncertainty in demand, *Networks and Spatial Economics*, Volume 6, 169-172.

Sumalee, A., Zhong, R. X., Pan, T. L., and Szeto, W. Y., (2011), Stochastic cell transmission model (SCTM): A stochastic dynamic traffic model for traffic state surveillance and assignment, *Transportation Research Part B*, Volume 45, 507 - 533.

Ukkusuri, S., and Waller, S.T., (2006), Single-point approximations for traffic equilibrium problem under uncertain demand, *Proceedings of the 85th Annual Meeting of the Transportation Research Board*, CD-ROM, Washington, D.C.

Waller, S. T., Schofer, J.L., and Ziliaskopoulos, A.K., (2001), Evaluation with traffic assignment under demand uncertainty. *Journal of the Transportation Research Board*, Volume 1771, 69-74.

Wang, Y., and Papageorgious, M., (2005), Real time freeway traffic state estimation based on extended Kalman filter: a general approach, *Transportation Research Part B*, Volume 39, 141 - 167.

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

Wang, H., Li, J., Chen, Q-Y., and Ni, D., (2011). Stochastic Modeling of the Equilibrium Speed-Density Relationship. *Journal of Advanced Transportation*, doi: 10.1002/atr.172.

Zhang, H. M., (1998), A theory of nonequilibrium traffic flow. *Transportation Research Part B*, Volume 32, 485-498.