Real-Time Ground Vehicle Parameter Estimation and System Identification for Improved Stability Controllers

Jeremy Joseph Kolansky

Dissertation submitted to the faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

In

Mechanical Engineering

Corina Sandu, Chair

Pablo A. Tarazaga

Steve C. Southward

Mehdi Ahmadian

P. Schalk Els

Hosam K. Fathy

March 18th, 2014

Blacksburg, Virginia

Keywords: Parameter Estimation, Polynomial Chaos, Bayesian Statistics, System Identification, Kalman Filter, Load Transfer, Rollover Stability

Real-Time Ground Vehicle Parameter Estimation and System Identification for Improved Stability Controllers

Jeremy Joseph Kolansky

ABSTRACT

Vehicle characteristics have a significant impact on handling, stability, and rollover propensity. This research is dedicated to furthering the research in and modeling of vehicle dynamics and parameter estimation.

Parameter estimation is a challenging problem. Many different elements play into the stability of a parameter estimation algorithm. The primary trade-off is robustness for accuracy. Lyapunov estimation techniques, for instance, guarantee stability but do not guarantee parameter accuracy. The ability to observe the states of the system, whether by sensors or observers is a key problem. This research significantly improves the Generalized Polynomial Chaos Extended Kalman Filter (gPC-EKF) for state-space systems. Here it is also expanded to parameter regression, where it shows excellent capabilities for estimating parameters in linear regression problems.

The modeling of ground vehicles has many challenges. Compounding the problems in the parameter estimation methods, the modeling of ground vehicles is very complex and contains many difficulties. Full multibody dynamics models may be able to accurately represent most of the dynamics of the suspension and vehicle body, but the computational time and required knowledge is too significant for real-time and realistic implementation. The literature is filled with different models to represent the dynamics of the ground vehicle, but these models were primarily designed for controller use or to simplify the understanding of the vehicle's dynamics, and are not suitable for parameter estimation.

A model is devised that can be utilized for the parameter estimation. The parameters in the model are updated through the aforementioned gPC-EKF method as applies to polynomial systems. The mass and the horizontal center of gravity (CG) position of the vehicle are estimated to high accuracy.

The culmination of this work is the estimation of the normal forces at the tire contact patch. These forces are estimated through a mapping of the suspension kinematics in conjunction with the previously estimated vehicle parameters. A proof of concept study is shown, where the system is mapped and the forces are recreated and verified for several different scenarios and for changing vehicle mass.

Acknowledgements

I would like to thank Jesus Christ, who is everything. And so much more.

I would like to thank Dr. Corina Sandu, who had to put up with a lot, and has taught me an immense amount. I would like to thank Dr. Pablo Tarazaga, who saved me from going bankrupt at cost to himself and gave me extremely interesting work. I would like to thank Dr. Schalk Els for letting me come visit them in South Africa to see the vehicle and spend an awesome month with their team and for all of his invaluable conversations. I would like to thank Dr. Steve Southward for teaching me that covariance matrices are really just “blackmagic” and for his help in getting the gPC-EKF to actually work. I would like to thank Dr. Mehdi Ahmadian for all the good conversations and his teaching of vehicle dynamics that helped me when I had no idea what the Land Rover had just done. I would like to thank Dr. Hosam Fathy, for his diplomatic advice that has forever changed how I interact with people and for his expertise in vehicle dynamics.

To all my friends at school, for your companionship, conversation and help (in no particular order): Sriram Malladi, Bryan Joyce, Carlos Garcia, Eduardo Pinto, Joe Hays, Elizabeth Armstrong, Anudeep Bhoopalam, Andrew Rogers, Bryce Lee, Anake Umsrithong, Yitao Zhu, Ravi Anant, Dragan Avirovic, Ethan Robinson, Ethan Hill, Alex Clark, Scott Naranjo, Chris Johnson, and Lindsey Teaters.

To all my friends at Gravitate, for your support and willingness to hold a conversation with me even when I couldn’t put words together or just randomly lost my marbles: Amy Mock, Emily Wright, Calvin and Kristen Martin, Dan and Leana Fischer, Marcus Aguilar, Chris Barb, Levi Napier, Jillian Derrick, Peter Mattson, Chris Hale, Heather McMillan, Lindsey Shafer, Lochlanina Tobey, and Jason and Whitney Ata.

To the crew, there are no words for all that you guys are to me: Tom Rigby, Brandon Ryan, Kyle Metzer, Brian Scott, Matt Mahoney, Rachel Rothspan, and Rusty Webb.

To my parents, Debra and Michael, and my siblings, Dan and Amanda, without whose love and support I would not be the man I am today.

TABLE OF CONTENTS

List of Figures

List of Tables

1 Introduction

Vehicle dynamics and parameter estimation are both rich and dense fields of research. The work presented here is at a crossroads between the two fields, and this section describes the motivation for investigating this intersection.

An overview of the work conducted provides a guide through the research presented in this dissertation, in order to tie together each of the sections, as well as to provide the big picture that helps understanding the connection between different components of the study.

1.1 Motivation

The National Highway and Traffic Safety Administration maintain several databases that record vehicle accidents and fatalities. In 2000, the Fatality Analysis Reporting System (FARS) database shows that 53% of light vehicle occupant fatalities in single-vehicle crashes involved a rollover event. The percentage of crashes that involve a rollover event increases with the size of the vehicle, 46% of passenger cars, approximately 60% for pickup trucks and vans, with 78% for sport utility vehicles [1].

Rollover events are second only to head-on collisions for passenger fatalities [2]. Combined with the increase in rollover propensity, demonstrated through the crash statistics, for increasing vehicle size it is apparent that vehicle characteristics play a key role in the vehicle's stability.

This research was started with the goal of improving vehicle stability to prevent the loss of life and injuries that are so common with vehicle rollover events. During the course of the research it became clear that improvement of vehicle stability controllers would require improved knowledge of the vehicle's parameters. The National Highway Traffic and Safety Administration (NHTSA) has demonstrated that vehicle parameters can change up to 40% for passenger sedans, so fuzzy assumptions of the vehicle's parameter are not reasonable assumptions [1, 3]. An adaptive controller could be proposed to improve the vehicle's stability, but because rollover events are fast and happen typically only once for a vehicle, it is difficult to train the parameters in the controller. Therefore, a method of estimating the parameters of a vehicle was desired.

In the course of investigating methods to improve vehicle stability it became clear that there was also a need for computationally efficient methods that could be employed in real-time for the vehicle parameter estimation. The more parameters that one attempts to estimate, the more computational cost is involved. For vehicle systems, and in general, the more generalized the model the larger the number of parameters that must be accounted for. The Generalized Polynomial Chaos method is only computationally efficient and tractable for real-time implementation in state-space systems for low complexity systems and a limited number of parameters (1-3). When applied to regression systems however, it is proven capable of performing, in real time, the estimation of several parameters, while maintaining the structure of the non-linear multi-input multi-output equations, unlike recursive least-squares. The method also demonstrates the ability to individually adjust the forgetting factor for each parameter.

1.2 Overview

This research is at the crossroads of vehicle dynamics and parameter estimation. Vehicle dynamics is a challenging topic that has been studied for many years, and the research has been primarily oriented towards vehicle models developed for implementation of control systems and rough approximation models. The range of vehicle models is very wide. They can be as complex as extremely detailed multi-body dynamics models and as simple as bicycle models. All of these models have their advantages and disadvantages, and, depending on the requirements, each has a place for implementation. The primary metric that can be used to distinguish between them is the accuracy and corresponding computational complexity. It is in general true that accuracy is proportional to computational cost and in vehicle dynamics this is no different.

Parameter estimation is a difficult task. There are two general types of parameter estimation techniques. The first one is model reference parameter estimation, which requires a state-space or other integrable model that is being integrated forward in time. The methods that can be shown to be stable through this parameter update are based on Lyapunov stability theory. However, those methods trade parameter convergence and accuracy for stability. Measurement noise and modeling errors will likely prevent the parameters from converging to the 'true' values, even if the sufficient richness requirement is met. True values being defined as the best value for a given type of modeling. The model reference parameter estimation methods are typically hard to stabilize due to the errors that are created in the system because of integrating forward with incorrect parameter values. The system has to perform both the state estimation and the parameter estimation from the same error residual, and, by definition, the system is nonlinear.

The second type of parameter estimation is through polynomial or regressive systems. These techniques use algebraic equations, or input-output relations, with uncertain parameter coefficients. In general, these methods are stable, because they are typically linear and there are many tools available to perform the parameter estimations.

Several vehicle models are investigated in the course of this research. These models range from complex integrable models to simplified roll and pitch plane models. Because this research aimed at creating a generalizable method that could be implemented to extract vehicle parameters, many effects not typically covered in the literature were investigated. There are several difficulties in modeling vehicle dynamics due to the many parameters and effects that cannot be directly measured. The goal of this work was to create a model that can be developed and implemented on a vehicle with as little information as possible, with the intention of being real-time viable. Most of the sensors are accelerometers or displacement sensors; however, the one possibly difficult requirement for implementation is the knowledge of the suspension forces. Most models use the suspension stiffness and damping to recreate these forces in conjunction with state observers, which may also be feasible with the models presented here. The more accurately the suspension forces can be known, the more accurately the parameter values of the vehicle will be estimated. Future work would include using strain gauges or other low-cost force and non-intrusive sensors so that the method is easily applicable to a vehicle.

2 Literature Review and Background

This chapter covers the background knowledge that was used in this study. Most of the information included here is at a high level, with key citations used to reinforce points. The sections that are covered are Parameter Estimation, Kalman and Particle Filters, Vehicle Modeling and Dynamics, Vehicle Parameter Estimation, and Vehicle Stability. Some of these sections are for mathematical background, and others, like the Vehicle Parameter Estimation, are presented as historical background.

2.1 Parameter Estimation Review

The field of parameter estimation can be split into two different areas: parameter regression and state-space systems. Regression systems are input-output relational equations, where a set of data is fed into the system and a set of outputs are produced from the internal model. These systems can be further divided into two more categories: “linear in the parameters” and “non-linear in the parameters”.

The “linear in the parameters” case is the most studied and well-known area of parameter estimation [4-7]. There are two very easy-to-use methods for this type of estimation: Least Squares (LS) and Recursive Least Squares (RLS). The LS method is an averaging technique across a set of data in an offline or batch estimation setting. The RLS method is used for real-time or online system parameter estimation. There are also Stochastic Gradient [8] or Stochastic Approximation [9] methods for situations when the covariance of the system is hard to recover, but these schemes have much slower convergence rates. There are a plethora of other methods, not mentioned here, all with their own benefits and drawbacks.

For “non-linear in the parameters” systems the estimation is much harder. For systems with explicit parameters there are two descriptors that define the type of method. The first type is “iterative”, which is the technique where the system space is locally mapped through perturbing the parameter values. The second type is “derivative”, where the system space is locally mapped through the gradient of the parameter values. This system space is usually the residual or the error between the desired output and the model output. Examples of iterative methods: Raster [4], Gibbs Sampling [10], Metropolis-Hastings [10]. Examples of derivative methods are: Newton-Raphson [11], non-linear least squares [4]. There are also methods that use both operations, such as the Levenberg-Marquart algorithm [12].

In general, all linear parameter regression methods will perform faster than real-time, as their parameter spaces are linear and do not have very complicated mathematics. Non-linear parameter estimation methods are not so easily implemented. Some of them, like the Gibbs or Metropolis-Hastings methods, are extremely computationally expensive and are only run in offline implementations [10].

State-space systems are a completely different sort of problem [13]. Usually described as System Identification, it can be difficult to estimate the parameters of a system explicitly in terms of the desired parameters. For “linear in the parameter” systems, this may be possible, but typically the filters used (Kalman) are employed to correct the errors in the states, and the parameter values are ignored. For non-linear systems, it is common to use black box models [4], such as Neural Networks, which are capable of highly complex function approximation. The parameters, however, are unobservable and, in most cases, completely meaningless to the physical parameters of the system. Adaptive control is a field that has been highly influential to the estimation of parameters in state-space systems. However, the parameter estimations are typically used for controlling a system with unknown parameters and less with the intention of estimating the uncertain parameters. This type of approach typically results in parameters that never converge, as the mathematical mechanics are used maintain system stability, and not for parameter estimation.

2.1.1 Kalman Filters

The Kalman Filter has been around since the 1960s. The filter is the extension of the Luenberger Observer when calculated to address the addition of process and measurement noise. The original filter is simply the Kalman Filter [14]. There is a method using linearization of the state and measurement matrices for the tracking of systems with nonlinear equations in the Kalman filter, known as the Extended Kalman Filter (EKF), that is a very popular filter in both state tracking and parameter estimation [15, 16]. The EKF is a very popular method, but it is also one that does not maintain the stability of the linear Kalman Filter, and therefore, further research has been performed.

The more recent developments in the Kalman Filter field are the Ensemble Kalman Filter and the Unscented Kalman Filter. The Ensemble Kalman Filter (EnKF) uses the same EKF equations but does so by using many iterations of the EKF across a space that adds statistical saturation to the estimation in addition to the linearization [17]. The Unscented Kalman Filter builds upon that by making a lateral move. The UKF uses the iterations of the EnKF but uses a different transform to calculate the covariance that does not require linearization [18-21]. The UKF is capable of handling significantly more nonlinear behavior than the traditional EKF.

In this research, the method of Generalized Polynomial Chaos (gPC) is applied to the Extended Kalman Filter. Previous implementations of the gPC mathematics have been used to propagate the uncertain parameters through state-space vehicle models to create a stochastic vehicle model [22-31]. There is a significant paradox that is created within the gPC-EKF method that is created by the blending of the gPC method with the EKF equations, as Blanchard noted and attempted to explain [25]. The filter exhibits problems with state estimation and convergence to the correct parameter value. The goal of the research presented here on parameter estimation theory is to thoroughly explain the problem the gPC-EKF exhibits in the earlier study, and to correct it.

The gPC-EKF mathematics is then expanded to non-linear regression systems. Past studies have shown that the gPC method is computationally efficient [22, 23, 32-35] for a small number of uncertain parameters, and quite capable of real-time online estimation.

2.2 Vehicle Modeling and Dynamics

This section highlights the difficulties involved with the modeling of vehicle systems. Some vehicle models may work in synthetic environments, but in the real world, when subject to regular driving conditions, they may not work.

Modeling of ground vehicle systems requires information about many effects, both internal and external to the vehicle. The internal effects of the vehicle are due to the kinematics and dynamics of the suspension components. The external effects are centrifugal accelerations, aerodynamics, and gravity.

The kinematics of a ground vehicle constrains the motion of the vehicle; this implies that simple dynamical models often used in vehicle dynamics simulations do not have very high accuracy. All forces applied to and generated by the vehicle are summed at the roll and the pitch centers of the vehicle. These centers are a consequence of the way the vehicle's motion is constrained by the various linkages and other kinematic connections, since no vehicle has a simple spring-damper assembly connecting the chassis to the wheels, as it is habitually considered in simplified vehicle models.

The primary external effects that are very difficult to model are the aerodynamic effects. Most authors neglect the aerodynamic effects on the vehicle. For an SUV with a value for $C_d A \approx 1$ (where C_d is the drag coefficient, and A is the area of the transverse plane to the direction of flow), the aerodynamic drag force applied to a vehicle traveling at $110\, km/h$ is approximately 575 N. For most systems, this might be small enough to be neglected, especially at slower speeds. For parameter estimation, though, it is important to model every possible effect, since the intent is to be able to capture even small variation in vehicle parameters in real time.

2.2.1 Aerodynamics

The aerodynamic effects that act upon a vehicle are very complicated. These effects have small subtleties that are very difficult to model or to estimate. For a ground vehicle, there are two primary aerodynamic effects: The aerodynamic drag force and the aerodynamic lift force. These two primary effects cause many different effects upon the vehicle.

The aerodynamic drag is caused primarily by the vehicle travelling forward. However, this is also caused by wind effects on the vehicle. Most authors, if they account for the aerodynamic effects, map the aerodynamics to the vehicle’s forward speed. This is, in general, not correct, as it does not account for wind, wake and drafting effects. However, as anyone in fluid dynamics knows, this is a very difficult area and they make the assumption that these added effects are trivial. For the most part, they are, but as many have seen, a strong wind can knock over a vehicle.

There is a major problem in addressing how to add in these effects to the vehicle model. The problem arises from not being able to accurately describe an aerodynamics model for the vehicle. One element is estimating where the aerodynamic effects impact the vehicle. These forces are not necessarily summed at the center of gravity (CG) of the vehicle either, which means that there will be moments created on the vehicle body. The faster the vehicle travels, the more roll moment will be created, for example. This is further compounded by the fact that the vehicle’s aerodynamic center can change easily during operation. Opening a window, putting an arm out of the window or using pop-up headlights will all have an effect on the location of the aerodynamic center and the magnitude and direction of the aerodynamic forces.

The aerodynamic lift component of the vehicle can be a nontrivial effect. In general, it should not be excluded, especially for race cars or other vehicles that design their aerodynamics to increase their down-force for better traction. The aerodynamic lift force has similar problems to the aerodynamic drag force in terms of modeling. An accurate location of where to sum the effects of the aerodynamics lift force is not easily recoverable, and special care should be paid to this when dealing with vehicles that have large lift force capabilities, as the moments generated will certainly not be trivial.

Here, for some of the models, the aerodynamic effects are considered, but they are assumed to impact the system at the vehicle’s CG. This approach seems to be the most appropriate in this case, allowing the algorithm to incorporate some amount of the aerodynamic effects into the global equations of the vehicle. The coefficients for these effects are scaled values of the actual ones, and the scaling is proportional to the error in the location for the summation of these effects. There is an error in this term, as the aerodynamics of the vehicle are mapped to the speed of the vehicle, due to ignoring the added effects of wind, drafting and wake. This error is partially compensated by the fact that the parameter is able to update in real-time to help compensate. Obviously this will lead to an incorrect value for the parameter, but the goal is not to estimate the aerodynamic effects on the vehicle to high accuracy, but rather to incorporate them so that their effects do not impact the mass and CG estimations.

An excellent review of vehicle aerodynamics can be found in [36].

2.2.2 Suspension Nonlinearities and Roll/Pitch Center

The roll and pitch centers of a vehicle are points in space (calculated under a given vehicle kinematic configuration) that do not have a physical location. Many authors [37-40] have discussed the roll and pitch centers of a vehicle and have detailed their location when the suspension is in the undisturbed state and stationary. All of these authors have either agreed that mapping the position of the roll and pitch centers is difficult when the suspension is moving or have neglected to discuss the topic. One of the main difficulties in addressing the estimation of the roll and pitch centers when the suspension is moving is the fact that the location of these points can move arbitrarily large distances, and there are no bounds on the limits of their location.

Extracting the CG height from experimental data is extremely difficult because the set of equations are evaluated at the roll or pitch center. The resulting estimation of CG height is the distance between the CG and the roll or pitch center.

The typical strategy for estimating parameters for vehicle systems is to use a state-space method to model the vehicle's dynamics. If the parameters of the vehicle are known, then this model in combination with a state estimator may function. However, because state estimation with unknown or uncertain parameters is a nonlinear process, the addition of modeling errors creates an environment that is unlikely to be able to produce correct parameter estimates. Lyapunov techniques will be able to preserve the model's stability but will not be able to guarantee parameter convergence.

Adding to these challenges are the non-linear elements in the suspension that are very difficult to measure. The primary candidate that causes problems is the friction in the suspension elements. Moreover, the time evolution of the system may make specific modeling impossible, as elements change over time. The suspension friction of the vehicle in this research is different depending on the direction of the velocity and the displacement of the strut, for example.

2.3 Vehicle Parameter Estimation Review

In this study, the method of Generalized Polynomial Chaos (gPC) is applied to the parameter regression. Previous implementations of the gPC mathematics have been used to propagate the uncertain parameters through state-space vehicle models to create a stochastic vehicle model [22-31]. In the present work, the system is non-linear in the parameters, but does not involve a state-space system. The gPC has been shown to be computationally efficient [22, 23, 32-35] and quite capable of real-time online estimation for a low number of parameters. The gPC mathematics are blended with the Extended Kalman Filter to enable the parameter estimation [25].

Much of vehicle dynamics modeling and analysis is performed with the intention of controller development. This leads to models that have large assumptions or very simplified models that may work well for a feedback controller but are unusable for parameter estimation. This is primarily because parameter estimations require high accuracy between the model and the data. Therefore, much of the literature on vehicle dynamics is unhelpful in the estimation of vehicle parameters.

The remaining body of literature for parameter estimation and vehicle dynamics can be subdivided roughly into three categories: validation through simulation, validation through special maneuvers, and validation with assumptions.

Validation of estimation techniques through simulation is an approach that allows the development of new ideas or mathematical tools in an environment that can be highly controlled. Unfortunately, due to the nature of this high level of control, it is difficult to simulate a real environment. Random effects like wind, uneven pavement, changes in road grade, drafting, potholes, engine vibrations, and vehicle kinematics are hard to simulate or simulate in real-time and, subsequently, they make it difficult to truly validate a method. What primarily makes these hard to validate, or impossible to use in reality, are the assumptions that are applied to the vehicle models: knowledge of road grade/bank angles or the ignoring of those effects [31, 41-45]; knowledge of or the ignoring of aerodynamic effects [46-48]; handling of or the ignoring of the roll center [43, 49].

Validation based on data collected during special maneuvers is, in some sense, similar to simulation. When a method is validated against special maneuvers' data, the environment is not quite as rigidly controlled as during a computer simulation, but it is still strongly controlled and enables what appears to be a validation. The validation thus performed will not hold under normal driving conditions, though (because one does not always drive in a constant radius turn on a flat surface, for example, if that was the maneuver against which the model/estimator was validated) [25, 27, 41]. So special maneuver studies are not necessarily wrong, but the results obtained from such studies need to be considered in the context of the respective maneuver; for example, Currier [46] argues that they may be needed when the vehicle is driven without significant excitation, and Fathy [5] posits the same idea. Which is essentially the idea that sufficient richness of signals needs to be maintained, and special maneuvers may need to be conducted to provide such information.

The third type of validation identified in the literature is validation of a method on a drive cycle, but with certain assumptions. The literature in this area is relatively limited, and these studies start to show many of the problems typically encountered with a standard drive cycle: sufficient richness, estimation of aerodynamic forces, rolling resistance estimates, roll center mechanics, pitch center mechanics, and instrumentation error.

The studies in this area use the drivetrain information to estimate the road grade and vehicle mass [5, 50, 51]. The parameters that are assumed to be known are the tire rolling resistance and the aerodynamic properties. The assumption made here about the aerodynamic properties is that there are no external wind effects, and that the wind speed is measurable simply by the speed of the vehicle. As discussed previously, this is not a good assumption [52]. Rolling resistance is not easily measured during vehicle operation, and extensive driving could pose a problem as air pressure and temperature change, which affect the rolling resistance of the vehicle. Also, this strategy is only applicable to on-road driving, where the road surface is smooth. Once the road surface becomes rough, the rolling resistance changes non-trivially, and the initial estimation of the rolling resistance may be imprecise.

2.4 Vehicle Rollover and Stability Review

Vehicle rollover events are highly nonlinear events. These events can be quite difficult to distinguish and detect, and accurate vehicle parameters are essential to decreasing the response time for detecting a vehicle rollover event.

There are multiple cases of vehicle rollover events. There are two main types of vehicle rollover: tripped and untripped. These two categories themselves have subcategories which have their own components that make detecting them difficult.

Tripped rollover is where the vehicle strikes a physical object. This impact could be significant enough to cause the vehicle to rollover, or to simply lift part way up off of the ground. The ability to predict this difference requires good knowledge of the vehicle's parameters (mass, center of gravity height, etc.).

Untripped rollover can be more difficult to detect. Unlike tripped rollover, where many sensors experience significant excitation, untripped rollover can be subtle. There are two primary categories of untripped rollover. An untripped rollover event will fit into at least one of these categories and more commonly be split part way between them.

The first untripped rollover category is a fast rollover event. The fast rollover event is where a resonant steering maneuver is performed that causes the vehicle to rollover quickly. This will induce high excitation to some of the vehicle’s sensors but may negatively impact other sensors on the vehicle. For example, an accelerometer on the vehicle that is measuring the lateral acceleration of the vehicle will register high excitation, but the suspension force sensors may not react as quickly, because the dynamics prevent the strut from extending and reducing the force measurement as quickly. The accelerometer will tell you that your vehicle is rolling over, but the force sensors will tell you that your vehicle still has contact forces on all suspension struts. The force sensor has a reading because as the vehicle rolls over the suspension still has some residual compression in it that takes time to dissipate due to the wheel assembly having mass, or in the case of the vehicle used in this research: non-trivial friction.

The second category of untripped rollover is the slow rollover event. This will happen for a vehicle that is driving at a constant speed around a circle with constantly increasing speed. The vehicle’s rotation occurs slowly enough that the suspension’s force sensors give accurate information about the vehicle’s contact with the ground. However, the vehicle’s accelerometers may not be showing a high enough excitation signal to properly alert the rollover control system. This is exponentially compounded when the vehicle’s parameters are not known well.

The interim case is where elements of both categories are present. This is where the vehicle rolls over at moderate speed, where the excitation to the accelerometers is high and the time delay in the suspension force sensors is trivial compared to the time of rollover.

3 Parameter Estimation and the Generalized Polynomial Chaos Extended Kalman Filter

The research into parameter estimation theory henceforth is dedicated to the study of the Generalized Polynomial Chaos (gPC) Extended Kalman Filter (EKF) and its idiosyncrasies. The filter shows many interesting phenomena, such as the ability to change the parameter's distribution shape in real-time to match time evolution behaviors.

This chapter will cover the basics of the gPC mathematics, the basics of the EKF and the integration of the gPC with the EKF. It will progress into the problems of the gPC-EKF and several proposed improvements with examples of how each improvement interacts with the different problems in the filter.

The next subsection will investigate choices of parameter update distributions and underlying basis function choices and their effects upon the filter's convergence rates and steady state tracking. The following investigation demonstrates the filter for systems with multiple parameters to be estimated, the requirement for sufficient excitation and how the filter can show if that requirement is not met.

Finally, a technique that can be employed to estimate whether the filter has converged will be demonstrated and discussed.

3.1 Generalized Polynomial Chaos

A deterministic system has a vector of state variables:

$$\boldsymbol{x} = [x_1, x_2, \dots, x_n]^T \tag{1}$$

A system of interest may also contain a number of uncertain parameters. Each uncertain parameter, p_i, when described through a gPC expansion, is expanded in an infinite series:

$$p_i = \sum_{j}^{\infty} p_i^j \, \psi_i^j(\xi_i) \tag{2}$$

The term ψ is a single dimensional basis function. These basis functions are orthogonal or orthonormal functions, such as Legendre, Jacobi, or Hermite polynomials, depending on the type of probability distribution of the uncertain parameter. The basis functions are functions of a random variable, ξ_i, which spans the domain of the basis function, and subsequently the uncertain parameter's domain. The index i corresponds to the i^{th} uncertain parameter.

When a parameter in the system is given a distribution the state vector has to be expanded in a distribution to account for the various forward solutions that the parameter distribution will cause. The i^{th} state variable's expansion:

$$x_i = \sum_{j}^{\infty} x_i^j \, \Psi^j(\boldsymbol{\xi}) \tag{3}$$

The state variables have a distribution that takes into account each individual parameter distribution. This comprehensive distribution is multidimensional, and is defined through a tensor product of the individual distributions:

$$\Psi^{j}(\xi_1 \dots \xi_n) = \prod_{i=1}^{n} \psi_i^{l_i}(\xi_i), \quad j = 1,2, \dots S \tag{4}$$

$$l_i = 1,2, \dots PO$$

Technically, a gPC expansion should be expanded into an infinite number of terms, but, for computational purposes, they are truncated at a selected polynomial order (PO) and number of parameters (n). This truncation introduces some error into the system. The total number of tensor products in the multidimensional basis function for a specified truncation is:

$$S = \frac{(n+PO)!}{n!\,PO!} \tag{5}$$

For a system with two uncertain parameters and a maximum polynomial order of three, there are 10 terms in the state variable's series expansions. For $i = 1, \dots, 10$ these are:

$$\begin{array}{ccccc} \psi_1^0\psi_2^0 & \psi_1^0\psi_2^1 & \psi_1^0\psi_2^2 & \psi_1^0\psi_2^3 & \psi_1^1\psi_2^0 \\ \psi_1^1\psi_2^1 & \psi_1^1\psi_2^2 & \psi_1^2\psi_2^0 & \psi_1^2\psi_2^1 & \psi_1^3\psi_2^0 \end{array} \tag{6}$$

3.1.1 Application to a State-space System

A deterministic system has a state-space representation as shown in:

$$\dot{\boldsymbol{v}} = F(t, \boldsymbol{x}, \boldsymbol{v}; \boldsymbol{p}) \tag{7}$$

Expanding this formulation using the gPC method, the state-space form becomes:

$$\dot{\boldsymbol{x}}^i = \boldsymbol{v}^i \tag{8}$$

$$\sum_{i=1}^{S} \dot{\boldsymbol{v}}^i \, \Psi^{\mathrm{i}}(\xi) = F\left(t, \sum_{m=1}^{S} \boldsymbol{x}^m \, \Psi^{\mathrm{m}}(\xi), \sum_{m=1}^{S} \boldsymbol{v}^m \, \Psi^{\mathrm{m}}(\xi); \sum_{m=1}^{S} \boldsymbol{p}^m \, \Psi^{\mathrm{m}}(\xi)\right)$$

The system is now described stochastically, with known basis functions, but unknown coefficients. There are several methods that can be used to calculate the coefficients, generally grouped in two main categories of methods: intrusive and non-intrusive.

One of the intrusive methods is the Galerkin Projection, which leverages the properties of the orthogonal basis functions in the inner product. Using this approach, equation (8) becomes:

$$\dot{\boldsymbol{x}}^i = \boldsymbol{v}^i \tag{9}$$

$$\dot{\boldsymbol{v}}^i < \Psi^{\mathrm{i}}, \Psi^{\mathrm{i}} > = < F\left(t, \sum_{m=1}^{S} \boldsymbol{x}^m \, \Psi^{\mathrm{m}}(\xi), \sum_{m=1}^{S} \boldsymbol{v}^m \, \Psi^{\mathrm{m}}(\xi); \sum_{m=1}^{S} \boldsymbol{p}^m \, \Psi^{\mathrm{m}}(\xi)\right), \Psi^{\mathrm{i}} >$$

This method can be used to calculate the exact values of each of the coefficients. The tradeoff is that there is a possibly large computational cost to calculate the inner products. The inner product calculation can typically be performed offline.

The method used in this study is one of the non-intrusive methods, which is a collocation technique that iterates through the system at each time step to approximate the solution of the inner products. The collocation technique introduces error into the system by approximating the integrals of the Galerkin projection method. If carefully chosen points are used to span the space over which the integration is performed, a high accuracy solution for the coefficients can be recovered. A matrix of collocation points, μ, is chosen, at which points the system will be evaluated. It can be shown that, for a viable solution, at least S number of points must be used, but for high accuracy, $3S \leq Q \leq 4S$ number of points are chosen [53]. The collocation points are generated, for a small (1 – 3) number of uncertain parameters, using the Hammersley distribution, or for a large number of parameters (>4) by a random number sequence. Using the collocation technique, the system is described by equations (10-11).

$$\boldsymbol{\mu}^i = \left[\mu_1^i \dots \mu_d^i\right]^T \, for \; all \; 1 \leq i \leq Q \tag{10}$$

$$\dot{\boldsymbol{x}}^i = \boldsymbol{v}^i \tag{11}$$

$$\sum_{j=1}^{S} \dot{\boldsymbol{v}}^j \, \Psi^j\left(\boldsymbol{\mu}^i\right) = F\left(t, \sum_{m=1}^{S} \boldsymbol{x}^m \, \Psi^{\mathrm{m}}(\boldsymbol{\mu}^i), \sum_{m=1}^{S} \boldsymbol{v}^m \, \Psi^{\mathrm{m}}(\boldsymbol{\mu}^i); \sum_{m=1}^{S} \boldsymbol{p}^m \, \Psi^{\mathrm{m}}(\boldsymbol{\mu}^i)\right), \qquad 1 \leq i \leq Q$$

The set of values of $\Psi^j\left(\mu^i\right)$ can be written in matrix form as $A_{i,j} = \Psi^j\left(\boldsymbol{\mu}^i\right),\; 1 \leq j \leq s,\; 1 \leq i \leq Q$. This allows us to write the system of equations (10-11) as:

$$\sum_{j=1}^{S} A_{i,j} \dot{\boldsymbol{v}}^j \; = F\left(t, \sum_{m=1}^{S} A_{i,j} \, \boldsymbol{x}^m, \sum_{m=1}^{S} A_{i,j} \, \boldsymbol{v}^m; \sum_{m=1}^{S} A_{i,j} \, \boldsymbol{p}^m\right), \qquad 1 \leq i \leq Q \tag{12}$$

If we rewrite these arguments in terms of a single term, such as:

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

Then equation (12) becomes:

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

The coefficients of the series are obtained by integrating equation (14) forward in time for each of the collocation points. Since there are Q number of collocation points, for each step forward in time, the model must be integrated Q times.

To recover the stochastic coefficients one uses the inverse of the A matrix:

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

$$\boldsymbol{v}^i(T) = \sum_{j=1}^{Q} (A^{\#})_{i,j}\, \boldsymbol{V}^j(T) \tag{16}$$

Where $A^{\#}$ is the Moore-Penrose pseudo-inverse of the A matrix. As a note, if $A^{\#}$ is not full rank, the solutions are questionable at best. If nothing else, the covariance matrix that is calculated from these solutions becomes entangled between multiple states and not just between two states per covariance matrix term.

3.1.2 Extension of the Method to a Regression System

A function may be expressed as:

$$G(\boldsymbol{u}, t; \boldsymbol{p}) = \boldsymbol{a} \tag{17}$$

Where $\boldsymbol{u}$ is a vector of inputs to the system, t is time, $\boldsymbol{p}$ a vector of parameters, and $\boldsymbol{a}$ is a vector of outputs. Estimation of the parameters $\boldsymbol{p}$ may be performed through a regression technique. An example is Newton's second law:

$$Force = \; m\, a \tag{18}$$

The function $G(\boldsymbol{u}, t; \boldsymbol{p})$ is $mass$ times $acceleration$ $(u = a, p = m)$. $\boldsymbol{a}$ from equation (17) is the measured force. To solve for the parameters using the gPC-EKF the function is expanded in a similar way as the state-space system:

$$\sum_{i=1}^{S} \boldsymbol{a}^{i}\ \Psi^{\mathrm{i}}(\boldsymbol{\xi}) = G\left(\boldsymbol{u}, t; \sum_{m=1}^{S} \boldsymbol{p}^{m}\ \Psi^{\mathrm{m}}(\boldsymbol{\xi})\right) \tag{19}$$

The Galerkin projection or the collocation techniques can be applied to this system to solve for the coefficients. The advantage is that for a system without a state-space representation, the covariances can still be calculated, which enables integration into the Kalman Filter structure.

The covariances of the state variables is easily calculable through gPC. The covariance of variable *d*, with variable *j*, at time *k* is calculated, for orthonormal basis functions, as:

$$cov\left(x_{d,k}, x_{j,k}\right) = \sum_{i=2}^{S} x_{d,k}^{i}\ x_{j,k}^{i} \tag{20}$$

3.2 Generalized Polynomial Chaos Extended Kalman Filter

In section 3.2.1 the Extended Kalman Filter (EKF) is defined. The EKF is then merged with the gPC expansions in section 3.2.2.

3.2.1 Extended Kalman Filter

The filtering technique of the Extended Kalman Filter (EKF) can be applied to the Generalized Polynomial Chaos (gPC) expansions. The basic EKF is as follows:

A differential equation system can be described in state-space form as:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{v} \tag{21}$$

Where $\boldsymbol{x}$ is the state vector, and $\boldsymbol{w}$ is the vector of process noise. The system measurement equation is defined as:

$$\boldsymbol{z} = \boldsymbol{h}(\boldsymbol{x}) + \boldsymbol{w} \tag{22}$$

$\boldsymbol{h}$ is the observation matrix that incorporates the state vector into an output solution. $\boldsymbol{v}$ is the vector of the sensor noise. The Kalman Filter is designed for linear systems. The EKF linearizes the system mechanics in an attempt to produce an approximately linear system. This is done through linearizing the system dynamics and observation matrices and evaluating them at each time step, k:

$$F_k = \frac{\delta f(\boldsymbol{x})}{\delta \boldsymbol{x}}|_{x=x_k} \tag{23}$$

$$H_k = \frac{\delta h(\boldsymbol{x})}{\delta \boldsymbol{x}}|_{x=x_k} \tag{24}$$

The resultant EKF equation is then written as:

$$\boldsymbol{x}_k^u = \boldsymbol{x}_k^f + K_k\ (\boldsymbol{z_k} - H_k\ \boldsymbol{x_k}) \tag{25}$$

The system takes the initial forecast (or model solution), $\boldsymbol{x}_k^f$, and updates it through the Kalman update equations, $\boldsymbol{K_k}$, and the residual, $(\boldsymbol{z_k} - H_k * \boldsymbol{x_k})$, to update the state variables, $\boldsymbol{x}^u$. The Kalman update gain is defined as:

$$K_k = M_k\ H_k^T\ (H_k\ M_k\ H_k^T + R_k)^{-1} \tag{26}$$

The covariance matrices, M_k, and, P_k are thus obtained as:

$$M_k = \Phi_k \, P_{k-1} \, \Phi_k + Q_k \tag{27}$$

$$P_k = (I - K_k \, H_k) \, M_k \tag{28}$$

The system covariance matrix, M_k, is created through the functional matrix, $\boldsymbol{\Phi_k}$**,** and the updated system covariance, P_{k-1}. The R_k matrix is the measurement noise matrix, defined as:

$$R_k = E(\boldsymbol{v} \, \boldsymbol{v}^T) \tag{29}$$

E is the mathematical expectation operator. Q_k is the matrix that describes the discrete process noise matrix, through the process noise matrix, Q.

$$\boldsymbol{\Phi_k = e^{F_k \, T_s}} \tag{30}$$

$$Q = E(\boldsymbol{w} \, \boldsymbol{w}^T) \tag{31}$$

$$Q_k = \int_0^{T_s} \boldsymbol{\Phi_k} \, Q \, \boldsymbol{\Phi_k} \, \boldsymbol{dt} \tag{32}$$

More thorough details and explanations for EKF can be found in [14, 16, 54].

3.2.2 EKF Integration with gPC Expansions

Normally, the covariance matrices are created through the functional form of the system dynamics matrix for the EKF. Because the gPC mathematics creates an estimation of the covariance matrix (equation (20)) there is no need to use the functional form of the matrix or its linearization.

We formally expand the state-space system to include the parameters. For time-invariant parameters, the state-space vector is defined as [27]:

$$\boldsymbol{x} = [x_1 \dots x_n, p_1 \dots p_d]^T \tag{33}$$

Expanding equation (21) to include the parameters:

$$\begin{bmatrix} \dot{\boldsymbol{x}}_{\boldsymbol{n}} \\ \dot{\boldsymbol{p}} \end{bmatrix} = \begin{bmatrix} f(\boldsymbol{x}) + \boldsymbol{v} \\ \boldsymbol{f_p} \end{bmatrix} \tag{34}$$

For time-invariant parameters, $\boldsymbol{f_p} = 0$. For time-varying parameters $\boldsymbol{f_p}$ is defined however those parameters vary.

The Extended Kalman Filter equations become:

$$\boldsymbol{x}_k^u = \boldsymbol{x}_k^f + K_k\left(\boldsymbol{z}_k - H_k\, \boldsymbol{x}_k^f\right) \tag{35}$$

The gPC method expands each element of the state vector into a series, which expands equation (33) into:

$$\boldsymbol{x} = \left[\sum_{\boldsymbol{i=1}}^{\boldsymbol{S}} x_1^i\, \Psi^{\mathrm{i}}(\boldsymbol{\xi}) \quad \cdots \quad \sum_{\boldsymbol{i=1}}^{\boldsymbol{S}} x_n^i\, \Psi^{\mathrm{i}}(\boldsymbol{\xi}) \quad \sum_{\boldsymbol{i=1}}^{\boldsymbol{S}} p_1^i\, \Psi^{\mathrm{i}}(\boldsymbol{\xi}) \quad \dots \quad \sum_{\boldsymbol{i=1}}^{\boldsymbol{S}} p_n^i\, \Psi^{\mathrm{i}}(\boldsymbol{\xi})\right]^T \tag{36}$$

The Kalman update equation

integrates the covariances generated from the gPC theory. This is advantageous, since it removes the need to linearize the system dynamics equations; this removes one layer of error from the system. This advantage is only realized if one uses enough terms in the series representations of the variables; using too few terms makes the estimate worse than if the functional matrices were linearized. The observation matrices may still need to be linearized. When evaluated at each time step, the functions that depend on the uncertain parameters are evaluated at the median value of the variable’s probability density function (PDF). A more thorough explanation of this can be found in [27, 55]. The sensor noise matrix remains the same, but the process noise matrix is tuned a bit differently, as explained later.

$$\boldsymbol{x}_k^{u,i} = \boldsymbol{x}_k^{f,i} + K_k\left(\boldsymbol{z}_k \delta(i-1) - H_k\, \boldsymbol{x}_k^{f,i}\right) \tag{37}$$

$$K_k = cov(\boldsymbol{x}_k, \boldsymbol{x}_k)\, H^T (R_k + H\; cov\; (\boldsymbol{x}_k, \boldsymbol{x}_k) H^T)^{-1} \tag{38}$$

The term $\delta(i-1)$ is the delta function. If the system were to be solved through the Galerkin Projection method, the sensor data $\boldsymbol{z}_k$ would be expanded into a series itself. However, since the sensor data is a given value, having a measured value and assumed to have no distribution, the series representation contains only the first term. Therefore, when each of the coefficients of the state-space vector is updated, only the first terms take into account the sensor data. The index i denotes the coefficients of the series in equation (37).

This is where the problems in the filter appear. When equation (36) is performed, the first term in the series is driven to the sensor value, as:

$$\boldsymbol{x}_k^{u,1} = \boldsymbol{x}_k^{f,1} + K_k(\boldsymbol{z}_k - H_k\, \boldsymbol{x}_k^{f,1}) \tag{39}$$

Where the rest of the series coefficients $(2, \dots, S)$ are driven to zero:

$$\boldsymbol{x}_k^{u,2\dots S} = \boldsymbol{x}_k^{f,2\dots S} + K_k(\boldsymbol{0} - H_k\, \boldsymbol{x}_k^{f,2\dots S}) \tag{40}$$

As defined above, the covariance is calculated through the series coefficients (2…S), which are being driven to zero by the EKF. This drives the Kalman update to zero, which makes the state update:

$$\boldsymbol{x}_k^{u,i} = \boldsymbol{x}_k^{f,i} + (0)\big(\boldsymbol{z}_k \delta(i-1) - H_k\, \boldsymbol{x}_k^{f,i}\big) \tag{41}$$

Because:

$$K_k = cov(\boldsymbol{x}_k, \boldsymbol{x}_k)\, H^T (R_k + H\, cov(\boldsymbol{x}_k, \boldsymbol{x}_k)\, H^T)^{-1} = 0 \tag{42}$$

3.3 gPC-Bayesian Method

Bayesian parameter estimation typically performed through Gibbs sampling or Metropolis-Hastings, is a powerful method of parameter estimation. The Bayesian framework for parameter estimation is defined as:

$$P[p|\boldsymbol{z}] = \frac{P[\boldsymbol{z}|p]\, P[p]}{P[\boldsymbol{z}]} \tag{43}$$

For the purposes of estimation, the term $P[z]$ can be ignored as a constant scaling factor. This reduces equation (43) to:

$$P[p|\boldsymbol{z}] \; \propto P[\boldsymbol{z}|p] P[p] \tag{44}$$

$P[p|z]$ is the posterior probability density function of the parameter values given the data. The term $P[z|p]$ is the statistical distribution of the error between the signal and the model. For a normal distribution, this is defined as:

$$P[\boldsymbol{z}|p] = e^{-\frac{1}{2}\sum_{t=T_i}^{T_f}(\boldsymbol{z}_t - h_t(\boldsymbol{x}))^T R_t^{-1} (\boldsymbol{z}_t - h_t(\boldsymbol{x}))} \tag{45}$$

Where z_t is the signal vector at time t, and h_t is the model output vector at the same time, t. The term R_t is the signal noise matrix. The term $P[p]$ is the prior distribution of the parameters. x is the state-space vector. This is one of the powerful tools of the Bayesian framework, as it incorporates previous knowledge about the distributions of the parameters. This term is used to allow the estimator to learn.

A sequence of measurements is collected over a time span, $[T_i \dots T_f]$, and the distribution $P[p|\boldsymbol{z}]$ is calculated. The Maximum A Posteriori (MAP) estimation finds the values of the parameters that maximize this function $P[p|\boldsymbol{z}]$. The probability density function of $P[p|\boldsymbol{z}]$ is then fed into the next estimation as the distribution for $P[p]$.

The values being estimated here are not the values of the parameters, p, but the values of the random variables, ξ. This redefines equation (45) as:

$$P[\boldsymbol{z}|\xi] = e^{-\frac{1}{2}\sum_{t=T_i}^{T_f}\left(\boldsymbol{z}_t - h_t(\boldsymbol{x},\xi)\right)^T R_t^{-1}(\boldsymbol{z}_t - h_t(\boldsymbol{x},\xi))} \tag{46}$$

$$P[\xi|\boldsymbol{z}] \propto P[\boldsymbol{z}|\xi]P[\xi] \tag{47}$$

The MAP estimate of the random variables from equation (46) is used in the collocation matrix to return the values of the state-space variables and the parameter values as:

$$A(\xi_{Est})\boldsymbol{x}(t,\xi_{Est}) \tag{48}$$

For systems with quasi-static parameters, this technique is highly recommended for the parameter estimation, and especially for initialization of the parameters for the gPC-EKF. The convergence rate of the gPC-EKF can be non-trivial, and the closer the initial estimates of the parameters, the faster the convergence.

3.4 gPC-EKF Improvements and Application to State-space Systems

When the gPC mathematics are blended with the EKF equations, a paradox in the system is created. The paradox is that, as the information in the parameter distributions becomes larger, the distribution becomes peaked at a value, the width of the distribution decreases. Normally, this would be good, but the covariance is calculated from the width of the distribution. As the distribution becomes more peaked, the ability to update the state and parameter values decreases.

These problems are particularly hard to identify for small initial variations in state or parameter values. There are three proposed improvements to the filter. Each of these has some mathematical basis and some empirical basis for how and why they work. One setup is used in demonstrating the problems and how each of the improvements work. These examples are somewhat contrived to demonstrate each of the problems. For different cases it may be possible to neglect some of the improvements; however, in general, they significantly improve the performance of the filter.

For the sake of clarity, the three different improvements are defined now. The improvements are listed in terms of their application. The first is an improvement that enables state estimation and filtering, and is essentially a process noise matrix. This is what has been found empirically to work. There may be some improvement that can be applied to the system that will operate upon the state distributions themselves, but this is unlikely. Inclusion of an extraneous parameter that can be used to add uncertainty to the state parameters without affecting the dynamics is a possibility, but can cause the same problem as too large of a process noise matrix. This addition is identically the addition of another parameter distribution, but not including it in the actual dynamics. There is also the concern of adding an additional parameter to the system, which would increase computational cost and complexity. The improvement that is proposed is to add a small process noise covariance matrix to the actual state covariance matrix. This matrix can be custom defined, but, at a minimum, it needs to have a positive definite element for the state desired to be tracked. Here it is shown as a diagonal matrix.

$$cov(\boldsymbol{x}_k, \boldsymbol{x}_k) = \begin{bmatrix} \sum_{i=2}^{S} x_{1,k}^i \, x_{1,k}^i & \cdots & \sum_{i=2}^{S} x_{1,k}^i \, x_{n,k}^i \\ \vdots & \ddots & \vdots \\ \sum_{i=2}^{S} x_{n,k}^i \, x_{1,k}^i & \cdots & \sum_{i=2}^{S} x_{n,k}^i \, x_{n,k}^i \end{bmatrix} + c \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \tag{49}$$

The coefficient c is a difficult parameter to tune, it depends on both the time step and the noise in the system, as well as other not well understood dynamics. The parameter enables the state estimation by artificially adding information to the state covariance. If that information is too small, the residual will not be able to propagate through the update laws, and the system will not be able to track the trajectory. If the parameter is too large, the states and the parameter values will decouple, and the parameter values will be able to drift, as the residual will be small regardless of the parameter values.

The second proposed improvement is with regards to the parameter distribution. As previously defined, when the system moves forward in time the filter drives the state and parameter distributions to zero by driving the higher order terms in the power series to zero. The problem is that the information in the covariance matrix is reduced as the information in the higher order power series coefficients is reduced. Normally the gPC-EKF is initialized with a parameter distribution. If at each time step, some amount of information is added to that distribution then the distribution width will never be able to collapse. How this is done is somewhat complicated. The information is added in the same way, but what information to add is highly complicated. The initial conditions for the forward integration as defined as:

$$\boldsymbol{X}_{k-1}^i = \sum_{j=1}^{S} A_{i,j} \, \boldsymbol{x}_{k-1}^j \tag{50}$$

With

$$\boldsymbol{x}_{k-1}^{j} = \boldsymbol{x}_{k-1}^{j} + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ c_1\, I_1^j \\ \vdots \\ c_d\, I_d^j \end{bmatrix} \tag{51}$$

How I_d^j is defined is the complicated part. In gPC-EKF there are three statistical distributions used. The first is the initialization of the system, or the uncertainty distribution associated with the parameter. The second is the underlying basis functions that are used in the gPC mathematics. And the third distribution is what is used for I_d^j. For a system that is using the Legendre basis functions, from empirical calculations, the points chosen to generate the collocation matrix are quite suitable. The parameter c_d is used to tune the system to improve the tracking stability. That parameter suffers from the same constraints that the state improvement parameter does. Too small and there's not enough effect, too large and the system becomes unstable. There are many different distributions that can be used. Random noise sequences are possible, though there are some complicating effects that are induced. The collocation points make good additions, especially for systems with asymmetric distributions, though that particular topic will be covered later. Distributions that are based off of the state trajectory error (so that the parameter additions become small as the tracking error converges), or distributions that are based off of the parameter mean value and distribution are also useful. There are many other different methods that can be employed. There are solid mathematical reasons for why the perturbation is needed, but very little aside from empirical results to show how each will affect the system's time evolution. Some of these distribution additions will be covered in more detail when investigating the effects of having an asymmetric distribution for the gPC basis functions.

The final addition is employed when the initial state values for the real system are unknown. This addition adds a significant amount of information to the system that is quickly removed by the filter. The improvement is made through the addition of white noise to the state values during the first forward integration:

$$\boldsymbol{x}_{t_0}^{j} = \boldsymbol{x}_{t_0}^{j} + \begin{bmatrix} N_1 \\ \vdots \\ N_n \\ c_1\, I_1^j \\ \vdots \\ c_1\, I_d^j \end{bmatrix} \tag{52}$$

Where N_n is a white noise process. It could also be a colored noise process, so long as it imparts enough statistical variations to supply the information needed to the filter.

3.4.1 Demonstrations of the gPC-EKF Applied to a Mass-Spring System

The gPC-EKF filter is applied to a simple harmonically excited spring-mass system to demonstrate the problems, improvements and capabilities of the filter. For this example the basis functions for the gPC basis are Legendre polynomials. Other basis functions could be used, but Legendre polynomials demonstrate the fewest quirks. The state vector, including uncertain parameters, for a spring-mass system is:

$$\boldsymbol{x} = [x_1, v_1; \boldsymbol{p}]^T \tag{53}$$

With the uncertain parameter vector:

$$\boldsymbol{p} = [m] \tag{54}$$

The state-space is defined as:

$$\begin{bmatrix} \dot{x} \\ \dot{v} \\ \dot{m} \end{bmatrix} = \begin{bmatrix} v \\ -\frac{k}{m} * x \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{200\sin(t)}{m} \\ 0 \end{bmatrix} + \boldsymbol{\nu} \tag{55}$$

Where $\boldsymbol{\nu}$ is a process noise vector. The collocation method is used for propagating the uncertain parameters through the state-space. This method defines the power series expansions and parameter values for the iteration as:

$$m_i = \sum_j m^j \psi_1^j(\mu_1^i) \tag{56}$$

At each time step, k, the system is integrated Q times, for index i.

$$\boldsymbol{X}_k^{\boldsymbol{i}} = \left[x_{k,i}, v_{k,i}, m_{k,i}\right]^T, \qquad for\ i = 1 \ldots Q \tag{57}$$

The initial conditions for each step forward are defined as:

$$\boldsymbol{X}_{k-1}^{i} = \sum_{j=1}^{S} A_{i,j}\, x_{k-1}^{j} + \left[0{,}0, c_1\, \mu_1^i\right]^T \tag{58}$$

With the observation matrix:

$$H_k = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{59}$$

The system is demonstrated without the improvements. **Table 3.1** shows all of the different filter parameter values.

Table 3.1 Parameter Values for the Unmodified gPC-EKF

Poly Order	3
Q	6
Time Step (S)	0.01
c	0
c_1	0
x_0	0
v_0	0
m^1	600
m^2	10
N_1	0
N_2	0
R	0.01

The case study demonstrated here is a classical second order mass-spring system. The actual value of the mass is $100\ kg$. To highlight the problems with the filter, 40 seconds into the simulation, the mass of the ‘real system’ increases by 50% to $150kg$. The state trajectory of the filter when no improvements are used is demonstrated in **Figure 3.1**. The filter does not diverge, which is good, but it also doesn’t perform the state or parameter estimation well. The harmonics observed in the velocity come from the high frequency vibrations for the filter system, because it has such a low mass value, but it tracks the velocity profile because the forcing term has the dominant frequency.

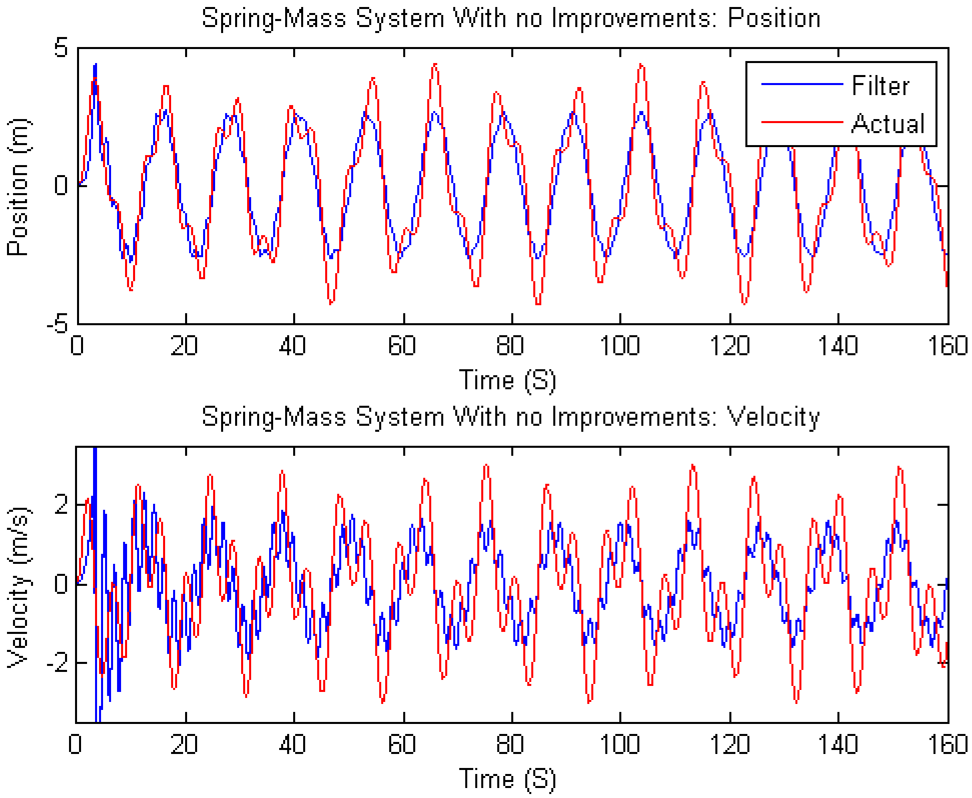


Figure 3.1 Spring-Mass System's State Trajectory from the gPC-EKF without any Improvements

Figure 3.2 shows the mean value of the gPC expansion for the mass parameter, or the m^1 parameter. This figure highlights the problem with the gPC-EKF for parameter estimation. The parameter trajectory starts to move, but before it can converge to the correct value, the distribution is driven to zero and the covariance matrix subsequently goes to zero.

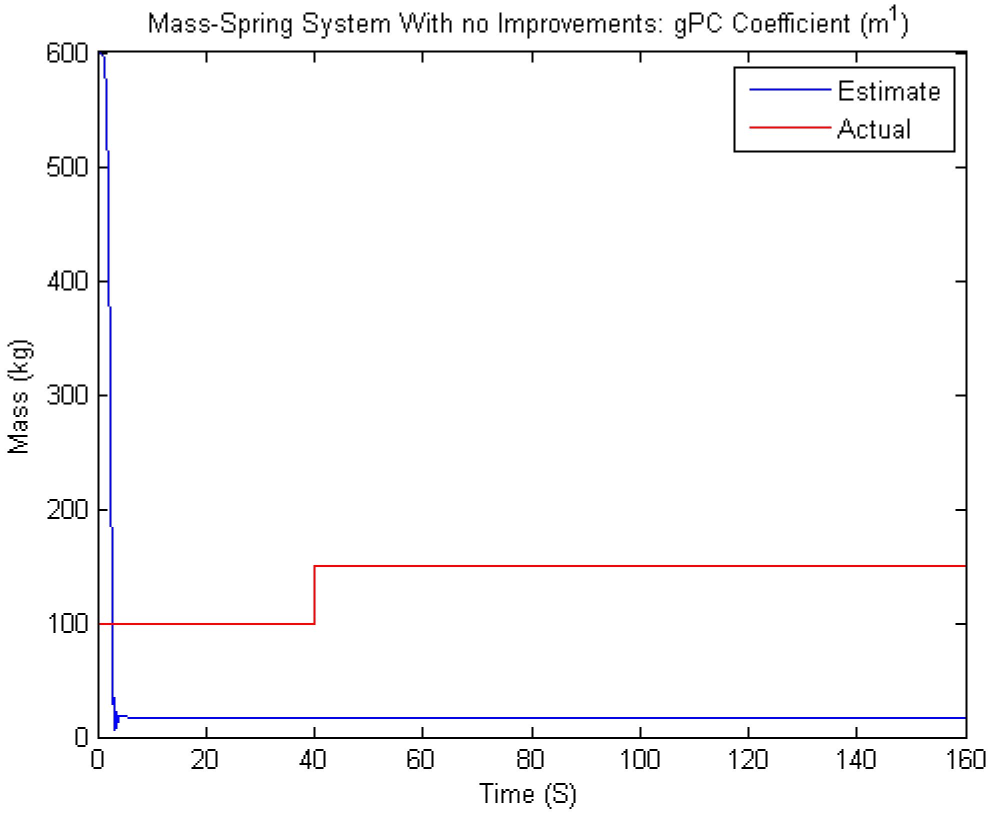


Figure 3.2 Mass Estimate of the gPC-EKF when no Improvements are Used

The covariance is dependent upon the distribution of the system. **Figure 3.3** shows the higher order parameters of the gPC expansion for the mass parameter. It can clearly be seen that the distribution starts out having some information, but it is quickly driven to zero; the effects of the distribution being zero have already been demonstrated.

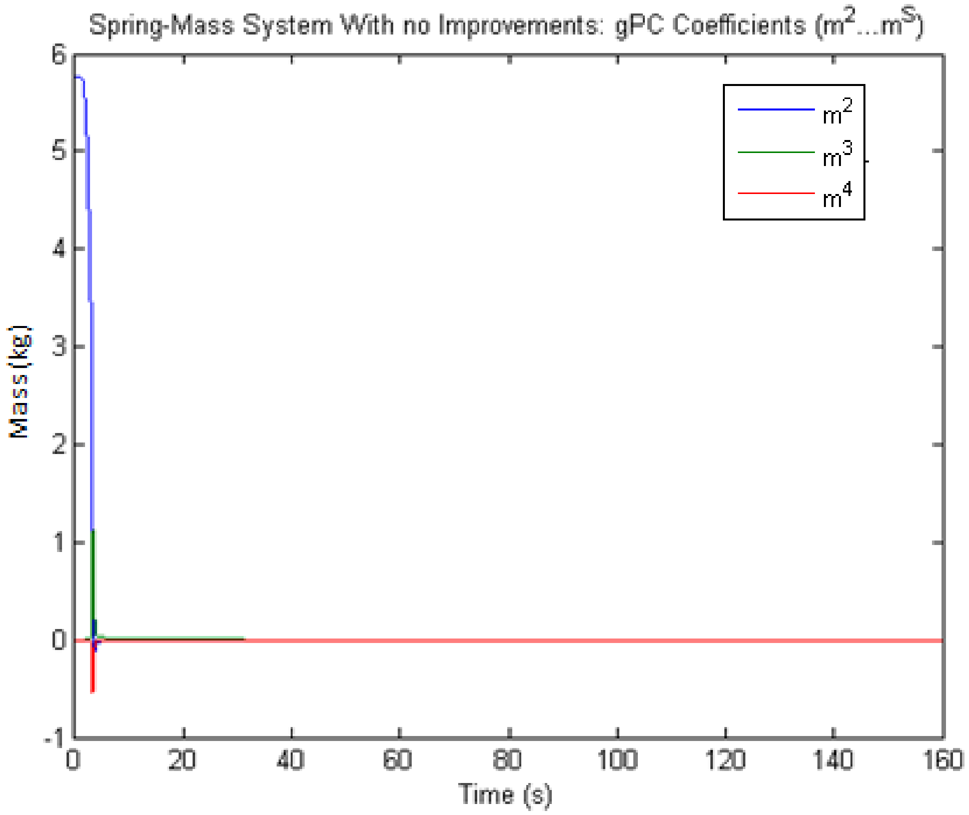


Figure 3.3 Mass Parameter's Distribution Coefficients in the gPC-EKF when no Improvements are Used

Next, the filter is performed with the first of the improvements: The perturbation of the covariance matrix. The filter's coefficients are given in **Table 3.2**.

Table 3.2 Parameter Values for the gPC-EKF with the Covariance Improvements

Poly Order	3
Q	6
Time Step (S)	0.01
c	0.0001
c_1	0
x_0	0
v_0	0
m^1	600
m^2	10
N_1	0
N_2	0
R	0.01

The filter starts off with a good estimate of the state values, and the trajectories track well; however, once the mass increases, the tracking performance degrades. If the amount added to the covariance were larger, the tracking would be better. This is a good example of how large the addition to the covariance should be; it should be large enough for general tracking, but small enough to enable the residual to propagate to the parameter update.

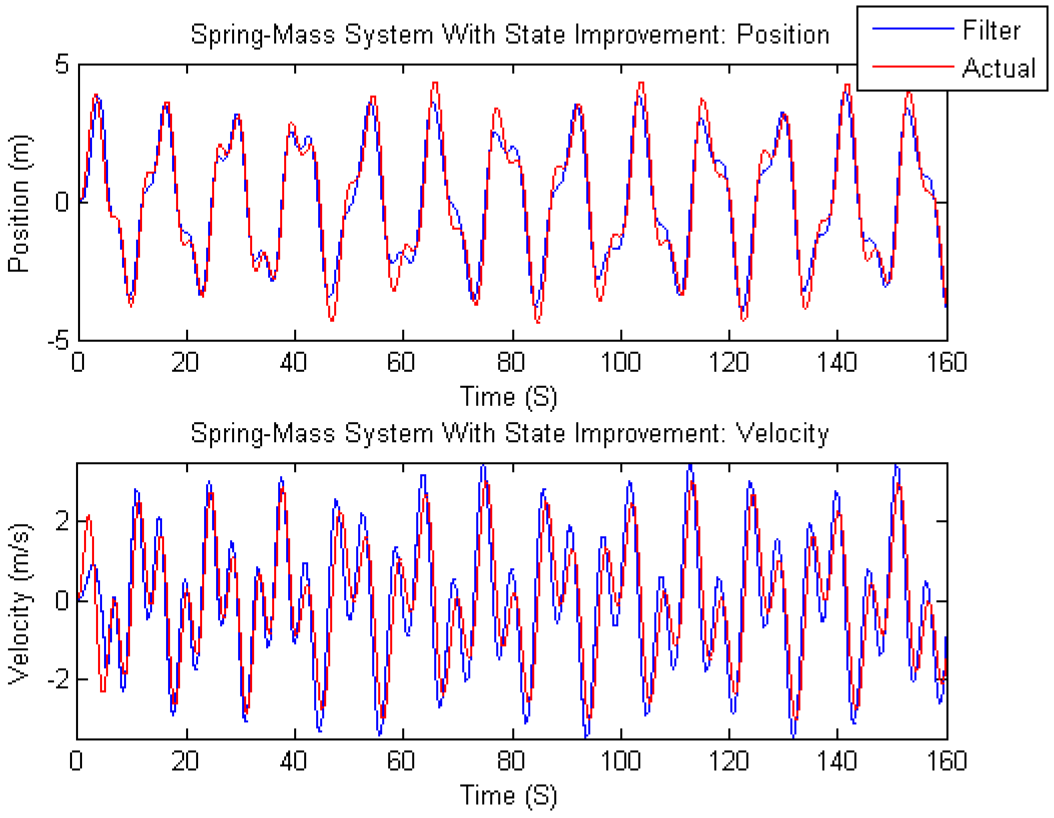


Figure 3.4 Spring-Mass State Response with State Tracking Improvement

The additions to the covariance matrix are also added to the parameter covariance, which prevents the system from driving the parameter power series coefficients to zero. Because of this, a small amount of parameter updating is enabled, and, as it can be seen in **Figure 3.5**, the mass parameter is slowly able to approach the actual value, because of the small element added to its covariance by the identity matrix style process noise covariance matrix addition.

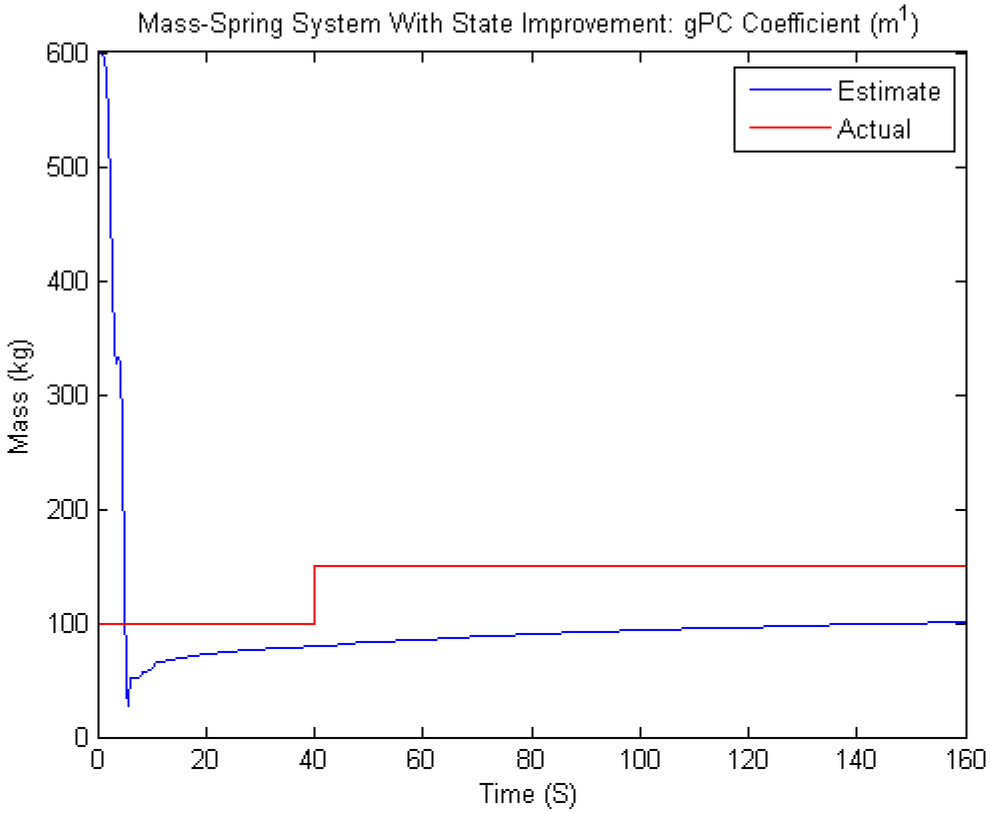


Figure 3.5 Mass Parameter Estimation with State Improvement

The distribution for the parameter is once again quickly driven towards zero, shown in **Figure 3.6**, but the added information in the covariance matrix prevents it from completely converging to zero. Though it is unclear if it will eventually converge to zero, the trend in the data indicates that it could.

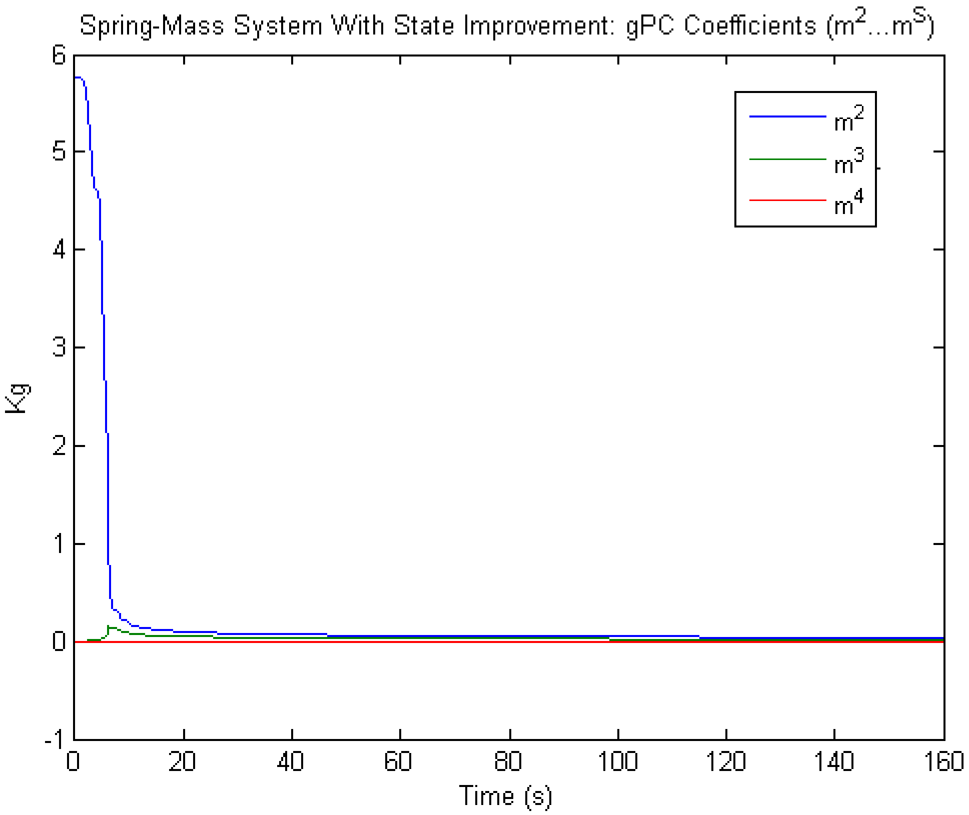


Figure 3.6 Mass Distribution Higher Order Terms with State Tracking Improvement

It can thus be seen why the next step is a logical one of adding information to the parameter distributions. The next set of examples shows the improvement that is gained by adding information to the parameter distributions. The parameter values for the unmodified gPC-EKF used in these examples are included in **Table 3.3**.

Table 3.3 Parameter Values for the Unmodified gPC-EKF

Poly Order	3
Q	6
Time Step (S)	0.01
c	0.005
c_1	0
x_0	0
v_0	0
m^1	600
m^2	10
N_1	0
N_2	0
R	0.01

The state trajectories quickly track the actual values, as it can be seen in **Figure 3.7**. The additions to the parameter distributions enable fast convergence of the parameter values. There is an important truth with this filter: If the parameters have converged, the non-tracked states will also converge.

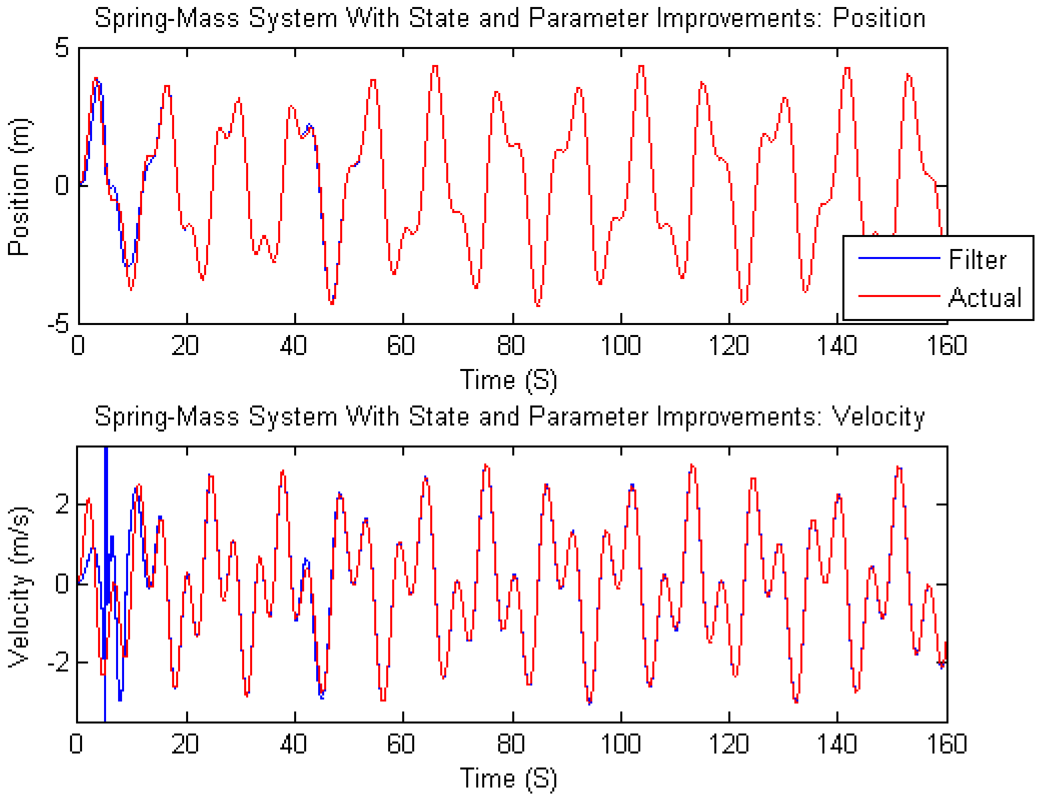


Figure 3.7 State Response with State Tracking and Parameter Distribution Improvements

The mass estimation is now able to track the actual mass of the system. The overshoot that happens is because the filter is not able to reduce the information in the covariance matrix fast enough. If that initial parameter distribution is removed, and only the additions to the parameter distribution at each time step are permitted, the parameter convergence can be much smoother. Also, if it takes a long time for the parameter to converge, the build-up of the added parameter distribution can cause an overshoot.

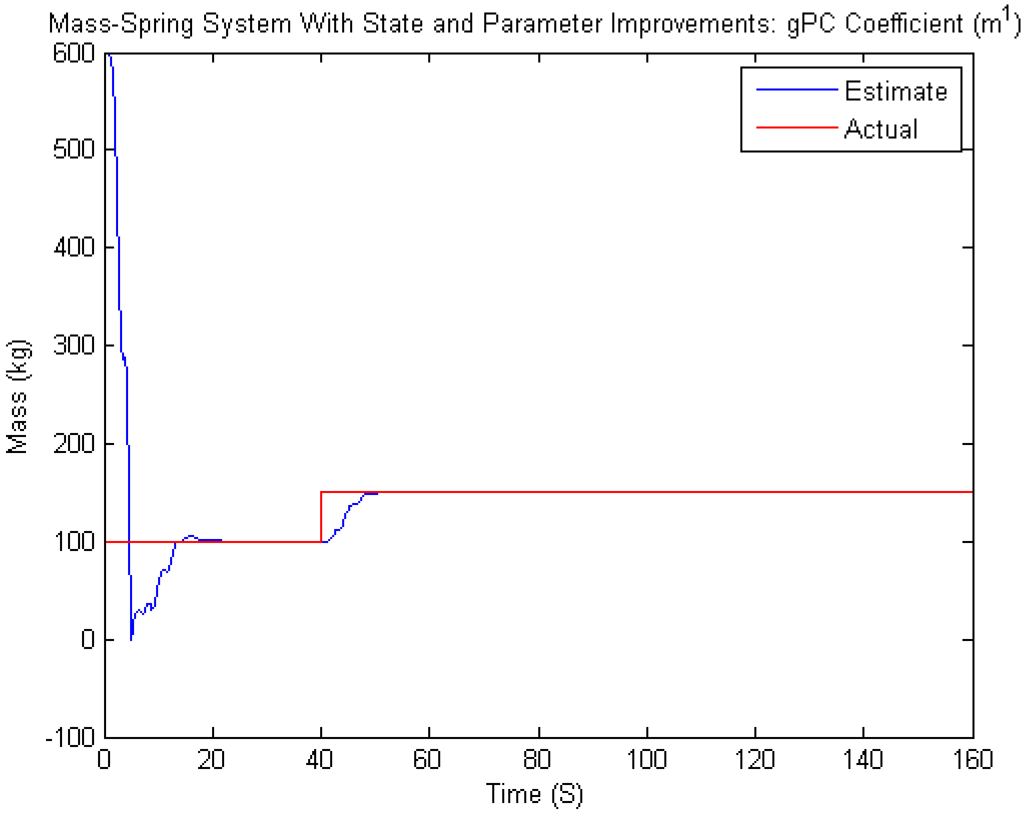


Figure 3.8 Mass Estimation with Both State and Parameter Improvements

The additions to the parameter distributions are clearly seen in **Figure 3.8**. They keep the second term in the distribution from converging to zero. If the additions are not linear additions, the higher-order terms would show non-zero values, as illustrated in **Figure 3.9**.

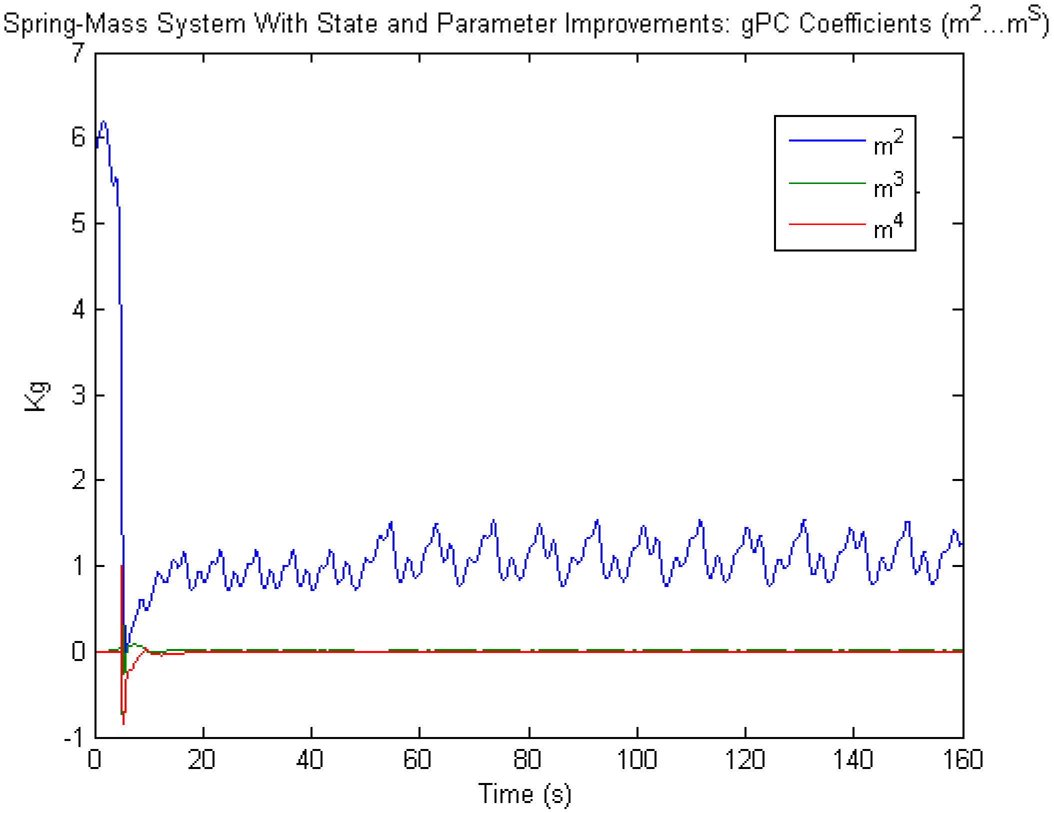


Figure 3.9 Higher Order Terms of the Mass Parameter Series for the System with Both State and Parameter Improvements

These two improvements are the cornerstone of the filter. Without them, the filter does not function very well. There is another case that must be considered, but is a subset of these other two improvements. When the filter is initialized, if the states of the actual system are not known well, information regarding that uncertainty must be added to the system to enable proper performance. This is demonstrated for the system with a colored noise signal, though it also works with the state of the actual system being different than the filter, and the filter having a white noise process. The parameter values used in this case study are provided in **Table 3.4**.

Table 3.4 Parameter Values for the Modified gPC-EKF with Initial Condition Noise

Poly Order	3
Q	6
Time Step (S)	0.01
c	0.005
c_1	0
x_0	0
v_0	0
m^1	600
m^2	0
N_1	5+GaussianWhiteNoise
N_2	5+GaussianWhiteNoise
R	0.01

The system is initialized with the colored noise signal. Also, the initial distribution of the parameter is set to zero, and the parameter trajectory does not vary nearly as much as the previous case.

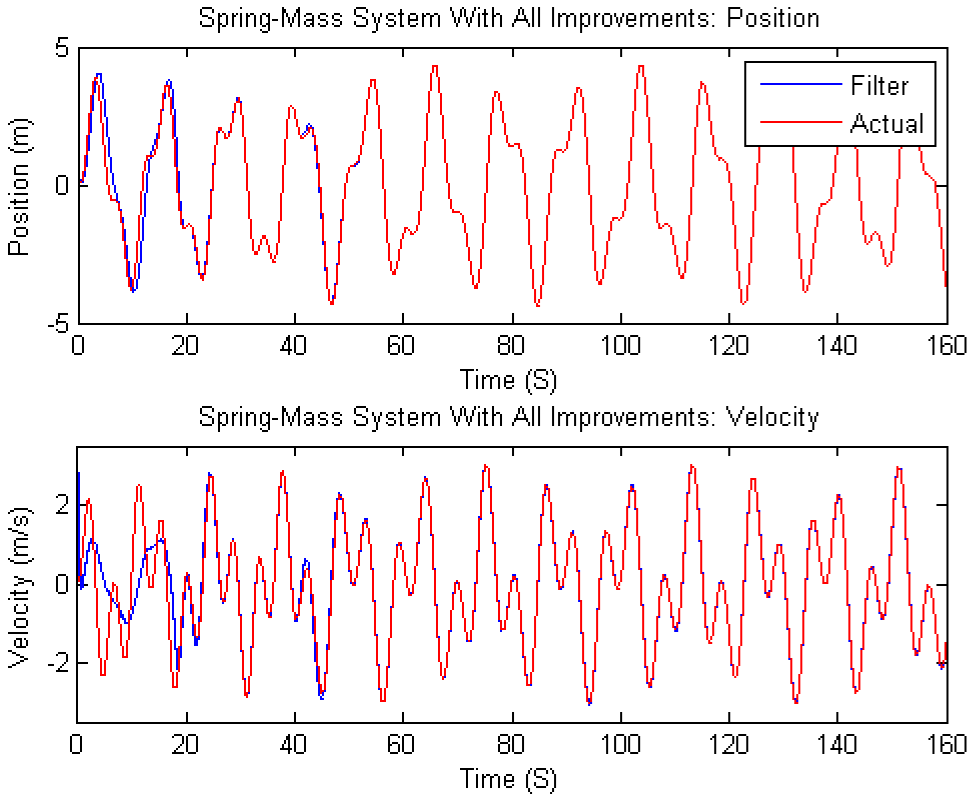


Figure 3.10 State Response of the Filter with all Improvements

It can be seen from **Figure 3.10** that, once the initial state errors have converged, the system proceeds like before. This simulation demonstrates both the effects of the initial state errors, as well as what happens when the initial parameter distribution is set to zero. The parameter estimation does not overshoot like it did before. Later on, the effects of different additions to the parameter distributions will be investigated further, but for now it can be seen that the information in the parameter distribution has a significant effect on the ability of the filter to estimate the parameter.

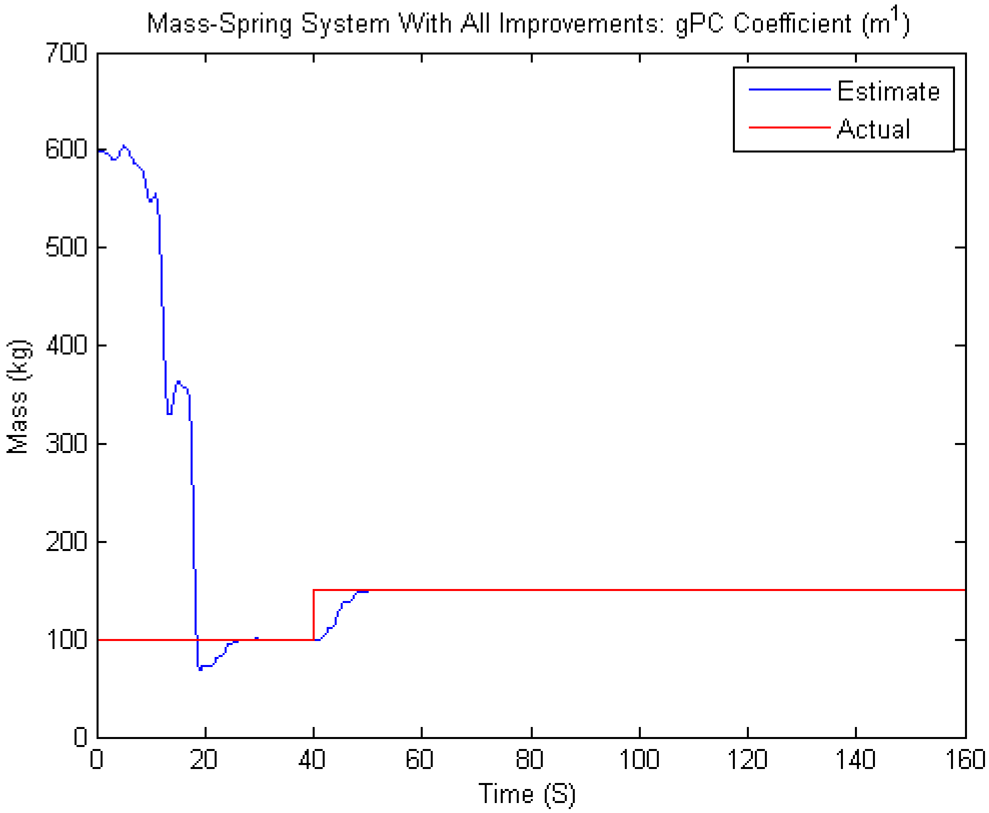


Figure 3.11 Mass Estimation of the Filter with all Improvements

The parameter distribution proceeds very similarly to the previous case, as shown in **Figure 3.11**. However, a very important effect is demonstrated here. Blanchard [25] noted that, if the information in the covariance matrix was not significant enough, the filter did not appear to converge. The parameter distribution plot gives a lot of credit towards that concept, as the parameter estimate does not move very much or converge until the information increases to a certain threshold. The higher order terms of the mass parameter expansion are shown in **Figure 3.12**. Since the information is being added at every time step, it slowly grows until it passes that threshold and the parameter value quickly moves.

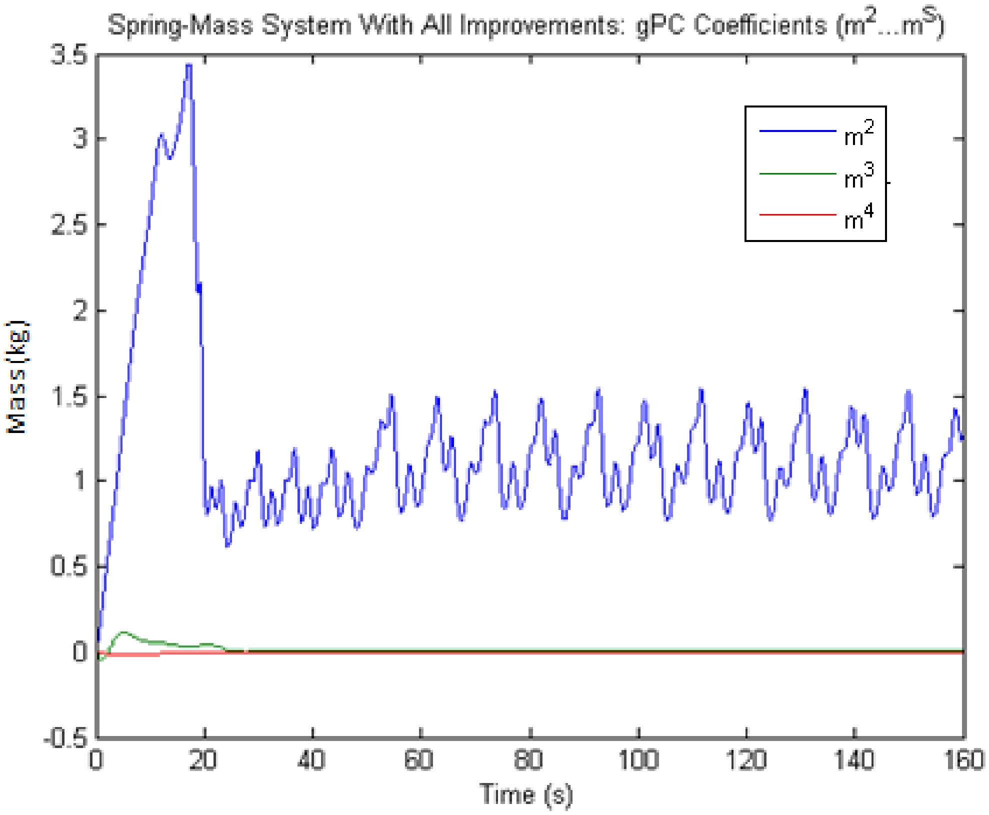


Figure 3.12 Higher Order Terms of the Mass Series for the Filter with all Improvements

The final case to be demonstrated for the filter is when there is noise in the measurements. If the filter cannot handle noise, then there is really no point to pursuing its development. The system has a Gaussian white noise signal added to the measurements, as shown in the **Table 3.5** of parameter values.

Table 3.5 Parameter Values for the Modified gPC-EKF with Initial Conditions Noise and Process Noise Matrix

Poly Order	3
Q	6
Time Step (S)	0.01
c	0.005
c_1	0
x_0	0
v_0	0
m^1	600
m^2	0
N_1	GaussianWhiteNoise
N_2	GaussianWhiteNoise
R	0.1

The filter is initialized with all of the improvements. The states track well, and the noise is properly filtered out, except for a small amount of error in the velocity, as presented in **Figure 3.13**.

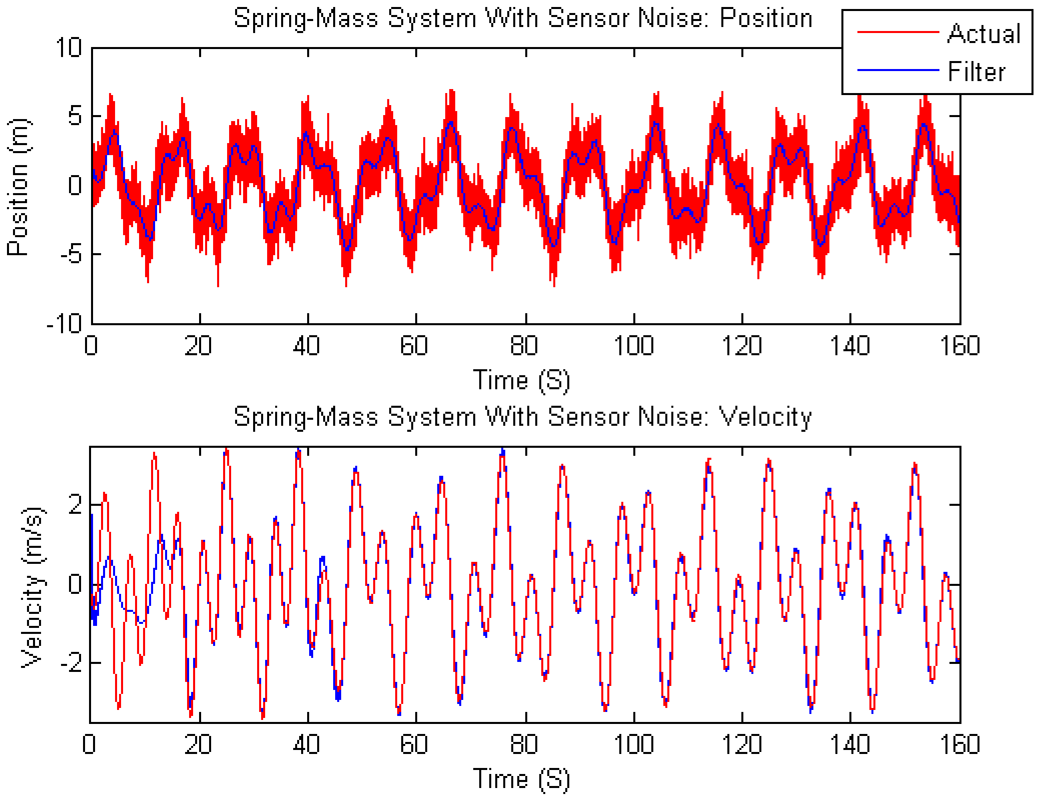


Figure 3.13 State Response for the Filter Tracking a Noisy Signal

The biggest problem that occurs is found in the parameter estimation. Because the parameter distribution always has some information in it, the errors induced because of the sensor noise filtering causes the parameter value to fluctuate, as it can be seen in **Figure 3.14**. Using accurate knowledge of the filters convergence state leads to an ability to stabilize the parameter values in the presence of sensor noise, which will be demonstrated later.

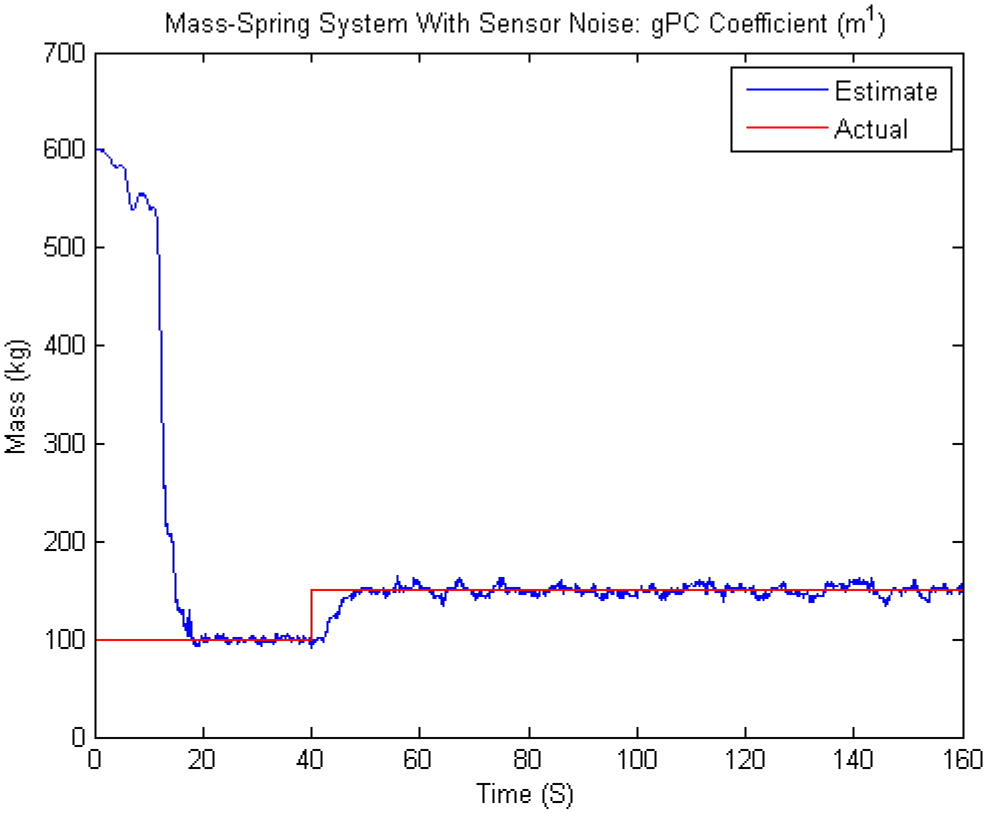


Figure 3.14 Mass Estimation of the Filter in the Presence of Sensor Noise

The additions to the parameter distribution are independent of the sensor noise, and thus, the higher order parameter's time evolutions are unaffected, which is shown in **Figure 3.15**.

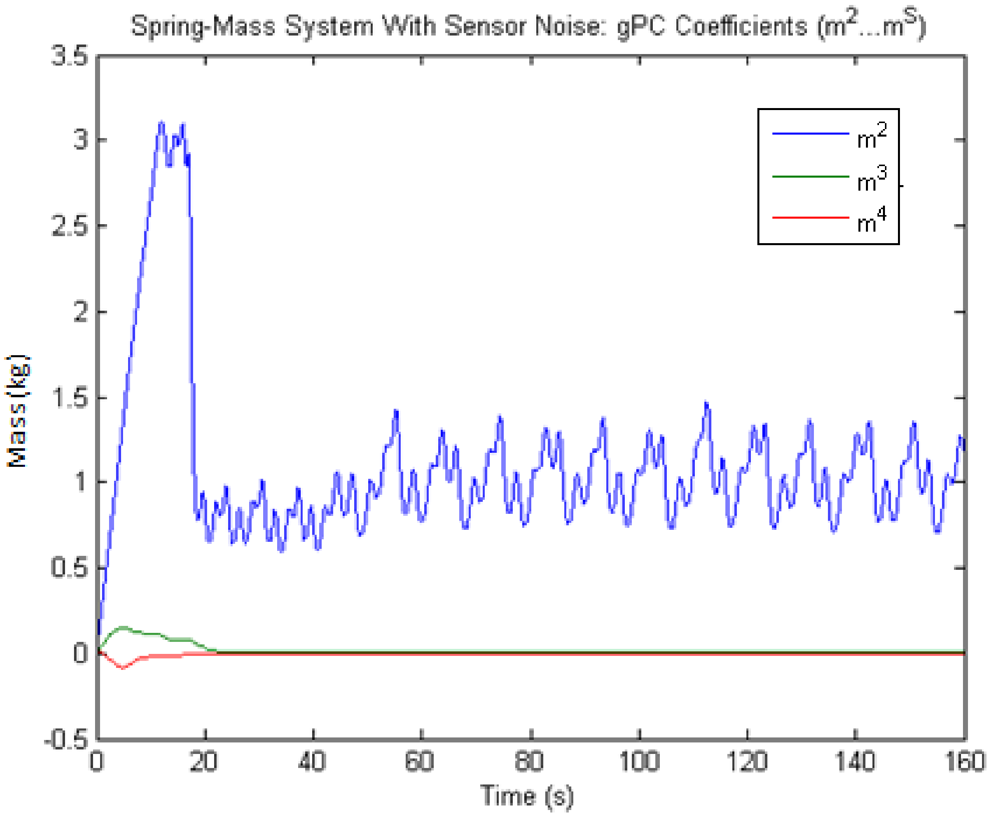


Figure 3.15 Higher Order Terms of the Mass Series for the System with Sensor Noise

As a note, the basis of attraction for this filter is highly non-linear. It depends on the initial conditions of the filter states, the initialized parameter distributions, the parameter update distributions, the process noise matrix additions, and the sensor noise matrix tuning.

3.4.2 Investigation of Asymmetric Basis Functions and Distributions

As mentioned previously, there are three separate distributions that make up the gPC-EKF. The first is the underlying basis functions of the gPC mathematics. The second is the initialization distribution of the parameter. The third is the distribution that is used to update the parameter distribution in the filter. The original method was called Polynomial Chaos, and was created with only one type of underlying distribution [56]. Later on, it was expanded to utilize any orthogonal or orthonormal distribution, which is why it is called gPC. Most of the work performed with this filter has utilized only the Legendre polynomials for the gPC basis functions. Now, however, we consider other types of functions, specific for each distribution. This section shows some of the effects of using an asymmetric Beta distribution via the Jacobi basis functions. If the distribution is a Beta(1,1) then the results are identical to the Legendre functions.

Using the same Mass-Spring system as in the previous section, the filter is simulated without the update laws or the improvements being active. The gPC method was developed to propagate parameter distributions through dynamical systems. This example is meant to show how the distributions evolve. The initialization of the mass parameter is a Beta(1,5) distribution. The beginning and ending values match each other, as illustrated in **Figure 3.16**, as they should, because no update laws are active. The distribution will approximate the true Beta(1,5) distribution as the poly chaos method increases in iterations to statistically saturate the distribution.

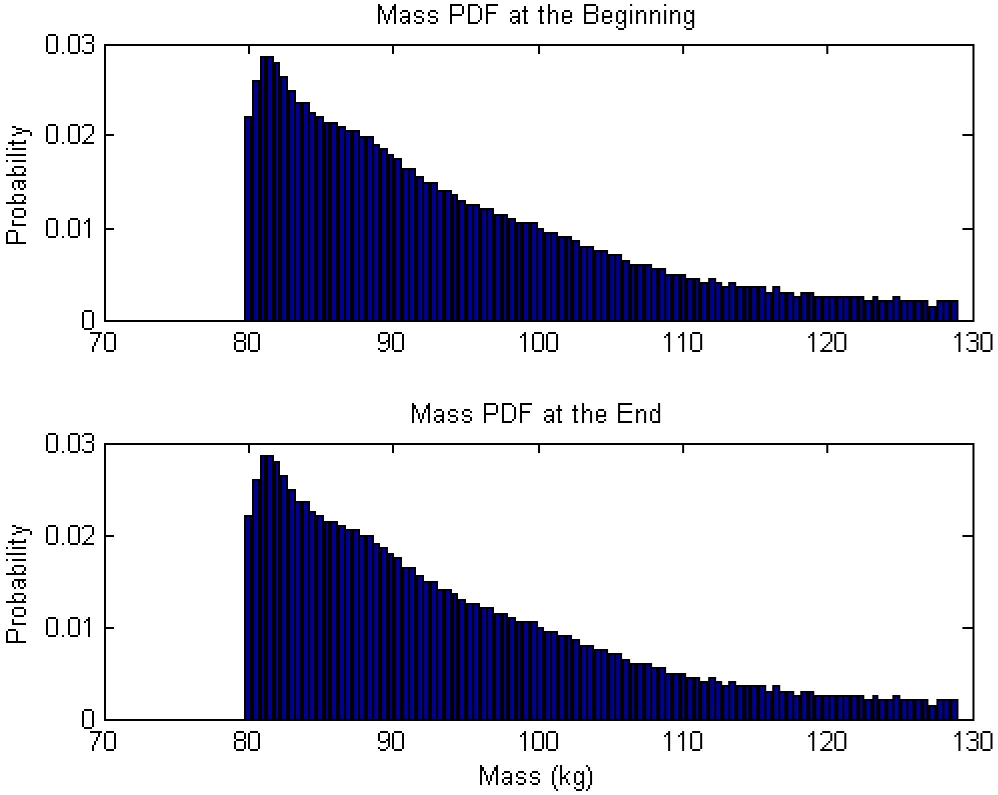


Figure 3.16 Mass PDF of the Beta(1,5) Distribution

Three different simulations are now performed, using the parameters provided in **Table 3.6**. These demonstrate different time evolutions of the system that happen because of the different distributions that are added to the parameter values. The system excitation is a band limited white noise signal. The proposed parameter update for the mass-spring system is:

$$\boldsymbol{x}_{k-1}^{j} = \boldsymbol{x}_{k-1}^{j} + \begin{bmatrix} 0 \\ 0 \\ U^{j} \end{bmatrix} \tag{60}$$

Table 3.6 gPC-EKF Parameters for Asymmetric Basis Function Filter Tests

	Beta(1,5)	Collocation Matrix	Hammersley
Poly Order	7	7	7
Q	6	6	6
Time Step (TS)	0.01	0.01	0.01
U^j	$0.01(-1+2Beta(1{,}5))$	$0.001\ A_{i,2}$	$0.01\ \mu_1^i$
c	0.0001	0.0001	0.0001
x_0	0	0	0
v_0	0	0	0
m^1	160	160	160
m^2	20	20	20
N_1	0	0	0
N_2	0	0	0
R	0.01	0.01	0.01

The system is initialized with the previous Beta(1,5) mass distribution. If the addition that is added is of the same distribution type, the resulting PDF for the mass estimation at the end of the simulation is the same type as the initial distribution, as the results in **Figure 3.17** show.

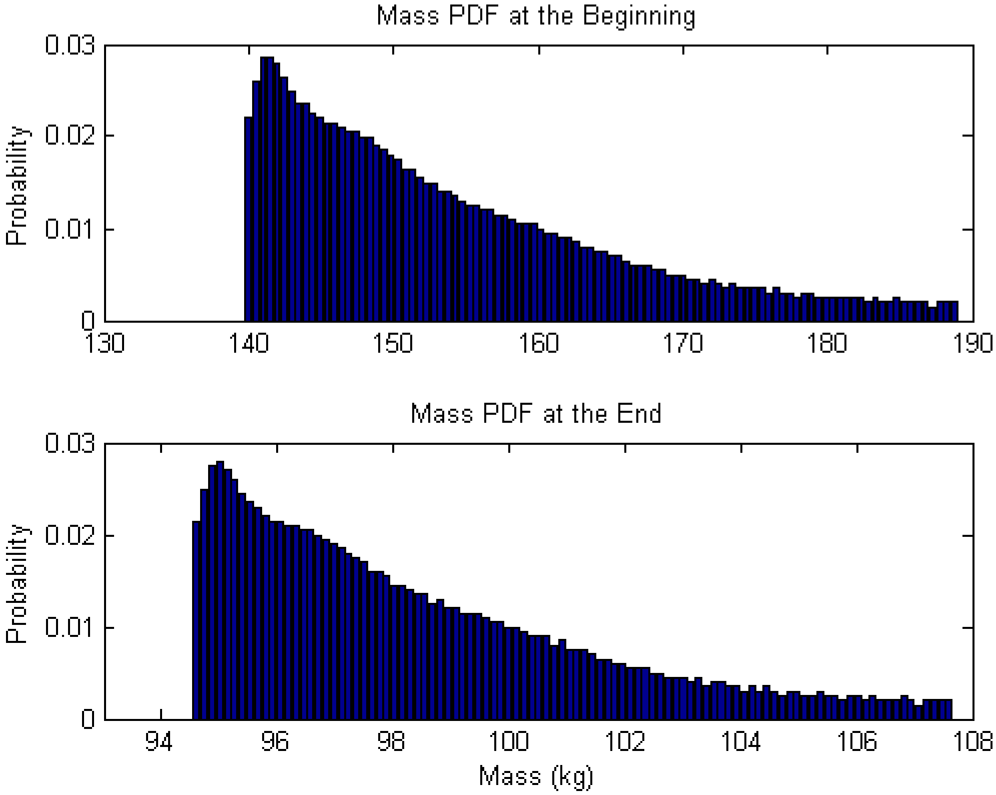


Figure 3.17 Mass PDF Using the Beta Distribution for the Initialization and Update

The next proposed method is employing the Hammersley points used to create the collocation matrix. The Hammersley algorithm is used to produce the points for the collocation technique to properly span the space of the inner product integrals. If those same points are used as the addition to the parameter distributions at each step, a linear distribution is added to the parameter values, which results in a linear parameter PDF, as presented in **Figure 3.18**.

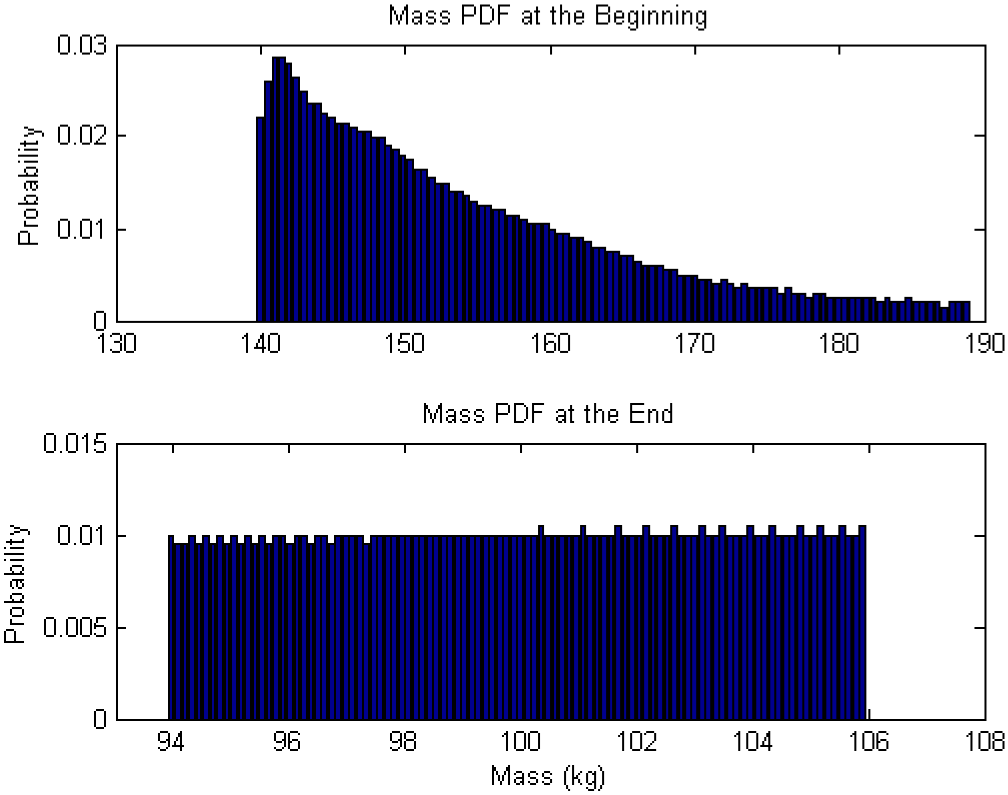


Figure 3.18 Mass Distribution Using the Beta Distribution as the Underlying Distribution and the Hammersley Points as the Update

The final distribution that is demonstrated here is the use of points from the collocation matrix. These points are based off of the Hammersley points, but are transformed through the underlying basis function. These points are therefore spaced in a manner that is congruent with the underlying basis functions and results in the least amount of error in the system. The final distribution is the most compact of the cases. The most important thing to note from **Figure 3.19** is that the distribution is not centered on the actual parameter value, but it is shifted to its side. This actually results in the most accurate mean value from the gPC-EKF, because the underlying distribution is shifted by the Beta(1,5) distribution.

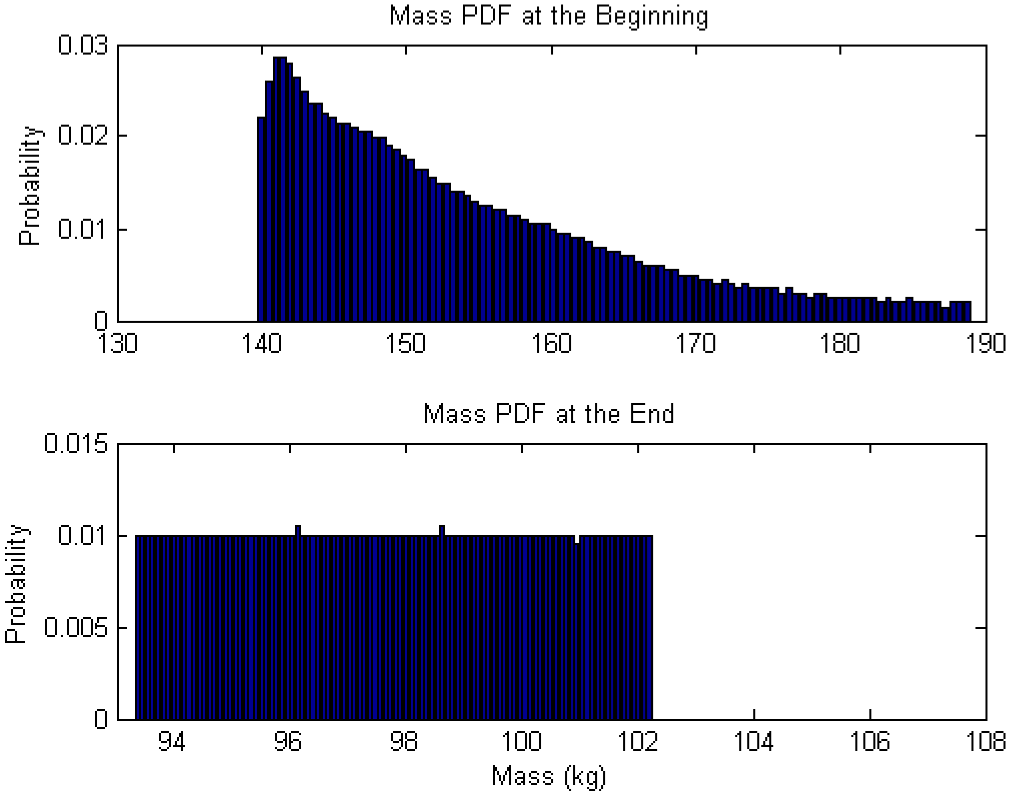


Figure 3.19 Mass Distribution Using the Beta Distribution as the Underlying and the Collocation Matrix Elements as the Update

The filter is aligning the mean value of the series expansion, and not of the PDF function. Because the collocation points are shifted in the same manner as the underlying basis functions, this results in the gPC-EKF's mean value lining up with the actual value. The other two distributions are attempting to line the PDF's mean value up to the actual value, which results in a shift in the gPC-EKF's mean value from the actual value due to the constant input from the applied distribution.

Looking at **Figure 3.20**, it is very difficult to tell which additional distribution has the fastest convergence rate. As the initial estimates degrade the use of the collocation matrix points prove more stable than the other two distributions, but this is purely empirical, and based on the magnitude of uncertainties that could be used before the filter would become unstable.

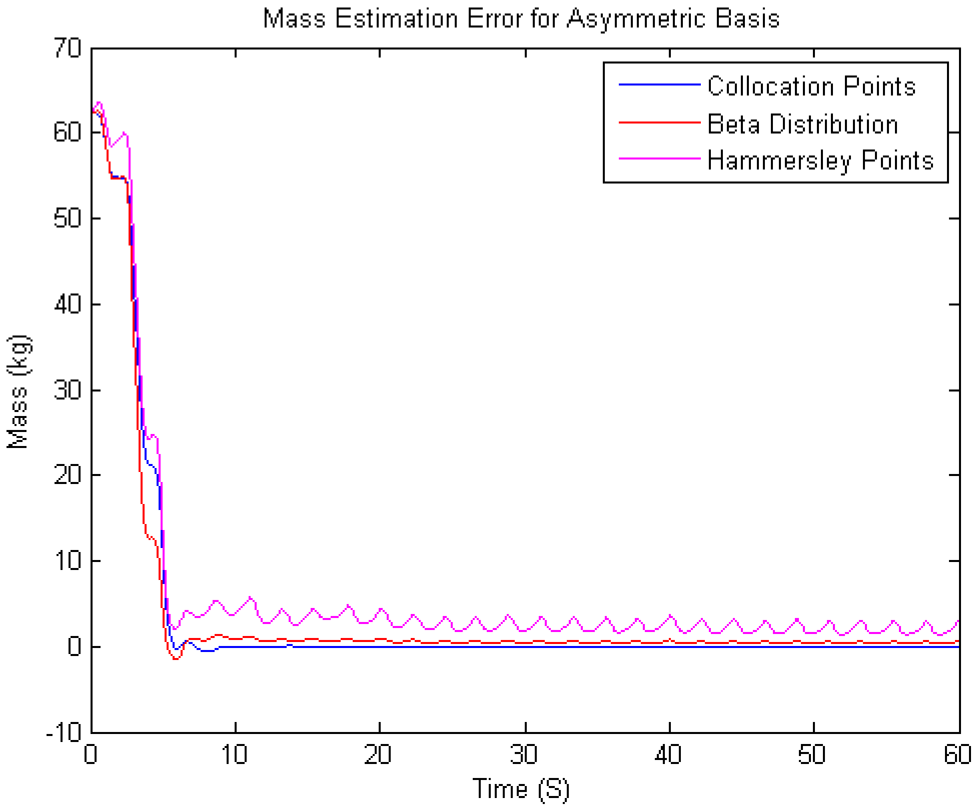


Figure 3.20 Mass Estimation Convergence for Various Update Distributions Using the Beta(1,5) Distribution as the Underlying Distribution

As the parameter update distributions change, the tracking performance is affected, because the parameters used in the forward integrations are not the same. The use of the linear distribution shows the worst tracking performance, as presented in **Figure 3.21**. The error appears to be getting worse over time, but the actual signal that its tracking is going into a limit cycle and the error is just scaling with that signal. The beta distribution shows the next worst performance. It has some similarities to the underlying basis functions, but it doesn't completely match it. Finally, the use of the collocation points shows the best matching of both the parameter values and of the tracking performance.

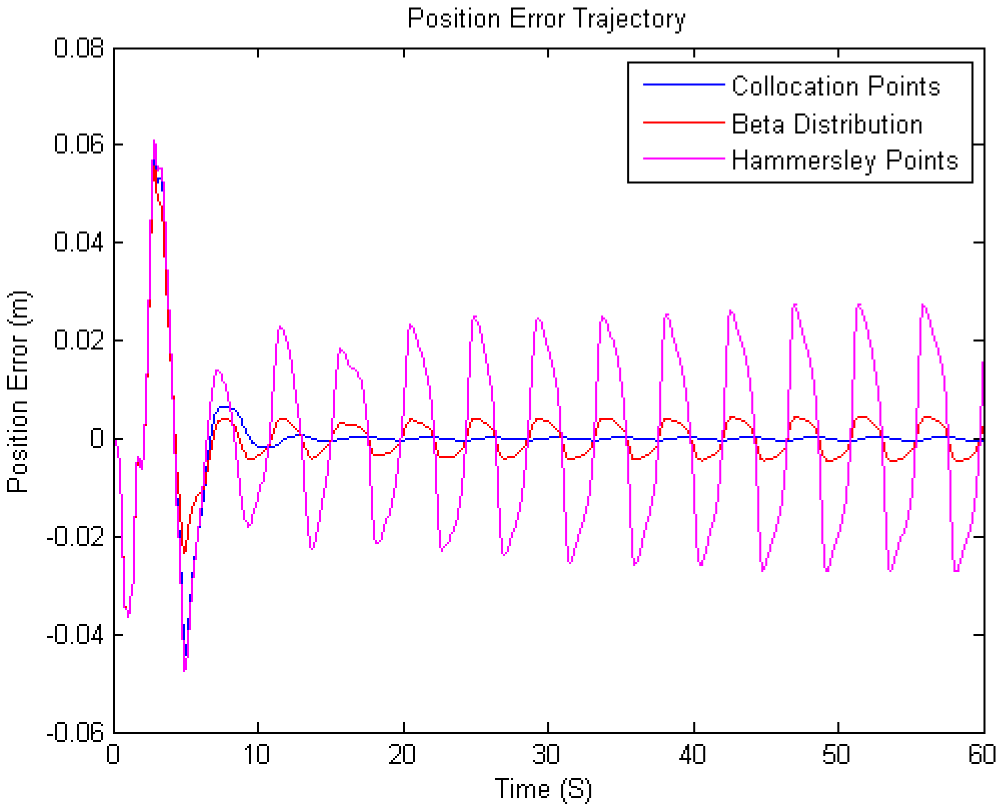


Figure 3.21 Tracking Error for Various Update Distributions Using the Beta(1,5) Underlying Distribution

These three different distributions that are used are not a complete list. Any distribution that one could think of may be of use. Some can cause the filter to become unstable, and some may cause the filter to become completely stable. It is, however, very difficult to determine how the additional points will affect the time evolution of the system. Use of more than just the $A_{i,2}$ points, and expanding to using the full set of values in the row shows positive results, but it is unclear how to scale each column of the A matrix.

When the underlying basis functions are the Legendre polynomials, similar effects to the Jacobi basis functions are observed. The points chosen from the collocation matrix are still the ones with the lowest tracking error, but the Hammersley points (which are uniform in distribution) show the next lowest error. The asymmetric distribution has the largest tracking error.

3.4.3 Spring-Mass-Damper and Sufficient Excitation

The state vector, including uncertain parameters, for a spring-mass-damper system is:

$$\boldsymbol{x} = [x, v; \boldsymbol{p}]^T \tag{61}$$

With the uncertain parameter vector:

$$\boldsymbol{p} = [m, k, c] \tag{62}$$

The state-space is defined as:

$$\begin{bmatrix} \dot{x} \\ \dot{v} \\ \dot{m} \\ \dot{k} \\ \dot{c} \end{bmatrix} = \begin{bmatrix} v \\ -\frac{k}{m} * x - \frac{c}{m} * x \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \sin(t) * \frac{200}{m} \\ 0 \\ 0 \\ 0 \end{bmatrix} + \boldsymbol{\nu} \tag{63}$$

Where $\boldsymbol{\nu}$ is again the process noise vector. The collocation method is used for propagating the uncertain parameters through the state-space. This method defines the power series expansions and parameter values for the iteration as:

$$m_i = \sum_j m^j \psi_1^j(\mu_1^i) \tag{64}$$

$$k_i = \sum_j k^j \psi_1^j(\mu_2^i)$$

$$c = \sum_j c^j \psi_1^j(\mu_3^i)$$

At each time step, k, the system is integrated Q times, for index i.

$$\boldsymbol{X}_k^{\boldsymbol{i}} = \left[x_{k,i}, v_{k,i}, m_{k,i}, k_{k,i}, c_{k,i}\right]^T, \qquad for\ i = 1 \ldots Q \tag{65}$$

The initial conditions for each step forward are defined as:

$$\boldsymbol{X}_{k-1}^{i} = \sum_{j=1}^{S} A_{i,j}\, x_{k-1}^{j} + \left[0{,}0, c_1\, \mu_1^i, c_2\, \mu_2^i, c_3\, \mu_3^i\right]^T \tag{66}$$

With the observation matrix:

$$H_k = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{67}$$

3.4.4 Spring-Mass-Damper Parameter Estimation Results

A spring-mass-damper system is used in conjunction with the gPC-EKF to estimate all three parameters of the system (mass, stiffness, damping). This system is a good example of the capabilities of the parameter estimation method. The system is excited by a sine wave, and the system response is shown in **Figure 3.22**.

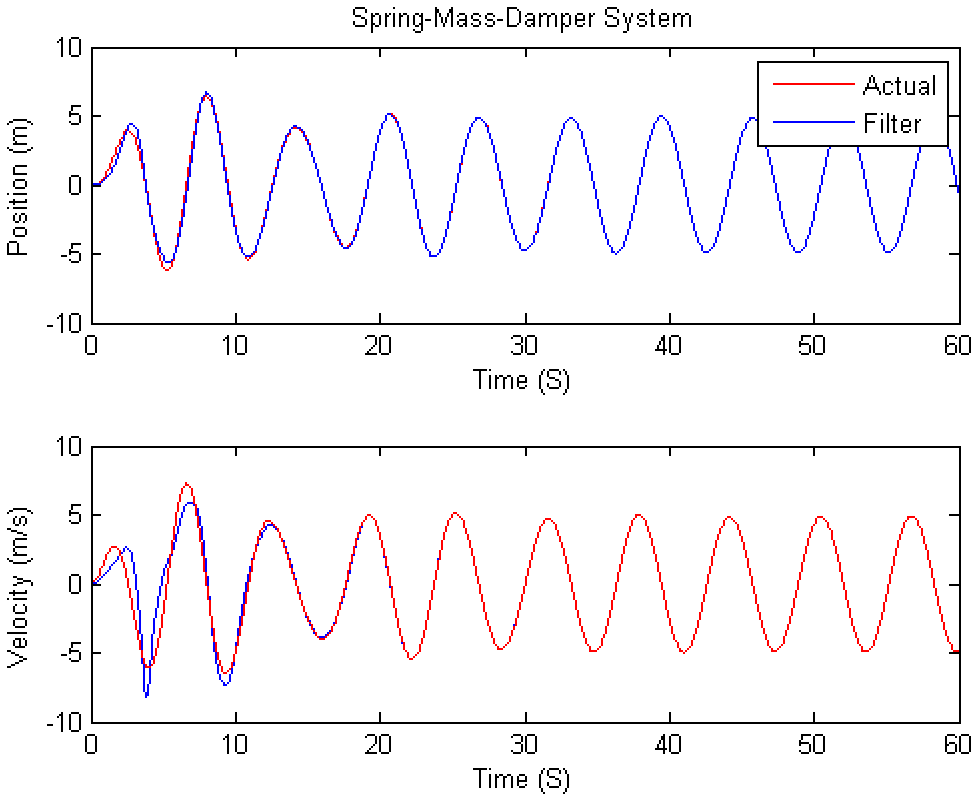


Figure 3.22 State Estimation for the Spring-Mass-Damper System with Poor Parameter Convergence

Figure 3.22 clearly shows that the estimator is able to converge to the correct state estimations. Even with incorrect initial conditions, the method converges and tracks the states well. However, as is seen in **Figure 3.23**, the parameter values do not converge to the correct values.

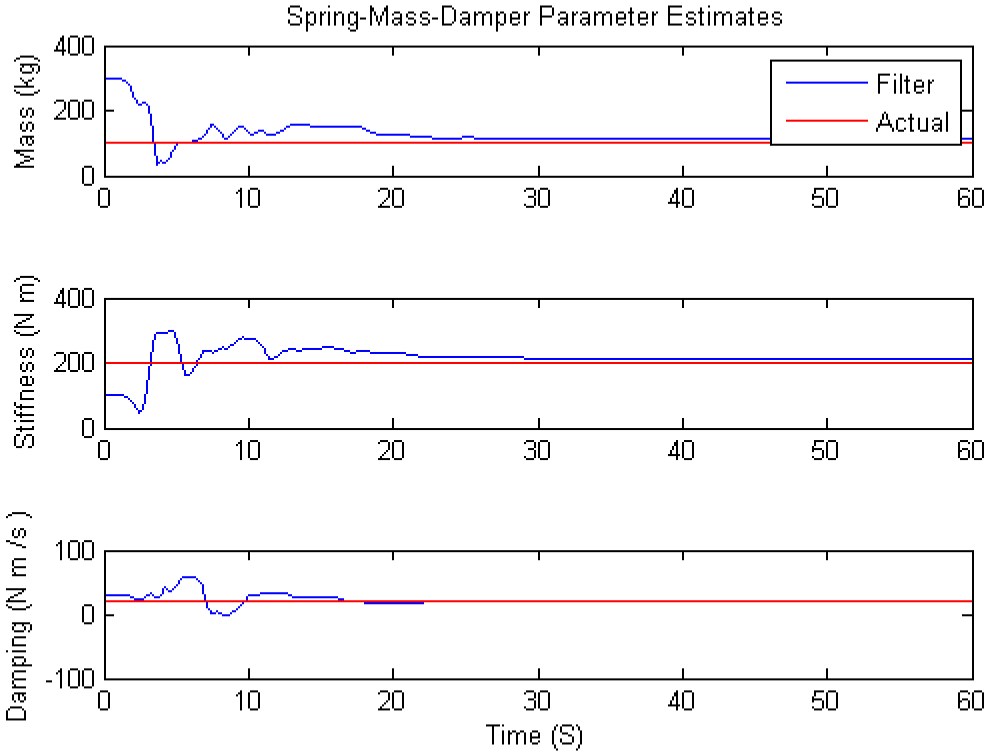


Figure 3.23 Parameter Estimation for the Spring-Mass-Damper System with Sine Wave Excitation

This problem is analogous to the persistent excitation requirement of adaptive control systems. The parameters find a suitable place that minimizes the system's error and plateau off at those values. The advantage of using the gPC-EKF is that it can be seen when the input signal is not sufficient, as illustrated in **Figure 3.24**. The parameter distributions expand because the filter doesn't have enough information to control the parameter distribution.

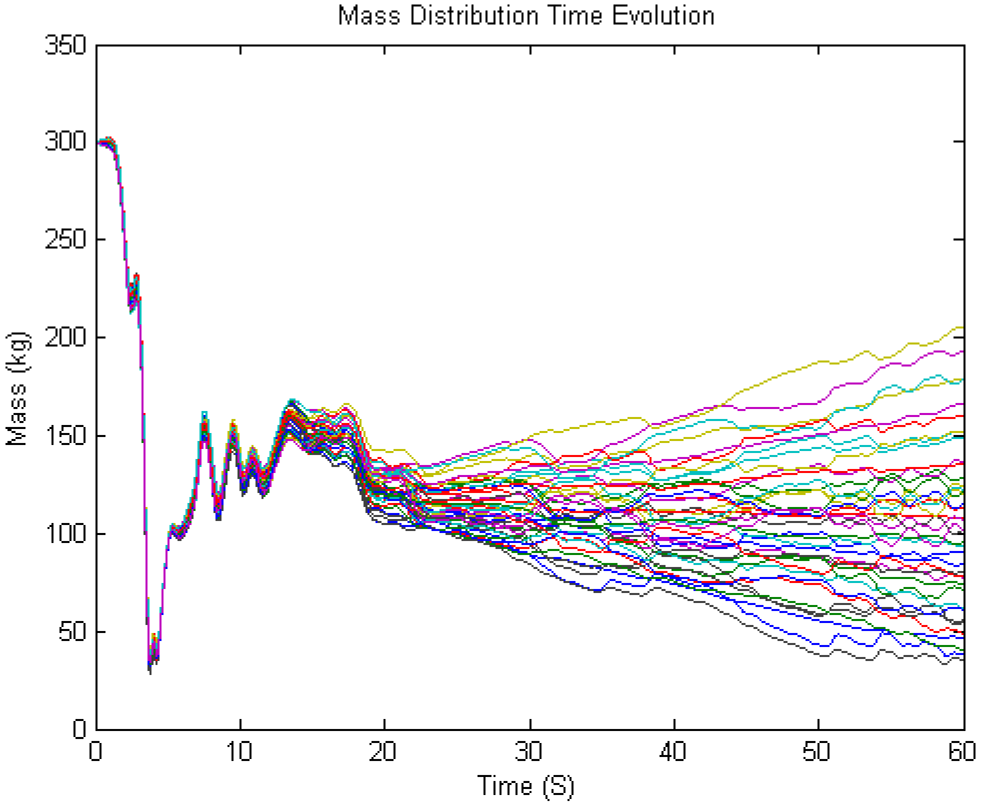


Figure 3.24 Mass Distribution Time Evolution for the Spring-Mass-Damper System without Sufficient Signal Richness

If however the excitation changes from a sin input to a band limited white noise signal the parameter estimations converge to the correct values. The state tracking can be seen in **Figure 3.25**.

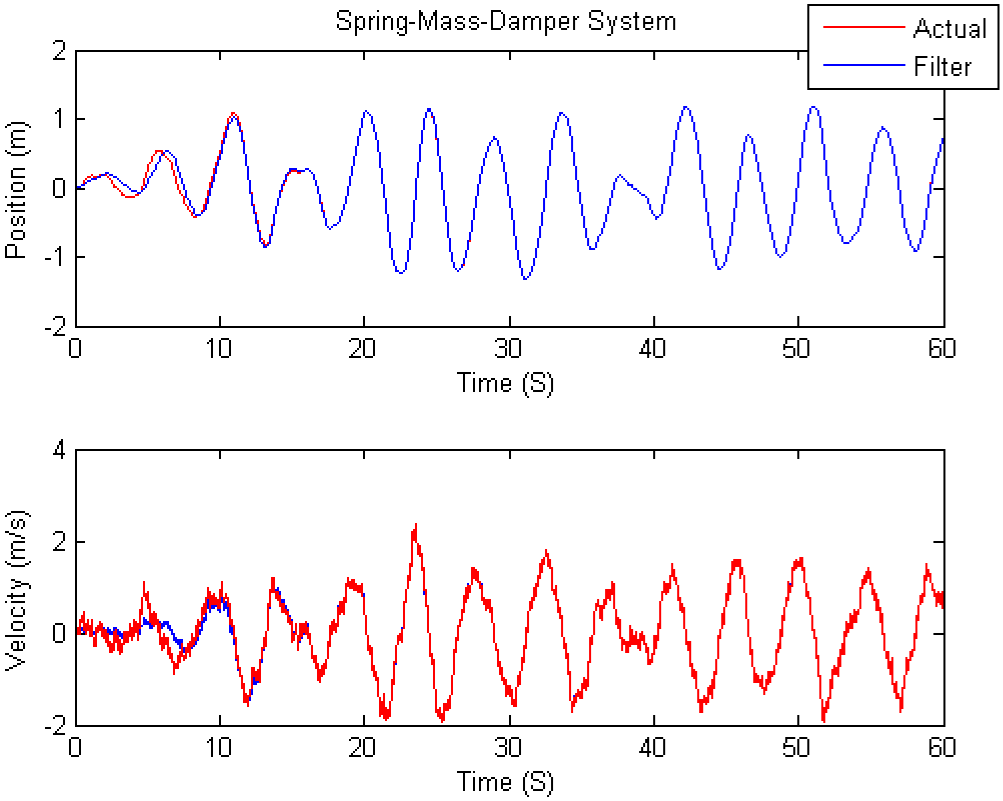


Figure 3.25 State Tracking of the Spring-Mass-Damper System for Band-Limited White Noise Input

For the sin input signal the parameters almost converge to the correct values, but once they get close they stop updating and the distributions begin to fan out. Using the random noise signal produces a much more useful excitation and the parameter values converge to the correct values without biases, shown in **Figure 3.26**.

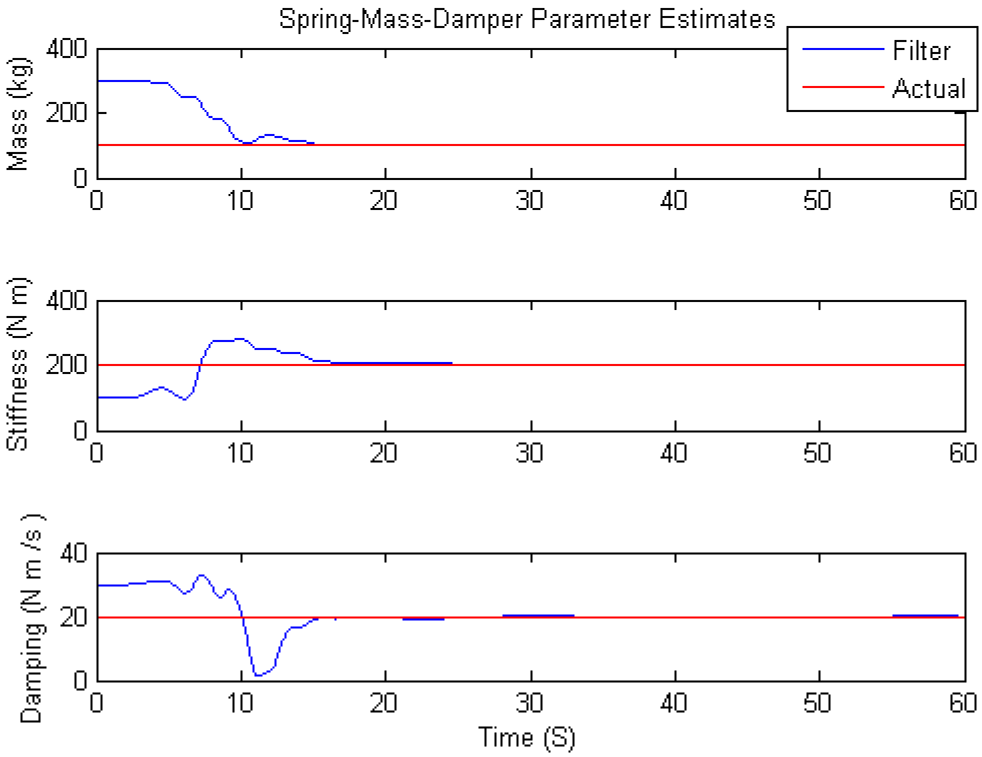


Figure 3.26 Parameter Trajectories for the Spring-Mass-Damper System Excited by Band-Limited White Noise

The parameter distribution for the sin excitation signal lacks enough information to control the parameter distributions. As such, once the parameter gets close enough the residual is small enough that the distribution begins to fan out. When the random noise signal is used the mass distribution stays well contained and by the constant boundary fluctuations in the distribution shows that the signal is informative, as can be seen in **Figure 3.27**.

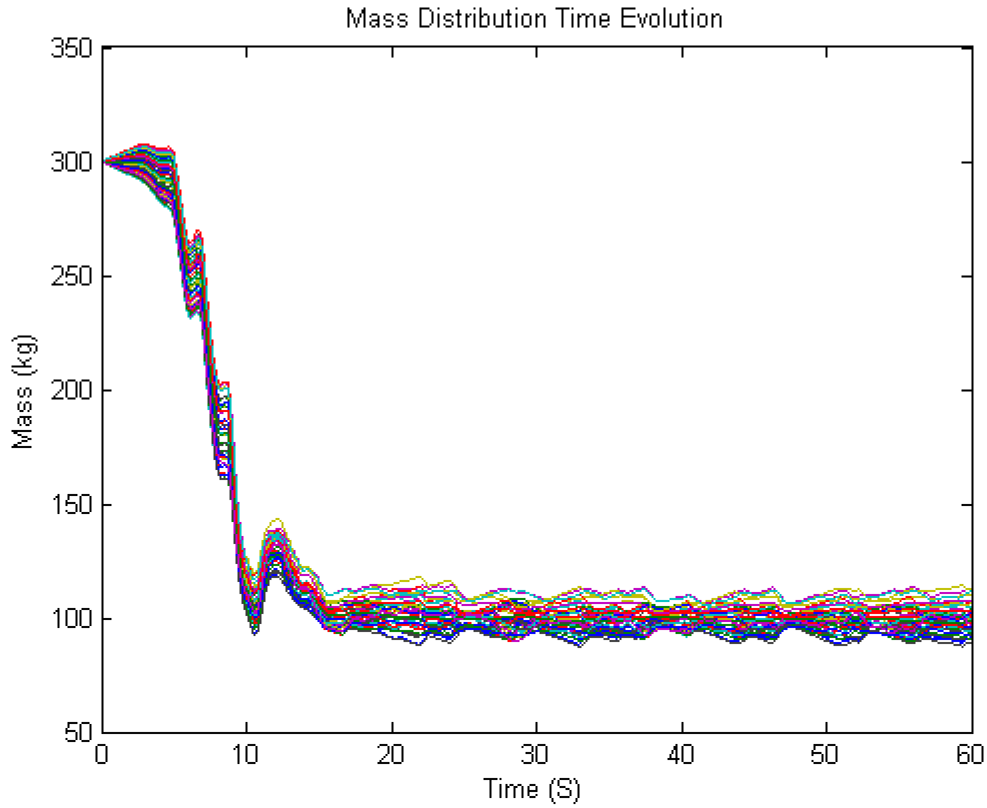


Figure 3.27 Parameter Convergence for the Spring-Mass-Damper System with a Band-Limited White Noise Signal

3.5 Convergence Estimation

Because the gPC-EKF can be difficult to tune, a test for checking convergence of the filter was investigated. Because of the custom adjusted covariance matrix for the tracked state the filter's estimate of the tracked state should be similar to the actual trajectory, and the error residual will show how accurate of an assumption that is. Once the residual is small, the convergence test can be used.

For the spring-mass system that is tracking the position, the convergence test is comparing the derivative of the filter's estimate of position to the filter's estimate of the velocity. The reason this works because the filter’s estimate of the velocity depends on the system parameters, where the derivative of the filter’s estimate of the position does not.

Simulating the same mass-spring system as defined in section 3.4.1, this convergence test is investigated. The gPC-EKF’s coefficients are listed in **Table 3.7**.

Table 3.7 Parameter values for the unmodified gPC-EKF

Poly Order	4
Q	3
Time Step (S)	0.01
c	0.0001
c_1	0.001
x_0	-5
v_0	10
m^1	600
m^2	0
N_1	15+3*WhiteNoise
N_2	15+3*WhiteNoise
R	0.01

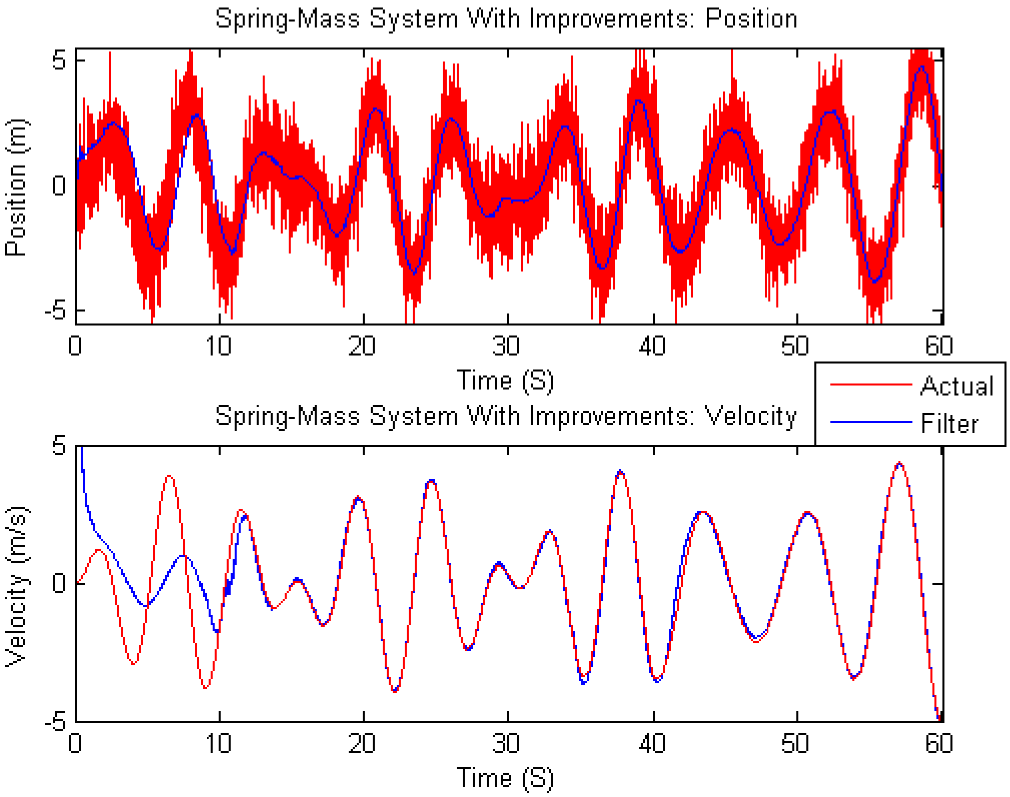


Figure 3.28 State Estimate with Gaussian White Noise for the Convergence Simulation

Figure 3.28 shows the filter's state values as compared to the actual values. As can clearly be seen, when the mass parameter converges, shown in **Figure 3.29**, both the velocity of the filter as compared to the actual, shown in **Figure 3.28**, and the velocity of the filter as compared to the derivative of the position from the filter, shown in **Figure 3.30**, converge.

The idea for the convergence test arose from looking at the state response of the system. When looking at the tracked states error, an estimate of the error between those two signals is well known, because that state is measured, except for the effects of the sensor's noise. When looking at the velocity state, however, it becomes clear that that state does not converge until the parameter converges, which is clearly seen in all of the examples shown in this work.

Because the tracked state is converging because of the artificial perturbation of the covariance matrix, the estimate of that state does not depend upon the system itself. This is powerful, because once this is tuned well the derivative of this is much smoother than the derivative of a noisy signal, like the one demonstrated.

Since the derivative of this signal can be estimated, its comparison to the filter's estimate of velocity, which does depends on the parameters in the system, can be used. A Savitzky-Golay second order filter with 41 points was used to smooth and calculate the derivative of the filter's estimate of the position. The Savitzky-Golay filter is used because it is a least-squares based recursive filter that excels at simultaneously calculating derivatives while robust to sensor noise.

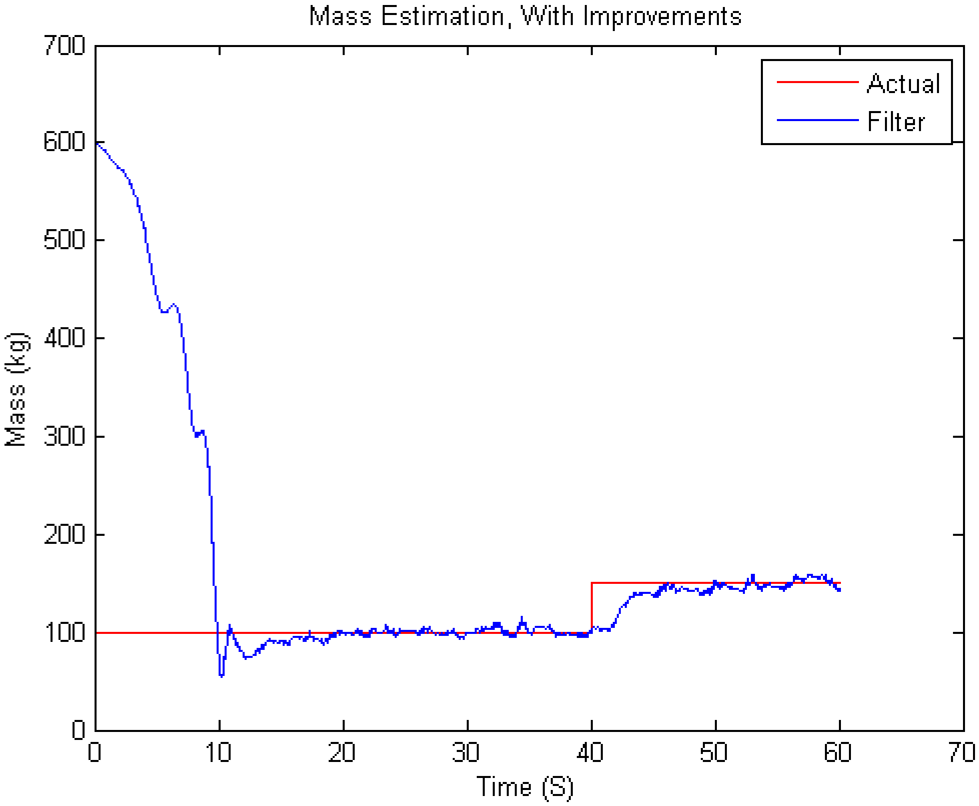


Figure 3.29 Mass Estimation for the System Demonstrating the Convergence Test

When the two velocity curves, shown in **Figure 3.30**, overlap is when the filter has converged. This test may be constructed for any system, given sufficient knowledge about the system's equations. It may not just involve differentials, but also integrals.

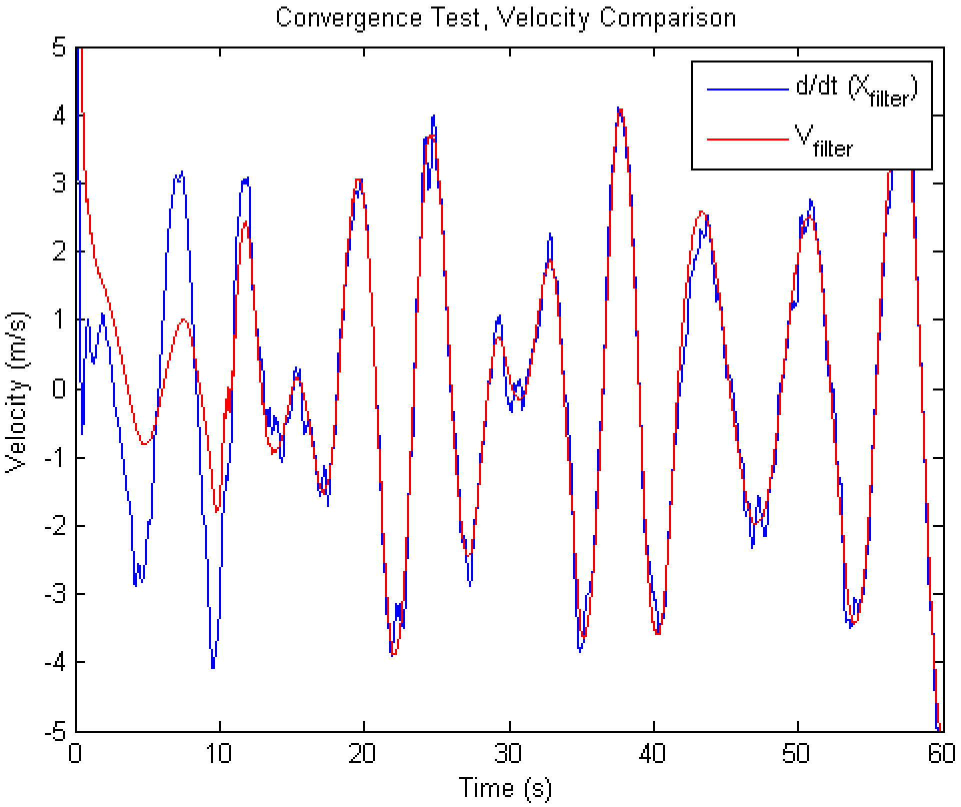


Figure 3.30 Convergence Test, Mass-Spring System

For this system, it is easy to see why it works. The analysis of the proposed convergence interval is shown as a simple proof:

$$\frac{d}{dt}\left(x_{filter}\right) - v_{filter} = \frac{d}{dt}\left(x_{filter}\right) + \int\left(\frac{k}{m_{filter}}x_{filter} - \frac{200\sin(t)}{m_{filter}}\right)dt \qquad (68)$$

If x_{filter} is sufficiently close to x_{actual}, then $\frac{d}{dt}\left(x_{filter}\right)$ will be close to $\frac{d}{dt}\left(x_{actual}\right)$ and

$$\int\left(\frac{k}{m_{filter}}x_{filter} - \frac{200\sin(t)}{m_{filter}}\right)dt = \int\left(\frac{k}{m_{actual}}x_{actual} - \frac{200\sin(t)}{m_{actual}}\right)dt \qquad (69)$$

Which will only be equal when m_{filter} is equal to m_{actual}, given $x_{filter} \approx x_{actual}$, which is known a priori.

3.6 gPC-EKF for Regression Systems

The gPC-EKF is essentially still a Kalman filter, which means that use on regression systems is quite simplistic. There is, however, a significant benefit to doing this, which come from the improvement additions.

3.7 Improvements to the gPC-EKF for Regression Systems

For the regression systems, the gPC-EKF operates a little differently than the state-space system. In the state-space systems the added diagonal matrix from equation (49), aids in the state estimation, which directly effects the parameter estimation. For regression systems, the parameter estimation needs to be based solely off of the error in the observed states. Therefore, the added diagonal matrix is not used, and the Kalman update for the states is set to zero:

$$K = \left[0_1 \dots\ 0_n, k_{p1} \dots k_{pn}\right]^T \tag{70}$$

The added state noise is also removed. When the parameters have information added to their distributions in the state-space systems, the forward integrations smooth out the effect and propagate it to the next time interval. Here, the added parameter information directly impacts the parameter values and can heavily affect the estimates. For systems with large numbers of uncertain parameters, the added parameter information should be very small. The larger the addition to the parameters at each time step, the larger their covariance matrix elements become, and the more the parameters move with regard to the residual.

3.8 Simple Roll Plane Vehicle Model Demonstrating gPC-EKF

A vehicle model that is non-linear in the parameters is used for demonstration of the regression estimator for multiple parameters at once. The vehicle diagram is shown in Figure 3.31 and the equations of motion are defined as:

$$m\,\ddot{Z} = \sum_{i=fl,fr,rl,rr} SF_i \tag{71}$$

$$J_\phi\,\ddot{\phi} = l\left(SF_{fl} + SF_{rl}\right) - r\left(SF_{fr} + SF_{rr}\right) + mhA_y \tag{72}$$

$$J_\theta\,\ddot{\theta} = -a\left(SF_{fl} + SF_{fr}\right) + b\left(SF_{rl} + SF_{rr}\right) + mhA_x \tag{73}$$

Definition of variables: $\ddot{Z}$ is the bounce acceleration of the vehicle, $\ddot{\theta}$ is the pitch acceleration, $\ddot{\phi}$ is the roll acceleration. Definition of parameters: l, r, a, b are geometric properties of the vehicle, with l as the distance from left side of the vehicle to the vehicle's center of gravity (CG), r is the distance from the right side to the CG, a is the distance from the front axle to the CG, and b is the distance from the rear axle to the CG. The physical properties of the vehicle, m, J_θ, J_ϕ are the mass, the roll inertia and the pitch inertia. SF_i represents the suspension forces at the front left (fl), front right (fr), real left (rl), and rear right (rr). The height of the CG from the ground is the parameter h.

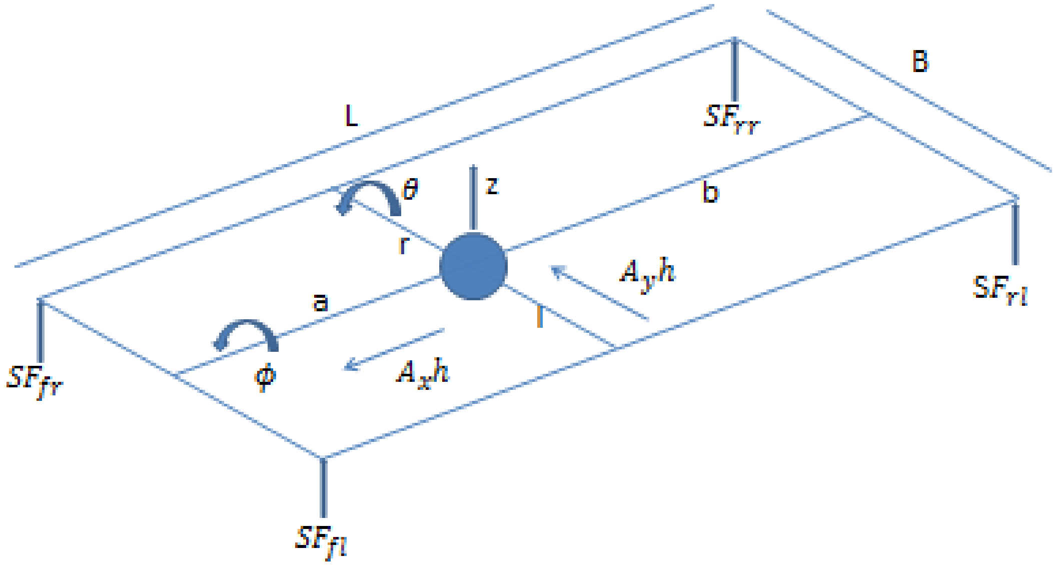


Figure 3.31 Vehicle Model Diagram

3.8.1 Integration of the Simple Roll Plane Vehicle Model with gPC-EKF

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

$$\boldsymbol{x} = \left[\ddot{Z}, \ddot{\theta}, \ddot{\phi}; \boldsymbol{p}\right]^T \tag{74}$$

Where the parameter vector is:

$$\boldsymbol{p} = \left[m, J_\theta, J_\phi, h\right] \tag{75}$$

Specifying the uncertain parameter’s domains and iterations:

$$m_i = \sum_j m^j \psi_1^j(\mu_1^i) \qquad J_{\theta,i} = \sum_j {J_\theta}^j \psi_2^j(\mu_2^i) \tag{76}$$

$$J_{\phi,i} = \sum_j {J_\phi}^j \psi_3^j(\mu_3^i) \qquad h_i = \sum_j h^j \psi_4^j(\mu_4^i)$$

Where

$$m^1 = 2850\ kg, \qquad {J_\theta}^1 = 2000\ kg, \tag{77}$$

$$m^2 = 200\ kg, \qquad {J_\theta}^2 = 400\ kg,$$

$$m^{3-\infty} = 0 \qquad {J_\theta}^{3-\infty} = 0$$

$$ {J_\phi}^1 = 600\ kg, \qquad h^1 = 0.4\ kg,$$

$$ {J_\phi}^2 = 100\ kg, \qquad h^2 = 0.2\ kg,$$

$$ {J_\phi}^{3-\infty} = 0 \qquad h^{3-\infty} = 0$$

The initial conditions at the next time step are:

$$\boldsymbol{X}_{k-1}^{i} = \sum_{j=1}^{S} A_{i,j}\, x_{k-1}^{j} + \left[0,0,0, c_1\, m^2\, \mu_1^i, c_2\, {J_\theta}^2\, \mu_2^i, c_3\, {J_\phi}^2\, \mu_3^i, c_4\, h^2\, \mu_4^i\right]^T \tag{78}$$

With the Kalman Filter measurement matrix:

$$H_k = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{79}$$

3.8.2 Simple Roll Plane Vehicle Model with RLS Method

A linear regression model can be described as:

$$y(t) = \theta^T\, \eta(t) + \nu(t) \tag{80}$$

The output or measurement of the system is $y(t)$, θ is the vector of parameters, $\phi(t)$ is the set of observed signals and $\nu(t)$ is a white noise error term.

For systems that are linear in the parameters the estimation is a simple set of equations.

$$P_k = \frac{1}{\lambda}(I - K_{k-1}\eta_k)P_{k-1} \tag{81}$$

$$K_k = \frac{P_k \eta_k}{\lambda + \eta_k^T P_{k-1}\eta_k} \tag{82}$$

$$\theta_k = \theta_{k-1} + K_k(y_k - \theta_{k-1}^T \phi_k) \tag{83}$$

These are the equations for recursive least squares with exponential forgetting. The system has to be initialized with P_0, θ_0. The matrix P is the covariance of the parameters. λ is the exponential forgetting factor. Typical values for this range from $0.95 \leq \lambda \leq 1$. If the value is set to 1, then the exponential forgetting is disabled, and all measurements have equal weighting. More detailing of the RLS algorithm can be found in [16, 57, 58].

For this vehicle model, the observation matrix is:

$$\eta(t) = \begin{bmatrix} \sum_{i=fl,fr,rl,rr} SF_i(t) & 0 & 0 \\ 0 & l\left(SF_{fl} + SF_{rl}\right) - r\left(SF_{fr} + SF_{rr}\right) & 0 \\ 0 & 0 & -a\left(SF_{fl} + SF_{fr}\right) + b\,(SF_{rl} + SF_{rr}) \\ 0 & A_x & 0 \\ 0 & 0 & A_y \end{bmatrix} \tag{84}$$

With parameter vector:

$$\theta^T(T) = \left[\frac{1}{m}, \frac{1}{J_\theta}, \frac{1}{J_\phi}, \frac{mh}{J_\theta}, \frac{mh}{J_\phi}\right] \tag{85}$$

3.9 Comparison of Parameter Estimation Methods for Regression Systems

The parameter estimation is performed for the vehicle model with no sensor noise. The excitation to the model is a set of sine waves at various frequencies and amplitudes. Table 3.8

lists the filter's coefficients for the estimation. **Table 3.9** lists the RLS algorithm's forgetting factor.

Table 3.8 Parameter values for gPC-EKF for the regression system

Poly Order	2
Q	4
Time Step (S)	0.01
c	0
c_1	1/5000
c_2	1/5000
c_3	1/5000
c_4	1/10000
R	0.001

Table 3.9 RLS Forgetting Factor

λ	0.999

The parameter estimates for both the RLS and gPC-EKF are demonstrated in the next several graphs. **Figure 3.32** shows the CG height estimation. **Figure 3.33** shows the mass estimation. **Figure 3.34** shows the pitch inertia estimation, and **Figure 3.35** shows the roll inertia estimation. For most all of these both methods perform similarly. The RLS method has an error for about 80 seconds during the Roll Inertia estimation.

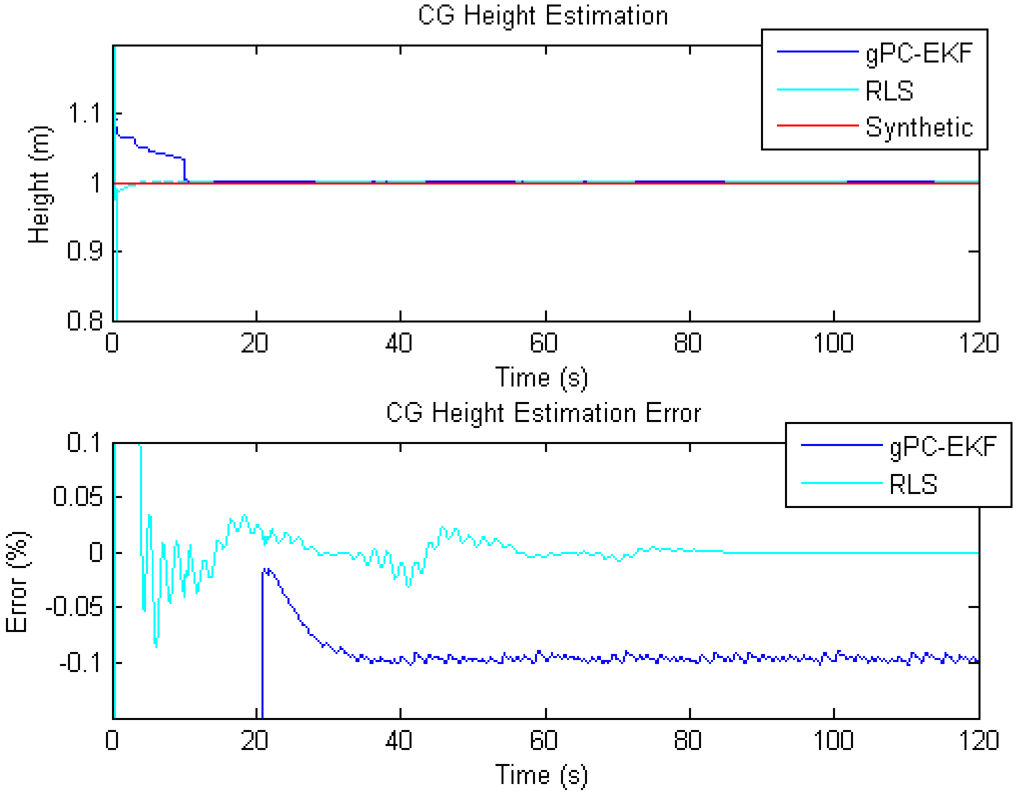


Figure 3.32 CG Height Estimation

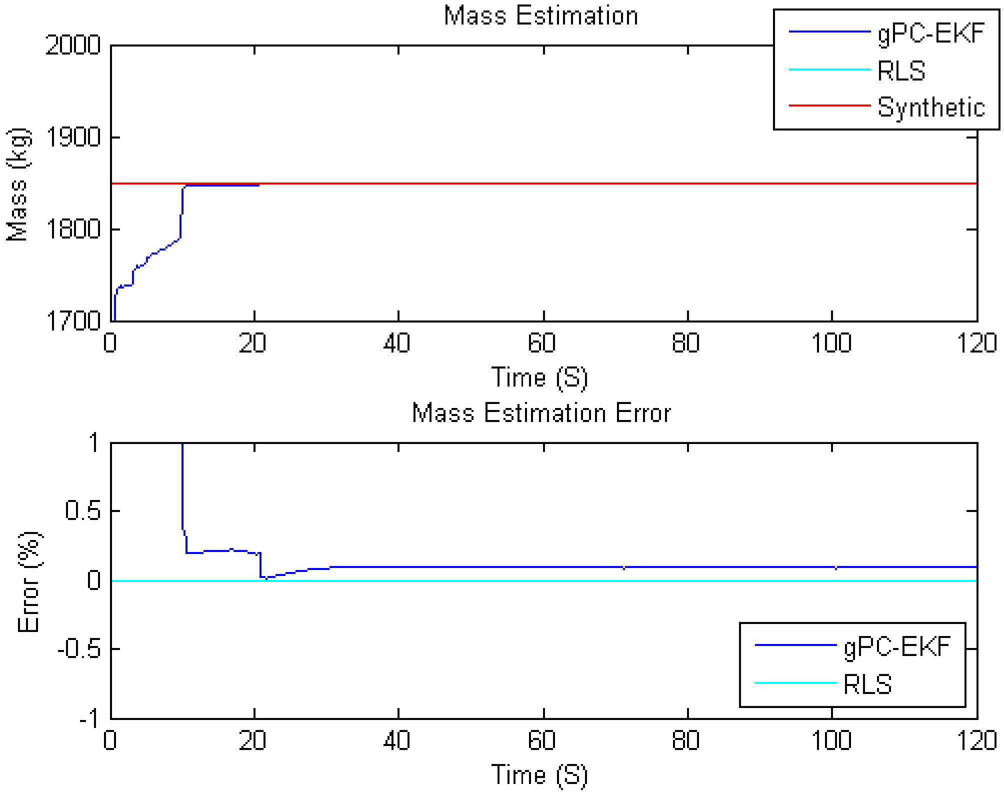


Figure 3.33 Mass Estimation

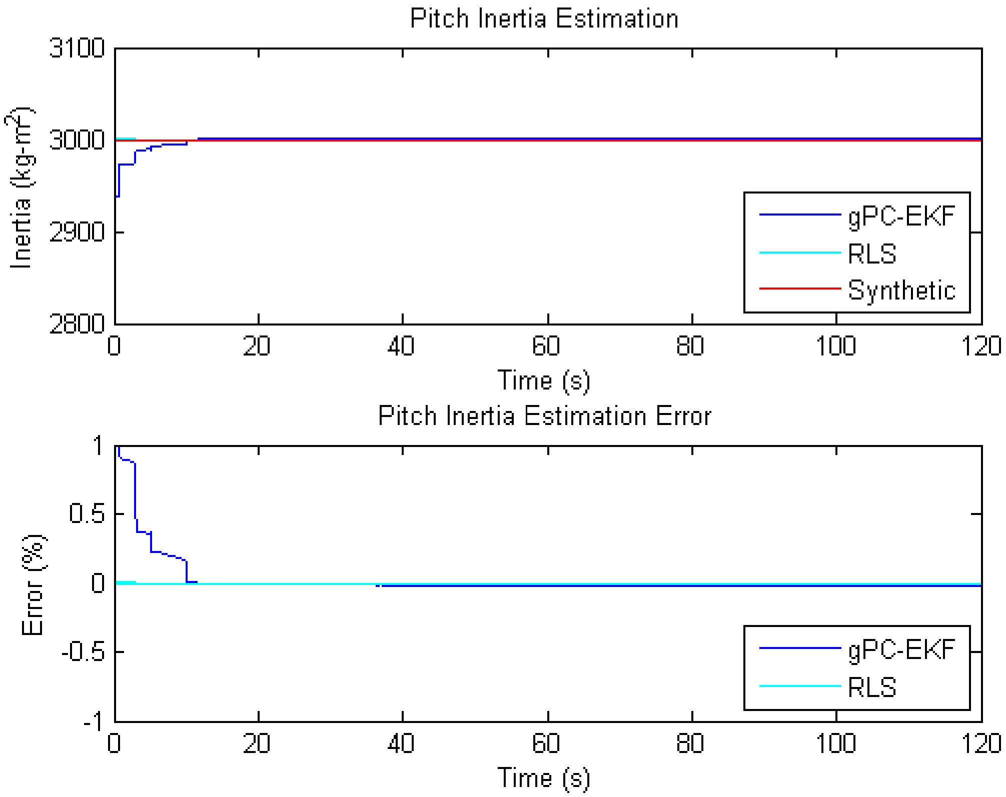


Figure 3.34 Pitch Inertia Estimation

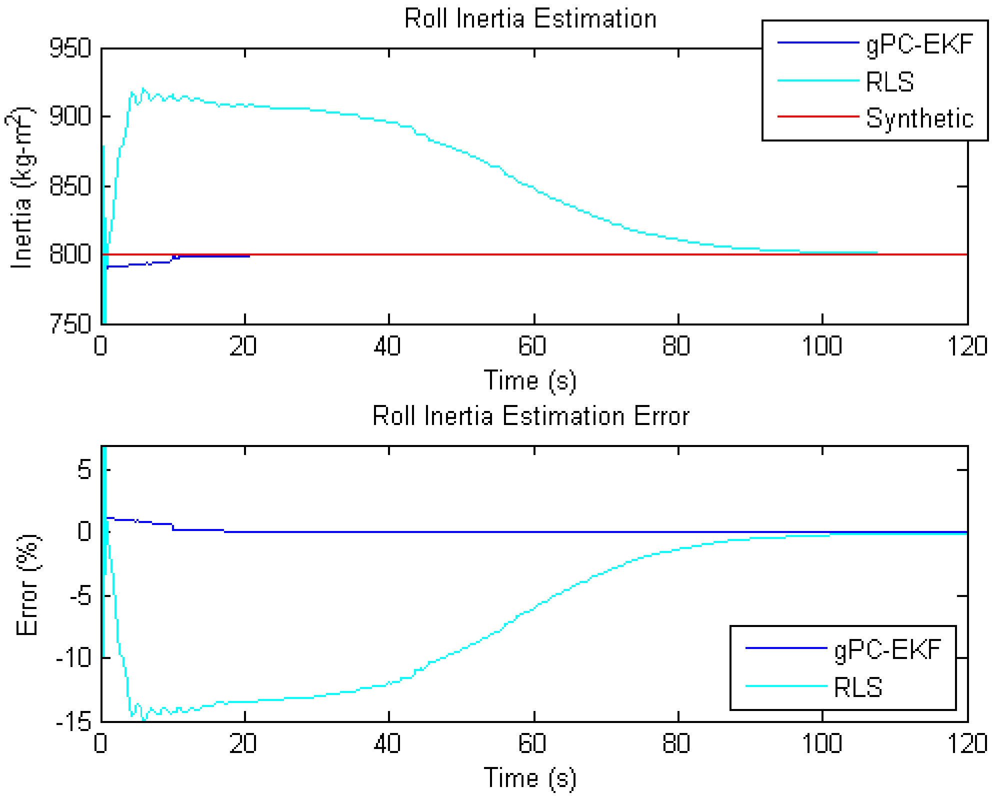


Figure 3.35 Roll Inertia Estimation

The purpose of having the RLS estimates is for comparison of the gPC-EKF with a known baseline. The RLS is a very well-known estimation technique and this shows that the gPC-EKF, when applied to regressive systems, performs similarly to the RLS. The gPC-EKF does have small biases in the parameter estimates compared to the RLS.

The same simulation is performed next, but with a white noise process added to the synthetic sensors. The noise is a 0.2 variance white noise process. **Table 3.10** lists the gPC-EKF's coefficients.

Table 3.10 Parameter Values for gPC-EKF Applied to the Regression System with Noise

Poly Order	2
Q	4
Time Step (TS)	0.01
c	0
c_1	1/10000
c_2	1/10000
c_3	1/10000
c_4	1/20000
R	0.1

The CG height estimation is shown in **Figure 3.36**. **Figure 3.37** shows the mass estimation, **Figure 3.38** shows the pitch inertia estimation, and **Figure 3.39** shows the roll inertia estimation.

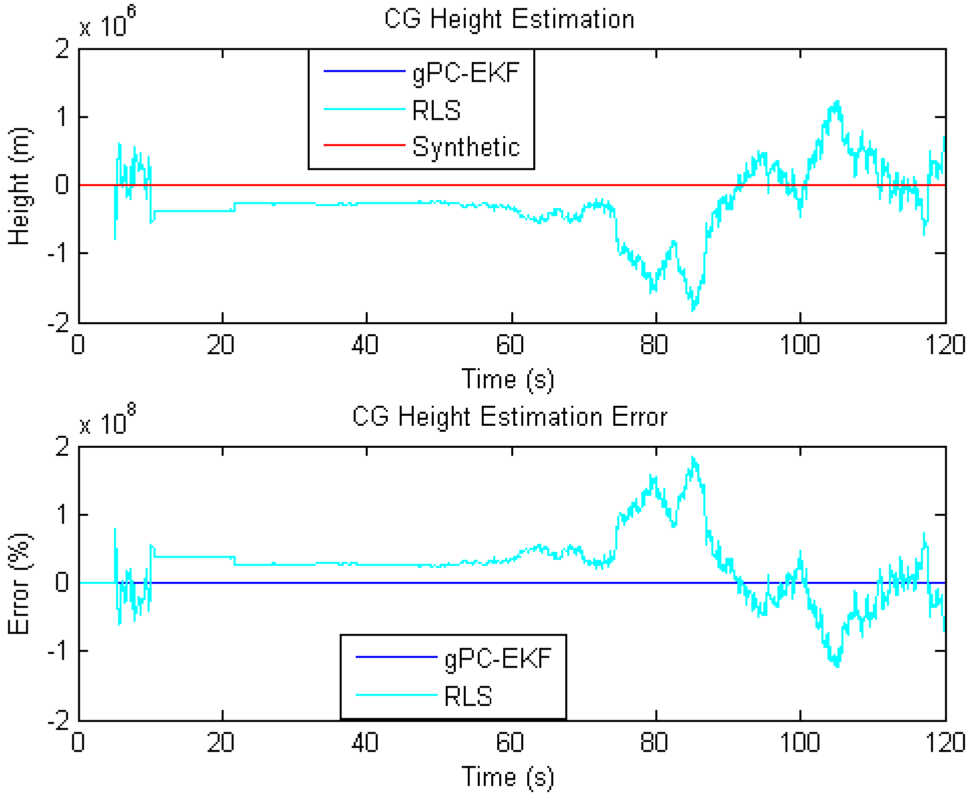


Figure 3.36 CG Height Estimation with Sensor Noise

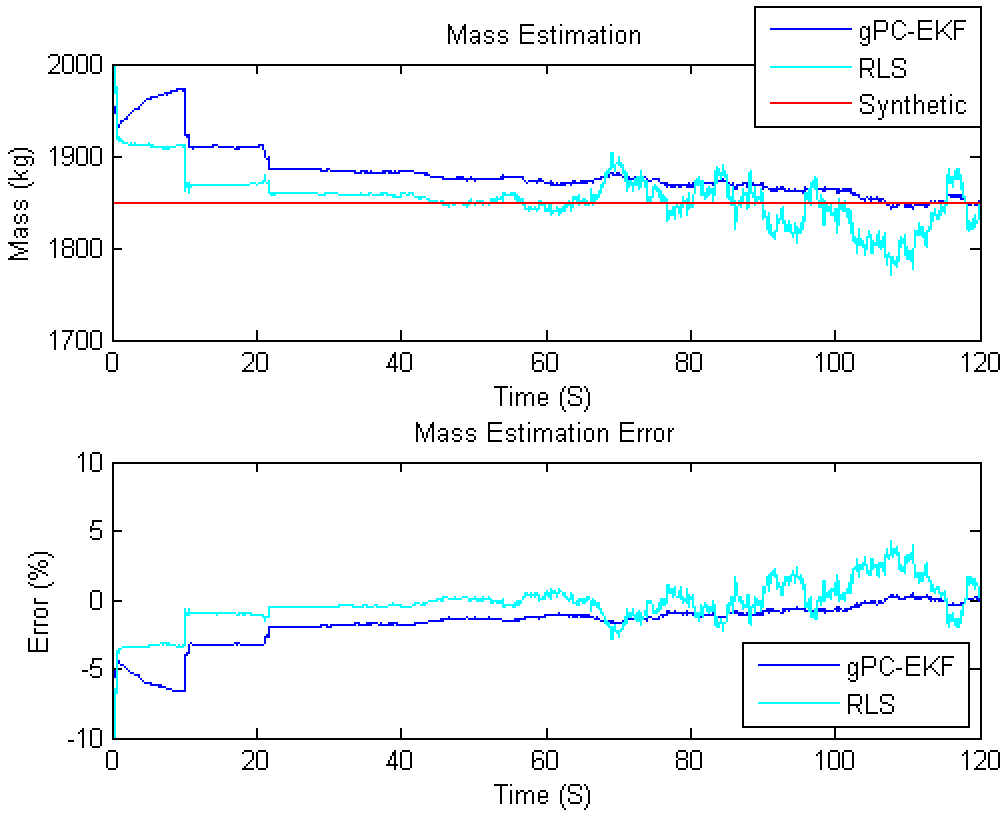


Figure 3.37 Mass Estimation with Sensor Noise

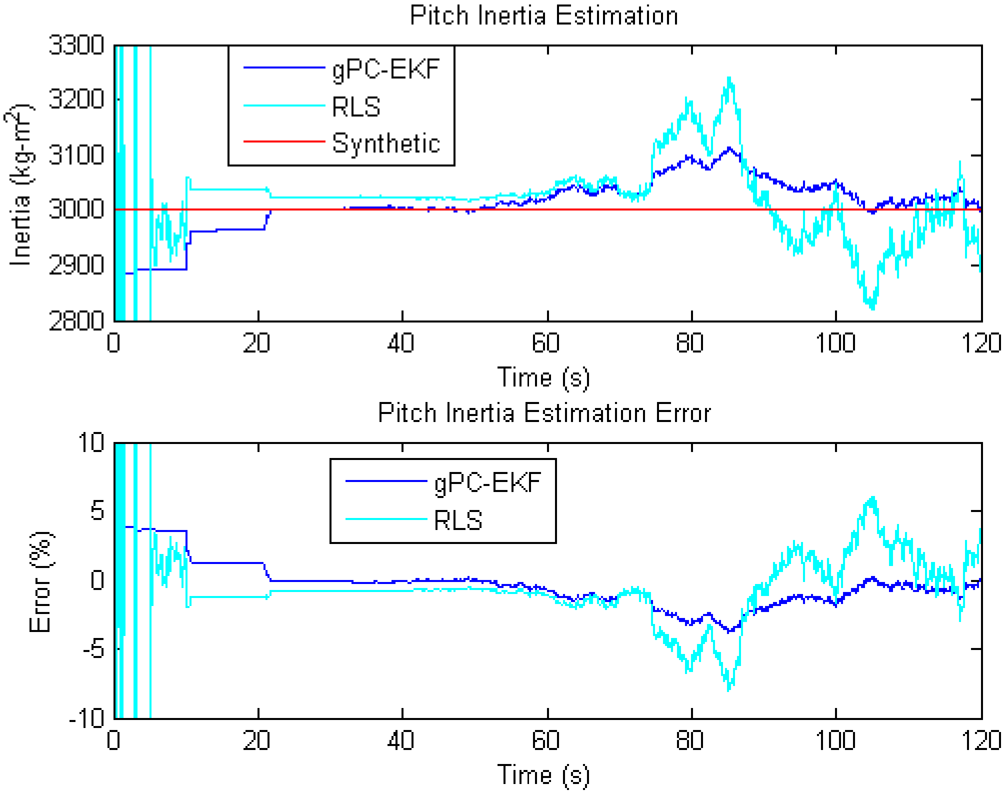


Figure 3.38 Pitch Inertia Estimation with Sensor Noise

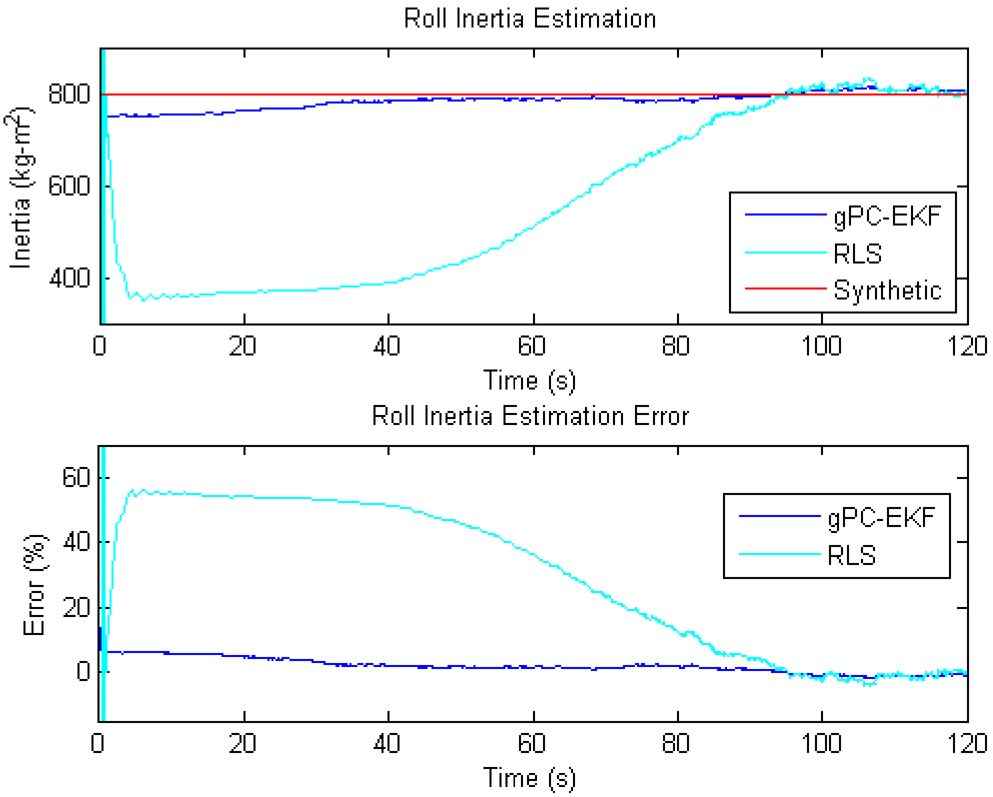


Figure 3.39 Roll Inertia Estimation with Sensor Noise

Previously, in the case where there was no noise, the RLS algorithm did slightly better than the gPC-EKF. However, here it can be clearly seen that the gPC-EKF performs a better estimate. This is because the gPC-EKF performs an estimate for the individual parameters across all of the equations, which makes it less sensitive to noise. The RLS method calculates an individual coefficient for each term in the observation matrix, which leads to the significant error observed for the CG height.

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.

A major benefit of using the gPC-EKF over the RLS is that the addition of the information to the parameter distributions enables separate update rates for each of the parameters. The RLS has a forgetting factor that is used for each sensor feed, and not for the individual parameters. For a system with time-varying and time-invariant parameters, the gPC-EKF can be configured to track both, without having the time-invariant parameters affected by the time-varying parameter's effects.

4 Experimental Vehicle Setup and Test Scenarios

This chapter contains information about the experimental vehicle and the test scenarios that were performed. The first section gives details about the vehicle, such as sensors used and configuration. The second section details the experiments performed and gives some background on each of the tests and their importance for identifying vehicle parameters.

4.1 Experimental Vehicle

The experimental data is collected on a Land Rover Defender 110 off-road vehicle, shown in **Figure 4.1**.

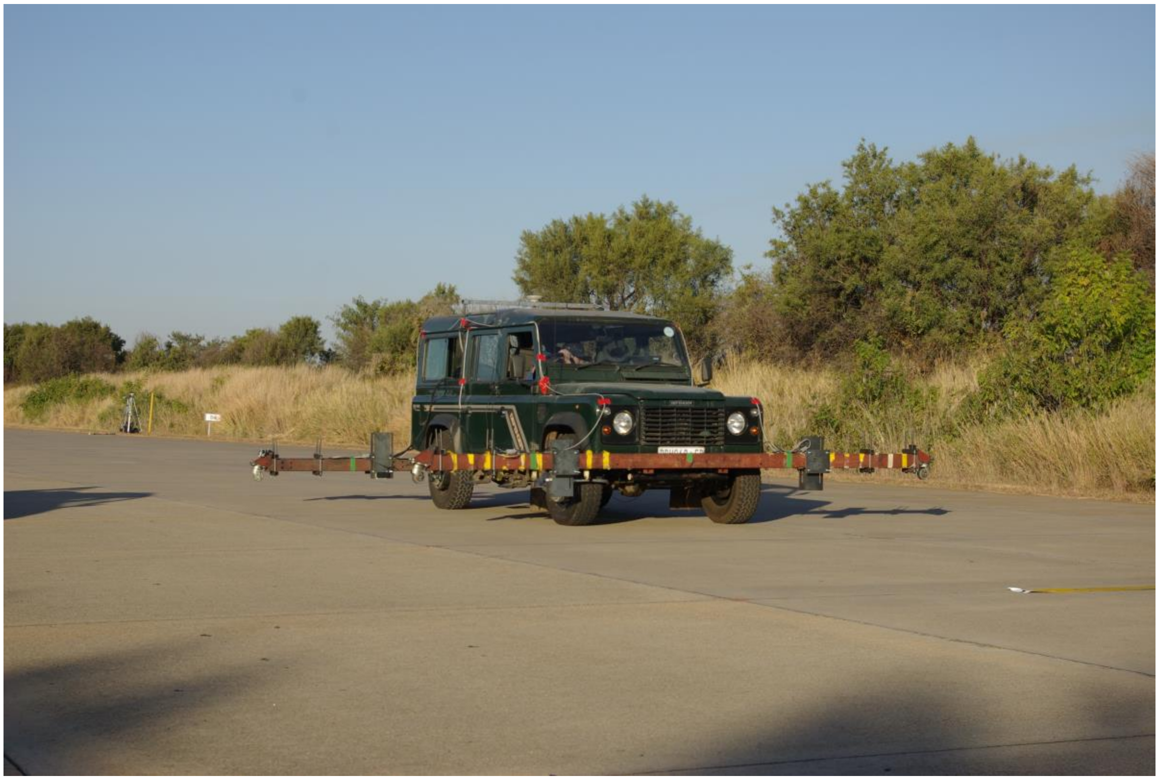


Figure 4.1 Instrumented Land Rover Defender 110

This vehicle is common in South Africa, and is easily modifiable, as shown by the Vehicle Dynamics Group at the University of Pretoria. It has a ladder style chassis, which allowed for significant modification of the suspension struts. The Vehicle Dynamics Group created a custom suspension that can be tuned for different stiffness and damping characteristics, as well as ride height control.

4.1.1 Instrumentation

Two accelerometers are mounted vertically on each strut, a 10g tri-axial accelerometer on the unsprung mass and a 4g tri-axial 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 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 that is recorded on both data acquisition systems. There are linear displacement sensors on each strut. A sensor is placed on the rear of the vehicle that measures lateral and longitudinal velocity, and calculates the vehicle's slip angle.

Each of the suspension struts has a pressure transducer built into it that measures the suspension force. The rear left strut also has a load cell measurement of the suspension force. There are laser displacement sensors mounted near each wheel to measure body displacement from the ground. A high precision GPS system, with a base station nearby is used for precise 3 dimensional position measurements, in addition to the Racelogic VBox. Each strut has a linear displacement potentiometer to record the suspension displacement. The rear left wheel has a custom-built wheel force transducer built into the wheel hub. The sensor transmits the 3 moments and 3 forces that are applied to the wheel.

4.1.2 Suspension Strut Design and Implementation

The Land Rover is fitted with a prototype Hydro-pneumatic spring-damper suspension system developed by the Vehicle Dynamics Group of the University of Pretoria. The schematic is illustrated in **Figure 4.2**, and the physical version is shown in **Figure 4.3 [59]**.

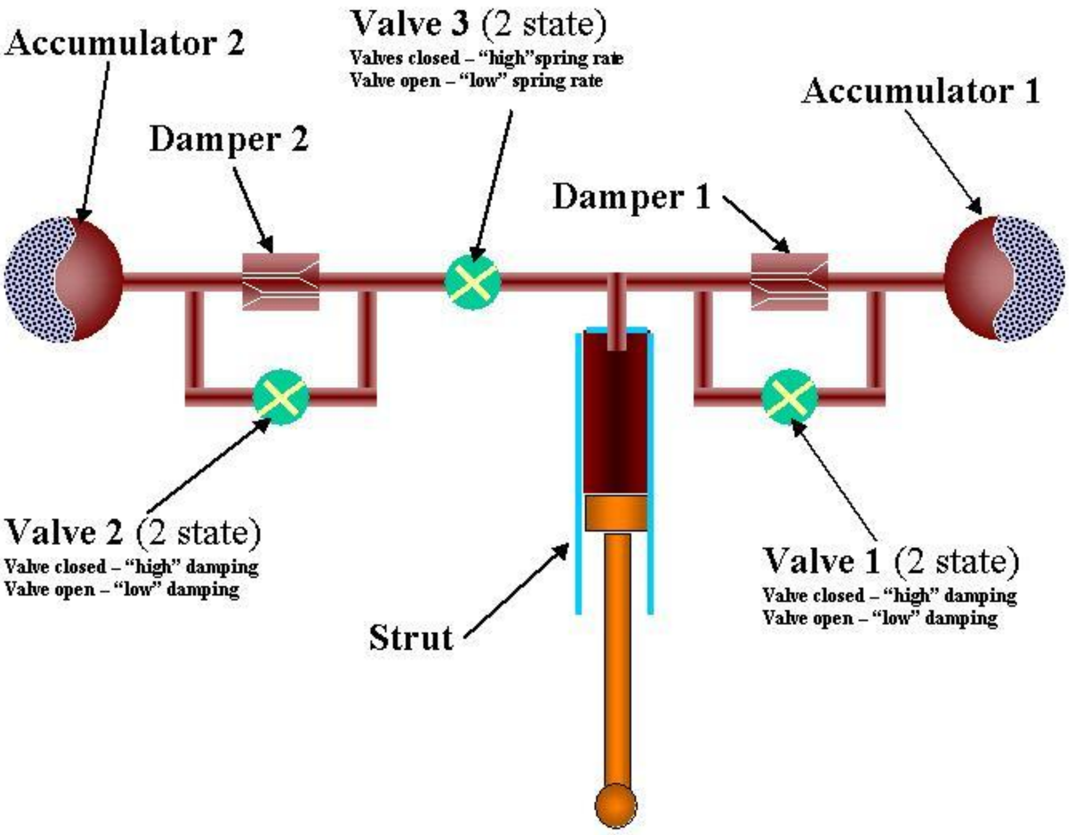

Accumulator 2
Valve 3 (2 state)
Valves closed – "high"spring rate
Valve open – "low" spring rate
Accumulator 1
Damper 2
Damper 1
Valve 2 (2 state)
Valve closed – "high" damping
Valve open – "low" damping
Strut
Valve 1 (2 state)
Valve closed – "high" damping
Valve open – "low" damping

Figure 4.2 Technical Schematic of Hydro-Pneumatic Spring-Damper

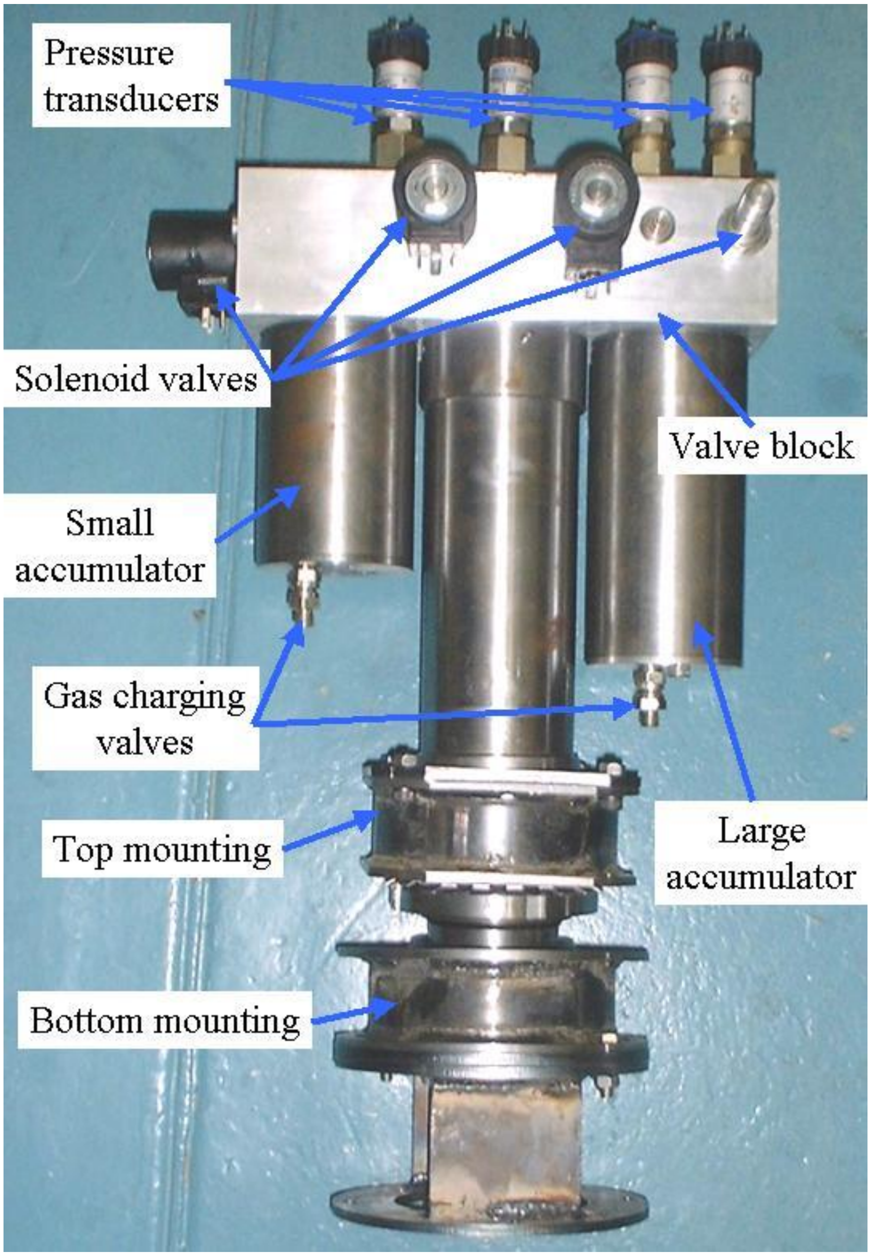


Figure 4.3 Picture of the Physical Hydropneumatic Spring-Damper

Each strut is equipped with a linear potentiometer displacement transducer that is used to obtain the strut's displacement. A pressure transducer is mounted at the top of each strut that 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 the columbic stick-slip friction force. The friction in the suspension has a very non-trivial effect upon the vehicle's performance and ride. The friction effects can be seen in **Figure 4.4** and **Figure 4.5 [59]**.

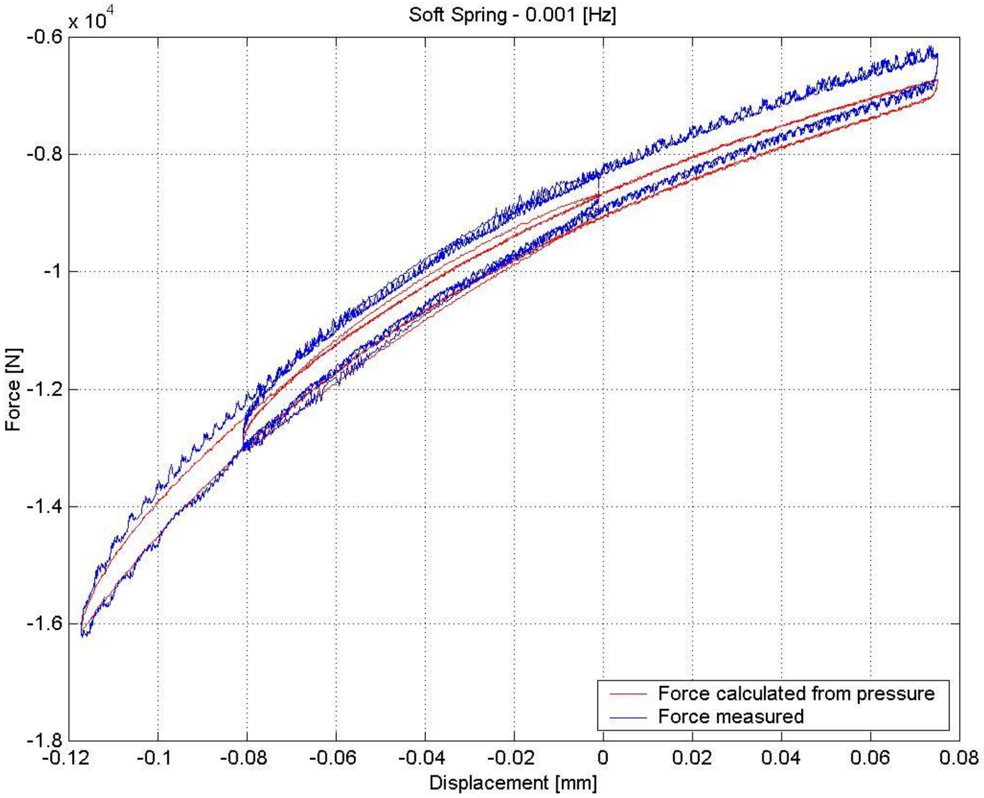


Figure 4.4 Suspension Friction Effects for Low Frequencies

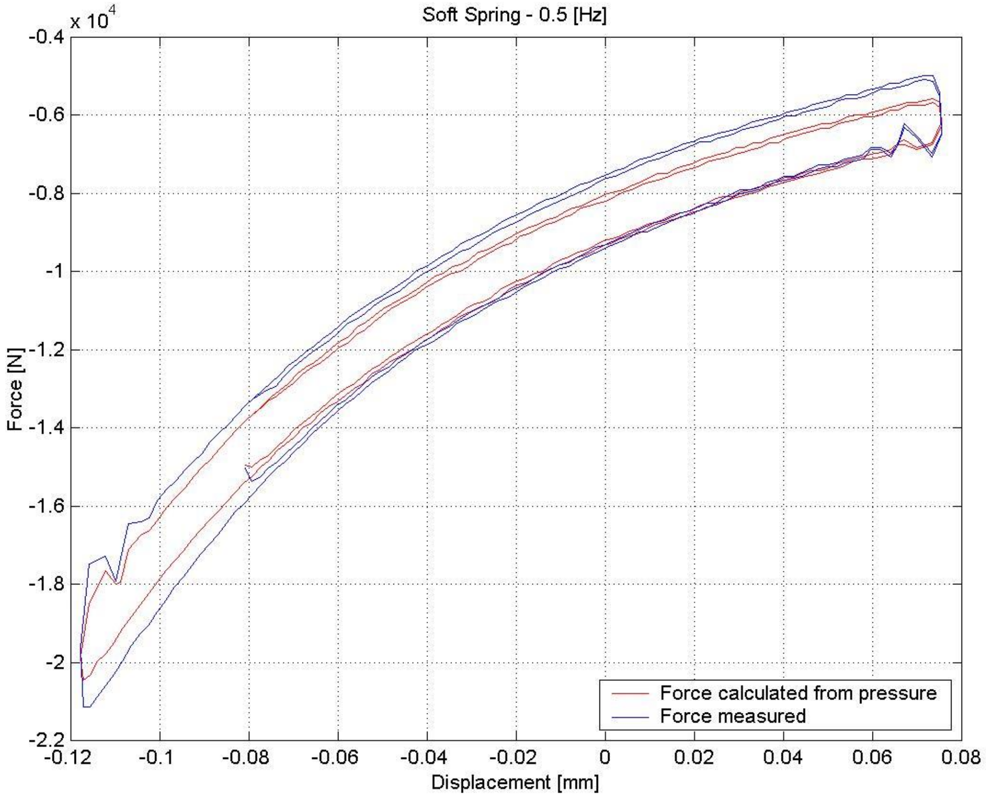


Figure 4.5 Suspension Friction Effects for Higher Frequency Oscillations

The effects of friction are present in both kinetic and static quantities. The effects are significant enough to cause constant rotations in the vehicle. During the constant radius tests the vehicle would gain a constant roll bias due to the friction effects. Driving of the vehicle would slowly remove this, or, as demonstrated in **Figure 4.6** someone can jump on an outrigger to provide a counter moment (while the vehicle is stationary).

Figure 4.6 Removal of Constant Rotation Bias Due to Suspension Friction

The effects of the friction are not a uniform effect, but are biased towards one direction of strut travel.

4.1.3 Suspension Configuration

The suspension of the vehicle is a fairly basic suspension. It was originally coil springs but has been retrofitted with the above suspension struts. The front suspension system is two horizontal leading arms with a lateral stabilizing panhard rod, and is diagramed in **Figure 4.7**.

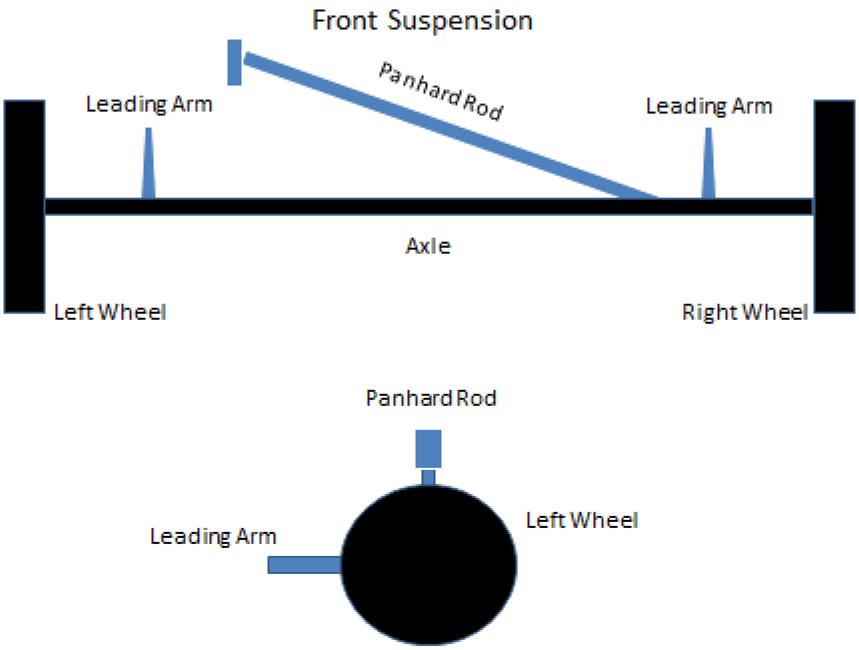


Figure 4.7 Front Suspension Linkage Diagram

The rear suspension on the vehicle is composed of two horizontal leading arms with an A arm suspension in the middle for guidance and lateral stabilization. A diagram of this configuration is shown in **Figure 4.8**.

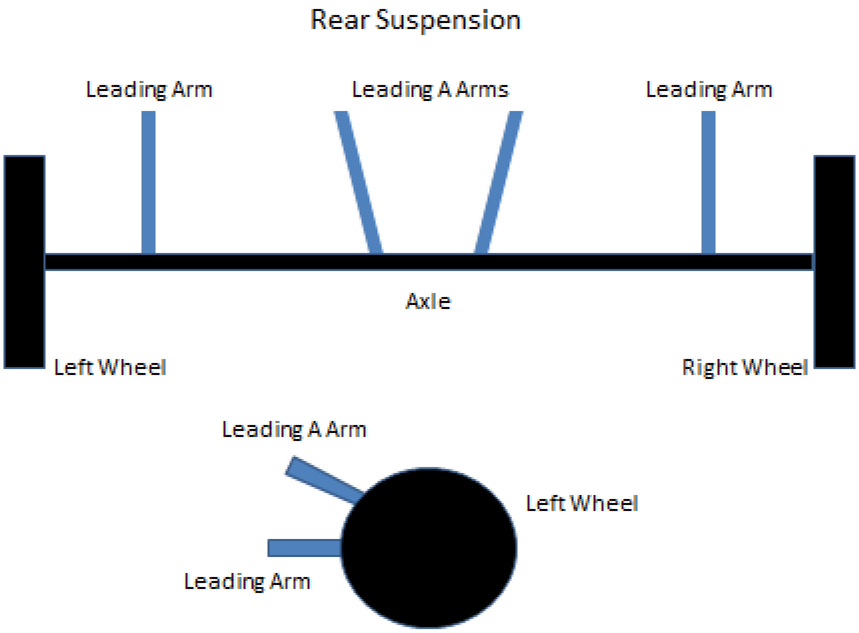


Figure 4.8 Rear Suspension Linkage Diagram

4.2 Vehicle Data Sets

Several vehicle scenarios were performed in order to gather a substantial data set. This section showcases each of the tests and describes their importance.

4.2.1 Constant Radius

One of the types of rollover that was described above was the event where the vehicle slowly rolls over due to a slowly increasing lateral force. For a vehicle driving in a constant radius, the lateral force is proportional to the square of the vehicle's velocity. A constant radius test is therefore an excellent example of a slow rollover environment. The speed of the vehicle is slowly increased as it drives around the circle. Eventually, if the speed reaches a critical level, the vehicle will begin to rollover. A picture of the track used for the constant radius test is shown in **Figure 4.9**.

Figure 4.9 Constant Radius Test

4.2.2 Dynamic Handling Track

The dynamic handling track contains many standard concepts found on a regular road. There are counter cambered turns, decreasing radius turns and combinations of road banks and grades. This track gives a compact recreation of many of the effects of standard driving.

Being able to control a vehicle on a flat surface is one thing, being able to do it for disadvantageous and complicated road surfaces is a completely different story. A diagram of the track used for the Dynamic Handling tests is shown in **Figure 4.10**.

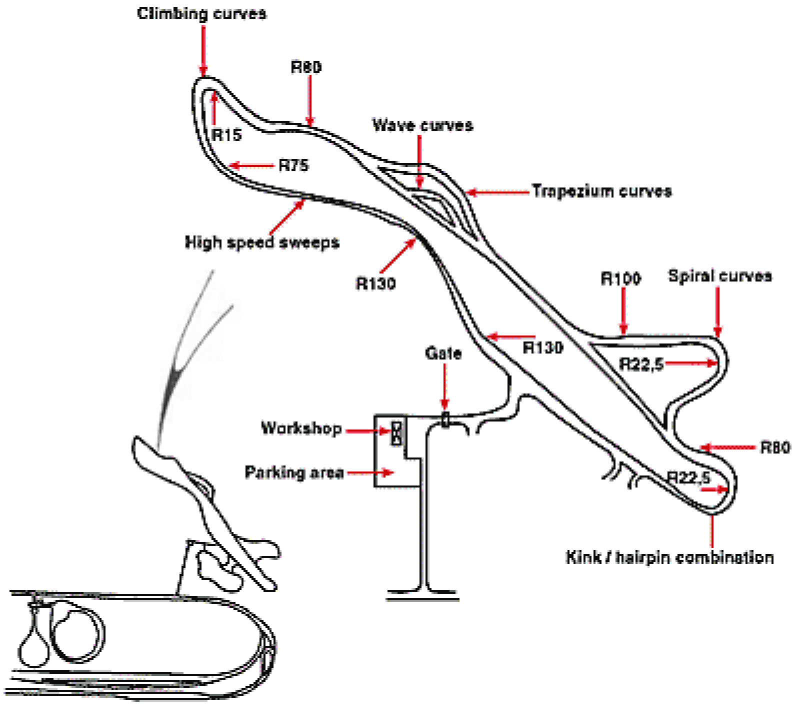


Figure 4.10 Dynamic Handling Track at Gerotek*

* With permission from Gerotek Test Facilities, a Test Facility in South Africa.

Tel No: +27 12 371 2065

www.gerotek.co.za

This test is a solid test of the vehicle, and its results are quite useful for both the estimation of the vehicle's parameters, as well as the mapping of its suspension as one data-set on this course can contain many important characteristics of the vehicle's suspension and response to high excitation maneuvers. The data is used both for parameter estimation as well as the mapping of the vehicle suspension, which is used to estimate the tire normal force without the need for the wheel force transducer.

4.2.3 Long Straight Track

The Long Straight Track, shown in **Figure 4.11**, is used for Double Lane Change maneuvers, and Sine Sweep tests. Data was collected for both of these maneuvers, but the Double Lane Change data was not used for validation because more sensors are needed to properly characterize an effect that happens during longitudinal accelerations when the outriggers are equipped. The vehicle gains a significant lateral load transfer during longitudinal acceleration only when the outriggers are attached to the vehicle.

The Sine Sweep data is used, and the un-modeled effect does show up, though much smaller due to the smaller initial accelerations.

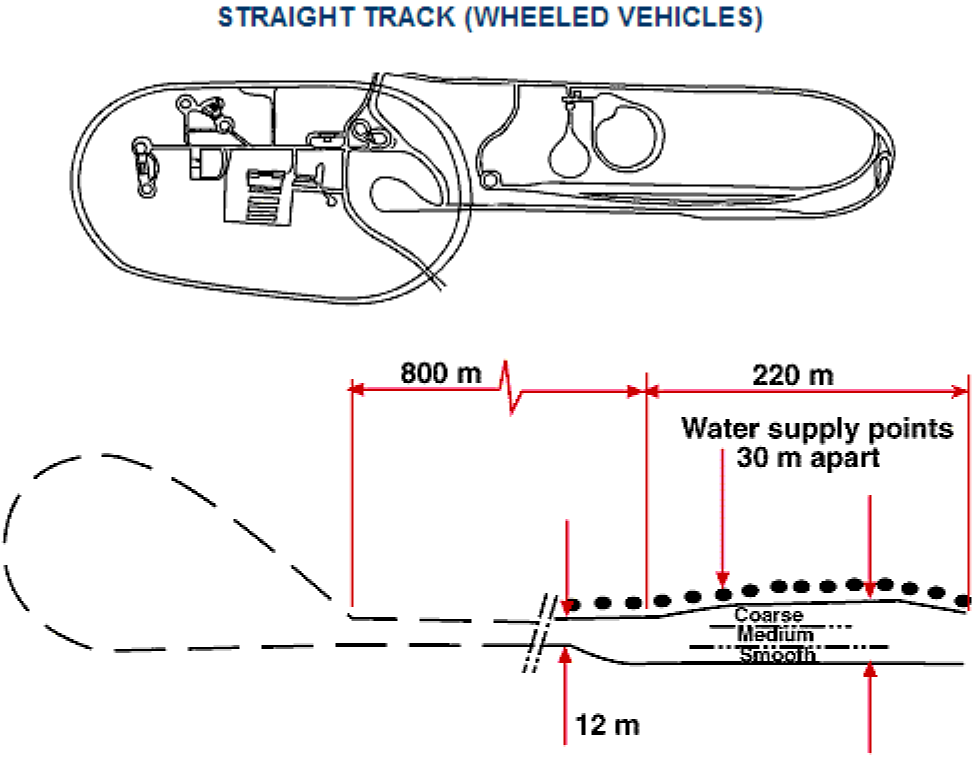


Figure 4.11 Long Straight Track*

* With permission from Gerotek Test Facilities, a Test Facility in South Africa.

Tel No: +27 12 371 2065

www.gerotek.co.za

4.2.4 Driving Around Gerotek

Data is collected from the vehicle as it drives around the Gerotek facility between tracks. This data is also used for validation purposes of the methods developed here. The maneuvers performed as relatively simple and unexciting, and their results are representative of the average drive around the neighborhood.

4.2.5 Rural Road

The four previous tracks are used to demonstrate the capabilities of the tire normal force estimation and the applicability of the method to maneuvers where it is designed to function. The next two scenarios highlight the vehicle parameter estimation technique during standard driving maneuvers. Two different data sets were used in evaluating the parameter estimation. The first data set was obtained on a winding rural road, shown in **Figure 4.12**, 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 patterns of an everyday driver. The uneven road surface and speed profile in conjunction with the road cornering, embankment and grade result in considerable excitation to the vehicle [60].

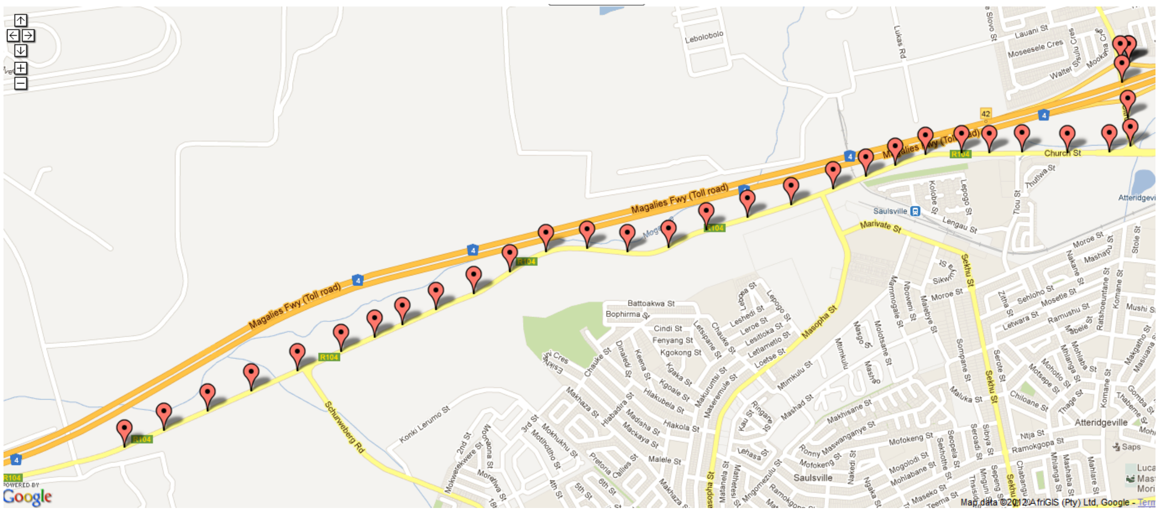


Figure 4.12 Rural Data Experiment Path

4.2.6 Urban Road

The second data set was obtained in an urban environment, with the route shown in **Figure 4.13**. The road surface was much smoother than the rural road, and there was 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 [60].

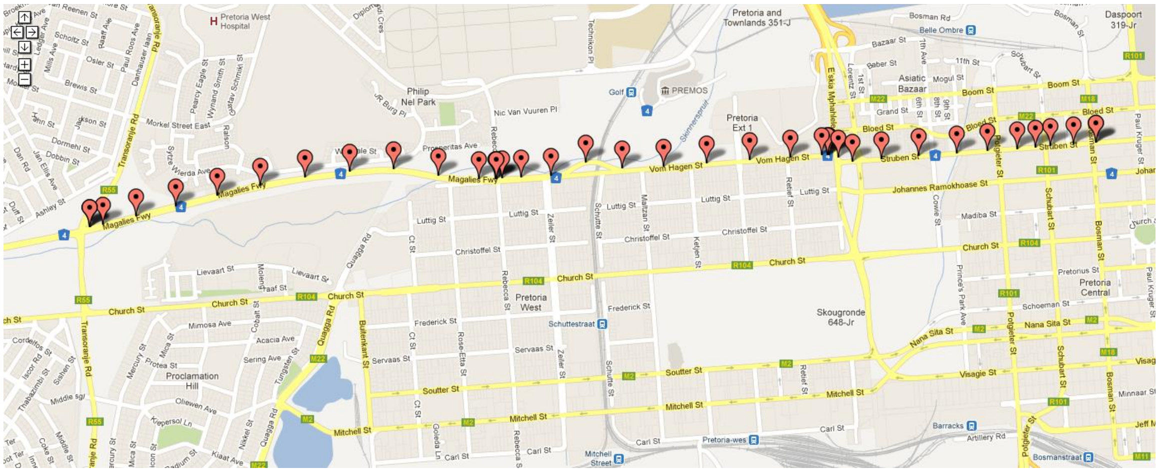


Figure 4.13 Urban Data Experiment Path

The data from this test also clearly highlights the effects of the suspension friction. The vehicle has to stop at several intersections, and during those stops the vehicle never returns to an equilibrium position, but has constant biases due to the friction.

5 First Attempt at Vehicle Parameter Estimation

This research started without a very clear idea of what types of models and methods to, or even could, be used. There is a very large set of vehicle models that have been developed over the years. During the course of the research several models were investigated and implemented. This section covers the first attempt at performing the parameter estimation. There were several reasons that the method used here was not considered viable: The model is too constrained to be used for general driving purposes, and it required that the suspension parameters were known. The thought process of trying to expand this model to using the suspension forces is what led to the use of the load transfer model later on.

5.1 Acceleration Based Model and Demonstration of Difficulties

A common model used in vehicle dynamics is a seven degree-of-freedom (DOF) base excitation model, such as the one presented in **Figure 5.1**. This model consists of a chassis (denoted as the sprung mass) and the suspension systems and wheels on the four corners (denoted as unsprung masses). The model uses the terrain profile to excite the unsprung masses through the tire dynamics; it then propagates the forces up through the suspension elements to excite the sprung mass dynamics. In the present study the terrain profile requirement has been removed; in addition to vertical bounce and pitch DOFs considered in [30], the model in this study includes the roll DOF. This is very important since without including the roll motion of the vehicle, the roll inertia of the vehicle would be ignored, and thus could not be estimated.

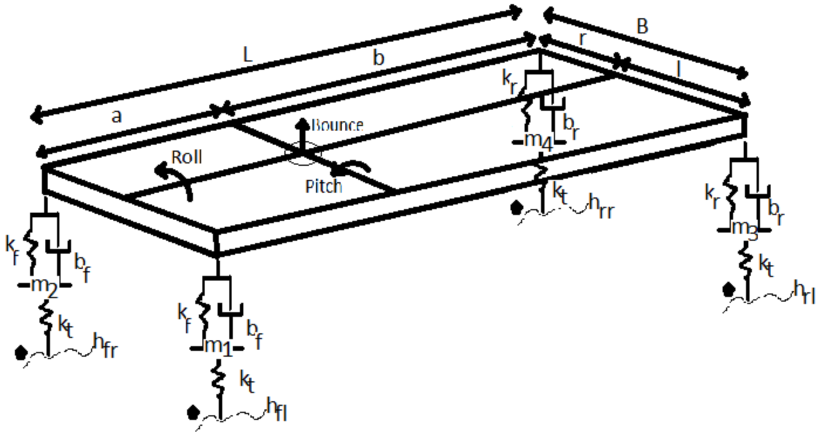


Figure 5.1 The dynamics of the seven degree of freedom vehicle model

The parameters m_1, m_2, m_3, m_4 are the mass values of the four unsprung masses. The parameter k_t is the tire stiffness value. The parameters a, b, r, l, L, B are the geometric properties of the sprung mass (a is the distance from the front axle to the center of mass of the sprung mass, b is the distance from the rear axle to the center of gravity of the sprung mass, r is the distance from the right side of the vehicle to the center of gravity of the sprung mass, l is the distance from the left side of the vehicle to the center of gravity of the sprung mass, L is the wheelbase and B is the trackwidth of the vehicle). The parameters k_f, b_f, k_r, b_r are the stiffness and damping of the front and rear wheels.

The modified model of the seven DOF system uses the four vertical acceleration motions of the wheels as inputs. This reduces the computational complexity, as well as removes the need for knowledge about the unsprung masses' stiffnesses, weights, and damping and knowledge about the terrain profile. Thus, the modified model, illustrated in **Figure 5.2**, has three DOF: vertical bounce, pitch rotation and roll rotation of the sprung mass.

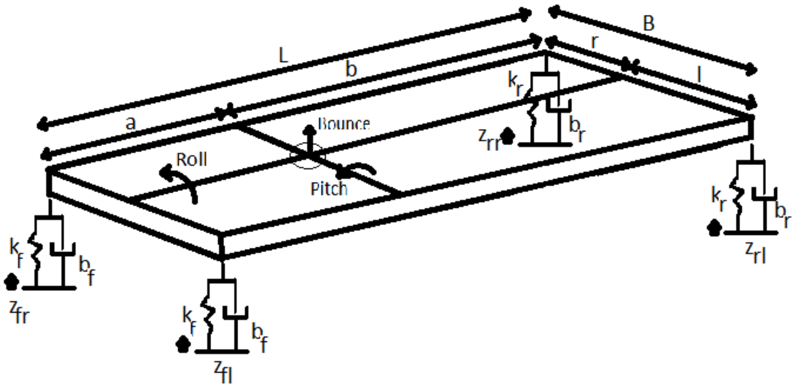


Figure 5.2 The dynamics of the three degree of freedom model

This model has been developed based on the following assumptions: small lateral velocity, small yaw velocity, small longitudinal acceleration, small lateral acceleration, small roll angle, small pitch angle, linear suspension elements, symmetry of front suspension elements, as well as rear elements: $k_{fr} = k_{fl} = k_f$.

The system uses two “centers”, one for the sprung mass, and one for the ensemble of the unsprung masses. For the sprung mass, the center is defined as the height, pitch, and roll of the center of mass. For the unsprung masses, the center is defined as the geometric average height, $z_{u,cg}$, roll, $\theta_{u,cg}$, and pitch, $\phi_{u,cg}$, for each body, making this an adaptation of a quarter car model; the center for the ensemble of the unsprung masses in vertical bounce, pitch, and roll are thus described as:

$$z_{u,cg} = (L-a)\frac{B-l}{L\,B}z_{fl} + (L-a)\frac{l}{L\,B}z_{fr} + a\frac{B-l}{L\,B}z_{rl} + a\frac{l}{L\,B}z_{rr} \tag{86}$$

$$\theta_{u,cg} = \frac{\left[-\left(r\,z_{fl} + l\,z_{fr}\right) + \left(r\,z_{rl} + l\,z_{rr}\right)\right]}{L\,B} \tag{87}$$

$$\phi_{u,cg} = \frac{\left[\left(b\,z_{fl} + a\,z_{rl}\right) - \left(a\,z_{rr} + b\,z_{fr}\right)\right]}{L\,B} \tag{88}$$

Similar relations to Equations (86), (87), and (88) can be written in terms of accelerations, corresponding directly to the measurement done using accelerometers on the instrumented vehicle. The accelerations thus computed are then fed into Equations (89), (90), and (91). (The parameters $z_{fl}, z_{fr}, z_{rl}, z_{rr}$ are the vertical displacements of the wheels (front left, front right, real left, rear right).)

Using the parameters to be estimated m_s, J_{pitch}, J_{roll} as the mass, pitch inertia, and roll inertia of the sprung mass, the dynamic equations of motion of the sprung mass are defined as:

$$m_s \ddot{Z} = \left(\sum_{i=fl,fr,rl,rr} F_i \right) - \ddot{z}_{u,cg} \tag{89}$$

$$J_{pitch} \ddot{\theta} = T_{pitch} - J_{pitch} \ddot{\theta}_{u,cg} \tag{90}$$

$$J_{roll} \ddot{\phi} = T_{roll} - J_{roll} \ddot{\phi}_{u,cg} \tag{91}$$

Where the relative displacements in vertical bounce (Z), pitch (θ) and roll (ϕ) between the ‘centers’ of the unsprung and sprung mass bodies are:

$$Z = z_{s,cg} - z_{u,cg} \tag{92}$$

$$\theta = \theta_{s,cg} - \theta_{u,cg} \tag{93}$$

$$\phi = \phi_{s,cg} - \phi_{u,cg} \tag{94}$$

The forces and moments for the sprung mass system using the relative displacements are:

$$F_{fl} = -k_f\, Z - b_f\, \dot{Z} + a\, k_f\, \theta + a\, b_f\, \dot{\theta} - l\, k_f\, \phi - l\, b_f\, \dot{\phi} \tag{95}$$

$$F_{fr} = -k_f\, Z - b_f\, \dot{Z} + a\, k_f\, \theta + a\, b_f\, \dot{\theta} + r\, k_f\, \phi + r\, b_f\, \dot{\phi} \tag{96}$$

$$F_{rl} = -k_r\, Z - b_r\, \dot{Z} - b\, k_r\, \theta - b\, b_r\, \dot{\theta} - l\, k_r\, \phi - l\, b_r\, \dot{\phi} \tag{97}$$

$$F_{rl} = -k_r\, Z - b_r\, \dot{Z} - b\, k_r\, \theta - b\, b_r\, \dot{\theta} + r\, k_r\, \phi + r\, b_r\, \dot{\phi} \tag{98}$$

$$T_{pitch} = -a\left(F_{fl} + F_{fr}\right) + b(F_{rl} + F_{rr}) \tag{99}$$

$$T_{roll} = -r\left(F_{fr} + F_{rr}\right) + l\left(F_{fl} + F_{rl}\right) \tag{100}$$

5.1.1 Parameter Estimation with the Accelerometer Based Model

Eight experiments are detailed below in Table 5.1and Table 5.2. For each of the estimation experiments, several of the parameters are changed. These are listed below as the initial estimation of the mass, pitch inertia and roll inertia mean values, the polynomial order (Poly Order) of the gPC expansions, the length of each time interval used for estimation and the number of estimations performed.

Table 5.1 Initial Parameters Used for the Bayesian MAP Estimation Algorithm

Run	Mass (kg)	Pitch Inertia ($kg.m^2$)	Roll Inertia ($kg.m^2$)	Poly Order	Time Interval (s)	# of Intervals
1	2250	3500	1000	6	24	1
2	2250	3500	1000	6	1	24
3	2250	3500	1000	4	24	1
4	2250	3500	1000	4	1	24

5	1650	2600	1000	6	24	1
6	2250	3500	1000	4	6	4
7	2250	3500	1000	4	60	1
8	2250	3500	1000	10	24	1

Table 5.2 details the results of the estimation algorithm. The table details what the final estimates are, and what their percent error is relative to the actual values of the synthetic data model.

Table 5.2 Results of the Bayesian Simulations

Run	Mass Est	Pitch Est	Roll Est	% Err Mass	% Err Pitch	% Err Roll
1	1905.685	3054.9	785.44	3.01%	1.83%	-1.82%
2	1897.915	3017.19	823.28	2.59%	0.57%	2.91%
3	1934.36	3159	780.08	4.56%	5.3%	-2.49%
4	1852.405	2877	833.44	0.13%	-4.1%	4.18%
5	1914	3069.4	766.7	3.46%	2.31%	-4.17%
6	1897.9	3048.3	858.2	2.59%	2.61%	7.28%
7	1802.5	2876.4	803.4	-1.59%	-4.12%	0.43%
8	1874.2	3047.3	777.7	1.31%	1.58%	-2.78%

The higher the polynomial order, the more accurate are the estimations. This is consistent with the proposed behavior of the gPC mathematics. The longer the time sequence fed into the

Bayesian estimation algorithm, the better the estimation, which is consistent with statistical theory.

5.1.2 Theoretical Problems with Accelerometer Based Model

This model has its parameters estimated in a purely synthetic environment. As discussed previously this model is a poor choice without special control laws employed to limit the parameter updates, as it does not encapsulate many effects that impact a vehicle during regular driving.

The point of this model was to devise a method by which some of the parameters could be estimated through only accelerometers. Research into this model was abandoned for two reasons. The first reason is that the model requires known suspension characteristics, and the test vehicle has measured forces but unknown suspension characteristics and dynamics. The second reason is that the model is unable to capture steady-state turning or accelerating, which would prevent the estimation of the vehicle's CG height.

When the model is subjected to a steady-state cornering force there is a transient response where the tires have a vertical acceleration. However, their steady-state value will only reflect gravity. This means that the lateral accelerometer is reading a non-zero value while there is no countering force from the suspension, since those accelerometers don't have a response. The model subsequently drifts and the parameter values diverge.

6 Second Attempt at Vehicle Parameter Estimation: Roll Plane, Pitch Plane, and Load Transfer Model

The next attempt was made by attempting to advance the previous model by using the suspension forces as an input, rather than doing the transform to use accelerations, since the vehicle data included the suspension forces. The thought was to make it work, first, and then try to refine it to reduce the required sensors or vehicle information.

The previous model operated in the bounce, pitch and roll degrees of freedom. In that light, the second attempt tried to analyze those degrees of freedom and to estimate those parameters.

6.1 Roll Plane Model

The model for this section is a classical model that also shows poor performance when applied to a set of experimental data.

The roll plane model is designed to look at the roll motion of the vehicle. A diagram of this model is illustrated in **Figure 6.1**.

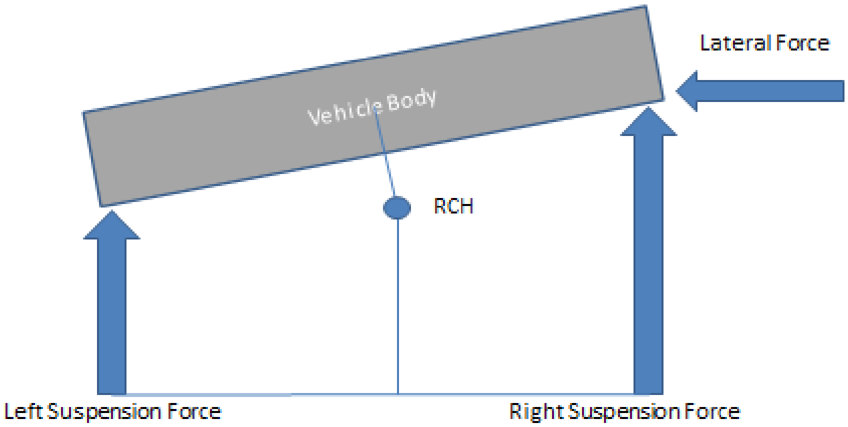


Figure 6.1 Simple Roll Plane Diagram

The simplified equations of motion for this model are demonstrated when using the gPC-EKF for regression systems in section 3.6. These equations however are very simplistic and do not model the experimental data well. Modifying the equations to encapsulate more effects of the vehicle, the equations of motion for the roll plane model are:

$$J_\phi \ddot{\phi} = \left(F_{fl}\frac{B}{2} - F_{fr}\frac{B}{2} + F_{rl}\frac{B}{2} - F_{rr}\frac{B}{2} - mh_r A_y A_z - mh_r A_x A_z \left(-l + \frac{B}{2}\right) + \left(-l + \frac{B}{2}\right) mA_z - (\phi h_r m A_z) \right) \tag{101}$$

The suspension forces $(F_{fl}, F_{fr}, F_{rl}, F_{rr})$ are determined from sensors, A_x, A_y are measured in g's, and A_z is measured in $\frac{m}{s^2}$. The mass, m, is known from estimation, as is the roll center height to the CG, h_r. The CG's lateral position, l, is also derived from the estimation. The track width of the vehicle is the parameter B. The roll inertia of the body is determined afterwards and set to $1200\ kg\ m^2$. This is a somewhat arbitrary choice, chosen from looking at the batch results, but since this parameter only serves to scale the response, it will be clear that the choice can be arbitrary without affecting the dynamics in a time dependent manner. The rotation of the body, ϕ, is derived from the suspension displacements. This term has been found to add significant enhancement to the load transfer models in terms of their ability to match the suspension forces during large displacements. The roll acceleration is derived from several vertical accelerometers as:

$$\ddot{\phi} = \frac{A_{z,fl} + A_{z,rl} - A_{z,fr} - A_{z,rr}}{B} \tag{102}$$

The terms $A_{z,i}$ $(i = fl, fr, rr, rl)$ are the vertical accelerations at each corner of the vehicle (Front left, Front right, Rear right, Rear left). This does not recreate the roll acceleration exactly, but is a decent approximation. **Figure 6.2** shows the results when the Roll Plane model is applied to the experimental data.

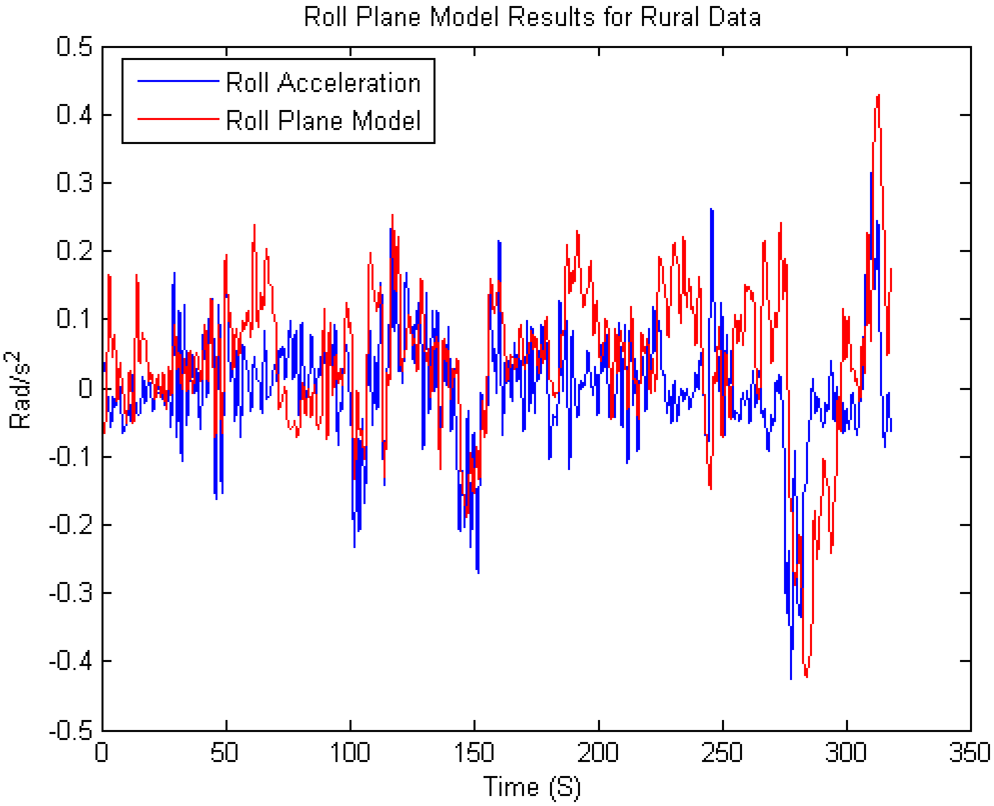


Figure 6.2 Roll Plane Vehicle Model of the Rural Data

There are several reasons why this model does not adequately model the data. First, the vehicle has linkages that restrict the roll motion, and second, the friction effects supply discontinuous forces. The model is very sensitive to the friction effects, which show up strongly during large excitations. For small excitations the model tends to capture the dynamics better, especially when the panhard rod effects are minimal, as seen for $t \in [100\ 150]$.

6.2 Pitch Plane Vehicle Model

Like the Roll Plane model, the Pitch Plane model looks at the pitch motion of the vehicle. This model shows similar performance to the Roll Plane model. The equations of motion for the pitch plane are defined as:

$$J_\theta \ddot{\theta} = \left(F_{fl}a + F_{fr}a - F_{rl}b - F_{rr}b - mh_pA_xA_z - mh_pA_yA_z\left(-l + \frac{L}{2}\right) - \left(\theta h_pmA_z\right)\right) \tag{103}$$

A similar sensor fusion used to estimate the roll acceleration is employed here:

$$\ddot{\theta} = \frac{A_{z,fl} - A_{z,rl} + A_{z,fr} - A_{z,rr}}{L} \tag{104}$$

where L is the wheelbase of the vehicle. The pitch inertia is approximated as $4000\ kg.m^2$, the unloaded inertia of the vehicle is measured at approximately $2900\ kg.m^2$ The model is applied to the rural data set, and the results are shown in **Figure 6.3**.

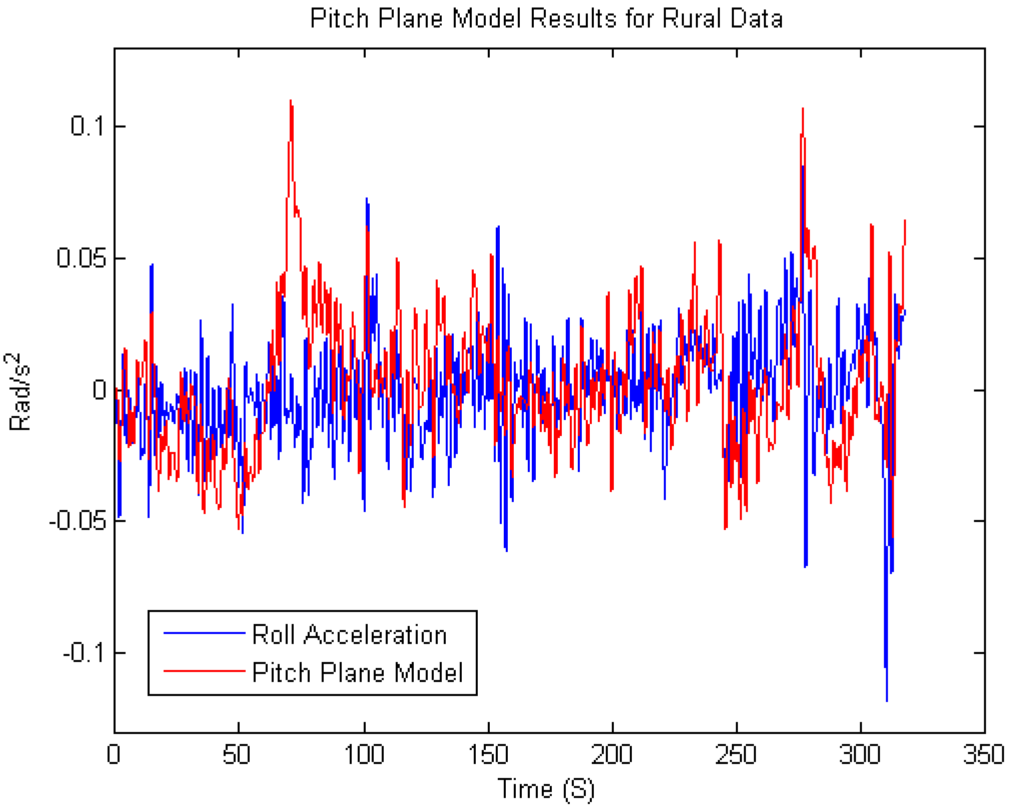


Figure 6.3 Pitch Plane Vehicle Model Applied to the Rural Data

Like the Roll Plane model, the Pitch Plane model is highly sensitive to the effects of the suspension friction. During hard braking at 65 seconds, the suspension friction causes the major spike that is seen, and similarly at the other points of mismatch. This model is also employed without an aerodynamic drag term, and those errors are also present.

For the roll and pitch plane models, long term data collection and batch processing can result in approximate values for the roll or pitch inertias of the vehicle. The suspension friction effects do tend to average out over long enough data sets, but recursive estimations are unlikely to produce satisfactory results. Integration of the vehicle’s kinematics would increase the performance, but unless the suspension friction effects can be accounted for, those models are too sensitive to that effect to be of much use.

6.3 Load Transfer Model (LTM)

The most basic building block used for these models is the concept of load transfer. 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 [40, 46]. This occurs purely because of the inertial forces applied to the vehicle, and is not an actual transfer of mass.

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

$$\Delta W = \frac{m\, A_z\, A_x\, h_r}{B} \tag{105}$$

Where m is the mass of the vehicle, A_z is the vertical acceleration that the vehicle feels, in $\frac{m}{s^2}$, A_x is the centrifugal acceleration in g's, h_r is the distance that the CG 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.

One of the major advantages of this technique is that components (load transfer terms) can easily be added or removed. The basic component is the lateral or longitudinal load transfer that occurs because of cornering or longitudinal accelerations. There are many other building blocks that can be used. These building blocks are other dynamical effects, such as aerodynamic forces, roll or pitch rotation effects, inertia effects. Integration with kinematic effects can be added if their resulting forces are known, such as from a multi-body dynamics model. A diagram of the simple load transfer model is shown in **Figure 6.4**.

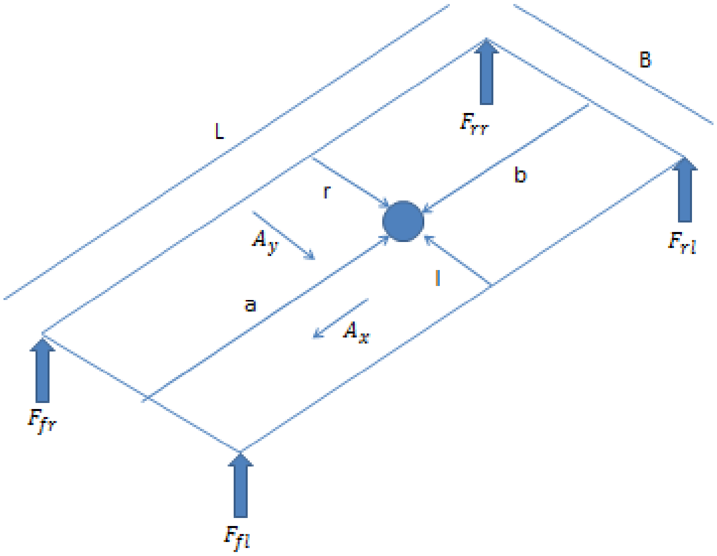


Figure 6.4 Vehicle LTM Diagram

The load transfer and resulting forces at each corner are:

$$F_{fl} = \frac{mA_z}{BL}\left[br - A_y b h_r - A_x r h_p\right] - \frac{r h_p V_x^2 C}{BL} - \frac{m b h_r A_z \phi}{LB} - \frac{m r h_p A_z \theta}{LB} \tag{106}$$

$$F_{fr} = \frac{mA_z}{BL}\left[bl + A_y b h_r - A_x l h_p\right] - \frac{l h_p V_x^2 C}{BL} + \frac{m b h_r A_z \phi}{LB} - \frac{m l h_p A_z \theta}{LB} \tag{107}$$

$$F_{rl} = \frac{mA_z}{BL}\left[ar - A_y a h_r + A_x r h_p\right] + \frac{r h_p V_x^2 C}{BL} - \frac{m a h_r A_z \phi}{LB} + \frac{m r h_p A_z \theta}{LB} \tag{108}$$

$$F_{rr} = \frac{mA_z}{BL}\left[al + A_y a h_r + A_x l h_p\right] + \frac{l h_p V_x^2 C}{BL} + \frac{m a h_r A_z \phi}{LB} + \frac{m l h_p A_z \theta}{LB} \tag{109}$$

$$F_{bounce} = mA_z \tag{110}$$

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 the aerodynamic drag coefficient.

6.3.1 Load Transfer Simulation and Results

The state vector, including uncertain parameters, for the force-based vehicle dynamics model is expressed as:

$$\boldsymbol{x} = \left[F_{fl}, F_{fr}, F_{rl}, F_{rr}, F_{bounce}; \boldsymbol{p}^T\right]^T \tag{111}$$

where

$$\boldsymbol{p} = \left[m, a, l, h_r, h_p, C\right]^T \tag{112}$$

Specifying the uncertain parameters domains:

$$\begin{aligned} m &= \sum_j m^j \psi_1^j(\xi_1) & a &= \sum_j a^j\ \psi_2^j(\xi_2) \\ l &= \sum_j l^j\ \psi_3^j(\xi_3) & h_r &= \sum_j h_r^j \psi_5^j(\xi_5) \\ h_p &= \sum_j h_p^j \psi_6^j(\xi_6) & C &= \sum_j C^j \psi_7^j(\xi_7) \end{aligned} \tag{113}$$

where

$$m^1 = 2000\ kg, \quad m^2 = 100\ kg, \quad m^{3\dots\infty} = 0 \quad a^1 = 2\ m, \quad a^2 = .2\ m, \quad a^{3\dots\infty} = 0 \tag{114}$$

$$l^1 = .8\ m, \quad l^2 = .1\ m, \quad l^{3\dots\infty} = 0 \quad h_r^1 = 1\ m, \quad h_r^2 = .5\ m, \quad h_r^{3\dots\infty} = 0$$

$h_p^1 = 1\,m, \qquad h_p^2 = .5\,m, \qquad h_p^{3\ldots\infty} = 0 \qquad\qquad C^1 = 1, \qquad C^2 = .5, \qquad C^{3\ldots\infty} = 0$

At each time step, t, the system is iterated Q times, for index i.

$$\boldsymbol{X_k^i} = [F_{fl,k}^i, F_{fr,k}^i, F_{rl,k}^i, F_{rr,k}^i, F_{bounce,k}^i, \tag{115}$$

$$m_k^i, a_k^i, l_k^i, h_{r,k}^i, h_{p,k}^i, C_k^i]^T$$

$$for\ i = 1 \ldots Q$$

The observation matrix at time *t* is defined as:

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{116}$$

The choice of the parameter distributions (mean and variance) are chosen based on a rough idea of what the vehicle mass is (though it doesn’t matter if the value is close or not), and the variance should be smaller than the mean, but its rather trivial what the value is after that. The higher order terms are set to zero because it just adds unnecessary complications if they are non-zero.

Table 6.1 Parameter Values for the gPC-EKF LTM Model

Poly Order	1
Q	1
Time Step (S)	0.001

c	0
c_1 (mass)	1e-7
c_2 (a)	1e-9
c_3 (l)	1e-9
c_4 (h_r)	1e-9
c_5 (h_p)	1e-9
c_6 (C)	1e-10
R(FL)	10
R(FR)	10
R(RL)	10
R(RR)	10
R(Bounce)	100

The LTM model has several flaws. Some of them were known a priori, and the rest of them were found later on when analyzing more complicated data, or looking at the post processing. The front left suspension force sensor value and model estimation are shown in **Figure 6.5**.

The LTM model, however, was the first attempt to accurately recreate the parameters in a recursive manner. The advantage of this is that in spite of its flaws, it can still estimate the horizontal CG location, and the mass of the vehicle. The advances from here are used to more accurately estimate the suspension forces and to model the inertia and suspension linkage effects. The description here will appear linear, but only for the sake of continuity.

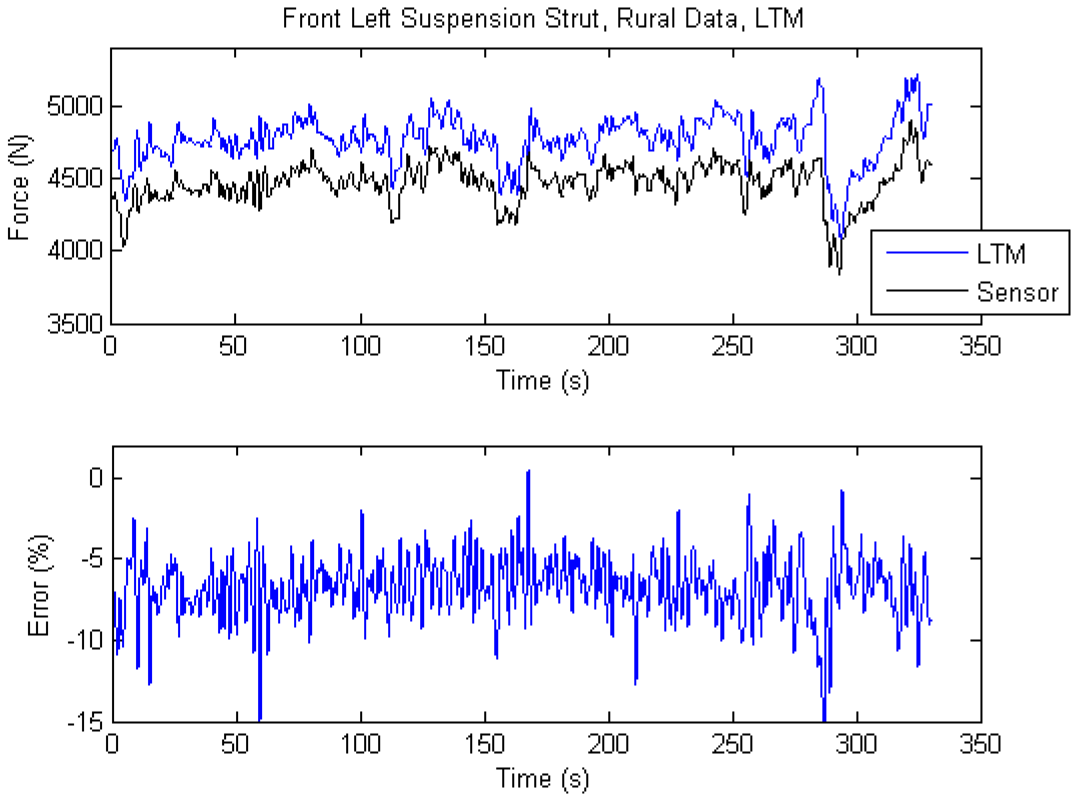


Figure 6.5 LTM Results for the Front Left Suspension

The constant bias of the LTM model is caused by the lumped mass assumption of the model. Which is validated by the weights recorded when scales are placed under the wheels. The gPC-EKF minimizes the cost function across the set of suspension forces.

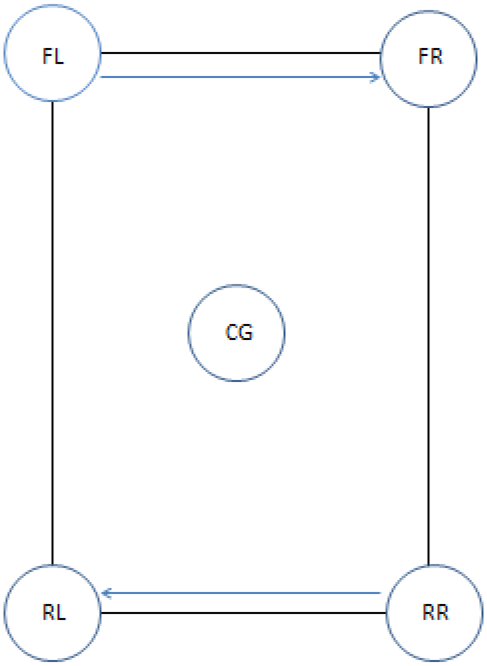


Figure 6.6 Fallacy of the Lumped Mass CG Assumption

Figure 6.6 shows the problem with the lumped mass assumption. If an amount of mass “X” is moved from the front left to the front right, and the same amount of mass “X” is moved from the rear right to the rear left of the vehicle, the CG location will remain unchanged, but the resulting suspension forces will not be the same. The LTM model does not account for this effect, and therefore has a constant bias.

The second problem with the LTM model is that it is implemented with a single parameter for the roll and pitch centers. When analyzing the data for various suspension stiffnesses it became clear that for the soft suspension, the roll and pitch centers for the front and the rear were about the same. When the suspension became stiffer, the roll centers for the front and rear split, with the front moving downward and the rear moving upward. Using a single roll center or pitch center became very disadvantageous.

The final problem with this implementation is that the roll and pitch centers are time varying quantities. Not because the roll or pitch centers move themselves, but because the suspension force values that are used do not include the linkage forces. This implementation was performed with a very slow update rate, which likely averages out where the roll or pitch center actually is, but does not match the suspension force estimate very well. The primary example of this is that when the vehicle breaks very quickly, the suspension linkages cause a jacking force in the front that gives the vehicle an anti-dive behavior, which the suspension force values reflect, but the accelerometers do not.

6.3.2 Load Transfer Model Parameter Estimations

The LTM Model was run on data for both the Rural and Urban driving scenarios. The vehicle's parameters do not change between the runs, which highlights the effects of the different driving scenarios upon the estimator's ability to estimate the vehicle's parameters. **Figure 6.7** shows the LTM's estimate of the vehicle mass. The LTM's estimate of the horizontal CG position is shown in Figure 6.8, for lateral position, and in **Figure 6.9** , for longitudinal position.

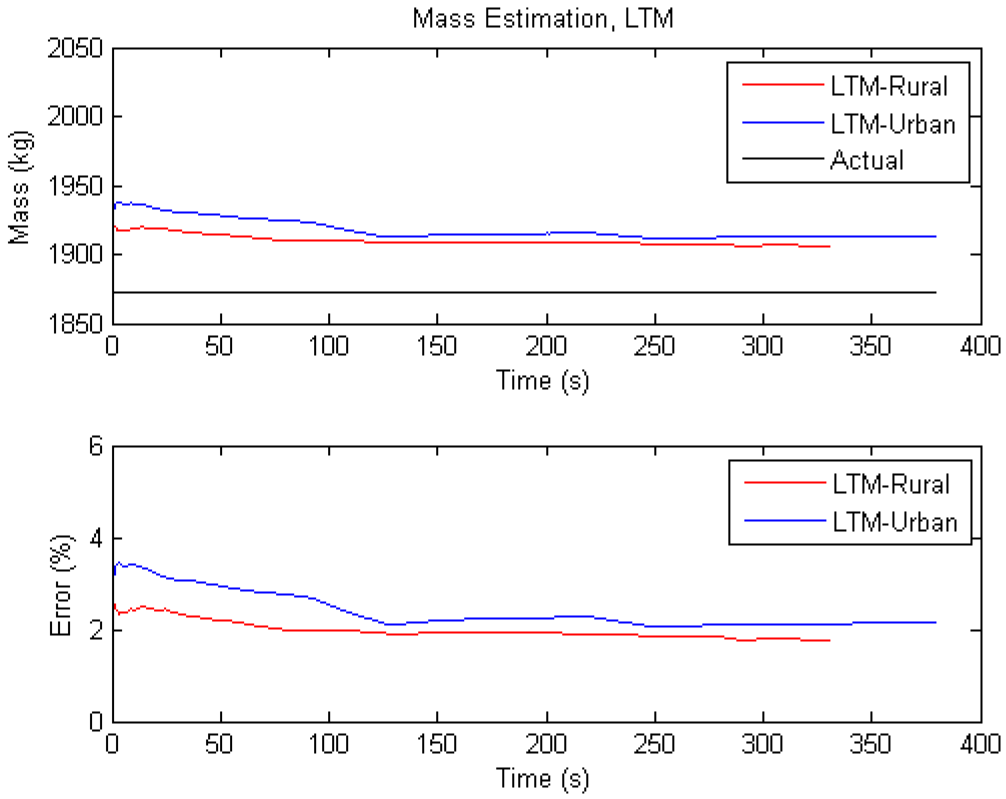


Figure 6.7 Mass Estimation from the LTM Model for the Rural and Urban Data Sets

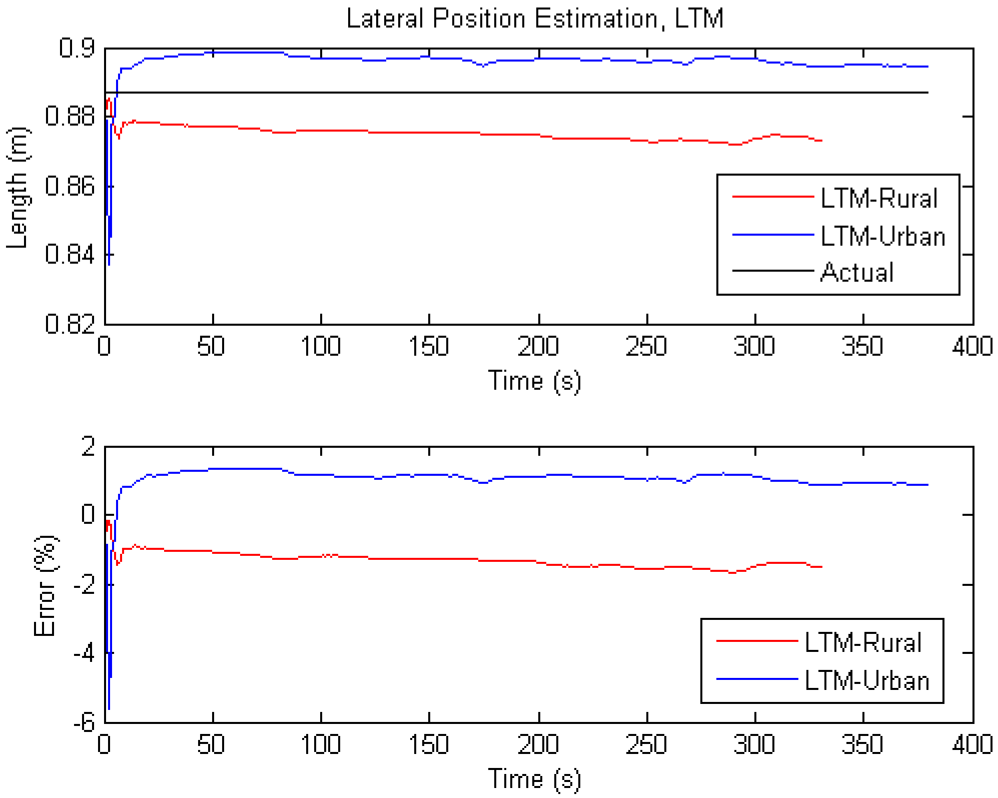


Figure 6.8 LTM Model's Estimate of the Vehicle Lateral Position for the Rural and Urban Data Sets

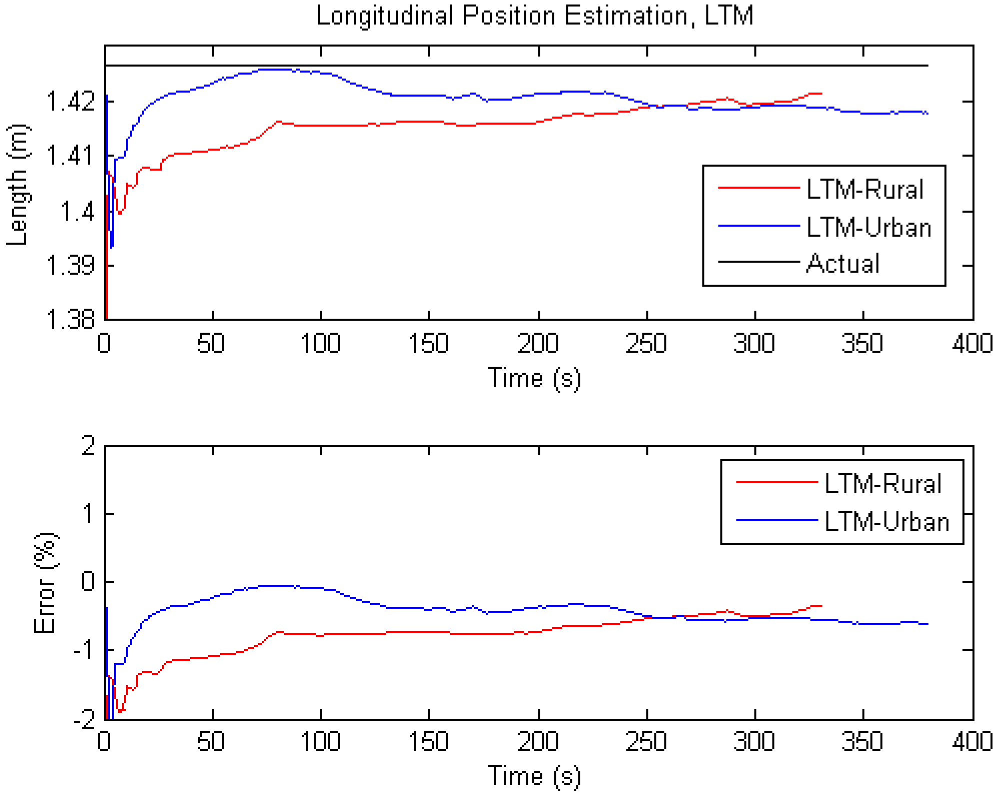


Figure 6.9 LTM Model's Estimate of the Vehicle's Longitudinal Position for the Rural and Urban Data Sets

The LTM model does an excellent job of estimating the parameters. Part of the reason for such accuracy is because it has the sensor values from approximately the CG location of the vehicle. If the sensors were not at the CG of the vehicle (because of loading conditions, etc) it would be given erroneous data and the estimate would not be as accurate. The problem with needing to know where the CG of the vehicle is, is one of the major reasons for the further development of the estimation method, as shown in subsequent chapters.

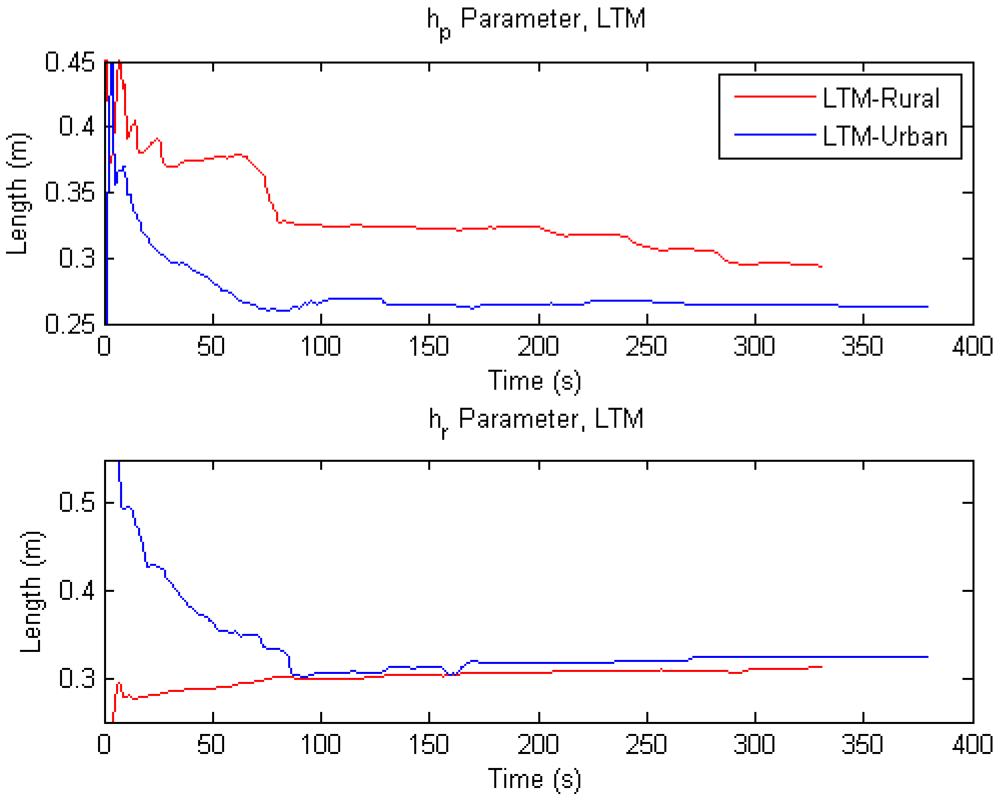


Figure 6.10 h_p and h_r Parameter Values from the LTM Model for the Rural and Urban Data Sets

For this simulation, the LTM model has a slow update for the parameter values, which is why the h_p and h_r values, shown in **Figure 6.10**, appear to be converging to a value. They are in fact converging to an average value, which corresponds to a reasonable estimate of the distance between the CG and the roll center for when the suspension is in its static equilibrium position.

7 Third Attempt at Vehicle Parameter Estimation: Modified Load Transfer Model

The modified load transfer model (MLTM) was developed primarily to compensate for the LTM's constant bias in the suspension force estimates, and to remove the need to know where the CG is a priori for sensor placements. This model constitutes the third attempt at estimating the vehicle parameters. The MLTM model shown here was performed with a slow update for the roll and pitch centers, and still uses a single roll and pitch center parameter.

7.1 Modified Load Transfer Model (MLTM)

Instead of using the CG of the vehicle to calculate the load transfer due to the horizontal accelerations, the Modified Load Transfer Model (MLTM) utilizes accelerometers placed at the suspension struts. In practice this is a more reasonable method of estimation as the acceleration at the CG is generally not known. For an unknown center of gravity, the measured CG vertical acceleration can be approximated as:

$$\ddot{Z} = \frac{1}{4}\left[A_{z,fl} + A_{z,fr} + A_{z,rl} + A_{z,rr} + 2(r - l)\ddot{\phi} + 2(b - a)\ddot{\theta}\right] \tag{117}$$

This model returns an estimate of the mass at each corner of the vehicle, which can be used to estimate 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 as follows:

$$F_{fl} = m_{fl}A_{z,fl} - \frac{\bar{m}\ddot{Z}}{BL}A_y bh_r - \frac{\bar{m}\ddot{Z}}{BL}A_x rh_p - \frac{rh_pV_x^2C}{BL} - \frac{\bar{m}bh_r\ddot{Z}\phi}{LB} - \frac{\bar{m}rh_p\ddot{Z}\theta}{LB} \tag{118}$$

$$F_{fr} = m_{fr}A_{z,fr} + \frac{\bar{m}\ddot{Z}}{BL}A_y bh_r - \frac{\bar{m}\ddot{Z}}{BL}A_x lh_p - \frac{lh_pV_x^2C}{BL} + \frac{\bar{m}bh_r\ddot{Z}\phi}{LB} - \frac{\bar{m}lh_p\ddot{Z}\theta}{LB} \tag{119}$$

$$F_{rl} = m_{rl}A_{z,rl} - \frac{\bar{m}\ddot{Z}}{BL}A_y ah_r + \frac{\bar{m}\ddot{Z}}{BL}A_x rh_p + \frac{rh_pV_x^2C}{BL} - \frac{\bar{m}ah_r\ddot{Z}\phi}{LB} + \frac{\bar{m}rh_p\ddot{Z}\theta}{LB} \tag{120}$$

$$F_{rr} = m_{rr}A_{z,rr} + \frac{\bar{m}\ddot{Z}}{BL}A_y ah_r + \frac{\bar{m}\ddot{Z}}{BL}A_x lh_p + \frac{lh_pV_x^2C}{BL} + \frac{\bar{m}ah_r\ddot{Z}\phi}{LB} + \frac{\bar{m}lh_p\ddot{Z}\theta}{LB} \tag{121}$$

$$F_{bounce} = F_{fl} + F_{fr} + F_{rl} + F_{rr} = m_{fl}A_{z,fl} + m_{fr}A_{z,fr} + m_{rl}A_{z,rl} + m_{rr}A_{z,rr} \tag{122}$$

With the total bounce of the vehicle calculated as:

$$F_{bounce} = F_{fl} + F_{fr} + F_{rl} + F_{rr} = m_{fl}A_{z,fl} + m_{fr}A_{z,fr} + m_{rl}A_{z,rl} + m_{rr}A_{z,rr} \tag{123}$$

Here the estimates for $\bar{m}$ are the sum of the estimates for $m_{fl}, m_{fr}, m_{rl}, m_{rr}$, where each of these is the mass estimate for each corner of the vehicle. The number of parameters here have several redundant ones, since the position of the CG can be directly calculated from the four mass estimations above each strut.

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

$$a = \frac{m_{rl} + m_{rr}}{\bar{m}}L \tag{124}$$

$$l = \frac{m_{fr} + m_{rr}}{\bar{m}} B \tag{125}$$

The state vector, including uncertain parameters, for the force based vehicle dynamics model:

$$\boldsymbol{x} = \left[F_{fl}, F_{fr}, F_{rl}, F_{rr}, F_{bounce}; \boldsymbol{p}^T\right]^T \tag{126}$$

where

$$\boldsymbol{p} = \left[m_{fl}, m_{fr}, m_{rl}, m_{rr}, h_r, h_p, C\right]^T \tag{127}$$

Using the substitutions for *a* and *l* the MLTM equations become:

$$\begin{aligned} F_{fl} = m_{fl}\, A_{z,fl} &- \frac{\left(m_{fl} + m_{fr}\right)\ddot{Z}}{B} A_y h_r - \frac{\left(m_{fl} + m_{rl}\right)\ddot{Z}}{L} A_x h_p \\ &- \frac{h_p V_x^2 C}{L} \left(\frac{m_{fl} + m_{rl}}{m_{fl} + m_{fr} + m_{rl} + m_{rr}}\right) - \frac{\left(m_{fl} + m_{fr}\right) h_r \ddot{Z} \phi}{B} \\ &- \frac{\left(m_{fl} + m_{rl}\right) h_p \ddot{Z} \theta}{L} \end{aligned} \tag{128}$$

$$\begin{aligned} F_{fr} = m_{fr}\, A_{z,fr} &+ \frac{\left(m_{fl} + m_{fr}\right)\ddot{Z}}{B} A_y h_r - \frac{\left(m_{fr} + m_{rr}\right)\ddot{Z}}{L} A_x h_p \\ &- \frac{h_p V_x^2 C}{L} \left(\frac{m_{fr} + m_{rr}}{m_{fl} + m_{fr} + m_{rl} + m_{rr}}\right) + \frac{\left(m_{fl} + m_{fr}\right) h_r \ddot{Z} \phi}{B} \\ &- \frac{\left(m_{fr} + m_{rr}\right) h_p \ddot{Z} \theta}{L} \end{aligned} \tag{129}$$

$$\begin{aligned} F_{rl} = m_{rl}\, A_{z,rl} &- \frac{\left(m_{rl} + m_{rr}\right)\ddot{Z}}{B} A_y h_r + \frac{\left(m_{fl} + m_{rl}\right)\ddot{Z}}{L} A_x h_p \\ &+ \frac{h_p V_x^2 C}{L} \left(\frac{m_{fl} + m_{rl}}{m_{fl} + m_{fr} + m_{rl} + m_{rr}}\right) - \frac{\left(m_{rl} + m_{rr}\right) h_r \ddot{Z} \phi}{B} \\ &+ \frac{\left(m_{fl} + m_{rl}\right) h_p \ddot{Z} \theta}{L} \end{aligned} \tag{130}$$

$$F_{rr} = m_{rr}\,A_{z,rr} + \frac{(m_{rl} + m_{rr})\ddot{Z}}{B}A_y h_r + \frac{(m_{fr} + m_{rr})\ddot{Z}}{L}A_x h_p + \frac{h_p V_x^2 C}{L}\left(\frac{m_{fr} + m_{rr}}{m_{fl} + m_{fr} + m_{rl} + m_{rr}}\right) + \frac{(m_{rl} + m_{rr})h_r\ddot{Z}\phi}{B} + \frac{(m_{fr} + m_{rr})h_p\ddot{Z}\theta}{L} \tag{131}$$

$$F_{bounce} = m_{fl}A_{z,fl} + m_{fr}A_{z,fr} + m_{rl}A_{z,rl} + m_{rr}A_{z,rr} \tag{132}$$

Specifying the uncertain parameters domains:

$$m_{fl} = \sum_j m_{fl}^j \psi_1^j(\xi_1) \qquad m_{fr} = \sum_j m_{fr}^j \psi_2^j(\xi_2) \tag{133}$$

$$m_{rl} = \sum_j m_{rl}^j \psi_3^j(\xi_3) \qquad m_{rr} = \sum_j m_{rr}^j \psi_4^j(\xi_4)$$

$$h_r = \sum_j h_r^j \psi_5^j(\xi_5) \qquad h_p = \sum_j h_p^j \psi_6^j(\xi_6)$$

$$C = \sum_j C^j \psi_7^j(\xi_7)$$

where

$$m_{fl}^1 = 400\ kg, \quad m_{fl}^2 = 100\ kg, \quad m_{fl}^{3\ldots\infty} = 0 \qquad m_{fr}^1 = 400\ kg, \quad m_{fr}^2 = 100\ kg, \quad m_{fr}^{3\ldots\infty} = 0 \tag{134}$$

$$m_{rl}^1 = 400\ kg, \quad m_{rl}^2 = 100\ kg, \quad m_{rl}^{3\ldots\infty} = 0 \qquad m_{rr}^1 = 400\ kg, \quad m_{rr}^2 = 100\ kg, \quad m_{rr}^{3\ldots\infty} = 0$$

$$h_r^1 = 1\ m, \quad h_r^2 = .5\ m, \quad h_r^{3\ldots\infty} = 0 \quad h_p^1 = 1\ m, \quad h_p^2 = .5\ m, \quad h_p^{3\ldots\infty} = 0$$

$$C^1 = 1, \quad C^2 = .5, \quad C^{3\ldots\infty} = 0$$

At each time step, k, the system is iterated Q times, for index i.

$$\boldsymbol{X_k^i} = [F_{fl,k}^i, F_{fr,k}^i, F_{rl,k}^i, F_{rr,k}^i, F_{bounce,k}^i, m_{fl,k}^i, m_{fr,k}^i, m_{rl,k}^i, m_{rr,k}^i, h_{r,k}^i, h_{p,k}^i, C_k^i]^T \quad for\ i = 1 \dots Q \tag{135}$$

The observation matrix at time *t* is defined as:

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{136}$$

Table 7.1 Parameter Values for the gPC-EKF for the MLTM Model

Poly Order	1
Q	1
Time Step (S)	0.001
c	0
$c_1\ (m_{fl})$	1e-7
$c_1\ (m_{fr})$	1e-7
$c_1\ (m_{rl})$	1e-7
$c_1\ (m_{rr})$	1e-7
$c_5\ (h_p)$	1e-9

$c_5\ (h_r)$	1e-9
$c_6\ (C)$	1e-10
R(FL)	10
R(FR)	10
R(RL)	10
R(RR)	10
R(Bounce)	100

7.2 Results for Modified Load Transfer Model

The MLTM is run using the Rural and Urban driving scenarios. The coefficients for the gPC-EKF are presented in **Table 7.1**. The purpose of the modified system was to give better matching of the sensor data. For comparison, the MLTM model is shown in contrast to both the LTM and the Sensor value for the front left suspension strut.

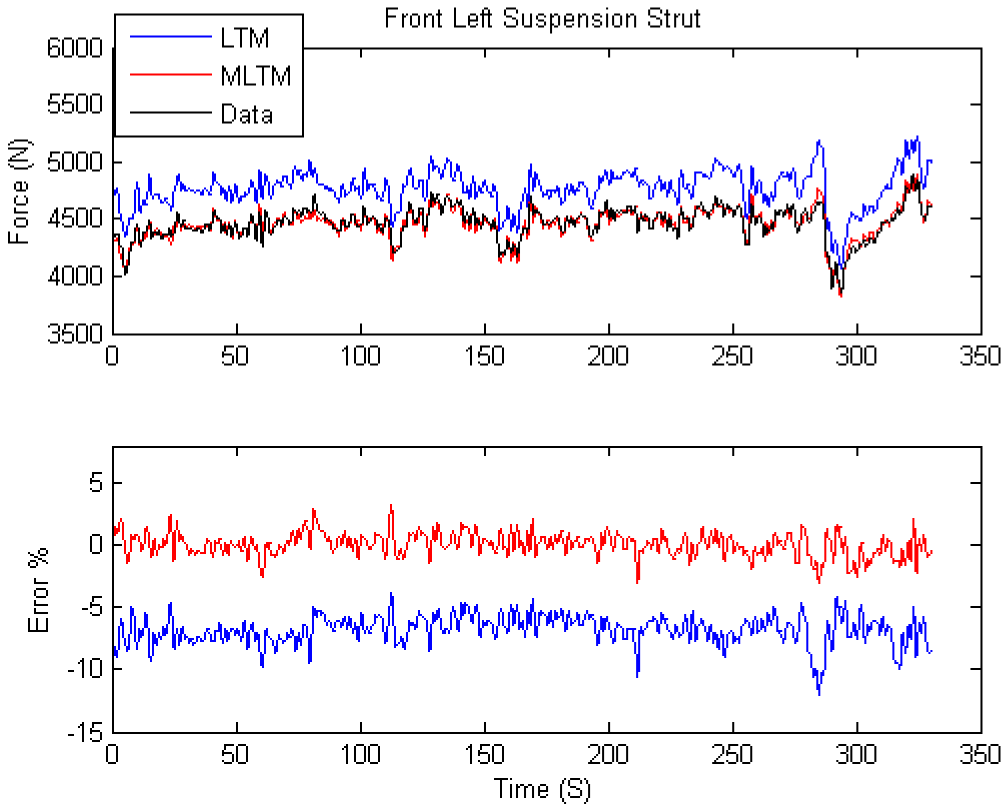


Figure 7.1 Suspension Strut Force for Front Left Suspension for the Rural Data Set, for the MLTM Model Compared to the LTM Model

The MLTM model compensates for the constant bias of the LTM model, as can be seen in **Figure 7.1**. The sensor tends to better estimate the suspension force, primarily for the peak at ~275 seconds.

7.2.1 Parameter Estimations for MLTM Model

Figure 7.2 shows the MLTM's estimate of the vehicle's mass for the Rural and the Urban driving scenarios. The MLTM shows a larger estimate than the LTM for two reasons. The first is because the sensors have biases (DC accelerometers, filtering), and how well each sensor is zeroed will play into the parameter error (talking about 0-2% bias). Second, the MLTM uses accelerometers at each suspension strut, and because the vehicle lacks gyros the accelerometers are not calibrated to reject the vehicle's roll and pitch motions. This is problematic because the philosophy of the load transfer modeling is that the vehicle is a block that has no rotational inertia properties and assumes all load transfers occur due to external or inertial forces.

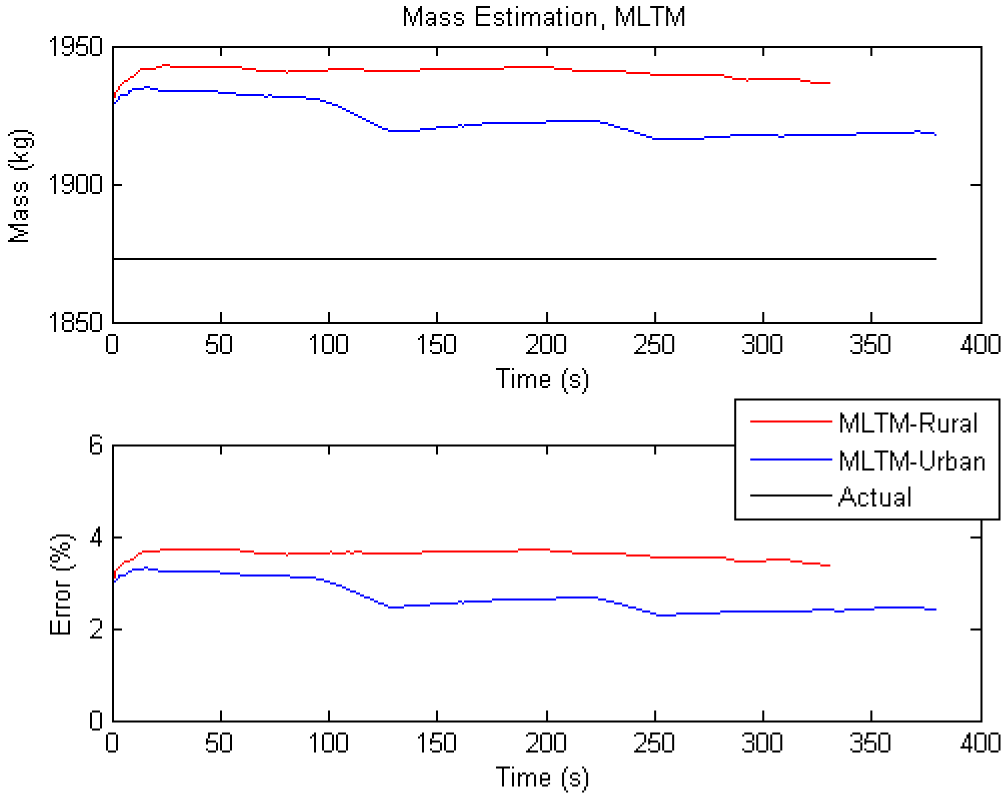


Figure 7.2 Mass Estimation from the MLTM Model for the Rural and Urban Data Sets

The lateral position estimate, shown in **Figure 7.3**, shows what appears to be a drift in the parameters, which is caused by the suspension friction slowly causing an increasing roll angle bias, and is not in fact a drift, as can be seen in **Figure 7.4**. The updating law is slow enough that the large spikes do not make enough of a difference to counter the slowly increasing bias, however their effect can be seen. This asymmetrical friction force in the suspension of the vehicle will appear later in the vehicle tire normal load estimations, and it causes some interesting effects.

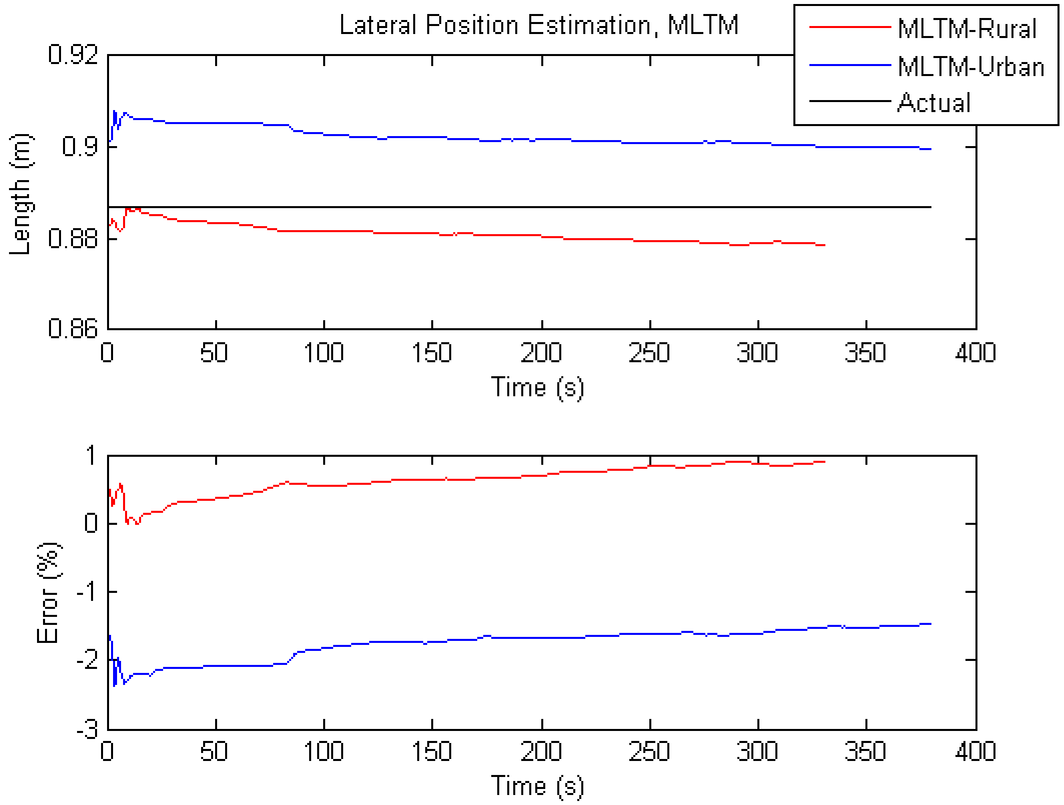


Figure 7.3 Lateral Position Estimate from the MLTM Model for the Rural and Urban Data Sets

The longitudinal parameter estimate, shown in **Figure 7.5**, updates quite nicely, with the MLTM showing a much better approximation of the position than the LTM. This estimation is also much smoother than the lateral position estimate. In general, across several data sets, the longitudinal position converges much better than the lateral position estimate, because the effects of friction are less significant in the pitch dynamics.

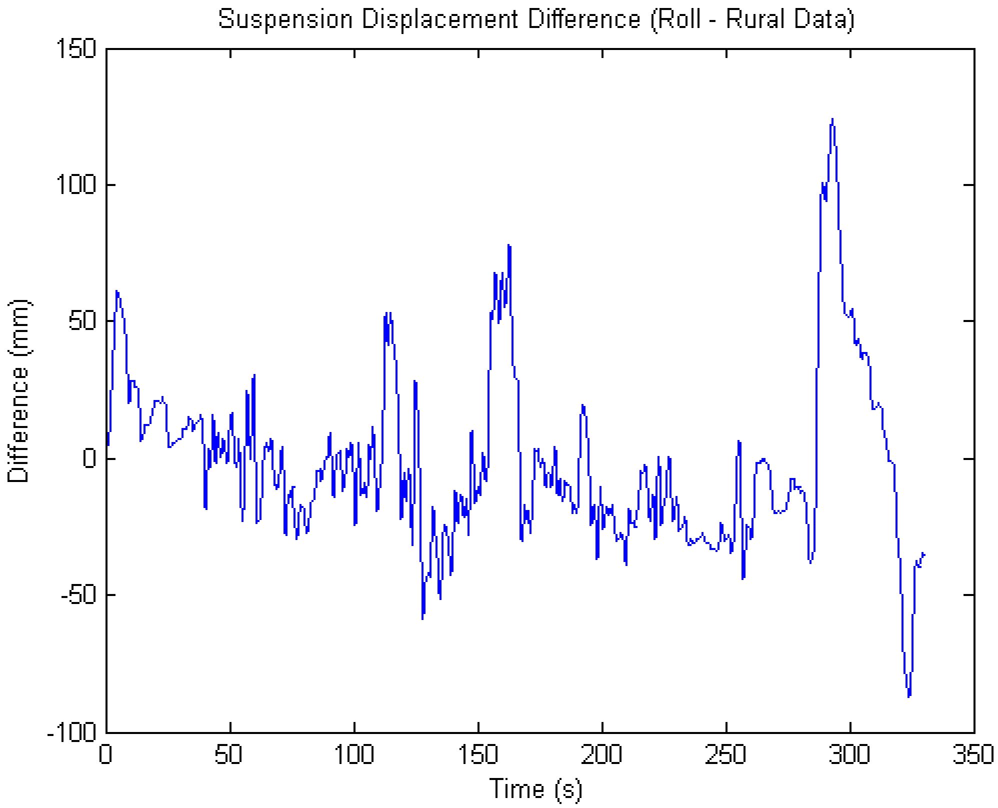


Figure 7.4 Suspension Displacement Difference (Body Roll)

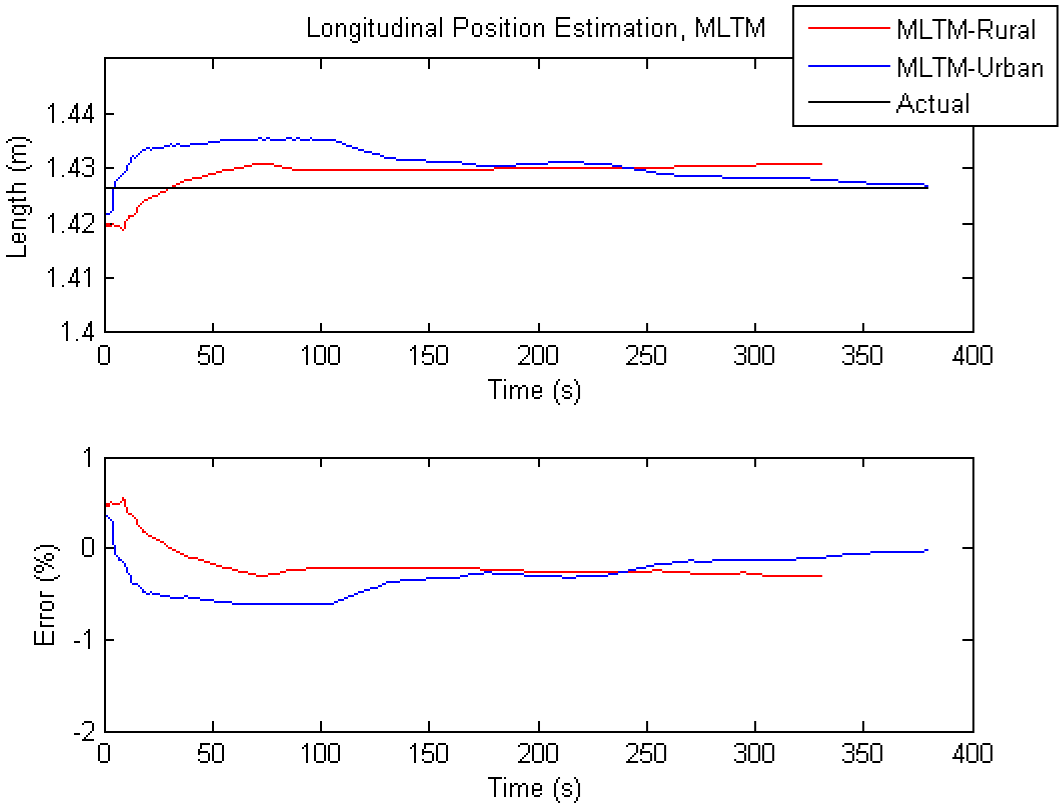


Figure 7.5 Longitudinal Position Estimate of the MLTM Model for the Rural and Urban Data Sets

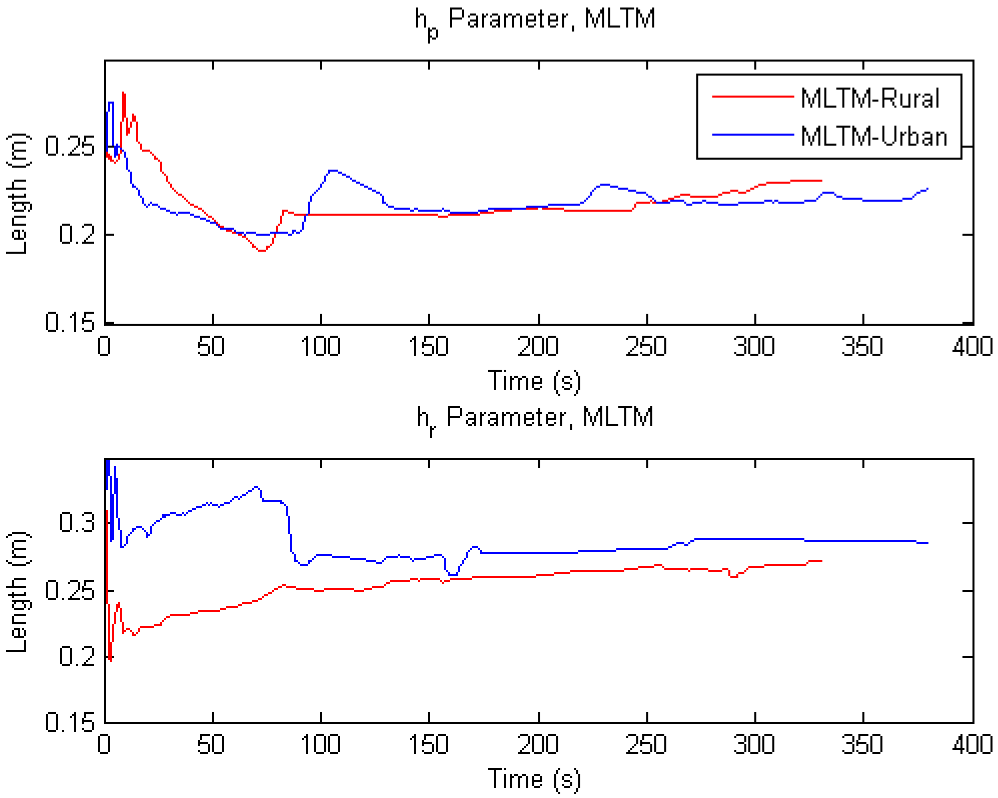


Figure 7.6 h_p and h_r Parameter Values from the MLTM Model for the Rural and Urban Data Sets

As mentioned previously, the MLTM model picks up on the roll and pitch effects of the vehicle because the accelerometers are located at the struts, rather than the CG (or, in reality, the instantaneous centers of the roll and pitch motions). Because of this, the h_r and h_p parameters, shown in **Figure 7.6**, for the MLTM are smaller than for the LTM model.

8 Fourth and Final Parameter Estimation Model: High Frequency Update Modeling of Vehicle's Suspension Forces

This section is the fourth attempt at modeling the vehicle's suspension forces and estimation of the vehicle's parameters. The MLTM model is expanded to include three roll centers, and two pitch centers. The effects of the aerodynamics and the roll and pitch angle effects are removed, as their effects are rolled into the rapid updating of the roll/pitch center parameters. Full implementation should include these effects with slow updating. The update rates for the roll and pitch center parameters are significantly increased, which enables real time compensation for un-modeled effects, and provides better matching of the suspension forces.

The need for three roll centers is due to the linkage effects. The front of the vehicle has a panhard rod that provides lateral stability, which causes asymmetric load transfer effects. The panhard rod provides a direct means of lateral load transfer between the body and the axle in the front, which results in the need for two roll centers, one for each wheel. The third roll center is used in the rear of the vehicle, where the suspension is approximately symmetric. The two pitch centers are obvious, one for the front and one for the rear of the vehicle.

The equations for the High Frequency Load Transfer Model (HFLTM) are:

$$F_{fl} = m_{fl}\,A_{z,fl} - \frac{(m_{fl} + m_{fr})\ddot{Z}}{B} A_y h_{rfl} - \frac{(m_{fl} + m_{rl})\ddot{Z}}{L} A_x h_{pf} \tag{137}$$

$$F_{fr} = m_{fr}\,A_{z,fr} + \frac{(m_{fl} + m_{fr})\ddot{Z}}{B} A_y h_{rfr} - \frac{(m_{fr} + m_{rr})\ddot{Z}}{L} A_x h_{pf} \tag{138}$$

$$F_{rl} = m_{rl}\,A_{z,rl} - \frac{(m_{rl} + m_{rr})\ddot{Z}}{B} A_y h_{rr} + \frac{(m_{fl} + m_{rl})\ddot{Z}}{L} A_x h_{pr} \tag{139}$$

$$F_{rr} = m_{rr}\,A_{z,rr} + \frac{(m_{rl} + m_{rr})\ddot{Z}}{B} A_y h_{rr} + \frac{(m_{fr} + m_{rr})\ddot{Z}}{L} A_x h_{pr} \tag{140}$$

where,

$$m_{fl} = \sum_j m_{fl}^j \psi_1^j(\xi_1) \qquad m_{fr} = \sum_j m_{fr}^j \psi_2^j(\xi_2) \tag{141}$$

$$m_{rl} = \sum_j m_{rl}^j \psi_3^j(\xi_3) \qquad m_{rr} = \sum_j m_{rr}^j \psi_4^j(\xi_4)$$

$$h_{rfl} = \sum_j h_{rfl}^j \psi_5^j(\xi_5) \qquad h_{pf} = \sum_j h_{pf}^j \psi_8^j(\xi_8)$$

$$h_{rfr} = \sum_j h_{rfr}^j \psi_6^j(\xi_6) \qquad h_{pr} = \sum_j h_{pr}^j \psi_9^j(\xi_9)$$

$$h_{rr} = \sum_j h_{rr}^j \psi_7^j(\xi_7)$$

Where

$$m_{fl}^1 = 400\,kg, \quad m_{fl}^2 = 100\,kg, \qquad m_{fr}^1 = 400\,kg, \quad m_{fr}^2 = 100\,kg, \tag{142}$$

$$m_{fl}^{3\ldots\infty} = 0 \qquad m_{fr}^{3\ldots\infty} = 0$$

$$m_{rl}^1 = 400\,kg, \quad m_{rl}^2 = 100\,kg, \qquad m_{rr}^1 = 400\,kg, \quad m_{rr}^2 = 100\,kg,$$

$$m_{rl}^{3\ldots\infty} = 0 \qquad m_{rr}^{3\ldots\infty} = 0$$

$$h^1_{rfr} = 1\,m, \qquad h^2_{rfr} = .5\,m, \qquad h^1_{pf} = 1\,m, \qquad h^2_{pf} = .5\,m,$$

$$h^{3\ldots\infty}_{rfr} = 0 \qquad h^{3\ldots\infty}_{pf} = 0$$

$$h^1_{rfl} = 1\,m, \qquad h^2_{rfl} = .5\,m, \qquad h^1_{pr} = 1\,m, \qquad h^2_{pr} = .5\,m,$$

$$h^{3\ldots\infty}_{rfl} = 0 \qquad h^{3\ldots\infty}_{pr} = 0$$

$$h^1_{rr} = 1\,m, \qquad h^2_{rr} = .5\,m, \qquad h^{3\ldots\infty}_{rr} = 0$$

At each time step, k, the system is iterated Q times, for index i.

$$\boldsymbol{X}^{\boldsymbol{i}}_k = [F^i_{fl,k}, F^i_{fr,k}, F^i_{rl,k}, F^i_{rr,k}, F^i_{bounce,k}, m^i_{fl,k}, m^i_{fr,k},$$

$$m^i_{rl,k}, m^i_{rr,k}, h^i_{rfr,k}, h^i_{rfl,k}, h^i_{rr,k}, h^i_{pf,k}, h^i_{pr,k}]^T \tag{143}$$

$$for\ i = 1 \ldots Q$$

The observation matrix at time *t* is defined as:

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{144}$$

Table 8.1 Parameter Values for the gPC-EKF for the HFLTM Model

Poly Order	1
Q	1

Time Step (S)	0.001
c	0
c_1 (m_{fl})	1e-8
c_1 (m_{fr})	1e-8
c_1 (m_{rl})	1e-8
c_1 (m_{rr})	1e-8
c_5 (h_{pf})	1e-6
c_5 (h_{pr})	1e-6
c_5 (h_{rfl})	1e-6
c_5 (h_{rfr})	1e-6
c_5 (h_{rr})	1e-6
R(FL)	10
R(FR)	10
R(RL)	10
R(RR)	10

8.1 Experimental Results for Dynamic Handling Track

The dynamic handling track data is the first set of data to contain both the soft and the hard suspension settings. The Rural and Urban drive cycles were both done using the soft suspension setting, which is why the need for multiple roll and pitch centers was not observed. For the soft suspension setting, these are relatively close to the same values. The actual values for the vehicle's mass and horizontal CG location are not precisely known for the tests shown in this chapter. Measurements were performed before the tests, but these included the unsprung components. The coefficients for the gPC-EKF for these tests are listed in **Table 8.1**.

The total mass measurement, which was performed with scales, was approximately 2400 *kg*. The total unsprung components weigh about 500 *kg*. The model estimates the vehicle's mass to be about 1940 *kg*, which is about 2% error, though in truth the accuracy is within instrumentation and measurement error ranges.

8.1.1 Dynamic Handling Track with Soft Suspension

The results for the soft suspension are presented first, as the previous attempts to estimate the vehicle's parameters were performed on a soft suspension configuration. The parameters have a fairly large update rate, and the effects of the friction can clearly be seen between the soft and the hard settings. The nonlinear behavior of the panhard rod and the need for multiple roll and pitch parameters is also immediately evident.

The estimation of the suspension force is good, shown in **Figure 8.1**, with large errors primarily only present during low dynamic events. The estimate is able to capture the peaks of the system, and track the model during dynamic maneuvers. The RMS error is 115.85 *N*, with a peak error of 518 *N* occurring at 71.66 seconds.

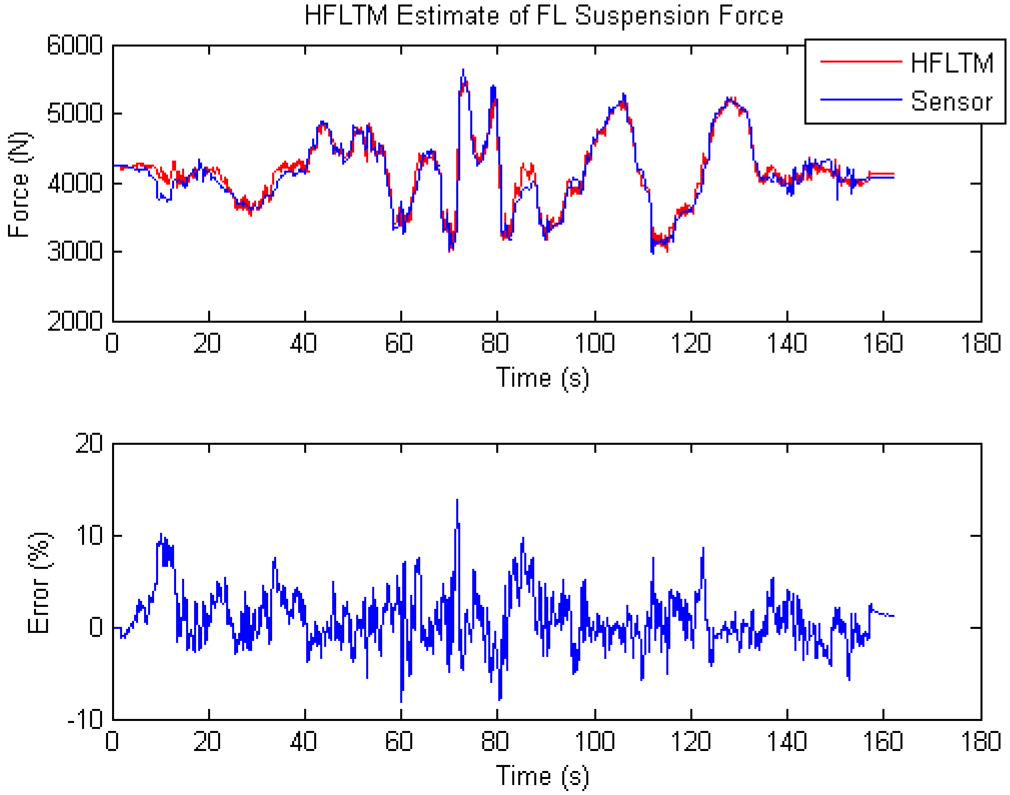


Figure 8.1 Dynamic Handling Track HFLTM FL Suspension Force Estimate for the Soft Suspension Setting

Because of the faster parameter update rates the friction effects can have a more noticeable effect. The longitudinal CG position converges fairly smoothly, but the longitudinal position shifts around a bit, as the vehicle tends to roll much more than pitch. These results are shown in **Figure 8.2**.

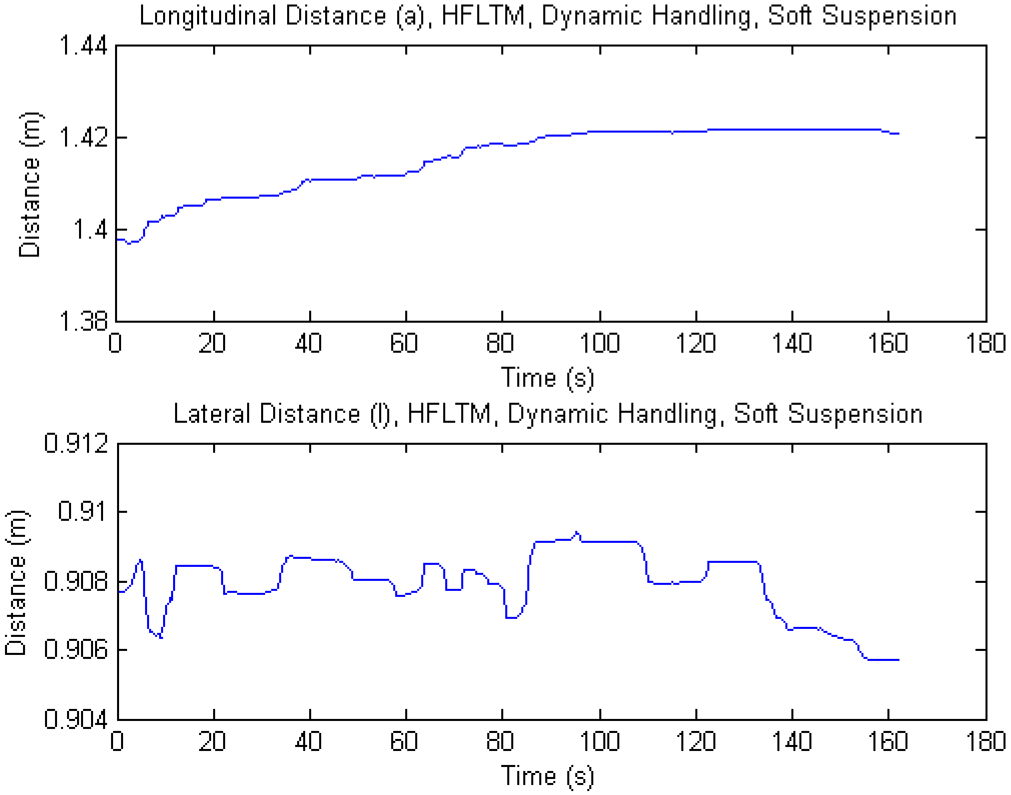


Figure 8.2 Horizontal CG Estimate for the HFLTM on the Dynamic Handling Track Using the Soft Suspension Setting

The mass parameter, results shown in **Figure 8.3**, which is a sum of the masses calculated at each corner, is used to calculate the horizontal CG position and shows why the lateral position is shifting around. The mass parameter is changing very slightly at each corner to help accommodate for the friction effects.

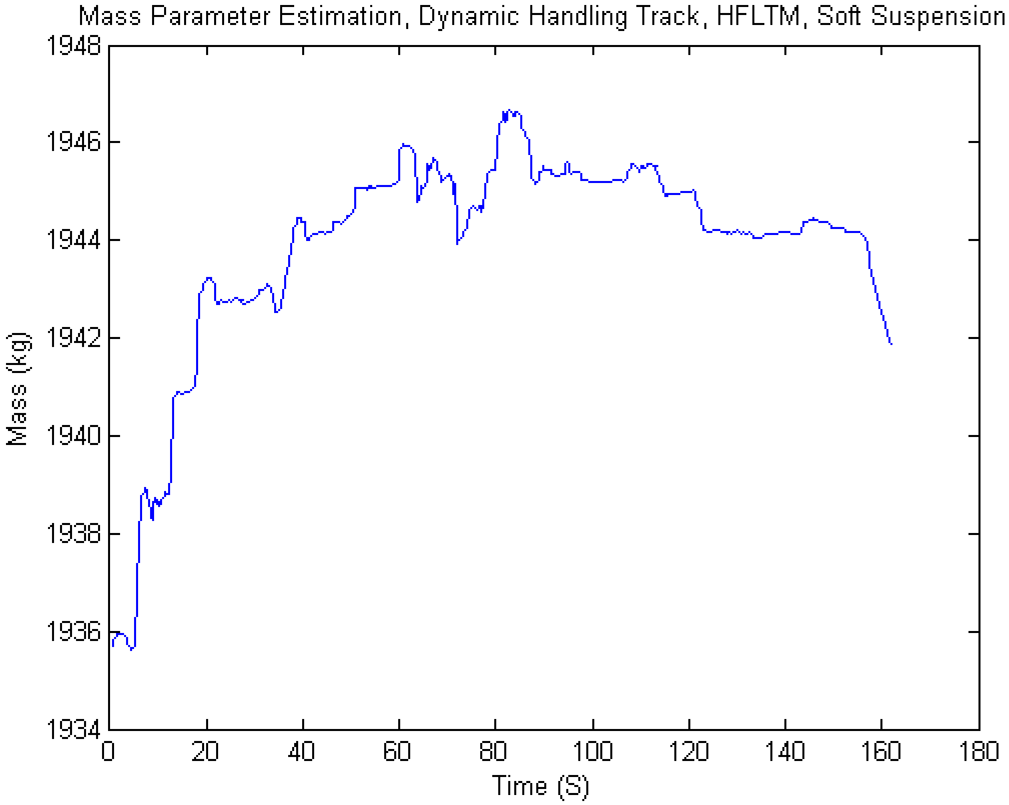


Figure 8.3 Mass Estimate for Dynamic Handling Track Using the Soft Suspension

It can be clearly seen why a rapid update for the pitch and roll centers are needed. The pitch center for the front can shift between positive and negative values. The negative values are when braking occurs at a magnitude that transfers enough load through the suspension linkages as to cause a jacking force, which is the anti-dive mechanic. The rear of the vehicle does get a negative value, and at that point the vehicle is driving up a grade while accelerating. A typical value for the rear is much larger than for the front, for this vehicle. **Figure 8.4** shows the pitch center estimates.

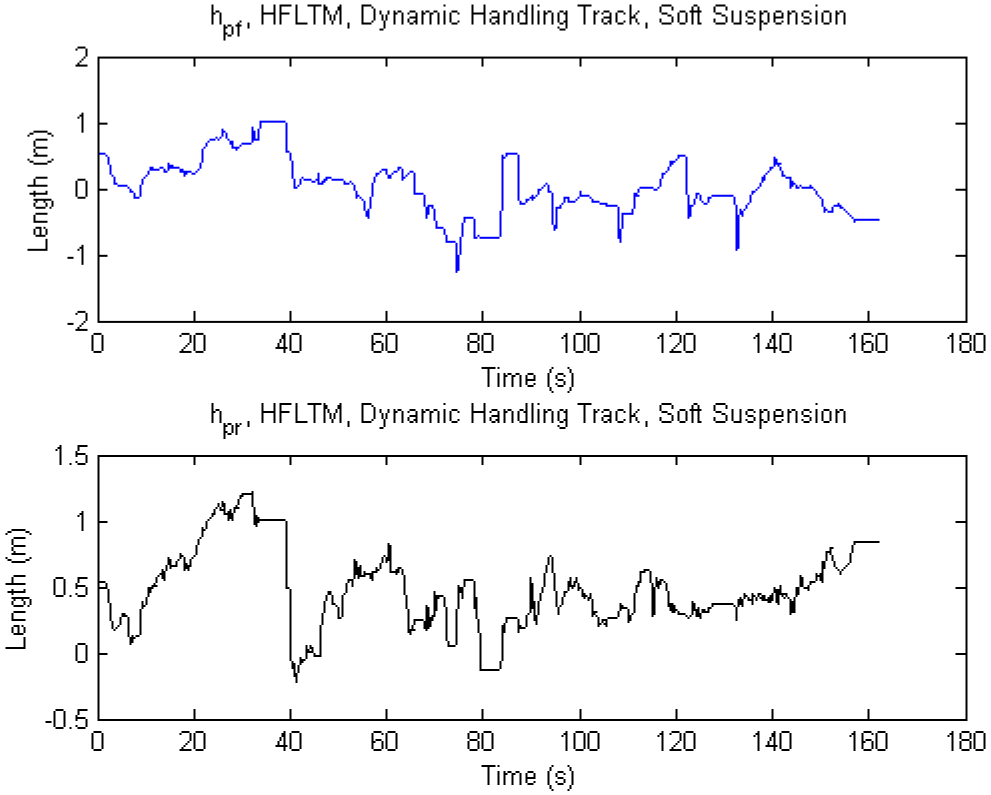


Figure 8.4 Pitch Center Height Estimates for the Dynamic Handling Track for the Soft Suspension

The nonlinear effects of the panhard rod are clearly seen in the roll center parameter estimates. The two roll centers have completely different trajectories. The rear roll center tends to oscillate about a constant DC value, which is what the previous studies converged to. The oscillations occur because of the un-modeled effects. The reason that the panhard rods effects are seen more clearly here, rather than in the Rural or Urban scenarios is that the magnitude of the dynamics is much larger and the roll angle of the vehicle is typically much larger. **Figure 8.5** shows the roll center parameter estimates.

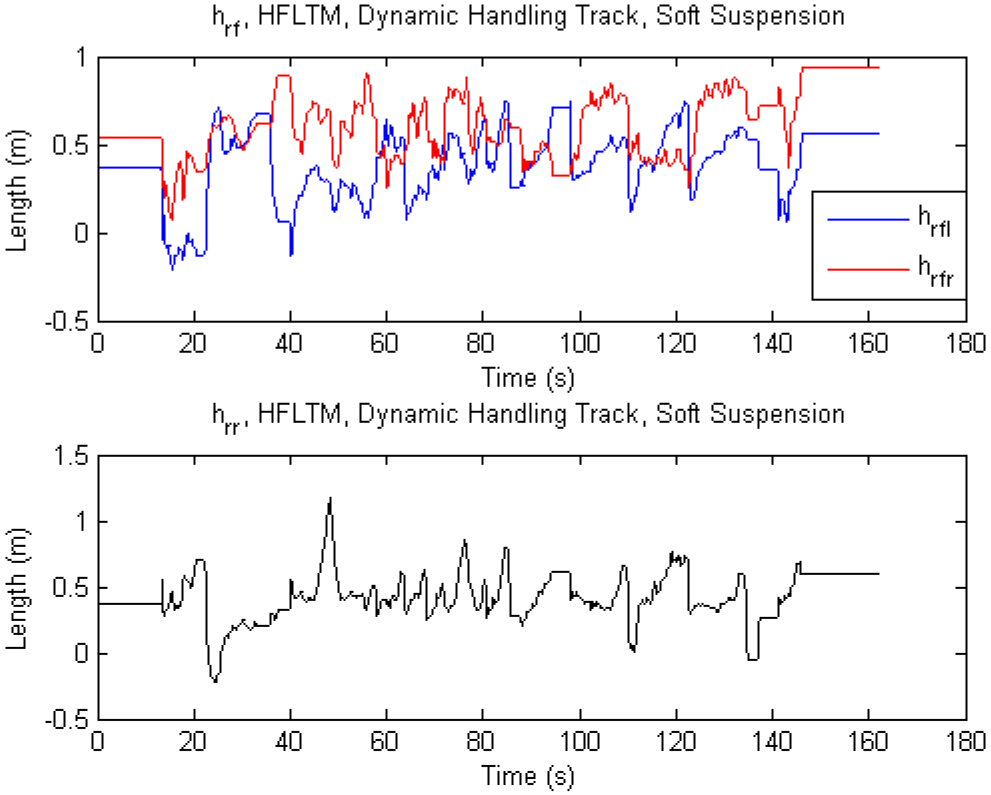


Figure 8.5 Roll Center Height for the Dynamic Handling Track using the Soft Suspension

The average of these lines is in the same neighborhood, which is why for lower dynamics the single coefficient was viable in the previous models. For larger dynamic events a faster update and multiple parameters are needed to adequately account for un-modeled effects.

8.1.2 Dynamic Handling Track with Hard Suspension

The hard suspension setting shows significantly different results than the soft setting. Much more of the load transfer is passed through the suspension, which results in significantly higher suspension forces. The suspension does not move very much in this setting, which causes significant tire deflections. The RMS error in the suspension force estimate is 253.96 *N* with a peak error of 1088 *N* at 114.6 seconds, shown in **Figure 8.6**.

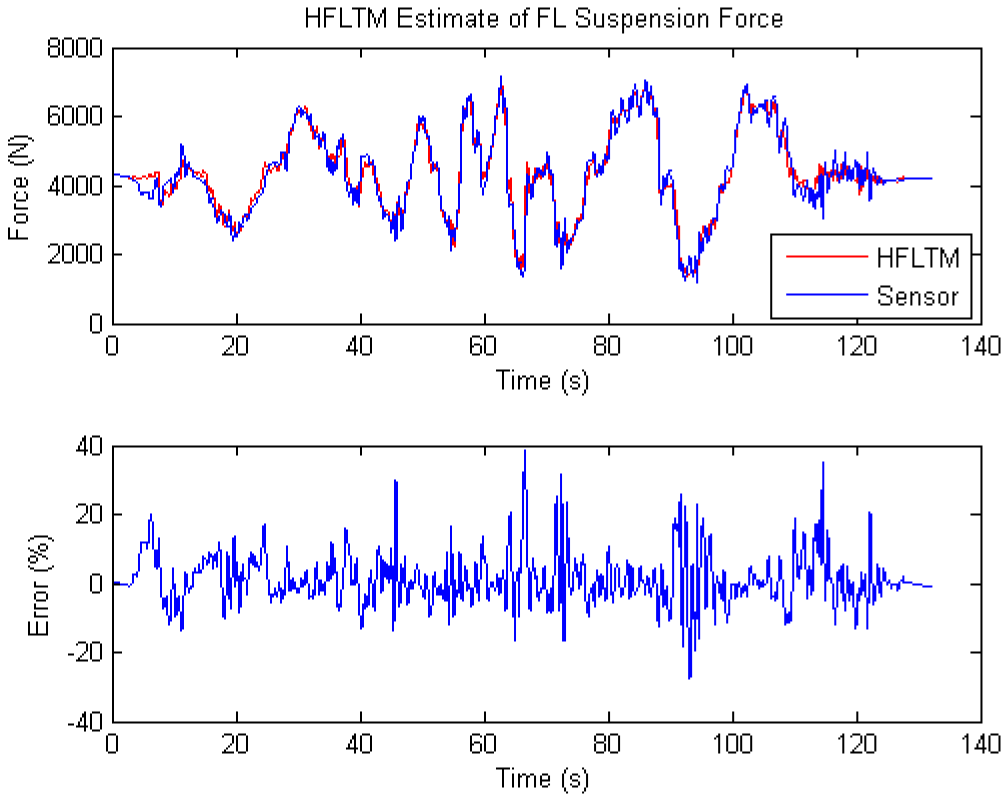


Figure 8.6 Front Left Suspension Force Estimate for the Dynamic Handling Track Using the Hard Suspension Setting

The longitudinal CG position converged to a larger value than when in the soft configuration, and the lateral position took much longer to stop drifting, and comes to a smaller estimate than the soft setting estimate. These estimates are shown in **Figure 8.7**.

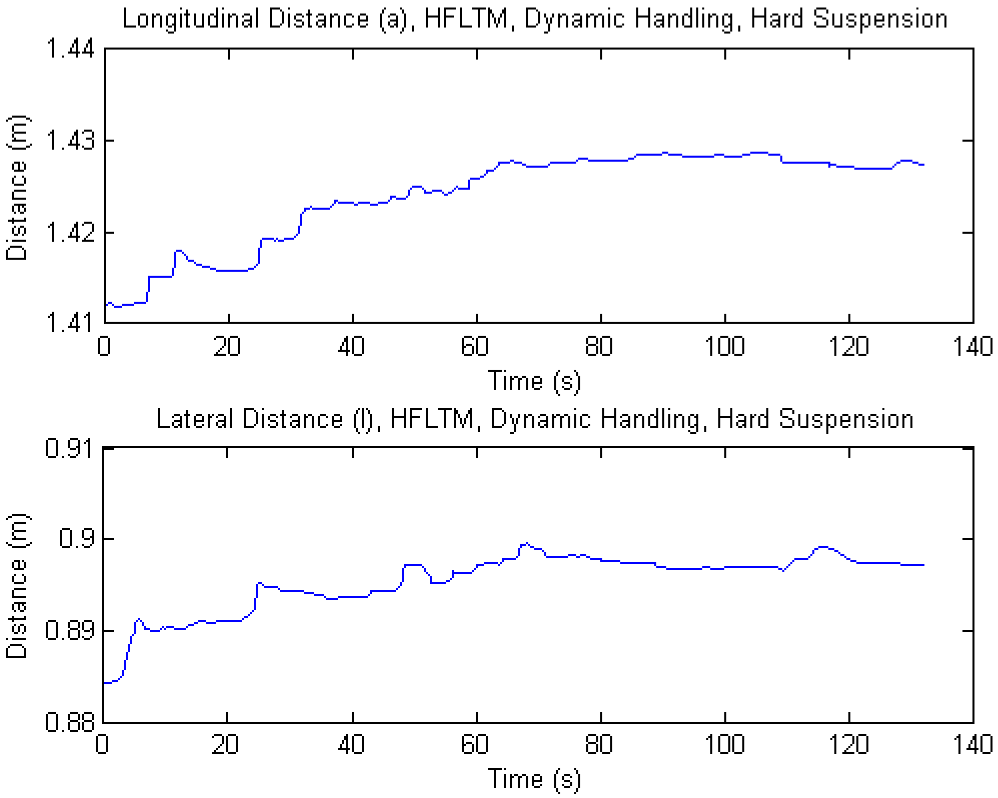


Figure 8.7 Horizontal CG Estimate for the Dynamic Handling Track for the Hard Suspension

The mass estimate of the vehicle for the hard suspension is different than for the soft suspension. The soft suspension has a small variation, but the hard suspension setting appears to be converging for the first two-thirds of the test. Its final value is also smaller than the soft setting. This is because of the asymmetrical (vertical, this time) friction effect that is slowly jacking the suspension down, which makes the vehicle appear to be heavier than it actually is. The results are shown in **Figure 8.8**.

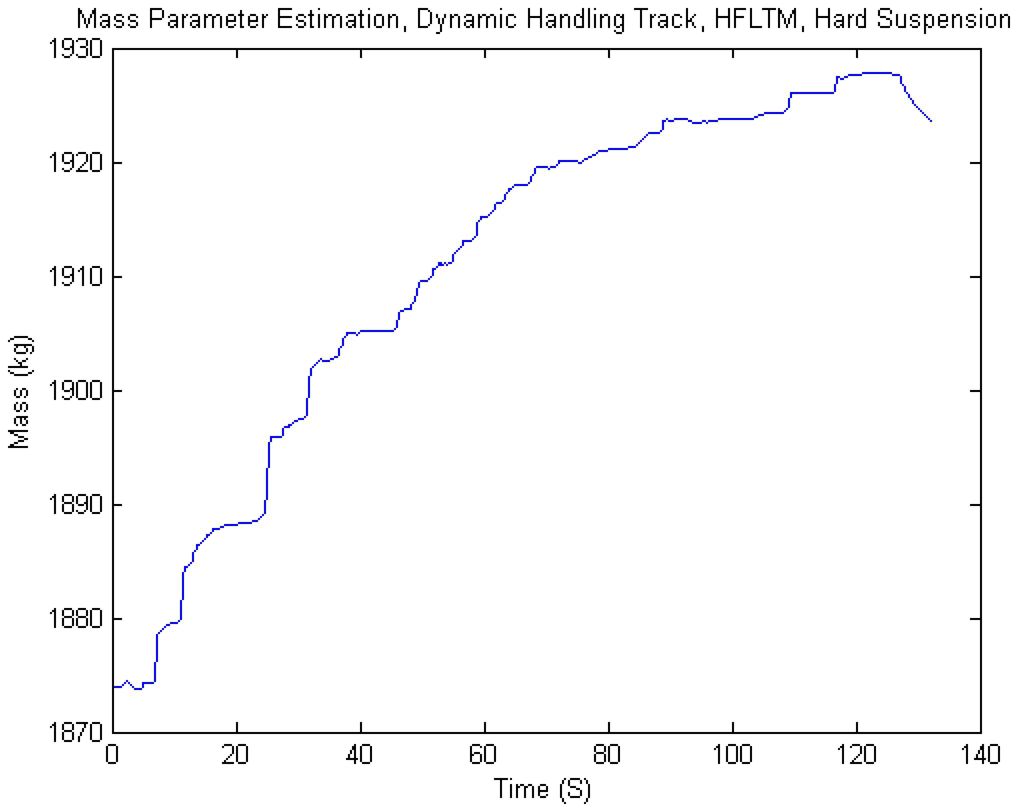


Figure 8.8 Mass Estimate for the Dynamic Handling Track Using the Hard Suspension

The pitch center estimates, shown in **Figure 8.9**, have significantly more variation, because the suspension doesn’t move as much, the vehicle body doesn’t move as much, and the inertia of the vehicle does not buffer the energy transfer so the suspension has to apply a much higher force.

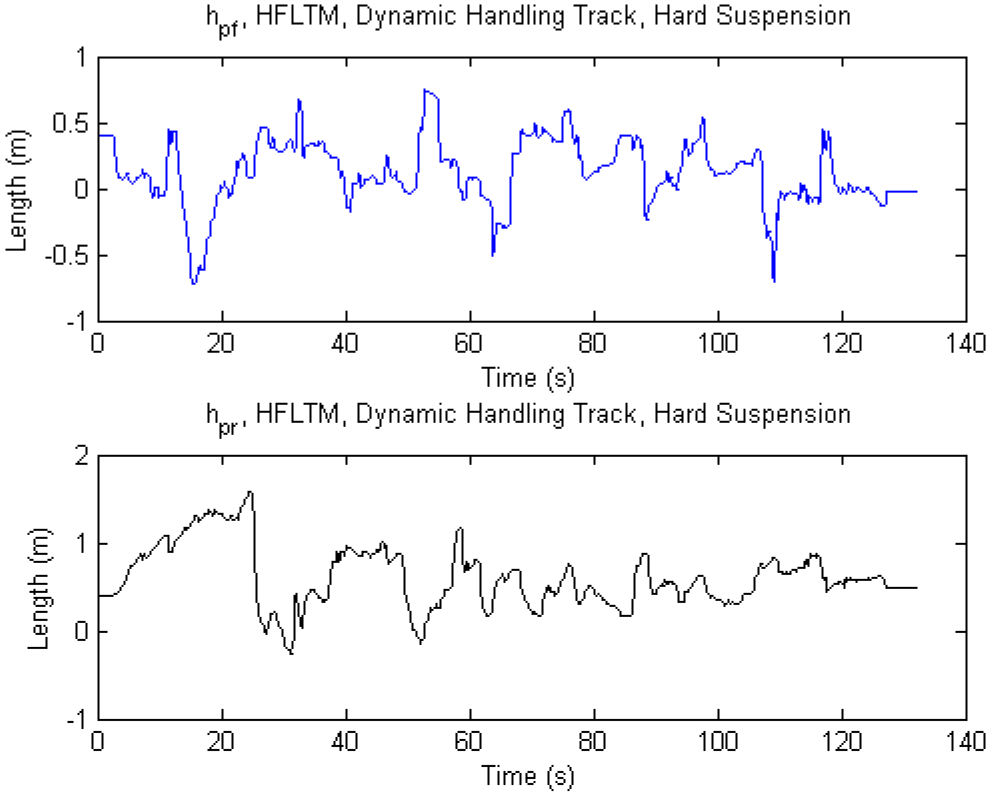


Figure 8.9 Pitch Center Height Estimates for the Dynamic Handling Track Using the Hard Suspension

Because the suspension doesn’t deflect as much, the panhard rod doesn’t appear to play as large of a role in the lateral load transfer. When looking at the roll center height estimates, shown in **Figure 8.10**, this is exactly what is observed. During very hard maneuvers the panhard does show an effect, but for softer maneuvers its effects are less significant.

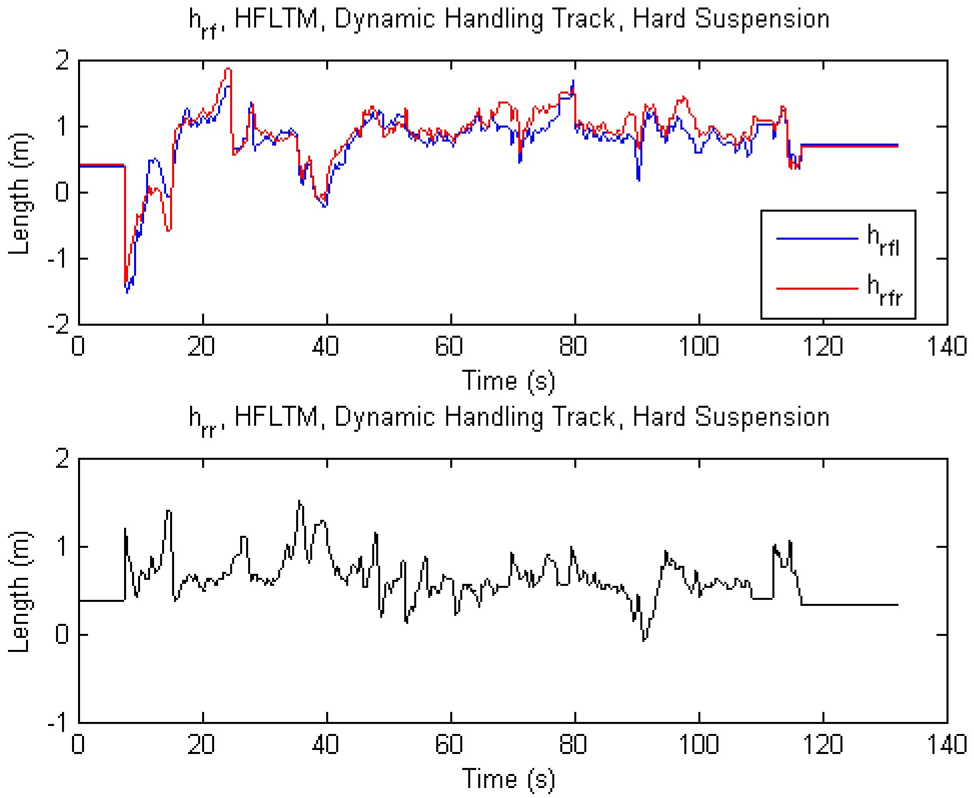


Figure 8.10 Roll Center Height Estimates for the Dynamic Handling Track Using the Hard Suspension

8.2 Why the HFLTM Works

The HFLTM is configured to compensate for un-modeled internal dynamic effects, such as linkage forces and roll/pitch angles. The reason that this is effective is because of how the mathematics are defined.

$$m\,A_y\,h_r = m\,A_y\,h_{rt} + F_{linkage} + F_{Aero} + F_{RollAngle} + F_{RollBar} \tag{145}$$

The right hand side has a roll center height, h_{rt}, that is a constant. Due to the additions of the linkage forces and other effects the left hand side estimate of the roll center height is constantly changing to compensate for the sum of the right side. One can see that if there are significant effects present when there is no lateral acceleration, the parameter h_r could diverge. For this reason if there is only a small acceleration present, the Kalman update for that parameter is disabled.

9 Axle Normal Force Estimate and Suspension Mapping

Classically, the estimate of the wheel vertical forces is simply the load transfer plus the static vertical weight. However, because of the linkage constraints this is not in general true or accurate, and secondary effects can cause significant differences from this estimation.

The assumption that is used in this work is that this classical constraint holds below the roll center point, and that all of the effects above the roll center point correspond to the roll center height parameter that the filter estimates, which contains all of the linkage forces etc.

$$m_{u,i}\, A_{zu,i} + h_{rc,i}\, m\, A_y + h_{pc,i}\, m\, A_x + SF_i = F_{ground,i} \tag{146}$$

The parameter $m_{u,i}$ is the unsprung mass, SF_i is the suspension force, and $F_{ground,i}$ is the ground force or normal force at the i^{th} corner. A_x and A_y are the longitudinal acceleration and lateral acceleration, and m is the total mass of the vehicle. The parameters $(h_{rc,i}, h_{pc,i})$ depend on the horizontal CG positions, and upon which wheel is being mapped.

$$h_{rc,fl} = \frac{b}{LB}\, h_{rc}\eta_1 \tag{147}$$

$$h_{pc,fl} = \frac{r}{LB}\, h_{pc}\eta_2 \tag{148}$$

For a system with a single roll center parameter. Considering all of the system's linkages is important for determining how many of these parameters to use. For the highest accuracy, a separate parameter should be used at each corner. The horizontal CG values are used to make this invariant with regard to the horizontal CG position. For a full mapping there would be eight η coefficients.

The parameter η is an empirically derived parameter that is defined as:

$$\eta_1 = \left(1 - sign\left(\frac{A_y\left(h_s + \frac{\sum D_{s,i}}{i}\right)}{h_s}\right)\frac{\left(h_s + \frac{\sum D_{s,i}}{i}\right)}{h_s}\right)^2 \left(1 - \frac{sign\left(A_y(D_{s,rl} - D_{s,rr})\right)(D_{s,rl} - D_{s,rr})}{B}\right)^2 \tag{149}$$

$$\eta_2 = \left(1 - sign\left(\frac{A_y\left(h_s + \frac{\sum D_{s,i}}{i}\right)}{h_s}\right)\frac{\left(h_s + \frac{\sum D_{s,i}}{i}\right)}{h_s}\right)^2 \tag{150}$$

There are several reasons for developing this method. The first is that wheel force transducers are expensive and not commonly available for long-term implementation. The second reason is that it is desired to have a method that does not require being tailored to the specific vehicle. Using a separate mapping for each wheel and with the provided mathematical support of a fast updating parameter to compensate for un-modeled dynamics, such a method is provided. To properly extract the vehicle height it is likely that the method used will have to be tailored to the specific vehicle to account for all of the linkages and other effects, which become costly both in computation and in manpower.

9.1 Results for Suspension Mapping Concept Study

The results presented here are a proof of concept study. The method is demonstrated for some changes in the vehicle's parameters, and for several different maneuvers. Data was only available for one wheel. Further work with sensors for all four wheels would likely result in improved results.

9.1.1 Training Set Results

The parameters are tuned from a single run of the dynamic handling track, by the Least Squares technique. The parameter h_s is chosen based on the static suspension displacement for this run, to represent when the suspension of the vehicle compresses or changes from run to run, and to make it invariant with respect to changing suspension heights.

Table 9.1 Parameter Values for Suspension Mapping

$M_{unsprung}$	$72\ kg$
h_{rc}	0.57
h_{pc}	0.96
h_s	$0.21\ m$

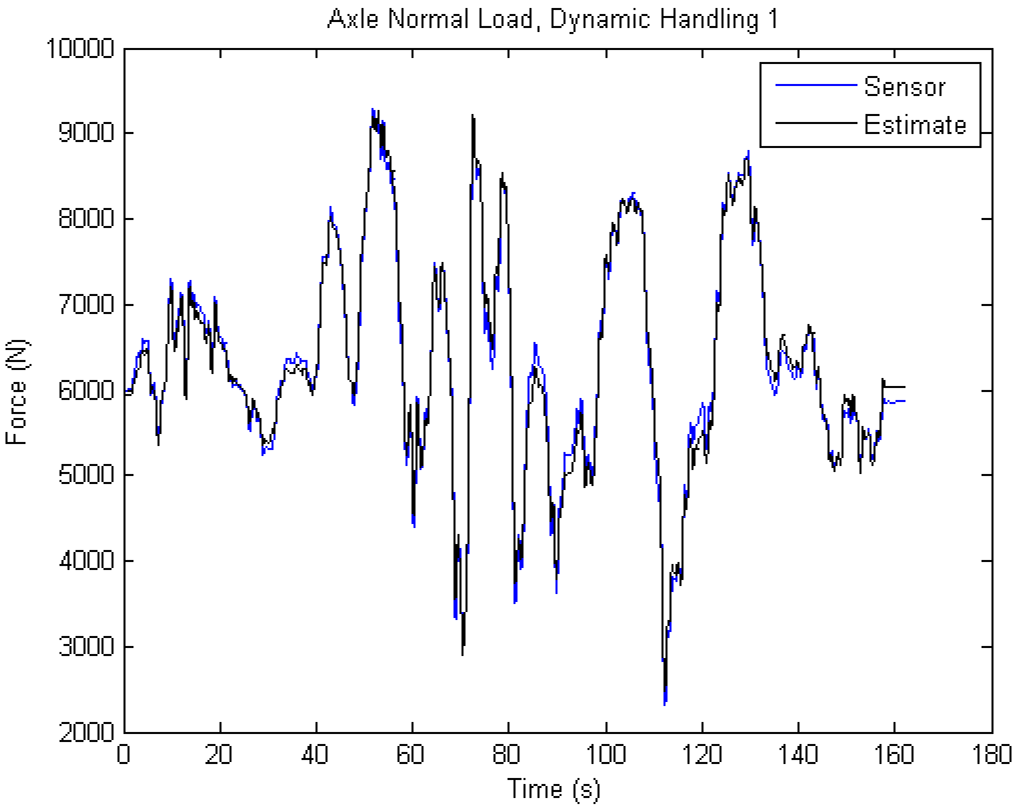


Figure 9.1 Dynamic Handling Training Data Run

Table 9.2 Training Data Error for Suspension Mapping

RMS Error	$122\ N$
Peak Error	$326\ N\ @(111\ s)$

The results, shown in **Figure 9.1**, are very good. The errors can primarily be attributed to the friction in the suspension. The friction acts in an asymmetric method that prevents the wheels from easily extending or contracting sometimes, which makes the vehicle look like it weighs more or less, as it is constrained to roll about a different axis. This effect will be seen quite clearly later on for the constant radius test and the sine sweep test.

9.1.2 Validation Case – Special Maneuvers

This section covers the investigation of how well the mapping technique validated for special maneuvers. The two maneuvers presented here highlight some of the problems of the method quite clearly. The first maneuver is a constant radius test.

The constant radius test involves the vehicle driving in a constant radius with a slowly increasing speed. The results are promising, but show a particular flaw in the method. The suspension friction effects are not accounted for and cause a bias in the results.

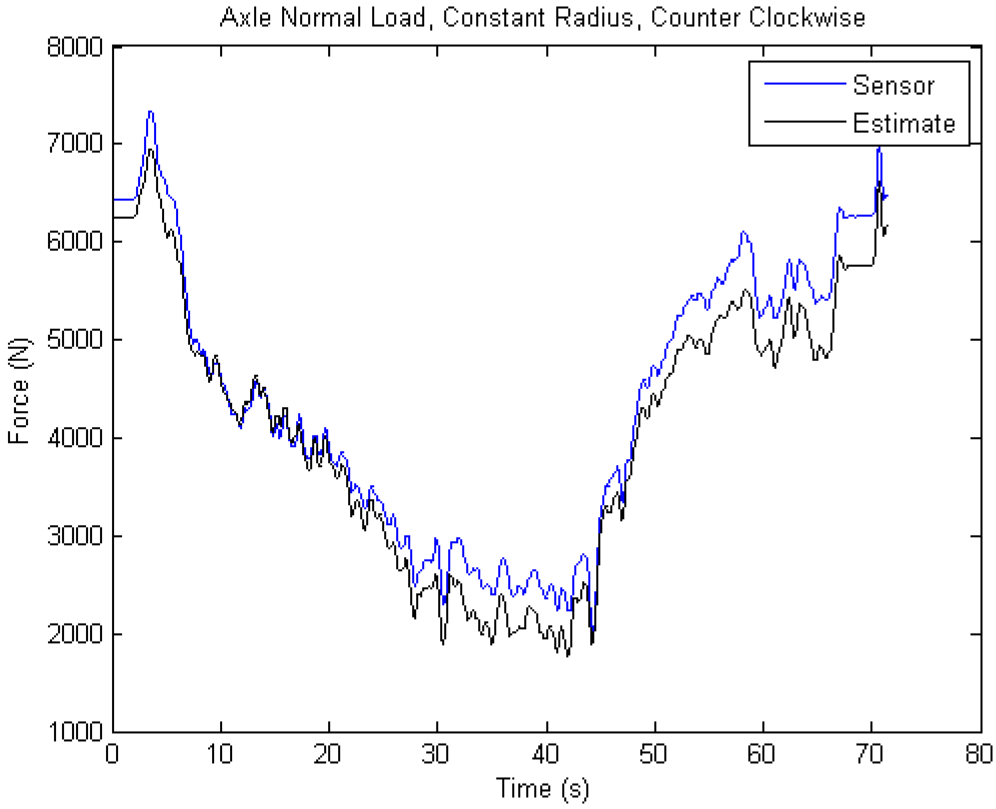


Figure 9.2 Counter Clockwise Constant Radius Validation Case

Table 9.3 Counter Clockwise Constant Radius Test Error Results

RMS Error	$346\ N$
Peak Error	$648\ N\ @(58.1\ s)$

The estimate, shown in **Figure 9.2**, is quite good up to about 28 seconds. At that point the suspension starts to lock because of friction, which prevents the wheel from extending further, and causing an un-modeled force to appear. As the vehicle slows down the friction effects dissipate a bit, but then prevent the vehicle from re-orienting to a level position. The vehicle is stuck with a constant roll angle that puts more of the weight on the right wheels, than the left, which is exactly what the suspension force sensor shows. This phenomenon is observed directly **Figure 9.3**.

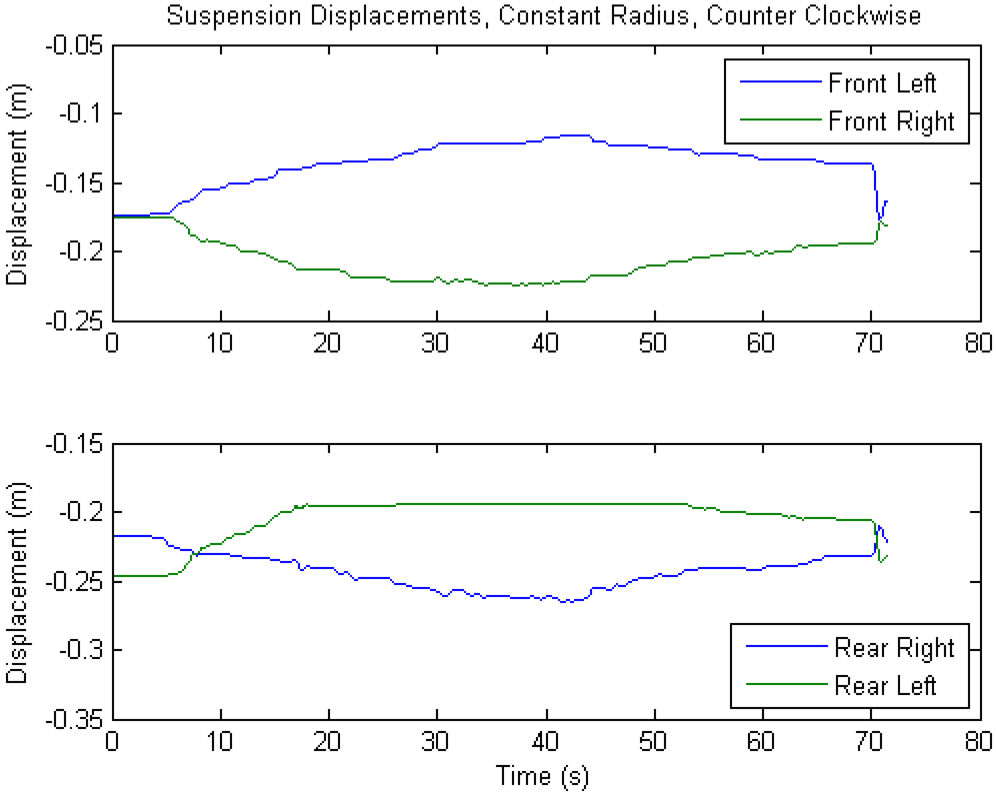


Figure 9.3 Suspension Displacements for the Constant Radius Counter Clockwise Test

Unlike the previous maneuver, the clockwise constant radius, results shown in **Figure 9.4**, only shows minor effects of friction. The friction effect is asymmetrical in two ways: First, it is more likely to contract than expand, and secondly it is more likely to have this contraction occur on one side of the vehicle, as seen in **Figure 9.5**. The biggest mismatch here is at the beginning and end, where the high longitudinal acceleration causes a lateral load transfer, which can be observed in the suspension displacements. Since that load transfer is not accounted for there is a big discrepancy between the model and the sensor. If the coefficient is reduced from 0.9 to 0.35 for the longitudinal component these spikes disappear and the matching of the sensor is maintained. This effect only seems to appear when the outriggers are added to the vehicle. It does not appear to correlate to gyroscopic motion, or to aerodynamic effects. These results are proof of concept, and further work with more sensors and sensor data is needed to properly characterize this effect.

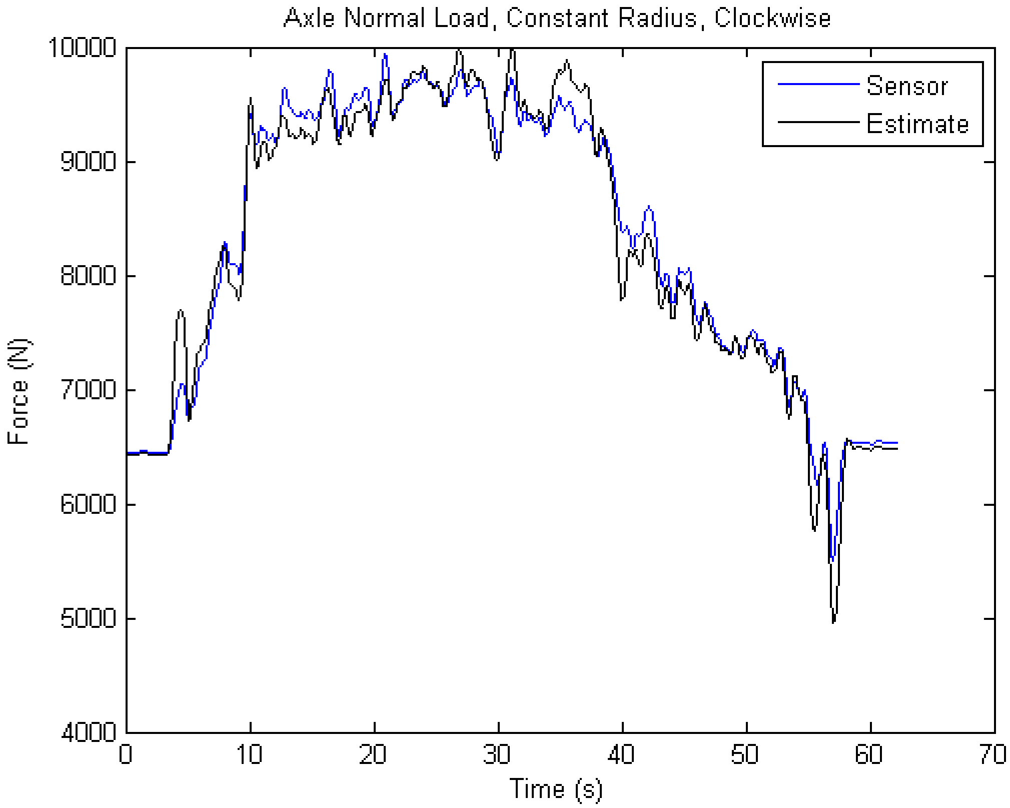


Figure 9.4 Clockwise Constant Radius Validation Case

Table 9.4 Clockwise Constant Radius Error Results

RMS Error	193 N
Peak Error	700 N @(4.2 s)

If the initial 5 seconds and last 10 seconds are ignored, the largest peak error is 611 N, at 35 seconds.

Table 9.5 Truncated Clockwise Constant Radius Error Results

RMS Error	172 N
Peak Error	611 N @(35 s)

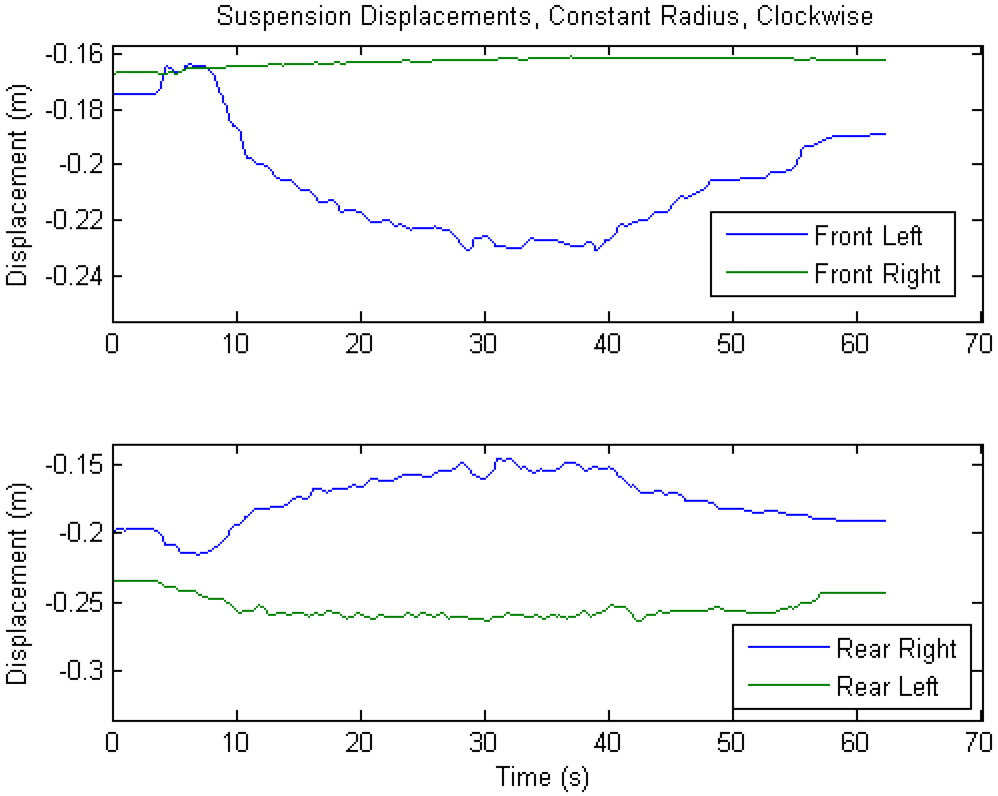


Figure 9.5 Clockwise Constant Radius Suspension Displacements

The next maneuver for validation is a sine sweep pattern. The vehicle sweeps left and right at a fairly constant speed. The test is performed by a driver and is not performed at a set frequency. The vehicle is simply steered left and right in a approximately-rhythmic manner. The results, shown in **Figure 9.6**, show exactly what was observed in the previous tests. The asymmetric friction effects cause the suspension to compress and then try to lift one of the wheels, rather than extending smoothly, which causes the vehicle to roll about a different point, which results in a shift in the tire normal force. The error at the beginning is caused by the lateral load transfer that is caused by a longitudinal acceleration, as noted previously.

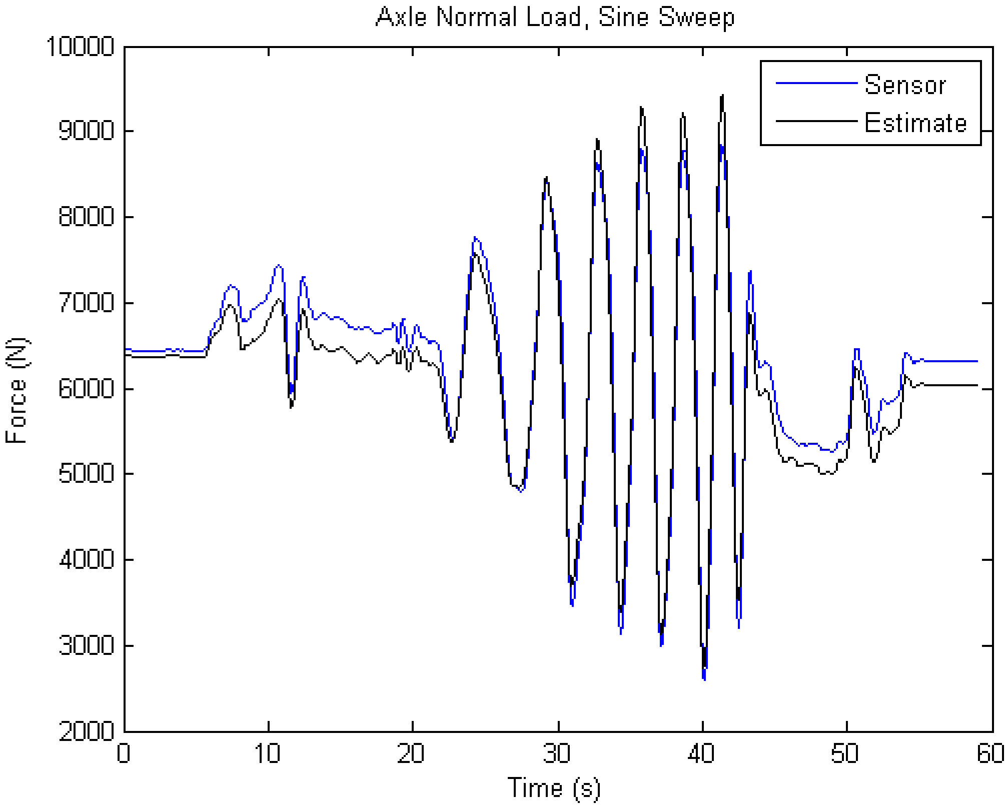


Figure 9.6 Sine Sweep Validation Case

Table 9.6 Sine Sweep Error Results

RMS Error	$277.4\ N$
Peak Error	$632\ N\ @(36.6\ s)$

The error at the end is a combination of the lateral load transfer and the friction effects, as seen in Figure 9.7.

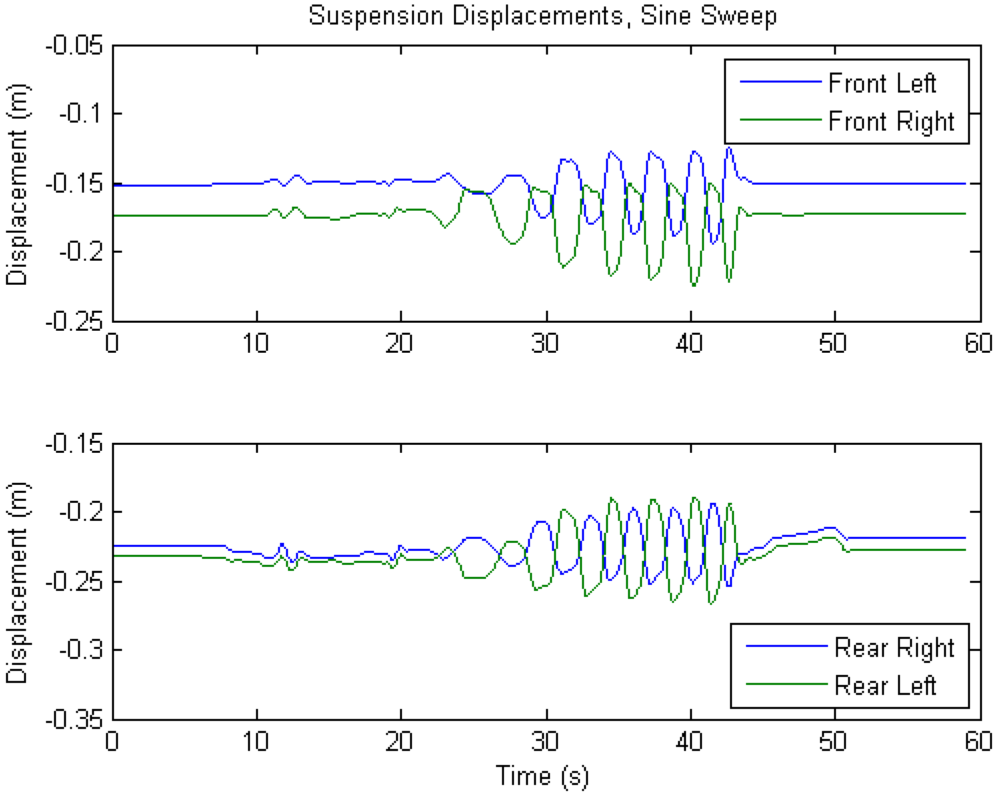


Figure 9.7 Sine Sweep Suspension Displacements

9.1.3 Validation Case – General Driving

This section demonstrates the devised strategy when applied to general driving and non-specialized cases. The first two cases are general driving around the testing facility, and the third case is another trip around the Dynamic Handling track.

The first general driving scenario results in good capture of the peaks. The friction effects are quite evident when there is limited dynamics, as seen in **Figure 9.8**. For these tests the suspension displacements have been omitted, as they do not add much to the discussion.

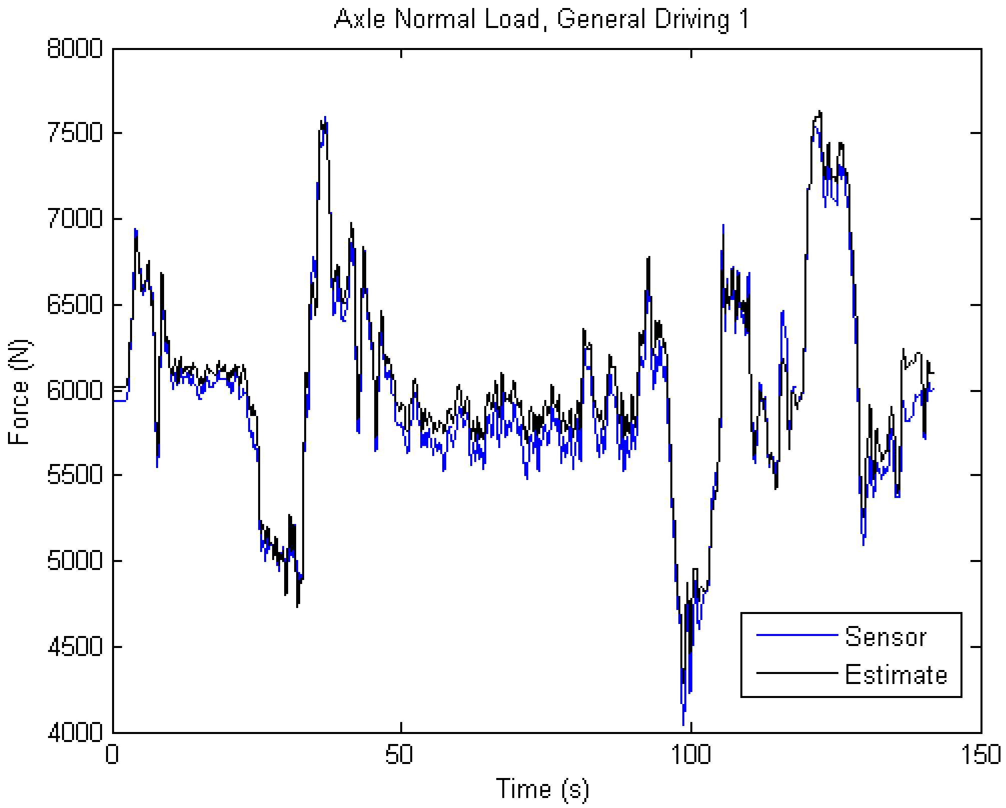


Figure 9.8 General Driving 1 Validation Case

Table 9.7 General Driving 1 Error Results

RMS Error	$123.4\ N$
Peak Error	$331\ N\ @(13.8\ s)$

The second general driving test, shown in **Figure 9.9**, performs remarkably better than the first one. There are two points that should be looked at. At the beginning of the simulation there is an initial offset caused because of the suspension friction. The section at 141 seconds is when the suspension friction causes a significant force, such that when the vehicle rolls from one side to the other it does not roll unconstrained. The vehicle picks up the wheels on the right side, which makes the vehicle essentially roll about a different point.

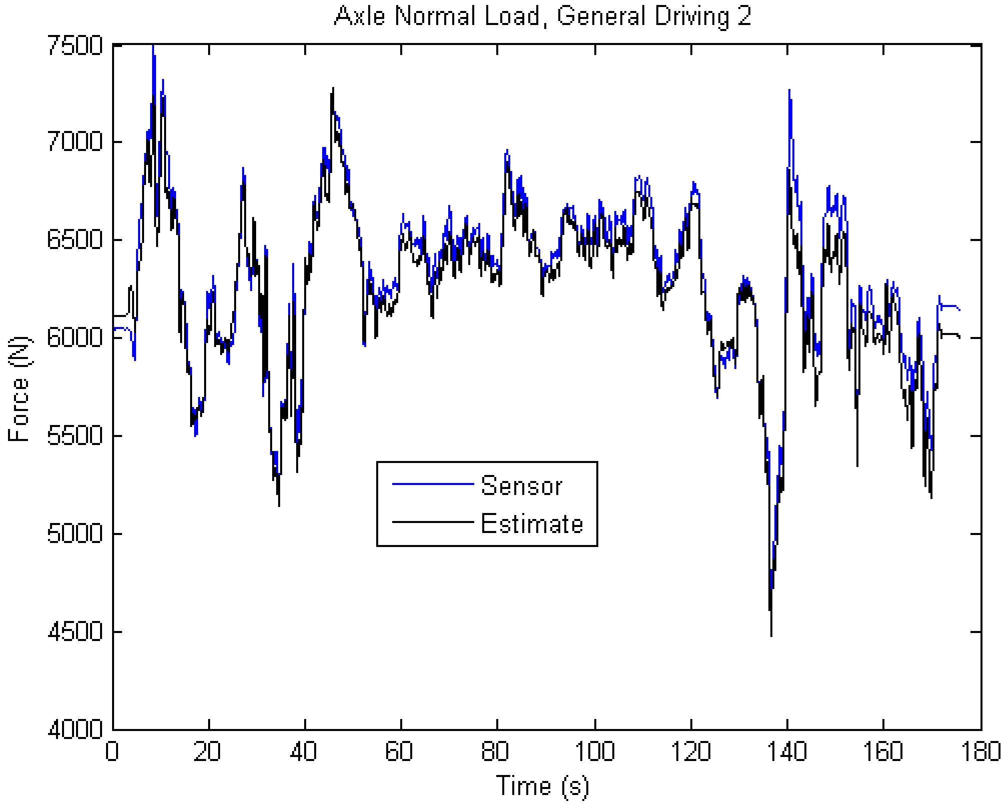


Figure 9.9 General Driving 2 Validation Case

Table 9.8 General Driving 2 Error Results

RMS Error	$111\ N$
Peak Error	$464.5\ N\ @(141\ s)$

As a comparison, the model is applied to a different run around the dynamic handling track. The first run was approximated better, and showed fewer signs of friction effects which was why it was used. Here those friction effects can be observed a bit clearer. Overall, the model does a good job, except for the peak at 41 seconds where it over-shoots by a fair margin. It does capture the notch at 96 seconds with about $50\ N$ error. The results are shown in **Figure 9.10**.

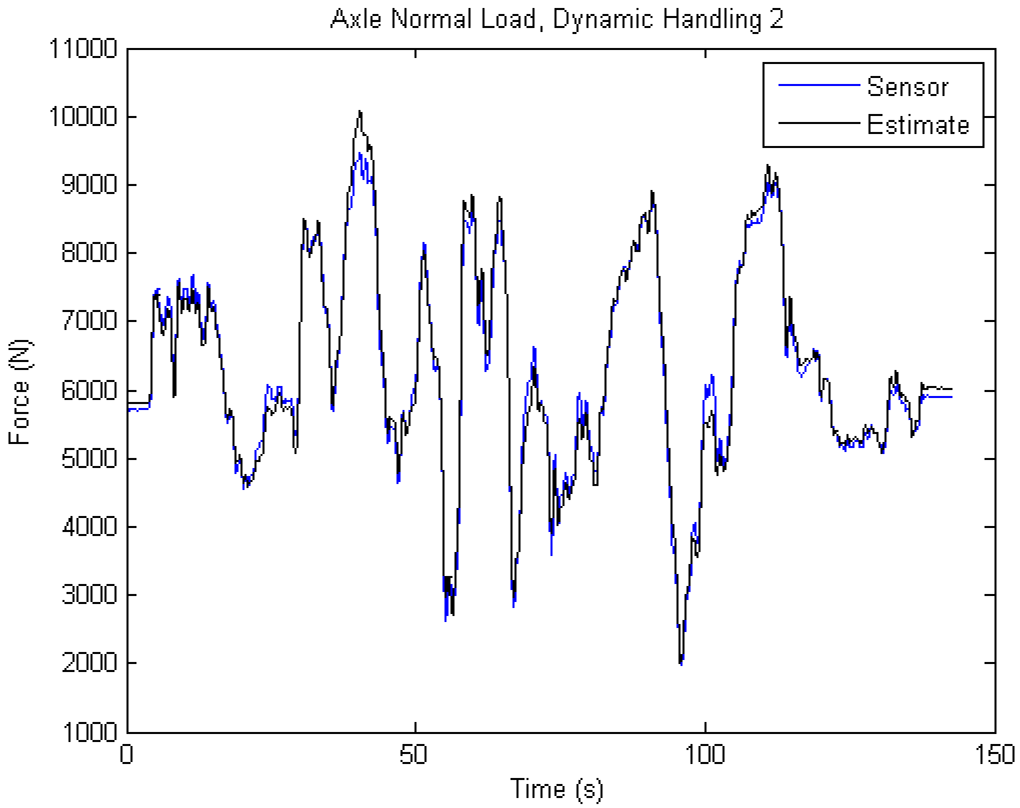


Figure 9.10 Dynamic Handling 2 Validation Case

Table 9.9 Dynamic Handling 2 Error Results

RMS Error	$175\ N$
Peak Error	$645\ N\ @(40.8\ s)$

10 Vehicle Stability and Rollover through LTM Model

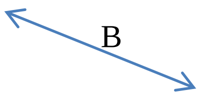
This chapter provides a brief interlude between the parameter estimation sections and the tire force estimations. This chapter provides several derivations of stability metrics based on the LTM model, and compares them to some of the current ones.

10.1 Vehicle Stability Through Load Transfer Model

This section investigates the stability of the vehicle through the lens of the LTM. The same investigations are viable with the MLTM, but the answers result in more intuitive results with the LTM. All the algorithms if implemented should be done so using the MTLM, or HFLTM for better accuracy.

10.1.1 LTM and Static Stability Factor (SSF)

The NHTSA has determined that a good first measurement of a vehicle's stability is determined through what they call the "Static Stability Factor (SSF)." The SSF measurement is determined from a block on a slanted hill, illustrated in **Figure 10.1**:



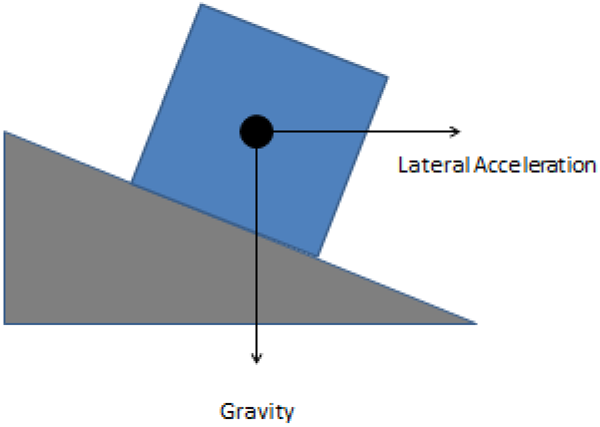


Figure 10.1 Static stability factor diagram

The SSF is the moment balance of the vertical acceleration to the lateral acceleration at the edge of the vehicle:

$$A_z\left(\frac{B}{2}\right) = A_y h \tag{151}$$

This yields the SSF:

$$\frac{A_y}{A_z} = \frac{B}{2h} \tag{152}$$

If A_z is the acceleration due to gravity, this equation gives a first order limit on the lateral acceleration that the vehicle can sustain before rolling over. There are some key problems in this model, such as the lateral position of the vehicle’s CG and how the roll of the body plays into the stability.

From the load transfer equations a similar equation can be calculated by solving for the necessary lateral acceleration to transfer all of the weight from one side of the vehicle to the other. This is the condition upon which traction on one side of the vehicle is lost. For easy conception of this, the equations are used from the regular LTM model, including the roll and pitch angle terms. The lateral acceleration that is calculated for reducing the load on the left side of the vehicle:

$$F_{fl} + F_{rl} = 0 \tag{153}$$

$$\frac{mA_z}{BL}\left(br - A_x r h_p - A_y b h_r\right) + \frac{mA_z}{BL}\left(ar + A_x r h_p - A_y a h_r\right) - \frac{m h_r \phi A_z}{B} = 0 \tag{154}$$

$$A_y = \frac{r}{h_r} - \phi \tag{155}$$

This is to remove all of the weight from one side of the suspension to the other. If this is transcribed to the wheel contacts, assuming a rigid roll center, the acceleration limit becomes:

$$A_y = \frac{r}{h} - \frac{\phi h_r}{h} \tag{156}$$

Which, if $\phi = 0$, and the lateral position of the CG is at the half-track:

$$A_y = \frac{B}{2h} \tag{157}$$

Remember, A_y is measured in g's in the LTM, so this equation returns the same value as the SSF. Therefore, the load transfer equations return better estimates of the vehicle's propensity to rollover than the SSF measurement. The method employed here is redone for the acceleration needed to remove the weight from the right side of the vehicle, as it requires a different amount of acceleration to do that, because of the asymmetry of the lateral CG position.

10.1.2 LTM and Rollover Coefficient (RC)

There is an alternative method for calculating the vehicle stability, called the Rollover Coefficient. The Rollover Coefficient is defined as the weight on the left side of the vehicle (W_l) minus the weight on the right side of the vehicle (W_r), normalized by the total weight of the vehicle (W):

$$RC = \frac{W_l - W_r}{W} \tag{158}$$

This returns a value that is $\in [-1\ 1]$. The vehicle is in danger when close to the bounds. If we evaluate this equation using the LTM we recover:

$$\frac{F_l - F_r}{\sum_{i=fl,fr,rl,rr} F_i} = \frac{1}{m\,A_z}\left[\frac{mA_z}{B}\left(r - l - 2A_y h_r\right) - \frac{2mh_r A_z \phi}{B}\right] \tag{159}$$

This results in:

$$RC = \frac{r-l}{B} - \frac{2A_y h_r}{B} - \frac{2h_r \phi}{B} \tag{160}$$

Evaluating this at the negative boundary, or when the weight on the left side of the vehicle goes to zero, for the lateral acceleration:

$$A_y = \frac{B}{2h_r} - \phi + \frac{r-l}{2h_r} \tag{161}$$

This is the same equation that was derived by the LTM when solving for the lateral acceleration when the force on the left side of the vehicle was zero.

The generalized equation that can be used for calculating the rollover threshold for the vehicle, regardless of which side of the vehicle is going to rollover, is:

$$A_y = sign(A_y)\frac{B}{2h_r} - \phi + \frac{r-l}{2h_r} \tag{162}$$

This equation is derived from the LTM model, and is in agreement with the Rollover Coefficient, and the SSF measurement. If applied to the total height of the vehicle, the equation for vehicle stability is:

$$A_y = sign(A_y)\frac{B}{2h} - \frac{\phi h_r}{h} + \frac{r-l}{2h} \tag{163}$$

10.2 Vehicle Stability through the Suspension Mapping Technique

The Suspension Mapping technique showed in Chapter 9 can be used to approximate the vehicle's stability. Repeating the suspension mapping equation (Eq. 146):

$$m_{u,i}\, A_{zu,i} + h_{rc,i}\, m\, A_y + h_{pc,i}\, m\, A_x + SF_i = F_{ground,i} \tag{164}$$

This equation, once the parameters are estimated, can be used to estimate when the vehicle will lose contact at a specific wheel, or when the vehicle will lose contact on one side of the vehicle. Further work could be done to estimate what the acceleration limits are that will cause loss of traction, in conjunction with a surface friction coefficient, or even at what point a significant acceleration becomes large enough to roll the vehicle over.

Because this equation has different values for each $h_{pc,i}$ or $h_{rc,i}$ the results of those equations are not very compact and do not result in intuitive equations. They do however result in excellent equations for controller design and analytical formulation of vehicle rollover or traction constraints.

11 Conclusions

Presented in this work are three major results. The first is the improvement and analysis of the gPC-EKF filter. The work here investigated the paradoxical nature of the filter: To be able to converge requires a populated covariance matrix, but to populate the covariance matrix requires the system to not converge. Empirical steps were proposed to compensate for this, and analytical solutions have yet to be found. The solution to this problem is first the solution to a Kalman filter tracking a desired frequency signal through only the process noise matrix which has not to my knowledge been solved. This solution would then have to be solved with the constraint that the system behave in such a manner as to help the parameter values to converge, which is a highly nonlinear problem that does not have an analytical formulation as of yet.

Results were demonstrated for different types of distributions (and their respective basis functions) and their effects on the systems solutions were discussed. The filter was able to provide information about whether the excitation was sufficient or not. The method was demonstrated for multiple parameters and for regression problems.

Finally, an investigation was conducted to be able to test for the system's convergence. A method that can be employed to test for convergence was developed, and shown in both analytical and empirical results that it is viable.

The second major work was the estimation of the vehicle's mass and horizontal CG position. To the best of my knowledge, this is the first time that, for a general driving scenario, these parameters were able to be estimated, and to the accuracy that they have been, despite the significant error induced by the suspension friction. The method was shown for various driving scenarios, and the initial method was improved over several iterations.

During the course of the previous step it became clear that being able to estimate the total height of the vehicle was very difficult, if not impossible. Dynamic effects that were originally thought to be second--order effects summed to create first--order errors. The resulting technique for estimating the contact forces proved much more reliable, and much more computationally efficient.

The method was demonstrated in a proof of concept study. Full implementation of the system will require that the vehicle be mapped at each corner independently, with four rather than one wheel force transducer's data. The results presented here are very encouraging, and lead me to believe that the method would be very successful when fully implemented.

12 Future Work

The continuation of this work would primarily revolve around four areas. The first is the analytical formulation of the required improvements for the gPC-EKF. The second would be the estimation and inclusion of the suspension friction effects into the above models. The third would be the final mapping of the vehicle’s suspension using data from four wheel force transducers to result in a comprehensive solution that could be deployed on the vehicle. The fourth and final area would the creation of traction and roll stability controllers that use the above suspension mapping technique.

References

[1] G. J. Forkenbrock, W. R. Garrot, M. Heitz, and B. C. O'Harra, "A Comprehensive Experimental Evaluation of Test Maneuvers That May Induce On-Road, Untripped, Light Vehicle Rollover. Phase IV of NHTSA's Light Vehicle Rollover Research Program," D. o. Transportation, Ed., ed: NHTSA, 2002.

[2] G. J. Forkenbrock, B. C. O'Hara, and D. Elsasser, "An Experimental Examination of 26 Light Vehicles Using Test Maneuvers That May Induce On-Road, Untripped Rollover and a Discussion of NHTSA's Refined Test Procedures - Phases VI and VII of NHTSA's Light Vehicle Rollover Research Program," D. o. Transportation, Ed., ed: NHTSA, 2003.

[3] G. J. Forkenbrock, B. C. O'Harra, and D. Elsasser, "A Demonstration of the Dynamic Tests Developed for NHTSA’s NCAP Rollover Rating System - Phase VIII of NHTSA’s Light Vehicle Rollover Research Program," D. o. Transportation, Ed., ed: NHTSA, 2004.

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

[5] H. K. Fathy, D. Kang, and J. L. Stein, "Online Vehicle Mass Estimation Using Recursive Least Squares and Supervisory Data Extraction," *American Control Conference,* 2008.

[6] O. Pizarro and D. Sbarbaro, "Parameter Subset Identification by Recursive Least Squares," *Proceedings of the American Control Conference,* 1998.

[7] M. M. Chensarkar and U. B. Desai, "A Robust Recursive Least Squares Algorithm," *IEEE Transactions on Signal Processing,* vol. 45, 1997.

[8] J. Chen and F. Ding, "Modified stochastic gradient identification algorithms with fast convergence rates," *Journal of Vibration and Control,* vol. 17, pp. 1281-1286, 2011.

[9] M. A. Kouritzin, "On the convergence of linear stochastic approximation procedures," *Information Theory, IEEE Transactions on,* vol. 42, pp. 1305-1309, 1996.

[10] J. Gill, *Bayesian Methods: A Social and Behavioral Sciences Approach*, 2nd ed.: Chapman and Hall/CRC, 2009.

[11] T. Choo Par, Y. H. Zweiri, K. Althoefer, and L. D. Seneviratne, "Online soil parameter estimation scheme based on Newton-Raphson method for autonomous excavation," *Mechatronics, IEEE/ASME Transactions on*, vol. 10, pp. 221-229, 2005.

[12] J. J. More, "The Levenberg-Marquardt algorithm: Implementation and theory," in *Proceedings of the Biennial Conference Held at Dundee, June 28–July 1, 1977*, 1978, pp. 105-116.

[13] K. J. Astrom and P. Eykhoff, "System Identification - A Survey," *Automatica,* vol. 7, pp. 123-162, 1971.

[14] G. Welch and G. Bishop, "An Introduction to the kalman Filter," University of North Carolina at Chapel Hill TR 95-041, 2006.

[15] T. A. Wenzel, K. J. Burnham, M. V. Blundell, and R. A. Williams, "Dual extended Kalman filter for vehicle state and parameter estimation," *Vehicle System Dynamics,* vol. 44, pp. 153-171, 2006.

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

[17] G. Evensen, "The Ensemble Kalman Filter: Theoretical Formulation and Practical Implementation," *Ocean Dynamics,* vol. 53, 2003.

[18] "Iterated Uscented Kalman Filter for Passive Target Tracking," *IEEE Transactions of Aerospace and Eletronic Systems,* vol. 43, 2007.

[19] P. Li, T. Zhang, and B. Ma, "Unscented Kalman Filter for Visual Curve Tracking," *Image and Vision Computing,* vol. 22, 2003.

[20] J.-H. Wang and J.-B. Chen, "Adaptive Unscented Kalman Filter for Initial Alightment of Strapdown Inertial Navigation Systems," *Proceedings of the Ninth International Conference on Machine Learing and Cybernetics,* 2010.

[21] S. Julier and J. Uhlmann, "Unscented Filtering and Nonlinear Estimation," *Proceedings of the IEEE,* vol. 92, 2004.

[22] A. Sandu, C. Sandu, and M. Ahmadian, "Modeling Multibody Systems With Uncertainties. Part I: Theoretical And Computational Aspects," *Multibody Syst Dyn,* vol. 15, pp. 373-395, 2006.

[23] C. Sandu, A. Sandu, and M. Ahmadian, "Modeling multibody systems with uncertainties. Part II: Numerical applications," *Multibody Syst Dyn,* vol. 15, pp. 241-262, 2006.

[24] J. Hays, "Parametric Optimal Design Of Uncertain Dynamical Systems," Ph.D Dissertation, Mechanical Engineering, Virginia Tech, Blacksburg, 2011.

[25] E. D. Blanchard, "Polynomial Chaos Approaches to Parameter Estimation and Control Design for Mechanical Systems with Uncertain Parameters," Ph.D Dissertation, Mechanical Engineering, Virginia Tech, Blacksburg, Virginia, 2010.

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

[27] 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.

[28] S. C. Southward, "REAL-TIME PARAMETER ID USING POLYNOMIAL CHAOS EXPANSIONS," *ASME International Mechanical Engineering Congress and Exposition,* 2007.

[29] B. L. Pence, "Recursive Bayesian Parameter Estimation Using Polynomial Chaos Theory Applied to Vehicle Mass Estimation for Rough Terrain," Ph.D, Mechanical Engineering, University of Michigan, 2011.

[30] B. Pence, J. Hays, H. Fathy, C. Sandu, and J. Stein, "Vehicle Sprung Mass Estimation for Rough Terrain," *International Journal of Vehicle Design,* vol. 61, pp. 3-26, 2013.

[31] S. K. Shimp, "Vehicle Sprung Mass Parameter Estimation Using an Adaptive Polynomial-Chaos Method," Masters, Mechanical Engineering, Virginia Tech, Blacksburg, 2008.

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

[33] H. Cheng and A. Sandu, "Efficient uncertainty quantification with the polynomial chaos method for stiff systems," *Mathematics and Computers in Simulation,* vol. 79, pp. 3278-3295, 2009.

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

[35] 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.

[36] W.-H. Hucho and G. Souvran, "Aerodynamics of Road Vehicles," *Annual Review of Fluid Mechanics,* vol. 25, pp. 485-537, 1993.

[37] J. C. Dixon, *Tires, Suspension, and Handling*: Society of Automotive Engineers, 1996.

[38] T. D. Gillespie, *Fundamentals of Vehicle Dynamics*: Society of Automotive Engineers Inc, 1992.

[39] R. N. Jazar, *Vehicle Dynamics: Theory and Application*: Springer, 2008.

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

[41] R. Rajamani, D. Piyabongkarn, V. Tsourapas, and J. Y. Lew, "Real-Time Estimation of Roll Angle and CG Height for Active Rollover Prevention Applications," *American Control Conference,* 2009.

[42] R. Whitehead, W. Travis, D. M. Bevly, and G. Flowers, "A Study of the Effect of Various Vehicle Properties on Rollover Propensity," *SAE Technical Paper,* 2004.

[43] S. Solmaz, M. Akar, R. Shorten, and J. Kalkkuhl, "Real-time multiple-model estimation of centre of gravity position in automotive vehicles," *Vehicle System Dyn.,* vol. 46, pp. 763-788, 2008.

[44] D. Cao, S. Rakheja, and C.-Y. Su, "Heavy Vehicle Pitch Dynamics and Suspension Tuning. Part I: Unconnected Suspension," *Vehicle System Dynamics,* vol. 46, 2008.

[45] S. Solmaz, M. Akar, and R. Shorten. (2008, Center of Gravity Estimation and Rollover Prevention Using Multiple Models & Controllers.

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

[47] B. L. Pence, H. K. Fathy, and J. L. Stein, "A Base -Excitation Approach to Polynomial Chaos-Based Estimation of Sprung Mass for Off-Road Vehicles," presented at the ASME Dynamic Systems and Control Conference, Hollywood, California, 2009.

[48] W. Gao, N. Zhang, and J. Dai, "A Stochastic Quater-Car Model for Dynamic Analysis of Vehicles With Uncertain Parameters," *Vehicle system Dynamics: International Journal of Vehicle Mechanics and Mobility,* vol. 46, 2008.

[49] J. Yoon, D. Kim, and K. Yi, "Design of a Rollover Index-based Vehicle Stability Control Scheme," *Vehicle System Dynamics,* vol. 45, 2007.

[50] M. L. McIntyre, T. J. Ghotikar, A. Vahidi, X. Song, and Darren M. Dawson, "A Two-Stage Lyapunov-Based Estimator for Estimation of Vehicle Mass and Road Grade," *IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY,* vol. 58, 2009.

[51] A. Vahidi, A. Stefanopoulou, and H. Peng, "Recursive Least Squares with Forgetting for Online Estimation of Vehicle Mass and Road Grade: Theory and Experiments," *International Journal of Vehicle Mechanics and Mobility,* vol. 43, pp. 31-55, 2005.

[52] A. Koehl, M. Boutayeb, H. Rafaralahy, and B. Martinez, "Wind-Disturbance and Aerodynamic Parameter Estimation of an Experimental Launched Micro Air Vehicle Using an EKF-like Observer," *IEEE Converence on Decision and Control,* 2010.

[53] H. Cheng and A. Sandu, "Collocation least-squares polynomial chaos method," presented at the Proceedings of the 2010 Spring Simulation Multiconference, Orlando, Florida, 2010.

[54] S. J. Julier and J. K. Uhlmann, "A New Extension of the Kalman Filter to Nonlinear Systems," in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series,* 1997.

[55] J. Li and D. Xiu, "A Generalized Polynomial Chaos Based Ensemble Kalman Filter with High Accuracy," *Journal of Computational Physics,* vol. 228, 2009.

[56] N. Wiener, "The Homogeneous Chaos," *American Journal of Mathematics*, vol. 60, pp. 897-936, 1938.

[57] L. Ljung, *System Identification: Theory for the User*: Prentice Hall PTR, 1999.

[58] S. Weisburg, *Applied Linear Regression*: Wiley, 1980.

[59] P. S. Els, "The ride comfort vs. handling compromise for off-road vehicles," Ph.D Dissertation, Mechanical Engineering, University of Pretoria, Pretoria, South Africa, 2006.

[60] S. Els, "Vehicle and test info," J. Kolansky, Ed., ed, 2012.

Nomenclature

Notation / Quantity	Meaning
Bold Vector	Vector quantity
Superscript	Poly Chaos series term or gPC-EKF iteration
gPC	Generalize Polynomial Chaos
CG	Center of Gravity
RLS	Recursive Least Squares
LS	Least Squares
gPC-EKF	Generalized Polynomial Chaos based Extended Kalman Filter
C_dA	Coefficient of aerodynamic drag times vehicle cross-sectional area
ψ	gPC Basis function series term
Ψ	gPC basis vector
ξ	Random variable ranging from -1 to 1 for spanning the basis function’s space
S	Number of terms in the gPC series expansion
PO	Polynomial order of truncation of the gPC series
t	Time
μ	Basis function random variable value for iteration
Q	Number of iterations for the gPC-EKF
A	Collocation A matrix (matrix of $\psi(\mu)$ points)
z	Sensor measurement
c_i	Scaling coefficient of the addition to the i^{th} gPC-EKF parameter value
I, U	Addition to a gPC-EKF parameter value
N	Noise for the gPC-EKF state variables
m	mass
c	Process noise addition scaling parameter, suspension damping parameter
x_0	State position initial value
v_0	State velocity initial value
R	Error noise covariance matrix
K	Kalman update or RLS update matrix
k	Spring stiffness or term of the K matrix
H	Measurement matrix
$\ddot{Z}, A_z$	Vertical acceleration at the CG
SF_i, F_i	Suspension force
A_y, A_x	Lateral and longitudinal acceleration

J_ϕ,J_θ	Roll inertia, pitch inertia
$\ddot{\phi}, \ddot{\theta}$	Roll acceleration, pitch acceleration
ϕ	Roll angle
l	Distance to the CG from the left side
r	Distance to the CG from the right side
a	Distance to the CG from the front
b	Distance to the CG from the rear
D_s	Suspension displacement
$\bar{m}$	Summed mass of each corner mass estimate
H	CG height
fl	Front left
fr	Front right
rl	Rear left
rr	Rear right
θ	Pitch angle or RLS parameter matrix
η	Suspension mapping coefficient or RLS observation matrix
y	Measurement
v, ν	Process noise vector or sensor noise for RLS
P	Covariance matrix
λ	RLS forgetting factor
Beta	Beta distribution
TS	Time step
h_i	Road height at each wheel
$z_{u,i}$	Unsprung height for each wheel
L	Wheelbase length
B	Trackwidth
T	Torque
RCH	Roll center height
PCH	Pitch center height
$A_{z,i}$	Vertical acceleration at each suspension strut of the vehicle
ΔW	Weight transfer
h_r, h_{rt}	Distance between CG and roll center
h_p	Distance between CG and pitch center
V_x	Longitudinal velocity
C	$C_d A$
m_i	Mass at each corner
$F_{ground,i}$	Normal force at tire contact patch for each wheel
h_{pc}, i	Pitch center height parameter for each wheel
$h_{rc,i}$	Roll center height parameter for each wheel
$D_{s,i}$	Displacement at the i^{th} strut

h_s	Averaged suspension displacement
RC	Rollover Coefficient
W_l, W_r	Weight on left or right side
W	Vehicle weight
F_l, F_r	Force on left or right side