

A polynomial chaos approach to the analysis of vehicle dynamics under uncertainty

Gaurav Kewlani^{a*}, Justin Crawford^b and Karl Iagnemma^a

^a*Department of Mechanical Engineering, Massachusetts Institute of Technology, 41-205, 77 Massachusetts Avenue, Cambridge, MA 02139, USA;* ^b*Quantum Signal LLC, Ann Arbor, MI, USA*

(Received 31 July 2011; final version received 3 November 2011)

The ability of ground vehicles to quickly and accurately analyse their dynamic response to a given input is critical to their safety and efficient autonomous operation. In field conditions, significant uncertainty is associated with terrain and/or vehicle parameter estimates, and this uncertainty must be considered in the analysis of vehicle motion dynamics. Here, polynomial chaos approaches that explicitly consider parametric uncertainty during modelling of vehicle dynamics are presented. They are shown to be computationally more efficient than the standard Monte Carlo scheme, and experimental results compared with the simulation results performed on ANVEL (a vehicle simulator) indicate that the method can be utilised for efficient and accurate prediction of vehicle motion in realistic scenarios.

Keywords: stochastic analysis; polynomial chaos; vehicle dynamics; parametric uncertainty

1. Introduction

A basic requirement for unmanned ground vehicle systems operating autonomously in unstructured environments is the ability to efficiently predict their movement over rugged terrain. Most methods for analysis of vehicle dynamics rely on deterministic analysis that assumes accurate knowledge of the vehicle and terrain parameters. In field conditions, however, ground vehicles frequently have access to only sparse and uncertain terrain parameter estimates drawn from sensors such as LIDAR and vision. Moreover, vehicle parameters may be uncertain and/or time-varying due to, for example, fuel consumption and mechanical wear. There is little research that explicitly addresses the challenge of modelling vehicle motion over a given terrain region while considering these parametric uncertainties.

Numerous techniques can be employed to estimate the output(s) for processes that are subject to uncertainty, including interval mathematics, probabilistic methods and fuzzy set theory, among others [1–3]. A traditional method for estimating the probability density function of a system's output response while considering uncertainty is the Monte Carlo method [4,5]. The approach involves sampling values for each uncertain parameter from its uncertainty range (often weighted by its probability of occurrence), followed by model simulation

*Corresponding author. Email: gkewlani@mit.edu

using this parameter set, and then repeating this process many times to obtain the probability distribution of an output metric. A large number of simulation runs are generally required to obtain reasonable results, leading to a (usually) high computational cost. While structured sampling techniques such as Latin hypercube sampling can be used to improve computational efficiency, the gains may be modest for complex problems [6,7].

More recent approaches include the generalised polynomial chaos (gPC) approach (based on Wiener's theory of homogeneous chaos) [8,9] and the stochastic response surface method [10,11]. While the gPC approach has been successfully applied to various problems, including prediction of vehicle dynamics [12–14], it has been shown to perform inadequately for problems containing discontinuities induced by random inputs and for long-term integration. These problems can be overcome through implementation of the multi-element gPC (MEgPC) framework [15]. In this paper, these polynomial chaos-based approaches are presented and their applicability to vehicle dynamic analysis is explored.

This paper is organised as follows. In Sections 2 and 3, respectively, the Monte Carlo and polynomial chaos methods are briefly introduced. The latter approach is then applied to modelling vehicle motion on unstructured terrain in Section 4. The effect of parametric uncertainty is studied and the simulation results are compared for Monte Carlo, gPC and MEgPC approaches. It can be seen that efficient statistical mobility prediction can be achieved using the proposed techniques, and for long-term prediction, the MEgPC approach yields more accurate results than the gPC method. This is followed in Section 5 by a presentation of the simulations performed on ANVEL (a ground vehicle simulator [16], which was extended to study vehicle motion under uncertainty using the above techniques). The results of field experiments are then presented and compared with the simulation results to validate the use of polynomial chaos-based techniques for dynamic vehicle analysis under uncertainty.

2. The Monte Carlo approach

Monte Carlo methods typically involve the analysis of a (usually) large number of simulation runs of an analytical or numerical system model with various combinations of model parameters. These parameters (termed 'input parameters') are sampled from their respective probability distributions, which are assumed to be known (or can be estimated) *a priori*, and multiple simulation runs are conducted using unique sets of the input parameter values to obtain corresponding outputs for each case. An estimate of the probability distribution of a user-defined output metric can then be estimated from this analysis.

In the 'standard' Monte Carlo approach, random sampling of the input parameter distributions is performed. However, to ensure representation of the entire parameter range, a large number of simulations must be performed, often resulting in extensive computational costs. Various methods have been developed for efficient sampling from the input parameter probability distributions, including stratified sampling, importance sampling and Latin hypercube sampling, among others [6,7]. Generally, these methods attempt to ensure that samples are generated from the entire range of the input parameter space while reducing computational costs and are thus an improvement over the standard Monte Carlo method that is based on random sampling. In this work, the simulation results from the Monte Carlo approach are used as a baseline reference to validate the accuracy of results from the polynomial chaos-based approaches.

2.1. Algorithmic implementation

The standard Monte Carlo approach considers functions of the following form:

$$Y = g(X), \quad (1)$$

where \mathbf{g} represents the model under consideration, \mathbf{X} is a vector of uncertain input variables and \mathbf{Y} represents a vector of estimated outputs. A general procedure for the present analysis is as follows.

- (1) Construct a vector \mathbf{X} consisting of n relevant vehicle and/or terrain parameters. To characterise the uncertainty in the elements of \mathbf{X} , define the range and probability distribution for each input parameter, based on corresponding engineering estimates. This defines the input parameter space.

While many forms of the input parameter distribution are possible, in this paper, the parameter values are assumed to be either uniformly or normally distributed and to be uncorrelated (i.e. independent of one another).

- (2) Generate a sample value for each of the n input variables from the corresponding probability distribution. More specifically, a sample set

$$X_j = [x_{j1}, x_{j2}, \dots, x_{jn}] \quad (2)$$

is generated from the input parameter space. This set may be generated randomly or using structured sampling techniques such as stratified sampling, importance sampling or Latin hypercube sampling.

- (3) Evaluate the output response from an analytical or numerical system model under consideration using the values from the input parameter set \mathbf{X}_j as model parameter values.
- (4) Repeat steps (2) and (3) to generate a distribution for the output metric. The number of simulations (N) is chosen to be large enough such that the output distribution converges to a stable value. The probability distribution of the output metric can then be determined, and various statistics such as its estimated expectation, μ , or variance, σ^2 , can be calculated as follows:

$$\mu = \frac{1}{N} \sum_{j=1}^N g(X_j) \quad (3)$$

$$\sigma^2 = \frac{1}{N} \sum_{j=1}^N (g(X_j) - \mu)^2. \quad (4)$$

For the analysis of vehicle motion over unstructured terrain, vehicle and/or terrain parameters are designated as uncertain input parameters. A fundamental assumption of the proposed approach is that while these parameters may not be precisely known, engineering estimates of their distributions are available. This is a reasonable assumption for vehicle physical parameter estimates, since the effects of loading, component wear and parameter uncertainty can generally be bounded with reasonable accuracy. It is also a reasonable assumption for terrain parameter estimates, since many methods exist for coarsely classifying terrain from standard robotic sensors such as LIDAR and vision [17,18].

2.2. Structured sampling techniques

Using standard Monte Carlo methods with random sampling can require extensive computational costs due to the large number of simulation runs required. Prominent structured sampling techniques, such as stratified sampling and Latin hypercube sampling, can lead to an improvement in computational efficiency.

In stratified sampling, the sample space is partitioned into a number of strata, with each stratum having a specified probability of occurrence. Random samples are then drawn from each stratum. While this ensures dense coverage of the parameter space, it requires definition

of the strata and the calculation of their probabilities. Latin hypercube sampling, however, can ensure dense coverage of the range of each input variable (say, n in number) while avoiding the difficulties associated with the stratified sampling technique. It achieves this by exhaustively dividing each input parameter's range into disjoint intervals (say, N in number) of equal probability and then randomly sampling a parameter value from each corresponding interval. This ensures representation from the entire range of each variable. The N values thus obtained for each of the n parameters are combined in a random manner with those from the other parameters until N n -tuplets (N n -dimensional input vectors to be used for multiple simulation runs) are formed. More details can be found in [7].

3. The polynomial chaos approach

More recent approaches to stochastic simulation of processes subject to uncertainty include the polynomial chaos approach, which is based on Wiener's theory of homogeneous chaos. Since its introduction, polynomial chaos has been extensively applied to fluid and structural mechanics problems [8–11]. According to Wiener, a stochastic process can be represented through a spectral expansion using orthogonal polynomials as

$$y = \sum_{j=0}^{\infty} y_j H_j(\xi), \quad (5)$$

where ξ is a vector of standard normal random variables (i.e. with zero mean and unit variance), H_j is the Hermite polynomial of order j and y_j is the corresponding deterministic coefficient to be calculated from a limited number of model simulations. The Hermite polynomial of degree q is given as

$$H_q(\xi_1, \xi_2, \dots, \xi_m) = (-1)^q e^{(1/2)\xi^T \xi} \cdot \frac{\partial^q}{\partial \xi_1 \cdot \partial \xi_2 \cdots \partial \xi_m} \cdot e^{-(1/2)\xi^T \xi}. \quad (6)$$

For this formulation, optimal convergence is obtained only for Gaussian stochastic processes. Recent extensions to the framework [9] have shown, however, that optimal convergence can be achieved for more general stochastic phenomena as well. The fundamental principle behind this is that random processes of interest can be reasonably approximated using orthogonal polynomial basis functions of the Askey scheme (in terms of the corresponding random variable). This allows treatment of a much broader range of stochastic problems. The generalised Askey polynomial chaos framework is discussed next.

3.1. The gPC method

The gPC method involves representing inputs and outputs of a system under consideration via series approximations using standard random variables, thereby resulting in a computationally efficient means for uncertainty propagation through complex numerical models. In this approach, the same set of random variables that is used to represent input stochasticity is used for representation of the output(s). An equivalent reduced model for the output can thus be expressed in the form of a series expansion consisting of orthogonal polynomials (of the Askey scheme) in terms of the corresponding multi-dimensional random variable as

$$y = \sum_{j=0}^{\infty} y_j \Phi_j(\xi), \quad (7)$$

where y refers to an output metric, $\xi = [\xi_{i1}, \dots, \xi_{im}]$ is the multi-dimensional random variable, $(\xi_{i1}, \xi_{i2}, \dots)$ are i.i.d. (independent, identically distributed) random variables,

$\Phi_j(\xi_{i1}, \xi_{i2}, \dots, \xi_{im})$ is the generalised Askey–Wiener polynomial chaos of degree j and y_j is the corresponding coefficient.

While for normal random variables Hermite polynomials are the basis functions, different orthogonal polynomial basis functions are used corresponding to the probability distributions of other non-normal variables. These are given in Table 1. Expressions for some standard normalised polynomial basis functions are also included in Appendix 1. The principle behind the pairing is that these orthogonal polynomials of the Askey scheme have the weighting function in their orthogonality relation identical to the probability density function of the corresponding random distribution [9].

Table 1. Polynomial basis functions and corresponding random variables.

Random variable	Polynomial Function
Gaussian	Hermite
Gamma	Laguerre
Beta	Jacobi
Uniform	Legendre

For notational simplicity, the series can be truncated to a finite number of terms and rewritten as

$$y = \sum_{j=0}^{N_q} y_j \Phi_j(\xi), \quad (8)$$

where $N_q + 1 = (q + p)! / (q!p!)$, q is the number of random variables and p is the maximum order of the polynomial basis.

The unknown coefficients in the expansion can be determined by projecting each state variable onto the polynomial chaos basis (i.e. using the Galerkin projection method) [9].

An alternative approach that is computationally more efficient is the probabilistic collocation method, where coefficient values are estimated from a limited number of model simulations, to generate the approximate reduced model [19]. This method imposes the requirement that the estimates of model outputs be exact at a set of selected collocation points, thus making the residual at those points equal to zero. The unknown coefficients are thus estimated by equating model outputs and the corresponding polynomial chaos expansion at this set of collocation points in the parameter space; the number of collocation points is equal to the number of unknown coefficients to be found. Thus, for each output metric, a set of linear equations are formed with the coefficients as the unknowns, which can be readily solved. If the governing equations are highly complex, the simplicity of the collocation framework results in a faster algorithm, particularly for high-dimensional problems.

While the accuracy of the gPC approach can be improved by increasing the polynomial order, it should also be noted that as the number of inputs and the expansion order increase, the number of unknown coefficients to be determined increases exponentially, thereby increasing the computational costs. The procedure describing the application of the gPC method is discussed below in more detail.

Algorithmic implementation

A summary of the gPC procedure, as applied to vehicle dynamic analysis, is presented below:

- (1) Represent uncertain input parameters in terms of standard random variables. An uncertain terrain and/or vehicle parameter X_j can thus be written as

$$X_j = \mu_j + \sigma_j \xi_j, \quad (9)$$

where μ_j is the mean, σ_j is a constant (and represents the standard deviation when X_j is normally distributed) and ξ_j is a standard random variable (i.e. the standard normal random variable $N(0,1)$ when X_j is normally distributed).

- (2) Express the model output (y) under consideration in terms of the same set of random variables as

$$y = \sum_{j=0}^{N_q} y_j \Phi_j(\xi). \quad (10)$$

- (3) Estimate the unknown coefficients of the approximating series expansion. This is accomplished here by computing the model output at a set of collocation points [19,20]. This results in a set of equations that can then be used to obtain the coefficient values.

$$\begin{pmatrix} \Gamma_0(\xi_0) & \Gamma_1(\xi_0) & \cdots & \Gamma_N(\xi_0) \\ \Gamma_0(\xi_1) & \Gamma_1(\xi_1) & \cdots & \Gamma_N(\xi_1) \\ \vdots & \vdots & & \vdots \\ \Gamma_0(\xi_M) & \Gamma_1(\xi_M) & & \Gamma_N(\xi_M) \end{pmatrix} \begin{pmatrix} y_0(t) \\ y_1(t) \\ \vdots \\ y_N(t) \end{pmatrix} = \begin{pmatrix} y(t, \xi_0) \\ y(t, \xi_1) \\ \vdots \\ y(t, \xi_M) \end{pmatrix}. \quad (11)$$

In the present analysis, the efficient collocation method (ECM) introduced in [10] is used.

- (4) Once the reduced-order model is formulated, the mean and variance can be directly obtained [20] as

$$\mu = y_0 \Phi_0(\xi) \quad (12)$$

$$\sigma^2 = \sum_{j=1}^{N_q} y_j^2 \langle \Phi_j^2 \rangle. \quad (13)$$

Since $\langle \Phi_j^2 \rangle = 1$ for orthonormal polynomials, Equations (12) and (13) give

$$\mu = y_0 \quad (14)$$

$$\sigma^2 = \sum_{j=1}^{N_q} y_j^2. \quad (15)$$

The advantage of using the gPC method is that the number of model simulations is greatly reduced relative to the Monte Carlo method, thereby reducing computational cost. However, the technique is known to fail for long-term integration, losing its optimal convergence behaviour and developing large error levels [15]. This behaviour can be somewhat mitigated by increasing the expansion order; however, this approach is undesirable for several reasons. First, the computational cost of the gPC method generally increases with increasing polynomial order. More importantly, increasing the polynomial order only postpones error growth. For a fixed polynomial degree, error levels will become increasingly large over time. Hence, continuing to increase the integration time will require an ever-increasing polynomial degree, which is not feasible in practice. The MEgPC technique, however, has been shown to solve these long-term integration issues faced in the gPC framework [15]. This is briefly discussed in the next section.

3.2. The MEgPC method

Wan and Karniadakis [15] have shown that if the domain of random inputs is subdivided into multiple elements, the accuracy of stochastic solutions can be improved, especially for cases

with discontinuities in stochastic solutions or for problems involving long-term integration. As a result, integration error at each time step can be reduced and the domain of solutions' discontinuity can be approximated more accurately within a smaller decomposed domain. Furthermore, a (relatively) lower order polynomial can be used in each random element since the local degree of perturbation has been scaled down, thereby enhancing the accuracy of solutions for long-term integration. Thus, while the range of application of the gPC method is limited (since the polynomial order cannot be increased arbitrarily high in practice), using the MEgPC method allows this range to be extended. In this work, the MEgPC approach (as in [15]) has been employed and the ECM has been utilised (over the Galerkin projection approach) to result in a computationally more efficient algorithm. The general procedure is outlined below.

Decomposition of random space

Let $\xi = [\xi_1, \xi_2, \dots, \xi_n]$ denote an n -dimensional random input vector, where ξ_i is (say) an i.i.d. uniform random variable, $U[-1, 1]$. Next, decompose the domain of the random input into N non-intersecting elements. The domain of each element is contained within a hypercube, $[a_1^k, b_1^k) \times [a_2^k, b_2^k) \times \dots \times [a_n^k, b_n^k)$, where a_i^k and b_i^k denote the lower and upper bounds of the local random variable ζ_i^k ($i = 1, 2, \dots, n$ and $k = 1, 2, \dots, N$). Then, define a local random vector within each element as $\zeta^k = [\zeta_1^k, \zeta_2^k, \dots, \zeta_n^k]$ and subsequently map it to a new random vector in $[-1, 1]^n$: $\xi^k = g_k(\zeta^k) = [\xi_1^k, \xi_2^k, \dots, \xi_n^k]$. This mapping is governed by the following relationship:

$$g_k(\zeta^k) : \zeta_i^k = \frac{(b_i^k - a_i^k)\xi_i^k}{2} + \frac{(b_i^k + a_i^k)}{2}. \quad (16)$$

Consequently, the gPC framework can be used locally to solve a system of differential equations, with the random inputs as ξ^k instead of ζ^k , to take advantage of orthogonality and related efficiencies by employing Legendre chaos. The global mean and variance can then be reconstructed once local approximations of the mean and the variance are obtained.

Algorithmic implementation

Decomposition of the random space can be done either *a priori* or adaptively. In the adaptive scenario, splitting of the random space occurs only when the local decay rate of the error of the gPC approximation η_k (see Equation (21)) exceeds a threshold value. The general procedure is briefly discussed here. Refer to [15] for more details.

Let the gPC expansion in random element k be given as

$$y_k = \sum_{j=0}^{N_g} y_j^k \Phi_j(\xi). \quad (17)$$

The approximated global mean and variance can then be written as

$$\mu_y = \sum_{k=1}^N y_0^k V_k, \quad (18)$$

where $V_k = \prod_{i=1}^n (b_i^k - a_i^k)/2$, and

$$\sigma_y^2 = \sum_{k=1}^N (\sigma_k^2 V_k + (y_0^k - \mu_y)^2 V_k), \quad (19)$$

where the local variance estimated by polynomial chaos (using orthonormal basis functions) is obtained as

$$\sigma_k^2 = \sum_{j=1}^{N_q} (y_j^k)^2. \quad (20)$$

To include adaptive decomposition of the random space, first, define η_k as

$$\eta_k = \frac{\sum_{j=N_{q-1}+1}^{N_q} (y_j^k)^2}{\sigma_k^2}. \quad (21)$$

Then, split a random element if the following criterion is satisfied:

$$(\eta_k)^\alpha V_k \geq \theta_1, \quad 0 < \alpha < 1, \quad (22)$$

where θ_1 is a suitable threshold parameter and α is a constant.

Another parameter θ_2 may be used to choose the more sensitive random dimensions for decomposition, as in [15]. For this, first, the sensitivity of each random dimension is defined as

$$r_i = \frac{(y_{i(m)}^k)^2}{\sum_{j=N_{m-1}+1}^{N_m} (y_j^k)^2}, \quad i = 1, \dots, N, \quad (23)$$

where $y_{i(m)}^k$ denotes the mode consisting only of random dimension ξ_i^k with polynomial order m . Only the random dimensions which satisfy the condition

$$r_i \geq \theta_2 \cdot \max_{j=1, \dots, N} r_j, \quad 0 < \theta_2 < 1, \quad i = 1, \dots, N, \quad (24)$$

are split into two equal random elements in the next time step, while the other random dimensions stay unchanged. This reduces the total element number while gaining efficiency.

A critical numerical implementation involves assigning the initial condition after splitting the random dimension into multiple elements. This may be done as follows.

First, represent the polynomial expansion of the current random field as

$$\hat{y}(\hat{\xi}) = \sum_{j=0}^{N_q} \hat{y}_j \Phi_j(\hat{\xi}). \quad (25)$$

Once the random space is split, let the next level expansion be denoted as

$$\tilde{y}(\tilde{\xi}) = \tilde{y}(g(\hat{\xi})) = \sum_{j=0}^{N_q} \tilde{y}_j \Phi_j(\tilde{\xi}). \quad (26)$$

To calculate the $N_q + 1$ coefficients in this new representation, choose an equal number of uniform grid points in $[-1, 1]^n$ and solve the linear system:

$$\begin{bmatrix} \Phi_{00} & \Phi_{10} & \cdots & \Phi_{N_q 0} \\ \Phi_{01} & \Phi_{11} & \cdots & \Phi_{N_q 1} \\ \vdots & \vdots & \cdots & \vdots \\ \Phi_{0N_q} & \Phi_{1N_q} & \cdots & \Phi_{N_q N_q} \end{bmatrix} \begin{bmatrix} \tilde{y}_0 \\ \tilde{y}_1 \\ \vdots \\ \tilde{y}_{N_q} \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^{N_q} \hat{y}_i \Phi_i(g^{-1}(\tilde{\xi}_0)) \\ \sum_{i=0}^{N_q} \hat{y}_i \Phi_i(g^{-1}(\tilde{\xi}_1)) \\ \vdots \\ \sum_{i=0}^{N_q} \hat{y}_i \Phi_i(g^{-1}(\tilde{\xi}_{N_q})) \end{bmatrix}, \quad (27)$$

where $\Phi_{ij} = \Phi_i(\hat{\xi}_j)$.

4. Vehicle dynamic analysis

The dynamic response of two separate vehicle models was analysed subject to parameter uncertainty. These models are representative of models that might be used during a vehicle design phase or during on-line predictive motion planning of an autonomous vehicle. First, a two-degree-of-freedom quarter-car model under uncertainty was studied, followed by the analysis of a three-degree-of-freedom vehicle model traversing uneven terrain. A standard Monte Carlo analysis was also performed to act as a baseline reference to evaluate the accuracy and efficiency of the polynomial chaos approaches.

4.1. Analysis of a quarter-car model

A two-degree-of-freedom quarter-car model of a vehicle suspension was considered (Figure 1) for which the sprung mass, m_s , and the unsprung mass, m_u , are connected by a nonlinear spring of stiffness k_s and a linear damper with damping coefficient c . An input was applied through a forcing function $z(t)$, to m_u , through a linear spring k_u . This represents the interaction of the quarter-car system with the terrain. The governing equations for the quarter-car system are given as

$$m_s \ddot{x}_1 = -k_s(x_1 - x_2)^3 - c(\dot{x}_1 - \dot{x}_2) \quad (28)$$

$$m_u \ddot{x}_2 = k_s(x_1 - x_2)^3 + c(\dot{x}_1 - \dot{x}_2) + k_u(z(t) - x_2). \quad (29)$$

Displacements of the two masses under parametric uncertainty were analysed, using the gPC and MEgPC approaches, and compared with results from a baseline standard Monte Carlo (SMC) analysis with random sampling.

Parametric uncertainty arises from the suspension stiffness. Now, for such parameters, there is random variation within a range where only the bounds are known from engineering estimates and thus a uniform or normal distribution can be used. The two springs were, therefore, considered to have uncertain spring constant values, uniformly distributed about a mean stiffness value. Normal distribution about the mean stiffness value was also considered. This can be represented as

$$k_s = \mu_{k_s} + \xi_1 \sigma_{k_s} \quad (30)$$

$$k_u = \mu_{k_u} + \xi_2 \sigma_{k_u}. \quad (31)$$

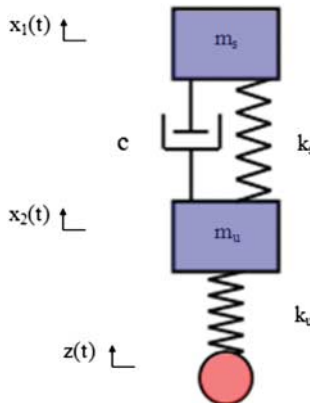


Figure 1. Quarter-car model.

Displacements were then expressed as a series expansion of Legendre polynomials of standard uniform random variables ξ_1 and ξ_2 (Hermite polynomials were used when normal distribution was assumed), and the system was analysed for various terrain inputs. In general, the state may be expressed as

$$\mathbf{X} = [x_1 x_2 \dot{x}_1 \dot{x}_2]^T \quad (32)$$

$$x_i(t, \xi) = \sum_{j=0}^P x_{i,j}(t) \Phi_j(\xi) \quad i = 1, 2 \quad (33)$$

$$\dot{x}_i(t, \xi) = \sum_{j=0}^P \dot{x}_{i,j}(t) \Phi_j(\xi) \quad i = 1, 2, \quad (34)$$

where $\xi = [\xi_1, \xi_2]$.

The steps involved in the application of the gPC method to this system are discussed in detail in Appendix 2. Stiffness parameter values used here are given in Table 2.

Table 2. Uncertain parameters in quarter-car model.

Parameter	μ	$\sigma_U (2\sigma_N)$
k_s	400 N/m ³	40 N/m ³
k_u	2000 N/m	200 N/m
m_s	20 kg	—
m_u	40 kg	—
c	600 Ns/m	—

4.1.1. Simulation results

For a step input with a step size of 0.2 m (which models traversal over a bump or obstacle), it was observed that parametric uncertainty causes significant variation in the resulting output of x_1 , the sprung mass displacement (Figure 2), thus indicating the importance of considering uncertainty during dynamic analysis.

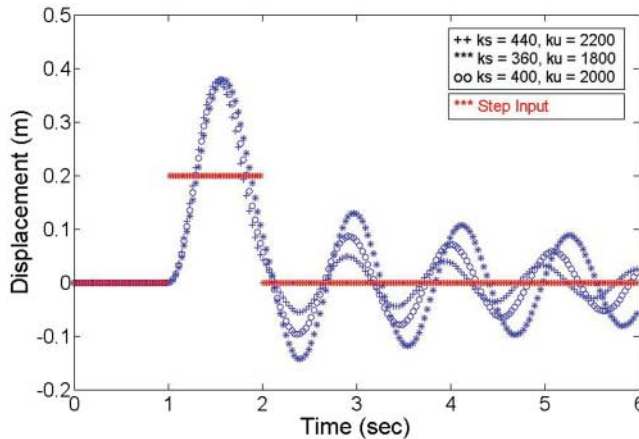


Figure 2. Sprung mass displacement (x_1) for various stiffness and damping values when subject to step input.

The time profile obtained for the standard deviation of the displacement of x_1 (i.e. the sprung mass) for a step input (with uniformly distributed uncertain parameter values) is shown in Figure 3 for the gPC (with $P = 3$), MEgPC (with $P = 3, \theta_1 = 0.001, \alpha = 0.5$) and SMC methods. It can be observed that even for relatively short times, there is substantial difference between the predicted standard deviation from the two techniques, with the MEgPC method yielding more precise results than the gPC method when compared with the baseline SMC analysis. (This difference decreases with time solely due to the nature of the input considered.) The corresponding results for normally distributed uncertain parameter values are shown in Figure 4.

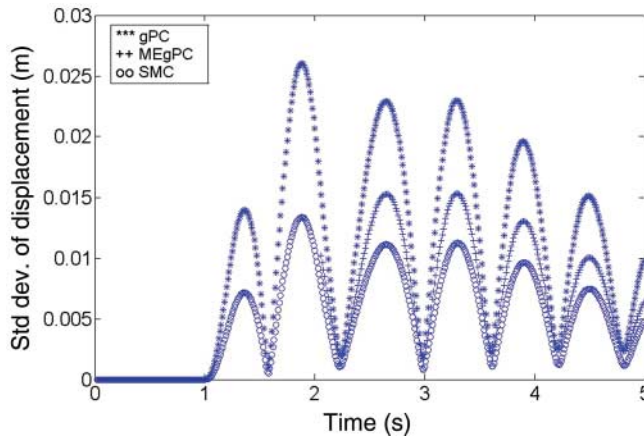


Figure 3. Standard deviation of sprung mass displacement (x_1) when subject to step input (uniformly distributed uncertain parameters).

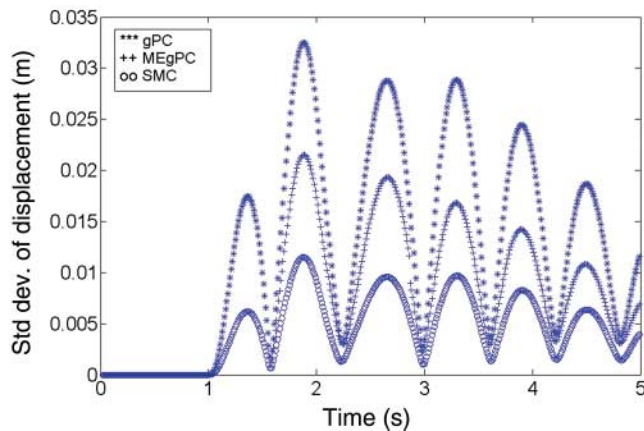


Figure 4. Standard deviation of sprung mass displacement (x_1) when subject to step input (normally distributed uncertain parameters).

Table 3 presents the computation times for the SMC and MEgPC approaches with respect to the gPC method, corresponding to the results shown in Figure 3. It can be seen that the computation time of the gPC analysis is approximately 200 times less than that of the SMC analysis.

Next, the system response was analysed when subjected to a sinusoidal terrain input with an amplitude of 0.1 m and period of 1.0 s. The results are shown in Figures 5–7. As expected,

Table 3. Computation time for the various approaches for system subject to step input.

Method	Ratio of simulation time
SMC (2000 runs)	191.96
MEgPC	1.585
gPC	1.00

some inconsistency was found in the predictions made using the gPC approach (with $P = 3$). The MEgPC approach (with $P = 3, \theta_1 = 0.001, \alpha = 0.5$), however, exhibited only a small bounded error over time. Furthermore, for the sinusoidal input, this difference in the standard deviation prediction did not decay as in the previous case. The computation times for the SMC and MEgPC approaches with respect to the gPC method corresponding to the results shown in Figure 6 are given in Table 4.

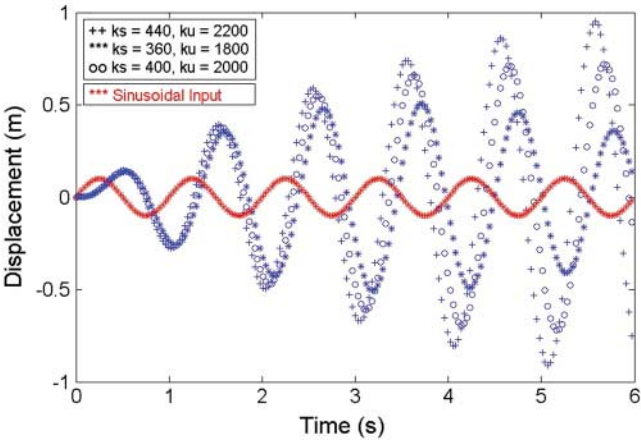


Figure 5. Sprung mass displacement (x_1) for various stiffness and damping values when subject to sinusoidal input.

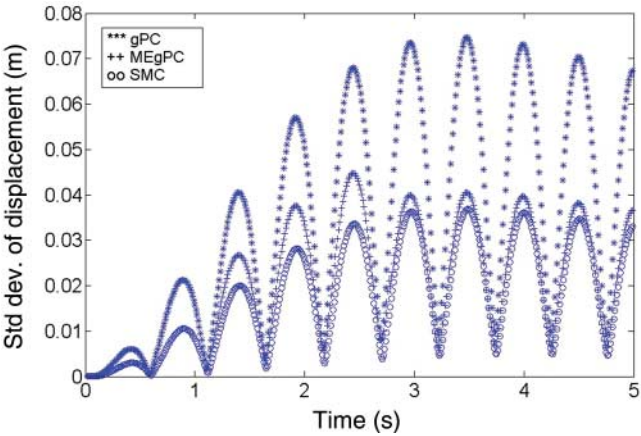


Figure 6. Standard deviation of sprung mass displacement (x_1) when subject to sinusoidal input (uniformly distributed uncertain parameters).

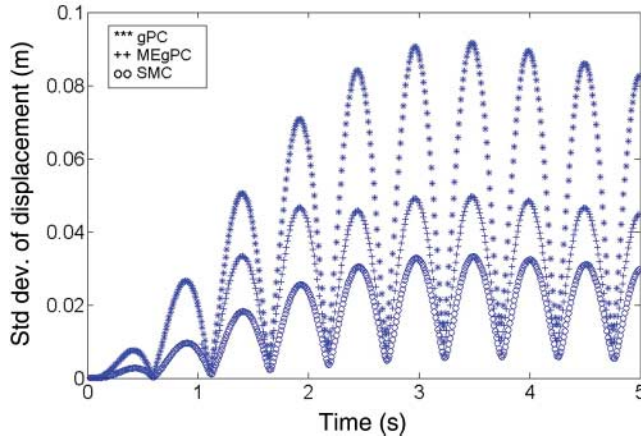


Figure 7. Standard deviation of sprung mass displacement (x_1) when subject to sinusoidal input (normally distributed uncertain parameters).

Table 4. Computation time for the various approaches for system subject to sinusoidal input.

Method	Ratio of simulation time
SMC (2000 runs)	197.76
MEgPC	2.33
gPC	1.00

These results show the applicability of polynomial chaos-based approaches (especially the MEgPC approach) and highlight their efficient performance over the conventional Monte Carlo-based technique in terms of reduced computational costs.

4.2. Analysis of a full-vehicle model

Next, a three-degree-of-freedom vehicle model (Figure 8) that includes lateral acceleration, yaw and roll dynamics [21] was considered. Such a model is useful for analysing or predicting the steering behaviour of a vehicle over uneven terrain. The linearised equations for this model are given as

$$mV(\dot{\beta} + \dot{\psi}) - m_s h \ddot{\varphi} = \sum F_y = C_f \left(\delta - \frac{\dot{\psi} l_f}{V} - \beta \right) + C_r \left(\frac{\dot{\psi} l_r}{V} - \beta \right) \quad (35)$$

$$I_{zz} \ddot{\psi} = \sum M_z = C_f \left(\delta - \frac{\dot{\psi} l_f}{V} - \beta \right) l_f - C_r \left(\frac{\dot{\psi} l_r}{V} - \beta \right) l_r \quad (36)$$

$$(I_{xx} + m_s h^2) \ddot{\varphi} = \sum M_x = m_s g h \varphi + m_s h V (\dot{\beta} + \dot{\psi}) + M_s, \quad (37)$$

where $M_s = -(k_f + k_r)\varphi - (b_f + b_r)\dot{\varphi}$.

The parameter values used in this study are given in Table 5.

In addition to the forces from tyre compliance, lateral components of contact forces on the vehicle can arise due to terrain unevenness. Given a terrain elevation map $z(x, y)$, modelled as a continuous, differentiable function of planar position, terrain disturbance force T_i acting at

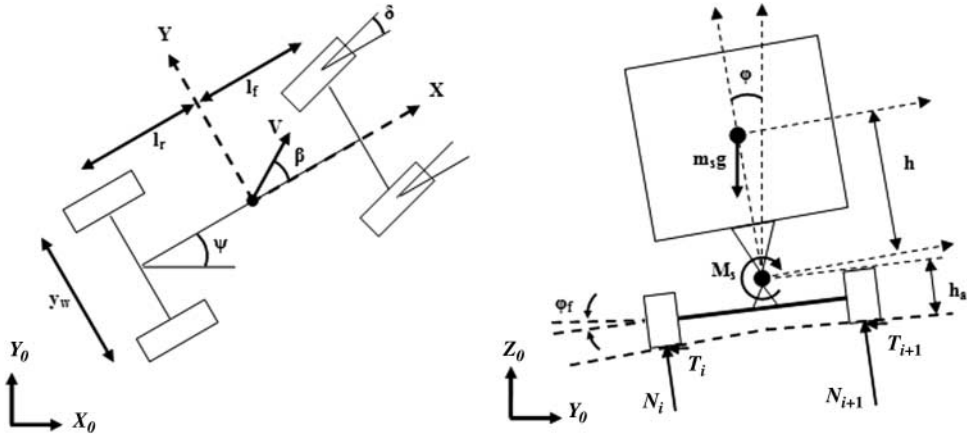


Figure 8. Vehicle model for dynamic analysis under uncertainty.

Table 5. Vehicle model parameters.

Symbol	Description	Mean value	Units
δ	Front wheel steering angle	—	rad
β	Slip angle	—	rad
ϕ	Roll angle	—	rad
ψ	Yaw angle	—	rad
I_{xx}	Roll moment of inertia	834	kg m ²
I_{zz}	Yaw moment of inertia	2050	kg m ²
m	Total vehicle mass	2030	kg
m_s	Sprung mass	1830	kg
V	Longitudinal velocity	10	m/s
h	Height of centre of gravity (CG) from roll axis	0.35	m
h_a	Height of roll axis from ground	0.21	m
y_w	Track width	1.56	m
C_f	Cornering stiffness (lumped front wheels)	1440	N/rad
C_r	Cornering stiffness (lumped rear wheels)	1280	N/rad
l_f	Distance of front axle from CG	0.43	m
l_r	Distance of rear axle from CG	0.33	m
k_f	Front roll stiffness	30,000	Nm/rad
k_r	Rear roll stiffness	30,000	Nm/rad
b_f	Front axle damping rate	3600	Nm s/rad
b_r	Rear axle damping rate	3600	Nm s/rad
g	Acceleration due to gravity	9.8	m/s ²

each wheel can be written as [22]

$$T_i = N_i \left(\left(\frac{\partial z}{\partial x_0} \right) \hat{x}_0 + \left(\frac{\partial z}{\partial y_0} \right) \hat{y}_0 \right) \cdot \hat{y} \quad i = 1, \dots, 4, \quad (38)$$

where N_i is the normal contact force at wheel i , \hat{x}_0 and \hat{y}_0 are unit vectors of the inertial reference frame and \hat{y} is a unit vector lateral to the reference path.

The suspension moment M_s , including the body roll due to terrain unevenness, is given as

$$M_s = -k_f(\phi - \phi_f) - k_r(\phi - \phi_r) - b_f(\dot{\phi} - \dot{\phi}_f) - b_r(\dot{\phi} - \dot{\phi}_r), \quad (39)$$

where ϕ_f and ϕ_r are the terrain roll angles, with $\dot{\phi}_f$ and $\dot{\phi}_r$ as their corresponding rates. To compute these terrain roll angles and rates, it was assumed that the wheels always remain in

contact with the terrain. Then, using the position and velocity of each wheel and the terrain elevation $z(x, y)$, these quantities were calculated as [22]

$$\varphi_i = \frac{(z_{i+1} - z_i)}{y_w} \quad (40)$$

$$\dot{\varphi}_i = \frac{(\dot{z}_{i+1} - \dot{z}_i)}{y_w}, \quad (41)$$

where the rate of elevation change can be computed as

$$\dot{z}_i = V \left(\left(\frac{\partial z}{\partial x} \right) \cos(\psi + \beta) + \left(\frac{\partial z}{\partial y} \right) \sin(\psi + \beta) \right). \quad (42)$$

Finally, the linearised equations for this model can be rewritten as

$$\dot{\beta} = -\frac{GC}{mV}\beta + \left(-1 + \frac{KG}{mV^2}\right)\dot{\psi} + \frac{C_f G}{mV}\delta + \frac{m_s h M_s}{mVI_{xx}^o} + \frac{m_s^2 g h^2}{mVI_{xx}^o}\phi + \frac{G}{mV} \sum T_i \quad (43)$$

$$\ddot{\phi} = \frac{m_s g h}{I_{xx}^o}\phi + \frac{M_s}{I_{xx}^o} - \frac{m_s C h}{mI_{xx}^o}\beta + \frac{m_s K h}{mVI_{xx}^o}\dot{\psi} + \frac{C_f m_s h}{mI_{xx}^o}\delta + \frac{m_s h}{mI_{xx}^o} \sum T_i \quad (44)$$

$$\ddot{\psi} = \frac{K}{I_{zz}}\beta - \frac{D}{VI_{zz}}\dot{\psi} + \frac{C_f l_f}{I_{zz}}\delta + \frac{1}{I_{zz}} \sum T_i l_i, \quad (45)$$

where

$$C = C_f + C_r, K = C_r l_r - C_f l_f, D = C_f l_f^2 + C_r l_r^2, \\ G = 1 + \frac{m_s^2 h^2}{mI_{xx}^o}, I_{xx}^o = I_{xx} + m_s h^2 \left(1 - \frac{m_s}{m}\right).$$

The terrain (Figure 9) in this case was represented as a combination of trigonometric functions:

$$z(x, y) = A_o \sin(0.5y) + B_o \sin(0.5x) + C_o \cos\left(0.5D\left(\sqrt{x^2 + y^2}\right)\right) \\ + E_o \cos(0.5y) + F_o \sin\left(0.5G_o\left(\sqrt{x^2 + y^2}\right)\right) + H_o \cos(0.5x), \quad (46)$$

where $A_o, B_o, C_o, D_o, E_o, F_o, G_o$ and H_o are suitable constants to yield a model of rolling terrain.

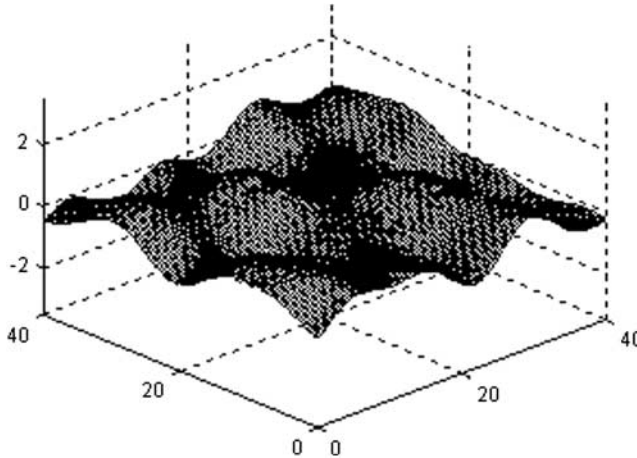


Figure 9. Terrain elevation map used in the analysis.

Then, to introduce uncertainty, roll stiffness parameters were represented as polynomial chaos expansions, using Legendre polynomials of standard uniform random variables ξ_1 and ξ_2 . The front and rear axle roll stiffness values were considered to be uniformly distributed about their mean values. (Normally distributed values resulted in similar predictions.) These may be represented as

$$k_f = \mu_{k_f} + \xi_1 \sigma_{k_f} \quad (47)$$

$$k_r = \mu_{k_r} + \xi_2 \sigma_{k_r}. \quad (48)$$

The state variables can be represented as

$$X_i(t, \xi) = \sum_{j=0}^P X_{i,j}(t) \Phi_j(\xi), \quad i = 1 \dots 7, \quad (49)$$

where $\xi = [\xi_1, \xi_2]$.

The roll stiffness parameter values considered in the study are given in Table 6.

Table 6. Uncertain vehicle parameters in dynamic analysis.

Parameter	M	$\sigma_U (2\sigma_N)$
k_f	$30 \times 10^3 \text{ Nm/rad}$	$4 \times 10^3 \text{ Nm/rad}$
k_r	$30 \times 10^3 \text{ Nm/rad}$	$4 \times 10^3 \text{ Nm/rad}$

4.2.1. Simulation results

Using the expansions in (49), a spectral stochastic analysis [21] was performed to obtain the time evolution of the mean and standard deviation of the roll angle (Figures 10 and 11). It can be seen that while the mean values match closely, the prediction from the gPC approach (with $P = 3$) for the standard deviation differs substantially from the MEgPC and SMC results, even for relatively short times.

Computation times for the SMC and MEgPC approaches with respect to the gPC method, while predicting the roll angle, are given in Table 7.

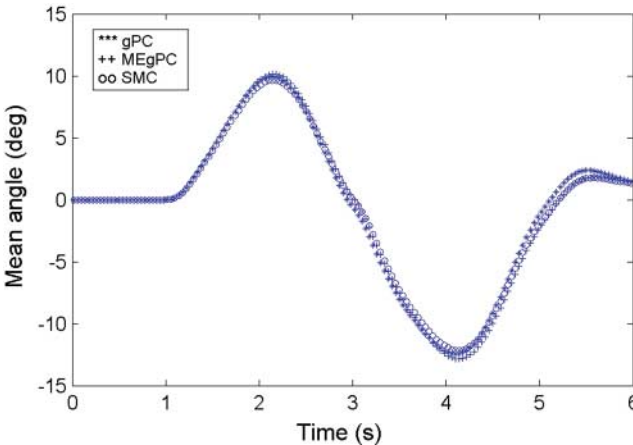


Figure 10. Prediction of mean roll angle.

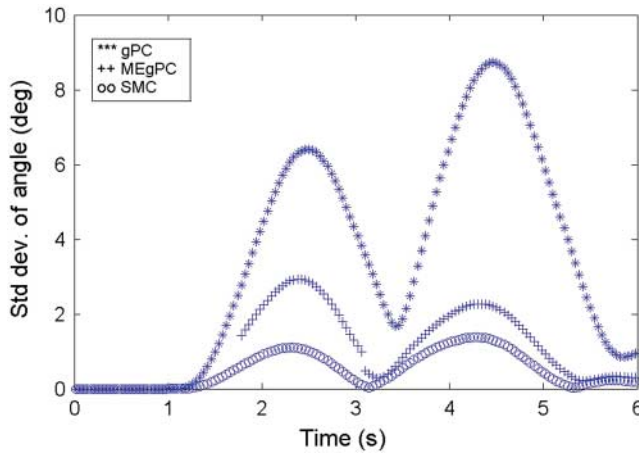


Figure 11. Prediction of standard deviation of roll angle.

Table 7. Computation time for the various approaches.

Method	Ratio of simulation time
SMC (2000 runs)	194.62
MEgPC	5.28
gPC	1.00

The simulation results show that the polynomial chaos-based methods perform more efficiently (approximately 200 times) than the standard Monte Carlo technique in terms of computational cost. Also, the MEgPC technique yields an improvement over the gPC approach in terms of the accuracy of predictions.

5. Validation of the gPC approach

To validate the performance of the gPC technique, the ground vehicle simulator ANVEL (based on open dynamics engine and extended to perform uncertainty analysis) was utilised to compare the results from the experiments conducted with a P3-AT mobile robot, the key technical specifications of which are given in Table 8. A description about the experimental procedure and setup, as well as the vehicle simulator ANVEL, is provided in the subsequent sections.

Table 8. P3-AT key specifications.

Parameter	Units	Value
Translational velocity (max)	m/s	0.7
Vehicle mass	kg	14
Payload mass (max)	kg	40
Fixed payload mass (experiment)	kg	12
Wheel diameter	m	0.22

5.1. Obstacle traversal scenarios

Two obstacle traversal scenarios considered for analysis are described now. Here, uncertainty appears in the form of the height of the obstacle, the surface type (and its corresponding ground friction) and the mass of the vehicle. These are described below in more detail.

5.1.1. Scenario 1

In this case, the vehicle was made to traverse over a single step obstacle in its path. As demonstrated in Figure 12, the vehicle was made to start from rest and attain a set velocity V . It then encountered a step obstacle of height H and width d_2 ($d_2 = 0.5$ m).

The coordinates of the trajectory followed by the vehicle, as well as its roll angle along the path, were tracked using a Vicon spatial motion capture system (<http://www.vicon.com/>). Various runs were performed corresponding to different values of the uncertain parameters, the corresponding ranges of which are given in Table 9.

5.1.2. Scenario 2

In this case, the vehicle was made to traverse over two step obstacles, each of height H and width d_2 , separated by a fixed distance d_3 ($d_3 = 0.2$ m) as shown in Figure 13. The vehicle started from rest and reached a set velocity V . It then encountered the first step obstacle, and after navigating the small gap (ditch), it faced the next obstacle.

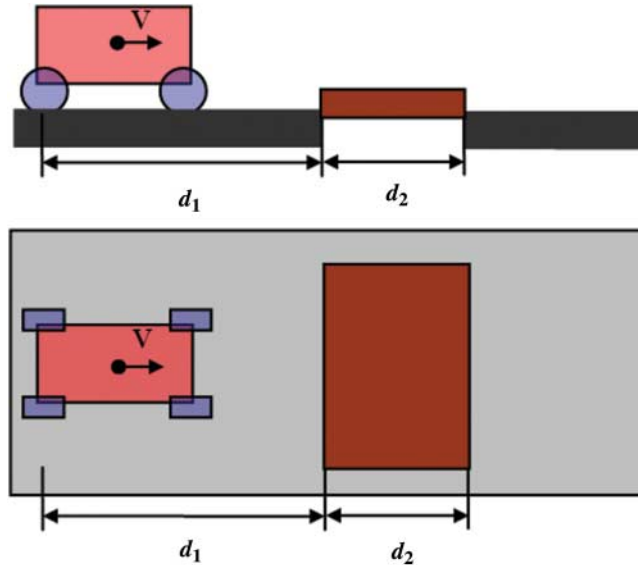


Figure 12. Obstacle placement for scenario 1.

Table 9. Uncertain parameters in dynamic analysis.

Parameter	Units	Minimum	Maximum
Additional mass on vehicle	kg	0	2.6
Height of obstacle (H)	cm	2.5	6.4
Surface type	Plywood, artificial turf, area rug ($\mu = 0.2$ to 0.7)		

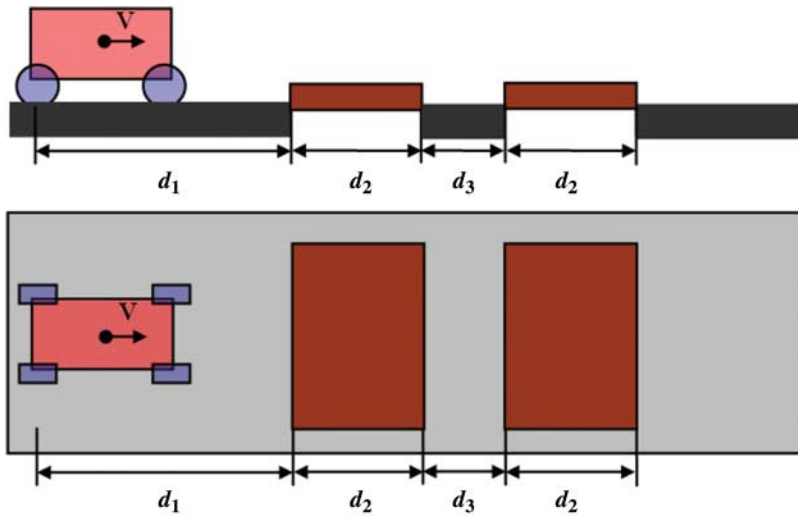


Figure 13. Obstacle placement for scenario 2.

As in the previous case, the coordinates of the trajectory followed by the vehicle, as well as its roll angle along the path, were tracked for each run corresponding to a different set of values for the uncertain parameters. The parameter ranges considered are given in Table 10.

Table 10. Uncertain parameters in dynamic analysis.

Parameter	Units	Minimum	Maximum
Additional mass on vehicle	kg	0	2.6
Height of obstacle	cm	2.5	4.5
Surface type	Plywood, artificial turf, area rug ($\mu = 0.2$ to 0.7)		

5.2. ANVEL overview

ANVEL is a desktop simulation software package based on VANE (Virtual Autonomous Navigation Environment [16]) that enables the analysis and testing of vehicle dynamics and terrain interaction in real time. It leverages physics-based vehicle and vehicle–terrain interaction models in conjunction with high-quality visualisation techniques (Figure 14). The system provides an adaptable and customisable simulation platform that provides developers a controlled, repeatable test bed for simulations. It has also been extended to be able to repeatedly execute simulation runs of a vehicle traversing a terrain, while manipulating a variety of parameters and performing uncertainty analysis. A brief overview of its key features when it is implemented for uncertainty analysis is provided in the next section. More details about the software can be found in [16].

5.2.1. Software plug-ins for uncertainty analysis

The algorithmic structure of ANVEL consists of three integrated parts:

- (i) Pre-processing
- (ii) Dynamic simulations and graphic visualisation
- (iii) Post-processing analysis

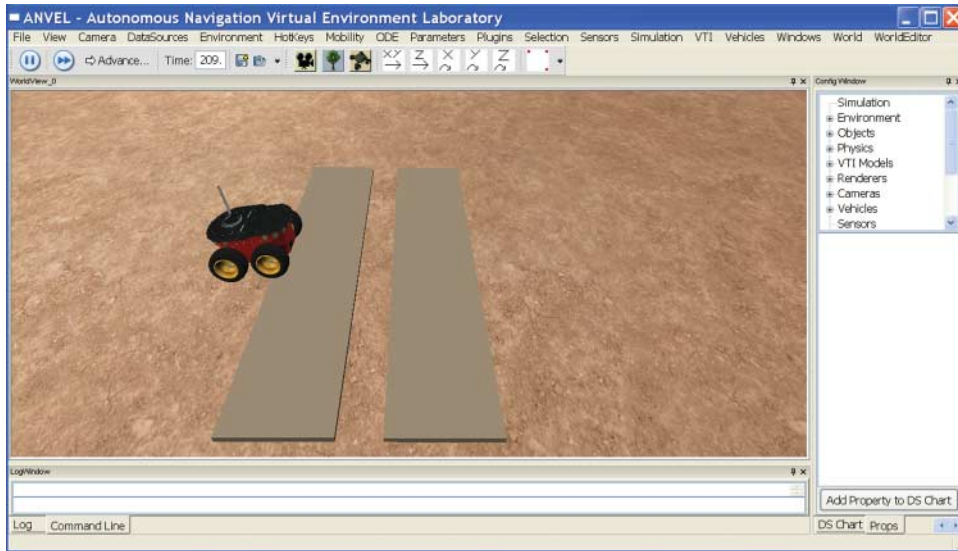


Figure 14. ANVEL desktop simulation software.

The software can be used to evaluate the dynamics of the vehicle during its operation over unstructured and uncertain terrain. For this, the gPC method is utilised to generate a reduced order expansion for the relevant output metrics, which is then used for the analysis of the system under uncertainty.

5.2.2. *Simulation procedure*

During pre-processing, the number of uncertain parameters and the order of the reduced order response surface-based expansion are first chosen. The ranges of values for the uncertain parameters are also defined. Subsequently, pre-simulation calculations are performed and the parameter values (based on collocation points) are sampled for each of the uncertain terrain and/or vehicle variables. This set of sampled values is used for the simulation runs conducted in ANVEL. Once the model runs are performed, results are written to a data file for post-processing. This involves generating the reduced order model for suitably defined output metric(s), such as the path coordinates and roll angle.

Relevant statistics, such as the mean and standard deviation, can then be calculated as the vehicle attempts to move along a reference trajectory in an uncertain environment.

Monte Carlo analysis can also be performed to validate the accuracy of the gPC approach employed in ANVEL. Identifying appropriate levels of complexity of physics-based vehicle models and the polynomial orders of the reduced order expansions, as well as analysing the modelling–performance trade-off, can further be incorporated.

5.3. *Experimental evaluation*

Field experiments were conducted to study vehicle motion under uncertainty using the P3-AT mobile robot system. The setup was designed corresponding to the scenarios described in Section 5.1, and simulations were performed in ANVEL (Figure 15). The Vicon motion tracking system was utilised to obtain the relevant outputs, which were later compared with the simulation results.

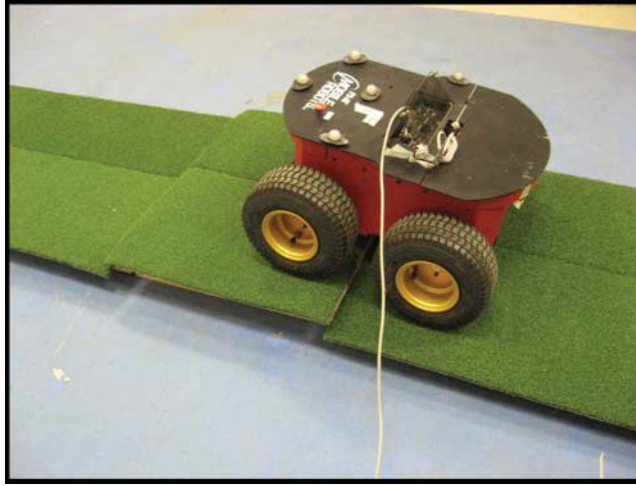


Figure 15. Experiment using P3-AT mobile robot.

5.4. Uncertainty analysis results

The results from the simulations performed in ANVEL and those obtained from the experiments are compared now. It was observed that for the scenarios considered, the results yielded by the experiments matched closely with those obtained from the ANVEL simulations. The slight discrepancy in the results may be attributed to the dynamic modelling of the vehicle in ANVEL differing from the actual vehicle dynamics.

5.4.1. Scenario 1

The variation in roll angle of the vehicle with time, considering uncertainty in obstacle height, vehicle mass and surface type, was studied. The results for $V = 0.4$ m/s are shown in Figure 16. The simulation results from ANVEL are included in Figure 17.

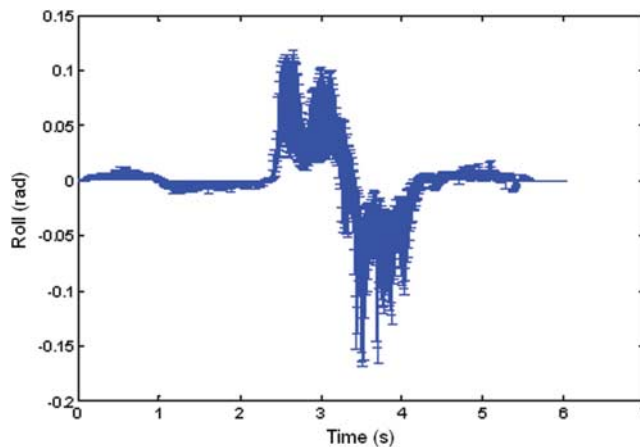


Figure 16. Experimental results for variation of roll angle vs. time ($V = 0.4$ m/s).

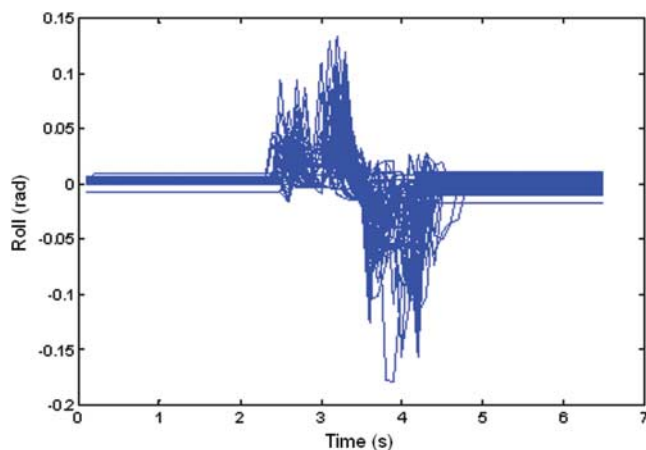


Figure 17. Simulation results for variation of roll angle vs. time ($V = 0.4$ m/s).

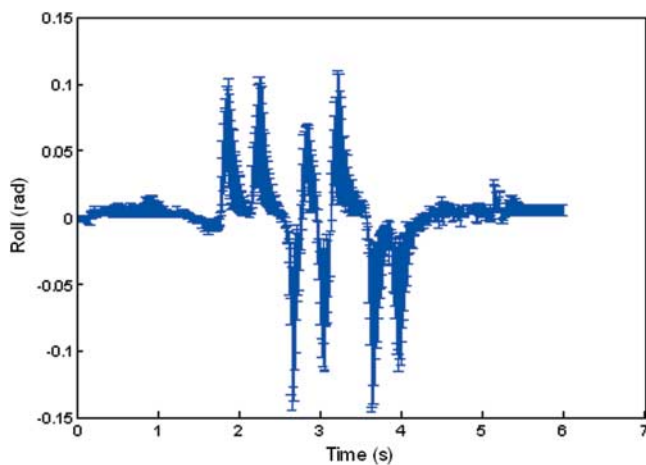


Figure 18. Experimental results for variation of roll angle vs. time ($V = 0.4$ m/s, $d_3 = 0.2$ m).

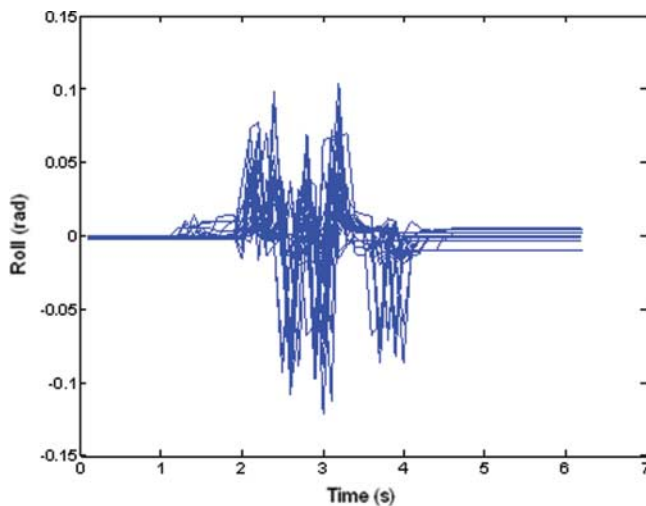


Figure 19. Simulation results for variation of roll angle vs. time ($V = 0.4$ m/s, $d_3 = 0.2$ m)

Although it is difficult to quantitatively compare the simulated and experimental results due to the difficulty in accurately aligning the data in time, it can be observed that the gPC approach (with $P = 3$) yields a reasonable model of the robot performance, even during a complex navigation task.

5.4.2. Scenario 2

Figure 18 shows the variation in the roll angle with time, with uncertainty in obstacle height, vehicle mass and surface type. The results are plotted for $V = 0.4$ m/s and gap width $d_3 = 0.2$ m. The simulation results are shown in Figure 19.

Again, it can be observed that the gPC approach (with $P = 3$) yields a reasonable model of the robot performance during a complex navigation task.

6. Conclusions

This paper has presented an approach to mobile vehicle dynamics prediction based on the gPC framework while explicitly considering uncertainty in vehicle and terrain parameter estimates. The simulation results show that in terms of computational cost, the gPC method performs more efficiently than the Monte Carlo technique. The approach can be applied to real-world situations, as demonstrated through the experiments conducted to validate the results obtained using the ANVEL simulator.

Acknowledgements

This work was supported by the US Army Research Office under contract number W912HZ-08-C-0060.

References

- [1] J.C. Helton, *Uncertainty and sensitivity analysis techniques for use in performance assessment for radioactive waste disposal*, Reliab. Eng. Syst. Safety 42(2–3) (1993), pp. 327–367.
- [2] M. Hanss, *The transformation method for the simulation and analysis of systems with uncertain parameters*, Fuzzy Sets Syst. 130(3) (2002), pp. 277–289.
- [3] P.D. Spanos and M.P. Mignolet, *ARMA Monte Carlo simulation in probabilistic structural analysis*, Shock Vib. Digest 21 (1989), pp. 3–14.
- [4] R.Y. Rubinstein, *Simulation and the Monte Carlo Method*, John Wiley & Sons, Inc., New York, NY, 1981.
- [5] M.H. Kalos and P.A. Whitlock, *Monte Carlo methods. Volume 1: Basics*, Wiley-Interscience, New York, NY, 1986.
- [6] M.D. McKay, *Latin hypercube sampling as a tool in uncertainty analysis of computer models*, in WSC'92: *Proceedings of the 24th Conference on Winter Simulation*, Arlington, Virginia, USA, J.J. Swain, D. Goldsman, R.C. Crain, and J.R. Wilson, eds., ACM, New York, 1992, pp. 557–564.
- [7] J.C. Helton and F.J. Davis, *Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems*, Reliab. Eng. Syst. Safety 81(1) (2003), pp. 23–69.
- [8] R. Ghanem and P.D. Spanos, *Stochastic finite elements: A spectral approach*, Springer-Verlag, New York, 1991.
- [9] D. Xiu and G.E. Karniadakis, *The Wiener–Askey polynomial chaos for stochastic differential equations*, SIAM J. Scient. Comput. 24(2) (2002), pp. 619–644.
- [10] S.S. Isukapalli, *Uncertainty analysis of transport-transformation models*, Ph.D. thesis, Rutgers University, 1999.
- [11] G. Kewlani and K. Iagnemma, *A Stochastic response surface approach to statistical prediction of robotic mobility*, Proceedings of the 2008 International Conference on Robots and Systems, Nice, France, 2008, pp. 2234–2239.
- [12] L. Li and C. Sandu, *On the impact of cargo weight, vehicle parameters, and terrain characteristics on the prediction of traction for off-road vehicles*, J. Terramech. 44(3) (2007), pp. 221–238.
- [13] A. Sandu, C. Sandu, and M. Ahmadian, *Modeling multi body dynamic systems with uncertainties. Part I: Theoretical and computational aspects*, Multibody Syst. Dyn. 15(4) (2006), pp. 1–23.
- [14] C. Sandu, A. Sandu and M. Ahmadian, *Modeling multi body dynamic systems with uncertainties. Part II: Numerical applications*, Multibody Syst. Dyn. 15(3) (2006), pp. 241–262.

- [15] X. Wan and G.E. Karniadakis, *An adaptive multi-element generalized polynomial chaos method for stochastic differential equations*, J. Comput. Phys. 209(2) (2005), pp. 617–642.
- [16] K. Iagnemma, G. Kewlani, D.A. Horner, R.A. Jones, C.L. Cummins, M. Toschlog, J. Crawford, and M.M. Rohde, *An interactive physics-based unmanned ground vehicle simulator leveraging open source gaming technology: Progress in the development and application of the virtual autonomous navigation environment (VANE) desktop*, in *Proceedings of SPIE Conference on Unmanned Systems Technology, Orlando, FL, USA*, Vol. 7332, G.R. Gerhart, D.W. Gage, and C.M. Shoemaker, eds., SPIE, Orlando, FL, 2009.
- [17] N. Vandapel, D.F. Huber, A. Kapuria, and M. Hebert, *Natural Terrain classification using 3-D ladar data*, Proceedings of 2004 International Conference on Robotics and Automation, New Orleans, LA, USA, 2004, pp. 5117–5122.
- [18] R. Manduchi, A. Castano, A. Thalukder, and L. Matthies, *Obstacle detection and terrain classification for autonomous off-road navigation*, Auton. Robots 18 (2005), pp. 81–102.
- [19] S. Huang, S. Mahadevan, and R. Rebba, *Collocation-based stochastic finite element analysis for random field problems*, Probab. Eng. Mech. 22 (2007), pp. 194–205.
- [20] E. Blanchard, A. Sandu, and C. Sandu, *Parameter estimation for mechanical systems using an extended Kalman filter*, Tech. Rep. TR-08-18 (2008), Computer Science Department, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, USA, 2008.
- [21] G. Kewlani and K. Iagnemma, *Mobility prediction for unmanned ground vehicles in uncertain environments*, in *Proceedings of SPIE Conference on Unmanned Systems Technology, Orlando, FL, USA*, Vol. 6962, G.R. Gerhart, D.W. Gage, and C.M. Shoemaker, eds., SPIE, Orlando, FL, 2008.
- [22] S. Peters and K. Iagnemma, *Mobile robot path tracking of aggressive maneuvers on sloped terrain*, Proceedings of 2008 International Conference on Robots and Systems, Nice, France, 2008, pp. 242–247.

Appendix 1

Some standard normalised polynomial basis functions are given below.

1. Hermite polynomials	2. Legendre polynomials
$H_0(x) = 1$	$L_0(x) = 1/\sqrt{2}$
$H_1(x) = x$	$L_1(x) = \sqrt{3/2}x$
$H_2(x) = \frac{1}{\sqrt{2}}(x^2 - 1)$	$L_2(x) = \frac{\sqrt{5}}{2\sqrt{2}}(3x^2 - 1)$
$H_3(x) = \frac{1}{\sqrt{6}}(x^3 - 3x)$	$L_3(x) = \frac{\sqrt{7}}{2\sqrt{2}}(5x^3 - 3x)$
$H_4(x) = \frac{1}{2\sqrt{6}}(x^4 - 6x^2 + 3)$	$L_4(x) = \frac{3}{8\sqrt{2}}(35x^4 - 30x^2 + 3)$
$H_5(x) = \frac{1}{2\sqrt{30}}(x^5 - 10x^3 + 15x)$	$L_5(x) = \frac{\sqrt{11}}{8\sqrt{2}}(63x^5 - 70x^3 + 15)$
$H_6(x) = \frac{1}{12\sqrt{5}}(x^6 - 15x^4 + 45x^2 - 15)$	$L_6(x) = \frac{\sqrt{13}}{16\sqrt{2}}(231x^6 - 315x^4 + 105x^2 - 5)$
$H_7(x) = \frac{1}{12\sqrt{35}}(x^7 - 21x^5 + 105x^3 - 105x)$	$L_7(x) = \frac{\sqrt{15}}{16\sqrt{2}}(429x^6 - 693x^4 + 315x^2 - 35)$

Appendix 2

A.1 Application of the gPC method to a quarter-car model

The gPC method for uncertainty analysis has been extended to solve differential equations related to vehicle dynamics in a stochastic framework. The general procedure for applying the gPC technique to the quarter-car model described in Section 4.1 is discussed below.

Consider the ordinary differential equations (ODEs) in Equations (28) and (29):

$$m_s \frac{d^2 x_1}{dt^2} + k_s (x_1 - x_2)^3 + c \left(\frac{dx_1}{dt} - \frac{dx_2}{dt} \right) = 0 \quad (\text{A1})$$

$$m_u \frac{d^2 x_2}{dt^2} - k_s (x_1 - x_2)^3 - c \left(\frac{dx_1}{dt} - \frac{dx_2}{dt} \right) - k_u (z(t) - x_2) = 0. \quad (\text{A2})$$

The uncertain parameter k_s can be considered to have a uniform probability distribution and can thus be represented in terms of ξ_1 as

$$k(\xi_1) = \mu_k + \sigma_k \xi_1. \quad (A3)$$

Therefore, for k_s and k_u as random variables, that is, $k_s = k_s(\xi_1)$ and $k_u = k_u(\xi_2)$, $x_1(t)$ and $x_2(t)$ will be stochastic processes $x_1(t, \xi)$ and $x_2(t, \xi)$, where ξ is a vector of uniform random variables ξ_1 and ξ_2 , that is, $\xi = [\xi_1, \xi_2]$. For simplicity of presentation, only Equation (A1) will be considered henceforth. It can be rewritten as

$$m_s \frac{d^2 x_1(t, \xi)}{dt^2} + k_s(\xi_1)(x_1(t, \xi) - x_2(t, \xi))^3 + c \left(\frac{dx_1(t, \xi)}{dt} - \frac{dx_2(t, \xi)}{dt} \right) = 0. \quad (A4)$$

Next, $x_1(t, \xi)$ and $x_2(t, \xi)$ are represented in the form of truncated series expansions, consisting of Legendre basis functions of ξ , as

$$x_1(t, \xi) = \sum_{j=0}^{N_0} x_{1j}(t) \Gamma_j(\xi) \quad (A5)$$

$$x_2(t, \xi) = \sum_{j=0}^{N_0} x_{2j}(t) \Gamma_j(\xi). \quad (A6)$$

Propagating this through the ODE, Equation (A4) can be rewritten as

$$\begin{aligned} m_s \sum_{j=0}^{N_0} \frac{d^2 x_1(t)}{dt^2} \Gamma_j(\xi) + k_s(\xi_1) \left(\sum_{j=0}^{N_0} x_1(t) \Gamma_j(\xi) - \sum_{j=0}^{N_0} x_2(t) \Gamma_j(\xi) \right)^3 \\ + c \left(\sum_{j=0}^{N_0} \frac{dx_1(t)}{dt} \Gamma_j(\xi) - \sum_{j=0}^{N_0} \frac{dx_2(t)}{dt} \Gamma_j(\xi) \right) = 0. \end{aligned} \quad (A7)$$

Choosing a set of Q pairs of collocation points, ξ^i with $0 \leq i \leq Q$, equality of Equation (A7) is enforced at these points:

$$\begin{aligned} m_s \sum_{j=0}^{N_0} \frac{d^2 x_1(t)}{dt^2} \Gamma_j(\xi^i) + k_s(\xi_1^i) \left(\sum_{j=0}^{N_0} x_1(t) \Gamma_j(\xi^i) - \sum_{j=0}^{N_0} x_2(t) \Gamma_j(\xi^i) \right)^3 \\ + c \left(\sum_{j=0}^{N_0} \frac{dx_1(t)}{dt} \Gamma_j(\xi^i) - \sum_{j=0}^{N_0} \frac{dx_2(t)}{dt} \Gamma_j(\xi^i) \right) = 0 \quad 0 \leq i \leq Q. \end{aligned} \quad (A8)$$

Also, define

$$Y_{j,i} = \Gamma_j(\xi^i). \quad (A9)$$

Therefore, Equation (A8) reduces to

$$\begin{aligned} m_s \sum_{j=0}^{N_0} \frac{d^2 x_1(t)}{dt^2} Y_{j,i} + k_s(\xi_1^i) \left(\sum_{j=0}^{N_0} x_1(t) Y_{j,i} - \sum_{j=0}^{N_0} x_2(t) Y_{j,i} \right)^3 \\ + c \left(\sum_{j=0}^{N_0} \frac{dx_1(t)}{dt} Y_{j,i} - \sum_{j=0}^{N_0} \frac{dx_2(t)}{dt} Y_{j,i} \right) = 0 \quad 0 \leq i \leq Q. \end{aligned} \quad (A10)$$

This represents the $Q + 1$ equations (for each ξ^i), with every equation having $N_0 + 1$ terms. Now, combine Equations (A5) and (A9) to get

$$x_1(t, \xi^i) = \sum_{j=0}^{N_0} x_{1j}(t) Y_{j,i} \quad (A11)$$

$$x_2(t, \xi^i) = \sum_{j=0}^{N_0} x_{2j}(t) Y_{j,i}, \quad (A12)$$

where $x_1(t, \xi^i)$ reflects the result obtained after solving the differential equation in time, using ξ^i .

As a result, the $Q + 1$ equations for each ξ^i can be expressed as

$$m_s \frac{d^2 x_1(t, \xi^i)}{dt^2} + k_s(\xi_1^i)(x_1(t, \xi^i) - x_2(t, \xi^i))^3 + c \left(\frac{dx_1(t, \xi^i)}{dt} - \frac{dx_2(t, \xi^i)}{dt} \right) = 0, \quad 0 \leq i \leq Q. \quad (\text{A13})$$

A similar procedure is followed to get the corresponding equation for Equation (29). Now, $x_1(t, \xi^i)$ and $x_2(t, \xi^i)$ can be solved with respect to time using the initial conditions provided. Once the time evolution for $x_1(t, \xi^i)$ and $x_2(t, \xi^i)$ at each ξ^i is obtained, the time evolution for the coefficients $x_{1j}(t)$ and $x_{2j}(t)$ can be determined as in Equation (11).

Once the coefficients are known, the mean value at time t is given by

$$\mu_{x_1} = x_{10}(t)\Gamma_0(\xi) \quad (\text{A14})$$

$$\mu_{x_2} = x_{20}(t)\Gamma_0(\xi), \quad (\text{A15})$$

and the variance is obtained as

$$\begin{aligned} \sigma_{x_1}^2(t) &= \sum_{j=0}^{N_0} (x_{1j}(t))^2 \langle \Gamma_j^2 \rangle - (x_{10}(t))^2 = \sum_{j=1}^{N_0} (x_{1j}(t))^2 \langle \Gamma_j^2 \rangle \\ \sigma_{x_2}^2(t) &= \sum_{j=0}^{N_0} (x_{2j}(t))^2 \langle \Gamma_j^2 \rangle - (x_{20}(t))^2 = \sum_{j=1}^{N_0} (x_{2j}(t))^2 \langle \Gamma_j^2 \rangle, \end{aligned} \quad (\text{A16})$$

where $\langle \cdot \rangle$ represents the ensemble average. For orthonormal polynomials, $\langle \Gamma_j^2 \rangle = 1$.

It should be noted that when implementing the gPC method, the number of collocation points is chosen to be equal to the number of coefficients to be determined ($Q = N_0$) and the points are sampled using the ECM.