

RESEARCH

Open Access

An accuracy comparison of polynomial chaos type methods for the propagation of uncertainties

Florian Augustin^{1*}, Albert Gilg², Meinhard Paffrath^{2*}, Peter Rentrop¹, Manuel Villegas^{1*} and Utz Wever^{2*}

*Correspondence:
augustin@ma.tum.de;
meinhard.paffrath@siemens.com;
villegas@ma.tum.de;
utz.wever@siemens.com

¹Department of Mathematics,
Technische Universität München,
Boltzmannstr. 3, Garching, 85748,
Germany

²Corporate Technology, Siemens
AG, Otto-Hahn-Ring 6, Munich,
81730, Germany

Abstract

In (Augustin *et al.* in *European J. Appl. Math.* 19:149-190, 2008) we considered the Polynomial Chaos Expansion for the treatment of uncertainties in industrial applications. For many applications the method has been proven to be a computationally superior alternative to Monte Carlo evaluations. In the current overview we compare the accuracy of Polynomial Chaos type methods for the propagation of uncertainties in nonlinear problems and verify them on two examples relevant for industry. For weakly nonlinear time-dependent models, the generalized Kalman filter equations define an efficient method, yielding good approximations if the quantities of interest are restricted to the first two moments of the solution. Secondly, stochastic collocation is discussed. The method is applied to delay differential equations and random ordinary differential equations. Finally, a generalized PC method is discussed which is based on a subdivision of the random space. This approach is even suitable for highly nonlinear models.

1 Introduction

Uncertainty quantification in industrial applications is an interesting and important field of research [1–4]. Due to uncertainties in system parameters, measurements and in the modeling of physical processes itself, deterministic approaches for the simulation of those processes are not appropriate. Uncertainty quantification became a vast field of research over the past decades and many methods have been proposed to deal with random data [5–10]. Amongst these methods, approaches based on Polynomial Chaos expansions have proven to be very efficient in dealing with uncertainties, see [3, 10, 11].

Those methods revealed to be efficient alternatives to Monte-Carlo (MC) simulations. They were successfully applied to solve stationary problems *e.g.* in the field of stochastic finite elements [10] and to solve differential equations with uncertain parameters and initial values [12]. The suitability of PC for industrial applications was considered in [1]. Additionally, an extensive discussion of the Polynomial Chaos approaches, applied to linear differential equations, was presented. In the current paper we consider numerical methods for the propagation of uncertainties through nonlinear differential equations. The main part is concerned with a comparison of Kalman filter techniques for state estimation and Polynomial Chaos Galerkin techniques. The methods themselves are known from literature, the contribution of this work is the comparison of these methods in selected examples relevant for industry. Our main focus is on the accuracies of the methods. Based on these

results, we try to answer the question what is the numerical effort in order to achieve a low, medium or high accuracy while, in terms of function evaluations, gaining at least one order compared to a crude Monte Carlo method.

- The most widespread method for treating differential equations with uncertainties are techniques of Kalman filtering. While the original Kalman filter was developed to identify noisy observations with linear differential equations [13], here we use it only to integrate the underlying system with uncertainties. A direct generalization to nonlinear equations is given by the extended Kalman filter [14]; it is based on the sensitivities of the equation. The unscented Kalman filter, a nonlinear filter which does not use derivative information, was developed in [15]. It directly approximates the covariance matrix and is, in this context, used just for state estimation.
- Stochastic collocation methods (SCMs) are introduced by [16, 17]. The first practical application of the methods are discussed in [10]. Here, we apply the methods to nonlinear differential equations with delay, which present a more complex behavior than the delay-free counterparts. Stochastic collocation methods have the advantage of using the differential equation as a black box.
- In [1] we presented a theory of the stochastic Galerkin method (SGM), see [3, 10], for ordinary differential equations with uncertain parameters. Nevertheless, this method suffers from the lack of convergence, especially when time evolves [18]. Thus, we have to consider generalized approaches to cope with this problem. Several methods have been proposed in literature, see for example the multi resolution method [9, 19] and the Multi-Element generalized Polynomial Chaos (MEgPC) method [20]. In [21] a rigorous convergence theory for MEgPC and a fully adaptive scheme in time and random space has been developed. In the current paper we compare this MEgPC approach to the other PC type methods mentioned above.

In Section 2 we introduce two examples from industrial applications, which are the base for the comparison of the discussed methods. The first example concerns crack propagation, while the second considers the quorum sensing in biofilms. Next, we present a summary of the Kalman filter equations for state estimation in Section 3. Especially, nonlinear filters in the context of uncertainty propagation of state are discussed. In Section 4 we consider the SCM, which we apply to the random delay differential equations of quorum sensing. We discuss the multi-element approach of the stochastic Galerkin method in Section 5. Finally we compare the three described methods on the base of the two applications, Sections 6.2 and 6.3. In Section 7 we close this article by drawing our conclusions.

2 Test sets of (delay) differential equations

In this section, we introduce the applications considered for the comparison of the methods of the Kalman filter, the SCM and the SGM.

- The first application deals with the growth of cracks, which is important in the context of life cycle analysis. Industrial applications are for example crack growth in turbine blades or in train wheelsets. Here the task is to define appropriate inspection schemes in order to prevent severe damages. The crack growth is modelled by a highly nonlinear ordinary differential equation.
- The second example is a model of the quorum sensing of biofilms, a delay differential equation. The control of biofilms requires a deep understanding of the structured community of microorganisms living on inert surfaces. Industrial applications range

from sewer systems, fresh water systems to corrosion prevention in cooling flow networks of (electric) power plants.

We restrict ourselves to a low number of uncertain variables. Nevertheless, due to the nonlinearities and the delay, differences between the methods with respect to their achieved accuracies will become visible. A more complex model of crack growth, for the evaluation of inspection schemes for train wheelsets, has been computed with the unscented approximation in [22]. The biofilm model, in particular the physical background, has been studied in more detail in [23].

2.1 Crack growth example

For the modeling of a growing crack we use a simplified model of fracture mechanics, see Harris [24], where a center cracked plate in tension is studied. The crack growth relation is based on a modified form of the Forman relation [25]. The crack size a is governed by the ordinary differential equation in the abstract form

$$\frac{\partial a}{\partial N} = f(a, K_{Ic}, \Delta\sigma, C_F), \quad a(0) = a_0, \quad 0 \leq N \leq N_L \tag{1}$$

where N denotes the number of load cycles and N_L the life cycle. The parameters are the stress intensity factor K_{Ic} , the stress range $\Delta\sigma$, the fatigue growth coefficient C_F and the initial crack size a_0 . In more detail the crack growth differential equation reads [24]

$$\frac{\partial a}{\partial N} = C_F \left(\frac{(1-f)\Delta K}{1-R} \right)^{n_f} \frac{(1 - \frac{4}{\pi} \frac{\Delta K_0}{\Delta K} \tan^{-1}(1-R))^p}{(1 - \frac{\Delta K}{(1-R)K_{Ic}})^q}, \tag{2}$$

where

$$\Delta K = \Delta\sigma \sqrt{\pi a} Y \left(\frac{2a}{h} \right), \quad Y(\alpha) = \frac{1 - 0.5\alpha + 0.370\alpha^2 - 0.044\alpha^3}{\sqrt{1-\alpha}}$$

and

$$f = \begin{cases} A_0 + A_1R + A_2R^2 + A_3R^3 & \text{for } R \geq 0, \\ A_0 + A_1R & \text{for } -1 < R < 0, \end{cases}$$

$$A_0 = (0.825 - 0.34\alpha' + 0.05\alpha'^2) [\cos(\pi s/2)]^{1/\alpha'},$$

$$A_1 = (0.415 - 0.071\alpha')s,$$

$$A_2 = 2A_0 + A_1 - 1,$$

$$A_3 = 1 - (A_0 + A_1 + A_3).$$

R is the stress ratio between maximum and minimum stress, h denotes the width of the plate and $n_f, \Delta K_0, p, q, \alpha', s$ are curve-fitting constants.

2.2 The biofilm model with quorum sensing

The model for antibiotics action on a biofilm follows closely the model in [26]. Since the antibiotics are not produced by the hosting organism, it must be supplied externally. Thus,

our equations read

$$\begin{aligned} \dot{A} &= S_A - \mu_A A(t), \\ \dot{B} &= \alpha_{20} \left(1 - \frac{B(t)}{\sigma}\right) B(t) + \alpha_2 B(t) B(t - \tau) - \alpha_3 B(t) A(t). \end{aligned} \quad (3)$$

Here, A denotes the concentration of antibiotics and B the concentration of bacteria. Moreover, the parameters are the logistic reproductive rate of the bacteria α_{20} , the effective carrying capacity of the environment σ , the reproductive rate due to quorum sensing, α_2 , the death rate of bacteria due to antibiotics S_A and death rate for the antibiotics μ_A . All these parameters are taken to be positive. The differential equation for B includes a standard logistic term

$$\alpha_{20} \left(1 - \frac{B(t)}{\sigma}\right) B(t), \quad (4)$$

and a quorum sensing (QS) term

$$\alpha_2 B(t) B(t - \tau). \quad (5)$$

It represents the extra increase in population after the QS mechanism has been initiated. The process is not instantaneous. Thus, assuming it takes a certain amount of time τ to produce the signal molecules, the population of bacteria at time t , $B(t)$, will receive these signal molecules from the population of bacteria at time $t - \tau$, *i.e.* $B(t - \tau)$. The coefficient α_2 accounts for the strength of the mechanism. The larger α_2 , the more efficient the QS mechanism takes place in the biofilm.

3 Solving stochastic differential equations by Kalman filters

In this section we consider stochastic dynamical systems of the form

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{w}(t)), \quad (6)$$

$$\mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t), \mathbf{v}(t)) \quad (7)$$

with $0 \leq t \leq T_L$,

$$\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^n,$$

$$\mathbf{h} : \mathbb{R}^n \times \mathbb{R}^{n_v} \rightarrow \mathbb{R}^m.$$

Here, \mathbf{f} , \mathbf{h} are linear or nonlinear functions, \mathbf{x} is the state vector, \mathbf{u} is the control vector and \mathbf{y} is the observation vector. We assume the process noise \mathbf{w} and the measurement noise \mathbf{v} to be white, stochastically independent and

$$\mathbf{w} \sim \mathcal{N}(0, \mathbb{R}_+^{n_w}), \quad \mathbf{v} \sim \mathcal{N}(0, \mathbb{R}_+^{n_v}). \quad (8)$$

For real processes, white noise is a somewhat simplified assumption. There are extensions of Kalman filtering to colored noise (*e.g.* [27]) but this will not be picked out here. Equations (6) and (7) describe a stochastic system which is continuous in time. In the fol-

lowing, we restrict ourselves to methods for discrete time systems which are obtained by time discretization of (6). Furthermore we concentrate on non-stiff systems. Of course for the numerical integration of stiff systems semi-implicit or implicit methods are required which means a much higher numerical effort. We begin with linear discrete time systems in Section 3.1. Thereafter, we discuss how nonlinear differential equations can be treated by the extended Kalman filter and the unscented Kalman filter in Sections 3.2 and 3.3. After a comparison of both filters in Section 3.4 we give a reinterpretation of the unscented Kalman filter as a sampling method, see Section 3.5.

3.1 Linear discrete time systems

This section adopts the presentation in [14]. A stochastic linear discrete time system can be written as

$$\begin{aligned} \mathbf{x}_k &= A_{k-1}\mathbf{x}_{k-1} + B_{k-1}\mathbf{u}_{k-1} + W_{k-1}\mathbf{w}_{k-1}, \\ \mathbf{y}_k &= H_k\mathbf{x}_k + V_k\mathbf{v}_k \end{aligned}$$

with $\mathbf{w}_{k-1} \sim \mathcal{N}(0, Q_{k-1})$ and $\mathbf{v}_k \sim \mathcal{N}(0, R_k)$ being independent normally distributed random variables. The predictor step of the Kalman filter is given by

$$\hat{\mathbf{x}}_k^- = A_{k-1}\hat{\mathbf{x}}_{k-1} + B_{k-1}\mathbf{u}_{k-1}.$$

For the a priori and a posteriori estimate errors $\mathbf{e}_k^- := \mathbf{x}_k - \hat{\mathbf{x}}_k^-$ and $\mathbf{e}_k := \mathbf{x}_k - \hat{\mathbf{x}}_k$ with corresponding covariances $P_k^- := E(\mathbf{e}_k^- \mathbf{e}_k^{-T})$ and $P_k := E(\mathbf{e}_k \mathbf{e}_k^T)$ holds

$$\mathbf{e}_k^- = A_{k-1}(\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}) + W_{k-1}\mathbf{w}_{k-1} \quad \text{and} \quad P_k^- = A_{k-1}P_{k-1}A_{k-1}^T + W_{k-1}Q_{k-1}.$$

The corrector of the Kalman filter is given by

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + K_k(\mathbf{y}_k - H_k\hat{\mathbf{x}}_k^-),$$

thus for the a posteriori error and its covariances holds

$$\begin{aligned} \mathbf{e}_k &= \mathbf{x}_k - \hat{\mathbf{x}}_k \\ &= \mathbf{x}_k - \hat{\mathbf{x}}_k^- - K_k(\mathbf{y}_k - H_k\hat{\mathbf{x}}_k^-) \\ &= \mathbf{x}_k - \hat{\mathbf{x}}_k^- - K_k(H_k\mathbf{x}_k + V_k\mathbf{v}_k - H_k\hat{\mathbf{x}}_k^-) \\ &= (I - K_kH_k)(\mathbf{x}_k - \hat{\mathbf{x}}_k^-) - K_kV_k\mathbf{v}_k \end{aligned}$$

and $P_k = (I - K_kH_k)P_k^-(I - K_kH_k)^T + K_kV_kR_kV_k^TK_k^T$. Minimization of the trace of P_k leads to

$$K_k = P_k^-H_k^T [H_kP_k^-H_k^T + V_kR_kV_k^T]^{-1}. \tag{9}$$

With this expression for K_k , we derive a simpler formula for P_k ,

$$P_k = (I - K_kH_k)P_k^-. \tag{10}$$

The algorithm is summarized in Table 1.

Table 1 Kalman filter

Model	$\mathbf{x}_k = A_{k-1}\mathbf{x}_{k-1} + B_{k-1}\mathbf{u}_{k-1} + W_{k-1}\mathbf{w}_{k-1}, \mathbf{w}_{k-1} \sim \mathcal{N}(0, Q_{k-1}),$ $\mathbf{y}_k = H_k\mathbf{x}_k + V_k\mathbf{v}_k, \mathbf{v}_k \sim \mathcal{N}(0, R_k)$
Initialization	$\hat{\mathbf{x}}_0, P_0 = E[\hat{\mathbf{x}}_0\hat{\mathbf{x}}_0^T]$
Predictor	$\hat{\mathbf{x}}_k^- = A_{k-1}\hat{\mathbf{x}}_{k-1} + B_{k-1}\mathbf{u}_{k-1},$ $P_k^- = A_{k-1}P_{k-1}A_{k-1}^T + W_{k-1}Q_{k-1}W_{k-1}^T$
Kalman gain	$K_k = P_k^-H_k^T[H_kP_k^-H_k^T + V_kR_kV_k^T]^{-1}$
Corrector	$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + K_k(\mathbf{y}_k - H_k\hat{\mathbf{x}}_k^-), P_k = (I - K_kH_k)P_k^-$

3.2 Extended Kalman Filter (EKF)

The stochastic nonlinear discrete time system can be written as

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}),$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{v}_k).$$

The predictor of the extended Kalman filter is given by

$$\hat{\mathbf{x}}_k^- = \mathbf{f}(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, 0)$$

with the a priori estimation error and its covariance being

$$\mathbf{e}_k^- = \mathbf{x}_k - \hat{\mathbf{x}}_k^- \doteq A_{k-1}(\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}) + W_{k-1}\mathbf{w}_{k-1},$$

$$P_k^- = A_{k-1}P_{k-1}A_{k-1}^T + W_kQ_{k-1}W_k^T.$$

Here we use the abbreviations

$$A_{k-1} := \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, 0)} \quad \text{and} \quad W_{k-1} := \left. \frac{\partial \mathbf{f}}{\partial \mathbf{w}} \right|_{(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, 0)}.$$

The corrector of the extended Kalman filter has the form

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + K_k(\mathbf{y}_k - \mathbf{h}(\hat{\mathbf{x}}_k^-, 0))$$

and we write the a posteriori estimation error as

$$\begin{aligned} \mathbf{e}_k &= \mathbf{x}_k - \hat{\mathbf{x}}_k \\ &= \mathbf{x}_k - \hat{\mathbf{x}}_k^- - K_k(\mathbf{y}_k - \mathbf{h}(\hat{\mathbf{x}}_k^-, 0)) \\ &= \mathbf{x}_k - \hat{\mathbf{x}}_k^- - K_k(\mathbf{h}(\mathbf{x}_k, \mathbf{v}_k) - \mathbf{h}(\hat{\mathbf{x}}_k^-, 0)) \\ &\doteq \mathbf{x}_k - \hat{\mathbf{x}}_k^- - K_k(H_k(\mathbf{x}_k - \hat{\mathbf{x}}_k^-) + V_k\mathbf{v}_k) \\ &= (I - K_kH_k)(\mathbf{x}_k - \hat{\mathbf{x}}_k^-) - K_kV_k\mathbf{v}_k \end{aligned}$$

with

$$H_k := \left. \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right|_{(\hat{\mathbf{x}}_k, 0)} \quad \text{and} \quad V_k := \left. \frac{\partial \mathbf{h}}{\partial \mathbf{v}} \right|_{(\hat{\mathbf{x}}_k, 0)}.$$

Table 2 Extended Kalman filter

Model	$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}), \mathbf{w}_{k-1} \sim \mathcal{N}(0, Q_{k-1}),$ $\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{v}_k), \mathbf{v}_k \sim \mathcal{N}(0, R_k)$
Initialization	$\hat{\mathbf{x}}_0, P_0 = E[\hat{\mathbf{x}}_0 \hat{\mathbf{x}}_0^T]$
Predictor	$\hat{\mathbf{x}}_k^- = \mathbf{f}(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, 0),$ $P_k^- = A_{k-1} P_{k-1} A_{k-1}^T + W_{k-1} Q_{k-1} W_{k-1}^T$
Kalman gain	$K_k = P_k^- H_k^T [H_k P_k^- H_k^T + V_k R_k V_k^T]^{-1}$
Corrector	$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + K_k (\mathbf{y}_k - \mathbf{h}(\hat{\mathbf{x}}_k^-, 0)),$ $P_k = (I - K_k H_k) P_k^-$

For the Kalman gain and the a posteriori error covariance the same formulae (9), (10) as in the linear case apply. The corresponding algorithm of the extended Kalman filter is summarized in Table 2.

3.3 Unscented Kalman Filter (UKF)

For the extended Kalman filter, derivatives of f and h have to be provided which is not possible in many practical applications. So in [15] a nonlinear filter, called unscented filter, without this drawback was introduced. The unscented filter can be used to estimate the mean and covariance of a nonlinear stochastic process $f(\mathbf{w})$ where $\mathbf{w} \in \mathbb{R}^{n_w}$ is a normally distributed random vector with mean $E(\mathbf{w})$ and covariance $P^{ww} \in \mathbb{R}^{n_w \times n_w}$. So-called sigma points $\mathcal{X}^{(i)}$, together with weights W_i^{mean} and W_i^{cov} , are constructed and mapped to $\mathcal{Z}^{(i)} = f(\mathcal{X}^{(i)})$ for $i = 0, \dots, p$. The unscented filter then yields an approximation of the mean μ and covariance P^{zz} of the nonlinear function by

$$E(z) \approx \sum_{i=0}^p W_i^{\text{mean}} \mathcal{Z}^{(i)},$$

$$P^{zz} \approx \sum_{i=0}^p W_i^{\text{cov}} (\mathcal{Z}^{(i)} - \mathbf{y})(\mathcal{Z}^{(i)} - E(z))^T.$$

By Taylor expansion one can show second order accuracy of mean and covariance, [15, 28].

Example A set of $p + 1 = 2n + 1$ weights and sigma points with respect to mean $E(x) \in \mathbb{R}^n$ and covariance $P^{xx} \in \mathbb{R}^{n \times n}$ is given by [29]

$$\mathcal{X}^{(i)} = \mathbf{x} + (\sqrt{n P^{xx}})_i,$$

$$W_i = 1/(2n),$$

$$\mathcal{X}^{(i+n)} = \mathbf{x} - (\sqrt{n P^{xx}})_i,$$

$$W_{i+n} = 1/(2n)$$

for $i = 1, \dots, n$ respectively, where $(\sqrt{n P^{xx}})_i$ denotes the i -th column of the matrix root L of P given by $P^{xx} = LL^T$. Thus, the corresponding weights are $W_i^{\text{mean}} = W_i^{\text{cov}} = W_i$ for $i = 1, \dots, 2n$.

Subsequently, we apply the unscented Kalman filter to the recursive estimation of dynamical processes.

3.3.1 Additive process and measurement noise

For the sake of simplicity we begin with the special case of additive process and measurement noise

$$\begin{aligned} \mathbf{x}_k &= \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) = \tilde{\mathbf{f}}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{w}_{k-1}, \\ \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k, \mathbf{v}_k) = \tilde{\mathbf{h}}(\mathbf{x}_k) + \mathbf{v}_k. \end{aligned}$$

At first, we construct a set of sigma points $\mathcal{X}_{k-1}^{(i)}$ with mean $\hat{\mathbf{x}}_{k-1}$ and covariance P_{k-1} . The dependence of the sigma points on $\hat{\mathbf{x}}_{k-1}$ and P_{k-1} is denoted by $\mathcal{X}_{k-1}^{(i)} = \mathcal{X}_{k-1}^{(i)}(\hat{\mathbf{x}}_{k-1}, P_{k-1})$. The predictor of the unscented Kalman filter and the error covariance are given by

$$\mathcal{X}_k^{- (i)} = \mathbf{f}(\mathcal{X}_{k-1}^{(i)}, \mathbf{u}_{k-1}, \mathbf{0}), \quad \text{for } i = 0, \dots, p, \quad (11)$$

$$\hat{\mathbf{x}}_k^- = \sum_{i=0}^p W_i^{\text{mean}} \mathcal{X}_k^{- (i)},$$

$$P_k^- = \sum_{i=0}^p W_i^{\text{cov}} (\mathcal{X}_k^{- (i)} - \hat{\mathbf{x}}_k^-) (\mathcal{X}_k^{- (i)} - \hat{\mathbf{x}}_k^-)^T + Q_{k-1}. \quad (12)$$

The predicted observation is

$$\mathcal{Y}_k^{(i)} = \mathbf{h}(\mathcal{X}_k^{- (i)}, \mathbf{0}), \quad \text{for } i = 0, \dots, p, \quad (13)$$

$$\hat{\mathbf{y}}_k^- = \sum_{i=0}^p W_i^{\text{mean}} \mathcal{Y}_k^{(i)} \quad (14)$$

with innovation covariance and cross covariance

$$P_k^{yy} = \sum_{i=0}^p W_i^{\text{cov}} (\mathcal{Y}_k^{(i)} - \hat{\mathbf{y}}_k^-) (\mathcal{Y}_k^{(i)} - \hat{\mathbf{y}}_k^-)^T + R_k, \quad (15)$$

$$P_k^{xy} = \sum_{i=0}^p W_i^{\text{cov}} (\mathcal{X}_k^{- (i)} - \hat{\mathbf{x}}_k^-) (\mathcal{Y}_k^{(i)} - \hat{\mathbf{y}}_k^-)^T.$$

The Kalman gain has the form

$$K_k = P_k^{xy} (P_k^{yy})^{-1}. \quad (16)$$

The corrector of the unscented Kalman filter and the a posteriori covariance are given by

$$\begin{aligned} \hat{\mathbf{x}}_k &= \hat{\mathbf{x}}_k^- + K_k (\mathbf{y}_k - \hat{\mathbf{y}}_k^-), \\ P_k &= P_k^- - K_k P_k^{yy} K_k^T. \end{aligned} \quad (17)$$

The relationship between the Kalman gains K_k of the extended and unscented filters (9), (16) is given by

$$P_k^{xy} = P_k^- H_k^T \quad \text{and} \quad P_k^{yy} = H_k P_k^- H_k^T + R_k.$$

Table 3 Unscented Kalman filter

Model	$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) = \tilde{\mathbf{f}}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{w}_{k-1}, \mathbf{w}_{k-1} \sim \mathcal{N}(0, Q_{k-1}),$ $\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{v}_k) = \tilde{\mathbf{h}}(\mathbf{x}_k) + \mathbf{v}_k, \mathbf{v}_k \sim \mathcal{N}(0, R_k)$
Initialization	$\hat{\mathbf{x}}_0, P_0 = E[\hat{\mathbf{x}}_0 \hat{\mathbf{x}}_0^T]$
Weights	$W_i^{\text{mean}}, W_i^{\text{cov}}, i = 0, \dots, p$
Sigma points	$\mathcal{X}_{k-1}^{(i)} = \mathcal{X}_{k-1}^{(i)}(\hat{\mathbf{x}}_{k-1}, P_{k-1}), i = 0, \dots, p$
Predictor	$\mathcal{X}_k^{-(i)} = \mathbf{f}(\mathcal{X}_{k-1}^{(i)}, \mathbf{u}_{k-1}, 0), i = 0, \dots, p,$ $\hat{\mathbf{x}}_k^- = \sum_{i=0}^p W_i^{\text{mean}} \mathcal{X}_k^{-(i)},$ $P_k^- = \sum_{i=0}^p W_i^{\text{cov}} (\mathcal{X}_k^{-(i)} - \hat{\mathbf{x}}_k^-)(\mathcal{X}_k^{-(i)} - \hat{\mathbf{x}}_k^-)^T + Q_{k-1}$
Predicted observation	$\mathcal{Y}_k^{(i)} = \mathbf{h}(\mathcal{X}_k^{-(i)}, 0),$ $\hat{\mathbf{y}}_k^- = \sum_{i=0}^p W_i^{\text{mean}} \mathcal{Y}_k^{(i)}, i = 0, \dots, p,$ $P_k^{yy} = \sum_{i=0}^p W_i^{\text{cov}} (\mathcal{Y}_k^{(i)} - \hat{\mathbf{y}}_k^-)(\mathcal{Y}_k^{(i)} - \hat{\mathbf{y}}_k^-)^T + R_k,$ $P_k^{xy} = \sum_{i=0}^p W_i^{\text{cov}} (\mathcal{X}_k^{-(i)} - \hat{\mathbf{x}}_k^-)(\mathcal{Y}_k^{(i)} - \hat{\mathbf{y}}_k^-)^T$
Kalman gain	$K_k = P_k^{xy} (P_k^{yy})^{-1}$
Corrector	$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + K_k(\mathbf{y}_k - \hat{\mathbf{y}}_k^-), P_k = P_k^- - K_k P_k^{yy} K_k^T$

Also P_k in (17) corresponds to (10) because substitution of (16) in (17) yields

$$\begin{aligned} P_k &= P_k^- - K_k P_k^{yy} K_k^T \\ &= P_k^- - P_k^{xy} (P_k^{yy})^{-1} (P_k^{xy})^T \\ &= P_k^- - K_k H_k P_k^- \\ &= (I - K_k H_k) P_k^- \end{aligned}$$

The algorithm is summarized in Table 3.

3.3.2 Nonlinear noise

For the general case of nonlinear noise augmented sets of sigma points for prediction and corrector steps have to be defined [29]. Let

$$\bar{\mathbf{x}} = (\mathbf{x}^T, \mathbf{w}^T)^T, \tag{18}$$

$$\bar{\bar{\mathbf{x}}} = (\mathbf{x}^T, \mathbf{v}^T)^T, \tag{19}$$

$$\bar{\mathbf{f}}(\bar{\mathbf{x}}, \mathbf{u}) = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}), \tag{20}$$

$$\bar{\mathbf{h}}(\bar{\bar{\mathbf{x}}}, \mathbf{u}) = \mathbf{h}(\mathbf{x}, \mathbf{v}), \tag{21}$$

For the predictor step, define sigma points $\bar{\mathcal{X}}_{k-1}^{(i)}$ with respect to mean

$$\bar{\mathbf{x}}_{k-1} := \begin{pmatrix} \hat{\mathbf{x}}_{k-1} \\ \mathbf{0} \end{pmatrix} \tag{22}$$

and covariance

$$\bar{P}_{k-1} = \begin{pmatrix} P_{k-1} & \mathbf{0} \\ \mathbf{0} & Q_{k-1} \end{pmatrix}. \tag{23}$$

Table 4 General unscented Kalman Filter

Model	$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}), \mathbf{w}_{k-1} \sim \mathcal{N}(0, Q_{k-1}),$ $\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{v}_k), \mathbf{v}_k \sim \mathcal{N}(0, R_k)$
Initialization	$\hat{\mathbf{x}}_0, P_0 = E[\hat{\mathbf{x}}_0 \hat{\mathbf{x}}_0^T]$
Weights	$W_i^{\text{mean}}, W_i^{\text{cov}}, i = 0, \dots, p$
Sigma points	$\mathcal{X}_{k-1}^{(i)} = \bar{\mathcal{X}}_{k-1}^{(i)}(\bar{\mathbf{x}}_{k-1}, \bar{P}_{k-1}), i = 0, \dots, p$
Predictor	$\mathcal{X}_k^{-(i)} = \bar{\mathbf{f}}(\mathcal{X}_{k-1}^{(i)}, \mathbf{u}_{k-1}), i = 0, \dots, p,$ $\hat{\mathbf{x}}_k^- = \sum_{i=0}^p W_i^{\text{mean}} \mathcal{X}_k^{-(i)},$ $P_k^- = \sum_{i=0}^p W_i^{\text{cov}} (\mathcal{X}_k^{-(i)} - \hat{\mathbf{x}}_k^-)(\mathcal{X}_k^{-(i)} - \hat{\mathbf{x}}_k^-)^T$
Sigma points	$\bar{\mathcal{X}}_k^{(i)} = \bar{\mathcal{X}}_k^{(i)}(\bar{\mathbf{x}}_k, \bar{P}_k), i = 0, \dots, p$
Predicted observation	$\mathcal{Y}_k^{(i)} = \bar{\mathbf{h}}(\bar{\mathcal{X}}_k^{(i)}),$ $\hat{\mathbf{y}}_k^- = \sum_{i=0}^p W_i^{\text{mean}} \mathcal{Y}_k^{(i)}, i = 0, \dots, p,$ $P_k^{yy} = \sum_{i=0}^p W_i^{\text{cov}} (\mathcal{Y}_k^{(i)} - \hat{\mathbf{y}}_k^-)(\mathcal{Y}_k^{(i)} - \hat{\mathbf{y}}_k^-)^T,$ $P_k^{xy} = \sum_{i=0}^p W_i^{\text{cov}} (\mathcal{X}_k^{-(i)} - \hat{\mathbf{x}}_k^-)(\mathcal{Y}_k^{(i)} - \hat{\mathbf{y}}_k^-)^T$
Kalman gain	$K_k = P_k^{xy} (P_k^{yy})^{-1}$
Corrector	$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + K_k(\mathbf{y}_k - \hat{\mathbf{y}}_k^-), P_k = P_k^- - K_k P_k^{yy} K_k^T$

Then (11) is substituted by

$$\mathcal{X}_k^{-(i)} = \bar{\mathbf{f}}(\bar{\mathcal{X}}_{k-1}^{(i)}, \mathbf{u}_{k-1}), \quad i = 0, \dots, p \tag{24}$$

and the Q -term in (12) vanishes. Accordingly define sigma points $\bar{\mathcal{X}}_k^{(i)}$ for the corrector step with mean

$$\bar{\bar{\mathbf{x}}}_k := \begin{pmatrix} \hat{\mathbf{x}}_k^- \\ 0 \end{pmatrix} \tag{25}$$

and covariance

$$\bar{\bar{P}}_k = \begin{pmatrix} P_k^- & 0 \\ 0 & R_k \end{pmatrix}. \tag{26}$$

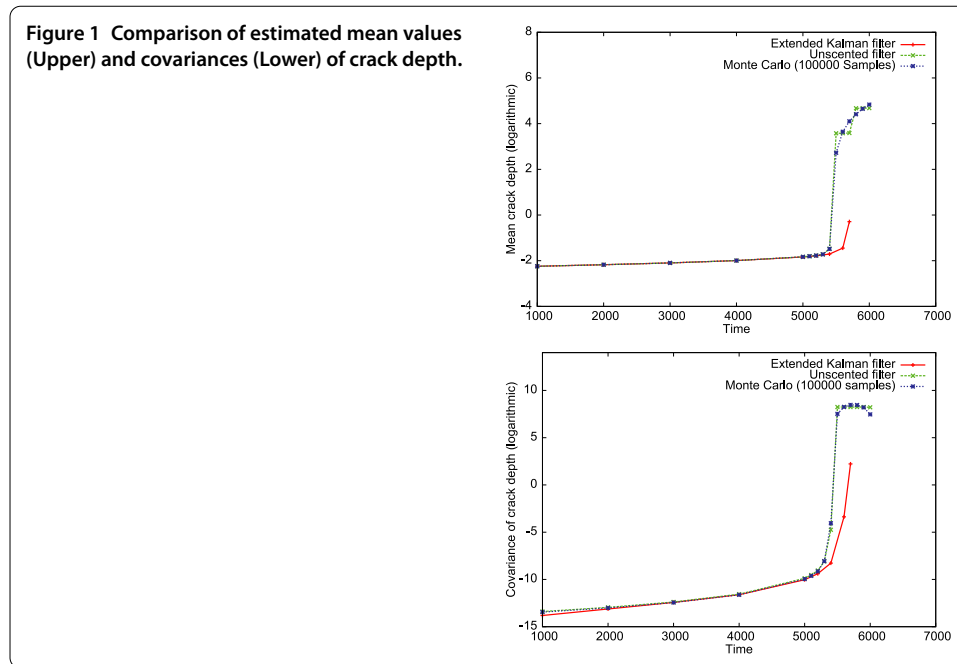
Then (13) is substituted by

$$\mathcal{Y}_k^{(i)} = \bar{\mathbf{h}}(\bar{\mathcal{X}}_k^{(i)}), \quad i = 0, \dots, p \tag{27}$$

and the R -term in (15) vanishes. The algorithm is summarized in Table 4.

3.4 Comparison of EKF and UKF in crack growth example

Figure 1 shows a comparison of the extended Kalman filter, the unscented Kalman filter and a Monte Carlo simulation for the estimation of mean and covariance of the time dependent crack depth. In this test, we only assume the initial crack size a_0 to be uncertain, whereas we set the other parameters in (1) to deterministic values. It can be seen from the figures that both extended and unscented filter are fairly accurate in the linear regime of the crack growth, whereas in the nonlinear regime only the unscented filter shows sufficient accuracy.



3.5 Reinterpretation of UKF

The extended Kalman filter does not treat the nonlinearities of the above example and is therefore no longer considered. The Kalman filter techniques are developed to deal with uncertainties in the state and the measurements. In order to allow a comparison to the methods based on Polynomial Chaos expansions, *cf.* Sections 4 and 5, we omit the treatment of measurements and only consider the state estimation. Thus, the resulting equations that we are considering within the following are ordinary differential equations with random parameters, *cf.* (30) and (33). In this setting, we interpret the construction of sigma points as special stochastic collocation method which is quadratic in each dimension (28). We reconstruct the solution between the sampled points by the polynomial

$$\mathbf{x}(T_L, \theta_1, \dots, \theta_{n_w}) = \sum_{i=1}^{n_w} (q_{0,i} + q_{1,i}\theta_i + q_{2,i}(\theta_i^2 - 1)). \quad (28)$$

In a post-processing step, we will apply a Monte-Carlo simulation based on this polynomial approximation, to obtain approximations of any quantity of interest, *i.e.* the moments or density of the solution.

4 Stochastic Collocation method and random delay differential equations

We will consider the delay differential equation (DDE) with one single, time-independent delay, defined as

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{x}(t - \tau)), \quad t > 0,$$

with $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ and $\tau : \mathbb{R} \rightarrow [0, +\infty]$. The parameter τ is called the delay. Given the initial data set (also called history function) $\mathbf{x}(t) = \phi(t)$, $t \in [-\tau, 0]$ we obtain the initial

value problem (IVP)

$$IVP := \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{x}(t - \tau)), & t > 0, \\ \mathbf{x}(t) = \phi(t), & t \in [-\tau, 0]. \end{cases} \quad (29)$$

When one or more parameters in (29) are affected by uncertainty we will be speaking of a random delay differential equation (RDDE)

$$\dot{\mathbf{x}}(t, \mathbf{w}) = \mathbf{f}(\mathbf{x}(t, \mathbf{w}), \mathbf{x}(t - \tau, \mathbf{w}), \mathbf{w}), \quad t > 0, \quad (30)$$

if $\mathbf{w} \in L_2$. Eventually, the initial history function can also exhibit a random behavior, $\mathbf{x}(t, \mathbf{w}) = \phi(t, \mathbf{w}), t \in [-\tau, 0]$.

In order to account for the uncertainties the stochastic collocation schema of the Polynomial Chaos Expansion is used. The schema is explained below, along with the numerical method applied to solve for the underlying deterministic equations.

4.1 Wiener expansions

We will denote a probability space by (Ω, \mathcal{F}, P) and the space of square integrable random variables on it by $L_2(\Omega, \mathcal{F}, P)$. We will abbreviate it by L_2 in the forthcoming. This space is equipped with the inner product

$$(w, v) := \int_{\Omega} w(\theta)v(\theta) dP(\theta)$$

and the induced norm $\|w\| := \sqrt{(w, w)}$. Let ξ be a standard Gaussian distributed random variable on (Ω, \mathcal{F}, P) , where \mathcal{F} is generated by ξ . N. Wiener [16] and later R. Ghanem and P. Spanos [10] showed, that every random variable $\mathbf{x} \in L_2$ can be expanded into a series

$$\mathbf{x}(\mathbf{w}(\cdot)) = \sum_{i=0}^{\infty} \mathbf{q}_i \Psi_i(\mathbf{w}(\cdot))$$

with basis polynomials $\{\Psi_i\}_{i=0}^{\infty}$. A detailed discussion on this expansions can be found in [1, 4, 30]. In [11] D. Xiu and G. Karniadakis generalized this expansion to polynomials of the Askey scheme. For convergence results of these generalized expansions we refer to [31].

The Wiener expansions can be used to solve operator equations with square integrable random parameters. Assume the operator $T : X \rightarrow \mathbb{R}^n$, mapping from the Banach space X into \mathbb{R}^n , given by

$$T(x; \mathbf{w}) = 0, \quad (31)$$

with $x \in X$ and parameters $\mathbf{w} \in L_2$. If the solution x is in L_2 , it can be expanded by a generalized Wiener expansion. A truncation of this expansion after $\hat{p} + 1$ terms results in an approximation for x in the finite dimensional subspace $\text{linspan}\{\Psi_i : i = 0, \dots, \hat{p}\} \subset L_2$

$$\mathbf{x}(\mathbf{w}) \approx \sum_{i=0}^{\hat{p}} \mathbf{q}_i \Psi_i(\mathbf{w}). \quad (32)$$

Thus the solution \mathbf{x} can be approximated by computing the coefficients $\{\mathbf{q}_i\}_{i=0}^{\hat{p}}$. Different methods can be used to calculate the coefficients. We opt here for the stochastic collocation.

4.2 Stochastic collocation

Stochastic collocation methods make use of the fact that the Wiener expansions are generalized Fourier series. Therefore, the coefficients in (32) are given by

$$\mathbf{q}_i = \frac{(\mathbf{x}, \Psi_i)}{(\Psi_i, \Psi_i)}, \quad i = 0, \dots, \hat{p}.$$

For the computation of the coefficients $\{\mathbf{q}_i\}_{i=0}^{\hat{p}}$, an estimation of the inner products is necessary. This can be accomplished with cubature rules, which approximate the multi-dimensional integral as a discrete sum

$$(\mathbf{x}, \Psi_i) \approx \sum_{j=1}^{N_Q} \mathbf{x}(\mathbf{w}^{(j)}) \Psi_i(\mathbf{w}^{(j)}) W^{(j)},$$

where $\mathbf{w}^{(j)}$, meant as $\mathbf{w}(\theta^{(j)})$, and $W^{(j)}$ are the nodes and weights of the N_Q -node cubature and are given for each type of cubature, see [4] for more details.

The procedure for the stochastic collocation scheme would be then as follows: For each of the given nodes calculate the solution of (31) and sum them up with the corresponding weights. The products (Ψ_i, Ψ_i) are independent from the operator equations and can be conveniently pre-computed and stored for efficient use in different calculations.

This approach offers some advantages. Contrary to MC techniques, the sampling is not random, but obey the cubature rules to minimize the needed number of nodes for a given precision. Thus, the amount of realizations is significantly reduced. Usually, 10 nodes per dimension are accurate enough. Nevertheless, for increasing number of random variables in the parametrization, the numbers of nodes grows exponentially (curse of dimensionality), and the dimension-independent MC becomes eventually more efficient. Sparse grid cubatures have been proposed to mitigate the curse of dimensionality [4].

Another important feature that makes the stochastic collocation schema attractive within the Wiener expansion methodology is the splitting of the uncertainty quantification and the system governing equations. Since the values of the random variables are chosen in advance, we only need to solve the underlying deterministic equations for the given value of the parameters. Hence, no modification of our existing deterministic software is needed in order to account for uncertainties. This is specially valuable in the case of commercial programs, which cannot be modified, or complex software, whose adaptation for uncertainty quantification can be error-prone. A further advantage is the modularity, making implementation of refinements in the Wiener expansion algorithms and the underlying governing equations independent.

For solving the RDDEs (30) we use the stochastic collocation schema combined with the program RADAR for the computation of the DDEs.

4.3 Numerical solution of the DDEs

The software RADAR is used as the deterministic solver for DDEs. It has been developed by Ernst Hairer and Nicola Guglielmi. For more details a manual with examples is available [32].

In a simplified form for presentation purposes, the DDE

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t), \mathbf{x}(\alpha(t, \mathbf{x}(t)))) , \quad \alpha(t, \mathbf{x}(t)) < t, \\ \mathbf{x}(t) &= \mathbf{g}(t), \quad t < t_0, \end{aligned}$$

with $\alpha(t, \mathbf{x}(t))$ the (eventually state-dependent) time-lag function, is solved using an implicit Runge-Kutta method called RADAU IIA [33]. If we denote the stepsize by $h_k = t_{k+1} - t_k$ and the coefficient matrix by $A = (a_{ij})_{i,j=1}^s$, $c_i = \sum_{j=1}^s a_{ij}$, weights $\{b_i\}_{i=1}^s$ the implicit schema reads

$$\mathbf{Y}_i^{(k)} - \mathbf{x}_k = h_k \sum_{j=1}^s a_{ij} \mathbf{f}(\mathbf{Y}_j^{(k)}, \mathbf{Z}_j^{(k)}),$$

where \mathbf{x}_k is the approximation of $\mathbf{x}(t_k)$ and $\mathbf{Y}_s = \mathbf{x}_{k+1}$, if we use the weights $b_i = a_{s,i}$, $i = 1, \dots, s$. $\mathbf{Z}_j^{(k)}$ is defined by

$$\mathbf{Z}_j^{(k)} = \begin{cases} \mathbf{g}(\alpha_j^{(k)}), & \alpha_j^{(k)} < t_0, \\ \mathbf{u}_m(\alpha_j^{(k)}), & \alpha_j^{(k)} \in [t_m, t_{m+1}], \end{cases}$$

where $\alpha_j^{(k)} = \alpha(t_k + c_j h_k, \mathbf{Y}_j^{(k)})$ and \mathbf{u}_m is a polynomial approximation of the solution $\mathbf{x}(t)$ on the interval $[t_m, t_{m+1}]$. Here, the Lagrange interpolation formula is used to build the approximation.

RADAR allows steps larger and smaller than the delay and performs stepsize control by estimating the local error at grid points and the error in the continuous numerical approximation to the solution.

5 Stochastic Galerkin method and random ordinary differential equations

Within Section 4 we discussed the treatment of RDDEs by the collocation method based on generalized Wiener expansions. In this section we will present the stochastic Galerkin method (SGM) for the approximation of the solution of differential equations with uncertain parameters, cf. [1]. We again consider the differential equation (30), $\dot{\mathbf{x}}(t, \mathbf{w}) = \mathbf{f}(\mathbf{x}(t, \mathbf{w}), \mathbf{x}(t - \tau, \mathbf{w}), \mathbf{w})$, and omit the delay within the forthcoming discussion to keep the notation simple:

$$\dot{\mathbf{x}}(t, \mathbf{w}) = \mathbf{f}(\mathbf{x}(t, \mathbf{w}), \mathbf{w}) \tag{33}$$

with initial condition $\mathbf{x}(0, \mathbf{w}) \in L_2$. The extension of the SGM including the treatment of delays is straightforward using interpolation for the approximation of the solution at past times. In Section 6 we will apply the SGM to the test example of the crack growth (an RODE) and the biofilm model (an RDDE) as stated in the introductory Section 1. Subsequently, we intend to present the main idea of the SGM and restate the main results of its convergence analysis from [21, 34].

For the numerical approximation of the solution of (33) we search for an approximation of $\mathbf{x}(t, \mathbf{w})$ within the finite dimensional subspace

$$\mathcal{S} = \text{linspan}\{\Psi_i : i = 0, \dots, \hat{p}\} \subset L_2,$$

the approximation space. Without loss of generality we assume that the orthogonal polynomials $\{\Psi_i\}_{i=0}^\infty$ are normalized, *i.e.* $(\Psi_i, \Psi_i) = 1$.

Besides RODE (33) we introduce the perturbed RODE

$$\frac{\partial}{\partial t} \tilde{\mathbf{x}}(t, \mathbf{w}) = \mathbf{f}(\pi_{\hat{p}} \circ \tilde{\mathbf{x}}(t, \mathbf{w}), \mathbf{w}) =: \tilde{\mathbf{f}}(\tilde{\mathbf{x}}(t, \mathbf{w}), \mathbf{w}), \tag{34}$$

where $\pi_{\hat{p}} : L_2 \rightarrow \mathcal{S}$ denotes the orthogonal projection from L_2 onto \mathcal{S} . We define the restricted stochastic weak form of (34) by

$$\left(\frac{\partial}{\partial t} \tilde{\mathbf{x}}(t, \mathbf{w}), \Psi(\mathbf{w}) \right) = (\tilde{\mathbf{f}}(\tilde{\mathbf{x}}(t, \mathbf{w}), \mathbf{w}), \Psi(\mathbf{w})) \tag{35}$$

for all $\Psi \in \mathcal{S}$. Equation (35) can be written as a system of ordinary differential equations (ODEs) for the coefficients $\{\tilde{\mathbf{q}}_i\}_{i \leq \hat{p}}$:

$$\frac{d}{dt} \tilde{\mathbf{q}}_i(t) = (\tilde{\mathbf{f}}, \Psi_i) \quad \text{for } i \leq \hat{p}. \tag{36}$$

This system of ODEs is deterministic and can be solved by standard integrators. The projected system (36) can be solved by means of a Runge-Kutta method (RKM) of order $p \geq 1$ and stepsize $h > 0$. We obtain the approximation $\hat{\mathbf{x}}$ of the solution of (34) in terms of the numerically computed coefficients $\{\hat{\mathbf{q}}_i\}_{i \leq \hat{p}}$ using the SGM and a RKM,

$$\tilde{\mathbf{x}}(t, \mathbf{w}) \approx \hat{\mathbf{x}}(t, \mathbf{w}) := \sum_{i \leq \hat{p}} \hat{\mathbf{q}}_i(t) \Psi_i(\mathbf{w}).$$

The following scheme depicts the stochastic Galerkin Runge-Kutta method (SGRKM):

$$\begin{aligned} \frac{\partial}{\partial t} \tilde{\mathbf{x}} = \tilde{\mathbf{f}}(\tilde{\mathbf{x}}, \mathbf{w}) &\xrightarrow{\text{SGM}} \frac{d}{dt} \tilde{\mathbf{q}} = \mathbf{F}(\tilde{\mathbf{q}}) \\ &\xrightarrow{\text{RKM}} \hat{\mathbf{x}} = \sum_{i \leq \hat{p}} \hat{\mathbf{q}}_i \Psi_i(\mathbf{w}). \end{aligned} \tag{37}$$

In [34] it is shown that the local discretization error of the SGRKM does not vanish for decreasing stepsize $h \rightarrow 0$. Thus, this global approach of the stochastic Galerkin method fails to converge to the exact solution \mathbf{x} . To overcome this problem a generalized approach has to be applied. X. Wan and G.E. Karniadakis proposed the Multi-Element generalized Polynomial Chaos (MEgPC) method to attain convergence of the SGM. We recall that this method is based on a partitioning of the parameter space Ω into subsets, so called random elements. The SGM is then applied locally on the partition of the parameter space, *i.e.* the problem is conditionally restricted to each random element. We refer to the literature for a detailed description [18].

Using the MEgPC approach it can be shown that the local discretization error of the SGRKM obeys

$$\|\mathbf{x}(t+h, \mathbf{w}) - \hat{\mathbf{x}}(t+h, \mathbf{w})\|_2 \leq \mathcal{O}(h\Delta^{\hat{p}+1}) + \mathcal{O}(h^{p+1}), \tag{38}$$

where we denote the maximal diameter of the stochastic elements by Δ , see [34].

5.1 The adaptive algorithm for the solution of RODEs

We use (38) to control the stepsize h in time and the maximal size Δ of the stochastic elements $\{\Omega_i\}_{i=1}^v$. In a first step, we compute the conditionally restricted solution $\tilde{\mathbf{x}}(t+h, \mathbf{w})$ on every element Ω_i and check the local truncation error of the projected restricted system (36). See [35] for details on adaptive stepsize control of RKMs. This procedure results in the term $\mathcal{O}(h^{p+1})$ being negligible up to a prescribed tolerance. In a second step, we check the error ERR, which is introduced by the truncation of $\tilde{\mathbf{x}}(t+h, \mathbf{w})$ to \mathcal{S} . Therefore, we split Ω_i into two equal subsets and compute the solution of (36) conditionally restricted to those subsets. We take the L_2 -norm of the difference of the refined solution and the solution $\tilde{\mathbf{x}}(t+h, \mathbf{w})$ on Ω_i as substitute for the actual truncation error ERR_i . We apply this procedure in every random element at every timestep, so we omit the corresponding index i of the subsets in the following. According to (38) it holds

$$\text{ERR} = ch\Delta^{\hat{p}+1}, \tag{39}$$

where we neglect the error term $\mathcal{O}(h^{p+1})$ of the RKM. Furthermore, we require the optimal size Δ_{opt} of the respective element to satisfy

$$\varepsilon = c\Delta_{\text{opt}}^{\hat{p}+1} \tag{40}$$

for prescribed tolerance $\varepsilon > 0$. Division of (40) and (39) results in

$$\Delta_{\text{opt}} = \sqrt[\hat{p}+1]{\frac{\varepsilon h}{\text{ERR}}} \Delta =: \mathcal{I}(\text{ERR})\Delta.$$

We use the estimator $\mathcal{I}(\text{ERR})$ to decide about the refinement of the elements $\{\Omega_i\}_{i=1}^v$. If $\mathcal{I}(\text{ERR}) < \rho_1$, then the element will be refined. If $\mathcal{I}(\text{ERR})$ exceeds a coarsening threshold $\rho_2 > \rho_1$, then the element will be coarsened. For a more detailed description of this procedure see [34].

6 Computational aspects and numerical results

In this section, we apply the three methods, the UKF approach, the stochastic collocation and the SGRKM, to the benchmark problems stated in the introduction. From the solution we compute the densities and the first two moments and compare the results. For the numerical time integration in the SGM we use an embedded explicit RKM of order 4(3). After a discussion of the efficiencies of the methods, we first consider the crack growth example (1) with the initial crack size being uncertain. Thereafter, we study the setting of two uncertain parameters in the model of crack growth. We finish this section by the comparison of the three methods applied to the biofilm model (3).

Table 5 Number of function evaluations for UKF, SCM and MEgPC approach based on a Gaussian cubature rule of order s

UKF	$2n + 1$
SCM	$(s + 1)^n$
MEgPC	$(s + 1)^n$

Numbers given per timestep of the time integration method.

6.1 Computational aspects

For UKF and SCM, the efficiency in terms of function evaluations can directly be given as function of the stochastic dimension n (Table 5). Here the number of sigma points for UKF has been chosen as in the example Section 3.3. For SCM a standard Gaussian cubature rule of order s has been used. SCM and MEgPC show an exponential growth in the number of function evaluations (suitable for 2D problems). Sparse cubature rules which are necessary to cope with high dimensional random spaces are considered in [22]. The MEgPC approach results in ν projected systems of ordinary differential equations (36) of dimension

$$\frac{(n + \hat{p})!}{n! \hat{p}!}$$

if a polynomial basis of total order \hat{p} is applied. The number of needed function evaluations depends on the cubature rule applied to compute the projection integrals, for example using a Gaussian cubature tensor product rule we need $(s + 1)^n$ function evaluations per timestep of the time integration method. However, the MEgPC approach is most efficient, if the projection integrals can be computed analytically in a preprocessing step before the time-integration starts. If this so called intrusive approach is not feasible, we have to resort to cubature rules to compute the projection. In the case of multiple random dimensions, $n \geq 2$, sparse integration methods have to be applied in order to reduce the curse of dimensionality.

6.2 Comparison of UKF, Stochastic Collocation and SGM in crack growth example

In this section, we compare the presented methods and Monte Carlo Sampling (10^6 samples) in the crack example (1).

6.2.1 One dimensional random space

At first, we assume only the initial crack size a_0 to be a random parameter following a uniform distribution with expected value 10^{-1} and variance 10^{-6} . The other parameters are set to $K_{Ic} = 10^{-12}$, $\Delta\sigma = 16$ and $C_F = 0$. We remark that although the right-hand side function f in (1) is not polynomial in the initial value a_0 , the convergence results from Section 5 apply, see [21, 34] for a detailed discussion.

To compute the numerical approximation $\hat{a}(t, a_0)$ of the crack size $a(t, a_0)$ by means of the MEgPC method, we use the parameters stated in Table 6. The adaptive MEgPC approach uses a truncated generalized Wiener expansion of degree $\hat{p} = 3$ for the approximation of the solution. The set of outcomes Ω is refined where the local error exceeds the prescribed tolerance $\varepsilon = 10^{-8}$. The approximated response surfaces at $t \in \{5,000, 5,300, 5,400, 5,420\}$ are shown in Figure 2. The respective elements of the approximation are indicated by dots on the axis of the outcomes Ω .

Table 6 Setting of parameters of the adaptive MEgPC approach for the solution of RODE (1)

Degree of approximation	\hat{p}	3
Order of approximation	p	3
Tolerance (time)	TOL	10^{-7}
Tolerance (elements)	ε	10^{-8}
Refinement threshold	ρ_1	1
Coarsening threshold	ρ_2	4

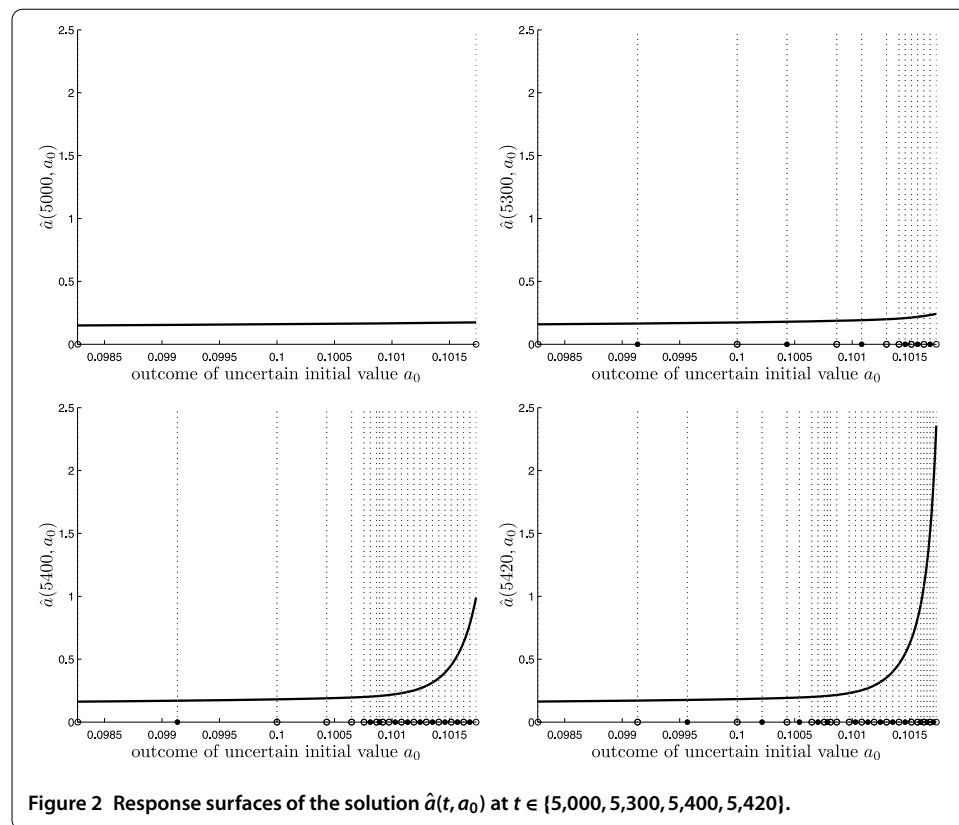


Figure 2 Response surfaces of the solution $\hat{a}(t, a_0)$ at $t \in \{5,000, 5,300, 5,400, 5,420\}$.

Although the initial crack size varies only by a small amount, the impact on the outcome is not negligible. A large variation in the crack size occurs at the final time $t = 5,420$. The dependence of the solution $a(t, a_0)$ on the uncertain initial crack size a_0 is highly non-linear. This is recognized by the MEgPC algorithm, which refines the parameter space where the response surface can not be represented by a polynomial of degree $\hat{p} = 3$ up to the prescribed tolerance.

In Figure 3 we compare the densities of the approximations $\hat{a}(t, a_0)$, $t \in \{5,000, 5,420\}$, computed by the stochastic collocation of order 5 (SCM5), MEgPC and the UKF approach respectively. We also plot the density computed by Monte Carlo simulation as a reference for the approximatively exact density. On the one hand, we observe that the UKF is not able to compute a good approximation of the density of the crack size a . This can be explained by the interpretation of the UKF as a stochastic collocation method of polynomial order 2, see Section 3.5. On the other hand, the MEgPC method, as we expect from our considerations in Section 5, yields a good approximation of the density of the crack size a in dependence on an uncertain initial crack size.

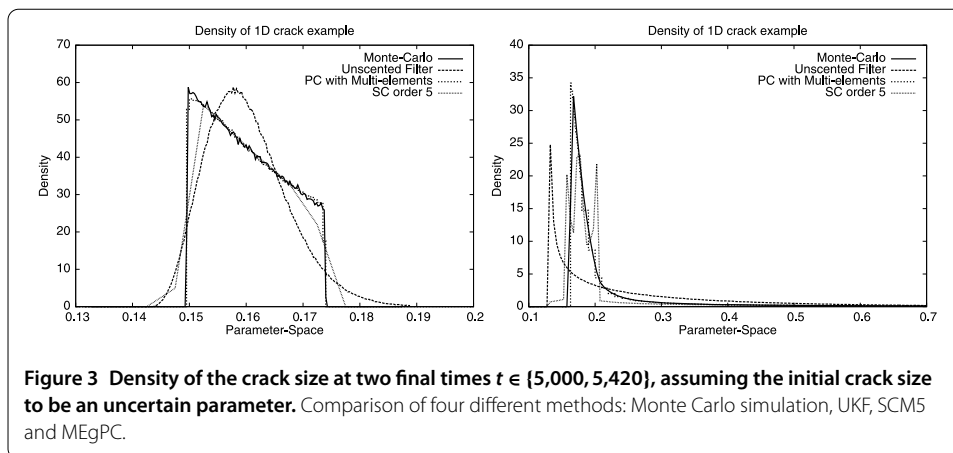


Figure 3 Density of the crack size at two final times $t \in \{5,000, 5,420\}$, assuming the initial crack size to be an uncertain parameter. Comparison of four different methods: Monte Carlo simulation, UKF, SCM5 and MEgPC.

Table 7 The expected values and variances of $a(t, a_0)$ at times $t \in \{5,000, 5,420\}$ computed by SCM5, UKF, MEgPC and Monte Carlo simulation

Time	Expected value				Variance			
	SCM5	MEgPC	UKF	MCS	SCM5	MEgPC	UKF	MCS
5,000	0.1.601	0.1.601	0.1.602	0.1601	$4.727 \cdot 10^{-5}$	$4.729 \cdot 10^{-5}$	$5.113 \cdot 10^{-5}$	$4.728 \cdot 10^{-5}$
5,420	0.2.681	0.2.726	0.2.671	0.2749	$5.655 \cdot 10^{-2}$	$7.791 \cdot 10^{-2}$	$2.449 \cdot 10^{-2}$	$8.627 \cdot 10^{-2}$

Table 8 Setting of parameters of the adaptive MEgPC approach for the solution of RODE (1)

Degree of approximation	\hat{p}	(3, 3)
Order of approximation	p	3
Tolerance (time)	TOL	10^{-8}
Tolerance (elements)	ε	10^{-8}
Refinement threshold	ρ_1	1
Coarsening threshold	ρ_2	4

In addition to the comparison of the densities we also compare the first two moments, see Table 7. We remark that although the UKF approach does not capture the overall density, it is able to approximate the first two moments in good accuracy. Moreover, the first two moments, which we computed by means of the MEgPC method and the Monte Carlo simulation match very good.

6.2.2 Two dimensional random space

In this section we consider the crack example (1) with two uncertain parameters. In addition to Section 6.2.1 we assume the parameter K_{Ic} to be uniformly distributed with expected value 0.25 and variance 0.02.

To compute the numerical approximation $\hat{a}(t, a_0, K_{Ic})$ of the crack size $a(t, a_0, K_{Ic})$ by means of the MEgPC method, we use the parameters as stated in Table 8. The notation $\hat{p} = (3, 3)$ means that we use a polynomial approximation of degree 3 in both random variables a_0 and K_{Ic} .

In Figure 4 we plot the densities computed by SCM5, UKF and MEgPC at times $t \in \{3,000, 3,500, 3,550\}$. Additionally, we plot the density of the reference solution computed by Monte Carlo simulation. We see that the difference between the density computed by the UKF and the reference density is negligible at $t = 3,000$ and becomes larger as time evolves. However, the overall structure of the density can be captured by the UKF ap-

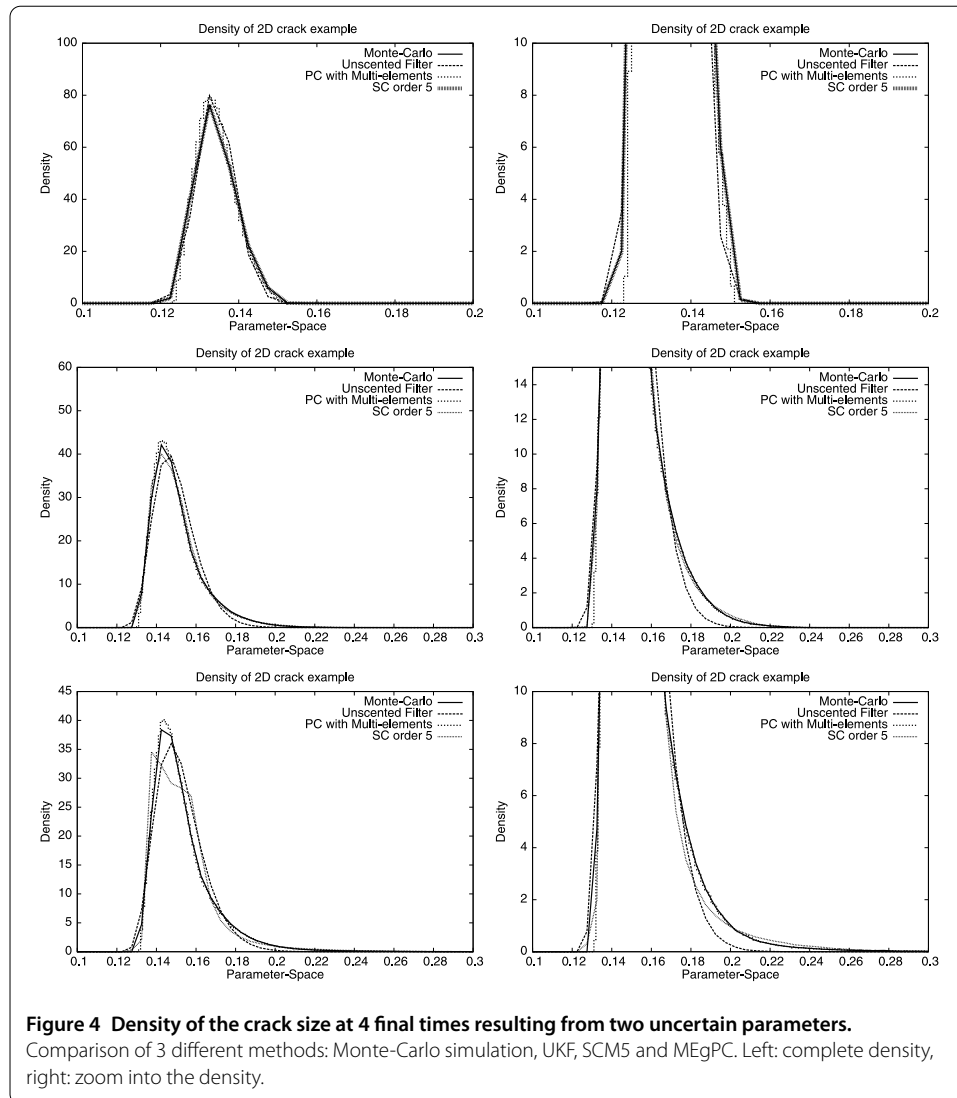


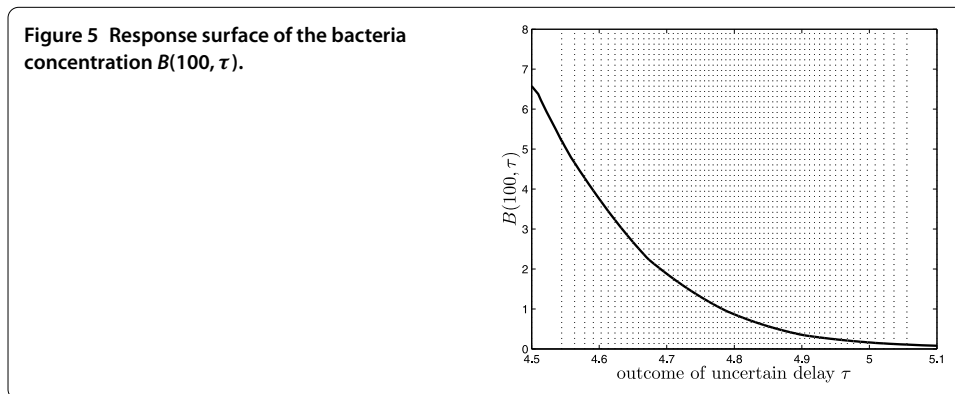
Table 9 The expected values and variances of $a(t, a_0, K_{lc})$ at times $t \in \{3,000, 3,500, 3,550\}$ computed by SCM5, UKF, MEgPC and Monte Carlo simulation based on 10^6 samples

Time	Expected value				Variance			
	SCM5	MEgPC	UKF	MCS	SCM5	MEgPC	UKF	MCS
3,000	0.1343	0.1343	0.1342	0.1343	$2.557 \cdot 10^{-5}$	$2.562 \cdot 10^{-5}$	$2.142 \cdot 10^{-5}$	$2.566 \cdot 10^{-5}$
3,500	0.1504	0.1504	0.1497	0.1504	$1.663 \cdot 10^{-4}$	$1.674 \cdot 10^{-4}$	$1.098 \cdot 10^{-4}$	$1.667 \cdot 10^{-4}$
3,550	0.1534	0.1534	0.1521	0.1534	$2.661 \cdot 10^{-4}$	$2.718 \cdot 10^{-4}$	$1.404 \cdot 10^{-4}$	$2.718 \cdot 10^{-4}$

proach. This is also reflected by the good approximation of the first two moments, see Table 9. Nevertheless, due to the increasing non-linear dependence on the crack size to the uncertain parameters the accuracy of the UKF approach becomes worse with increasing time. Again, this can be explained by the interpretation of the UKF as a polynomial approximation of degree 2 of the solution, see Section 3.5. Similar to the case of only one random parameter the densities computed by the MEgPC method and Monte Carlo simulation show perfect agreement. Moreover, the first two moments of the solutions agree very well, see Table 9.

Table 10 Setting of parameters of the adaptive MEgPC approach for the biofilm model

Degree of approximation	\hat{p}	5
Order of approximation	p	3
Tolerance (time)	TOL	$5 \cdot 10^{-6}$
Tolerance (elements)	ε	\times
Refinement threshold	ρ_1	\times
Coarsening threshold	ρ_2	\times



6.3 Comparison of UKF, Stochastic Collocation and SGM in biofilm example

The introductory model for the biofilm (3) with delay Quorum Sensing has been studied in [36] in great detail. Here we compare the numerical approximations of the solution by means of its densities and first two moments. Therefore, we compute them by the UKF approach, the stochastic collocation and the MEgPC method at the time $t = 100$. The uncertain parameter is assumed to be the delay τ , a beta(2,2) distributed random variable with expected value 4.8. Moreover, we use the choice of parameters $\alpha_{20} = 0.25$, $\sigma = \alpha_2 = \alpha_3 = 0.6$, $S_A = 0.4$ and $\mu_A = 0.5$ as well as the initial populations $B(0) = 1$ and $A(0) = 0$.

For the MEgPC method we use the choice of parameters as stated in Table 10.

As indicated by the crosses we do not use the adaptive mesh refinement strategy in this example. The reason is that we have to evaluate the solution at past timesteps in order to get an evaluation of the right-hand side function. In order to evaluate the solution at a time in the past the interpolation of the solution between two timesteps is needed. Therefore, we choose to have the solution defined on the same mesh of elements at every timestep. In order to avoid different meshes at different timesteps we switch off the mesh adaptivity and prescribe a sufficiently fine mesh, see the dotted lines in Figure 5, from the beginning of the computations. The response surface of the solution at time $t = 100$, computed by the MEgPC approach, is shown in Figure 5.

From the solution of the 3 methods we compute the density at time $t = 100$, see Figure 6 (left). The right plot shows a magnification of the density to the tail of the density, *i.e.* to the interval [3, 7]. On the one hand, we see that the density computed by the stochastic collocation of order 5 simulation and the MEgPC method match very good (therefore we here omit the Monte-Carlo solution). On the other hand, we again observe a difference to the density computed by the UKF approach. The UKF density shows a much smaller weight in [0, 0.5] than the two other methods. This can be explained by the polynomial approximation of degree 2, where more mass of the response surface lies within this range

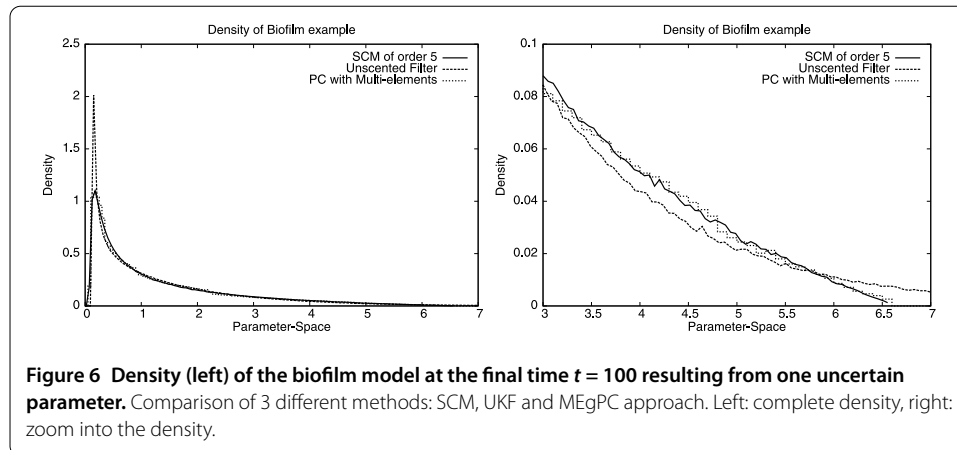


Figure 6 Density (left) of the biofilm model at the final time $t = 100$ resulting from one uncertain parameter. Comparison of 3 different methods: SCM, UKF and MEgPC approach. Left: complete density, right: zoom into the density.

Table 11 The expected values and variances of the biofilm model at time $t = 100$ computed by the MEgPC approach, the UKF and the stochastic collocation method

Method	Expected value	Variance
MEgPC	1.349	1.707
UKF	1.388	1.801
SCM order 5	1.352	1.706
SCM order 9	1.352	1.706

[0, 0.5] of outcomes. Nevertheless, the overall shape of the density can be captured well by the UKF. In Table 11 we state the first two moments, respectively computed by the three methods. We see good agreement of the first two moments.

7 Conclusions

In the present article, we discussed the numerical treatment of problems with uncertain parameters. We gave a detailed introduction to the unscented Kalman filter and its application to random ordinary differential and random delay differential equations, see Section 3. We dropped the measurements in the equations and model the uncertainty by random variables. In this setting, the unscented Kalman filter can be interpreted as a stochastic collocation method of order 2, see Section 3.5. Sampling of the resulting surrogate polynomial model yields an approximation of the moments of the solution. Due to the fact that the sampling is restricted to the directions given by the columns of the root of the covariance matrix P^{xx} , only $2n + 1$ realizations of the underlying random differential equation (RDE), resp. an RDDE or an RODE, are needed in the UKF approach. Here, n denotes the number of uncertain parameters. Thus, it is a cheap, in terms of computational costs, method to approximate the random behavior of the solution.

We compare this simplified UKF approach with the SCM and the MEgPC method. In the SCM more realizations, as compared to the UKF, of the solution of the RDE have to be computed. The number of realizations depends on the order of the approximation and the underlying cubature method. For most cubature rules, even in sparse grid approximations, the number of function evaluations is considerably greater than $2n + 1$ and therefore the SCM is computationally more expensive. The SGM, as opposed to the SCM and the UKF, is not based on sampling, but on a spectral expansion of the solution of the RDE into orthogonal polynomials. An approximate solution is found by a Galerkin approach which results in the stochastic weak form (34). If we are able to compute it in a pre-processing

step, only a system of deterministic differential equations for the coefficients of the truncated generalized Wiener expansion has to be solved. The dimension of the projected RDE is small, because we only need a low order approximation within the respective stochastic elements. The overall computational costs scale linearly with the number of stochastic elements. In the present article we did not take advantage of the pre-processing of the projection in the stochastic weak form. On the one hand, the projection in the examples from Section 6 can be efficiently performed by an adaptive Simpson cubature rule. On the other hand, pre-processing the stochastic weak form would result in a projection error, which can not be easily estimated a priori.

In Section 6 we applied the three methods to the benchmark problems of the crack growth model (1) and the biofilm model (3). In the case of the crack growth, which is an RODE, we discussed the prescription of up to two random parameters. It revealed that the UKF approach results in a non-sufficient representation of the overall density, but gives a good approximation of the first two moments. Opposed to that, the MEgPC method computes a good approximation of the density and of the first two moments. This is in agreement of the convergence results in Section 5. In the second benchmark problem, the example of the biofilm (3), we again compared the densities of the solutions computed by the three methods. The SCM, as well as the SGM method showed perfect agreement of the results - in the density and in the first two moments. Again, opposed to that the UKF approach yielded only a rough approximation of the overall density. Nevertheless, it is able to compute a good approximation of the first two moments.

From the numerical results in Section 6 we see that the UKF approach is a cheap alternative to get a rough impression of the random behavior of the solution of the RDE. If detailed information about the solution process is needed, more sophisticated methods like SCM and multi-element SGM have to be applied. This is for example in failure detection the case, where the tail of the probability distribution is of great interest.

Received: 22 June 2012 Accepted: 15 March 2013 Published: 21 March 2013

References

1. Augustin F, Gilg A, Paffrath M, Rentrop P, Wever U: **Polynomial chaos for the approximation of uncertainties: chances and limits.** *Eur. J. Appl. Math.* 2008 **19**:149-190.
2. Xiu D: *Numerical Methods for Stochastic Computations: A Spectral Method Approach.* Princeton: Princeton University Press; 2010.
3. Karniadakis GE, Su CH, Xiu D, Lucor D, Schwab C, Todor RA: **Generalized polynomial chaos solution for differential equations with random inputs.** Report Nr. 2005-01, ETH Zürich. Seminar für Angewandte Mathematik; 2005.
4. Le Maître OP, Knio OM: *Spectral Methods for Uncertainty Quantification.* Scientific Computation. New York: Springer; 2010.
5. Fox BL: *Strategies for Quasi-Monte Carlo.* Berlin: Springer; 1999.
6. Babuska I, Tempone R, Zouraris G: **Galerkin finite element approximation of stochastic elliptic partial differential equations.** *SIAM J. Numer. Anal.* 2004, **42**(2):800-825.
7. Pulch R: **Polynomial chaos for multirate partial differential algebraic equations with random parameters.** *Appl. Numer. Math.* 2009, **59**(10):2610-2624.
8. Xiu D: **Fast numerical methods for stochastic computations.** *Commun. Comput. Phys.* 2008, **5**:242-272.
9. Maître OPL, Knio OM, Najm HN, Ghanem RG: **Uncertainty propagation using Wiener-Haar expansions.** *J. Comput. Phys.* 2004, **197**:28-57.
10. Ghanem R, Spanos P: *Stochastic Finite Elements: A Spectral Approach.* New York: Springer; 1991.
11. Xiu D, Karniadakis G: **The Wiener-Askey polynomial chaos for stochastic differential equations.** *SIAM J. Sci. Comput.* 2002, **24**(2):619-644.
12. Schevenels M, Lombaerts G, Degrande G: **Application of the stochastic finite element method for Gaussian and non-Gaussian systems.** In *ISMA2004 International Conference on Noise and Vibration Engineering.* Leuven, Belgium; 2004.
13. Kalman R: **A new approach to linear filtering and prediction problems.** *Trans. ASME, J. Basic Eng.* 1960, **82**:35-45.
14. Welch G, Bishop G: **An introduction to the Kalman filter.** Tr 95-041, UNC-Chapel Hill; 2006.
15. Julier SJ, Uhlmann JK, Durrant-Whyte HF: **A new method for the nonlinear transformation of means and covariances in filters and estimators.** *IEEE Trans. Autom. Control* 2000, **45**(3):477-482.

16. Wiener N: **The homogeneous chaos.** *Am. J. Math.* 1938, **60**:897-936.
17. Cameron R, Martin W: **The orthogonal development of nonlinear functionals in series of Fourier-Hermite functionals.** *Ann. Math.* 1947, **48**(2):385-392.
18. Wan X, Karniadakis G: **An adaptive multi-element generalized polynomial chaos method for stochastic differential equations.** *J. Comput. Phys.* 2005, **209**:617-642.
19. Maître OPL, Njam HN, Ghanem RG, Knio OM: **Multi-resolution analysis of Wiener-type uncertainty propagation schemes.** *J. Comput. Phys.* 2004, **197**(2):502-531.
20. Wan X, Karniadakis G: **Multi-element generalized polynomial chaos for arbitrary probability measures.** *SIAM J. Sci. Comput.* 2006, **26**(3):901-928.
21. Augustin F, Rentrop P: **Stochastic Galerkin techniques for random ordinary differential equations.** *Numer. Math.* 2012, **122**:399-419.
22. Paffrath M, Wever U: **Stochastic integration methods and their application to reliability analysis.** In *ASME Turbo Expo*. Copenhagen, Denmark; 2012.
23. Villegas M: **Random delay differential equations: application to biofilm modeling.** *Phd thesis.* Techn. Universität München (Fachbereich Mathematik, Prof. Rentrop); 2011.
24. Harris DO: **Probabilistic crack growth and modeling.** In *Reliability-Based Mechanical Design*. Edited by Cruse T. New York: Marcel Dekker; 1997.
25. Forman RG: **Fatigue crack growth computer program, NASA/FLAGRO version 2.0.** Johnson Space Center Report JSC-22267A, NASA; 1993.
26. Juan Z, Zhongua Z, Yaohong S: **Hopf bifurcation of a bacteria-immunity system with delayed quorum sensing.** *Appl. Math. Comput.* 2010, **215**:3936-3949.
27. Kuhlmann H: **Kalman-filtering with coloured measurement noise for deformation analysis.** In *Proceedings, 11th FIG Symposium on Deformation Measurements*. Santorini, Greece; 2003.
28. Weber B: **Numerik nichtlinearer Systeme mit Unsicherheiten.** *Master thesis.* Technische Universität München, Prof. Rentrop; 2008.
29. Julier S, Uhlmann J: **Unscented filtering and nonlinear estimation.** *Proc. IEEE* 2004, **92**(3):401-422.
30. Janson S: *Gaussian Hilbert Spaces.* Cambridge: Cambridge University Press; 1997.
31. Ernst OG, Mugler A, Starkloff HJ, Ullmann E: **On the convergence of generalized polynomial chaos expansions.** *ESAIM Math. Model. Numer. Anal.* 2012, **46**:317-339.
32. Guglielmi N, Hairer E: **Users' guide for the code RADAR5 - version 2.1.** *Technical Report - available at* <http://www.unige.ch/~hairer/software.html> 2005.
33. Hairer E, Wanner G: *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems.* Berlin: Springer; 1996.
34. Augustin F: *Generalized Wiener Expansions for the Numerical Solution of Random Ordinary Differential Equations.* München: Dr. Hut; 2012.
35. Hairer E, Nørsett SP, Wanner G: *Solving Ordinary Differential Equations I.* Berlin: Springer; 1993.
36. Villegas M: **Biofilms with quorum sensing: A numerical approach for delay differential equations with uncertainties.** *Berichte aus den numerischen Arbeitsgruppen.* Technische Universität München; 2011, 8/2011.

doi:10.1186/2190-5983-3-2

Cite this article as: Augustin et al.: An accuracy comparison of polynomial chaos type methods for the propagation of uncertainties. *Journal of Mathematics in Industry* 2013 3:2.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com
