Commun. Comput. Phys.
doi: 10.4208/cicp.020709.211209a

Vol. **8**, No. 5, pp. 947-975
November 2010

Nonlinear Stochastic Galerkin and Collocation Methods: Application to a Ferromagnetic Cylinder Rotating at High Speed

Eveline Rosseel[1], Herbert De Gersem[2] and Stefan Vandewalle[1,*]

[1] *Computer Science Department, K. U. Leuven, Celestijnenlaan 200A, 3001 Leuven, Belgium.*
[2] *Faculty of Sciences, K. U. Leuven, Etienne Sabbelaan 53, 8500 Kortrijk, Belgium.*

Received 2 July 2009; Accepted (in revised version) 21 December 2009

Available online 31 May 2010

Abstract. The stochastic Galerkin and stochastic collocation method are two state-of-the-art methods for solving partial differential equations (PDE) containing random coefficients. While the latter method, which is based on sampling, can straightforwardly be applied to nonlinear stochastic PDEs, this is nontrivial for the stochastic Galerkin method and approximations are required. In this paper, both methods are used for constructing high-order solutions of a nonlinear stochastic PDE representing the magnetic vector potential in a ferromagnetic rotating cylinder. This model can be used for designing solid-rotor induction machines in various machining tools. A precise design requires to take ferromagnetic saturation effects into account and uncertainty on the nonlinear magnetic material properties. Implementation issues of the stochastic Galerkin method are addressed and a numerical comparison of the computational cost and accuracy of both methods is performed. The stochastic Galerkin method requires in general less stochastic unknowns than the stochastic collocation approach to reach a certain level of accuracy, however at a higher computational cost.

AMS subject classifications: 35K60, 65N35, 65C50, 78M25

Key words: Nonlinear PDE with random coefficients, polynomial chaos, stochastic collocation method, stochastic Galerkin method, electromagnetics.

1 Introduction

Ferromagnetic cylinders rotating at high speeds can be found as part of solid-rotor induction machines in various machining tools and in magnetic brakes. At high speeds, when the surface layer of a ferromagnetic rotor gets fully saturated, solid-rotor induction

*Corresponding author. *Email addresses:* Eveline.Rosseel@cs.kuleuven.be (E. Rosseel), Herbert.DeGersem@kuleuven-kortrijk.be (H. De Gersem), Stefan.Vandewalle@cs.kuleuven.be (S. Vandewalle)

machines and magnetic brakes produce a higher torque [7]. As a consequence, designing solid-rotor devices with high-speed conductive parts requires to take ferromagnetic saturation effects into account. These nonlinear material properties can typically not be quantified exactly. A reliable design needs to deal with uncertainty. One of the goals of this paper is to determine to what extent uncertainty on the magnetic material parameters influences the machine properties. We will express this uncertainty by introducing stochastic variables into the mathematical model, which will take the form of a nonlinear stochastic partial differential equation.

A standard tool for solving stochastic partial differential equations is the Monte Carlo simulation method [11]. Recently also other techniques have been developed that enable to compute high-order accurate stochastic solutions and that try to reduce the large computational cost of Monte Carlo simulations. Amongst these, the stochastic Galerkin finite element method [3,14,39] and the stochastic collocation method [1,20,36] turn out to be very successful [22,35]. Similar to Monte Carlo, the stochastic collocation method is based on sampling. The multidimensional samples, called collocation points, are chosen in order to obtain an exponential convergence rate [1]. The stochastic collocation approach suffers from a curse of dimensionality for problems with many random variables. This problem can be alleviated by a so-called 'sparse grid' construction of collocation points [24,36]. The stochastic Galerkin finite element method, on the other hand, applies spectral finite element theory to convert a stochastic PDE into a set of coupled deterministic PDEs. The number of deterministic PDEs is in general smaller than for the stochastic collocation method for a same level of accuracy. Yet, the coupling of the deterministic PDEs leads to high-dimensional algebraic systems, which are expensive to solve.

In this paper, we will point out how the high-order stochastic Galerkin and collocation techniques can be applied to a nonlinear stochastic model of a ferromagnetic rotor. We consider randomness on the magnetic material properties, and allow variability on the conductivity and on the boundary conditions. The application of the stochastic collocation method to a nonlinear stochastic PDE is fairly straightforward, as it reuses deterministic simulation code. Applying a stochastic Galerkin method to a nonlinear stochastic problem, however, is nontrivial. Only few results on the stochastic Galerkin method applied to a nonlinear stochastic problem are available in the literature [19–22], and none of them treat this particular type of nonlinear convection-diffusion problem. We address the numerical and implementation issues that arise in applying the stochastic Galerkin method to the specified nonlinear model. Further, the question of which method, stochastic Galerkin or stochastic collocation, yields the most accurate solution in the lowest computational time, remains open. We shall compare the accuracy and computational cost of both stochastic solution methods through various numerical experiments. Comparative studies between the performance of the stochastic collocation and Galerkin method are rare and mostly limited to linear PDEs [5,20,35].

This paper is organized as follows. In Section 2, the mathematical model needed to describe a magnetic field in a ferromagnetic rotating cylinder, is presented. Uncertainty on the material parameters is formulated by introducing random variables into

the model. Sections 3 and 4 describe two solution approaches for the stochastic model, namely the stochastic collocation and the stochastic Galerkin method. The latter requires substantial changes to the deterministic solver routines and its computational aspects are further detailed in Section 5. A comparison of the accuracy and the computational cost of both stochastic solution methods is given in Section 6. In Section 7, the propagation of the uncertainty on the material parameters towards the uncertainty on the torque is illustrated by numerical simulations. The main conclusions of the paper and some remaining open issues are formulated in Section 8.

2 Magnetic field in a ferromagnetic rotating cylinder

In this section, we describe the equations that are used to model a magnetic field in a ferromagnetic rotating cylinder. First the main characteristics of the deterministic model and its discretization are given before randomness is introduced into the problem.

2.1 Deterministic model

Model equations. The magnetic field in a rotating cylinder can be described by an Eulerian formulation of the Maxwell equations [7],

$$\nabla\times(\nu\nabla\times\vec{A})-\sigma\vec{v}\times\nabla\times\vec{A}+\sigma\frac{\partial\vec{A}}{\partial t}=\vec{J}_s. \tag{2.1}$$

In (2.1), $\vec{A}$ expresses the magnetic vector potential, which is related to the magnetic flux density $\vec{B}$ as $\vec{B}=\nabla\times\vec{A}$. $\vec{J}_s$ is the applied current density, ν the reluctivity, σ the conductivity and $\vec{v}$ the mechanical velocity.

We shall apply this equation to a 2D model of a ferromagnetic, hollow cylinder modelling the rotor of a cylindrical magnetic brake, as depicted in Fig. 1. The inner radius of the cylinder equals $r_i=0.020m$, the outer radius $r_o=0.050m$ and the length $l_z=1m$. Given that the current is perpendicular to the plane and that the magnetic flux density lays in the plane, (2.1) can be rewritten in a cylindrical coordinate system (r,θ,z) as

$$-\frac{1}{r}\frac{\partial}{\partial r}\left(\nu r\frac{\partial A_z}{\partial r}\right)-\frac{1}{r}\frac{\partial}{\partial\theta}\left(\frac{\nu}{r}\frac{\partial A_z}{\partial\theta}\right)+\sigma\omega_m\frac{\partial A_z}{\partial\theta}+\sigma\frac{\partial A_z}{\partial t}=J_{s,z}, \tag{2.2}$$

with $\vec{A}=(0,0,A_z)$ and $\vec{J}_s=(0,0,J_{s,z})$. The cylinder rotates with an angular velocity ω_m, which is, for our problem, in the range between 0 and 300 rad/s. We assume that the magnetic material behavior is isotropic and nonlinear, i.e., $\vec{H}=\nu(|\vec{B}|)\vec{B}$, where the reluctivity ν depends on the magnitude of the magnetic flux density $\vec{B}=(B_r,B_\theta,0)$, and $\vec{H}=(H_r,H_\theta,0)$ represents the magnetic field strength. The deterministic $|\vec{B}|$-$|\vec{H}|$-characteristic is illustrated in the right part of Fig. 1. The conductivity σ is assumed to be linear. The cylinder is placed in a vertical DC magnetic field with $B_x=0$, $B_y=-\hat{B}_y$. The inner region is free

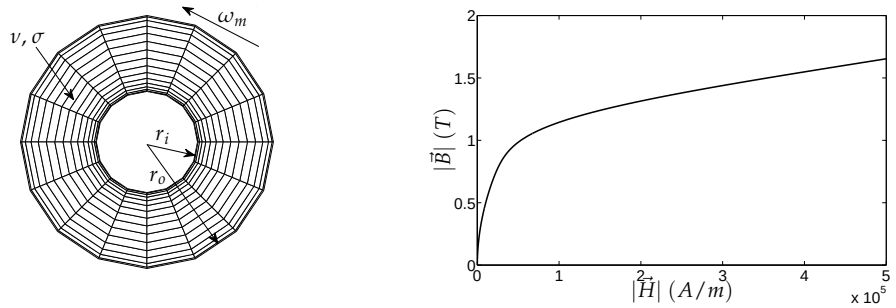


Figure 1: Left: geometry of the model problem with spectral-element grid. Right: $|\vec{B}|$-$|\vec{H}|$-characteristic of the ferromagnetic material.

of magnetic flux. These operating conditions are applied by the boundary conditions $A_z(r_i,\theta)=0$ and $A_z(r_o,\theta)=\hat{B}_y r_0\cos(\theta)$. The magnetic brake is operated without electrical excitation; $J_{s,z}=0$. The applied magnetic field is steady-state, allowing to omit the time-dependent term. Hence, we shall consider the convection-diffusion PDE:

$$-\frac{1}{r}\frac{\partial}{\partial r}\left(\nu r\frac{\partial A_z}{\partial r}\right)-\frac{1}{r}\frac{\partial}{\partial\theta}\left(\frac{\nu}{r}\frac{\partial A_z}{\partial\theta}\right)+\sigma\omega_m\frac{\partial A_z}{\partial\theta}=0. \tag{2.3}$$

Spectral discretization. We discretize (2.3) with a spectral method. In [7], it was shown that this approach offers many advantages. The numerical approximation converges faster than in the case of a finite element method and no stabilization techniques, such as upwinding, are needed. To apply a spectral discretization, the magnetic vector potential is discretized by using Chebyshev polynomials $\hat{T}_m(r)$ in the r-direction and harmonic functions $e^{-\imath\lambda\theta}$ in the θ-direction,

$$A_z(r,\theta)=\sum_{m\in M}\sum_{\lambda\in\Lambda}\mathbf{A}_{z,m,\lambda}\hat{T}_m(r)e^{-\imath\lambda\theta}, \tag{2.4}$$

where M and Λ denote respectively the set of the orders of the Chebyshev polynomials and the harmonic functions, and $\imath$ the imaginary unit. The functions $\hat{T}_m(r)$ correspond to Chebyshev polynomials $T_m(s)$ with $s\in[-1,1]$, shifted and scaled to $r\in[r_i,r_o]$, i.e.,

$$\hat{T}_m(r)=T_m\left(\frac{1}{e_m}\ln\frac{r}{r_m}\right). \tag{2.5}$$

Here, $r_m=\sqrt{r_i r_o}$ is the geometric mean radius of the hollow cylinder with inner radius r_i and outer radius r_o, and $e_m=\ln\sqrt{r_o/r_i}$ is a form factor. The spectral method proceeds by collocating the error at the nodes of a tensor product grid, combining Chebyshev points in the r-direction and equidistantly distributed points in the θ-direction, see Fig. 1.

Linearization. The nonlinear reluctivity ν is linearized by means of Newton's method. The Jacobian of the spectral discretization of (2.3) is expressed in terms of the differential

reluctivity tensor $\bar{\bar{\nu}}_d$ [7]. This tensor, which is function of r and θ, equals

$$\bar{\bar{\nu}}_d = \nu_c \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + 2 \begin{bmatrix} \mathbf{B}_r \\ \mathbf{B}_\theta \end{bmatrix} \left(\frac{d\nu}{d|\vec{B}|^2} \right) \begin{bmatrix} \mathbf{B}_r & \mathbf{B}_\theta \end{bmatrix}. \tag{2.6}$$

The chord reluctivity $\nu_c(|\vec{B}|)$ and $\frac{d\nu}{d|\vec{B}|^2}(|\vec{B}|)$ have to be evaluated from the material characteristic for each spatial collocation point (r_p,θ_p) according to the local magnitude $|\vec{B}|$ of the magnetic flux density, which follows from the current approximation to the solution A_z. At Newton step $k+1$, the following linear system needs to be solved:

$$\left(\mathbf{C}_{\text{dual}} \mathbf{M}^{(k)}_{\bar{\bar{\nu}}_d} \mathbf{C}_{\text{prim}} + \mathbf{W}_{\text{conv}} \right) \mathbf{A}_z^{(k+1)} = -\mathbf{C}_{\text{dual}} \mathbf{H}_f^{(k)}. \tag{2.7}$$

In Eq. (2.7), the matrices $\mathbf{C}_{\text{dual}} \in \mathbb{R}^{N\times 2N}$, $\mathbf{C}_{\text{prim}} \in \mathbb{R}^{2N\times N}$ and $\mathbf{W}_{\text{conv}} \in \mathbb{R}^{N\times N}$ represent respectively the dual curl matrix, the primary curl matrix and the convection matrix, with N the total number of spatial collocation points used to discretize A_z (2.4). The matrix $\mathbf{M}^{(k)}_{\bar{\bar{\nu}}_d} \in \mathbb{R}^{2N\times 2N}$ is a 2 by 2 block matrix, composed of four diagonal blocks that respectively correspond to $\bar{\bar{\nu}}^{(k)}_{d,rr}$, $\bar{\bar{\nu}}^{(k)}_{d,r\theta}$, $\bar{\bar{\nu}}^{(k)}_{d,\theta r}$ and $\bar{\bar{\nu}}^{(k)}_{d,\theta\theta}$. Each entry of the right-hand side $\mathbf{H}_f^{(k)} \in \mathbb{R}^{2N\times 1}$ represents the magnetic field strength at the crossing point of the tangent line that linearizes the $|\vec{B}|$-$|\vec{H}|$-curve in the operation point $\mathbf{B}_p^{(k)}$: $\mathbf{H}^{(k)}_{f,p} = \left(\nu^{(k)}_{c,p} - \bar{\bar{\nu}}^{(k)}_{d,p}\right) \mathbf{B}^{(k)}_p$. Further details on the linearization can be found in [8]. In [8], also a comparison of the solution resulting from a linear and a nonlinear reluctivity is given in order to determine the effect of the nonlinearity induced by ν.

Motional eddy current density. Because of the rotation of the cylinder, a current density is induced, which is equal to

$$J_z = \sigma \omega_m \frac{\partial A_z}{\partial \theta}. \tag{2.8}$$

Torque calculation. The simulation procedure is set up to compute the torque as a function of the mechanical velocity. The torque around the rotor axis is computed using the Lorentz-force method from the θ-component of the force density f_θ, as

$$M_z = l_z \int_{r_i}^{r_o} \int_0^{2\pi} f_\theta r^2 d\theta dr. \tag{2.9}$$

For an incompressible, nonlinear, isotropic material, the θ-component of the force density is given by [33]

$$f_\theta = J_z B_r + \frac{B}{r}\frac{\partial H}{\partial \theta} - \frac{1}{r}\frac{\partial w_{\text{magn,co}}}{\partial \theta}, \tag{2.10}$$

where r is the radius and $w_{\text{magn,co}} = \int_0^H B dH$ the magnetic co-energy. Notice that H, B and $w_{\text{magn,co}}$ are scalar fields. The first term in (2.10) is the Lorentz force whereas the

other terms represent the reluctance force. If the material distribution is homogeneous and invariant in the direction of motion, the reluctance force vanishes. Here, the nonlinearity introduces a heterogeneity in the permeability. The torque calculation by volume integration combined with a spectral-element method enables to obtain highly accurate torque values. The alternative Maxwell stress approach is known to be less accurate and is therefore not considered in this work.

2.2 Stochastic model

In the stochastic model, random variables ξ_1, ξ_2, ξ_3 are introduced to represent the variability of the material parameters and the imposed boundary conditions. The stochastic equivalent of (2.3) corresponds to

$$-\frac{1}{r}\frac{\partial}{\partial r}\left(\xi_1 \nu r \frac{\partial A_z}{\partial r}\right)-\frac{1}{r}\frac{\partial}{\partial \theta}\left(\frac{\xi_1 \nu}{r}\frac{\partial A_z}{\partial \theta}\right)+\xi_2 \sigma \omega_m \frac{\partial A_z}{\partial \theta}=0, \tag{2.11}$$

for $r\in]r_i,r_o[$ satisfying (2.5) and $\theta\in[0,2\pi[$. Periodic boundary conditions are applied in the θ-direction, while for $r=r_i$ and $r=r_o$ Dirichlet boundary conditions hold,

$$A_z(r_i,\theta,\xi)=0 \qquad \text{and} \qquad A_z(r_o,\theta,\xi)=\xi_3 \hat{B}_y r_o \cos(\theta). \tag{2.12}$$

In (2.11)-(2.12), ξ is a random vector collecting the random variables present in the model. Besides ξ_1, ξ_2 and ξ_3, additional random variables are introduced in the definition of the parameters that describe the $|\vec{B}|$-$|\vec{H}|$-characteristic. In this paper, we focus on a tanh magnetization curve, as shown in the right panel of Fig. 1. The description of this curve depends on the magnitude of the magnetic flux density at a so-called knee point, H_{knee}, on the final reluctivity at the full saturated range, ν_{final}, and on two reference values for the magnetic flux densities, B^1_{ref} and B^2_{ref}:

$$|\vec{H}|=C(|\vec{B}|)(\xi_4 H_{\text{knee}}+\xi_5 \nu_{\text{final}}|\vec{B}|) \tag{2.13}$$

with

$$C(|\vec{B}|)=\frac{1}{2}\tanh\left(\frac{|\vec{B}|}{\xi_6 B^1_{\text{ref}}}\right)+\frac{1}{2}\tanh\left(\frac{|\vec{B}|}{\xi_7 B^2_{\text{ref}}}\right)^{30}.$$

We randomize these parameters by respectively ξ_4, ξ_5, ξ_6 and ξ_7. We denote by L the total number of random variables. Note that this stochastic model can easily be extended to other $|\vec{B}|$-$|\vec{H}|$-characteristics, in which case other parameters can be perturbed in a similar way. Since the reluctivity ν is nonlinear, it depends on the entire ξ-vector.

We assume that all random variables are independent. A joint probability density function $\rho(y)$ can then be computed as $\rho(y)=\prod_{i=1}^{L}\rho_i(y_i)$, where $\rho_i(y_i)$ corresponds to the

marginal probability distribution of ξ_i with $y_i \in \Gamma_i$, the support of ξ_i. The expected value of ξ, denoted by $\langle \xi \rangle$, is defined as

$$\langle \xi \rangle = \int_\Gamma y\rho(y)dy, \quad \text{with} \quad y=(y_1,\cdots,y_L)\in\Gamma=\prod_{i=1}^{L}\Gamma_i.$$

Remark 2.1. In our numerical simulations, we will model the random variables ξ_i, $i=1,\cdots,L$, as uniformly or lognormally distributed random variables. In the case of a lognormal random variable we have that $\xi_i^{\log}=\exp(\xi_i^{\mathrm{g}})$, where ξ_i^{g} is a Gaussian random variable with mean μ_g and standard deviation σ_g. A lognormal random variable $\xi_i^{\log}$ can be approximated by a Hermite polynomial expansion [13],

$$\xi^{\log}=\exp\left(\mu_g+\frac{1}{2}\sigma_g^2\right)\sum_{q=0}^{\tilde{P}}\frac{\sigma_g^q}{\sqrt{q!}}\psi_q(\xi^{\mathrm{n}}). \tag{2.14}$$

In (2.14), ξ^{n} is a standard normally distributed variable and ψ_q a normalized 1D Hermite polynomial of degree q. This transformation is particularly important for the stochastic Galerkin method, as it enables to represent the solution by a similar expansion with Hermite polynomials. In that case, when the maximum degree of the polynomials used for the discretization of the solution equals P, the maximum degree of the polynomials in (2.14) should be at least $2P$, i.e., $\tilde{P}\geq 2P$ [21]. The mean $\mu_{\log}$ and variance $\sigma_{\log}^2$ of a lognormal random variable can be determined from μ_g and σ_g as follows:

$$\mu_{\log}=\exp\left(\mu_g+\frac{1}{2}\sigma_g^2\right) \qquad \text{and} \qquad \sigma_{\log}^2=\mu_{\log}^2\left(\exp(\sigma_g^2)-1\right).$$

3 Stochastic collocation method

The stochastic collocation method [1, 36] represents the stochastic solution discretely by an expansion with multivariate Lagrange polynomials, $l_k(\xi)$,

$$A_z(r,\theta,\xi)\approx\sum_{k=1}^{N_c}A_z(r,\theta,\zeta_k)l_k(\xi). \tag{3.1}$$

The Lagrange polynomials $l_k(\xi)$ are interpolatory polynomials, defined by a set of multidimensional collocation points, $\{\zeta_1,\cdots,\zeta_{N_c}\}$. Each collocation point ζ_k consists of L components, $\zeta_k=(\zeta_{k,1},\cdots,\zeta_{k,L})$, according to the L random variables present in the problem. The set of polynomials, $\{l_1(\xi),\cdots,l_{N_c}(\xi)\}$, belongs to the space of tensor product polynomials with degree at most $P=(p_1,\cdots,p_L)$. For example, in the case of a full tensor product construction of the polynomials l_k, we have that $N_c=\prod_{i=1}^{L}(p_i+1)$.

The collocation approach proceeds by requiring that the residual, $R(A_z)=\mathcal{L}(A_z)-f$, with $\mathcal{L}(A_z)$ the differential operator given by (2.11), vanishes at each collocation point,

$$R(A_z(r,\theta,\xi))|_{\zeta_k}=0 \qquad \text{for} \quad k=1,\cdots,N_c.$$

From the properties of Lagrange interpolation, this is equivalent to solving a set of N_c decoupled deterministic PDEs, each of type (2.11), with ξ replaced by the appropriate collocation point ζ_k.

3.1 Solution statistics

After solving the deterministic PDEs, the mean of the solution, denoted by $\langle A_z \rangle$, is obtained by evaluating the expected value of (3.1):

$$\langle A_z \rangle(r,\theta) = \sum_{k=1}^{N_c} A_z(r,\theta,\zeta_k)\langle l_k \rangle \qquad \text{with} \quad \langle l_k \rangle = \int_\Gamma l_k(y)\rho(y)dy,$$

where $\rho(y)$ is the joint probability density function of the random vector ξ and $y \in \Gamma$, the support of ξ. In a similar way other moments of the solution can be obtained, for example the variance corresponds to

$$\mathrm{var}(A_z)(r,\theta) = \int_\Gamma \left(A_z(r,\theta,y)^2 - \left(\langle A_z \rangle(r,\theta)\right)^2 \right) \rho(y)dy.$$

Note that an explicit evaluation of the Lagrange polynomials when computing the expected values $\langle l_k \rangle$ can be avoided by choosing the set of collocation points to be a cubature point set [35, 36]. In that case, the integrals $\langle l_k \rangle$ reduce to the cubature weights $w_1, \cdots, w_{N_c}$ and the variance can be written as [18]

$$\mathrm{var}(A_z)(r,\theta) = \sum_{k=1}^{N_c} A_z^2(r,\theta,\zeta_k) w_k - \left(\langle A_z \rangle(r,\theta)\right)^2.$$

Remark 3.1. Instead of an expansion with Lagrange polynomials, also an expansion with multivariate orthogonal polynomials can be employed [34]. The solution is then represented by the same type of expansion, called generalized polynomial chaos (gPC) expansion, as in the stochastic Galerkin method, see also Section 4 and Eq. (4.1). As in the case of expansion (3.1), the stochastic collocation method results in a number of decoupled deterministic PDEs. The solution statistics are readily obtained from the gPC expansion. Since the stochastic collocation method only computes an approximation for the gPC expansion of the solution, an aliasing error occurs [35].

3.2 Construction of collocation points

The main advantage of the stochastic collocation method is that it leads to a set of decoupled deterministic PDEs. Simulation code available for the deterministic problem [7] can immediately be reused and parallelization of the algorithm is straightforward. However, to obtain a high accuracy and at the same time a limited computational complexity, the choice of the set of collocation points turns out to be crucial.

The multidimensional collocation points are based on one-dimensional collocation points which are typically constructed as Clenshaw-Curtis points or as Gauss points, i.e., zeros of polynomials that are orthogonal w.r.t. the probability density function of a random variable [24]. A multidimensional set is then obtained by either considering a tensor product grid or a sparse grid of the one-dimensional points.

Tensor product grid. Given a 1D set of collocation points, $y^i=\{y^i_1,\cdots,y^i_{p_i+1}\}$, a 1D Lagrange interpolation operator is defined for a smooth function $f(y)$ as

$$\mathcal{U}^i(f)(y)=\sum_{j=1}^{p_i+1} f(y^i_j)h^i_j(y) \qquad \text{with} \quad h^i_j(y)=\prod_{m=1,m\neq j}^{p_i+1}\frac{y-y^i_m}{y^i_j-y^i_m}, \tag{3.2}$$

where $h^i_j(y)$ is a basis Lagrange polynomial of degree p_i. A tensor product formula combines sequences of 1D interpolation operators into a multidimensional operator:

$$(\mathcal{U}^{i_1}\otimes\cdots\otimes\mathcal{U}^{i_L})(f)(y)=\sum_{j_1=1}^{p_{j_1}+1}\cdots\sum_{j_L=1}^{p_{j_L}+1} f(y^{i_1}_{j_1},\cdots,y^{i_L}_{j_L})\left(h^{i_1}_{j_1}(y)\otimes\cdots\otimes h^{i_L}_{j_L}(y)\right).$$

This multidimensional operator defines the interpolatory polynomials $l_k(\xi)$ in Eq. (3.1), with y replaced by ξ. The tensor product collocation method based on Gauss points results in an exponential convergence of the probability error w.r.t. the degree of the Lagrange polynomials [1]. It requires however $N_c=\prod_{j=1}^{L}(p_j+1)$ deterministic simulations. This amount becomes rapidly impractical for a large L or p_j.

Sparse grids. The sparse grid construction of the collocation points aims at reducing the number of points needed for a full tensor product grid, while maintaining the approximating quality up to a logarithmic factor. An algebraic convergence of the error w.r.t. the total number of collocation points was demonstrated in [24]. Sparse grids correspond to a subset of full tensor product grids. The Smolyak algorithm [4,29] introduces an integer w, called level, and defines an L-dimensional interpolation operator $\mathcal{A}(w,L)$ as

$$\mathcal{A}(w,L)=\sum_{w-L+1\leq|\mathbf{i}|\leq w}(-1)^{w-|\mathbf{i}|}\cdot\begin{pmatrix}L-1\\ w-|\mathbf{i}|\end{pmatrix}\cdot(\mathcal{U}^{i_1}\otimes\cdots\otimes\mathcal{U}^{i_L}), \tag{3.3}$$

with $|\mathbf{i}|=i_1+\cdots+i_L$, $\mathbf{i}\in\mathbb{N}^L$ and $\mathcal{U}^{i_m}$ defined by (3.2). To compute $\mathcal{A}(w,L)(f)$, one only needs to know function values evaluated at the sparse grid,

$$\bigcup_{w-L+1\leq|\mathbf{i}|\leq w}(y^{i_1}\times\cdots\times y^{i_L}).$$

When Clenshaw-Curtis points are used, the collocation points corresponding to various Smolyak levels are nested by choosing the number of collocation points, p_i+1, in each level according to the relation $p_i=2^i$ for $i>0$ and $p_0=0$ [12, 24]. As a consequence, fewer deterministic simulations are needed in comparison to a non-nested sparse grid construction with Gauss points, for a fixed Smolyak level w.

4 Stochastic Galerkin method

The stochastic Galerkin method [14,39] proceeds in two steps. First, a spectral discretization of the stochastic space is constructed. This enables to represent the stochastic solution discretely as

$$A_z(r,\theta,\xi) \approx \sum_{q=1}^{Q} A_z(r,\theta)^{(q)} \Psi_q(\xi). \tag{4.1}$$

Expansion (4.1) is called a generalized polynomial chaos (gPC) expansion. It employs a set of multivariate, orthogonal polynomials $\Psi = [\Psi_1, \cdots, \Psi_Q]^T$ that are defined on the random variables ξ present in the stochastic model. By choosing the polynomials to be orthogonal w.r.t. the joint probability density function of ξ, exponential convergence can be achieved [3,38]. In the case of Gaussian distributed random variables, Hermite polynomials will be chosen, and, in the uniformly distributed case, Legendre polynomials. The total number of polynomials Q in expansion (4.1) depends on the number of random variables L and on the order P of the polynomials Ψ_q as follows: $Q = \frac{(L+P)!}{L!P!}$ [16].

Secondly, a Galerkin projection is applied to the residual, $R(A_z) = \mathcal{L}(A_z) - f$,

$$\langle R(A_z)\Psi_q \rangle \equiv \int_\Gamma R(A_z)\Psi_q(y)\rho(y)dy = 0 \qquad \forall q = 1, \cdots, Q. \tag{4.2}$$

This yields a set of Q coupled deterministic PDEs. For general nonlinearities, the integral in (4.2) cannot be calculated analytically [22]. It can either be numerically approximated or the problem can be linearized first before applying the Galerkin condition. In the former case, linearization can be avoided by applying a quasi-Newton method [21]. The latter approach enables to apply standard stochastic Galerkin techniques for linear stochastic PDEs to the linearized problem. Since a Newton linearization is already available for the deterministic problem, we consider the second approach and extend the Newton linearization to the stochastic Galerkin case.

4.1 Newton linearization

Differential reluctivity tensor. The differential reluctivity tensor, $\bar{\bar{\nu}}_d$, defined in (2.6), becomes for the stochastic model (2.11) a random field that depends on r, θ and ξ. Based on the gPC representation of A_z in (4.1), $\bar{\bar{\nu}}_d$ can be reformulated by a similar expansion:

$$\bar{\bar{\nu}}_d(r,\theta,\xi) = \sum_{q=1}^{\infty} \bar{\bar{\nu}}_d^{(q)}(r,\theta)\Psi_q(\xi). \tag{4.3}$$

We consider first an infinite term expansion. After constructing the linearized systems and applying the stochastic Galerkin projection, we will show that the high-order terms for

$$q > Q_\nu = \frac{(L+2P)!}{L!(2P)!} \tag{4.4}$$

cancel out due to the orthogonality properties of the polynomials Ψ_q. This corresponds to using a gPC expansion of order $2P$, with P the order for approximating the solution in (4.1).

To calculate the coefficient functions $\bar{\bar{\nu}}_d^{(q)}$, we start from definition (2.6) where we represent $\vec{B}$ by a polynomial chaos expansion:

$$\vec{B}=(B_r,B_\theta,0)\approx\sum_{q=1}^{Q}(B_r^{(q)},B_\theta^{(q)},0)\Psi_q(\xi),\quad\text{with}\quad B_r^{(q)}=\frac{1}{r}\frac{\partial A_z^{(q)}}{\partial\theta},\quad B_\theta^{(q)}=-\frac{\partial A_z^{(q)}}{\partial r}. \tag{4.5}$$

Eq. (4.5) immediately follows from (4.1) and the relation $\vec{B}=\nabla\times\vec{A}$. The stochastic differential reluctivity tensor $\bar{\bar{\nu}}_d$ then results in

$$\bar{\bar{\nu}}_d(r,\theta,\xi)=\nu_c(r,\theta,\xi)\begin{bmatrix}1&0\\0&1\end{bmatrix}+2\sum_{i=1}^{Q}\sum_{j=1}^{Q}\begin{bmatrix}B_r^{(i)}\\B_\theta^{(i)}\end{bmatrix}\left(\frac{d\nu(r,\theta,\xi)}{d|\vec{B}|^2}\right)\begin{bmatrix}B_r^{(j)}&B_\theta^{(j)}\end{bmatrix}\Psi_i\Psi_j. \tag{4.6}$$

This tensor is to be computed for each spatial collocation point (r_p,θ_p) and evaluated for the current approximation to the solution A_z. In order to represent (4.6) by expansion (4.3), a separate gPC representation for the two terms in the right-hand side of (4.6) is constructed. First, a gPC representation for the chord reluctivity $\nu_c(r,\theta,\xi)$ is constructed,

$$\nu_c(r,\theta,\xi)=\sum_{q=1}^{\infty}\nu_c(r,\theta)^{(q)}\Psi_q(\xi)\quad\text{with}\quad\nu_c(r,\theta)^{(q)}=\langle\nu_c(r,\theta,\xi)\Psi_q\rangle/\langle\Psi_q^2\rangle. \tag{4.7}$$

From the orthonormality of the polynomials Ψ_q, it follows that $\langle\Psi_q^2(\xi)\rangle\equiv1$. The chord reluctivity is defined as $\nu_c=|\vec{H}|/|\vec{B}|$, where the magnitude of $\vec{B}$ is given by

$$|\vec{B}|=\sqrt{\left(\sum_{q=1}^{Q}B_r^{(q)}\Psi_q\right)^2+\left(\sum_{q=1}^{Q}B_\theta^{(q)}\Psi_q\right)^2}=\sqrt{\sum_{i=1}^{Q}\sum_{j=1}^{Q}\left(B_r^{(i)}B_r^{(j)}+B_\theta^{(i)}B_\theta^{(j)}\right)\Psi_i\Psi_j}, \tag{4.8}$$

as follows from Eq. (4.5), and where $|\vec{H}|$ depends on the definition of the magnetization curve (2.13). Since $|\vec{B}|$ cannot be exactly represented by a polynomial chaos expansion — due to the square root in (4.8) — the gPC coefficients $\nu_c(r,\theta)^{(q)}$ need to be calculated approximately by using numerical integration methods. Therefore, no separate gPC expansion is constructed for $|\vec{B}|$, but Eq. (4.8) is applied directly to evaluate ν_c in the numerical integration procedure. The numerator $\langle\nu_c(r,\theta,\xi)\Psi_q\rangle$ in (4.7) corresponds to an L-dimensional integral. It can be approximated by using sparse cubature rules [15, 21]:

$$\begin{aligned}\langle\nu_c(r,\theta,\xi)\Psi_q(\xi)\rangle&=\int_{\Gamma_1}\cdots\int_{\Gamma_L}\nu_c(r,\theta,z)\Psi_q(z)\rho(z)dz\\&\approx\sum_{k=1}^{N_c}w_k\nu_c(r,\theta,\zeta_k)\Psi_q(\zeta_k)\hat{\rho}(\zeta_k),\end{aligned} \tag{4.9}$$

where $\{\zeta_1,\cdots,\zeta_{N_c}\}$ is a set of N_c integration points with corresponding weights $w_1,\cdots,w_{N_c}$. The construction of the integration points is similar to the sparse grid construction described in Section 3.2 based on Gauss points. The scaling factor $\hat{\rho}(\zeta_k)$ takes the difference between the weighting function used to compute the Gauss points and the probability density function $\rho(z)$ into account. For example, in the case of L random variables ξ_i, uniformly distributed on $[-1,1]$, and Gauss-Legendre cubature points, $\hat{\rho}=1/2^L$.

Next, a gPC representation for the second term in (4.6) is constructed. A small derivation shows that the derivative $d\nu/d|\vec{B}|^2$ is equal to

$$\frac{d\nu}{d|\vec{B}|^2}=\frac{d|\vec{H}|/d|\vec{B}|-\nu_c}{2|\vec{B}|^2},$$

where $d|\vec{H}|/d|\vec{B}|$ can be determined from the derivative of the magnetization curve w.r.t. $|\vec{B}|$. As before, $d\nu/d|\vec{B}|^2$ cannot exactly be represented by a gPC expansion. Therefore, no gPC expansion is constructed separately for $d\nu/d|\vec{B}|^2$, instead an approximate gPC expansion is numerically computed for the entire second term in (4.6). The gPC coefficients are calculated by sparse high-dimensional cubature rules similar to (4.9).

Linearized system. The gPC representation of $\bar{\bar{\nu}}_d$ (4.3) together with a spectral spatial discretization results in the following linearized system at Newton step $k+1$:

$$\begin{aligned}&\left(\sum_{i=1}^{\infty}\Psi_i\mathbf{C}_{\text{dual}}\mathbf{M}^{(k)}_{\bar{\bar{\nu}}_d^{(i)}}\mathbf{C}_{\text{prim}}+(1+\xi_2)\mathbf{W}_{\text{conv}}\right)\left(\sum_{q=1}^{Q}\mathbf{A}_z^{(q)(k+1)}\Psi_q\right)\\&=-\left(\sum_{i=1}^{\infty}\Psi_i\mathbf{C}_{\text{dual}}\left(\mathbf{M}^{(k)}_{\nu_c^{(i)}}-\mathbf{M}^{(k)}_{\bar{\bar{\nu}}_d^{(i)}}\right)\mathbf{C}_{\text{prim}}\right)\left(\sum_{q=1}^{Q}\mathbf{A}_z^{(q)(k)}\Psi_q\right),\end{aligned}$$

where $\mathbf{M}^{(k)}_{\bar{\bar{\nu}}_d^{(i)}}\in\mathbb{R}^{2N\times 2N}$ is composed of 4 diagonal $(N\times N)$-matrices containing respectively $\bar{\bar{\nu}}^{(k)(i)}_{d,rr}$, $\bar{\bar{\nu}}^{(k)(i)}_{d,r\theta}$, $\bar{\bar{\nu}}^{(k)(i)}_{d,\theta r}$, $\bar{\bar{\nu}}^{(k)(i)}_{d,\theta\theta}$ on the diagonal. The computation of the right-hand side relies on a gPC expansion for the chord reluctivity ν_c, as defined in (4.7).

Galerkin condition. The stochastic Galerkin method proceeds by imposing orthogonality of the residual w.r.t. the polynomial chaos. This yields a linear algebraic system of NQ equations:

$$\begin{aligned}&\left(\sum_{i=1}^{Q_\nu}\mathbf{D}_i\otimes\mathbf{C}_{\text{dual}}\mathbf{M}^{(k)}_{\bar{\bar{\nu}}_d^{(i)}}\mathbf{C}_{\text{prim}}+(I_Q+\mathbf{G}_2)\otimes\mathbf{W}_{\text{conv}}\right)\mathbf{A}_z^{(k+1)}\\&=-\sum_{i=1}^{Q_\nu}\mathbf{D}_i\otimes\mathbf{C}_{\text{dual}}\left(\mathbf{M}^{(k)}_{\nu_c^{(i)}}-\mathbf{M}^{(k)}_{\bar{\bar{\nu}}_d^{(i)}}\right)\mathbf{C}_{\text{prim}}\mathbf{A}_z^{(k)}\qquad\text{with}\quad\mathbf{A}_z^{(k)}=[\mathbf{A}_z^{(1)(k)}\cdots\mathbf{A}_z^{(Q)(k)}]^T.\end{aligned}\tag{4.10}$$

The matrix $I_Q=\langle\Psi\Psi^T\rangle\in\mathbb{R}^{Q\times Q}$ is an identity matrix, as follows from the orthonormality of the polynomials Ψ_q. The matrices $\mathbf{D}_i,\mathbf{G}_2\in\mathbb{R}^{Q\times Q}$ characterize the stochastics of the

problem and are respectively defined by $\mathbf{D}_i = \langle \Psi_i \Psi \Psi^T \rangle$ and $\mathbf{G}_2 = \langle \xi_2 \Psi \Psi^T \rangle$. Properties of these matrices are given in [10, 27]. Each individual matrix is a sparse matrix, however the sum, $\sum_{i=1}^{Q_\nu} \mathbf{D}_i$, is a full matrix [15]. Due to the orthogonality of the polynomials, all matrices

$$\mathbf{D}_i = \langle \Psi_i \Psi \Psi^T \rangle \quad \text{with} \quad i > \frac{(L+2P)!}{L!(2P)!}$$

are identically zero so that the gPC expansion of $\bar{\bar{\nu}}_d$ (4.3) and ν_c (4.7) can be truncated after Q_ν terms, where Q_ν is defined by (4.4). This explains the range of the summations in (4.10). The gPC expansions (4.3) and (4.7) can therefore be limited to a gPC order $2P$, with P the order used for approximating the solution in (4.1).

4.2 Post-processing

Solution statistics. As noted in Remark 3.1, solution statistics can be easily obtained from a polynomial chaos representation (4.1). For example, the mean and variance of A_z correspond respectively to

$$\langle A_z \rangle (r,\theta) = A_z(r,\theta)^{(1)} \qquad \text{and} \qquad \text{var}(A_z)(r,\theta) = \sum_{q=2}^{Q} \left(A_z(r,\theta)^{(q)} \right)^2. \tag{4.11}$$

Torque calculation. Since the torque (2.9) indirectly depends on A_z, a gPC representation for M_z can be constructed as follows:

$$\begin{aligned}
M_z &\approx \sum_{i=1}^{Q_\nu} M_z^{(q)} \Psi_q, \\
M_z^{(q)} &= -l_z \sigma \omega_m \sum_{i=1}^{Q} \sum_{j=1}^{Q} \langle \xi_2 \Psi_q \Psi_i \Psi_i \rangle \int_{r_i}^{r_o} \int_0^{2\pi} \frac{\partial A_z^{(i)}}{\partial \theta} B_r^{(j)} r^2 d\theta dr \\
&\quad + l_z \sum_{i=1}^{Q} \sum_{j=1}^{Q} \langle \Psi_q \Psi_i \Psi_i \rangle \int_{r_i}^{r_o} \int_0^{2\pi} \frac{|\vec{B}|^{(i)}}{r} \frac{\partial |\vec{H}|^{(j)}}{\partial \theta} r^2 d\theta dr \\
&\quad - l_z \int_{r_i}^{r_o} \int_0^{2\pi} \frac{1}{r} \frac{\partial w_{\text{magn,co}}^{(q)}}{\partial \theta} r^2 d\theta dr,
\end{aligned}$$

with J_z replaced by $\xi_2 \sigma \omega_m \frac{\partial A_z}{\partial \theta}$ and $|\vec{B}|$ given by (4.8). This gPC expansion is based on gPC expansions for $|\vec{B}|$, $|\vec{H}|$ and $w_{\text{magn,co}}$, which can be constructed approximately by using the definition of the tanh magnetization curve and sparse cubature rules, as in (4.9). From this representation, statistics of the torque can be computed, in a similar way to (4.11). Alternatively, one could sample the gPC representation of the solution A_z, and apply a Monte Carlo approach to determine the statistics of the torque.

5 Computational aspects of the stochastic Galerkin method

5.1 Newton's method

Existence of a solution and convergence. Newton's method applied to the deterministic problem (2.3) converges generally within 10 to 20 iterations. A similar convergence is observed for the stochastic nonlinear problem. This property does only hold, however, when the introduction of random variables into the model does not violate the necessary conditions so that a solution to (2.11) exist. These conditions include that the stochastic coefficient $\nu(r,\theta,\xi;A_z)$ is bounded and strictly positive [2], i.e., that there exist positive constants α and β such that

$$\mathcal{P}(0<\alpha\leq\nu(r,\theta,\xi;A_z)\leq\beta<\infty)=1 \quad \forall(r,\theta)\in[r_i,r_o]\times[0,2\pi[,$$

where $\mathcal{P}$ is the probability measure used to describe the probability space upon which the random variables ξ are defined.

In practice, we have observed that the convergence of the Newton's iteration (4.10) can deteriorate in the presence of roundoff and approximation errors. Especially the computation of the gPC representations for ν_c (4.7) and $\bar{\bar{\nu}}_d$ (4.3) is a potential source of roundoff errors: the numerical integration should be accurate enough. The Smolyak cubature formulas applied in (4.3) and (4.7) integrate multivariate polynomials exactly if their total polynomial degree is at most $2w+1$ [21], with w the Smolyak level as defined in (3.3). Besides these errors it is also important to note that certain parameter variations do not yield convergent Newton iterations. This issue originates from the description of the magnetization curve: also for the deterministic problem not all parameter combinations lead to convergent Newton iterations. Therefore, for some parameters, for example B^1_{ref}, only small variances of the corresponding random variables are allowed.

Remark 5.1. The solution (4.1) is represented by a global polynomial approximation as a function of the random input variables. This global approximation can only be justified when the behavior of the solution does not abruptly change with respect to the random input parameters. Otherwise, local polynomial approximations could be a solution, e.g., a multi-element generalized chaos or multi-wavelet expansion as discussed in [3,17,31]. The derivation of the linear algebraic systems (4.10) remains the same in the case of a multi-element polynomial approximation. Only the matrices $\mathbf{D}_i=\langle\Psi_i\Psi\Psi^T\rangle$ and $\mathbf{G}_2=\langle\xi_2\Psi\Psi^T\rangle$ will change and might require other iterative solvers than the ones proposed in Section 5.2. For the relatively small uncertainty on the input parameters considered here, a smooth behavior of the solution is observed, thus enabling the use of a global polynomial approximation.

Initial guess. Each Newton iteration (4.10) involves an expensive solve of a large system. As such, the choice of a good initial guess can reduce the computational time significantly. The hierarchical structure of the polynomial chaos enables a straightforward procedure for determining a good initial guess:

- set $A_z = 0$;
- for p from 0 to P:
  - initial guess $\hat{A}_z = [A_z^T \, \mathbf{0}^T]^T \in \mathbb{R}^{NQ \times 1}$, with $Q = \frac{(L+p)!}{L!p!}$ and $\mathbf{0} \in \mathbb{R}^{NQ_d \times 1}$, an all-zero vector with $Q_d = \frac{(L+p-1)!}{(L-1)!p!}$;
  - solve the nonlinear stochastic problem (2.11) with initial guess $\hat{A}_z$ for A_z represented by a chaos expansion (4.1) of order p.

Note that the first iteration, with chaos order 0, corresponds to solving a deterministic problem. The following example illustrates the possible reduction in computational time by using the above initialization procedure.

Example 5.1. The stochastic problem (2.11) is solved with 5 uniformly distributed random variables: ξ_1 on $[-0.97,1.03]$, ξ_2 and ξ_3 on $[-0.99,1.01]$, ξ_5 and ξ_6 on $[-0.98,1.02]$. The boundary condition (2.12) is imposed with magnitude $\hat{B}_y r_o = 1$ Tm. The conductivity σ is equal to 10^7 S/m. The spatial discretization, represented in Fig. 1, uses 16 degrees of freedom in both the r- and θ-dimension, corresponding to $N = 256$ spatial unknowns. The angular velocity ω_m equals 291 rad/s. In Table 1, a comparison of the number of Newton steps and the total computational time is given between using an all-zero initial guess or an initial guess created by the described procedure. The computations were performed on a 2.00 GHz Intel Dual Core processor T7300 with 2.0 GByte RAM.

Table 1: Number of Newton steps and total computational time when solving the stochastic problem (2.11) starting from an all-zero initial guess for A_z or an iteratively refined initial guess.

all-zero initial guess: total time = 8187 sec.	
$P=4$, $Q=126$	13 Newton steps
hierarchically constructed initial guess: total time = 1984 sec.	
$P=4$, $Q=126$	2 Newton steps
intermediate Newton steps for $p<P$	13 steps for $p=0, Q=1$
	6 steps for $p=1, Q=6$
	3 steps for $p=2$, $Q=21$
	3 steps for $p=3$, $Q=56$

Remark 5.2. The choice of a good initial guess might also speed-up the stochastic collocation method. When using the solution obtained with a previous stochastic collocation point as initial guess for computing the solution with another collocation point, this will generally not result in a reduction of the number of Newton steps, unless perhaps a certain pre-ordering of the stochastic collocation points is applied. However, as each Newton iteration for the stochastic collocation method is relatively cheap — the size of the Jacobian just equals the number of deterministic unknowns, finding a good initial guess is less important for the stochastic collocation method.

5.2 Solving the linearized systems

A spectral spatial discretization of the linearized deterministic problem (2.2) typically leads to small, but almost dense systems as in Eq. (2.7). The systems are nonsymmetric and poorly conditioned. To solve these systems, direct solution methods are generally most appropriate [6]. The stochastic Galerkin method results after linearization in a Kronecker product system matrix (4.10) containing such dense blocks. For a small number of spatial and random unknowns, direct solution methods can still be applied. The number of random unknowns, however, rapidly increases when a high order polynomial chaos is used. For increasing system sizes, the computational cost and memory requirements of direct solution methods become prohibitively large. In that case, iterative solution schemes are required for solving (4.10).

Iterative solvers. In [27], various iterative solvers for linear stochastic finite element discretizations with the same Kronecker product structure as (4.10) are discussed. It was shown there that a multigrid based solver [28] yields optimal convergence properties, i.e., a convergence rate independent of the spatial and stochastic discretization parameters. For spectral spatial discretizations, multigrid methods can be designed [40]. However, their improvement in convergence rate compared to single grid schemes is rather modest, especially for variable-coefficient problems [6].

Alternatively, block splitting methods that make use of the Kronecker product structure of (4.10), can be applied [27]. These methods only split the stochastic discretization matrices, i.e., the $\mathbf{D}_i$ and $\mathbf{G}_2$ matrices in (4.10), so that in every iteration several systems with the size of the number of deterministic unknowns need to be inverted. When these block systems are exactly solved, the cost of one Newton iteration (4.10) is proportional to the cost of Q deterministic Newton iterations, with Q the number of stochastic unknowns. The block systems can be factorized once at the beginning of the block splitting iteration so that in every iteration only triangular solves of the factorized system have to be carried out. The convergence rate of block splitting methods typically decreases with increasing polynomial chaos order or input variance. In practice however, depending on the chaos type used, low computing times can be obtained, especially when the block systems are solved approximately. We shall apply these splitting methods as preconditioner for the Generalized Minimum Residual (GMRES) method.

In Table 2, the performance of several block splitting based preconditioners is illustrated. We consider the mean-based preconditioner [25], the Kronecker product preconditioner [30] and a symmetric block Gauss-Seidel splitting. The deterministic model is described in Section 2.1 and Fig. 1. The angular velocity ω_m is set to 151 rad/s. The spatial discretization uses 16 degrees of freedom in the r- and θ-dimension, corresponding to $N=256$ spatial unknowns. We consider 6 random variables (ξ_1, ξ_3, ξ_4, ξ_5, ξ_6 and ξ_7), that are either lognormally (configuration A) or uniformly (configuration B) distributed. The former corresponds to a Hermite expansion of the solution (4.1), the latter to an expansion with Legendre polynomials. A second-order and a third-order chaos is considered, corresponding to $Q=28$, respectively $Q=84$ random unknowns. In configuration (A) the

Table 2: Number of iterations and solution time in seconds for solving one linearized system (4.10) with GMRES, preconditioned by a mean-based preconditioner, a Kronecker product preconditioner and a symmetric block Gauss-Seidel preconditioner.

Preconditioner	Number of iterations		Solution time	
	lognormal (A)	uniform (B)	lognormal (A)	uniform (B)
chaos order 2				
mean-based	41.8	23.5	18.0	9.9
Kronecker product	40	23	17.3	10.2
symmetric Gauss-Seidel	16.4	11.8	183	149
chaos order 3				
mean-based	69	29.7	344	176
Kronecker product	65	30.7	362	189
symmetric Gauss-Seidel	22.6	14	3958	2724

lognormal distribution (see also Remark 2.1) of the random variables is based on $\mu_g = 0$ and $\sigma_g = 0.05$. In configuration (B) the random variables are uniformly distributed on $[-0.95, 1.05]$. The linear systems are solved to a relative accuracy equal to 10^{-12}. Table 2 presents the average number of iterations and solution time needed for solving the linearized systems, when performing several Newton iterations until convergence of the nonlinear solution. The initial guess for the Newton iterations was constructed with the procedure described in Section 5.1.

The tests indicate that the mean-based preconditioner yields the best performance. Despite the good results for the Kronecker product preconditioner in [30], it does not improve the results of the mean-based preconditioner for this problem. In [30] it was reported that the convergence of the Kronecker product preconditioner is less sensitive to large variations of the random variables than the convergence of the mean-based preconditioner. Therefore, such a preconditioner might perform better than the mean-based preconditioner in the case of a large standard deviation of the random variables.

Because in the mean-based and Kronecker product preconditioner case all block systems use the same system matrix, these methods result in a lower computational cost compared to the block Gauss-Seidel splitting method. The block system matrix needs to be factorized only once and the factorized blocks are then used to apply the preconditioner. The mean-based preconditioner is used in subsequent numerical experiments.

Memory requirements. As the linearized systems (4.10) are almost dense, a lot of memory is required for storing and solving these systems. In practice the linearized systems are not stored entirely, but only the tensor product blocks [26]. For example, a problem with 7 random variables and a 4th order chaos results in a total of 330 random unknowns. Combined with 256 unknowns from the spatial discretization, system (4.10) has dimension 84480×84480. When storing only the dense blocks of the Kronecker product, 6435 matrices of size 256×256 are to be stored. This amounts to about 5.9% of the storage that would be required for the full matrix.

6 Stochastic collocation versus Galerkin method: numerical results

This section deals with a numerical comparison between the stochastic collocation and Galerkin method. While both methods enable to compute all statistics of the stochastic solution of (2.11), their difference in implementation cost, computation cost and accuracy can be significant. The difference in implementation cost was addressed in the previous sections. The stochastic collocation method showed a definite advantage over the stochastic Galerkin method since the former can reuse a deterministic simulation code. To compare the computational cost and accuracy, a set of numerical simulations was performed. In this section we focus on solving the magnetic vector potential from Eq. (2.11). In that case the mechanical velocity ω_m remains fixed and we set ω_m=151 rad/s in our experiments. In the next section, the mechanical velocity is varied and the statistics of the torque are computed. As the torque computations are directly related to the magnetic vector potential, its accuracy obviously follows the accuracy of the latter. The second set of experiments is used to reveal the influence of uncertainty on the magnetic material parameters onto the torque.

The Newton iterations for both stochastic solvers are performed until the relative norm of the Newton increments

$$\left\| A_z^{\text{new}} - A_z^{\text{old}} \right\| / \left\| A_z^{\text{init}} \right\|_2 < 10^{-12}$$

and the absolute norm

$$\left\| A_z^{\text{new}} - A_z^{\text{old}} \right\|_2 < 10^{-2}.$$

The linear systems within a Newton step are solved until the relative residual is smaller than 10^{-12}. When a larger tolerance is used for the linear systems — as is often done in a Newton-Krylov strategy — errors tend to accumulate and disturb the Newton convergence, especially in the stochastic Galerkin case.

A spatial discretization based on 16 degrees of freedom in the r- and θ-dimension is used, which corresponds to $N=256$ spatial unknowns. The computations are performed on a 2.2 GHz Dual Core Opteron processor with 4 GByte RAM.

Remark 6.1. In order to compare the accuracy of the stochastic collocation and Galerkin approach, an "exact" stochastic solution is needed. However, since no analytic solution can be calculated, a high order stochastic solution has been used as reference solution A_z^{ref}. The error on the mean and variance of A_z is then respectively calculated as

$$E_{\text{mean}} = \frac{\| \langle A_z \rangle - \langle A_z^{\text{ref}} \rangle \|_2}{\| \langle A_z^{\text{ref}} \rangle \|_2}, \qquad E_{\text{var}} = \frac{\| \text{var}(A_z) - \text{var}(A_z^{\text{ref}}) \|_2}{\| \text{var}(A_z^{\text{ref}}) \|_2}. \tag{6.1}$$

Approximating an exact solution by a high-order stochastic solution is widely applied in the literature [9,31,36].

6.1 Lognormal random variables and Hermite polynomials

As a first problem, we consider the stochastic problem (2.11), perturbed by 4 independent, lognormally distributed random variables, $\xi = \{\xi_1, \xi_4, \xi_5, \xi_7\}$, all based on a Gaussian variable with $\mu_g = 0$ and $\sigma_g = 0.08$. Since the lognormal random variables are approximated by a Hermite polynomial expansion with standard normal random variables (2.14), the stochastic Galerkin method also uses a Hermite chaos for representing the solution (4.1). The stochastic collocation method constructs 4D collocation points from a full tensor product or a Smolyak sparse grid based on the zeros of Hermite polynomials. The error computation in (6.1) applies a stochastic collocation solution corresponding to a full tensor product grid of 4096 collocation points based on the zeros of 7th degree Hermite polynomials as reference solution.

Convergence and accuracy. In Fig. 2 the error of the mean and variance of A_z is presented as a function of respectively the polynomial order in the stochastic Galerkin case, the polynomial degree p_i in the dense stochastic collocation case, or the Smolyak level w in the sparse stochastic collocation case. The same polynomial degrees p_i, $i = 1, \cdots, L$, are used for all random variables. We observe that the convergence rate of the stochastic Galerkin and the dense stochastic collocation solution is fairly similar. Although the stochastic discretization is based on scaled Hermite polynomials which are normalized w.r.t. the normal probability density function and the lognormal random variables are approximated following (2.14) by normally distributed random variables, neither of the methods attain exponential convergence w.r.t. to the polynomial degree or polynomial order. In [1], the stochastic collocation method applied to a linear diffusion problem with lognormal diffusion coefficient yielded an exponential convergence of the error w.r.t. polynomial degree. These results apparently do not extend to this nonlinear convection-diffusion problem.

Computational cost. While Fig. 2 illustrates the convergence rate, it does not give information on the computational cost of the various methods. A certain polynomial chaos order P does not require the same number of stochastic unknowns as a certain Smolyak level or polynomial degree used to construct stochastic collocation points. More importantly, the computational cost of the stochastic Galerkin method for a certain number of stochastic unknowns is generally substantially higher than the cost of the stochastic collocation method for the same number of unknowns due to the coupling of the stochastic unknowns in the Galerkin case. To illustrate the computational cost, Fig. 3 displays the decay of the error as a function of either the number of stochastic unknowns or the total computational time.

For this problem the stochastic Galerkin method is more expensive than the stochastic collocation method to attain a similar accuracy.

Remark 6.2. It is remarkable that a sparse grid construction of stochastic collocation points leads for a same number of random unknowns to a substantial less accurate solution than a full tensor product grid collocation point construction. It is possible that the

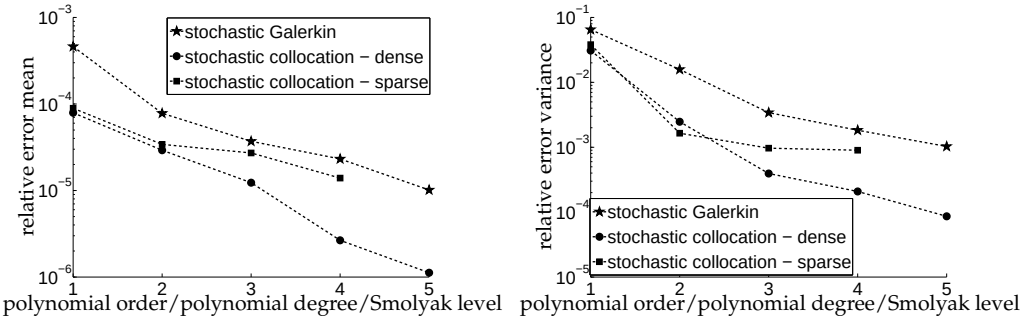


Figure 2: Error of the mean and variance of A_z as a function of respectively the polynomial order in the stochastic Galerkin case, the polynomial degree p_i in the dense stochastic collocation case, or the Smolyak level w in the sparse stochastic collocation case. Problem (2.11) is solved with ξ_1, ξ_4, ξ_5 and ξ_7 modeled as independent, lognormally distributed random variables.

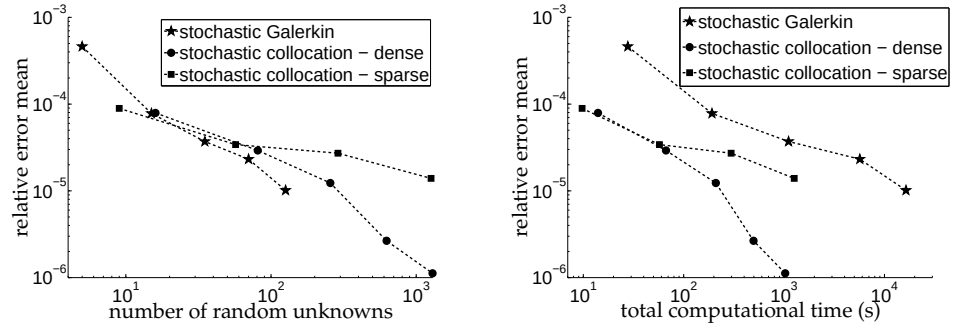


Figure 3: Error of the mean of A_z as a function of either the number of random unknowns (left) or the total computational time (right). Problem (2.11) is solved with ξ_1, ξ_4, ξ_5 and ξ_7 lognormally distributed.

number of stochastic dimensions, i.e., only 4, is not large enough for demonstrating the effectiveness of the sparse collocation method over the dense collocation method. Also we note that the convergence rate of the sparse stochastic collocation solution apparently slows down for the higher Smolyak levels. This might be an artifact of using an insufficiently accurate reference solution, as also occurred in [9]. The reference solution was chosen experimentally as the best possible reference out of the set of obtained solutions.

6.2 Uniform random variables and Legendre polynomials

We consider (2.11)-(2.12) with 7 random variables, uniformly distributed on $[-0.97,1.03]$. The uniform distribution leads in the stochastic Galerkin case to a Legendre polynomial chaos expansion of the solution (4.1). The stochastic collocation points are constructed in a sparse Smolyak or a full tensor product grid framework, based on 1D Clenshaw-Curtis or Gauss-Legendre points. In the full tensor product grid case, the 1D grids correspond to Gauss-Legendre points. Fig. 4 illustrates the mean and variance of the z-component

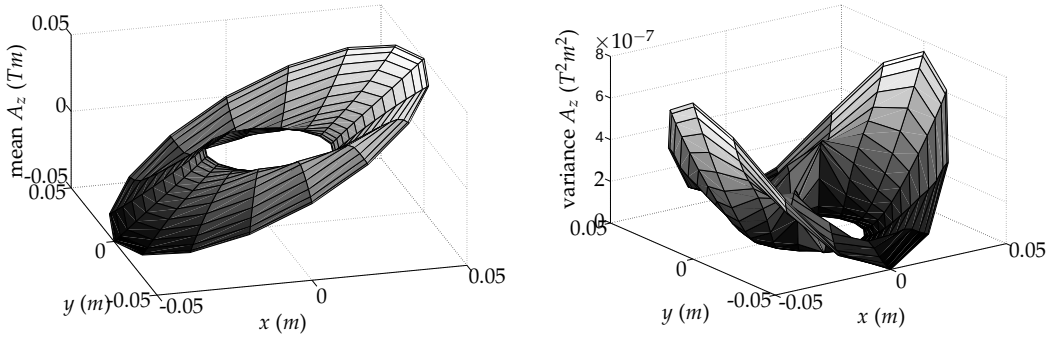


Figure 4: Mean (left) and variance (right) of A_z, the stochastic Galerkin solution of (2.11)-(2.12) with 7 random variables, all uniformly distributed on $[-0.97,1.03]$. The solution is based on a third-order Legendre chaos.

A_z of the magnetic vector potential. A stochastic collocation solution based on 279936 stochastic collocation points that were built by a tensor product grid of zeros of 5th degree Legendre polynomials, is used as reference solution for the error computations.

Convergence and accuracy. In Fig. 5, the error of the mean and variance of A_z is presented as a function of either the polynomial chaos order, or the degree of the polynomials to construct the Gauss points in the stochastic collocation case, or the Smolyak level. An exponential decay of the error as a function of the polynomial degree or the polynomial chaos order is observed. This is consistent with the exponential convergence of the stochastic Galerkin and collocation solution reported in [1,37] which results from the correspondence between the weighting function of Legendre polynomials and an uniform probability density function. A collocation solution based on Clenshaw-Curtis points does not possess such a convergence rate since Clenshaw-Curtis points do not have a similar connection to the uniform distribution function as Gauss-Legendre points.

Computational cost. The computational cost is illustrated in Fig. 6 which displays the error decay as a function of either the number of stochastic unknowns or the total computational time. The stochastic Galerkin method clearly results in the most accurate solution for a fixed number of random unknowns. With respect to the total solution time however, the stochastic Galerkin method does not longer perform better than the stochastic collocation method due to the large cost of solving the high-dimensional linearized systems.

6.3 Combination of distributions

In a third set of experiments, the stochastic model (2.11) contains 7 random variables: ξ_1, ξ_2, ξ_3 ξ_5 and ξ_6 are uniformly distributed on $[-0.99,1.01]$; ξ_4 and ξ_7 are lognormally distributed based on a Gaussian random variable with $\mu_g = 0$ and $\sigma_g = 0.05$. Both Hermite as well as Legendre polynomials are used for representing the stochastic Galerkin

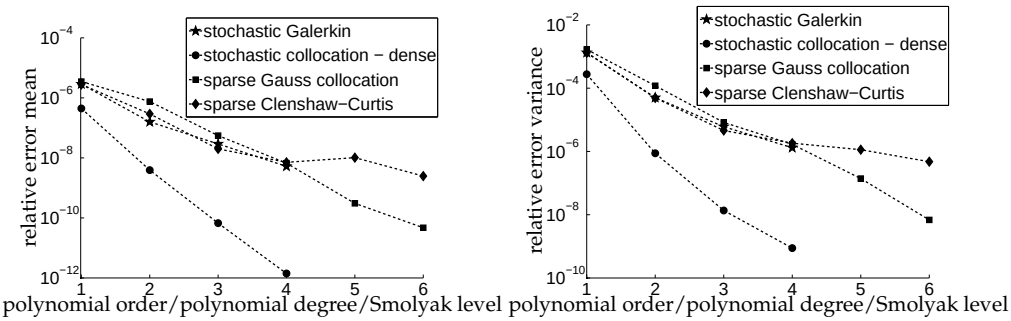


Figure 5: Error of the mean and variance of A_z as a function of either the polynomial order, or the polynomial degree, or the Smolyak level depending on the type of stochastic discretization. Problem (2.11) contains 7 independent and uniformly distributed random variables on $[0.97,1.03]$.

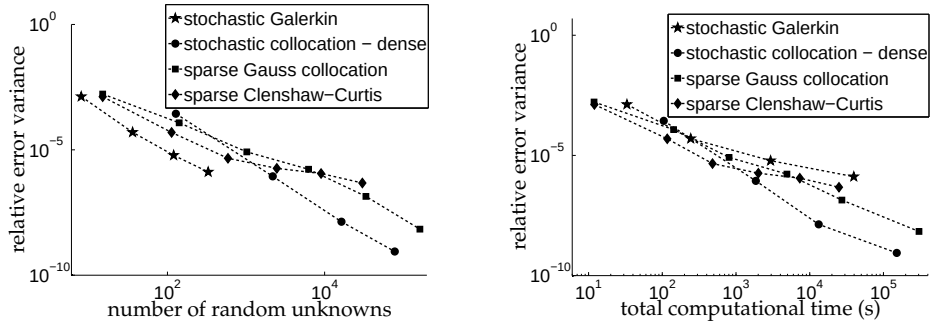


Figure 6: Error of the variance of A_z as a function of the number of stochastic unknowns (left) and of the total solution time (right). Problem (2.11) contains 7 independent and uniformly distributed random variables on $[0.97,1.03]$.

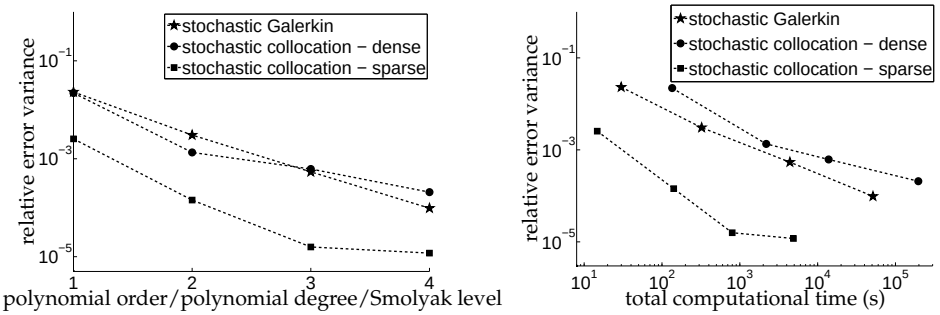


Figure 7: Error of the variance of A_z as a function of the polynomial degree/order/Smolyak level (left) and as a function of the total computational time (right). Problem (2.11) is solved with ξ_1, ξ_2, ξ_3 ξ_5 and ξ_6 modeled as independent, uniformly distributed random variables, and ξ_4 and ξ_7 as lognormally distributed random variables.

solution (4.1) and for calculating the stochastic collocation points. The reference solution is based on a sparse collocation grid with Smolyak level 7 and 163213 collocation points.

Fig. 7 shows the norms of the relative error of the variance of A_z. In contrast to the

former example with lognormal random variables in Section 6.1, the error of the stochastic collocation and Galerkin solution behaves decreases exponentially, except for the last result of the sparse collocation solution. The use of Legendre and Hermite polynomials to approximate the solution could explain this convergence rate due to the abovementioned relationship between these polynomials and the probability density function of uniform and Gaussian random variables. This example might indicate that the first example with lognormal random variables is not representative for the general convergence behavior of the discussed stochastic solution methods.

Secondly, we remark that the sparse stochastic collocation method achieves the most accurate solution for this example, although its convergence rate is not completely monotone. This latter effect could be a result of an insufficiently accurate reference solution, as also encountered in Remark 6.2.

7 Influence of uncertainty on the torque

In this section, we describe how uncertainty on the material parameters influences the torque statistics. The mean behavior of the torque corresponds in each case to the solution of the deterministic model. This section demonstrates how a small input uncertainty can lead to a large uncertainty of the corresponding torque and thus motivates the use of stochastic models for reliably designing machining tools composed of a ferromagnetic rotor. The torque is calculated as a function of the mechanical velocity, ω_m, in the range of 0 to 300 rad/s. First, we focus on modelling this uncertainty by only one random variable — either ξ_1 or ξ_2. Next, we consider the effect of uncertainty on the parameters of the magnetization curve, i.e., ξ_4 up to ξ_7.

7.1 Modeling uncertainty by one random variable

First, we express the uncertainty on the magnetic material parameters by one random variable, ξ_1. In the left column of Fig. 8, the mean, standard deviation and probability density function of the torque is illustrated for the case ξ_1 is uniformly distributed on $[1-0.08\sqrt{3},1+0.08\sqrt{3}]$. The probability density function corresponds to $\omega_m=111$ rad/s. The results for the stochastic collocation and stochastic Galerkin method are given, with the number of random unknowns indicated between brackets. The mean solution corresponds to the solution of the original deterministic problem. In case of only 4 random unknowns, the stochastic collocation and Galerkin results visually coincide. We observe that the variance on the torque — which corresponds to the square of the standard deviation, is substantially larger than the variance on the underlying magnetic potential A_z, see for example Fig. 4. This confirms that the torque is highly sensitive to uncertainty on the magnetic material parameters. Moreover, this effect becomes more important at high speeds.

In the right part of Fig. 8, similar results are shown for the case ξ_1 is modeled as a lognormal random variable, based on a Gaussian variable with mean $\mu_g=0$ and standard

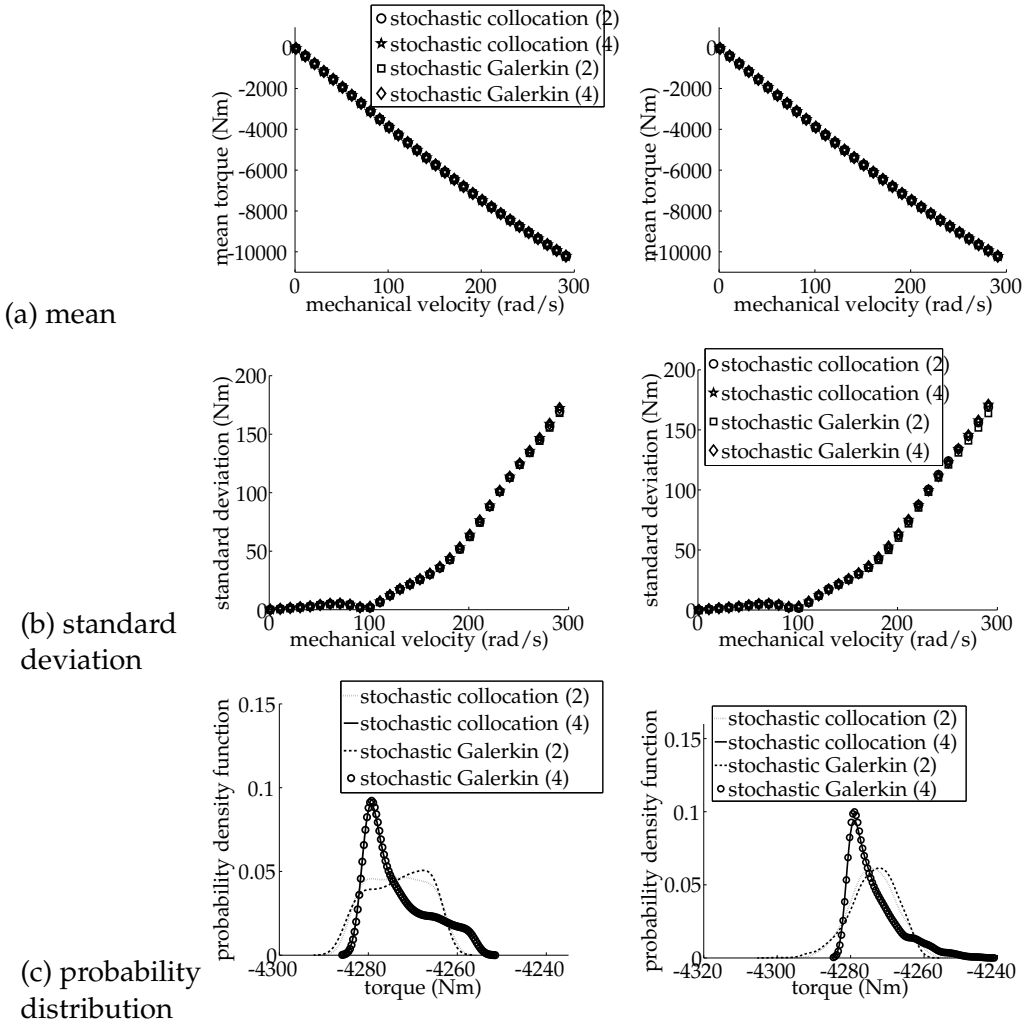
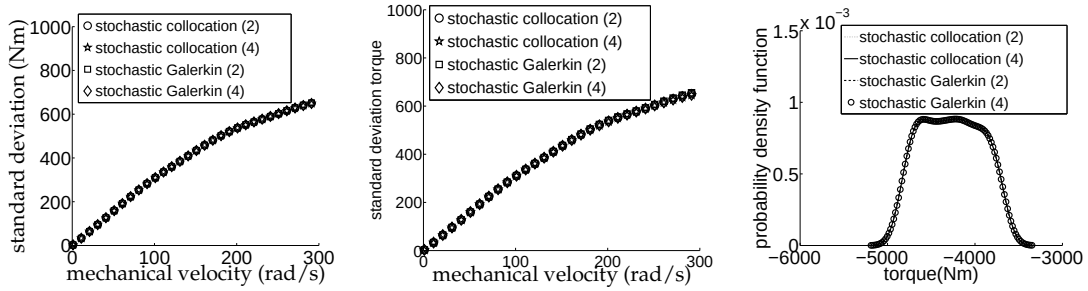


Figure 8: Statistics of the torque, in the case of one random variable ξ_1 in the stochastic model, either uniformly (left) or lognormally (right column) distributed.

deviation $\sigma_g = 0.08$. The lognormal case requires more stochastic unknowns, i.e., a larger number of stochastic collocation points or a high order polynomial chaos, to obtain an accurate solution in comparison to the uniformly distributed case.

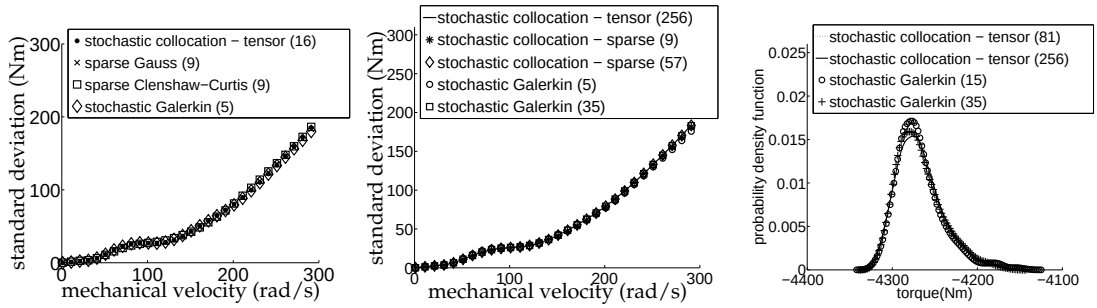
Fig. 8 shows that a similar standard deviation of the torque results from using the same input standard deviation both in the uniform case as well as for the underlying Gaussian variable in the lognormal case. This suggests that the variance of the torque is mainly determined by the input variances and not by the type of probability density function.

Similarly, we can consider the case of uncertainty on the conductivity, as expressed by ξ_2. Fig. 9 presents the results for ξ_2 either uniformly distributed on $[1-0.08\sqrt{3},1+$



(a) standard deviation - uniform (b) standard deviation - lognormal (c) pdf - uniform

Figure 9: Statistics of the torque, in the case that the stochastic model contains one uniformly or lognormally distributed random variable ξ_2.



(a) standard deviation - uniform (b) standard deviation - lognormal (c) pdf - lognormal

Figure 10: Statistics of the torque, in the case that the stochastic model contains 4 uniformly or lognormally distributed random variables: ξ_4, ξ_5, ξ_6 and ξ_7.

$0.08\sqrt{3}]$, or lognormally with $\mu_g=0$ and $\sigma_g=0.08$. The range of the uniformly distributed variable is chosen so that its standard deviation equals σ_g. The mean value of the torque is not shown, as it corresponds to Fig. 8(a). The uncertainty on the torque is even more influenced by ξ_2 then by ξ_1, as indicated by the larger standard deviation in Fig. 9 compared to Fig. 8. The standard deviation of the torque in the uniform and lognormal case are again vary similar by using the same input standard deviation σ_g.

7.2 Uncertainty on the parameters of the magnetization curve

Instead of considering only one random variable ξ_1, the uncertainty on the magnetization curve can be expressed by 4 random variables, ξ_4 up to ξ_7, each perturbing one of the parameters in (2.13). Since multiple random variables are present, both a tensor product as well as a sparse stochastic collocation point grids can be applied. Fig. 10 illustrates statistics of the torque. In the uniform case, all 4 random variables are uniformly distributed on $[1-0.08\sqrt{3},1+0.08\sqrt{3}]$; in the lognormal case they correspond to $\mu_g=0$ and $\sigma_g=0.08$. Between brackets, the number of random unknowns is given. The mean value

of the torque is not shown, as it corresponds to the solution of the deterministic problem (2.2). Comparing these results to Fig. 8, we note that modeling the uncertainty on the magnetic material by one random variable instead of the more accurate representation by 4 random variables, yields already a first approximation to the variability of the torque.

8 Conclusion

This paper considers a nonlinear stochastic model which represents a solid-rotor magnetic brake as a ferromagnetic cylinder rotating at high speed. High-order stochastic solutions have been obtained with both the stochastic Galerkin as well as the stochastic collocation approach. Both approaches were compared w.r.t. their computational cost, implementation cost and accuracy. For the considered stochastic model, the stochastic collocation approach scores better than the stochastic Galerkin method w.r.t. the implementation and computational cost. Concerning the implementation cost, the stochastic Galerkin method requires a re-implementation of the nonlinear solver routine whereas deterministic simulation code can be reused in the stochastic collocation case. Moreover, convergence of the Newton iterations can become problematic for the stochastic Galerkin method. This problem occurs particularly in case of a high-order polynomial chaos and requires good initial guesses. Concerning the computational cost, numerical experiments illustrate that more computational time is required by the stochastic Galerkin approach to reach the same level of accuracy as the stochastic collocation method.

We note that the stochastic Galerkin method can achieve a more accurate solution than the stochastic collocation method for the same number of unknowns. Further research to improve its computational cost can make the use of the stochastic Galerkin method more interesting compared to the stochastic collocation method. The large computational cost of the stochastic Galerkin method is caused by the expensive solution of the large linearized systems. Due to the spectral discretization in space, these systems are almost dense and difficult to solve. For other spatial discretizations, for example a finite element discretization, it may well be possible to create specialized fast solvers for the linearized systems, for example based on multigrid techniques.

In order to further improve the stochastic Galerkin method, an alternative basis could be used for the polynomial chaos representation of the solution (4.1). An adapted polynomial basis might reduce the Newton convergence problems in case of a high-order polynomial chaos. For example, the convergence problems of time-dependent problems in case of a high-order polynomial chaos are partly resolved by the use of a multi-element polynomial chaos [32]. The efficiency of the chaos expansions could be increased by taking the anisotropy of the problem into account: not all random variables require the same order of polynomials. The anisotropy can also be used in the stochastic collocation method when constructing multidimensional collocation points [23].

Numerical experiments revealed how uncertainty on the magnetic material parameters has a large influence on machine properties such as the torque, especially at high

speeds. With the described stochastic solvers, the randomness or uncertainty on other material properties, for example in the case of a different description of the magnetization curve, can be investigated in a similar way.

Acknowledgments

E. Rosseel is a research assistant of the Research Foundation – Flanders.

References

[1] I. Babuška, F. Nobile, and R. Tempone. A stochastic collocation method for elliptic partial differential equations with random input data. SIAM J. Numer. Anal., 45:1005–1034, 2007.
[2] I. Babuška, R. Tempone, and G. Zouraris. Galerkin finite element approximations of stochastic elliptic partial differential equations. SIAM J. Numer. Anal., 42:800–825, 2004.
[3] I. Babuška, R. Tempone, and G. E. Zouraris. Solving elliptic boundary value problems with uncertain coefficients by the finite element method: the stochastic formulation. Comput. Methods Appl. Mech. Engrg., 194:1251–1294, 2005.
[4] V. Barthelmann, E. Novak, and K. Ritter. High dimensional polynomial interpolation on sparse grids. Advances in Computational Mathematics, 12:273–288, 2000.
[5] M. Bieri and C. Schwab. Sparse high order FEM for elliptic sPDEs. Computer Methods in Applied Mechanics and Engineering, 198(13-14):1149–1170, 2009.
[6] C. Canuto, A. Quarteroni, M. Y. Hussaini, and T. A. Zang. Spectral Methods: Fundamentals in Single Domains. Springer, Berlin, Germany, 2006.
[7] H. De Gersem. Spectral-element methods for high-speed rotating cylinders. COMPEL: the international journal for computation and mathematics in electrical and electronic engineering, 28(3):730–740, 2009.
[8] H. De Gersem, I. Munteanu, and T. Weiland. Construction of differential material matrices for the orthogonal finite-integration technique with nonlinear materials. IEEE Transactions on Magnetics, 44(6):710–713, 2008.
[9] M. Eldred and J. Burkardt. Comparison of non-intrusive polynomial chaos and stochastic collocation methods for uncertainty quantification. AIAA, (2009-0976), 2009.
[10] O. G. Ernst and E. Ullmann. Stochastic Galerkin matrices. SIAM J. Matrix Anal. Appl., 31(4):1848–1872, 2010.
[11] G. Fishman. Monte Carlo: Concepts, Algorithms and Applications. Springer-Verlag, New York, USA, 1996.
[12] T. Gerstner and M. Griebel. Numerical integration using sparse grids. Numer. Algorithms, 18:209–232, 1998.
[13] R. Ghanem and J. Red-Horse. Propagation of probalistic uncertainty in complex physical systems using a stochastic finite element approach. Physica D, 133:137–144, 1999.
[14] R. Ghanem and P. Spanos. Stochastic Finite Elements, a Spectral Approach. Dover, Mineola, New York, 2nd edition, 2003.
[15] A. Keese. Numerical Solution of Systems with Stochastic Uncertainties. PhD Thesis, Technische Universität Braunschweig, Braunschweig, Germany, 2004.
[16] O. Le Maître, O. Knio, B. Debusschere, H. Najm, and R. Ghanem. A multigrid solver for two-dimensional stochastic diffusion equations. Comput. Methods Appl. Mech. Engrg., 192:4723–4744, 2003.

[17] O. Le Maître, H. Najm, R. Ghanem, and O. Knio. Multi-resolution analysis of wiener-type uncertainty propagation schemes. Journal of Computational Physics, 197:502–531, 2004.
[18] G. Loeven, J. Witteveen, and H. Bijl. Probabilistic collocation: an efficient non-intrusive approach for arbitrary distributed parametric uncertainties. In AIAA, Aerospace sciences meeting and exhibit, 2007, Reno, Nevada, number AIAA 2007-317, 2007.
[19] D. Lucor and G. Karniadakis. Adaptive generalized polynomial chaos for nonlinear random oscillations. SIAM J. Sci. Comput., 26(2):720–735, 2005.
[20] L. Mathelin, M. Y. Hussaini, and T. A. Zang. Stochastic approaches to uncertainty quantification in CFD simulations. Numerical Algorithms, 38:209–236, 2005.
[21] H. Matthies and A. Keese. Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations. Comput. Methods Appl. Engrg., 194:1295–1331, 2005.
[22] H. G. Matthies. Stochastic finite elements: computational approaches to stochastic partial differential equations. Z. Angew. Math. Mech., 88(11):849–873, 2008.
[23] F. Nobile, R. Tempone, and C. Webster. An anisotropic sparse grid stochastic collocation method for elliptic partial differential equations with random input data. SIAM J. Num. Anal., 46(5):2411–2442, 2008.
[24] F. Nobile, R. Tempone, and C. Webster. A sparse grid stochastic collocation method for partial differential equations with random input data. SIAM J. Num. Anal., 46:2309–2345, 2008.
[25] C. Powell and H. Elman. Block-diagonal preconditioning for spectral stochastic finite element systems. IMA J. Numer Anal., 29:350–375, 2009.
[26] E. Rosseel, T. Boonen, and S. Vandewalle. Algebraic multigrid for stationary and time-dependent partial differential equations with stochastic coefficients. Numer. Linear Algebra Appl., 15:141–163, 2008.
[27] E. Rosseel and S. Vandewalle. Iterative solvers for the stochastic finite element method. SIAM J. Sci. Comput., 32(1):372–397, 2010.
[28] B. Seynaeve, E. Rosseel, B. Nicolaï, and S. Vandewalle. Fourier mode analysis of multigrid methods for partial differential equations with random coefficients. J. Comput. Phys., 224:132–149, 2007.
[29] S. Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. Soviet Mathematics Dokl., 4:240–243, 1963.
[30] E. Ullmann. A Kronecker product preconditioner for stochastic Galerkin finite element discretizations. SIAM J. Sci. Comput., 32(2):923–946, 2010.
[31] X. Wan and G. E. Karniadakis. An adaptive multi-element generalized polynomial chaos method for stochastic differential equations. Journal of Computational Physics, 209:617–642, 2005.
[32] X. Wan and G. E. Karniadakis. Long-term behavior of polynomial chaos in stochastic flow simulations. Comput. Methods Appl. Mech. Engrg., 195(41–43):5582–5596, 2006.
[33] H. Woodson and J. Melcher. Electromechanical Dynamics, volume 2. John Wiley and Sons, New York, USA, 1968.
[34] D. Xiu. Efficient collocational approach for parametric uncertainty analysis. Communications in computational physics, 2(2):293–309, 2007.
[35] D. Xiu. Fast numerical methods for stochastic computations: a review. Communications in computational physics, 5(2-4):242–272, 2009.
[36] D. Xiu and J. S. Hesthaven. High-order collocation methods for differential equations with random inputs. SIAM J. Sci. Comput., 27(3):1118–1139, 2005.
[37] D. Xiu and G. Karniadakis. Modeling uncertainty in steady state diffusion problems via

generalized polynomial chaos. Comput. Methods Appl. Mech. Engrg., 191:4927–4948, 2002.
[38] D. Xiu and G. E. Karniadakis. The Wiener–Askey polynomial chaos for stochastic differential equations. SIAM J. Sci. Comput., 24(2):619–644, 2002.
[39] D. Xiu, D. Lucor, S.-H. Su, and G. Karniadakis. Stochastic modeling of flow structure interactions using generalized polynomial chaos. J. Fluids Eng., 124:51–59, 2002.
[40] T. A. Zang, Y. S. Wong, and M. Y. Hussaini. Spectral multigrid methods for elliptic equations II. Journal of Computational Physics, 54:489–507, 1984.