Numer. Math. Theor. Meth. Appl.
doi: 10.4208/nmtma.2011.m12si04

Vol. **5**, No. 1, pp. 62-84
February 2012

Newton-Multigrid for Biological Reaction-Diffusion Problems with Random Coefficients

Eveline Rosseel[1], Nico Scheerlinck[1] and Stefan Vandewalle[1,*]

[1] *Computer Science Department, Katholieke Universiteit Leuven,
Celestijnenlaan 200A, B-3001 Leuven, Belgium.*

Received 11 January 2011; Accepted (in revised version) 05 May 2011

Available 21 December 2011

Abstract. An algebraic Newton-multigrid method is proposed in order to efficiently solve systems of nonlinear reaction-diffusion problems with stochastic coefficients. These problems model the conversion of starch into sugars in growing apples. The stochastic system is first converted into a large coupled system of deterministic equations by applying a stochastic Galerkin finite element discretization. This method leads to high-order accurate stochastic solutions. A stable and high-order time discretization is obtained by applying a fully implicit Runge-Kutta method. After Newton linearization, a point-based algebraic multigrid solution method is applied. In order to decrease the computational cost, alternative multigrid preconditioners are presented. Numerical results demonstrate the convergence properties, robustness and efficiency of the proposed multigrid methods.

AMS subject classifications: 35K57, 35Q92, 65M55, 65N35

Key words: Multigrid, stochastic Galerkin finite element method, reaction-diffusion problems, implicit Runge-Kutta method and PDEs with random coefficients.

1. Introduction

Stochastic Galerkin finite element methods are being applied to a wide range of stochastic applications, e.g., to elasticity problems [10], heat transfer problems [21], in fluid-structure interactions [31] and in computational fluid mechanics [14]. In most cases, linear partial differential equations with random coefficients are considered. The extension to nonlinear problems is generally not straightforward for stochastic Galerkin methods [18]. The stochastic Galerkin projection may not be computed analytically, except for polynomial nonlinearities. Also, the stochastic Galerkin method transforms a stochastic problem into high-dimensional deterministic systems, for which special solvers need to be designed.

[*]Corresponding author. *Email addresses:* `eveline.rosseel@cs.kuleuven.be` (E. Rosseel), `nico.scheerlinck@cs.kuleuven.be` (N. Scheerlinck), `stefan.vandewalle@cs.kuleuven.be` (S. Vandewalle)

As an alternative to the stochastic Galerkin finite element method, so-called stochastic collocation methods were proposed [1, 17]. These methods enable a black-box reuse of deterministic simulation codes and do not require modifications for solving nonlinear stochastic problems. Although very good results can be obtained with the stochastic collocation method, its convergence rate is typically somewhat slower than the stochastic Galerkin convergence, in terms of the number of deterministic PDEs to be solved [7, 24]. In terms of computational cost, the success of the stochastic Galerkin method depends on the solution method for the high-dimensional deterministic Galerkin systems. For linear stochastic problems, efficient solvers can easily be designed [8,22,25]. This paper explores the construction of a multigrid solution approach for stochastic Galerkin discretizations of systems of nonlinear PDEs.

The stochastic Galerkin solution of a set of nonlinear, time-dependent reaction-diffusion equations is considered. We focus on a particular application, namely the conversion of starch into sugars in growing fruit. The accuracy of the simulation depends on the accuracy of a large set of parameters which model the chemical composition and shape of the fruit under consideration. Many of these parameters are inherently variable; hence a stochastic simulation is needed.

In the case of linear, time-dependent partial differential equations (PDE) with random coefficients, efficient multigrid methods exist for the systems resulting from a stochastic Galerkin finite element discretization combined with an implicit Runge-Kutta (IRK) time discretization [23]. The implicit Runge-Kutta method guarantees a high-order and stable time discretization and enables one to take larger time steps than with explicit time discretization schemes. In this paper we investigate whether such multigrid approaches can be extended to systems of nonlinear, stochastic PDEs.

This paper is structured as follows. In Section 2, the model equations are presented. Section 3 details the discretization of the set of nonlinear, stochastic PDEs. Section 4 proposes an algebraic multigrid (AMG) method to solve the high-dimensional discretized systems efficiently. Some implementation issues are addressed in Section 5. The biological application that motivated this research is detailed in Section 6. The properties of the AMG method are demonstrated by numerical experiments in Section 7. Section 8 summarizes the main conclusions of this paper.

2. Model description

The stochastic reaction-diffusion problem that we consider in this paper, can be described by the following set of equations:

$$\left\{\begin{aligned} \partial_t u_1(x,t,\omega) &= \nabla\cdot(a_1(x,\omega)\nabla u_1(x,t,\omega) + R_1(u_1(x,t,\omega),u_2(x,t,\omega)),\\ \partial_t u_2(x,t,\omega) &= \nabla\cdot(a_2(x,\omega)\nabla u_2(x,t,\omega) + R_2(u_1(x,t,\omega),u_2(x,t,\omega)),\end{aligned}\right. \tag{2.1}$$

$$\text{in } D\times[0,T_f]\times\Omega,$$

$$\vec{n}\cdot\nabla u_1(x,t,\omega) = 0 \quad\text{and}\quad \vec{n}\cdot\nabla u_2(x,t,\omega) = 0 \qquad \text{in } \partial D\times[0,T_f]\times\Omega.$$

The symbol ∂_t corresponds to differentiation with respect to the time variable t. The concentrations of two different types of chemical particles are represented respectively by $u_1(x,t,\omega)$ and $u_2(x,t,\omega)$. Here, $x \in D$ denotes the spatial coordinates with $D \subset \mathbb{R}^d$ the d-dimensional spatial domain, $t \in [0, T_f]$ is the time variable and ω represents the stochastic nature of the solutions, i.e., $\omega \in \Omega$ with Ω a sample space. Together with a σ-algebra $\mathscr{F}$ and a probability measure $\mathscr{E}$, a complete probability space $(\Omega, \mathscr{F}, \mathscr{E})$ is defined. The diffusion coefficients a_1 and a_2 are modelled as random fields. A random field is defined by a mapping $D \times \Omega \to \mathbb{R}$; each sample of a random field corresponds to a deterministic function in space. In order to guarantee the existence and uniqueness of the solution to (2.1), we assume that the diffusion coefficients a_1 and a_2 are bounded and uniformly coercive, see [21]. The reaction terms R_1 and R_2 are typically nonlinear functions of u_1 and u_2. The model is completed with appropriate initial conditions, which can be stochastic as well. Zero Neumann boundary conditions are applied in order to model the fact that chemical particles cannot leave the volume boundary ∂D.

2.1. Reaction model

The *Gray-Scott* model [20] represents the following cubic auto-catalytic chemical reaction between chemicals U and V:

$$\begin{aligned} U + 2V &\to 3V, \\ V &\to W, \end{aligned}$$

where W is an inert product. These two reactions occur at different rates throughout the volume according to the relative concentrations at each point. The reaction kinetics R_1 and R_2 respectively correspond to

$$R_1(u_1,u_2) = -\kappa_A u_1 u_2^2 + \kappa_f(1-u_1), \tag{2.2}$$

$$R_2(u_1,u_2) = \kappa_A u_1 u_2^2 - \kappa_2 u_2, \tag{2.3}$$

where κ_f denotes the feed rate of chemical U, κ_2 is the reaction rate of the second reaction and κ_A a dimensionless rate constant corresponding to the first reaction. The last term of Eq. (2.2) expresses that the chemical U is replenished at a rate proportional to the difference between its current value and its maximum value, which is equal to 1.

2.2. Finite-dimensional noise

We assume that the random fields a_1, a_2 and the random rate parameters in R_1, R_2 depend on a finite number of random variables. A random variable ξ_i is a function of elements of a sample space, i.e., $\xi_i : \Omega \to \mathbb{R}$, and is characterized by a probability density function $\varrho(y_i)$ defined on Γ_i, with Γ_i the image of ξ_i. Based on the Doob-Dynkin lemma [2], the above assumption yields that the solutions u_1 and u_2 can be expressed as a function of ξ, $u_1(x,t,\omega) = u_1(x,t,\xi)$ and $u_2(x,t,\omega) = u_2(x,t,\xi)$. Here, ξ denotes a random vector collecting all the random variables present in the problem description. We will assume that ξ consists of L independent random variables ξ_i, $\xi = (\xi_1, \cdots, \xi_L)$ and has a joint probability density function $\varrho(y) = \prod_{i=1}^{L} \varrho_i(y_i)$ for $y \in \Gamma = \prod_{i=1}^{L} \Gamma_i$. This enables one to rewrite (2.1) as a parametric deterministic problem in $D \times [0, T_f] \times \Gamma$:

$$\partial_t u_j(x,t,y) = \nabla \cdot (a_j(x,y) \nabla u_j(x,t,y) + R_j(u_1(x,t,y), u_2(x,t,y)) \qquad (j = 1,2). \quad (2.4)$$

2.3. Growing spatial domain

Since we want to simulate ripening and maturation processes of growing fruit, the spatial domain D is time-dependent: $D(t)$. A *uniform* domain growth will be considered. That means that any two points of the volume move apart with a relative velocity that depends only on the separation and is independent of the spatial position, i.e., which is a function of time only. As a result, the evolution of the boundary $\partial D(t)$ can be described by $r_0 r(t)$, with r_0 the initial radius of the domain and $r(t)$ a growth function. In order to describe the reaction-diffusion problem (2.4) in a fixed domain $\overline{D}$, a reference coordinate system for the growing domain is chosen. The origin of the coordinate system corresponds to a reference point which remains fixed while the domain grows. For the apple domain that will be described in Section 6, this reference domain coincides with the center of the apple. The coordinates $(\overline{x}, t, y)$ of the fixed domain correspond to the scaling,

$$(x,t,y) \to (\overline{x},t,y) = \left(\frac{x}{r_0 r(t)}, t, y \right).$$

In addition to this coordinate transformation, extra dilution terms are introduced in (2.4) in order to model the decrease of the local concentration as the containing volume increases by uniform domain growth. Denote by $\widehat{r}(t) = 1/(r_0^2 r^2(t))$ and $\widetilde{r}(t) = (3\partial_t\, r(t))/r(t)$, the transformed reaction-diffusion equations are then given by

$$\begin{aligned} \partial_t u_j(\overline{x},t,y) =& \widehat{r}(t) \nabla \cdot \left(a_j(\overline{x},y) \nabla u_j(\overline{x},t,y) \right) + R_j(u_1(\overline{x},t,y), u_2(\overline{x},t,y)) \\ & - \widetilde{r}(t) u_j(\overline{x},t,y) \qquad \text{for } j = 1,2 \quad \text{in } \overline{D} \times [0,T_f] \times \Gamma. \end{aligned} \quad (2.5)$$

3. Overview of discretization

The system of nonlinear stochastic PDEs (2.5) can be linearized first, i.e., by applying Newton's method, after which a stochastic Galerkin discretization of the Jacobian can be constructed. Alternatively, constructing the Jacobian can be avoided by applying a quasi-Newton strategy, see, e.g., [19]. We will adopt the former approach since a Jacobian can easily be constructed for polynomial nonlinearities as in (2.2)–(2.3).

3.1. Newton linearization

Newton's method starts from an initial guess $u_j^{(0)}$ and iteratively updates this approximation by solving a linearized problem in every Newton step. We construct a linearized model for (2.5) by starting from the weak formulation. That is, find $u_j \in L_2(0,T_f;H^1(\overline{D}))\otimes L_2(\Gamma)$ such that $u_j = u_{j,\text{init}}$ and

$$\int_{\Gamma\times D} \partial_t u_j\, v_j \varrho\, d\overline{x} dy + \widehat{r}(t)\int_{\Gamma\times D} a_j \nabla u_j \nabla v_j \rho\, d\overline{x} dy - \int_{\Gamma\times D} R_j(u_1,u_2) v_j \rho\, d\overline{x} dy \\ + \widetilde{r}(t)\int_{\Gamma\times D} u_j v_j\, \rho\, d\overline{x} dy = 0 \qquad \forall v_j \in H_1^0(\overline{D}) \otimes L_2(\Gamma), \quad (j=1,2). \tag{3.1}$$

The Hilbert space $L_2(0,T_f;H^1(\overline{D}))$, with $[0,T_f]$ the time interval on which (2.5) holds, is defined in [21]. The Lebesgue space $L_2(\Gamma)$ contains all square-integrable functions on Γ. The left-hand side of (3.1) is called the *nonlinear residual* associated with the weak formulation and is denoted as $F_j(u_1,u_2)$ $(j=1,2)$. For $u_j = u_j^{(k)} + \delta_j^{(k)}$ in (3.1), we have that the corrections $\delta_j^{(k)} \in L_2(0,T_f;H_0^1(\overline{D})) \otimes L_2(\Gamma)$ satisfy

$$\int_{\Gamma\times D} \partial_t \delta_j^{(k)} v_j\, \varrho\, d\overline{x} dy + \widehat{r}(t)\int_{\Gamma\times D} a_j \nabla \delta_j^{(k)} \nabla v_j \varrho\, d\overline{x} dy + \widetilde{r}(t)\int_{\Gamma\times D} \delta_j^{(k)} v_j\, \varrho\, d\overline{x} dy \\ - \int_{\Gamma\times D} \left(R_j\left(u_1^{(k)} + \delta_1^{(k)}, u_2^{(k)} + \delta_2^{(k)}\right) - R_j\left(u_1^{(k)}, u_2^{(k)}\right)\right) v_j\, \varrho\, d\overline{x} dy \\ = -F_j\left(u_1^{(k)}, u_2^{(k)}\right) \qquad \forall v_j \in H_0^1(\overline{D}) \otimes L_2(\Gamma), \quad (j=1,2). \tag{3.2}$$

Based on the Gray-Scott reaction (2.2)–(2.3), we have that

$$\begin{aligned} & R_1\left(u_1^{(k)} + \delta_1^{(k)}, u_2^{(k)} + \delta_2^{(k)}\right) - R_1\left(u_1^{(k)}, u_2^{(k)}\right) \\ = & -\kappa_A \left(u_1^{(k)} + \delta_1^{(k)}\right)\left(u_2^{(k)} + \delta_2^{(k)}\right)^2 + \kappa_A u_1^{(k)}\left(u_2^{(k)}\right)^2 \\ & + \kappa_f\left(1 - (u_1^{(k)} + \delta_1^{(k)})\right) - \kappa_f\left(1 - u_1^{(k)}\right) \\ = & -\kappa_A \Big(2u_1^{(k)} u_2^{(k)} \delta_2^{(k)} + u_1^{(k)}\left(\delta_2^{(k)}\right)^2 + \delta_1^{(k)}\left(u_2^{(k)}\right)^2 \\ & +2\delta_1^{(k)} u_2^{(k)} \delta_2^{(k)} + \delta_1^{(k)}\left(\delta_2^{(k)}\right)^2\Big) - \kappa_f \delta_1^{(k)}, \end{aligned}$$

and similarly for R_2. Following the approach in [6, p. 325], we obtain the linearized problem by dropping the high-order terms in $\delta_1^{(k)}$ and $\delta_2^{(k)}$ in (3.2). This yields: find $\delta_1, \delta_2 \in L_2(0,T_f;H_0^1(\overline{D})) \otimes L_2(\Gamma)$ such that $\forall v_1, v_2 \in H_0^1(\overline{D}) \otimes L_2(\Gamma)$,

$$\int_{\Gamma\times D} \partial_t \delta_1^{(k)} v_1\, \varrho\, d\overline{x} dy + \widehat{r}(t)\int_{\Gamma\times D} a_1 \nabla \delta_1^{(k)} \nabla v_1 \varrho\, d\overline{x} dy + \left(\widetilde{r}(t) + \kappa_f\right)\int_{\Gamma\times D} \delta_1^{(k)} v_1\, \varrho\, d\overline{x} dy \\ + \kappa_A \int_{\Gamma\times D} \left(2u_1^{(k)} u_2^{(k)} \delta_2^{(k)} + \delta_1^{(k)}\left(u_2^{(k)}\right)^2\right) v_1\, \varrho\, d\overline{x} dy = -F_1\left(u_1^{(k)}, u_2^{(k)}\right), \tag{3.3}$$

$$\int_{\Gamma\times D} \partial_t \delta_2^{(k)} v_2\, \varrho d\overline{x} dy + \widehat{r}(t) \int_{\Gamma\times D} a_2 \nabla \delta_2^{(k)} \nabla v_2 \varrho d\overline{x} dy + \left(\widetilde{r}(t) + \kappa_2\right) \int_{\Gamma\times D} \delta_2^{(k)} v_2\, \varrho d\overline{x} dy$$
$$- \kappa_A \int_{\Gamma\times D} \left(2u_1^{(k)} u_2^{(k)} \delta_2^{(k)} + \delta_1^{(k)} \left(u_2^{(k)}\right)^2\right) v_2\, \varrho d\overline{x} dy = -F_2\left(u_1^{(k)}, u_2^{(k)}\right). \quad (3.4)$$

The solution to (3.3)–(3.4) is called the *Newton correction*. At Newton step $k+1$, the current approximation $u_j^{(k)}$ is updated as $u_j^{(k+1)} = u_j^{(k)} + \delta_j^{(k)}$, for $j = 1,2$. The remainder of this section describes the spatial, stochastic and time discretization of (3.3)–(3.4) which will lead to a large algebraic system. The multigrid solution of the algebraic, linearized equations is then discussed in Section 4.

3.2. Stochastic Galerkin finite element discretization

By introducing finite-dimensional subspaces $V_h \subset H^1$, $V_{h,0} \subset H_0^1$ and $W_p \subset L_2(\Gamma)$, a stochastic Galerkin finite element discretization of (3.3)–(3.4) leads to an equation similar to (3.3)–(3.4), with $\delta_1^{(k)}$, $\delta_2^{(k)}$ replaced by respectively $\delta_{1,hp}^{(k)}, \delta_{2,hp}^{(k)} \in L_2(0, T_f; V_h(\overline{D})) \otimes W_p$ and v_1, v_2 replaced by $v_{1,hp}, v_{2,hp} \in V_{h,0} \times W_p$.

We will use a spatial finite element discretization defined on a conforming triangulation of the domain $\overline{D}$. The finite-dimensional subspace $V_{h,0}$ is defined as the span $\{s_1(\overline{x}), \ldots, s_N(\overline{x})\}$, where s_n is a nodal basis function and N the total number of spatial degrees of freedom. Since no Dirichlet boundary is present, the subspaces V_h and $V_{h,0}$ coincide. The stochastic discretization applies a set of orthonormal, multi-dimensional polynomials $\Psi_q(y)$. The polynomials are chosen to be orthogonal with respect to the probability density function $\varrho(y)$ of the random variables present in the problem. This guarantees a fast convergence of the stochastic discretization error [21]. For example, in the case of a Gaussian distribution, normalized Hermite polynomials will be applied; in the case of a uniform distribution, Legendre polynomials. The subspace W_P is then defined as $W_P = \text{span}\{\Psi_1(y), \ldots, \Psi_Q(y)\}$. The polynomials $\Psi = [\Psi_1, \ldots, \Psi_Q]$ form a finite-dimensional generalized polynomial chaos basis [30]. The number of degrees of freedom Q depends on the number of random variables L and on the order P of the polynomials. A complete polynomial basis consists of $Q = (L+P)!/(L!P!)$ polynomials. The unknowns $\delta_{1,hp}^{(k)}$ and $\delta_{2,hp}^{(k)}$ are then given for $j = 1,2$ by

$$\delta_{j,hp}^{(k)}(\overline{x}, t, y) = \sum_{n=1}^{N} \sum_{q=1}^{Q} \delta_{j,n,q}^{(k)}(t) s_n(\overline{x}) \Psi_q(y). \quad (3.5)$$

Remark 3.1. Instead of a global polynomial subspace W_p, also local polynomial basis functions can be considered, e.g., [15, 29]. These so-called multi-element stochastic approximations are particularly useful for approximating discontinuities in probability space which may result from the presence of bifurcations. In this paper, the use of global polynomial approximations is justified since we consider a range of uncertain input parameters so that no bifurcations are present.

In order to rewrite the finite-dimensional formulation of (3.3)–(3.4) in matrix notation, we will first reformulate the random fields a_1 and a_2 so that the spatial and stochastic part can easily be separated, i.e.,

$$a_j(\overline{x}, y) = \sum_{i=1}^{S_j} a_{j,i}(\overline{x})\varphi_{j,i}(y) \qquad (j = 1, 2). \tag{3.6}$$

For example, when the random field a_j has the form of a truncated Karhunen-Loève expansion [16], $S_j = \hat{L} + 1$, $\varphi_{j,i}(y) = y_{i-1}$ and $y_0 = 1$. When a_j is represented by an $\hat{L}$-dimensional generalized polynomial expansion [11,30] of order $\hat{P}$, $S_j = (\hat{L} + \hat{P})!/(\hat{L}!\hat{P}!)$ and $\varphi_{j,i}(y) = \Psi_i(y)$. Based on (3.6), a set of S_j stiffness matrices $K_{j,i} \in \mathbb{R}^{N\times N}$ can be constructed whose elements are defined as

$$K_{j,i}(m,n) := \int_{\overline{D}} a_{j,i}(\overline{x}) \nabla s_m(\overline{x}) \nabla s_n(\overline{x}) d\overline{x},$$

for $m, n = 1, \dots, N$; $j = 1, 2$ and $i = 1, \dots, S_j$. The stochastic part of (3.6) leads to a set of matrices $C_i \in \mathbb{R}^{Q\times Q}$, each defined as

$$\begin{aligned} C_i &:= \langle \varphi_{1,i} \Psi \Psi^T \rangle, & i &= 1, \cdots, S_1, \\ C_i &:= \langle \varphi_{2,i-S_1+1} \Psi \Psi^T \rangle, & i &= S_1 + 1, \cdots, S_1 + S_2 - 1, \end{aligned}$$

with $\Psi = [\Psi_1 \dots \Psi_Q]^T$. The notation $\langle \cdot \rangle$ corresponds to the integral $\int_\Gamma \cdot \varrho dy$. We assume that $\varphi_{1,1} \equiv \varphi_{2,1} \equiv 1$, since the first term in (3.6) is typically the mean of a_j. Matrix $C_1 := \langle \Psi \Psi^T \rangle \in \mathbb{R}^{Q\times Q}$ corresponds then to an identity matrix, due to the orthonormality of the polynomials Ψ_q.

The linearized reaction terms in (3.3)–(3.4) involve products of random fields. These products define the random fields $w_{hp}^{(k)} := u_{1,hp}^{(k)} u_{2,hp}^{(k)}$ and $z_{hp}^{(k)} := (u_{2,hp}^{(k)})^2$, which can be represented by a generalized polynomial expansion [30] as follows:

$$w_{hp}^{(k)}(\overline{x}, t, y) := u_{1,hp}^{(k)}(\overline{x}, t, y) u_{2,hp}^{(k)}(\overline{x}, t, y) = \sum_{i=1}^{\infty} \hat{w}_i^{(k)}(\overline{x}, t) \Psi_i(y), \tag{3.7}$$

$$z_{hp}^{(k)}(\overline{x}, t, y) := \left(u_{2,hp}^{(k)}(\overline{x}, t, y) \right)^2 = \sum_{i=1}^{\infty} \hat{z}_i^{(k)}(\overline{x}, t) \Psi_i(y). \tag{3.8}$$

The functions $\hat{w}_i^{(k)}$ and $\hat{z}_i^{(k)}$ are respectively defined as

$$\hat{w}_i^{(k)}(\overline{x}, t) := \sum_{c=1}^{Q} \sum_{q=1}^{Q} \hat{u}_{1,c}^{(k)}(\overline{x}, t) \hat{u}_{2,q}^{(k)}(\overline{x}, t) \langle \Psi_c \Psi_q \Psi_i \rangle / \langle \Psi_i^2 \rangle, \tag{3.9}$$

$$\hat{z}_i^{(k)}(\overline{x}, t) := \sum_{c=1}^{Q} \sum_{q=1}^{Q} \hat{u}_{2,c}^{(k)}(\overline{x}, t) \hat{u}_{2,q}^{(k)}(\overline{x}, t) \langle \Psi_c \Psi_q \Psi_i \rangle / \langle \Psi_i^2 \rangle, \tag{3.10}$$

where $\hat{u}_{j,q}^{(k)}$ ($j = 1,2$ and $q = 1,\ldots,Q$) correspond to the polynomial chaos coefficient functions of $u_{j,hp}^{(k)}$. The orthonormality of the polynomials Ψ_i gives that $\langle\Psi_i^2\rangle \equiv 1$ and that the summation in (3.7)–(3.8) can be truncated after $Z = (L+2P)!/(L!(2P)!)$ terms, with P the polynomial order used to construct to polynomial basis of W_P. Based on (3.7)–(3.8), the matrix formulation of the linearized reaction terms results in a set of Z time-dependent reaction matrices $J_i^{(k)}$ and $G_i^{(k)}$, whose elements respectively equal

$$J_i^{(k)}(t)(m,n) := \int_{\overline{D}} \hat{w}_i^{(k)}(\overline{x},t)s_m(\overline{x})s_n(\overline{x})d\overline{x}, \tag{3.11}$$

$$G_i^{(k)}(t)(m,n) := \int_{\overline{D}} \hat{z}_i^{(k)}(\overline{x},t)s_m(\overline{x})s_n(\overline{x})d\overline{x}, \qquad m,n = 1,\cdots,N,\ i = 1,\cdots,Z. \tag{3.12}$$

The corresponding stochastic discretization matrices $\underline{C}_i \in \mathbb{R}^{Q\times Q}$ are defined as

$$\underline{C}_i := \langle\Psi_i\Psi\Psi^T\rangle \qquad i = 1,\cdots,Z. \tag{3.13}$$

Combining the above results, we can rewrite the finite-dimensional formulation of (3.3)–(3.4) in matrix formulation:

$$\left((I_2\otimes C_1\otimes M)\partial_t + \widetilde{r}(t)\mathscr{K} + \widetilde{R}(t)\otimes C_1\otimes M + \kappa_A\sum_{i=1}^{Z}\mathscr{G}^{(k)}(t)\right)\boldsymbol{\delta}^{(k)}(t) = \mathbf{f}(t), \tag{3.14}$$

with

$$\widetilde{R}(t) := \begin{bmatrix} \widetilde{r}(t)+\kappa_f & 0 \\ 0 & \widetilde{r}(t)+\kappa_2 \end{bmatrix},$$

$I_2 \in \mathbb{R}^{2\times2}$ an identity matrix, matrix $\mathscr{K}$ defined as

$$\mathscr{K} := \begin{bmatrix} \sum_{i=1}^{S_1} C_1\otimes K_{1,i} & \mathbf{0}_{NQ} \\ \mathbf{0}_{NQ} & C_1\otimes K_{2,1} + \sum_{i=2}^{S_2} C_{i+S_1-1}\otimes K_{2,i} \end{bmatrix} \in \mathbb{R}^{2NQ\times 2NQ},$$

$\mathbf{0}_{NQ} \in \mathbb{R}^{NQ\times NQ}$ an all-zero matrix and the time-dependent matrix $\mathscr{G}(t)$ given by

$$\mathscr{G}^{(k)}(t) := \begin{bmatrix} \underline{C}_i\otimes G_i^{(k)}(t) & \underline{C}_i\otimes 2J_i^{(k)}(t) \\ -\underline{C}_i\otimes G_i^{(k)}(t) & -\underline{C}_i\otimes 2J_i^{(k)}(t) \end{bmatrix}. \tag{3.15}$$

The time-dependent vector $\boldsymbol{\delta}^{(k)}(t)$ collects the $2NQ$ functions $\delta_{j,n,q}^{(k)}(t)$ in (3.5), sorted first per type of chemical particle, then per stochastic basis function and finally per spatial finite element node. The elements of the mass matrix $M \in \mathbb{R}^{N\times N}$ equal

$$M(m,n) = \int_{\overline{D}} s_m(\overline{x})s_n(\overline{x})d\overline{x} \qquad m,n = 1,\cdots,N.$$

The right-hand side $\mathbf{f}(t)$ in (3.14) is a vector of $2NQ$ time-dependent functions corresponding to the right-hand side in (3.3)–(3.4). In order to evaluate $F_1(u_{1,hp}^{(k)},u_{2,hp}^{(k)})$ and $F_2(u_{1,hp}^{(k)},u_{2,hp}^{(k)})$ in (3.3)–(3.4), the matrices J_i and G_i from (3.11)–(3.12) can be reused.

3.3. Time discretization

A stable and high-order time discretization is achieved by applying a suitably chosen implicit Runge-Kutta (IRK) method [3] to the system of ordinary differential equations given by (3.14). To introduce some notation, consider a system of ODEs of the form

$$\frac{du}{dt} = f(t,u), \qquad \text{with } u(t_0) = u_0 \in \mathbb{R}^N.$$

An IRK method computes an approximation $u_{\mu+1}$ to the solution $u(t_{\mu+1})$ at time $t_{\mu+1}$ from an approximation u_μ at time t_μ. To this end, it introduces a number of auxiliary variables x_j, $j = 1, \cdots, s_{\text{irk}}$, called stage values or stage vectors, at times $t_\mu + c_j \Delta t$ with time step $\Delta t = t_{\mu+1} - t_\mu$. The IRK procedure can be written in terms of the stage value increments $\Delta x_j := x_j - u_\mu$ as

$$u_{\mu+1} = u_\mu + [\Delta x_1 \cdots \Delta x_{s_{\text{irk}}}] A_{\text{irk}}^{-T} b_{\text{irk}}, \tag{3.16}$$

$$\Delta x_i = \Delta t \sum_{j=1}^{s_{\text{irk}}} a_{ij} f(t_\mu + c_j \Delta t, u_\mu + \Delta x_j), \qquad i = 1, \cdots, s_{\text{irk}}, \tag{3.17}$$

Eq. (3.16) expresses $u_{\mu+1}$ as an update to u_μ in terms of the stage value increments Δx_j, $j = 1, \cdots, s_{\text{irk}}$. Eq. (3.17) describes the system of equations to be solved to compute the stage value increments. An IRK method is fully characterized by the parameters $A_{\text{irk}} = [a_{ij}]_{i,j=1,\cdots,s_{\text{irk}}}$, $b_{\text{irk}} = [b_1 \cdots b_{s_{\text{irk}}}]^T$ and $c_{\text{irk}} = [c_1 \cdots c_{s_{\text{irk}}}]^T$. Some example IRK parameters are given in [12]. In the present paper we will apply the popular class of Radau IIA fully implicit Runge-Kutta methods. Note that a one-stage Radau IIA IRK method is equivalent to a backward or implicit Euler time discretization.

Applying an IRK time discretization to the system (3.14) introduces at every time t_μ, s_{irk} additional unknowns. Hence, at every time step μ and Newton step k, a system of $2NQs_{\text{irk}}$ unknowns results. We group the unknowns together in a long vector $\mathbf{x}$, where the stage vector increments of $\boldsymbol{\delta}^{(k)}(t_\mu)$ are numbered first along the type of chemical particle, then along the random dimension, next along the spatial dimension and finally according to the IRK stages. The system (3.17) for the stage vector increments $\mathbf{x}$ then becomes

$$\Bigg(I_2 \otimes C_1 \otimes M \otimes \left(I_{s_{\text{irk}}} + \Delta t \widetilde{A}_{\text{irk}} \right) + \Delta t \mathscr{K} \otimes \widehat{A}_{\text{irk}} + \Delta t \begin{bmatrix} \kappa_f & 0 \\ 0 & \kappa_2 \end{bmatrix} \otimes C_1 \otimes M \otimes A_{\text{irk}}$$
$$+ \kappa_A \Delta t \left[\sum_{i=1}^{Z} \mathscr{G}^{(k)(c_{s_1})} \otimes A_{\text{irk}}(:,1) \cdots \sum_{i=1}^{Z} \mathscr{G}^{(k)(c_{s_{\text{irk}}})} \otimes A_{\text{irk}}(:,s_{\text{irk}}) \right] P \Bigg) \mathbf{x} = \mathbf{b}. \tag{3.18}$$

In (3.18), the matrices $\widetilde{A}_{\text{irk}}, \widehat{A}_{\text{irk}} \in \mathbb{R}^{s_{\text{irk}} \times s_{\text{irk}}}$ are respectively defined as

$$\widetilde{A}_{\text{irk}} := \left[\widetilde{r}(t_\mu + c_1 \Delta t) A_{\text{irk}}(:,1) \cdots \widetilde{r}(t_\mu + c_{s_{\text{irk}}} \Delta t) A_{\text{irk}}(:,s_{\text{irk}}) \right],$$
$$\widehat{A}_{\text{irk}} := \left[\widehat{r}(t_\mu + c_1 \Delta t) A_{\text{irk}}(:,1) \cdots \widehat{r}(t_\mu + c_{s_{\text{irk}}} \Delta t) A_{\text{irk}}(:,s_{\text{irk}}) \right].$$

The vector $\mathbf{b} \in \mathbb{R}^{2NQs_{\text{irk}}}$ denotes the right-hand side. The matrix $P \in \mathbb{R}^{2NQs_{\text{irk}} \times 2NQs_{\text{irk}}}$ permutes the columns of the matrix that it is multiplied with, in such a way that consecutive IRK stages are grouped together in blocks of s_{irk} columns. The notation $A_{\text{irk}}(:,j)$ is used to select the jth column of A_{irk}. Matrix $\mathscr{G}^{(k)(c_{s_j})} \in \mathbb{R}^{2NQ \times 2NQ}$ is defined by (3.15), where the time-dependent matrices are evaluated at $t = t_\mu + c_j \Delta t$ for stage vector increment $\mathbf{x}_{i,j}$, corresponding to random unknown i and IRK stage j. For example, the elements of matrix $J_i^{(k)(c_j)} \in \mathbb{R}^{N \times N}$ correspond to

$$J_i^{(k)(c_j)}(m,n) := \int_{\overline{D}} \hat{w}_i^{(k)(c_j)}(\overline{x}) s_m(\overline{x}) s_n(\overline{x}) d\overline{x}, \qquad m,n = 1,\cdots,N;\ i = 1,\cdots,Z,$$

$$\hat{w}_i^{(k)(c_j)}(\overline{x}) := \sum_{c=1}^{Q} \sum_{q=1}^{Q} \left(\hat{u}_{1,c}^{(k)}(\overline{x},t) + \mathbf{x}_{i,j}(\overline{x}) \right) \left(\hat{u}_{2,q}^{(k)}(\overline{x},t) + \mathbf{x}_{i,j}(\overline{x}) \right) \langle \Psi_c \Psi_q \Psi_i \rangle. \tag{3.19}$$

In (3.19), the orthonormality of the polynomials Ψ_q is already taken into account. Matrix $G_i^{(k)(c_j)}$ is defined similarly using (3.12). The Newton update $\boldsymbol{\delta}^{(k)}$ at time $t_{\mu+1}$ and Newton step k is obtained by evaluating (3.16) based on the solution $\mathbf{x}$ of (3.18).

4. An algebraic multigrid solver

In every Newton iteration, a large algebraic system is to be solved. This system is typically ill conditioned due to a poor conditioning of the stiffness matrices $K_{j,i}$, $j = 1,2$ and $i = 1,\ldots,S_j$. Previous studies of stochastic parabolic problems [23] showed that efficient and robust multigrid algorithms can be constructed for algebraic systems resulting from a combined stochastic Galerkin finite element and implicit Runge-Kutta discretization. In this work, these multigrid algorithms are extended to discretized *systems* of stochastic PDEs, e.g., given by the high-dimensional algebraic system (3.18). Since we consider problems discretized on a highly unstructured spatial finite element grid, only the algebraic variant of multigrid (AMG) [27] is used. That is, prolongation and restriction operators are constructed during a setup phase [28].

Applying AMG to a system of PDEs typically requires important modifications to the construction of the multigrid hierarchy and operators in order to maintain the fast convergence of AMG for scalar PDEs [28]. One usually discerns three different strategies for extending AMG to systems of PDEs: the *variable*-based, *unknown*-based and *point*-based AMG approaches. Each strategy is based on a particular grouping of the unknowns. The former applies AMG for scalar PDEs unchanged to a discretized system of PDEs and typically is not very efficient. The unknown-based approach requires a weak coupling between the different unknown solution functions. The point-based approach requires that all unknowns functions are discretized on the same spatial grid.

The stochastic system of PDEs (2.5) contains two unknown functions, $u_1(\overline{x},t,y)$ and $u_2(\overline{x},t,y)$, discretized on the same finite element grid. We will consider therefore a point-based AMG solution method. Below an overview is given of the multigrid components required for the proposed multigrid solver.

4.1. Smoothing operator

We will consider a point-oriented relaxation method. This approach is known to be effective for deterministic reaction-diffusion models [4]. For the stochastic model (2.1), this corresponds to a $(2Qs_{\text{irk}} \times 2Qs_{\text{irk}})$-block Gauss-Seidel smoother. At every collective Gauss-Seidel iteration, N systems of size $(2Qs_{\text{irk}} \times 2Qs_{\text{irk}})$ need to be solved, one for each spatial node n. Since both A_{irk} and the sum $\sum_{i=1}^{Z} \underline{C}_i$ are dense matrices - the latter is proven in [13] - these block systems are entirely dense.

4.2. Multigrid hierarchy and transfer operators

The multigrid coarsening is performed only on the spatial discretization so that all $2Qs_{\text{irk}}$ unknowns per spatial node are transferred simultaneously to a coarser level. Assuming that the mean stochastic diffusion coefficients in (2.5) are equal up to a constant factor - which, in practice, is quite often the case - then this also holds for the mean stiffness matrices $K_{1,1}$ and $K_{2,1}$. This motivates the use of a multigrid hierarchy for $K_{1,1}$ as a basis for a multigrid hierarchy for the discretized stochastic two-particle problem. Let P be the AMG prolongation operator derived for $K_{1,1}$. One can construct the prolongation $\mathscr{P}$ and restriction operator $\mathscr{R}$ for the system (3.18) as

$$\mathscr{P} = I_{2Q} \otimes \mathsf{P} \otimes I_{s_{\text{irk}}} \qquad \text{and} \qquad \mathscr{R} = I_{2Q} \otimes \mathsf{P}^T \otimes I_{s_{\text{irk}}}.$$

The Galerkin coarse grid operator $\mathscr{A}_H$ for the Gray-Scott reaction model equals

$$\begin{aligned}
\mathscr{A}_H &= I_2 \otimes C_1 \otimes \mathsf{P}M\mathsf{P}^T \otimes \left(I_{s_{\text{irk}}} + \Delta t\widetilde{A}_{\text{irk}}\right) \\
&\quad + \Delta t \begin{bmatrix} \sum_{i=1}^{S_1} C_i \otimes \mathsf{P}K_{1,i}\mathsf{P}^T & \mathbf{0} \\ \mathbf{0} & C_1 \otimes \mathsf{P}K_{2,1}\mathsf{P}^T + \sum_{i=2}^{S_2} C_{i+S_1-1} \otimes \mathsf{P}K_{2,i}\mathsf{P}^T \end{bmatrix} \otimes \widehat{A}_{\text{irk}} \\
&\quad + \Delta t \begin{bmatrix} \kappa_f C_1 & 0 \\ 0 & \kappa_2 C_1 \end{bmatrix} \otimes \mathsf{P}M\mathsf{P}^T \otimes A_{\text{irk}} \\
&\quad + \kappa_A \Delta t \left[\sum_{i=1}^{Z} \begin{bmatrix} \underline{C}_i \otimes \mathsf{P}G_i^{(k)(c_1)}\mathsf{P}^T & \underline{C}_i \otimes \mathsf{P}J_i^{(k)(c_1)}\mathsf{P}^T \\ -\underline{C}_i \otimes \mathsf{P}G_i^{(k)(c_1)}\mathsf{P}^T & -\underline{C}_i \otimes \mathsf{P}J_i^{(k)(c_1)}\mathsf{P}^T \end{bmatrix} \cdots \right. \\
&\qquad \left. \sum_{i=1}^{Z} \begin{bmatrix} \underline{C}_i \otimes \mathsf{P}G_i^{(k)(c_{s_{\text{irk}}})}\mathsf{P}^T & \underline{C}_i \otimes \mathsf{P}J_i^{(k)(c_{s_{\text{irk}}})}\mathsf{P}^T \\ -\underline{C}_i \otimes \mathsf{P}G_i^{(k)(c_{s_{\text{irk}}})}\mathsf{P}^T & -\underline{C}_i \otimes \mathsf{P}J_i^{(k)(c_{s_{\text{irk}}})}\mathsf{P}^T \end{bmatrix} \otimes A_{\text{irk}}(:, s_{\text{irk}}) \right] P.
\end{aligned}$$

Since the multigrid hierarchy does not depend on the implicit Runge-Kutta discretization or the Newton iteration, the AMG setup needs to be performed only once, independent of the number of time or Newton steps.

5. Implementation aspects

5.1. Matrix formulation and storage

System matrix. The huge algebraic system matrix (3.18) resulting from a stochastic Galerkin discretization is typically never constructed explicitly [25]. Only the separate building blocks are stored in sparse matrix format. As a consequence, for every matrix-vector multiplication or smoothing iteration, the system matrix needs to be assembled on the fly. This increases the computational cost, especially due to iterative assembly of the Zs_{irk} linearized reaction matrices resulting from the Gray-Scott reaction (2.2)–(2.3). In [22], the computational cost of a matrix-vector multiplication is compared in the two cases in which the system matrix is stored either as a whole or in parts.

When the system matrix is stored as a whole, the memory cost equals approximately $Q^2 s_{\mathrm{irk}}^2$ times the cost of storing a deterministic discretization matrix. The above suggested storage format requires storing only $(S_1+S_2+2Z)s_{\mathrm{irk}}$ deterministic discretization matrices, where $Z = \tilde{Q}Q$ with $\tilde{Q} \ll Q$ [22]. Note that in the case of parallel computing, storing the system matrix as a whole may become a better option.

Matrix formulation. In order to implement block relaxation operations and matrix-vector multiplications in a cache efficient way, the linear system (3.18) can be rewritten as a system of matrix equations, as explained in [23] for the discretization of a time-dependent stochastic diffusion problem. The unknown vector $\mathbf{x}$ and right-hand side $\mathbf{b}$ correspond then respectively to the multivectors $X, B \in \mathbb{R}^{N \times 2Qs_{\mathrm{irk}}}$. This storage format provides an easy access of the unknowns per spatial point: they are given by a row of X. The $2Qs_{\mathrm{irk}}$ unknowns per row are ordered first per physical unknown, then per stochastic unknown and finally per IRK stage. This leads to an easy distinction between the physical unknown functions $u_1(\overline{x}, t, y)$ and $u_2(\overline{x}, t, y)$.

5.2. Block smoothing

At every smoothing step, N systems of size $2Qs_{\mathrm{irk}} \times 2Qs_{\mathrm{irk}}$ need to be inverted. Since these local systems are entirely dense, one possibility is to factorize the block systems in advance so that at every smoothing iteration only back substitutions need to be performed. This factorization has to be done at every Newton and time discretization step. This approach however rapidly becomes too expensive from memory cost point of view, for realistic values of N and Q. Therefore, the block systems will be solved at every smoothing iteration directly with SuperLU [5]. Alternative smoothing methods are proposed in Section 5.3.

5.3. Krylov preconditioning

The AMG method developed in Section 4 can be used stand-alone or as preconditioner for a Krylov method. Due to the non-symmetry of the A_{irk}-matrix, BiCGStab is applied for the Krylov acceleration. The computational cost of the multigrid method is dominated by the cost of the matrix-vector multiplication and smoothing operations, which can become very expensive due to the storage format, as discussed above. Cheaper Krylov preconditioners with comparable convergence properties can however be constructed. Their efficiency will be demonstrated numerically in Section 7.

5.3.1. Multigrid for stochastic diffusion problems

In the case of reaction-diffusion problems with a dominant diffusion term, multigrid for time-dependent, linear stochastic diffusion problems [23] can be used as preconditioner for the linearized algebraic systems. The multigrid preconditioner is constructed similarly to the multigrid method described in Section 4. It considers as algebraic system the system (3.18) with all matrices due to reaction removed. The residual computation and smoothing operator therefore do not need to loop over the reaction matrices. This results in a lower computational and memory cost than the AMG method of Section 4.

5.3.2. Multigrid with modified smoother

Alternatively, the multigrid method from Section 4 with a modified smoother is used as preconditioner. The high computational cost of the collective smoother is reduced by omitting any linearized reaction matrices in the smoother. That is, a block smoother for a stochastic Galerkin and implicit Runge-Kutta discretization of a time-dependent diffusion problem is applied as relaxation method. Note that this AMG preconditioner is not convergent when used stand-alone.

5.4. Computation of Gray-Scott Jacobian matrices

The construction of Gray-Scott linearized reaction matrices of the form (3.11)–(3.12) can be very expensive in case of large N and Q. This cost is reduced by employing known information about the precise sparsity structure. Consider, for example, the matrix $J_i^{(k)}(t)$ (3.11), evaluated at time $\hat{t}$ with $\hat{w}_i^{(k)}$ defined by (3.9). The (m,n)-th element of this matrix equals

$$J_i^{(k)}(\hat{t})(m,n) := \sum_{c=1}^{Q}\sum_{q=1}^{Q}\langle \Psi_c \Psi_q \Psi_i\rangle \int_{\overline{D}} \hat{u}_{1,c}^{(k)}(\overline{x},\hat{t})\hat{u}_{2,q}^{(k)}(\overline{x},\hat{t}) s_m(\overline{x}) s_n(\overline{x}) d\overline{x}. \qquad (5.1)$$

From the sparsity structure of matrix $\underline{C}_i$ (3.13), as detailed in [9], it follows that many scalars $\langle \Psi_c \Psi_q \Psi_i\rangle$ in (5.1) are zero. Denote by $nz(c)$ the number of nonzero elements in row c of matrix $\underline{C}_i$, with corresponding vector of column indices cn_c. The double summation in (5.1) can then be rewritten as

$$J_i^{(k)}(\hat{t})(m,n) := \sum_{c=1}^{Q}\sum_{q=1}^{nz(c)}\langle \Psi_c \Psi_{cn_c(q)} \Psi_i\rangle \int_{\overline{D}} \hat{u}_{1,c}^{(k)}(\overline{x},\hat{t})\hat{u}_{2,cn_c(q)}^{(k)}(\overline{x},\hat{t}) s_m(\overline{x}) s_n(\overline{x}) d\overline{x}. \qquad (5.2)$$

After representing both $\hat{u}_{1,c}^{(k)}$ and $\hat{u}_{2,cn_c(q)}^{(k)}$ in (5.2) with a spatial finite element discretization, we have that

$$J_i^{(k)}(\hat{t})(m,n) := \sum_{c=1}^{Q}\sum_{q=1}^{nz(c)}\langle\Psi_c\Psi_{cn_c(q)}\Psi_i\rangle\sum_{r=1}^{N}\sum_{\ell=1}^{N}\hat{u}_{1,cr}^{(k)}(\hat{t})\hat{u}_{2,cn_c(q)\ell}^{(k)}(\hat{t}) \\ \times\int_{\overline{D}} s_r(\overline{x})s_\ell(\overline{x})s_m(\overline{x})s_n(\overline{x})d\overline{x}, \tag{5.3}$$

where $\hat{u}_{1,cr}^{(k)}$ and $\hat{u}_{2,cn_c(q)\ell}^{(k)}$ are the coefficient functions of the finite element representation of the functions $\hat{u}_{1,c}^{(k)}$ and $\hat{u}_{2,cn_c(q)}^{(k)}$, corresponding to node r and ℓ, respectively. The cost of evaluating the double summation over N in (5.3) becomes excessively large for practical values of N. Since the piecewise linear nodal basis functions $s_n(\overline{x})$ are nonzero on only a small part of the domain $\overline{D}$, this information can be used to reduce the computational cost. The sparsity pattern of, e.g., the mass matrix M reveals which nodal basis functions, s_r and s_ℓ, do not overlap. Therefore, we can rewrite (5.3) as

$$J_i^{(k)}(\hat{t})(m,n) := \sum_{c=1}^{Q}\sum_{q=1}^{nz(c)}\langle\Psi_c\Psi_{cn_c(q)}\Psi_i\rangle\sum_{r=1}^{N}\sum_{\ell=1}^{nz^M(r)}\hat{u}_{1,cr}^{(k)}(\hat{t})\hat{u}_{2,cn_c(q)cn_r^M(\ell)}^{(k)}(\hat{t}) \\ \times\int_{\overline{D}} s_r(\overline{x})s_{cn_r^M(\ell)}(\overline{x})s_m(\overline{x})s_n(\overline{x})d\overline{x},$$

where $nz^M(r)$ is the number of nonzero elements of row r in matrix M, and cn_r^M contains the corresponding indices of the nodal basis functions.

Remark 5.1. When the reaction model R in (2.1) contains quadratic nonlinearities instead of the cubic terms in (2.2)–(2.3), the construction of the linearized reaction matrices involves only one random field instead of a product of two random fields as in (3.7)–(3.8). Therefore, a similar construction procedure is not required in the quadratic case. This also means that a quadratic nonlinear problem results typically in a lower computational cost than a cubic problem.

6. Bio-engineering application

We consider the conversion of starch into sugars in a growing apple. The physical unknowns u_1 and u_2 in (2.1) represent the concentrations of two hormones U and V which control the sugar production process in an apple. The identification of the precise hormones is still the subject of ongoing research, see, e.g., [26].

The 3D geometry, a horizontal and vertical cut, and the wire frame model are illustrated in Fig. 1. The model consists of four different subdomains. The small black spots represent the *seeds*; typically 5 to 10 seeds are present in the apple. The seeds are positioned inside the *ovary* of the apple. The ovary is part of the *core* or *pith of receptacle*. The

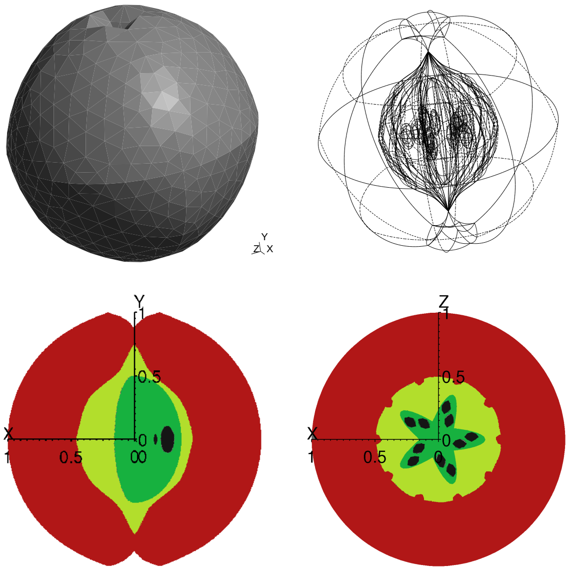


Figure 1: Computer model of an apple; vertical and horizontal cut of the apple model.

large remaining part of the apple up to the skin of the apple is the *cortex of receptacle*, i.e., the *fleshy tissue* or apple flesh. These four different regions are characterized by different material properties. The boundary of the domain corresponds to the *skin* of the apple.

Growth function for apples. Experiments have shown that the mass of apples follows a logistic growth curve with maximum growth rate α. Assuming a constant apple density, this logistic trend applies to the volume of an apple. From the relation between the volume and radius of a spherical object, the evolution of the boundary, $r_0 r(t)$, of an apple can be deduced. The growth function $r(t)$ is then given by

$$r(t) = \frac{1}{\left((1-\beta^3)\exp(-\alpha t)+\beta^3\right)^{\frac{1}{3}}}, \tag{6.1}$$

with $\beta = r_0/r_\infty$, r_0 the initial apple radius, i.e., radius at time $t = 0$, and r_∞ the final, stationary, apple radius. In the numerical experiments, the parameters of (6.1) are set to $\alpha = 0.048$, $r_0 = 3.5$ cm and $r_\infty = 5$ cm.

Initial condition. At time $t = 0$, a particle of substance U is released from the seeds:

$$u_1(\overline{x},0,y) = \begin{cases} 1/n_s, & \overline{x} \in \overline{D}_{\text{seeds}}, \\ 0, & \overline{x} \notin \overline{D}_{\text{seeds}}, \end{cases} \tag{6.2a}$$

$$u_2(\overline{x},0,y) = 1 - u(\overline{x},0,y), \tag{6.2b}$$

with n_s the number of seeds in the apple. In order to avoid Gibbs phenomena due to a discontinuous initial condition as in (6.2), a smooth approximation to (6.2) is applied at time $t_{\text{init}} > 0$,

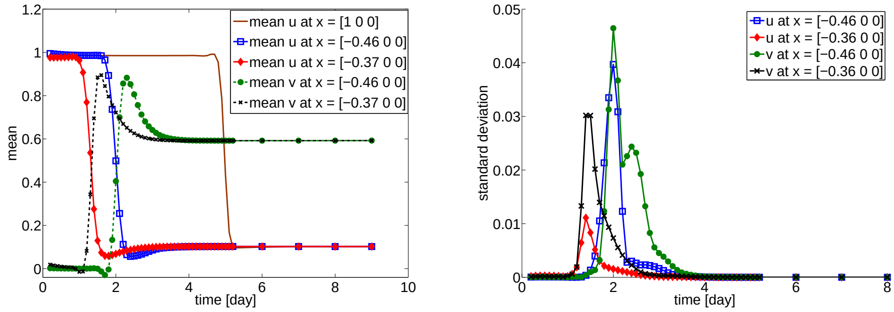
$$u_1(\overline{\boldsymbol{x}}, t_{\text{init}}, y) = \frac{1}{n_s} \sum_{c=1}^{n_s} \frac{\operatorname{erfc}(\|\overline{\boldsymbol{x}} - \overline{\boldsymbol{x}}_{\text{seed}_c}\|_2)}{2\sqrt{\hat{a} t_{\text{init}}}}, \tag{6.3}$$

where $\hat{a}$ is a constant representing the mean diffusion coefficient, n_{seeds} equals the number of seeds, and $\overline{\boldsymbol{x}}_{\text{seed}_c}$ corresponds to the coordinates of the center of seed number c. The numerical results will be based on $\hat{a} = 0.04$ and $t_{\text{init}} = 0.1$. The initial condition (6.3) is based on a 1D diffusion solution, where the denominator $2\sqrt{\hat{a} t_{\text{init}}}$ expresses how far the concentration propagates in one dimension by diffusion at time t_{init}. Since the 3D model is nearly axisymmetric, such an extension of a 1D solution is suitable as a rough approximate initial solution.

Random coefficients. The diffusion coefficients a_1 and a_2 in (2.1) are modelled as piecewise constant random fields, thereby expressing the heterogeneity of the apple tissue and core. In particular, the following setup is used in the numerical tests:

$$a_1 = \begin{cases} 0.04\xi_1, \\ 0.035\xi_2, \\ 0.032, \\ 0.03, \end{cases} \quad \text{and} \quad a_2 = \begin{cases} 0.02\xi_3 & \overline{\boldsymbol{x}} \in D_{\text{tissue}}, \\ 0.015\xi_4 & \overline{\boldsymbol{x}} \in D_{\text{core}}, \\ 0.012 & \overline{\boldsymbol{x}} \in D_{\text{ovary}}, \\ 0.01 & \overline{\boldsymbol{x}} \in D_{\text{seeds}}. \end{cases}$$

The random variables $\xi_1, \cdots, \xi_4$ are independent and lognormally distributed, based on zero-mean Gaussian variables with standard deviation 0.1. The dimensionless Gray-Scott reaction coefficients (2.2)–(2.3) are assumed to be deterministic and equal to $\kappa_A = 50$, $\kappa_2 = 3$ and $\kappa_f = 2$. Figs. 2–4 illustrate the time evolution of the mean and standard deviation of the concentrations u_1 and u_2 using these configurations.



(a) evolution of the mean in time (b) evolution of the standard deviation in time

Figure 2: Solution of the Gray-Scott reaction model (2.1), where the diffusion coefficients a_1 and a_2 are each modeled by two independent lognormal random variables. The problem is discretized with a spatial finite element mesh with 44581 nodes, a first-order Hermite expansion and an implicit Euler time discretization.

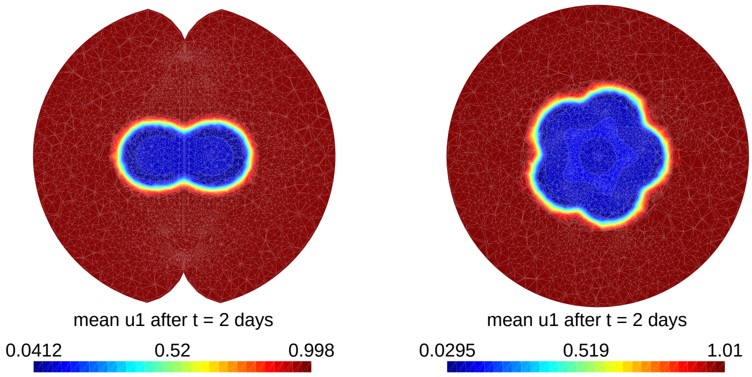


Figure 3: Vertical and horizontal cut of the mean of u_1. The same configuration as in Fig. 2 is used.

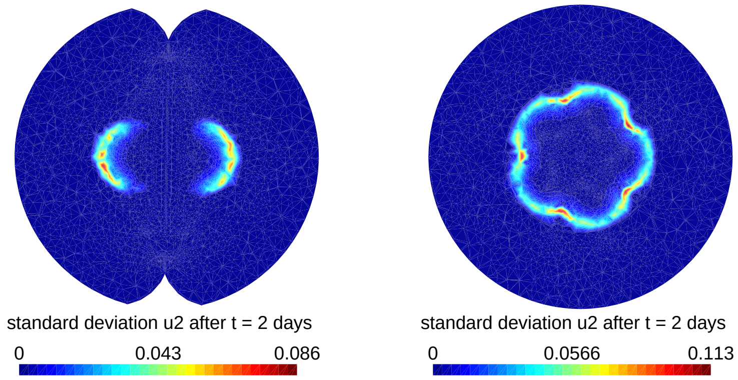


Figure 4: Vertical and horizontal cut of the standard deviation of u_2 (same configuration as in Fig. 2).

7. Numerical results

In this section, the performance and convergence properties of the various multigrid solution and preconditioning methods proposed in the previous sections are illustrated numerically. The Gray-Scott reaction model is solved on the 3D apple domain depicted in Fig. 1, using the configurations described in Section 6.

Unless stated differently, multigrid F-cycles will be applied, with 3 pre-smoothing and 2 post-smoothing iterations. In order to study the multigrid convergence behavior in depth, the linearized equations are solved up to machine precision, i.e., until the relative residual $||r||/||b||$ is smaller than 10^{-14}. The iteration counts and solution times given are averaged over multiple Newton and time steps. The time discretization uses as time step $\Delta t = 0.02$ day (about 0.5h). The computations are performed on a 2.66 GHz Intel Xeon X5550 8-core machine with 32 GByte RAM.

Table 1: Average iteration counts and solution time (in seconds) when solving linearized systems of the form (3.18) with BiCGStab preconditioned by the multigrid method described in Section 4. The geometry and model parameters are described in Section 6. A Hermite stochastic discretization defined on 4 random variables is applied.

Spatial nodes	Polynomial order	IRK stages	$2Qs_{\text{irk}}$	iterations	time
11 799	2	2	60	7.5	424
11 799	3	2	140	7.5	2514
24 477	1	3	30	11.5	231
24 477	2	2	60	12.0	1204
24 477	3	2	140	12.0	9243
43 393	1	2	20	11.7	259
43 393	2	2	60	11.8	2095

7.1. Multigrid convergence properties

7.1.1. Multigrid convergence properties

First, we verify the robustness properties of the multigrid method presented in Section 4 by solving the linearized systems (3.18) for different spatial, stochastic and time discretization parameters. Table 1 illustrates the multigrid convergence properties. For different configurations of the discretization, a similar number of iterations results. This suggests that the proposed multigrid method has so-called optimal convergence properties, i.e., that the convergence rate is independent of the spatial, stochastic and time discretization parameters.

7.1.2. Alternative multigrid preconditioners

The large computational cost of multigrid, as illustrated in Table 1, is caused to a large extent by the expensive block smoothing iterations. At every smoothing iteration, N blocks of size $(2Qs_{\text{irk}} \times 2Qs_{\text{irk}})$ have to be inverted. Alternatively, one could also construct a block relaxation method with blocks of size $(2s_{\text{irk}} \times 2s_{\text{irk}})$ in order to reduce the computational cost. The IRK unknowns for u_1 and u_2 are then updated simultaneously. In one iteration of the $(2s_{\text{irk}} \times 2s_{\text{irk}})$ block smoother, NQ systems of size $(2s_{\text{irk}} \times 2s_{\text{irk}})$ need to be solved, one for each spatial and stochastic unknown. Moreover, since the system matrix is not stored as a whole in memory, the block systems have to be re-assembled during every smoothing iteration. The computational cost of assembling the block systems is reduced by omitting the reaction matrices in the smoothing operations. This motivates the alternative Krylov preconditioners discussed in Section 5.3. The performance of these different multigrid preconditioners is compared in Table 2.

Table 2 shows that the alternative multigrid preconditioners from Section 5.3 effectively reduce the total solution time while maintaining a similar convergence rate to the original multigrid method proposed in Section 4. A $(2s_{\text{irk}} \times 2s_{\text{irk}})$ block relaxation method results in a higher computational cost than a $(2Qs_{\text{irk}} \times 2Qs_{\text{irk}})$ block smoother. This indicates that the cost of assembling the block systems possibly dominates the cost of solving the block systems at every smoothing iteration.

7.1.3. Convergence properties of alternative multigrid preconditioners

Table 2 shows that the multigrid preconditioner A (see Section 5.3) results in the lowest total solution time. In order to verify the robustness of the preconditioner, the experiments from Table 1 are repeated for this multigrid preconditioner, see Table 3. On average, a robust convergence behavior is maintained.

7.2. Alternative reaction model

7.2.1. Quadratic reaction model

In order to check the robustness of the proposed multigrid methods, the methods are applied to an alternative reaction model. We consider the following, so-called competing species, quadratic reaction model:

$$R_1(u_1,u_2)=\epsilon_1 u_1-\varsigma_1 u_1^2-\chi_1 u_1 u_2, \tag{7.1}$$

$$R_2(u_1,u_2)=\epsilon_2 u_2-\varsigma_2 u_2^2-\chi_2 u_1 u_2, \tag{7.2}$$

where ϵ_1, ϵ_2, ς_1, ς_2, χ_1 and χ_2 are positive reaction constants. In the numerical experiments, we will perturb the diffusion coefficients by 8 random variables:

$$a_1=\begin{cases}0.04\xi_1,\\ 0.035\xi_2,\\ 0.032\xi_3,\\ 0.03\xi_4,\end{cases} \quad \text{and} \quad a_2=\begin{cases}0.02\xi_5 & \overline{\boldsymbol{x}}\in D_{\text{tissue}},\\ 0.015\xi_6 & \overline{\boldsymbol{x}}\in D_{\text{core}},\\ 0.012\xi_7 & \overline{\boldsymbol{x}}\in D_{\text{ovary}},\\ 0.01\xi_8 & \overline{\boldsymbol{x}}\in D_{\text{seeds}}.\end{cases}$$

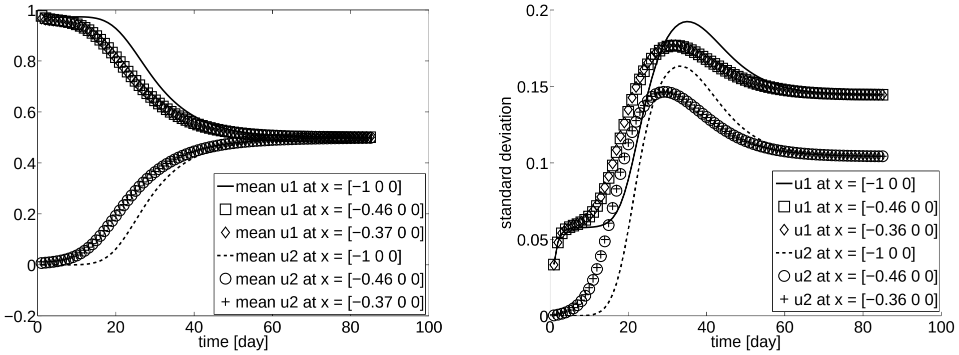
The random variables ξ_1-ξ_8 are independent and lognormally distributed, based on zero-mean Gaussian random variables with standard deviation 0.1. The reaction coefficients ϵ_1 and ϵ_2 are random variables, uniformly distributed on $[0.9,1.1]$ and $[0.675,0.825]$. The other reaction coefficients are constant and given by $\varsigma_1=1$, $\varsigma_2=1$, $\chi_1=1$ and $\chi_2=0.5$. Fig. 5 shows the evolution of the statistics of u_1 and u_2 in time. The same initial condition

Table 2: Average iteration counts and and solution time (in seconds) when solving systems of the form (3.18) with BiCGStab preconditioned by one of the multigrid methods described in Sections 4 and Section 5.3. A spatial discretization of 24477 nodes is used, together with a second-order Hermite stochastic discretization defined on 4 random variables and a Radau IIA IRK discretization of order 3.

Block size smoother	Multigrid variant	iterations	time
$(2Qs_{\text{irk}}\times 2Qs_{\text{irk}})$	original (see Section 4)	12.0	1204
	alternative A (see Section 5.3)	11.9	282
	alternative B (see Section 5.3)	12.7	471
$(2s_{\text{irk}}\times 2s_{\text{irk}})$	original (see Section 4)	12.1	5584
	alternative A (see Section 5.3)	11.7	542
	alternative B (see Section 5.3)	12.5	753

Table 3: Average iteration counts and solution time (in seconds) when solving linearized systems of the form (3.18) with BiCGStab preconditioned by the multigrid preconditioner A (see Section 5.3). A Hermite stochastic discretization defined on 4 random variables is applied.

Spatial nodes	Polynomial order	IRK stages	$2Qs_{\text{irk}}$	iterations	time
11 799	2	2	60	8.9	102
11 799	3	2	140	8.7	428
24 477	1	3	30	10.8	85.6
24 477	2	2	60	11.9	282
24 477	3	2	140	11.9	1991
43 393	1	2	20	11.8	123
43 393	2	2	60	12.2	534



(a) mean u_1 and u_2

(b) standard deviation u_1 and u_2

Figure 5: Solution of the stochastic two-particle model (2.1) with competing species reaction (7.1)–(7.2) and 10 random variables. A spatial discretization with 11 799 nodes, a 2$^{\text{nd}}$ order Hermite-Legendre expansion and an implicit Euler time discretization are applied.

and geometry as described in Section 6 were used. The implicit Runge-Kutta time step equals $\Delta t = 0.1$ day for all experiments in this section.

Table 4: Average iteration counts and solution time (in seconds) when solving linearized systems of the form (3.18), where the Gray-Scott reaction is replaced by the linearization of the quadratic reaction (7.1)–(7.2). A Hermite-Legendre stochastic discretization is applied.

Spatial nodes	Polynomial	IRK	$2Qs_{\text{irk}}$	BiCGStab–AMG		AMG prec B	
	order	stages		iter	time	iter	time
8 lognormal random variables							
11 799	3	1	330	15.4	26535	17.1	6472
24 477	2	3	270	16	11013	16	7813
43 393	2	2	180	15.8	11075	16.5	4684
8 lognormal and 2 uniform random variables							
11 799	3	1	572	14.3	67292	16.9	14612
24 477	2	2	264	16	13830	17.3	4623
43 393	2	1	132	15.9	9769	17.4	3103

Remark 7.1. The linearization of the quadratic model results in an expansion with only Q linearized terms instead of the $Z = (L + 2P)!/(L!(2P)!)$ terms in the Gray-Scott case, see (3.14). The special procedure for constructing the linearized reaction matrices presented in Section 5.4 is therefore not needed. As a consequence, the computational and memory cost of this quadratic reaction model are substantially lower than in the cubic Gray-Scott reaction case for problems with the same number of unknowns per spatial point.

7.2.2. Multigrid convergence properties

Table 4 presents the iteration counts and solution time when solving the linearized system (3.18), based on the quadratic reaction terms (7.1)–(7.2), for various configurations of the discretization parameters. When a model with only 8 random variables is considered, the reaction coefficients ϵ_1 and ϵ_2 are constant and equal 1 and 0.75, respectively. Results for both the multigrid method of Section 4 and the multigrid preconditioner B in Section 5.3 are given. No results for preconditioner A in Section 5.3 are shown, since for this reaction model, the convergence rate of BiCGStab with preconditioner A is unsatisfactory. Preconditioner B succeeds in reducing the computational cost of the original multigrid method substantially, without deterioration of the multigrid convergence rate. By comparing the iteration counts corresponding to different discretization sizes we find that the good multigrid convergence properties shown in Section 7.1 also hold for this alternative reaction model.

8. Conclusions

This paper presents an algebraic multigrid method to solve the high-dimensional algebraic systems that result from a stochastic Galerkin finite element discretization of a system of nonlinear, stochastic reaction-diffusion problems. Numerical experiments indicate that the proposed method has very good and robust convergence properties. In order to check the dependency of the convergence results on the reaction model, two different reaction models were tested.

The computational cost of the point-oriented algebraic multigrid solver may become excessively large due to the iterative assembly and inversion of local systems in the block relaxation method. To alleviate this problem, cheaper multigrid preconditioners were proposed. These succeed in a substantial reduction of the computational cost while maintaining a similar convergence behavior as the original multigrid method.

The presented simulation procedure enables one to compute the concentration of chemical particles in a growing apple. This simulation can be used to model the transition of starch into sugar. In order to set up more realistic simulations, additional experiments are needed to determine an accurate reaction model and parameters.

Acknowledgments E. Rosseel is a research assistant of the Research Foundation - Flanders (Belgium).

References

[1] I. Babuška, F. Nobile, and R. Tempone. A stochastic collocation method for elliptic partial differential equations with random input data. *SIAM Rev.*, 52(2):317–355, 2010.

[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] J. C. Butcher and G. Wanner. Runge-Kutta methods: some historical notes. *Appl. Numer. Math.*, 22:113–151, 1996.

[4] T. Clees. *AMG strategies for PDE systems with applications in industrial semiconductor simulation*. PhD thesis, University of Cologne, Cologne, Germany, 2004.

[5] J. W. Demmel, S. C. Eisenstat, J. R. Gilbert, X. S. Li, and J. W. H. Liu. A supernodal approach to sparse partial pivoting. *SIAM J. Matrix Anal. A.*, 20(3):720–755, 1999.

[6] H. Elman, O. G. Ernst, D. P. O'Leary, and M. Stewart. Efficient iterative algorithms for the stochastic finite element method with application to acoustic scattering. *Comput. Methods Appl. Mech. Engrg.*, 194:1037–1055, 2005.

[7] H. C. Elman, C. W. Miller, E. T. Phipps, and R. S. Tuminaro. Assessment of collocation and Galerkin approaches to linear diffusion equations with random data. *Journal for Uncertainty Quantification*, 1:19-34, 2011.

[8] O. G. Ernst, C. E. Powell, D. Silvester, and E. Ullmann. Efficient solvers for a linear stochastic galerkin mixed formulation of diffusion problems with random data. *SIAM J. Sci. Comput.*, 31(2):1424–1447, 2009.

[9] O. G. Ernst and E. Ullmann. Stochastic Galerkin matrices. *SIAM J. Matrix Anal. Appl.*, 31(4):1848–1872, 2010.

[10] J. Foo, Z. Yosibash, and G. E. Karniadakis. Stochastic simulation of riser-sections with uncertain measured pressure loads and/or uncertain material properties. *Comput. Methods Appl. Mech. Engrg.*, 196:4250–4271, 2007.

[11] R. G. Ghanem and P. D. Spanos. *Stochastic finite elements, a spectral approach*. Dover, Mineola, New York, 2nd edition, 2003.

[12] E. Hairer and G. Wanner. *Solving ordinary differential equations II: stiff and differential-algebraic problems*. Springer, Berlin, Germany, 2002.

[13] A. Keese. *Numerical solution of systems with stochastic uncertainties.* PhD thesis, Technische Universität Braunschweig, Braunschweig, Germany, 2004.

[14] O. M. Knio and O. P. Le Maître. Uncertainty propagation in CFD using polynomial chaos decomposition. *Fluid Dynamics Research*, 38:616–640, 2006.

[15] O. P. Le Maître, O. M. Knio, H. N. Najm, and R. Ghanem. Uncertainty propagation using Wiener-Haar expansions. *J. Comput. Phys.*, 197:28–57, 2004.

[16] M. Loève. *Probability theory*. Springer, New York, USA, 1977.

[17] L. Mathelin, M. Y. Hussaini, and T. A. Zang. Stochastic approaches to uncertainty quantification in CFD simulations. *Numerical Algorithms*, 38:209–236, 2005.

[18] H. G. Matthies. Stochastic finite elements: computational approaches to stochastic partial differential equations. *Z. Angew. Math. Mech.*, 88(11):849–873, 2008.

[19] H. G. Matthies and A. Keese. Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations. *Comput. Methods Appl. Engrg.*, 194:1295–1331, 2005.

[20] J. S. McGough and K. Riley. Pattern formation in the Gray-Scott model. *Nonlinear analysis: real world applications*, 5(1):105–121, 2004.

[21] F. Nobile and R. Tempone. Analysis and implementation issues for the numerical approximation of parabolic equations with random coefficients. *Int. J. Numer. Meth. Eng.*, 80:979–1006, 2009.

[22] C. E. Powell and E. Ullmann. Preconditioning stochastic Galerkin saddle point systems. SIAM J. Matrix. Anal. Appl., 31(5):2813-2840, 2010.

[23] 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.

[24] E. Rosseel, H. De Gersem, and S. Vandewalle. Nonlinear stochastic Galerkin and collocation methods: application to a ferromagnetic cylinder rotating at high speed. *Commun. Comput. Phys.*, 8(5):947–975, 2010.

[25] E. Rosseel and S. Vandewalle. Iterative solvers for the stochastic finite element method. *SIAM J. Sci. Comput.*, 32(1):372–397, 2010.

[26] N. Scheerlinck, C. Escudero, R. E. Baker, and P. K. Maini. Modelling spatio-temporal starch degradation patterns in apples during fruit growth. *Journal of Agricultural Science*, 145(6):645–646, 2007.

[27] K. Stüben. A review of algebraic multigrid. *J. Comput. Appl. Math.*, 128:281–309, 2001.

[28] U. Trottenberg, C. W. Oosterlee, and A. Schüller. *Multigrid*. Academic Press, San Diego, USA, 2001.

[29] X. Wan and G. E. Karniadakis. An adaptive multi-element generalized polynomial chaos method for stochastic differential equations. *J. Comput. Phys.*, 209:617–642, 2005.

[30] D. Xiu and G. E. Karniadakis. The Wiener–Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.*, 24(2):619–644, 2002.

[31] D. Xiu, D. Lucor, S.-H. Su, and G. E. Karniadakis. Stochastic modeling of flow structure interactions using generalized polynomial chaos. *J. Fluids Eng.*, 124:51–59, 2002.