A least-squares approximation of high-dimensional uncertain systems

By A. Doostan, G. Iaccarino AND N. Etemadi †

1. Motivation and objectives

The behavior and evolution of complex systems are known only partially due to lack of knowledge about the governing physical laws or limited information regarding their operating conditions and input parameters (e.g., material properties). Uncertainty quantification (UQ) plays a crucial role in the construction of credible mathematical/computational models for such systems. In this context a set of a partial differential equations (PDEs) are considered. The uncertain parameters are assimilated based on the available information and are described as random variables in a probabilistic framework. A computational model is then defined to approximate the statistics of the solution of these PDEs, thus propagating the uncertainty from the input parameters to the response of the system.

Bringing the uncertainty propagation schemes into focus, the Monte Carlo sampling has been utilized for a long time as a general purpose scheme. There has recently been an increasing interest in developing more efficient computational models for the analysis of uncertain systems as compared to Monte Carlo techniques that are generally known to have a slow rate of convergence. In particular, *perturbation-based* techniques, Kleiber & Hien (1992), are shown to be effective for situations where the input parameters exhibit small variability. *Stochastic Galerkin schemes* have been successfully applied to different areas of engineering and have been proven efficient for situations where the number of uncertain parameters are not large, e.g., Ghanem & Spanos (2003); Xiu & Karniadakis (2002); Babuška, Tempone & Zouraris (2004). More recently, there has been a great deal of attention to *collocation-based* techniques that take advantage of existing deterministic solvers for medium range, $\mathcal{O}(10)$, number of uncertain variables Xiu & Hesthaven (2005); Nobile, Tempone & Webster (2006, 2007).

In many applications, one has to deal with a large number of uncertain input parameters. In such situations, with the exception of the Monte Carlo technique, the above-mentioned methods (in their conventional form) suffer from the so-called *curse of dimensionality*: the associated computational cost grows exponentially as a function of the number of random variables defining the underlying probability space of the problem. More specifically, the computational cost of the stochastic Galerkin schemes with global polynomials, e.g., Wiener Hermite chaos, Ghanem & Spanos (2003), depends on the number of terms in the solution expansion. For a case of $p-th$ order approximation in d independent random variables, the cardinality of the associated basis is $P = \frac{(p+d)!}{p!d!} - 1$, which increases exponentially with respect to p and d. Model reduction techniques have been developed to reduce this drawback for the case of stochastic Galerkin schemes by Doostan, Ghanem & Red-Horse (2007); Ghanem, Saad & Doostan (2007); Nouy (2007).

Stochastic collocation schemes, Xiu & Hesthaven (2005); Nobile, Tempone & Webster (2006, 2007) based on isotropic and anisotropic sparse grids, reduce the problem

† Affiliates University of Illinois at Chicago, Chicago, IL, USA.

of the curse of dimensionality associated to conventional tensor product schemes. The computational cost of the stochastic collocation scheme using a tensor-product grid and a sparse-grid consisting of M points in each direction of a d-dimensional space is $\mathcal{O}(M^d)$ and $\mathcal{O}(C^d M(\log M)^{d-1})$, respectively. Note that both estimates grow exponentially with respect to d.

This work applies the *alternating least-squares* (ALS) approximation technique of Beylkin & Mohlenkamp (2002, 2005) to tackle the curse of dimensionality issue. In theory, the computational cost of this algorithm grows linearly with respect to the dimension of the probability space of the system. The fundamental feature is the adoption of a low *separation-rank* approximation of functions in the probability space.

The test problem considered here is a thermal problem on a domain with uncertain thermal conductivity and initial temperature. The thermal diffusivity is modeled as a random field obtained from a truncated Karhunen-Loeve expansion. The associated covariance kernel is such that a large number of terms (hence random variables) are required to capture a significant portion of the signal energy. The uncertainty in the initial temperature is characterized by two random variables. A simple implicit finite difference scheme is used for the spatial-temporal discretization of the governing equation. Given the uncertainty, the discrete problem is essentially a random linear system of equations. A low separation-rank approximation of the solution of this system is obtained by minimizing the L_2 norm of the residual on a tensor product grid in the physical-probability space. The particular form of the separation-rank representation of uncertain quantities allow one to perform the minimization task alternately in $d+1$ dimensions. In theory, the associated computational cost of this analysis is linear with respect to the dimension of the corresponding probability space d see Beylkin & Mohlenkamp (2002, 2005).

The paper is organized as follows. Section 2 briefly summarizes the ALS approximation of a general d-dimensional function proposed by Beylkin & Mohlenkamp (2002, 2005). It will then be shown how one can cast the discrete form of the governing stochastic partial differential equation (SPDE) in a form that can be readily incorporated in the proposed least-squares approximation. Finally, in Section 3, a numerical experiment is performed on 1-D unsteady thermal problem to illustrate the performance of the proposed procedures.

2. Numerical method

As mentioned above, the computational cost of two commonly used uncertainty propagation schemes, i.e., stochastic Galerkin and stochastic collocation schemes, generally speaking, grows exponentially with respect to the dimension of the underlying probability space. The present work tackles this problem by adopting the formalism of *alternating least-squares* approximation of high-dimensional functions of Beylkin & Mohlenkamp (2002, 2005), which will be introduced in Section 2.1.

2.1. *Separated representation*

The separation-of-variable techniques have been used to approximate high-dimensional functions using one-dimensional operations, thus virtually eliminating the curse of dimensionality, e.g., Pereyra & Scherer (1973). Let u be a d-dimensional function. It can be approximated as

$$u(y_1, \cdots, y_d) \approx \phi_1(y_1) \cdots \phi_d(y_d). \tag{2.1}$$

This approximation can be improved by introducing a series of such representations, i.e.,

$$u(y_1, \cdots, y_d) = \sum_{l=1}^{r} s_l \phi_1^l(y_1) \cdots \phi_d^l(y_d) + \mathcal{O}(\epsilon), \tag{2.2}$$

which is called a *separated representation* with *separation rank* r (Beylkin & Mohlenkamp (2002, 2005)). Given a target accuracy ϵ, the approximation (2.2) can be achieved by tailoring unknown quantities $\{\phi_i^l(y_i)\}$, $\{s_l\}$, and an optimal separation rank r, for instance, through a nonlinear optimization scheme. The main advantage of adopting the separated representation (2.2) is that many algebraic operations in d dimensions can be performed using a series of one-dimensional operations. Therefore, in theory, the computational efforts build up linearly with respect to d. Our goal is to present the ALS algorithm of Beylkin & Mohlenkamp (2002, 2005) for the separated representation of the solution of a linear system of equations arising from spatial/temporal discretization of SPDEs using only one-dimensional operations. For this sake some preliminary notations and definitions are introduced.

Notation 1. A scalar function $u(y_1, \cdots, y_d)$ in d dimensions is a mapping from the Euclidean space $\mathbb{R}^d$ to real line $\mathbb{R}$, $u : \mathbb{R}^d \to \mathbb{R}$. A vector $\mathbf{u}$ is a discrete representation of function u on a d-dimensional hyper-cube grid, i.e., $\mathbf{u} = u(j_1, \cdots, j_d)$ with $j_k = 1, \cdots, M_k$. Without loss of generality it is assumed that $M_k = M$ for $k = 1, \cdots, d$.

Definition 1 [Beylkin & Mohlenkamp (2002, 2005)] (Separated Representation of a Vector). *For a given accuracy, ϵ, a vector $\mathbf{u} = u(j_1, \cdots, j_d)$ in d dimension is approximated by*

$$\sum_{l=1}^{r} s_l u_1^l(j_1) u_2^l(j_2) \cdots u_d^l(j_d) \equiv \sum_{l=1}^{r} s_l \mathbf{u}_1^l \otimes \mathbf{u}_2^l \otimes \cdots \otimes \mathbf{u}_d^l, \tag{2.3}$$

with s_l being a scalar and $\mathbf{u}_k^l$ being one-dimensional vectors with entries $u_k^l(j_k)$ and unit norm. The approximation error is required to be less than ϵ, i.e.,

$$\|\mathbf{u} - \sum_{l=1}^{r} s_l \mathbf{u}_1^l \otimes \mathbf{u}_2^l \otimes \cdots \otimes \mathbf{u}_d^l\| \leq \epsilon, \tag{2.4}$$

where $\|\cdot\|$ denotes the Frobenius norm over all grid points.

2.1.1. *Reducing the separation rank r*

Let v be a function whose discrete form $\mathbf{v}$ has a large separation rank r_v. The ALS algorithm aims at reducing the separation rank of $\mathbf{v}$ while maintaining an accurate representation. More specifically, let

$$\mathbf{v} = \sum_{l=1}^{r_\mathbf{v}} s_l^\mathbf{v} \mathbf{v}_1^l \otimes \mathbf{v}_2^l \otimes \cdots \otimes \mathbf{v}_d^l \tag{2.5}$$

be known with large r_v. Given an ϵ, the goal is to find

$$\mathbf{u} = \sum_{l=1}^{r_\mathbf{u}} s_l^\mathbf{u} \mathbf{u}_1^l \otimes \mathbf{u}_2^l \otimes \cdots \otimes \mathbf{u}_d^l, \tag{2.6}$$

with $r_u \ll r_v$ such that

$$\|\mathbf{v} - \mathbf{u}\| \leq \epsilon. \tag{2.7}$$

It is assumed that such lower separation-rank approximation of $\mathbf{v}$ exists, otherwise the following reduction algorithm is not efficient.

For a fixed $r_{\mathbf{u}}$, one can minimize the distance between $\mathbf{v}$ and $\mathbf{u}$ as much as possible by adapting $\{\mathbf{u}_i^l\}$ and $\{s_l^{\mathbf{u}}\}$. Due to the already separated form of $\mathbf{v}$ and $\mathbf{u}$ such minimization can be performed in the form of a series of linear one-dimensional optimization problems and is referred to as alternating least-squares algorithm (ALS) in Beylkin & Mohlenkamp (2002, 2005).

Alternating Least-Squares (ALS) *algorithm*. For a fixed $r_{\mathbf{u}}$, an initial guess for $\mathbf{u}$ is constructed by randomly initializing $\{\mathbf{u}_i^l\}$ and $s_l^{\mathbf{u}}$. The optimization steps are then as follows:

- Loop over dimensions $k = 1, \cdots, d$
  - Loop over grid points $j_k = 1, \cdots, M$ in direction k
    — Fix $\{\mathbf{u}_i^{\tilde{l}}\}_{i \neq k}$ and solve the following normal equation associated with a linear least-squares problem to update $\{\mathbf{u}_k^{\tilde{l}}\}$ and thus $s_{\tilde{l}}^{\mathbf{u}}$:

$$\mathbb{B}\mathbf{c}_{j_k} = \mathbf{b}_{j_k}, \tag{2.8}$$

with

$$\mathbb{B}(\hat{l}, \tilde{l}) = \prod_{i \neq k} \langle \mathbf{u}_i^{\hat{l}}, \mathbf{u}_i^{\tilde{l}} \rangle$$

and

$$\mathbf{b}_{j_k}(\hat{l}) = \sum_{l=1}^{r_{\mathbf{v}}} s_l^{\mathbf{v}} v_k^l(j_k) \prod_{i \neq k} \langle \mathbf{v}_i^l, \mathbf{u}_i^{\hat{l}} \rangle.$$

  - Update $s_{\tilde{l}}^{\mathbf{u}} = \left(\sum_j c_{j_k}^2(\tilde{l}) \right)^{1/2}$ and $u_k^{\tilde{l}}(j_k) = c_{j_k}(\tilde{l}) / s_{\tilde{l}}^{\mathbf{u}}$, where $\tilde{l} = 1, \cdots, r_{\mathbf{u}}$ and $j_k = 1, \cdots, M$.

Where $\langle \cdot, \cdot \rangle$ denotes the usual inner-product of two vectors. The above algorithm monotonically reduces the difference between $\mathbf{v}$ and $\mathbf{u}$ until the rate of the reduction is small. This suggests that the presumed separation rank $r_{\mathbf{u}}$ is not large enough to reduce the error further and thus needs to be increased if the desired accuracy has not been achieved. In general, there is no clear way of knowing the optimal $r_{\mathbf{u}}$ *a priori*. It is possible to start from a low separation rank, e.g., $r_{\mathbf{u}} = 1$, and reduce the representation error using the ALS algorithm and, if necessary, increasing $r_{\mathbf{u}}$ until $\|\mathbf{v} - \mathbf{u}\| \leq \epsilon$ is obtained. The overall procedure is then summarized as:

Separation-rank reduction algorithm:

- Set $r_{\mathbf{u}} = 1$; (randomly) initialize $\{\mathbf{u}_i^1\}$ and $s_1^{\mathbf{u}}$
- Loop while $\|\mathbf{v} - \mathbf{u}\| > \epsilon$
  - Perform the ALS algorithm until $\|\mathbf{v} - \mathbf{u}\|$ decrease only slighgtly
  - Set $r_{\mathbf{u}} = r_{\mathbf{u}} + 1$; (randomly) initialize $\{\mathbf{u}_i^{r_{\mathbf{u}}}\}$ and $s_{r_{\mathbf{u}}}^{\mathbf{u}}$.

It can be shown that one full ALS analysis requires $\mathcal{O}\left(d \cdot r_{\mathbf{u}}(r_{\mathbf{u}}^2 + r_{\mathbf{v}} \cdot M)\right)$ operations.

In theory, as far as $r_{\mathbf{u}}$ is finite and does not depend on d, the complexity of ALS algorithm scales linearly with d. However, in practice, it has been shown that $r_{\mathbf{u}}$ mildly depends on d, and therefore in practical cases the computational cost of the ALS scheme is near-linear with respect to d (Beylkin & Mohlenkamp (2002, 2005)).

In the following, the ALS scheme will be extended to the solution of the linear system of equations arising from spatial-temporal discretization of a stochastic partial differential equation (SPDE).

2.2. *Solution of an SPDE*

Consider a linear partial differential equation with stochastic operator and deterministic input, which consists of finding a stochastic function $u(x,t,\omega) : \bar{\mathcal{D}} \times [0,T] \times \Omega \to \mathbb{R}$, such that the following equation holds almost surely in Ω,

$$\begin{aligned} \mathcal{L}(x,t,\omega;u) &= f(x,t,\omega) \quad (x,t) \in \mathcal{D} \times [0,T] \\ \mathcal{B}(x,t,\omega;u) &= g(x,t) \quad (x,t) \in \partial\mathcal{D} \times [0,T] \\ \mathcal{I}(x,0,\omega;u) &= h(x,\omega) \quad x \in \mathcal{D}, \end{aligned} \tag{2.9}$$

where Ω is the set of elementary events, and $\omega \in \Omega$. The randomness of $\mathcal{L}$ is induced by the uncertainty in the underlying parameters of the corresponding physical system, e.g., heat conductivity, viscosity, etc., and is assumed to be a function of a finite, possibly very large, number of random variables. Such finite-dimensional noise representation can be obtained through, for instance, spectral decomposition of the covariance kernel of the underlying (second-order) random field/processes, which is referred to as Karhunen-Loeve expansion. Therefore, the random operator of Eq. 2.9 can be represented as

$$\mathcal{L}(x,t,\omega;u) = \mathcal{L}\left(x,t,\eta_1(\omega),\cdots,\eta_d(\omega);u\right), \tag{2.10}$$

where $\{\eta_i(\omega)\}_{k=1}^{d}$ are the random variables that define the finite-dimensional noise in the system, e.g., Karhunen-Loeve random variables. Similarly, one can rewrite $f(x,t,\omega) = f\left(x,t,\eta_1(\omega),\cdots,\eta_d(\omega)\right)$ and $h(x,\omega) = h\left(x,\eta_1(\omega),\cdots,\eta_d(\omega)\right)$. Let $\Gamma_k \equiv \eta_k(\Omega)$ be the image of the random variable $\eta_k(\omega)$. The underlying probability space Γ is then the product of images of random variables $\eta_k(\omega)$, i.e., $\Gamma \equiv \prod_{k=1}^{d} \Gamma_k$.

The solution of Eq. 2.9 is a mapping of $\{x,t,\eta_1(\omega),\cdots,\eta_d(\omega)\}$,

$$u(x,t,\omega) := u(x,t,\eta_1(\omega),\cdots,\eta_d(\omega)), \tag{2.11}$$

which is in general non-linear.

Considering a spatial-temporal discretization scheme, the discrete equivalent form of Eq. 2.9 typically simplifies to a solution of a random linear system of equations of the form

$$\mathbb{A}_n(\omega)\mathbf{u}_n(\omega) = \mathbf{f}_n(\omega) \quad \forall n, \tag{2.12}$$

where n denotes the index associated with the time integration scheme. These indices are omitted for the sake of simpler notation henceforth. Considering a tensor product grid, $\mathcal{G}$, of size M^d in Γ, and constructing a separated representation of the form (2.2) for the discrete form of $\mathbb{A}(\omega)$ and $\mathbf{f}(\omega)$ on $\mathcal{G}$, Eq. 2.12 is

$$\left(\sum_{l=1}^{r_{\mathbb{A}}} \mathbb{A}_0^l \otimes \mathbf{A}_1^l \otimes \cdots \otimes \mathbf{A}_d^l\right) \mathbf{u} = \sum_{l=1}^{r_{\mathbf{f}}} \mathbf{f}_0^l \otimes \mathbf{f}_1^l \otimes \cdots \otimes \mathbf{f}_d^l, \tag{2.13}$$

where $\mathbb{A}_0^l \in \mathbb{R}^{N,N}$ and $\mathbf{f}_0^l \in \mathbb{R}^N$ are deterministic matrices and vectors, respectively, whose sizes are determined by the spatial discretization scheme. As specified in Definition 1., $\mathbf{A}_k^l$ and $\mathbf{f}_k^l$ are one-dimensional vectors associated with direction η_k. Finally, $\mathbf{u}$ is the tensor-product representation of the solution to be calculated.

A separated representation of $\mathbf{u}$ with low separation rank is sought to approximate the solution of Eq. 2.13. This can be achieved as in Section 2.1.1 using the ALS algorithm. In particular,

$$\mathbf{u} = \sum_{l=1}^{r_{\mathbf{u}}} s_l^{\mathbf{u}} \mathbf{u}_0 \otimes \mathbf{u}_1^l \otimes \mathbf{u}_2^l \otimes \cdots \otimes \mathbf{u}_d^l \tag{2.14}$$

is searched such that it satisfies

$$\| \sum_{l=1}^{r_{\mathbb{A}}} \mathbb{A}_0^l \otimes \mathbf{A}_1^l \otimes \cdots \otimes \mathbf{A}_d^l \left(\sum_{l=1}^{r_{\mathbf{u}}} s_l^{\mathbf{u}} \mathbf{u}_0 \otimes \mathbf{u}_1^l \otimes \mathbf{u}_2^l \otimes \cdots \otimes \mathbf{u}_d^l\right) - \sum_{l=1}^{r_{\mathbf{f}}} \mathbf{f}_0^l \otimes \mathbf{f}_1^l \otimes \cdots \otimes \mathbf{f}_d^l \| \leq \epsilon \tag{2.15}$$

for some given ϵ. To this end, the separation-rank reduction algorithm of Section 2.1.1 is modified as follows,

Linear system of equations separation-rank reduction algorithm:

- Set $r_{\mathbf{u}} = 1$; (randomly) initialize $\{\mathbf{u}_i^1\}_{i=0}^d$ and thus $s_1^{\mathbf{u}}$.
- Loop while $\|\mathbf{A}\mathbf{u} - \mathbf{f}\| > \epsilon$
  - Perform a modified ALS algorithm, as described below, until $\|\mathbf{A}\mathbf{u} - \mathbf{f}\|$ does not decrease
  - Set $r_{\mathbf{u}} = r_{\mathbf{u}} + 1$; (randomly) initialize $\{\mathbf{u}_i^{r_{\mathbf{u}}}\}_{i=0}^d$ and thus $s_1^{\mathbf{u}}$.

Due to spatial-temporal discretization of Eq. 2.9, one has to solve different normal equations in the alternating least-squares algorithm of Section 2.1.1. The modified normal equations for each direction η_k in the probability space and also the spatial direction, denoted by the subscript 0, are as follows.

Normal equations along direction η_k. Let $\circ$ denote the Hadamard product of two vectors. For each grid point j_k along direction η_k, $\{\mathbf{u}_i^{\tilde{l}}\}_{i=1}^d$ are updated by solving

$$\mathbb{B}_{j_k} \mathbf{c}_{j_k} = \mathbf{b}_{j_k}, \tag{2.16}$$

where

$$\mathbb{B}_{j_k}(\hat{l}, \tilde{l}) = \sum_{l=1}^{r_{\mathbb{A}}} \sum_{\check{l}=1}^{r_{\mathbb{A}}} A_k^l(j_k) A_k^{\check{l}}(j_k) \prod_{i \neq k} \langle \mathbf{A}_i^l \circ \mathbf{u}_i^{\tilde{l}}, \mathbf{A}_i^{\check{l}} \circ \mathbf{u}_i^{\hat{l}} \rangle \tag{2.17}$$

and

$$\mathbf{b}_{j_k}(\hat{l}) = \sum_{l=1}^{r_{\mathbf{f}}} \sum_{\check{l}=1}^{r_{\mathbb{A}}} f_k^l(j_k) A_k^{\check{l}}(j_k) \prod_{i \neq k} \langle \mathbf{f}_i^l, \mathbf{A}_i^{\check{l}} \circ \mathbf{u}_i^{\hat{l}} \rangle. \tag{2.18}$$

More precisely, $s_{\tilde{l}}^{\mathbf{u}} = \left(\sum_j c_{j_k}^2(\tilde{l}) \right)^{1/2}$ and $u_k^{\check{l}}(j_k) = \mathbf{c}_{j_k}(\tilde{l}) / s_{\tilde{l}}^{\mathbf{u}}$ for all $\tilde{l} = 1, \cdots, r_{\mathbf{u}}$.

Normal equation along spatial directions. For the case of updates along the spatial directions, one has to solve one linear system of equations to solve for components of vectors $\mathbf{u}_0^l$ simultaneously, thus meaning that the size of the normal equation is the number of spatial degrees of freedom, N, times the separation rank $r_{\mathbf{u}}$. The quantities $\{\mathbf{u}_0^{\check{l}}\}$ and thus $s_0^{\check{l}}$ are updated by solving the following normal equation,

$$\mathbb{B}_{j_0} \mathbf{c}_{j_0} = \mathbf{b}_{j_0}, \tag{2.19}$$

where $\mathbb{B}_{j_0}$ contains $r_{\mathbf{u}} \times r_{\mathbf{u}}$ blocks of size $N \times N$. The $(\hat{l}, \tilde{l})$-th block is obtained from

$$\mathbb{B}_{j_0,(\hat{l},\tilde{l})}(i_0, j_0) = \sum_{l=1}^{r_{\mathbb{A}}} \sum_{\check{l}=1}^{r_{\mathbb{A}}} \langle \mathbb{A}_{0,j_0}^l, \mathbb{A}_{0,i_0}^{\check{l}} \rangle \prod_{i=1}^{d} \langle \mathbf{A}_i^l \circ \mathbf{u}_i^{\tilde{l}}, \mathbf{A}_i^{\check{l}} \circ \mathbf{u}_i^{\hat{l}} \rangle, \tag{2.20}$$

where $\mathbb{A}_{0,j_0}^l$ denotes the j_0-th column of the $\mathbb{A}_0^l$ matrix. $\mathbf{b}_{j_0}$ contains $r_{\mathbf{u}}$ vectors of size N in which the $\hat{l}$-th one is computed as

$$\mathbf{b}_{j_0,\hat{l}}(j_0) = \sum_{l=1}^{r_{\mathbf{f}}} \sum_{\check{l}=1}^{r_{\mathbb{A}}} \langle \mathbf{f}_0^l, \mathbb{A}_{0,j_0}^{\check{l}} \rangle \prod_{i=1}^{d} \langle \mathbf{f}_i^l, \mathbf{A}_i^{\check{l}} \circ \mathbf{u}_i^{\hat{l}} \rangle. \tag{2.21}$$

It is worthwhile to note that, in theory, the computational cost of the above procedures, similar to those in Section 2.1.1, to obtain a low separation rank of solution $\mathbf{u}$ remains linear with respect to d. However, in practice, $r_{\mathbf{u}}$ depends mildly on d thus increasing the computational cost to more than linear growth.

In the following section, it will be illustrated how one can apply the above procedures for the solution of a stochastic heat equation in one-dimensional physical space but with high-dimensional probability space.

3. Results

Consider the following 1-D stochastic heat equation that holds a.s. in Ω:

$$\begin{aligned} &\frac{\partial u(x,t,\omega)}{\partial t} - \frac{\partial}{\partial x}\left(a(x,\omega) \frac{\partial u(x,t,\omega)}{\partial x} \right) = 0 \qquad (x,t) \in (0,1) \times [0,1] \\ &u(0,t,\omega) = u(1,t,\omega) = 0 \qquad t \in [0,1] \\ &u(x,0,\omega) = h(x,\omega) \qquad x \in (0,1), \end{aligned} \tag{3.1}$$

where the thermal diffusivity coefficient $a(x,\omega)$ is random but a.s. bounded away from zero and also bounded from above. In order to investigate the performance of the proposed algorithm for high-dimensional probability spaces, the following model, similar to that of Xiu & Hesthaven (2005), for a is considered,

$$a(x,\omega) = \bar{a} + \sigma \sum_{k=1}^{d_a} \frac{1}{k^{1/4}\pi^4} \cos(2\pi k x)\eta_k(\omega), \tag{3.2}$$

where $\{\eta_k(\omega)\}_{k=1}^{d_a}$ are *i.i.d.* uniform random variables on $[-1,1]$. Such representation is similar to those obtained from d_a term truncated Karhunen-Loeve expansion of an underlying random field for which the eigenvalues of its covariance kernel decay with the rate of $1/\sqrt{k}$. Moreover, the initial temperature, $h(x,\omega)$, is also assumed to be random given by

$$h(x,\omega) = \sum_{k=1}^{d_h} \sin(k\pi x)\left(\alpha_k + \beta_k \xi_k(\omega)\right), \tag{3.3}$$

with $\{\xi_k(\omega)\}_{k=1}^{d_h}$ being *i.i.d.* uniform random variables on $[-1,1]$ and are independent from $\{\eta_k(\omega)\}_{k=1}^{d_a}$. Having the representations (3.2) and (3.3) for a and h, respectively, the underlying probability space of the problem is readily discretized as required for the procedures of Section 2.2 and has dimension $d \equiv d_a + d_h$. In particular, for the present study, the parameters of the thermal diffusivity coefficient, $\bar{a}$, σ and d_a are chosen to be $\bar{a} = 1$, $\sigma = 0.3$ and $d_a = 40$, respectively. For such a choice of these parameters, one can easily verify that a is both bounded away from zero and bounded from above. The initial temperature is defined by setting $d_h = 2$, $\alpha_1 = 0.80$, $\alpha_2 = 0.16$, $\beta_1 = -0.20$ and $\beta_2 = -0.04$. Therefore, the underlying probability space of the problem has dimension $d = 42$.

3.1. *Discretization schemes*

Spatial-temporal discretization. A standard Euler backward-implicit scheme is utilized for space-time discretization of the heat equation (3.1). Considering a uniform grid of size Δx, here $\Delta x = 0.02$, on $(0,1)$ and time integration size Δt, here $\Delta t = 0.002$, this will lead to

$$(\mathbb{I} + \Delta t \mathbb{C})\,\mathbf{u}_{n+1} = \mathbf{u}_n \qquad n \geq 0, \tag{3.4}$$

with $\mathbf{u}_0(j_0) = \sum_{k=1}^{d_h} \sin(k\pi j_0 \Delta x)\left(\alpha_k + \beta_k \xi_k(\omega)\right)$ and $\mathbf{u}_{n+1}(0) = \mathbf{u}_{n+1}(N+1) = 0$ for all $n \geq 0$. In Eq. 3.4, $\mathbb{I} \in \mathbb{R}^{N,N}$ is the identity matrix and $\mathbb{C} \in \mathbb{R}^{N,N}$ is given by

$$\mathbb{C} = \frac{1}{(\Delta x)^2} \begin{pmatrix} a_1^m + a_1^p & -a_1^p & 0 & \cdots & 0 \\ -a_2^m & a_2^m + a_2^p & -a_2^p & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & -a_{N-1}^m & a_{N-1}^m + a_{N-1}^p & -a_{N-1}^p \\ 0 & \cdots & 0 & -a_N^m & a_N^m + a_N^p \end{pmatrix}, \tag{3.5}$$

in which a_j^m and a_j^p denote $a(j\Delta x - \Delta x/2, \omega)$ and $a(j\Delta x + \Delta x/2, \omega)$, respectively. Replacing the components of $\mathbb{C}$ from Eq. 3.2, $\mathbb{C}$ can be rewritten as

$$\mathbb{C} = \mathbb{C}_0 + \sum_{k=1}^{d_a} \mathbb{C}_k \eta_k(\omega), \tag{3.6}$$

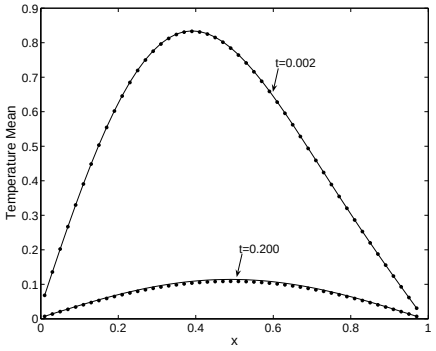

FIGURE 1. Temperature mean at $t = 0.002$ and $t = 0.200$ (Monte Carlo simulation ——; ALS algorithm with $r_{\mathbf{u}} = 2$ ● ● ●).

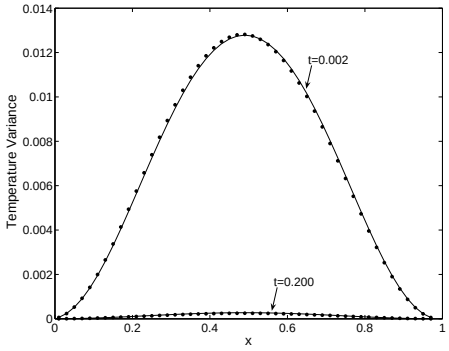

FIGURE 2. Temperature variance at $t = 0.002$ and $t = 0.200$ (Monte Carlo simulation ——; ALS algorithm with $r_{\mathbf{u}} = 2$ ● ● ●).

in which $\mathbb{C}_0$ and $\mathbb{C}_k$ correspond to the contribution of mean of thermal diffusivity $\bar{a}$ and the coefficients of $\eta_k(\omega)$, in the representation (3.2), respectively.

Discretization of probability space. In order to implement the ALS algorithm of Section 2.2, one has to define a grid $\mathcal{G}$ in Γ. In the present study, a uniform grid of size $M = 5$ in any direction η_k or ξ_k is considered. Accordingly, the (discrete) separated representation of the problem (3.4) is given by

$$\left(\sum_{l=1}^{d_a+1} \mathbb{A}_0^l \otimes \mathbf{A}_1^l \otimes \cdots \otimes \mathbf{A}_d^l\right) \mathbf{u}_{n+1} = \sum_{l=1}^{r_{\mathbf{u}_n}} \mathbf{u}_0^l \otimes \mathbf{u}_1^l \otimes \cdots \otimes \mathbf{u}_d^l, \tag{3.7}$$

where $\mathbb{A}_0^1 = \mathbb{I} + \Delta t \mathbb{C}_0$ and, for $l = 2, \cdots, d_a + 1$, $\mathbb{A}_0^l = \Delta t \mathbb{C}_{l-1}$. Also, $A_k^1(j_k) = 1$ and, for $l = 2, \cdots, d_a + 1$, $A_k^l(j_k) = -1 + (j_k - 1)/5$ if $k = l - 1$ and $A_k^l(j_k) = 1$ otherwise. Moreover, the right-hand side of Eq. 3.7 is the separated representation of $\mathbf{u}_n$. For the case of $n = 0$ (t=0), $r_{\mathbf{u}_0} = 3$, $u_0^1(j_0) = \sum_{k=1}^{d_h} \alpha_k \sin(k\pi j_0 \Delta x)$, and, for $l = 2, \cdots, d_h + 1$, $u_0^l(j_0) = \beta_{l-1} \sin((l-1)\pi j_0 \Delta x)$ $u_k^1(j_k) = 1$. Finally, $u_k^1(j_k) = 1$ and, for $l = 2, \cdots, d_h + 1$, $u_k^l(j_k) = -1 + (j_k - 1)/5$ if $k = l - 1$ and $u_k^l(j_k) = 1$ otherwise.

To verify the proposed procedures, a Monte Carlo simulation on an identical problem (as defined above) is performed. The quantities of interest are mean and variance of u in time as well as the probability distribution of the $u(x = 0.5, t)$ at the first, $t = 0.002$, and last time step, $t = 0.200$, of the analysis, which are compared in Figs. 1–4. For all time steps of the analysis, a rank 2, $r_{\mathbf{u}} = 2$, approximation of $\mathbf{u}$ was sufficient to achieve a target accuracy of $\epsilon = 1 \times 10^{-2}$. As is seen from Figs. 1–4 the algorithm has an acceptable agreement with predictions from the Monte Carlo simulation. The impact of long time integration on the proposed algorithm is illustrated in Fig. 4. As is expected, the accuracy of the scheme decreases as it moves forward in time due to an accumulation of approximation error ϵ at each time step and propagating to proceeding time steps based on relation 3.4.

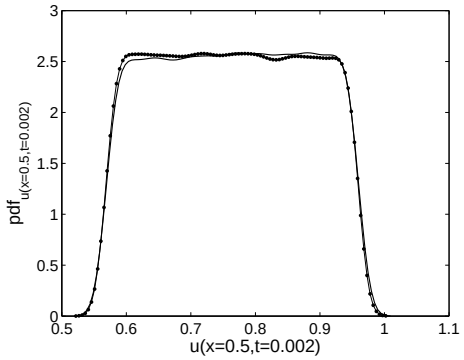

FIGURE 3. Probability density function of temperature at $x = 0.5$ and $t = 0.002$ (Monte Carlo simulation ——— ; ALS algorithm —●—)

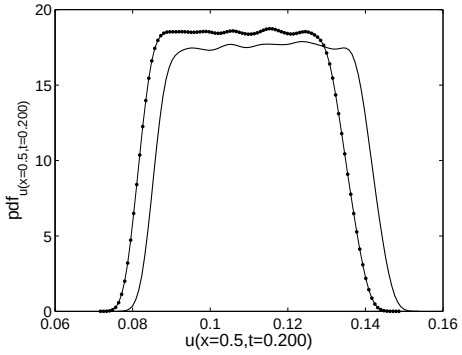

FIGURE 4. Probability density function of temperature at $x = 0.5$ and $t = 0.200$ (Monte Carlo simulation ——— ; ALS algorithm —●—)

4. Conclusion and future directions

The work of Beylkin & Mohlenkamp (2002, 2005) has been extended here to approximation of high-dimensional stochastic partial differential equations. The method is based on an alternating least-squares (ALS) algorithm that minimizes the difference between the quantity to be estimated and a general separation-rank representation. This is considered to be a generalization of the singular value decomposition technique without being optimal. The minimum separation-rank approximation is achieved as a result of the ALS implementation. It can be shown that the number of the required operations, and hence the computational cost, for the proposed scheme is linear with respect to the dimension of the underlying probability space in which the solution exists. The ALS algorithm is implemented for the solution of a 1-D (in spatial domain) heat equation with uncertain thermal diffusivity as well as initial temperature. The total number of random variables defining the underlying probability space is 42. Reasonable agreement between the Monte Carlo simulation and that of the ALS algorithm with only rank 2 approximation, $r_{\mathbf{u}} = 2$, has been observed to achieve an accuracy of $\epsilon = 1 \times 10^{-2}$.

The effect of long time integration has been observed in the analysis. The target accuracy, ϵ, of the ALS algorithm can be adjusted *a priori* based on a desirable accuracy at the final analysis time. However, this might require a relatively small ϵ to begin with, thus making the proposed procedures less efficient. The issue of long time integration therefore requires further attention.

5. Acknowledgments

This work was supported by the United States Department of Energy under the Advanced Simulation and Computing (ASC) program at Stanford University.

REFERENCES

BABUŠKA, I., NOBILE, F. & TEMPONE, R. 2007 A stochastic collocation method for elliptic partial differential equations with random input data. *SIAM Journal on Numerical Analysis* **45** 1005–1034.

BABUŠKA, I., TEMPONE, R. & ZOURARIS, G. 2004 Galerkin finite element approxima-

tions of stochastic elliptic partial differential equations. *SIAM Journal on Numerical Analysis* **42** 800–825.

Beylkin, G. & Mohlenkamp, M. J. 2002 Numerical operator calculus in higher dimensions. *Proceedings of the National Academy of Science* **99** 10246–10251.

Beylkin, G. & Mohlenkamp, M. J. 2005 Algorithms for numerical analysis in high dimensions. *SIAM J. Sci. Comput.* **26** 2133–2159.

Doostan, A., Ghanem, R. & Red-Horse, J. 2007 Stochastic model reduction for chaos representations. *Computer Methods in Applied Mechanics and Engineering* **196**, 3951–3966.

Ghanem, R., Saad, G. & Doostan, A. 2007 Efficient solution of stochastic systems: Application to the embankment dam problem. *Structural Safety* **29** 238–251.

Ghanem, R. & Spanos, P. 2003 *Stochastic Finite Elements: A Spectral Approach.* Dover.

Kleiber, M. & Hien, T. D. 1992 *The Stochastic Finite Element Method: Basic Perturbation Technique and Computer Implementation.* John Wiley & Son, New York.

Nobile, F., Tempone, R. & Webster, C. 2006 An sparse grid stochastic collocation method for elliptic partial differential equations with random input data. *Technical Report #85, MOX, Dipartimento di Matematica, Politencnico di Milano, Italy.*

Nobile, F., Tempone, R. & Webster, C. 2007 An anisotropic sparse grid stochastic collocation method for elliptic partial differential equations with random input data. *Technical Report #04/2007, MOX, Dipartimento di Matematica, Politencnico di Milano, Italy.*

Nouy, A. 2007 A generalized spectral decomposition technique to solve a class of linear stochastic partial differential equations. *Computer Methods in Applied Mechanics and Engineering.* **196** 4521–4537.

Pereyra, V. & Scherer, G. 1973 Efficient computer manipulation of tensor products with applications to multidimensional approximation. *Math. Comput.* **27** 595–605.

Xiu, D. & Hesthaven, J. S. 2005 High-order collocation methods for differential equations with random inputs. *SIAM J. Sci. Comput.* **27** 1118–1139.

Xiu, D. & Karniadakis, G. 2002 The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.* **24** 619–644.