ITERATIVE SOLVERS FOR A SPECTRAL GALERKIN APPROACH TO ELLIPTIC PARTIAL DIFFERENTIAL EQUATIONS WITH FUZZY COEFFICIENTS∗

SAMUEL CORVELEYN†, EVELINE ROSSEEL‡, AND STEFAN VANDEWALLE†

Abstract. Mathematical models of physical systems often contain parameters with an imprecisely known and uncertain character. It is quite common to represent these parameters by means of random variables. Numerous methods have been developed to compute accurate approximations to solutions of equations with such parameters. This approach, however, may not be entirely justified when the uncertainty is due to vagueness or incomplete knowledge. For such cases, alternative uncertainty representations using tools from imprecise probability theory have been suggested. Among those, the fuzzy representation is probably most popular. In this paper, we consider numerical methods for solving partial differential equations with fuzzy coefficients. We demonstrate that spectral expansion methods, quite common in the random variable approach, can also be used effectively for solving fuzzy equations. We motivate the use of Chebyshev polynomials in the spectral representation and apply a Galerkin projection to convert the fuzzy problem into a high-dimensional deterministic one. Two preconditioners are proposed in order to efficiently solve the resulting high-dimensional algebraic system. A Fourier analysis demonstrates that both preconditioners yield a convergence rate that is independent of the spatial resolution and independent of the number of fuzzy variables and the polynomial order. The practical applicability of the algorithm is illustrated by means of two numerical experiments: a fuzzy heat transfer problem on an L-shaped domain and a fuzzy elasticity problem.

Key words. fuzzy partial differential equations, epistemic uncertainty, spectral method, polynomial chaos, Chebyshev polynomials, multigrid, Fourier analysis

AMS subject classifications. 35R13, 65F08, 65N22, 65N35, 65N55

DOI. 10.1137/120881592

1. Introduction. Mathematical models in science and engineering often contain uncertain parameters, such as the material properties of an inhomogeneous medium or the geometric size of an object that is manufactured to specific engineering tolerances. When the uncertainty is due to an inherently random underlying physical process or characteristic, it is called irreducible and aleatoric. Such parameters are typically represented as random variables, fields, or processes with a precisely defined probability structure. In many practical applications, however, the uncertainty has a different character. For example, in the early stages of an engineering design one may have only some vague idea about the material properties or dimensions that will be used in the final product. Such uncertainties, which arise due to lack of knowledge, imprecision, or vagueness, are called epistemic [33, 28].

Whether or not probability theory is adequate or sufficient for the modeling of epistemic uncertainties is a matter of debate [35, 12, 17]. Laplace’s principle of insufficient reason, for example, states that an uncertain parameter for which only bounds are known (i.e., an interval parameter) should be modeled by a uniform prior distribution function in a Bayesian updating framework. Against this reasoning, some

∗Received by the editors June 18, 2012; accepted for publication (in revised form) July 8, 2013; published electronically October 28, 2013. http://www.siam.org/journals/sisc/35-5/88159.html
†Department of Computer Science, KU Leuven—University of Leuven, Celestijnenlaan 200A, 3001 Heverlee, Belgium (samuel.corveleyn@cs.kuleuven.be, eveline.rosseel@cs.kuleuven.be, stefan.vandewalle@cs.kuleuven.be). Work supported in part by the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen).

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.
argue that one should work with the information that is available, instead of adding information which could lead to wrong conclusions in engineering applications. In this light, “uncertainty” is seen as the lack of information or the type of information one has of a parameter [21].

Over the last decades, many uncertainty models have been proposed. Some attempts have also been made at unifying them into a general theory of uncertainty. The theory of imprecise probabilities [47] is such a unifying framework. This framework encompasses interval and fuzzy uncertainties, possibility theory, Dempster–Shafer theory, and also probability theory itself. All of those theories lead to a definition of upper and lower probabilities, also called interval probabilities [49]. Some more general models which do not necessarily allow a description by interval probabilities, like probability boxes and credal sets, also fit in the imprecise probability framework.

Here, we will work with one of the more restrictive uncertainty models, i.e., the fuzzy set theory. In particular, this paper will deal with the numerical solution of an elliptic partial differential equation (PDE) with fuzzy coefficients. Such equations have been extensively applied in the engineering literature; see, e.g., [26, 27, 1, 8, 31]. As such, we refer to that literature for motivating examples, for a discussion on the modeling aspects, and for a critical comparison to the alternative uncertainty models. The focus of the present paper will instead be on the computational aspects. We will demonstrate that such equations can be solved efficiently and accurately.

A first difficulty that arises when dealing with fuzzy PDEs is that there is no unanimity in the literature on their exact definition. Quite a few alternative (though related) interpretations exist; see, e.g., the discussions in [5, 24, 32, 46]. We will choose the interpretation using sample path-based fuzzy fields [6, 10, 23] and Zadeh’s extension principle [52], a core principle in fuzzy set theory. It will be recalled in section 2 together with some necessary background material on fuzzy sets, numbers, fields, and equations. Using this approach, the solution to a fuzzy PDE is defined as the fuzzified solution of a corresponding deterministic parametric PDE.

For the assessment of the accuracy of a numerical fuzzy solution, we use the supremum distance [9]. As we will show, this naturally leads to the requirement of accuracy over the parameter domain of the numerical solution to the parametric PDE in the $L^\infty$-norm. The importance of $L^\infty$-accuracy has been recognized quite a few times in the case of epistemic uncertainty; see, for example, [18, 13, 15]. This is in contrast with the stochastic case [51, 3, 2], where weighted $L^2$ accuracy w.r.t. the probability density function is needed. Of the solvers that are used to achieve $L^\infty$-accuracy, sparse grid collocation is often preferred because it is nonintrusive and has a high accuracy when the parametric solution is smooth over the parameter domain [41, 48, 18, 13]. In [18, 48], the spectral Galerkin method is briefly discussed. The Galerkin method minimizes the residual error in a weighted $L^2$-norm. As such, high accuracy in the $L^\infty$-norm is not immediate, although it can be expected in case of a smooth parametric solution.

This paper will focus on a spectral Galerkin method with Chebyshev polynomials in the parameter domain. When combined with a classical spatial discretization, the spectral Galerkin approach transforms the parametric PDE into a high-dimensional linear algebraic system. Based on ideas from the literature on probabilistic coefficient PDEs, we will propose two preconditioners for solving those algebraic systems. The first one will be the fuzzy analogue to the well-known mean-based preconditioner [34], whereas the second one will be of multigrid type and related to the algorithms in [14, 22, 38]. The convergence of these preconditioners will be analyzed by means of a local Fourier mode analysis, which will take the specific properties of the Chebyshev polynomials into account.
The remainder of this paper is organized as follows. Section 2 introduces the concept of a fuzzy PDE and describes the steps of the solution process. In section 3, the spectral Galerkin method for fuzzy PDEs is presented. Section 4 elaborates on the two preconditioners for efficiently solving the high-dimensional algebraic problems resulting from the Galerkin discretization. By means of a local Fourier analysis (LFA), their convergence properties are analyzed in section 5. It will be shown that the convergence is independent of the discretization parameters, such as the size of the spatial mesh, the dimension of the fuzzy parameter space, and the polynomial order. Robustness against the magnitude of the input uncertainty will also be investigated. These convergence results are verified by a set of model problem numerical experiments. The accuracy of the spectral Galerkin approximation and the efficiency of the proposed iterative solvers is numerically demonstrated on two nontrivial problems in section 6. Section 7 summarizes the main conclusions of this paper.

2. Fuzzy elliptic PDEs. We will start this section by recalling some necessary background material and basic concepts from fuzzy set theory. Next, we will state the model problem considered in this paper and elaborate on the concept of a solution to a fuzzy PDE. We conclude with a discussion on a distance function for fuzzy sets and why it leads to the requirement of $L^\infty$-accuracy. For a more thorough treatise on fuzzy sets and fuzzy numbers, we refer to [19, 35].

2.1. Definition of fuzzy sets, fuzzy numbers, and fuzzy fields. Fuzzy sets generalize the concept of classical sets. Classical sets either contain an element or not. This is reflected in the fact that the indicator function of a classical set only takes values 0 or 1. Fuzzy sets, on the other hand, allow any degree of membership between 0 and 1. As such, a fuzzy set is represented by a membership function that can take any value in the interval $[0, 1]$.

Definition 2.1 (fuzzy set, membership function). A fuzzy set $\tilde{a}$ is a pair $(V, \mu_{\tilde{a}})$, where $V$ is a set and $\mu_{\tilde{a}} : V \to [0, 1]$ is a membership function.

Fuzzy sets will be denoted by the superimposition of a tilde on the set name. The set of all fuzzy sets over $V$ will be denoted by $\mathcal{F}(V)$.

There are different possible interpretations of what the membership degree represents. Initially, fuzzy sets were introduced to better represent vague linguistic knowledge in logic [52]. The membership degree then represents a degree of similarity, of preference, of acceptability, of suitability, etc. It was later recognized, however, that the membership degree can also represent a degree of possibility [53] in possibility theory. We refer to [11, 12] for an extensive discussion on the semantics of fuzzy sets and the interpretation of the membership degree.

An important notion in fuzzy set theory is that of $\alpha$-cuts.

Definition 2.2 ($\alpha$-cut). The $\alpha$-cuts of a fuzzy set $\tilde{a} = (V, \mu_{\tilde{a}})$ are defined as

$$[\tilde{a}]_\alpha := \begin{cases} \{a \in V : \mu_{\tilde{a}}(a) \geq \alpha\} & \alpha \in (0, 1], \\ \text{closure} \left( \bigcup_{\alpha \in (0,1)} [\tilde{a}]_\alpha \right) & \alpha = 0. \end{cases}$$

The $\alpha = 0$ cut and $\alpha = 1$ cut are the so-called support and core of the fuzzy set.

In this paper, we will work with a special class of fuzzy sets, called fuzzy numbers.

Definition 2.3 (fuzzy number). A fuzzy number $\tilde{a}$ is a fuzzy set over $\mathbb{R}$ for which the following conditions hold:

- $[\tilde{a}]_1$ is nonempty (normality),
- $[\tilde{a}]_\alpha$ is compact for all $\alpha \in [0, 1]$ (compactness).
The definition of functions of fuzzy numbers, or more generally functions of fuzzy sets, is based on a very fundamental axiom in fuzzy set theory: \textit{Zadeh’s extension principle}. This principle defines how a map \( f: V \rightarrow W \) operating on deterministic elements of \( V \) should be extended toward a map \( f: \mathcal{F}(V) \rightarrow \mathcal{F}(W) \), operating on fuzzy sets in \( \mathcal{F}(V) \). It is the so-called fuzzification of that map.

\textbf{Definition 2.4 (Zadeh’s extension principle).} If \( f: V \rightarrow W \), then \( f: \mathcal{F}(V) \rightarrow \mathcal{F}(W) \) is defined by \( f: \tilde{a} = (V, \mu_a) \rightarrow f(\tilde{a}) = (W, \mu_{f(\tilde{a})}) \) with

\begin{equation}
\mu_{f(\tilde{a})}(z) = \begin{cases} 
\sup_{a \in f^{-1}(z)} \mu_a(a) & \text{if } f^{-1}(z) \neq \emptyset, \\
0 & \text{if } f^{-1}(z) = \emptyset.
\end{cases}
\end{equation}

The direct application of the extension principle to compute a function of a fuzzy set numerically can be quite cumbersome. Fortunately, such a computation can be simplified considerably in many cases. It is proved in [30] that the \( \alpha \)-cuts \( [f(\tilde{a})]_\alpha \) are equal to \( f([\tilde{a}]_\alpha) \) if \( f \) is continuous and \( \tilde{a} \) is normal and compact. As such, the theorem below holds for fuzzy numbers and shows that the computation of a function of a fuzzy number is equivalent to a sequence of interval computations on successive \( \alpha \)-cuts.

\textbf{Theorem 2.1 (\( \alpha \)-cut approach).} Let \( f: V \rightarrow W \) be a continuous map, and \( \tilde{a} \in \mathcal{F}(V) \) a compact normal fuzzy set. Then, the Zadeh extension of \( f \) satisfies

\begin{equation}
[f(\tilde{a})]_\alpha = f([\tilde{a}]_\alpha).
\end{equation}

This further simplifies to

\begin{equation}
[f(\tilde{a})]_\alpha = \left[ \min_{a \in [\tilde{a}]_\alpha} f(a), \max_{a \in [\tilde{a}]_\alpha} f(a) \right]
\end{equation}

when \( W := \mathbb{R} \).

The extension principle also applies to the fuzzification of functions of several arguments. The situation is, however, somewhat more complex because the dependency structure between the arguments has to be taken into account. This mutual dependency between two or more fuzzy sets is defined by a joint membership function. In fuzzy set terminology, this dependency is called interactivity.

\textbf{Definition 2.5 (interactivity, joint membership function).} Let \( \tilde{a} \in \mathcal{F}(V) \) and \( \tilde{b} \in \mathcal{F}(W) \), and then the interactivity of \( \tilde{a} \) and \( \tilde{b} \) is defined by the joint membership function \( \mu_{\tilde{a}, \tilde{b}}: V \times W \rightarrow [0, 1] \).

It is often assumed in fuzzy modeling that the different parameters are noninteractive.

\textbf{Definition 2.6 (noninteractivity).} Two fuzzy sets \( \tilde{a} \in \mathcal{F}(V) \) and \( \tilde{b} \in \mathcal{F}(W) \) are said to be noninteractive, i.e., independent, if \( \mu_{\tilde{a}, \tilde{b}}(a, b) = \min(\mu_{\tilde{a}}(a), \mu_{\tilde{b}}(b)) \).

An important consequence of noninteractivity which we will need later is that for noninteractive fuzzy numbers \( \xi_1, \ldots, \xi_d \in \mathcal{F}(\mathbb{R}) \), the \( \alpha \)-cuts of \( \xi := (\xi_1, \ldots, \xi_d) \) are hyperrectangles, i.e., \( [\xi]_\alpha = [\xi_1]_\alpha \times \cdots \times [\xi_d]_\alpha \), for all \( \alpha \in [0, 1] \).

Important for the definition and interpretation of fuzzy PDEs is the concept of a fuzzy field. A scalar fuzzy field \( \tilde{a} \) is a fuzzy set which varies over a spatial coordinate \( x \in \Omega \subset \mathbb{R}^d \), i.e., \( \tilde{a}: \Omega \rightarrow \mathcal{F}(\mathbb{R}) \). While this is a useful intuitive definition, it does not provide direct information about the interactivity in the fuzzy field, i.e., the mutual interactivity of \( \tilde{a}(x) \) for different \( x \in \Omega \). This interactivity is, however, essential in
the fuzzification process of differential operators, and hence in fuzzy PDEs. The joint membership function \( \mu_a(x) \) for all \( x \in \Omega \) which assigns a degree of membership to different realizations \( a \) of the fuzzy field \( a \) is therefore much more convenient for defining a fuzzy field in the context of fuzzy PDEs.

**Definition 2.7** (fuzzy field). A fuzzy field \( a \) over the spatial domain \( \Omega \subset \mathbb{R}^{d_a} \) is a fuzzy set over a function space \( V \) of functions defined on \( \Omega \), i.e., \( a \in \mathcal{F}(V) \).

### 2.2. The fuzzy elliptic model problem and its solution.

In this paper, we shall concentrate on an elliptic PDE defined on a \( d_\Omega \)-dimensional noise assumption for the fuzzy input field \( a \).

**Theorem 2.3** (extension principle). Using the solution operator

\[
\tilde{a}(x) \in \mathcal{F}(L^\infty(\Omega)) \text{ the fuzzy input field, and } \tilde{u} \in \mathcal{F}(H^1_0(\Omega)) \text{ the unknown fuzzy solution field.}
\]

Finally, in order to ensure the problem is well-posed, we assume that all \( a \in [\bar{a}]_0 \) are strictly positive, i.e., there exists an \( \epsilon > 0 \) for which \( a(x) \geq \epsilon \) almost everywhere for all \( a \in [\bar{a}]_0 \).

Following the interpretation of fuzzy PDEs by Zadeh’s extension principle, the fuzzy PDE is defined as the fuzzification of the parametric PDE

\[
-\nabla \cdot (a(x)\nabla u(x,a)) = f(x) \quad \text{in } \Omega
\]

(4)

with \( f \in L^2(\Omega) \) the deterministic source term, \( a \in \mathcal{F}(L^\infty(\Omega)) \) the fuzzy input field, and \( \tilde{u} \in \mathcal{F}(H^1_0(\Omega)) \) the unknown fuzzy solution field. Using the solution operator \( S: L^\infty(\Omega) \ni [\bar{a}]_0 \rightarrow H^1_0(\Omega) \): \( a \rightarrow u(\cdot,a) \) this translates to the following.

**Definition 2.8** (solution to fuzzy PDE). The solution \( \tilde{u} \) to the fuzzy PDE (4) is the fuzzy field obtained through fuzzification of the solution operator \( S \) of the corresponding deterministic parametric PDE (5), i.e., \( \tilde{u} = S(\bar{a}) \), with \( \bar{a} \) the fuzzy parameter.

The fuzzy field \( \tilde{u} \) thus contains all possible realizations of the form \( S(a) \) with \( a \in [\bar{a}]_0 \), and each of those realizations is equipped with a membership level computed according to the extension principle (1).

#### 2.3. Finite-dimensional noise assumption.

We will now introduce a finite-dimensional noise assumption for the fuzzy input field \( \bar{a} \). It resembles the typical approximation of stochastic fields in PDEs with random coefficients by a truncated Karhunen–Loève expansion. That is, we will consider a fuzzy input field that is (or has been approximated by) a finite expansion of the form

\[
\tilde{a}(x) = a_0(x) + \sigma \sum_{k=1}^{d_\xi} a_k(x) \tilde{\xi}_k,
\]

(6)

where \( \tilde{\xi}_1, \ldots, \tilde{\xi}_{d_\xi} \) are assumed to be noninteractive fuzzy numbers and have a support equal to \([-1,1]\). Function \( a_0 \in L^\infty(\Omega) \) represents the main component of the fuzzy field \( \bar{a} \), while the other terms with \( a_k \in L^\infty(\Omega) \), \( k = 1, \ldots, d_\xi \), express the fuzzy variations of the field. The scaling factor \( \sigma \) is a positive scalar which will be used in the numerical results section to vary the magnitude of the fuzzy variation.

The parameterized representation of (4) now becomes

\[
-\nabla \cdot (a(x,\xi)\nabla u(x,\xi)) = f(x) \quad \text{in } \Omega
\]

(7)

with parameter \( \xi := (\tilde{\xi}_1, \ldots, \tilde{\xi}_{d_\xi}) \), parameter domain \( \Xi := [\bar{\xi}]_0 = [-1,1]^{d_\xi} \), and solution operator \( S: \mathbb{R}^{d_\xi} \ni \xi \rightarrow H^1_0(\Omega): \xi \mapsto u(\cdot,\xi) \). The specific form (6) of the
fuzzy input field combined with the assumption that all \( a \in [\tilde{a}]_0 \) are strictly positive ensures that \( \xi \mapsto u(\cdot, \xi) \) is analytic in a domain strictly larger than \( \Xi \) \([7]\). This is the main reason why spectral methods in the parameter domain are popular for solving this equation.

Remark. The modeling of fuzzy fields, or, more generally, epistemically uncertain fields, is a topic of intense, ongoing research; see, e.g., \([29, 25, 45, 39, 16]\) and the references therein. A discussion on this modeling aspect is outside the scope of this paper. In our experiments we will use artificially constructed fields. The model will in fact be constructed from a truncated Karhunen–Loève expansion of a stochastic Gaussian field through the replacement of the random numbers with the fuzzy numbers \( \xi_1, \ldots, \xi_d \). The experiments will demonstrate the robustness of our method against different choices for the field.

2.4. Distance function for fuzzy sets. In order to measure the accuracy of a numerical approximation of the solution \( \tilde{u} \) of (4), we need a distance function. We will use the so-called supremum distance \( d_\infty \) for fuzzy sets \([9]\).

Definition 2.9 (supremum distance). Let \( d_H \) be the Hausdorff distance on the metric space \( W \). The supremum distance \( d_\infty \) on \( \mathcal{F}(W) \) is then defined by

\[
d_\infty(\tilde{u}, \tilde{v}) := \sup_{0 < \alpha \leq 1} d_H([\tilde{u}]_\alpha, [\tilde{v}]_\alpha).
\]

Computing that distance can be very hard in practice. For fuzzy sets which are defined as \( \tilde{u} := u(\tilde{\xi}) \) and \( \tilde{v} := v(\tilde{\xi}) \) with \( u, v: [\tilde{\xi}]_0 \to W \) continuous functions, we can derive the following upper bound.

Theorem 2.2. Let \( \tilde{\xi} \in \mathcal{F}(V) \) be a compact normal fuzzy set over some topological space \( V \), and \( W \) a metric space with metric \( d \). If \( u: [\tilde{\xi}]_0 \to W \) and \( v: [\tilde{\xi}]_0 \to W \) are continuous maps, then

\[
d_\infty(\tilde{u}, \tilde{v}) \leq \sup_{\xi \in [\tilde{\xi}]_0} d(u(\xi), v(\xi)),
\]

where \( \tilde{u} := u(\tilde{\xi}) \) and \( \tilde{v} := v(\tilde{\xi}) \).

Proof. Combining Definition 2.9 and Theorem 2.1, we get

\[
d_\infty(u(\tilde{\xi}), v(\tilde{\xi})) = \sup_{0 < \alpha \leq 1} d_H(u([\tilde{\xi}]_\alpha), v([\tilde{\xi}]_\alpha)).
\]

By definition of the Hausdorff distance, \( d_H(u([\tilde{\xi}]_\alpha), v([\tilde{\xi}]_\alpha)) \) is equal to

\[
\max \left\{ \sup_{\xi_1 \in [\tilde{\xi}]_0} \inf_{\xi_2 \in [\tilde{\xi}]_0} d(u(\xi_1), v(\xi_2)), \sup_{\xi_2 \in [\tilde{\xi}]_0} \inf_{\xi_1 \in [\tilde{\xi}]_0} d(u(\xi_1), v(\xi_2)) \right\}.
\]

This can be bounded as

\[
d_H(u([\tilde{\xi}]_\alpha), v([\tilde{\xi}]_\alpha)) \leq \sup_{\xi \in [\tilde{\xi}]_0} d(u(\xi), v(\xi)) \leq \sup_{\xi \in [\tilde{\xi}]_0} d(u(\xi), v(\xi)).
\]

Together with (10), this completes the proof. \( \square \)

Corollary 2.1. Under the same assumptions as in Theorem 2.2, \( V := \mathbb{R}^d \) with \( d \) a positive integer, and \( W \) a Banach space, we have

\[
d_\infty(\tilde{u}, \tilde{v}) \leq \| u - v \|_{L^\infty([\tilde{\xi}]_0; W)}.
\]
The use of the $L^\infty$-norm is somewhat artificial here, but because the functions are continuous, the $L^\infty$-norm (i.e., the essential supremum norm) is equal to the sup-norm. The $L^\infty$-norm was chosen to be in line with the existing literature about epistemic modeling.

3. Discretization of the parameter space and of the physical space.

3.1. A response surface approach. For engineering applications, one may be interested in computing the pointwise values of the fuzzy field $\tilde{u}$ at different nodes of a spatial mesh. This can be done by the $\alpha$-cut approach described in Theorem 2.1 using the exact solution operator $S$ of a spatially discretized version of (7). This amounts to a minimization and maximization procedure for each $\alpha$-level considered, as well as for each grid point at which the fuzzy solution is required. Due to the cost of evaluating $S$ and due to the possibly huge number of optimization problems, the above procedure rapidly becomes prohibitively expensive. In order to reduce the computational cost, it is therefore common to construct an approximating response surface over the parameter domain $\Xi$, during a preprocessing stage [1, 8, 20]. The fuzzy solution $\tilde{u}$ is then approximated by $\tilde{u}^r$, which is in its parameterized form given by

\begin{equation}
\tilde{u}^r(x, \xi) = \sum_{j=1}^{n_r} u^r_j(x) r_j(\xi),
\end{equation}

where $\{r_j\}_{j=1}^{n_r}$ is a set of given basis functions and $\{u^r_j\}_{j=1}^{n_r}$ the set of coefficients. Such a response surface can be constructed by interpolation, although other criteria are equally possible. Here, we will construct a response surface for model problem (4) by a Galerkin approach.

Once constructed, the response surface can be used as a cheap alternative to the expensive solution operator $S$ to compute different quantities of interest, such as pointwise evaluations or more complicated functionals of $\tilde{u}$. Assuming that both $\xi \mapsto u(\cdot, \xi)$ and $\xi \mapsto u^r(\cdot, \xi)$ are $W$-valued continuous functions over $\Xi$, with $W$ some Banach space, and assuming that the quantity of interest is represented by a continuous linear functional $L \in W^*$, we can compute $L(\tilde{u}^r)$ by the $\alpha$-cut approach (see Theorem 2.1):

\begin{equation}
[L(\tilde{u}^r)]_\alpha = \left[ \min_{\xi \in [\xi_\alpha]} L(u^r(\cdot, \xi)), \max_{\xi \in [\xi_\alpha]} L(u^r(\cdot, \xi)) \right].
\end{equation}

The error introduced by the use of the response surface instead of the exact solution operator can be estimated using Corollary 2.1:

\begin{align}
d_\infty(L(\tilde{u})), L(\tilde{u}^r)) &\leq \|L(u(\cdot, \xi)) - L(u^r(\cdot, \xi))\|_{L^\infty(\Xi)} \\
&\leq \|L\|_{W^*} \|u - u^r\|_{L^\infty(\Xi; W)}.
\end{align}

3.2. Spectral Galerkin approximation. The Galerkin approximation to the fuzzy solution starts from the weak formulation of the parameterized problem (7). First, we define the space of all square integrable functions $L^2_{w}(\Xi)$ for a weighting function $w$. The weak formulation of (7) then reads as follows: find $u \in H^1_0(\Omega) \otimes L^2_{w}(\Xi)$ such that for all $v \in H^1_0(\Omega) \otimes L^2_{w}(\Xi)$,

\begin{equation}
\int_{\Xi} \int_{\Omega} a(x, \xi) \nabla u(x, \xi) \cdot \nabla v(x, \xi) w(\xi) \mathrm{d}x \mathrm{d}\xi = \int_{\Xi} \int_{\Omega} f(x) v(x, \xi) w(\xi) \mathrm{d}x \mathrm{d}\xi.
\end{equation}
Next, we construct a discrete formulation of (16) by introducing finite-dimensional subspaces. For the spatial domain, we introduce a space \( X_k \subset H^1_0(\Omega) \) of standard Lagrange finite element functions on a triangulation of \( \Omega \). This space is spanned by a set of \( n_\phi \) basis functions \( \{ \phi_i \}_{i=1}^{n_\phi} \), yielding \( n_\phi \) spatial degrees of freedom. A finite-dimensional space \( Y_p \subset L^2_0(\Xi) \) is constructed from a multivariate orthogonal polynomial basis of total degree \( p \). Given \( d_\Xi \) fuzzy numbers, the space \( Y_p \) is spanned by a set of \( n_\psi = (d_\Xi + p)!/(d_\Xi)!p! \) orthogonal polynomials \( \{ \psi_j \}_{j=1}^{n_\psi} \). We then look for a Galerkin representation \( u^n_{\psi,n_\phi}(x) \) of the response surface \( u^\gamma \) (12) in the form

\[
(17) \quad u^n_{\psi,n_\phi}(x, \xi) = \sum_{j=1}^{n_\psi} \sum_{i=1}^{n_\phi} u_{i,j} \phi_i(x) \psi_j(\xi).
\]

Discretizing (16) with \( a(x, \xi) \) represented by (6), we arrive at the algebraic matrix formulation

\[
(18) \quad \left( G_0 \otimes A_0 + \sum_{k=1}^{d_\Xi} G_k \otimes A_k \right) \mathbf{u} = \mathbf{g} \otimes \mathbf{f}.
\]

Here, \( G_0 \) equals the identity matrix \( I_{n_\psi} \), and

\[
(19) \quad [G_k]_{i,j} = \int_{\Xi} \xi_k \psi_i(\xi) \psi_j(\xi) w(\xi) \, d\xi, \quad i, j = 1, \ldots, n_\psi, k = 1, \ldots, d_\Xi,
\]

\[
(20) \quad [A_0]_{i,j} = \int_{\Omega} a_0(x) \nabla \phi_i(x) \nabla \phi_j(x) \, dx, \quad i, j = 1, \ldots, n_\phi,
\]

\[
(21) \quad [A_k]_{i,j} = \int_{\Omega} \sigma a_k(x) \nabla \phi_i(x) \nabla \phi_j(x) \, dx, \quad i, j = 1, \ldots, n_\phi, k = 1, \ldots, d_\Xi,
\]

\[
(22) \quad [\mathbf{g}]_{j} = \int_{\Xi} \psi_j(\xi) w(\xi) \, d\xi, \quad j = 1, \ldots, n_\psi,
\]

\[
(23) \quad [\mathbf{f}]_{i} = \int_{\Omega} \phi_i(x) f(x) \, dx, \quad i = 1, \ldots, n_\phi.
\]

The vector \( \mathbf{u} \) collects the scalars \( u_{i,j} \) in (17) columnwise.

We choose the (normalized) Chebyshev weight function

\[
(24) \quad w(\xi) = \prod_{k=1}^{d_\Xi} \frac{1}{\pi \sqrt{1 - \xi_k^2}}.
\]

This is motivated by the fact that a \( L^2_\omega \) projection, with Chebyshev weight \( w \), of a continuous function on a Chebyshev basis results in a quasi-optimal approximation in the \( L^\infty \)-norm.

4. Two preconditioners for the discrete system. The major computational cost in solving a fuzzy PDE results from solving the high-dimensional \( n_\psi n_\phi \times n_\psi n_\phi \) system (18). In the context of stochastic PDEs—which after a polynomial chaos discretization yield a system with a similar structure to (18)—a lot of research on iterative solvers for the high-dimensional algebraic systems has been done; see, for example, [37]. In this section, we adapt some of the popular solvers for stochastic Galerkin finite element systems to fuzzy Galerkin discretizations, and we analyze how the use of Chebyshev polynomials influences the convergence.
4.1. Center-based preconditioner. Similar to a mean-based preconditioner [34, 43], a straightforward preconditioner to (18) is given by
\begin{equation}
I_n \otimes A_0.
\end{equation}

The concept of a mean value does not really make sense in a fuzzy context, so we shall use a different name. We call this preconditioner the center-based preconditioner, since it corresponds to the diffusion coefficient obtained by evaluating the parameters at the center of the parameter hypercube. In practice when applying this preconditioner, the inversion of $A_0$ is approximated by, e.g., one multigrid cycle. The convergence properties of this preconditioner follow from Theorem 4.1. They are summarized subsequently in Corollary 4.2.

**Theorem 4.1.** With $G_k$ and $A_k$ defined in (19)–(21), using normalized multidimensional Chebyshev polynomials on $[-1,1]$ and with $w(\xi)$ given by (24), the eigenvalues $\lambda$ of the generalized eigenvalue problem $\left(\sum_{k=0}^{d_\Xi} G_k \otimes A_k\right)u = \lambda (G_0 \otimes A_0) u$ lie in the interval $[1 - \tau, 1 + \tau]$, where
\begin{equation}
\tau = \sigma \left\| \frac{1}{a_0} \right\|_{L^\infty(\Omega)} \sum_{k=1}^{d_\Xi} \|a_k\|_{L^\infty(\Omega)}.
\end{equation}

**Proof.** The eigenvalues $\lambda$ can be written as $\lambda = \theta + 1$, where $\theta$ satisfies
\begin{equation}
\sum_{k=1}^{d_\Xi} \left( G_0 \otimes A_0 \right)^{-1} \left( G_k \otimes A_k \right) v = \theta v.
\end{equation}

Using properties of the Kronecker product and the fact that $G_0$ is the identity matrix results in
\begin{equation}
\sum_{k=1}^{d_\Xi} \left( G_k \otimes A_0^{-1} A_k \right) v = \theta v.
\end{equation}

Applying Lemma 3.2 in [34], we find that the eigenvalues of $A_0^{-1} A_k$ belong to the interval $[-\gamma, \gamma]$ with $\gamma = \sigma \left\| \frac{1}{a_0} \right\|_{L^\infty(\Omega)} \|a_k\|_{L^\infty(\Omega)}$.

From [38], we have that the eigenvalues of $G_k$ lie in the interval $[\zeta_{p+1,0}, \zeta_{p+1,p}]$, where $\zeta_{p+1,0}$ is the smallest zero and $\zeta_{p+1,p}$ the largest zero of a one-dimensional Chebyshev polynomial of degree $p + 1$. Since the zeros of a Chebyshev polynomial on $[-1,1]$ are bounded by $-1$ and $1$, the eigenvalues of the Kronecker product $G_k \otimes A_0^{-1} A_k$ lie also in the interval given by (28). Bounding the eigenvalues of (27) by the eigenvalue bounds of the matrices in the sum yields
\begin{equation}
-\tau \leq \theta_{\text{min}} \quad \text{and} \quad \theta_{\text{max}} \leq \tau
\end{equation}
with $\tau$ specified as (26). From this, the result follows.

**Corollary 4.2.** The number of iterations required to solve the algebraic system (18) with the conjugate gradients (CG) method, preconditioned by the center-based preconditioner (25), is independent of the mesh size $h$ and of the polynomial order $p$ when a Chebyshev polynomial discretization is used to define the matrices $G_k$ in (19).

From Theorem (4.1), we do notice that the convergence may degrade when large parameter variations occur (e.g., by a large value $\sigma$).

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.
4.2. Collective smoothing multigrid method. Based on earlier positive experiences with multigrid for stochastic Galerkin finite element discretizations [22, 38, 14, 36], we also propose a point-based multigrid strategy for the fuzzy Galerkin systems (18). The pointwise nature means that all the unknowns resulting from the fuzzy discretization will be updated simultaneously per grid point.

Smoothing operator. We suggest the use of a collective smoothing operator. Two variants will be considered. In the block Gauss–Seidel smoother each mesh point is visited sequentially, one after the other, and, locally, a linear system of size \( n_\psi \times n_\psi \) is solved. That linear system couples all degrees of freedom that are physically located at that particular grid point. The second variant is the block Jacobi smoother, where the values at the different grid points are updated in parallel. In terms of a classical matrix splitting iteration, the smoothing operator can be written as

\[
\begin{align*}
\left( \sum_{k=0}^{d_\psi} G_k \otimes A_k^+ \right) u^{\text{new}} &= g \otimes f - \left( \sum_{k=0}^{d_\psi} G_k \otimes A_k^- \right) u^{\text{old}}
\end{align*}
\]

with \( A_k^+ = A_k - A_k^- \). Matrix \( A_k^+ \) equals the (scaled) diagonal part of \( A_k \) or the lower triangular part in the case of a block Jacobi or a block Gauss–Seidel smoother, respectively. Every smoothing iteration (29) entails the solution of \( n_\psi \) sparse systems, each of size \( n_\psi \times n_\psi \). These systems can be solved, for example, by a direct solver.

Intergrid transfer operators and coarse grid operators. Coarsening will be done in the spatial dimensions only. A hierarchy of spatial grids is constructed, and the same hierarchy is used for each of the fuzzy unknowns. For the Fourier analysis and accompanying numerical results in section 5, we use standard coarsening by a factor of 2 in each spatial dimension, and the coarse grid operator is constructed by rediscritization. For the irregular mesh numerical results in section 6, an algebraic coarsening is used based on the matrix \( A_0 \). The latter matrix is provided as input to a standard algebraic multigrid (AMG) code for deterministic PDEs, which analyzes the properties of the matrix and generates a sequence of meshes together with the corresponding intergrid transfer operators. Denote by \( \hat{R}_{l-1}^l \) a restriction operator originating from a multigrid hierarchy for the deterministic matrix \( A_0 \) at multigrid level \( l \), and by \( \hat{P}_l^{l-1} \), the corresponding prolongation operator. The intergrid transfer operators for the system (18) are then given by

\[
\begin{align*}
R_{l-1}^l &= I_{n_\psi} \otimes \hat{R}_{l-1}^l \quad \text{and} \quad P_l^{l-1} = I_{n_\psi} \otimes \hat{P}_{l-1}^l.
\end{align*}
\]

The coarse grid operator is then constructed in the classical Galerkin way.

5. Local Fourier mode convergence analysis. In [38, 37], an LFA of multigrid applied to a stochastic Galerkin discretization of stochastic elliptic problems is detailed and numerical results are given for the case of a Hermite or Legendre polynomial stochastic discretization. The LFA of multigrid applied to fuzzy PDEs proceeds similarly, but the modified definition of the fuzzy discretization matrices has to be taken into account, which now uses Chebyshev polynomials. Below we summarize the main components of the Fourier analysis. Our model problem and its discretization are such that a direct comparison is possible with the corresponding results for the stochastic Galerkin case in [38, 37]. The considered problem is a two-dimensional diffusion equation, (31), discretized by finite differences. In principle, a Fourier analysis for a spatial finite element discretization could also be possible. However, the latter is somewhat technically more involved, especially for variable coefficient problems,
and does not really lead to an additional insight for the problem considered here. The numerical results in section 5.5 show the effect of the Chebyshev polynomial discretization on the convergence factors.

5.1. Model problem for LFA. We apply an LFA analysis to the following two-dimensional model problem:

\[
\frac{\partial^2 u(x,\tilde{\xi})}{\partial x_1^2} - a(x,\tilde{\xi}) \frac{\partial^2 u(x,\tilde{\xi})}{\partial x_2^2} = f(x) \quad \text{in} \ \Omega,
\]

where \( \tilde{\xi} := (\tilde{\xi}_1, \ldots, \tilde{\xi}_d) \in \mathcal{F}(\mathbb{R}^d) \) is a vector of noninteractive fuzzy numbers and \( \Xi := [\xi]_{10} \) is the parameter domain. An infinite spatial domain \( \Omega := \mathbb{R}^d \) is assumed in order to eliminate the effect of boundary conditions. We apply the fuzzy Galerkin discretization described in section 3.2 to (31). The spatial discretization uses a standard five-point finite difference scheme on a rectangular grid \( \Omega_h = \{(ih,jh)\}_{i,j \in \mathbb{Z}} \) with grid spacing \( h \) in \( x_1 \) - and \( x_2 \)-directions. Using the orthonormality of the Chebyshev polynomials, we arrive at the following algebraic system:

\[
-M_{i,j} (u_{i,j-1} - 2u_{i,j} + u_{i,j+1}) = h^2 f_{i,j} \quad \text{with} \quad M_{i,j} = \int_{\Xi} a(x,\xi) \psi_r(\xi) \psi_s(\xi) d\xi.
\]

Note that in the case of a variable coefficient problem, LFA is performed by freezing each coefficient to its value at the considered grid point \((ih,jh)\) [50]. For problem (31), this corresponds to replacing the fuzzy field \( a(x,\tilde{\xi}) \) by a fuzzy number \( a(\xi) \). As such, \( M_{i,j} \) can be considered to be a fixed known matrix, which will be denoted as \( M \).

When the equations are collected over all grid points, a linear system of equations results,

\[
L_h u_h = f_h.
\]

The dimension of \( L_h \) equals the number of spatial grid points multiplied by \( n_\psi \).

5.2. Local Fourier representation. In order to set up a LFA, we decompose the iteration error into a sum of exponential Fourier grid modes of the form

\[
ee(\theta, z) = \exp(\imath (i\theta_1 + j\theta_2)) z,
\]

where \( z \in \mathbb{R}^n, \theta := (\theta_1, \theta_2) \in [-\pi, \pi]^2 \), and \( \imath \) represents the imaginary unit. Note that the linear discrete operator \( L_h \) in (33) is invariant to \( e(\theta, z) \):

\[
L_h e(\theta, z) = \hat{L}_h(\theta) e(\theta, z),
\]

where the symbol \( \hat{L}_h(\theta) \) of \( L_h \) is defined as

\[
\hat{L}_h(\theta) = \frac{1}{h^2} \left( \frac{\sin^2 (\theta_1/2) I_{n_\psi} + \sin^2 (\theta_2/2) M}{\exp(-i\theta_1) - 2 + \exp(i\theta_1)} + M \left( \exp(-i\theta_2) - 2 + \exp(i\theta_2) \right) \right)
\]

\[= -\frac{4}{h^2} \left( \sin^2 (\theta_1/2) I_{n_\psi} + \sin^2 (\theta_2/2) M \right).
\]
If $z$ in (34) is selected to be one of the eigenvectors $z_m$ of $M$ with corresponding eigenvalue $\lambda_m$, equality (35) simplifies to $L_h e(\theta, z_m) = \hat{L}_h(\theta, \lambda_m) e(\theta, z_m)$ with

$$
\hat{L}_h(\theta, \lambda_m) = -\frac{4}{h^2} \left( \sin^2 (\theta_{x_1}/2) + \sin^2 (\theta_{x_2}/2) \lambda_m \right).
$$

Hence, the Fourier mode $e(\theta, z_m)$ is an eigenfunction of the (frozen) discrete differential operator.

### 5.3. Smoothing analysis.

For many stationary iterative methods, e.g., Jacobi and lexicographic Gauss–Seidel iterations, the Fourier modes (34) are eigenfunctions of the corresponding iteration operator $S_h$. The corresponding eigenvalues are called the amplification factor or Fourier symbol of the iteration operator, denoted by $\hat{S}_h(\theta)$, and determine the asymptotic convergence factor. For a variable coefficient problem with a sufficiently smooth coefficient $a$, this convergence factor is defined as

$$
\rho = \max_{x=(ih,jh) \in \Omega_h} \max_{\theta \in [-\pi,\pi]^2} \rho \left( \hat{S}_h(\theta) \right)
$$

with $\rho(\hat{S}_h(\theta))$ the spectral radius of $\hat{S}_h(\theta)$.

Applying the block Gauss–Seidel and damped block Jacobi iteration operator characterized by (29) to the Fourier mode (34), we find that

$$
S_h e(\theta, z_m) = \hat{S}_h(\theta, \lambda_m) e(\theta, z_m),
$$

where the symbols $\hat{S}_h$ are, respectively, given by

$$
\hat{S}_h^{\text{GS}}(\theta, \lambda_m) = \frac{\exp(i\theta_{x_1}) + \exp(i\theta_{x_2}) \lambda_m}{(2 - \exp(-i\theta_{x_1})) + (2 - \exp(-i\theta_{x_2})) \lambda_m},
$$

$$
\hat{S}_h^{\text{Jac}}(\theta, \lambda_m) = 1 - \omega + \omega \frac{\cos(\theta_{x_1}) + \cos(\theta_{x_2}) \lambda_m}{1 + \lambda_m}.
$$

Here, $\omega$ represents the Jacobi damping factor and $\lambda_m$ equals the eigenvalue of $M$ corresponding to the eigenvector $z_m$. Proceeding analogously to [38, 37], the optimal damping factor $\omega$ can be determined, as can the symbols of other classical (block) splitting iterations.

### 5.4. Two-grid analysis.

To determine the action of a two-grid operator on the Fourier modes (34), the Fourier space is decomposed into subspaces spanned by four harmonics, $\mathcal{H}(\theta, z) := \text{span}[e(\theta_1, z), e(\theta_2, z), e(\theta_3, z), e(\theta_4, z)]$, for a given $(\theta_{x_1}, \theta_{x_2}) \in [-\pi, \pi]^2$

$$
\theta_1 = (\theta_{x_1}, \theta_{x_2}), \quad \theta_3 = (\theta_{x_1} - \text{sign}(\theta_{x_1}) \pi, \theta_{x_2}),
$$

$$
\theta_2 = (\theta_{x_1}, \theta_{x_2} - \text{sign}(\theta_{x_2}) \pi), \quad \theta_4 = (\theta_{x_1} - \text{sign}(\theta_{x_1}) \pi, \theta_{x_2} - \text{sign}(\theta_{x_2}) \pi).
$$

These spaces are invariant under the fine and coarse grid discrete differential operators and under certain smoothing operators. The action of a smoothing operator on an element of such a space can be described by a $(4 \times 4)$ diagonal matrix $\hat{S}_h(\theta, \lambda_m)$ with

$$
\hat{S}_h(\theta, \lambda_m) := \text{diag}(\hat{S}_h(\theta_1, \lambda_m), \hat{S}_h(\theta_2, \lambda_m), \hat{S}_h(\theta_3, \lambda_m), \hat{S}_h(\theta_4, \lambda_m)).
$$

A similar diagonal matrix representation for the action of $L_h$ holds and is denoted by $\hat{L}_h$. On the coarse grid, $L_{2h}$ is constructed by discretizing (31) with a standard
five-point finite difference scheme on a rectangular grid with grid spacing \(2h\). Its action can be represented by

\[
\widehat{L}_{2h}(\theta, \lambda_m) := \frac{1}{h^2} \left( \sin^2(\theta_{x_1}) + \sin^2(\theta_{x_2}) \lambda_m \right).
\]

The prolongation operator (30) maps the mode \(e(\theta, z)\) onto \(\mathcal{H}(\theta, z)\) [42]. In the case of bilinear interpolation, it is characterized by the symbol \(\widehat{P}_{2h}^h(\theta)\), which is given by

\[
\widehat{P}_{2h}^h(\theta) := \frac{1}{4} \begin{bmatrix}
1 + \cos(\theta_{x_1}) & 1 + \cos(\theta_{x_2}) \\
1 + \cos(\theta_{x_1}) & 1 - \cos(\theta_{x_2}) \\
1 - \cos(\theta_{x_1}) & 1 + \cos(\theta_{x_2}) \\
1 - \cos(\theta_{x_1}) & 1 - \cos(\theta_{x_2})
\end{bmatrix}.
\]

Using standard coarsening, the restriction operator maps the space \(\mathcal{H}(\theta, z)\) onto the single mode \(e(\theta_1, z)\). The corresponding Fourier representation is given by

\[
\widehat{R}_{2h}^h(\theta) = (\widehat{P}_{2h}^h(\theta))^T.
\]

In summary, the action of the two-grid operator, specified in section 4.2 and applied to the differential operator (32) on the space \(\mathcal{H}(\theta, z)\), is characterized by

\[
\widehat{T}_h(\theta, \lambda_m) := (\widehat{S}_h(\theta, \lambda_m))^{\nu_2} (I_4 - \widehat{P}_{2h}^h(\theta)) \left(\widehat{L}_{2h}(\theta, \lambda_m)\right)^{-1} \widehat{R}_{2h}^h(\theta) \widehat{L}_h(\theta, \lambda_m) \left(\widehat{S}_h(\theta, \lambda_m)\right)^{\nu_1},
\]

where \(\nu_1\) and \(\nu_2\) are the number of presmoothing, respectively, postsmoothing, steps, and \(I_4 \in \mathbb{R}^{4x4}\) is an identity matrix. Under the assumption that the variation of the coefficient \(a\) is sufficiently smooth, the asymptotic convergence factor of the two-grid scheme is defined as

\[
\rho_{TG} = \max_{x=(2h, 2h)} \max_{\lambda_m \in \sigma(M_{i,j})} \max_{\theta \in [-\pi/2, \pi/2]^2} \rho \left(\widehat{T}_h(\theta, \lambda_m)\right).
\]

5.5. Numerical results. We shall demonstrate the correctness and accuracy of the Fourier analysis and comment on the convergence properties of the methods proposed in section 4. We consider model problem (31) on a unit square \(\Omega = [0, 1]^2\) with zero Dirichlet boundary conditions. The fuzzy diffusion coefficient \(\tilde{a}\) is given by a linear combination of \(d\) triangular fuzzy numbers as in (6). The functions \(a_k(x)\) are constructed as \(a_k(x) = \sqrt{\kappa_k v_k(x)}\) with \(\kappa_k\) and \(v_k\), respectively, the eigenvalues and eigenfunctions of an exponential kernel \(C(x, x') = \exp(-\|x - x'\|_1)\).

5.5.1. Collective smoothing multigrid method. Table 1 shows the theoretical multigrid convergence factors obtained by the LFA for various choices of the discretization parameters. These values were obtained numerically from (39) by an exhaustive search over the grid \(\Omega_k\) and the spectrum \(\sigma(M_{i,j})\) and a fine grid sampling of \([-\pi/2, \pi/2]^2\). The eigenvalues \(\sigma(M_{i,j})\) were computed numerically from the explicitly constructed matrices \(M_{i,j}\). Further, the table also provides numerically observed convergence factors, obtained with an implementation of the algorithm. The numerical results confirm the accuracy of the LFA results.

Table 1 demonstrates the robust convergence behavior of multigrid: the convergence factors are independent of the spatial and fuzzy discretization parameters and are only slightly influenced by the width of the support of the fuzzy field, as determined by \(\sigma\).
Numerical ($\rho_{\text{num}}$) and theoretical ($\rho_{\text{theo}}$) convergence factors for the two-grid cycle TG(2,1) with lexicographic Gauss–Seidel smoother. The default parameters are $h = 2^{-5}$, $d_\sigma = 4$, $p = 2$, $\sigma/a_0 = 0.2$. Every block row corresponds to the case when one of those default values is varied.

<table>
<thead>
<tr>
<th>Grid spacing $h$</th>
<th>$h = 2^{-4}$</th>
<th>$h = 2^{-5}$</th>
<th>$h = 2^{-6}$</th>
<th>$h = 2^{-7}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rho_{\text{theo}}$</td>
<td>0.120</td>
<td>0.120</td>
<td>0.120</td>
<td>0.120</td>
</tr>
<tr>
<td>$\rho_{\text{num}}$</td>
<td>0.106</td>
<td>0.110</td>
<td>0.112</td>
<td>0.112</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Fuzzy numbers $d_\sigma$</th>
<th>$d_\sigma = 1$</th>
<th>$d_\sigma = 5$</th>
<th>$d_\sigma = 8$</th>
<th>$d_\sigma = 10$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rho_{\text{theo}}$</td>
<td>0.119</td>
<td>0.120</td>
<td>0.120</td>
<td>0.120</td>
</tr>
<tr>
<td>$\rho_{\text{num}}$</td>
<td>0.109</td>
<td>0.111</td>
<td>0.111</td>
<td>0.111</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Polynomial order $p$</th>
<th>$p = 1$</th>
<th>$p = 3$</th>
<th>$p = 4$</th>
<th>$p = 5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rho_{\text{theo}}$</td>
<td>0.119</td>
<td>0.120</td>
<td>0.121</td>
<td>0.122</td>
</tr>
<tr>
<td>$\rho_{\text{num}}$</td>
<td>0.111</td>
<td>0.111</td>
<td>0.110</td>
<td>0.111</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scaling factor $\sigma/a_0$</th>
<th>$\sigma/a_0 = 0.1$</th>
<th>$\sigma/a_0 = 0.4$</th>
<th>$\sigma/a_0 = 0.6$</th>
<th>$\sigma/a_0 = 0.7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rho_{\text{theo}}$</td>
<td>0.119</td>
<td>0.127</td>
<td>0.175</td>
<td>0.247</td>
</tr>
<tr>
<td>$\rho_{\text{num}}$</td>
<td>0.109</td>
<td>0.114</td>
<td>0.160</td>
<td>0.221</td>
</tr>
</tbody>
</table>

Table 2 illustrates the LFA convergence factors of (40). The computation of these convergence factors is based on the LFA derivation given in [37], adapted to the use of Chebyshev polynomials. Although the convergence factors in Table 2 behave less regularly than in Table 1, we observe an asymptotic independence of

5.5.2. Center-based preconditioner. The LFA presented for the collective smoothing multigrid method can also be applied to other iterative methods. As a standalone solver, the center-based preconditioner can be interpreted as block Jacobi method. For the model problem, it can be formulated as

\[
(40) \quad -(u_{i-1,j}^{\text{new}} - 2u_{i,j}^{\text{new}} + u_{i+1,j}^{\text{new}}) - M^+ (u_{i,j}^{\text{new}} - 2u_{i,j}^{\text{old}} + u_{i,j+1}^{\text{new}}) = h^2 f_{i,j} + M^- (u_{i,j-1}^{\text{old}} - 2u_{i,j}^{\text{old}} + u_{i,j+1}^{\text{old}}), \quad i, j = 1, \ldots, 
\]

where $M^+$ equals the diagonal part of $M$ and $M^- = M - M^+$. The equivalence between the preconditioner (25) and the iteration (40) follows from the orthonormality of the polynomial basis. Indeed, $G_0 \equiv I_{n_x}$ and the diagonal elements of $G_k$, $k = 1, \ldots, d_\sigma$ are zero, as proved in [37].

Table 2 illustrates the LFA convergence factors of (40). The computation of these convergence factors is based on the LFA derivation given in [37], adapted to the use of Chebyshev polynomials. Although the convergence factors in Table 2 behave less regularly than in Table 1, we observe an asymptotic independence of
the convergence behavior on the discretization parameters. This property is also confirmed by Theorem 4.1.

As a straightforward extension to iteration (40), we can also consider the block Gauss–Seidel variant, in which case $M^+ = M$ in (40) equals the lower triangular part of $M$. The LFA convergence factors of the block Gauss–Seidel case are given in Table 3. Comparing the results in Table 1 to those in Table 3, we note the smaller convergence factors for the block Gauss–Seidel method in comparison to the multigrid method. Each iteration of (40) is, however, substantially more expensive than one multigrid iteration since several systems of the size of the number of deterministic unknowns have to be solved during every iteration. The same holds for the method corresponding to Table 2. These expensive solves will be approximated by one deterministic multigrid cycle when applying the center-based preconditioner to practical examples.

6. Numerical experiments. In this section, we verify the accuracy of our spectral Galerkin discretization approach numerically, and we demonstrate the convergence properties of the proposed iterative solvers on two model problems. The accuracy of the computed solutions will be assessed in the $L^\infty(\Xi; H^1(\Omega))$-norm; see Corollary 2.1. In all numerical experiments, we iterate until the Euclidean norm of the relative residual is smaller than $10^{-9}$.

6.1. Diffusion equation on an L-shaped domain.

6.1.1. Problem setup. We consider the diffusion equation (4) on an L-shaped domain, as depicted in Figure 1(a), together with Dirichlet and Neumann boundary conditions. The source term $f(x)$ is set to zero. The spatial domain is partitioned in three regions, $\Omega = \Omega_1 \cup \Omega_2 \cup \Omega_3$. The fuzzy diffusion coefficient is modeled independently in each of the subdomains by an expansion of the form (6) with, respectively, $d_{\Xi,1}$, $d_{\Xi,2}$, and $d_{\Xi,3}$ terms. The total number of fuzzy parameters in the model is then $d_\Xi = d_{\Xi,1} + d_{\Xi,2} + d_{\Xi,3}$. For each of the subdomains, we model the $\xi_k$ as noninteractive triangular fuzzy numbers with support $[-1, 1]$ and core equal to 0. The functions $a_k(x)$ are constructed as $\frac{\alpha_k(x)}{\kappa_k(x)}$ with $\kappa_k$ and $\alpha_k$ the eigenvalues and eigenfunctions of an exponential kernel $C(x, x') = \exp(-\|x - x'\|_1/L_c)$ with $L_c = 0.7$ on $\Omega_1$, $L_c = 0.3$ on $\Omega_2$, and $L_c = 0.1$ on $\Omega_3$. The functions $a_0$ in (6) are taken equal to 30, 5, and 100 on subdomains $\Omega_1$, $\Omega_2$, and $\Omega_3$, respectively. Finally, we apply $\sigma = 10$ on $\Omega_1$, $\sigma = 2$ on $\Omega_2$, and $\sigma = 20$ on $\Omega_3$. 

<table>
<thead>
<tr>
<th>Table 3</th>
<th>Theoretical ($\rho_{theo}$) and numerical ($\rho_{num}$) convergence factors for the block Gauss–Seidel iteration (40) applied to the LFA model problem (31). (Default: $h = 2^{-5}$, $d_\Xi = 4$, $p = 2$, $\sigma/a_0 = 0.2$).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grid spacing $h$</td>
<td>$h = 2^{-4}$</td>
</tr>
<tr>
<td>$\rho_{theo}$</td>
<td>0.0388</td>
</tr>
<tr>
<td>$\rho_{num}$</td>
<td>0.0950</td>
</tr>
<tr>
<td>Fuzzy numbers $d_\Xi$</td>
<td>$d_\Xi = 1$</td>
</tr>
<tr>
<td>$\rho_{theo}$</td>
<td>0.0216</td>
</tr>
<tr>
<td>$\rho_{num}$</td>
<td>0.0275</td>
</tr>
<tr>
<td>Polynomial order $p$</td>
<td>$p = 1$</td>
</tr>
<tr>
<td>$\rho_{theo}$</td>
<td>0.0168</td>
</tr>
<tr>
<td>$\rho_{num}$</td>
<td>0.0160</td>
</tr>
<tr>
<td>Scaling factor $\sigma/a_0$</td>
<td>$\sigma/a_0 = 0.1$</td>
</tr>
<tr>
<td>$\rho_{theo}$</td>
<td>0.00987</td>
</tr>
<tr>
<td>$\rho_{num}$</td>
<td>0.00862</td>
</tr>
</tbody>
</table>
ITERATIVE SOLVERS FOR FUZZY PDES

Material 1 \((\Omega_1)\)

\[ \frac{\partial u}{\partial n} = 0 \]

Material 2 \((\Omega_2)\)

\[ \frac{\partial u}{\partial n} = 0 \]

Material 3 \((\Omega_3)\)

\[ \frac{\partial u}{\partial n} = 0 \]

\(u = 30\)

\(\tilde{u}_1 = 0\)

\(\tilde{u}_2 = 0\)

\(\tilde{u}_3 = 0\)

\(a\)

\(b\)

Fig. 1. (a) The L-shaped domain. (b) Cross-section of the fuzzy solution along the dotted line in (a), with the gray-scale colors representing the membership value \(\mu_{\tilde{u}(x)}\).

The fuzzy solution is depicted in Figures 1(b) and 2. A third-order Chebyshev discretization of the fuzzy parameter space with \(d_{\Xi,1} = 1\), \(d_{\Xi,2} = 2\), and \(d_{\Xi,3} = 3\) is applied, along with a spatial triangular finite element mesh using \(n_\phi = 32,499\) degrees of freedom.

6.1.2. Accuracy of spectral Galerkin approximation. Figures 3 and 4 illustrate the accuracy of the Chebyshev response surface obtained via the spectral Galerkin approach for different sets of parameters. Only the error by the discretization in the fuzzy dimension is considered, not the error from the spatial discretization. The size of the fuzzy error can be estimated by using Theorem 2.1, which provides an upper bound for the distance between two fuzzy fields. That is, we compute the \(L^\infty(\Xi; H^1(\Omega))\)-norm of the error, which is given by

\[
\|u_{\nu_0}^\tilde{u} - u_{\nu_0, n_\phi}^\tilde{u}\|_{L^\infty(\Xi; H^1(\Omega))}.
\]

Here \(u_{\nu_0, n_\phi}^\tilde{u}\) is the computed response surface (17), and \(u_{\nu_0}^\tilde{u}\) is the finite element solution of the parametric PDE (7). The \(L^\infty\)-norm over the uncertainty domain \(\Xi\) in (41) is approximated on the basis of a uniform quasi-Monte Carlo sampling of the parameter domain \(\Xi\) with \(10^4\) samples. Each such sample requires the solution of one deterministic PDE of the form (7) and one evaluation of the response surface. The figures show the relative error \(\|u_{\nu_0}^\tilde{u} - u_{\nu_0, n_\phi}^\tilde{u}\|_{L^\infty(\Xi; H^1(\Omega))}/\|u_{\nu_0}^\tilde{u}\|_{L^\infty(\Xi; H^1(\Omega))}\).
Fig. 3. The relative error in the $L^\infty(\Xi; H^1(\Omega))$-norm of the polynomial response surface of the solution to the fuzzy diffusion equation, (4), as (a) a function of the polynomial order $p$ and as (b) a function of the number of polynomial basis functions $n_\psi$ for different numbers $d_\Xi$ of fuzzy parameters ($n_\phi = 11258$, $L_c = [0.7, 0.3, 0.1]$, $\sigma/a_0 = [1/3, 0.4, 0.2]$).

Fig. 4. The relative error in the $L^\infty(\Xi; H^1(\Omega))$-norm of the polynomial response surface of the solution to the fuzzy diffusion equation, (4), as a function of the polynomial order $p$ (a) for different correlation lengths $L_c$ and (b) different $\sigma/a_0$ ($n_\phi = 11258$, $d_\Xi = [1, 2, 3]$, and unless specified otherwise, $L_c = [0.7, 0.3, 0.1]$, $\sigma/a_0 = [1/3, 0.4, 0.2]$).

Figures 3(a) and 4 show the error as a function of the polynomial order for different numbers of fuzzy parameters $d_\Xi$, different correlation lengths $L_c$, and different $\sigma/a_0$. For all sets of the parameters, an exponential decay of the fuzzy error is observed. The deviations of the exponential convergence in Figure 4 for polynomial degree $p = 5$ are due to the relative accuracy of $10^{-9}$ in the Euclidian norm of the iterative solver. The difference in slope of the convergence graphs for different sets of parameters is caused by a difference in the width of the support of the fuzzy diffusion coefficient $\tilde{a}$. Finally, Figure 3(b) shows the error as a function of the number of fuzzy degrees of freedom $n_\psi = \frac{(d_\Xi + p)!}{d_\Xi!p!}$. The number of fuzzy degrees of freedom determines the size of the $G_k$-matrices and is a good indicator for the amount of computational work (see section 6.1.3 below).

6.1.3. Convergence properties of iterative solvers. In this section, we compare the convergence properties and execution times of the iterative solvers proposed in section 4 when applied to the fuzzy diffusion equation on the L-shaped domain. In the case of the center-based preconditioner (25), we approximate the inversion of $A_0$ with one multigrid V(1,1)-cycle. Given an unstructured finite element discretization,
we use Ruge–Stüben AMG [40] with Gauss–Seidel smoother. In the case of the collective smoothing multigrid method, we also consider the AMG variant and use this as preconditioner for the CG method. We apply W(2,1)-cycles with a block Jacobi smoother with damping factor $4/5$. In both cases, the same AMG building blocks are used to construct the prolongation and coarse grid operators.

Table 4 shows the required number of iterations and solution time of CG preconditioned by either the center-based preconditioner or the collective smoothing multigrid method for a variety of parameter settings. We observe for both preconditioners a robust convergence behavior with respect to the number of spatial nodes and the polynomial order. The slight increase in iteration count for the center-based preconditioner in the case of a growing number of fuzzy numbers can be explained from the enlargement of the support of $\tilde{a}$ in (6) when more—nonnegligible—expansion terms are taken into account. The effect of the width of the support is more clearly visible in the last row of Table 4, which shows an increase of iteration counts for the center-based preconditioner for large values of $\sigma/a_0$.

This table also presents solution times of the center-based and the collective smoothing multigrid preconditioner. Obviously, it is a delicate matter to compare timing results of two different codes, since they reflect not just algorithmic issues but also a multitude of implementation aspects. For what it is worth, we observe that both codes perform similarly overall. However, our current implementation of the center-based preconditioner appears to be faster than the collective smoothing multigrid implementation. This is mainly due to a lower set-up cost and a cheaper matrix-vector product.


**6.2.1. Problem setup.** As a second test case, we consider a plane stress elasticity problem. The geometry of the problem is illustrated in Figure 5. It represents a two-dimensional plate with two clamped boundaries on the left side and
one traction boundary on the right. All other boundaries are free. Denote by \( \mathbf{u} := (u_{x_1}(x), u_{x_2}(x)) \) the displacement vector describing the deformation of the material depicted in Figure 5 under a load vector \( \mathbf{f} := (f_{x_1}(x), f_{x_2}(x)) \) and a boundary traction \( \mathbf{t} := (t_{x_1}(x), t_{x_2}(x)) \). Assuming isotropic and isothermal conditions, the displacement \( \mathbf{u} \) is governed by

\[
-\nabla \cdot (C : \nabla \mathbf{u}) = \mathbf{f},
\]

where \( C \) is the fourth-order stiffness tensor and \( : \) denotes the matrix inner product [4].

The fourth-order stiffness tensor can be written as composed of four 2-by-2 matrices

\[
C_{1,1} = \begin{bmatrix}
\frac{E}{1-\nu^2} & 0 \\
0 & \frac{E \nu}{2(1+\nu)}
\end{bmatrix}, \quad
C_{1,2} = \begin{bmatrix}
0 & \frac{E}{2(1+\nu)} \\
\frac{E \nu}{2(1+\nu)} & 0
\end{bmatrix},
\]

\[
C_{2,1} = C_{1,2}^T, \quad
C_{2,2} = \begin{bmatrix}
\frac{E \nu}{1-\nu^2} & 0 \\
0 & \frac{E}{1-\nu^2}
\end{bmatrix},
\]

given the Young’s modulus \( E \) and Poisson’s ratio \( \nu \) of the material. Equations (42)–(43) follow from the constitutive equations for the stress, i.e., \( \mathbf{\sigma} = C : \mathbf{\epsilon} \), combined with the strain-displacement equations for the strain tensor \( \mathbf{\epsilon} \)

\[
\mathbf{\epsilon} = \frac{1}{2} \left( \nabla \mathbf{u} + (\nabla \mathbf{u})^T \right).
\]

Assuming uncertain material parameters, we model the Young’s modulus as a fuzzy field \( \tilde{E} \). As a consequence, we have a fuzzy displacement \( \tilde{\mathbf{u}} \), a fuzzy strain \( \tilde{\mathbf{\epsilon}} \), and fuzzy stress \( \tilde{\mathbf{\sigma}} \). The notation \( \tilde{\cdot} \) denotes that all vector/tensor entries are given by fuzzy fields instead of deterministic fields.

In the numerical experiments, we consider a 1-by-1 meter aluminium plate with a thickness of 1 mm and a Poisson’s ratio of \( \nu = 0.35 \). The right boundary is subjected to an outward oriented normal load of 5 MN/m, corresponding to a surface traction of \( \mathbf{t} = (5000, 0) \text{MN/m}^2 \). The volume force is equal to \( \mathbf{f} = (0, -400) \text{MN} \). The Young's modulus is given by a linear combination of \( d_k \) noninteractive triangular fuzzy numbers with support \([-1, 1]\) and core equal to 0 as in (6), where the deterministic functions \( E_k(x) \) are constructed as \( E_k(x) = \sqrt{\kappa_k} v_k(x) \) with \( \kappa_k \) and \( v_k \), respectively, the eigenvalues and eigenfunctions of an exponential kernel \( C(x, x') = \exp(-\|x - x'\|_1) \) on a square. The deterministic function \( E_0 \) is taken equal to the Young's modulus of aluminium, \( E_0 = 70 \cdot 10^3 \text{MN/m}^2 \). Figures 6 and 7 illustrate the core, the width of the support, and a cross section of, respectively, the \( x_1 \) and \( x_2 \)-components of the fuzzy...
displacement vector using $\sigma = 5 \cdot 10^{-2} E_0$, a spatial triangular mesh with $n_\phi = 59,666$ degrees of freedom, $d_\Xi = 6$, and a third-order Chebyshev polynomial response surface. The cross section is taken at $x_1 = 0.25$.

6.2.2. Accuracy of spectral Galerkin approximation. Similarly as for the fuzzy diffusion problem, we investigate the accuracy of the Chebyshev response surface approximation. To that end, we again apply Corollary 2.1 and measure the fuzzy error in the $L^\infty(\Xi; H^1(\Omega))$-norm, see (41), using a uniform quasi–Monte Carlo sampling of the parameter space $\Xi$ with $10^4$ samples. Figures 8 and 9 show that for all variations of the parameters an exponential convergence of the fuzzy error is obtained for the plane stress problem.

6.2.3. Convergence properties of iterative solvers. Although the iterative solvers proposed in section 4 were originally developed for solving the Galerkin discretization of a scalar fuzzy PDE, they can easily be extended to the solution of a system of fuzzy elliptic PDEs, as in (42). In the case of a PDE system, the definition of the center-based preconditioner remains given by (25), but the multigrid components of the collective smoothing multigrid method described in section 4.2 require modifications. These modifications are also needed for the multigrid solver that is used to invert the $A_0$-matrix in the center-based preconditioner (25) approximately.

Applying AMG for scalar PDEs straightforwardly to PDE systems is known to often result in a performance degradation [42]. Typical multigrid solutions for PDE systems include the unknown-based and point-based multigrid approach [40]. These multigrid methods apply a blockwise construction of the prolongation and coarse grid operators, and possibly also of the relaxation operators. The block size is determined...
either by the number of degrees of freedom per physical unknown in the unknown-based case, or by the number of unknowns per spatial node in the point-based case. An additional multigrid difficulty arises in the context of elasticity problems. Classical AMG interpolation operators do not adequately represent the null-space of the corresponding PDE operator, which consists of the so-called rigid body modes. In the case of a two-dimensional plane stress problem, the rigid body modes are composed of two translations and one rotation vector. The translations are captured in the coarse grid operator when using classical (Ruge–Stüben) AMG, but the rotation is not. As a result, the convergence of unknown-based classical AMG deteriorates in the case that the system is nearly singular, i.e., when Dirichlet boundary conditions are imposed on only a small part of the domain. To resolve this issue, several solutions have been proposed. We follow the approach detailed in [44], where a point-based smoothed aggregation AMG is proposed that makes use of the prior knowledge of the rigid body modes.

The smoothed aggregation AMG setup is used both as part of the center-based preconditioner and for constructing the multigrid hierarchy for the collective smoothing multigrid preconditioner. That is, the inversion of the center-based matrix $A_0$ in the center-based preconditioner is approximated by a $V(1,1)$-AMG cycle with Gauss–Seidel smoother and smoothed aggregation AMG interpolation matrices. The
Table 5 illustrates the convergence of the methods. We note that similar optimal convergence properties are observed as for the fuzzy diffusion problem: a convergence rate that is independent of the size of the spatial and fuzzy discretization parameters. The center-based preconditioner is, however, in contrast to the multigrid method, not robust with respect to the width of the support of the fuzzy input. The computational cost of the collective smoothing method is again overall higher than the cost of the center-based preconditioner.

7. Conclusions. This paper presented a numerical study and analysis of a spectral Galerkin method for solving fuzzy PDEs. In the literature, the response surfaces for solving fuzzy PDEs are typically constructed by means of a sparse grid interpolation or very generic response surface techniques like Kriging. Here, we demonstrated that spectral Galerkin methods, originally created for stochastic PDEs, can also be applied very effectively for solving fuzzy PDEs.

In the fuzzy context, the accuracy of an approximation is measured in a quite different norm than in the stochastic context. This leads us to use Chebyshev expansions rather than the more classical Hermite or Legendre expansions that are used for stochastic PDEs. This Chebyshev representation turns out to yield very accurate (in the fuzzy sense) approximations. We numerically demonstrated exponential convergence for two nontrivial example problems. The convergence properties of the two preconditioners developed in this paper were shown to be optimal w.r.t. the discretization parameters. This was shown both theoretically, by means of a local Fourier mode analysis, as well as numerically, by means of an extensive set of numerical experiments. A numerical convergence study on a fuzzy diffusion problem and a fuzzy elasticity problem showed that both methods perform quite well, also on problems defined on irregular meshes and for systems of equations. The center-based preconditioner turned out to be the most efficient solver overall. On the other hand, the collective smoothing multigrid method applies \( W(2, 2) \)-cycles with a collective Jacobi smoother with a damping factor of \( 2/3 \).
collective smoothing multigrid outperforms the center-based preconditioner w.r.t. robustness, by showing an almost constant number of iterations over a very wide range of parameter values.

REFERENCES

ITERATIVE SOLVERS FOR FUZZY PDES


