Numerical study of a parametric parabolic equation and a related inverse boundary value problem

Lauri Mustonen

Aalto University, Department of Mathematics and Systems Analysis, PO Box 11100, FI-00076 Aalto, Finland

E-mail: lauri.mustonen@aalto.fi

Received 5 June 2015, revised 14 July 2016
Accepted for publication 20 July 2016
Published 1 September 2016

Abstract
We consider a time-dependent linear diffusion equation together with a related inverse boundary value problem. The aim of the inverse problem is to determine, based on observations on the boundary, the nonhomogeneous diffusion coefficient in the interior of an object. The method in this paper relies on solving the forward problem for a whole family of diffusivities by using a spectral Galerkin method in the high-dimensional parameter domain. The evaluation of the parametric solution and its derivatives is then completely independent of spatial and temporal discretizations. In the case of a quadratic approximation for the parameter dependence and a direct solver for linear least squares problems, we show that the evaluation of the parametric solution does not increase the complexity of any linearized subproblem arising from a Gauss–Newtonian method that is used to minimize a Tikhonov functional. The feasibility of the proposed algorithm is demonstrated by diffusivity reconstructions in two and three spatial dimensions.

Keywords: parametric PDEs, inverse boundary value problems, nonlinear least squares, thermal tomography

(Some figures may appear in colour only in the online journal)

1. Introduction

Inverse boundary value problems arise in situations where one tries to find information about interior properties of an object by boundary measurements. In electrical impedance tomography (EIT), for example, an electric current is injected into the body and the corresponding voltages are measured across the boundary [8]. The aim is to reconstruct the electrical
conductivity as a function, or to locate conductivity anomalies having prescribed (e.g., constant or vanishing) conductivity values. The inverse problem is nonlinear and ill-posed, whereas the forward problem, namely, determining the boundary voltages when the conductivity and current patterns are given, is governed by a well-posed elliptic partial differential equation. Mathematically equivalent applications include electrical capacitance tomography. 

In this paper, we consider the inverse boundary value problem of a time-dependent diffusion equation. This could be a model for thermal tomography, where the thermal diffusivity is to be reconstructed. We assume that the heat flux at the boundary, as well as the initial temperature, are controlled and that the boundary temperatures are measured. The diffusivity is assumed to be time-independent. Unlike in the EIT problem, where the steady-state voltages are measured for several current patterns, in this paper we treat a single (or a few) initial and boundary conditions and measure the boundary temperatures at several instances of time. Similar inverse boundary value problems for stationary inclusion-type diffusivities have been examined in, e.g., [10, 11, 24, 27]. In [29], both heat capacity and thermal conductivity were reconstructed simultaneously by using a least squares approach with several boundary conditions. In [42], an extension to unknown surface flux was proposed and the feasibility of the method was verified with experimental three-dimensional data in [43]. The case of time-varying inclusions has been studied in [17]. For theoretical treatment of general inverse parabolic problems, we refer to [28].

Our approach to the inverse problem is based on writing the forward problem as a parametric differential equation. To that end, the temperature is viewed as a function depending not only on spatial and temporal variables, but also on parameters that define the diffusivity function. For the high-dimensional parameter domain, we adopt the spectral Galerkin method when the numerical solution is sought. The spatial domain is discretized with the finite element method (FEM). The combination was dubbed stochastic Galerkin FEM in, e.g., [4], where the parameters were interpreted as random variables. In that context, the spectral discretization is often called (generalized) polynomial chaos. Since [19], both Galerkin and collocation methods have been thoroughly studied and analyzed for several kinds of uncertainty quantification problems, including the time-dependent random diffusion equation [5, 33, 35, 40, 46–48].

The time integration is performed by using an additive semi-implicit Euler method, a special case of so called implicit–explicit (IMEX) methods [2]. In particular, in [49] these methods were proposed for stiff systems resulting from hypersonic transient reactive flows. Although only first-order in time, in [48] the method was shown to be unconditionally stable for various discretizations of parametric and stochastic diffusion equations. We demonstrate that when using locally supported functions to represent the diffusivity, the semi-implicit method is memory-optimal in the sense that storing and solving the resulting linear system requires strictly less space than the solution itself.

After solving the parametric forward problem, obtaining the boundary values for different diffusivities is merely a task of polynomial evaluation. Indeed, in this paper we show that for a second-order approximation of the parameter dependence, the evaluation costs of the parametric solution and its Jacobian matrix do not increase the overall complexity of the inverse problem, if a (regularized) least squares minimization scheme based on Gauss–Newton method and QR decomposition is used to find the solution. These minimization schemes include the standard Gauss–Newton method as well as its trust region counterparts, which from computational point of view are very similar. Higher-order approximations increase the computational burden, but the workload of the inverse problem is still completely independent of the spatial and temporal discretizations of the forward problem. However, for
large-scale nonlinear inverse problems it is arguably recommendable to resort to conjugate gradient iterations instead of QR decompositions when solving the linearized subproblems (see, e.g., [23]). The analysis of combining polynomial approximations and iterative linear least squares solvers is left for future studies.

An algorithmically similar inversion approach, albeit with a Bayesian paradigm, was recently carried out for time-independent EIT problem in [22]. To our knowledge, this is the first time when a parametric spectral solution to the forward problem is utilized for solving an inverse parabolic boundary value problem. The numerical results show that the method is capable of reconstructing the diffusion coefficient based on boundary measurements. Naturally, the quality of the reconstruction depends on the noise level of the measurement. In addition, the method is more suitable for smooth diffusivities. The distinctive feature of the approach is that once the parametric solution is saved, it takes only a few seconds to create the reconstruction from a given set of measurements. Indeed, the obvious advantage of the method is that spatial and temporal discretizations of the forward problem do not affect the complexity of reconstructing the diffusivity.

This paper is organized as follows. In section 2, the precise mathematical formulation of the model problem is given. Then we re-formulate the problem in a parametric sense and discretize the equation in spatial, parametric and temporal dimensions. Section 3 considers the inverse problem from the computational point of view. Numerical examples in two and three spatial dimensions are provided in section 4 and some conclusions are drawn in section 5.

2. Parametric forward model

2.1. Problem setting

Let $\Omega \subset \mathbb{R}^d$, $d \geq 2$, be a bounded domain with Lipschitz boundary $\partial \Omega$ and exterior unit normal $\hat{n} \in [L^\infty(\partial \Omega)]^d$. Furthermore, let $T > 0$ be given. The parabolic initial/boundary value problem considered in this paper is to find $u: \Omega \times [0, T] \rightarrow \mathbb{R}$ satisfying

$$
\begin{cases}
\partial_t u - \nabla \cdot (a \nabla u) = f & \text{in } \Omega \times (0, T), \\
\alpha \nabla u \cdot \hat{n} = g & \text{on } \partial \Omega \times (0, T), \\
u = u_0 & \text{in } \Omega \times \{0\},
\end{cases}
$$

(1)

where $f \in L^2((0, T); L^2(\Omega))$ and $g \in L^2((0, T); H^{-1/2}(\partial \Omega))$ denote the interior source and the boundary flux, respectively, and $u_0 \in L^2(\Omega)$ is the initial condition. The diffusivity (or diffusion coefficient) $a$ belongs to $L^\infty_+(\Omega)$, where

$$
L^\infty_+(\Omega) := \{ a \in L^\infty(\Omega) | \text{ess inf } a > 0 \}.
$$

(2)

In particular, the diffusivity is assumed to be independent of the time variable $t$. The variational formulation of the problem (1) is to find $u$ such that

$$
(\partial_t u, v)_{L^2(\Omega)} + (a \nabla u, \nabla v)_{L^2(\Omega)} = (f, v)_{L^2(\Omega)} + (g, \gamma v),
$$

(3)

accompanied with an appropriate initial condition, holds for all $v \in H^1(\Omega)$ and for almost every $t \in (0, T)$. Here, $\gamma: H^1(\Omega) \rightarrow H^{1/2}_0(\partial \Omega)$ denotes the trace operator and $\langle \cdot, \cdot \rangle$ is the duality pairing between $H^{-1/2}(\partial \Omega)$ and $H^{1/2}_0(\partial \Omega)$. We refer to solving $u$ from (1) or (3) as the forward problem. For the variational form (3) there exists a unique solution $u$ satisfying

$$
u \in L^2((0, T); H^1(\Omega)) \bigcap C^0([0, T]; L^2(\Omega))
$$

(4)

(see, e.g., [14], chapter XVIII).
By inverse problem we mean finding a diffusivity \( a \in L^\infty_+ (\Omega) \) such that the solution \( u \) of the corresponding forward problem matches with given data. More precisely, we consider some given boundary data \( \bar{U} : \partial \Omega \times (0, T) \to \mathbb{R} \) and the corresponding trace \( U(a) := \gamma u \in L^2((0, T); H^{1/2}(\partial \Omega)) \) of the forward solution. It is not obvious whether the equation \( U(a) = \bar{U} \) admits a (unique) solution \( a \); see [28, chapter 9], for some related results with slightly different assumptions. In any case, the inverse problem is ill-posed in the sense that the solution (if it exists) does not depend continuously on the data \( \bar{U} \) in any reasonable metric.

A physical interpretation of the problem (1) could be that \( u \) represents the temperature and \( a \) is the thermal diffusivity in an object \( \Omega \) which has unit heat capacity. The setting is easily extended to a nonhomogeneous but known heat capacity, in which case \( a \) denotes the thermal conductivity [6]. Although we restrict ourselves to the model problem (1), the presented methods are, with minor changes, widely applicable to other types of problems. For example, different boundary conditions, including Dirichlet, Robin and mixed, can be handled easily. Many of the observations apply to elliptic parametric problems as well.

2.2. Parametrization of the diffusivity

In order to treat the inverse problem \( U(a) = \bar{U} \) numerically, we assume that the diffusivity is characterized by a finite number of parameters and an injective mapping \( a : \Theta \to L^\infty_+ (\Omega) \), where \( \Theta \subseteq \mathbb{R}^p \) is a high-dimensional parameter domain, is given. Besides this, our aim in this section is to establish an explicit parameter dependence for the solution \( u \) as well. That is, we seek a numerical solution to the forward problem in the form \( u : \Omega \times [0, T] \times \Theta \to \mathbb{R} \). The existence and uniqueness (similar to (4)) of such solution is obvious.

The most general methods for obtaining the parametric forward solution are based on solving the regular problem (1) or (3) for a set of collocation points in the parameter domain. The full parametric solution would then be written by either using the collocation points as quadrature nodes and projecting the solution to a chosen basis by numerical integration, or by interpolating the solution to the whole domain \( \Theta \) by using a high-dimensional interpolation rule. In both cases, the number of collocation points needed is typically very high and the construction of the points is often based on sparse Smolyak grids [3, 36]. A related strategy is to use Galerkin method with double orthogonal polynomials, which also results in a decoupled system of equations [16].

In this paper, we discretize the parameter domain by the spectral Galerkin approach that yields a large coupled system. We introduce a positive weight function \( w \in L^1(\Theta) \) and employ the corresponding weighted \( L^2 \) spaces. The variational form of the forward problem then becomes to find \( u : \Omega \times [0, T] \times \Theta \to \mathbb{R} \) such that for all \( v \in L^2_\Theta(\partial \Omega; L^2(\Omega)) \) and a.e. in \( (0, T) \)

\[
(\partial_t u, v)_{L^2_\Theta(\partial \Omega; L^2(\Omega))} + (a \nabla u, \nabla v)_{L^2_\Theta(\partial \Omega; L^2(\Omega))} = (f, v)_{L^2_\Theta(\partial \Omega; L^2(\Omega))} + ((g, \gamma v)),
\]

(5)

where \( (\cdot, \cdot) \) denotes the duality pairing between the weighted spaces \( L^2_\Theta(\partial \Omega; H^{-1/2}(\partial \Omega)) \) and \( L^2_\Theta(\partial \Omega; H^{1/2}(\partial \Omega)) \). Here, the functions \( f \) and \( g \) are assumed to be constant with respect to the parameters. We assume that there exists a unique solution

\[
u \in L^2_\Theta(\partial \Omega; L^2((0, T); H^1(\Omega))) \cap C^0([0, T]; L^2(\Omega))).
\]

This is guaranteed, for example, if \( \Theta \) is bounded and there exist positive constants \( a_{\min} \) and \( a_{\max} \) such that
for almost every \( (x, \psi) \in \Omega \times \Theta \).

A convenient way to define the parametrized diffusivity \( a: \Theta \to L^\infty_+(\Omega) \) is to write
\[
a(\psi) = \sum_{p=1}^P a_p \psi_p
\]
for \( \psi \in \Theta \), or slightly more generally
\[
a(\psi) = \bar{a} + \sum_{p=1}^P a_p \psi_p
\]
for some ‘background diffusivity’ \( \bar{a} \in L^\infty_+(\Omega) \). In this paper, we resort to (6) and choose \( \{\psi_p\}_{p=1}^P \subset L^\infty(\Omega) \) to be non-negative and such that they form a partition of unity. Hence, the resulting diffusivity \( a \) is bounded by the extremal values of the parameters, i.e.
\[
\inf_{\psi \in \Theta} \min_{1 \leq p \leq P} \partial_p \psi \leq a(\xi) \leq \sup_{\psi \in \Theta} \max_{1 \leq p \leq P} \partial_p \psi \quad \text{in} \ \Omega
\]
for all \( \xi \in \Theta \). Naturally, the positivity requirement (2) imposes restrictions on choosing the parameter domain \( \Theta \) once the functions \( \psi_p \) are fixed.

One possible family \( \{\psi_p\}_{p=1}^P \) is a set of B-splines, which for a rectangular domain \( \Omega \) are easy to construct as tensor products of univariate splines. Let us briefly recap the basic properties of B-splines, following [26]. A standard univariate uniform B-spline of degree \( s \in \mathbb{N} \) can be defined recursively as a convolution
\[
b_s(x) \equiv \int_{-\infty}^{\infty} b_{s-1}(x-y)b_0(y) \, dy,
\]
where \( b_0(x) \) is the indicator function of the interval \([0, 1)\). We see that \( b_s \in C^{s-1}(\mathbb{R}) \) and \( \text{supp}(b_s) = [0, s+1] \). The splines \( b_0(x), b_1(x), \ldots, b_s(x) \) form a partition of unity on the interval \([0, 1)\). The multivariate splines can be constructed as
\[
b_{s,d}(x) \equiv \prod_{i=1}^d b_s(x_i)
\]
and again \( b_{s,d} \in C^{s-1}(\mathbb{R}^d) \). Transforming the standard splines to form a desired partition of unity is elementary. See [26] for constructing splines for nonrectangular domains. For polygonal and polyhedral domains one can also use piecewise linear functions \( \psi_p \) such as those used in finite element solvers.

It can be shown that if \( u \) and \( \tilde{u} \) are two forward solutions corresponding to arbitrary diffusivities \( a \) and \( \bar{a} \), respectively (in \( L^\infty_+(\Omega) \), but not necessarily of the form (6)), then there exists \( C_1 > 0 \) such that
\[
\|u - \bar{u}\|_{L^2(\Omega)} \leq C_1 \|a - \bar{a}\|_{L^\infty(\Omega)},
\]
see, e.g., [25]. On the other hand, the approximation error result for tensor product B-splines \( \{\psi_p\}_{p=1}^P \) states that if \( S \) contains all the functions of the form (6) with \( \Theta = \mathbb{R}^d \), then
\[
\inf_{a \in S} \|a - \bar{a}\|_{L^\infty(\Omega)} \leq C_2 \kappa^{s+1} \max_{1 \leq i \leq d} \|\partial_i^{s+1} \bar{a}\|_{L^\infty(\Omega)}
\]
for any diffusivity \( \bar{a} \in L^\infty(\Omega) \) and spline degree \( s \geq 0 \), where \( C_2(s) > 0 \) and \( \kappa \sim P^{-1/d} \) is a characteristic distance between spline knots [13].
Substituting (6) into (5) leads to
\[(\partial_t u, v) + \sum_{p=1}^{P} (t_p \psi_p \nabla u, \nabla v) = (f, v) + ((g, \gamma v)),\] (8)

where we have dropped the subscripts for brevity, and where the projection \(t_p: \Theta \rightarrow \mathbf{R}\) is defined by \(t_p(\theta) = \theta_p\). In the following, we discretize the spaces \(L^2_w(\Theta)\) and \(H^1(\Omega)\) in order to recast (8) as a matrix equation, which is then solved by performing a finite difference discretization of the temporal domain.

For a separable Hilbert space \(H\), the isomorphism \(L^2_w(\Theta; H) \approx L^2_w(\Theta) \otimes H\) holds, if \(L^2_w(\Theta)\) is also separable (see, e.g., [40, section B.3] and references therein). Now if \(\{\phi_i\}\) and \(\{\varphi_j\}\) are countable bases for \(H^1(\Omega)\) and \(L^2_w(\Theta)\), respectively, then \(\{\phi_i \varphi_j\}\) forms a basis for \(L^2_w(\Theta; H^1(\Omega))\). This suggests looking for the numerical solution to the parametric forward problem in the form
\[u_{M,N}(x, t, \theta) = \sum_{i,j=1}^{M,N} \hat{a}_{ij}(t) \phi_i(x) \varphi_j(\theta),\]

(where the finite-dimensional bases are not necessarily subsets of the aforementioned countable bases) with the time-independent test functions \(v \in L^2_w(\Theta; H^1(\Omega))\) having the form \(v_{ij} = \phi_i \varphi_j\).

The semi-discrete equation for the parametric forward problem (8) can now be written as
\[\partial_t \hat{u}(t) + A \hat{u}(t) = \hat{f}(t),\] (9)

where \(\hat{u}: (0, T) \rightarrow \mathbf{R}^{MN}\) denotes the vector of unknown coefficients
\[\hat{u}(t) := [\hat{u}_{1,1}(t), \hat{u}_{2,1}(t), \ldots, \hat{a}_{M,1}(t), \hat{a}_{1,2}(t), \ldots, \hat{a}_{M-1,N}(t), \hat{u}_{M,N}(t)]^T,\] (10)

\(B \in \mathbf{R}^{MN \times MN}\) is the parametric mass matrix defined by its entries
\[B_{i,j,k,l} = (\varphi_j, \varphi_l)_{L^2_w(\Theta)}(\phi_i, \phi_k)_{L^2_w(\Theta)}\] (11)

and \(A \in \mathbf{R}^{MN \times MN}\) is the parametric stiffness matrix satisfying
\[A_{i,j,k,l} = \sum_{p=1}^{P} (t_p \phi_i, \phi_j)_{L^2_w(\Theta)}(\psi_p \varphi_l, \varphi_k)_{L^2_w(\Theta)}\] (12)

The vector \(\hat{f}: (0, T) \rightarrow \mathbf{R}^{MN}\) on the right-hand side is
\[\hat{f}_{k,l}(t) = (1, \varphi_l)_{L^2_w(\Theta)}((f(t), \varphi_k)_{L^2_w(\Theta)} + (g(t), \phi_k)).\] (13)

2.3. Finite element discretization

We construct a finite-dimensional subspace \(V_h \subset H^1(\Omega)\) by following the standard FEM procedures [32]. The finite element basis functions are denoted by \(\{\phi_i\}_{i=1}^{M}\) and \(h > 0\) is the mesh size parameter of a regular enough mesh. The contribution of the spatial parts in (11) can be thought of as the symmetric positive-definite mass matrix \(B' \in \mathbf{R}^{M \times M}\) defined by
\[B_{i,k} = (\phi_i, \phi_k)_{L^2_w(\Theta)}\]

For a nonhomogeneous heat capacity, the elements of the mass matrix should be modified accordingly.

The spatial term in (12) involves symmetric positive-semidefinite matrices of the stiffness type
$$A_{i,k}^{(p)} = (\psi_p \nabla \phi_i, \nabla \phi_k)_{L^2(\Omega)}.$$  

Note that the individual matrices $A^{(p)}$ may be very sparse and rank-deficient if the functions $\psi_p$ are locally supported. Let us denote by $\text{nnz}(V)$ the number of nonzero elements of an arbitrary matrix $V$ and introduce a sparsity quantity

$$\eta := \frac{\sum_{p=1}^P \text{nnz}(A^{(p)})}{\text{nnz}(A^n)},$$  \hspace{1cm} (14)

where $A^n \in \mathbb{R}^{M \times M}$ is the standard stiffness matrix defined by $A_{i,k}^n = (\nabla \phi_i, \nabla \phi_k)$. Clearly, $1 \leq \eta \leq P$. Note also that

$$\sum_{p=1}^P A^{(p)} = A^n,$$

due to the partition of unity property of $\{\psi_p\}_{p=1}^P$.

We do not consider the convergence with respect to $h$ in detail. However, recall that the familiar a priori error estimate

$$\|u - u_h\|_{H^1(\Omega)} \leq Ch^{r-1}\|u\|_{H^r(\Omega)}$$

for the numerical solution $u_h$ of a canonical elliptic problem depends on $r \geq 1$, for which

$$r \leq \sup_{q \in \mathbb{R}} \{q|u \in H^q(\Omega)\},$$

and for which the employed finite element type also sets an upper bound. Since for a noncontinuous coefficient function (corresponding to zeroth order splines and $s = 0$), the solution $u$ is not in $H^2(\Omega)$, the convergence becomes sublinear and thus deteriorates even for piecewise linear FEM basis functions. Similarly, higher order FEM basis functions should be used only if the spline degree is high enough. In practice, however, the cardinality of the finite element space is much larger than the number of splines. Thus, the solution may converge rapidly regardless of the nonsmoothness of the diffusivity. The compatibility of the diffusivity representation and finite element discretization, in terms of forward computations, is discussed in [34].

### 2.4. Spectral Galerkin method

Next, we discretize the space $L^2_\omega(\Theta)$ by introducing an $N$-dimensional subspace $W \subset L^2_\omega(\Theta)$. We adopt the spectral Galerkin method, and therefore choose basis functions $\{\varphi_j\}_{j=1}^N$ that are orthogonal in the $L^2_\omega$-sense. For convenience, we also assume normality, thus

$$(\varphi_j, \varphi_i)_{L^2_\omega(\Theta)} = \delta_{ij},$$

where $\delta$ is the Kronecker delta. It immediately follows that the matrix corresponding to the parametric part of (11) is the identity matrix $I \in \mathbb{R}^{N \times N}$ and therefore $B = I \otimes B^r$.

We only consider hyperrectangular parameter domains, and for simplicity assume that the domain is a hypercube $\Theta = E^P$ for some interval $E \subset \mathbb{R}$. The $P$-variate basis functions can then be easily constructed as tensor products of univariate polynomials $\{\tilde{\varphi}_n\}_{n=0}^P$, where $n \in \mathbb{N}_0$ is the maximum univariate degree. More precisely, we define

$$\varphi_j(\theta) = \prod_{p=1}^P \tilde{\varphi}_{n_p}(\theta_p),$$  \hspace{1cm} (15)
for \( j = 1, \ldots, N \). In addition, the weight function is assumed to be separable in the sense that
\[
 w(\overline{\theta}) = \prod_{p=1}^{P} \tilde{w}_p(\overline{\eta}_p).
\]
(Often we may even have \( \tilde{w}_p = \tilde{w} \) for all \( p = 1, \ldots, P \).) The univariate degrees \( \Lambda_{j,p} \) can be stored in a matrix \( \Lambda = \mathbb{N}_0^{N \times P} \) which has \( N \) (yet unspecified) distinct rows. For convenience, we assume that the first row contains only zeros, that is, \( \varphi_1 \equiv \|w\|_{L^2(\Theta)}^{-1/2} \) is the constant polynomial.

Due to orthogonality, all but the first block in the right hand side vector (13) vanish. The stiffness matrix (12) can be written as
\[
 A = \sum_{p=1}^{P} (Y^{(p)} \otimes A^{(p)}),
\]
where the symmetric matrices of size \( N \times N \) involving the actual parameters are defined by
\[
 Y_{j,k}^{(p)} = (i_p \varphi_j, \varphi_k)_{L^2(\Theta)}
\]
for \( p = 1, \ldots, P \).

For a fixed univariate degree \( n \), the largest possible degree matrix \( \Lambda \) contains \( N = (n + 1)^P \) rows, which is too much for most practical purposes. A widely used alternative is to limit the row sum of the degree matrix to equal \( n \). This results in a so called total degree polynomial space, which is spanned by
\[
 \{ \varphi_j \}_{j=1}^{N} = \left\{ \prod_{p=1}^{P} \varphi_{j_p}(\overline{\eta}_p) \left| \sum_{p=1}^{P} r_p \leq n \right. \right\}.
\]
We collect some combinatorial results related to the total degree space in the following theorem. Hereafter, a square matrix \( V \) whose diagonal entries are replaced by zeros, is denoted by \( \text{offdiag}(V) \).

**Theorem 2.4.** Let \( \Lambda \in \mathbb{N}_0^{N \times P} \) be a degree matrix corresponding to the polynomial basis of a total degree space of \( P \) variables and a total degree \( n \), where \( 1 \leq n \ll P \). Then

(a) \( N = \binom{P+n}{n} = \frac{(P+n)!}{P! n!} = \frac{P^n}{n!} + \mathcal{O}(P^{n-1}) \)

(b) \( \text{nnz}(\Lambda) = P \binom{P+n-1}{n-1} = \frac{P^n}{P + n} N = \frac{P^n}{(n-1)!} + \mathcal{O}(P^{n-1}) \)

(b) \( \text{nnz}(\text{offdiag}(Y^{(p)})) = \frac{2}{P} \text{nnz}(\Lambda) = \frac{2P^{n-1}}{(n-1)!} + \mathcal{O}(P^{n-2}) \)

and the positions of the nonzero elements of \( \text{offdiag}(Y^{(p)}) \) are disjoint for \( 1 \leq p \leq P \).

**Proof.** We skip the proof here but refer to, e.g., [15, 19] for similar results with proofs. \( \square \)
In what follows, we will use the result \( \text{nnz}(A) = \mathcal{O}(P^n) \) from part (b) of the theorem. This is, of course, quite elementary consequence of part (a), since the definition of the total degree space immediately yields \( \text{nnz}(A) \leq nN \). Note that the quantities in parts (a) and (b) of the theorem are symmetric in terms of \( P \) and \( n \) (as is part (c) after summing over \( p \)). Thus, the asymptotic formulae for the case \( n \to \infty \) can be obtained by switching the roles of \( P \) and \( n \).

Some alternatives for the total degree spaces can be found in [12]. We also mention that due to the locality of the splines, it seems reasonable to ignore some cross-terms including variables that correspond to coefficients of splines being supported far away from each other. This idea is not completely heuristic, see the asymptotic formula in elliptic case with small inclusions [1, chapter 5]. We have also numerically observed this kind of behavior.

Ordinarily, one uses Legendre polynomials and a constant weight function, but other choices are possible as well. We refer to [18] for discussion on orthogonal polynomials and [9] for more detailed analysis of different types of spectral approximations. The convergence rate of spectral approximation depends on the smoothness of the forward solution with respect to the parameters. The affine representations (6) and (7) are known to result in an analytic parameter dependence in \( \Omega \times (0, T) \), see [25, 35]. On the other hand, when the number of parameters is large, it is only possible to use a low polynomial degree and thus the asymptotic convergence is not the main interest. Finally, we mention that the rest of this paper is equally applicable to the case where the parameter dependence is approximated with a spectral collocation method. In some cases, for example when the diffusivity parametrization is based on a boundary curve between two different diffusivity values, the collocation method may be significantly more straightforward to formulate and implement.

2.5. Time integration

Let us continue by discussing how the semi-discrete equation (9) can be solved in time. In principle, any time integration method can be used to solve the resulting system of ordinary differential equations. However, diffusion equations typically require very small time steps if an explicit time integration method is used. This applies to parametric equations as well, see, e.g., [38] for some eigenvalue bounds. Implicit methods, on the other hand, require the solution of a full system of size \( MN \) having a nontrivial sparsity structure.

Explicit methods with larger stability regions for parabolic equations can be derived from the class of so called Runge–Kutta–Chebyshev methods [45]. An alternative to explicit and implicit methods are semi-implicit or IMEX methods [2, 49]. Here we resort to a first-order semi-implicit Euler method, which in [48] was shown to be unconditionally stable for parametric diffusion equations with symmetric Jacobi weights (in particular, with a constant weight and Legendre polynomials). We already know that the mass matrix \( B \) is block-diagonal and actually has constant blocks. The feasibility of the semi-implicit method is based on the diagonal dominance of the matrices \( Y^{(p)} \) and the decomposition

\[
A = \mu I \otimes A + \sum_{p=1}^{P} (\text{offdiag}(Y^{(p)}) \otimes A^{(p)}) = D + S,
\]

where \( D \in \mathbb{R}^{MN \times MN} \) is a block-diagonal matrix with constant blocks and \( \mu > 0 \). This decomposition follows somewhat directly if the diffusivity is parametrized as in (7) with \( \bar{a} \equiv \mu \), see also [15].
Let us denote \( \hat{u}^k := \hat{u}(k\delta) \), where \( k \in \mathbb{N} \) and \( \delta > 0 \) is a time step. Furthermore, let
\[
\hat{r}^{(k+1/2)} := \frac{1}{\delta} \int_{k\delta}^{(k+1/2)\delta} \hat{r}(t) \, dt
\]
be the mean value of the right-hand side vector (13) over one time step. Starting from an initial vector \( \hat{u}^0 \), which similar to \( \hat{r}(t) \) contains in general only \( M \) nonzero values, the semi-implicit Euler method presented in [48] can be written as
\[
(B + \delta D)\hat{u}^{(k+1)} = (B - \delta S)\hat{u}^k + \delta \hat{r}^{(k+1/2)}. \tag{16}
\]

Let us define
\[
\text{mat}(\hat{u}(t)) := \begin{bmatrix}
\hat{u}_{1,1}(t) & \cdots & \hat{u}_{1,N}(t) \\
\vdots & \ddots & \vdots \\
\hat{u}_{M,1}(t) & \cdots & \hat{u}_{M,N}(t)
\end{bmatrix} \in \mathbb{R}^{MN \times MN}
\]
as the matricization of the vector \( \hat{u}(t) \) defined in (10). Correspondingly, we define the vectorization \( \text{vec(mat}(\hat{u}(t))):= \hat{u}(t) \in \mathbb{R}^{MN} \). Due to the block structure of the matrices \( B \) and \( D \), the algorithm (16) can be efficiently implemented based on the following rules:

1. Compute \( \Xi = B^* \text{mat}(\hat{u}^k) \).
2. Compute \( \xi = \text{vec}(\Xi) - \delta S\hat{u}^k + \delta \hat{r}^{(k+1/2)} \).
3. Solve \( \Upsilon \in \mathbb{R}^{d \times N} \) from \( (B^* + \delta \mu A^*)\Upsilon = \text{mat}(\xi) \).
4. Set \( \hat{u}^{k+1} = \text{vec}(\Upsilon) \).

The solution to the system in the third step can be obtained by explicitly storing the inverse of the (time-independent) matrix \( B^* + \delta \mu A^* \) or storing the Cholesky factor and performing triangular substitutions. For large \( M \), iterative methods such as the deflated conjugate gradient method may also be used [39].

The matrices \( B^* \) and \( A^* \) have \( O(M) \) nonzero elements and their sparsity structure resulting from standard FEM discretization is well-studied and can be exploited. The only matrix of size \( MN \times MN \) that has to be stored in the proposed algorithm is \( S \), which is very sparse. Indeed, theorem 2.4 and the definition (14) yield
\[
\text{nnz}(S) = \sum_{p=1}^{P} \text{nnz(offdiag}(Y^{(p)})\text{nnz}(A^{(p)}) = \frac{2nN\eta \text{nnz}(A^*)}{P + n}.
\]

If \( \eta = O(P) \), then \( \text{nnz}(S) = O(MN) \). However, if \( P \) is large and the functions \( \psi_p \) are splines or otherwise supported on a small region, then \( \eta = O(1) \) and the matrix \( S \) has essentially less nonzero elements than the vectors \( \hat{u}^k \) for \( k > 0 \).

Naturally, the vectors \( \hat{u}^k \) can be discarded after computing the next solution \( \hat{u}^{(k+1)} \). Thus, arbitrarily small time steps can be used without imposing massive memory requirements, and the fact that the method is only first-order, is not necessarily an issue. As suggested in [48], the proposed method can be modified to perform a Jacobi iteration for parametric or stochastic elliptic equations. In addition, we note that preconditioned Krylov subspace methods for time-independent parametric or stochastic problems may employ structures that are similar to those presented here. See, e.g., [44] and the references therein for more information on that subject.
3. Estimating parameters from boundary data

In this section, we consider the inverse problem of determining the diffusion coefficient from boundary measurements. As presented in section 2.1, the continuous formulation of the inverse problem is to find a diffusivity \( a \in L^\infty_\gamma(\Omega) \) such that \( \gamma u \), which is the trace of the solution corresponding to \( a \), equals (or is close to) the measurement \( \tilde{U} : \partial \Omega \times (0, T) \to \mathbb{R} \).

A practical measurement contains only finitely many, say \( Q \), values. That is, we consider a measurement vector \( \tilde{U} \in \mathbb{R}^Q \) satisfying \( \| x_{Ut} \|_2 \), \( q \in \{1, \ldots, Q\} \) defines the physical coordinates of the observations. We denote the measurement only as an approximation of the temperature due to unavoidable errors in measurements and uncertainties in the problem setting. The parametric numerical solution corresponding to the coordinates \( x_{Ut} \), \( q \in \{1, \ldots, Q\} \) can be written as \( U : \Theta \to \mathbb{R}^Q \), which satisfies

\[
U(\theta) = \begin{bmatrix}
\mu_{M,N}(x^{(1)}, t^{(1)}, \theta) \\
\vdots \\
\mu_{M,N}(x^{(Q)}, t^{(Q)}, \theta)
\end{bmatrix}.
\]

(form 17)

Formally, the inverse problem can now be written as a parameter estimation problem

\[
\arg \min_{\theta \in \Theta} \| U(\theta) - \tilde{U} \|_2^2,
\]

where \( \| \cdot \|_2 \) denotes the Euclidean norm and the diffusivity \( a \) can be computed from (6). Due to the ill-posedness, however, the minimization has to be regularized in order to avoid meaningless reconstructions.

3.1. Regularized nonlinear least squares

Let us briefly sketch a simple Gauss–Newton algorithm with line search for a minimization problem of the type (18). Assume \( \Theta = \mathbb{R}^P \) for a moment. Starting from \( \theta^{(0)} \in \Theta \) and \( k = 1 \), the algorithm produces a sequence of parameter vectors according to the following steps:

1. Solve a linear least squares problem

\[
\Delta_{opt} := \arg \min_{\Delta \in \mathbb{R}^P} \| J_U(\theta^{(k-1)}) \Delta + U(\theta^{(k-1)}) - \tilde{U} \|_2^2.
\]

2. Solve a one-dimensional optimization problem

\[
\alpha_{opt} := \arg \min_{\alpha \in \mathbb{R}_+} \| U(\theta^{(k-1)}) + \alpha \Delta_{opt} - \tilde{U} \|_2^2.
\]

3. Set \( \theta^{(k)} = \theta^{(k-1)} + \alpha_{opt} \Delta_{opt} \) and increase \( k \) by one.

These steps are repeated until a suitable stopping criterion is satisfied. Here, \( J_U : \mathbb{R}^P \to \mathbb{R}^{Q \times P} \) is the Jacobian matrix of the mapping \( U \). The linear least squares problem in the first step is usually solved by QR decomposing the Jacobian [20, 37], although it is also possible to make the algorithm more efficient by employing the conjugate gradient method in case \( P \) is large [23, 30, 31]. (As our spectral Galerkin method does not allow very large \( P \), we exclude such considerations.) Assuming that the Jacobian \( J_U \) and the vector \( U \) have already been evaluated at \( \theta^{(k-1)} \), the computational complexity of the QR decomposition is \( \mathcal{O}(QP^2) \). After that, the search direction \( \Delta_{opt} \) can be obtained easily with triangular substitution. The
line search of the second step requires only few evaluations of \( U \), since an approximate solution for \( \alpha_{\text{opt}} \) is usually enough.

We refer to [37] for more detailed discussion about nonlinear least squares algorithms. If \( \Theta \neq \mathbb{R}^P \), we also need to specify and implement constraints, which we ignored above. A common alternative to the standard Gauss–Newton algorithm (or the one with line search) is the Levenberg–Marquardt method, which essentially consists of the same subproblems as Gauss–Newton: evaluating \( U \) and its Jacobian \( J_U \), and solving a linear least squares problem involving the Jacobian.

If the problem (18) is replaced by a Tikhonov regularized version

\[
\arg \min_{\vartheta \in \Theta} \{ \| U(\vartheta) - \bar{U} \|^2_2 + \lambda^2 \| G(\vartheta) \|^2_2 \} = \arg \min_{\vartheta \in \Theta} \left\| \begin{bmatrix} U(\vartheta) - \bar{U} \\ \lambda G(\vartheta) \end{bmatrix} \right\|_2^2 ,
\]

the same principles still apply, assuming that the regularization function \( G \) is easy to evaluate and differentiate, e.g., \( G \) is a matrix. A Bayesian interpretation for the inverse problem often results in a similar minimization problem as (19), if the maximum a posteriori estimate is sought. It is also possible to employ more advanced methods such as the iteratively regularized Gauss–Newton method and Lepski balancing principle [7], while still having essentially the same subproblems.

### 3.2. Evaluating polynomials and derivatives

The nonlinear mapping \( U : \Theta \to \mathbb{R}^Q \) in (17) can be decomposed as

\[
U(\vartheta) = V \varphi(\vartheta),
\]

where the matrix \( V \in \mathbb{R}^{Q \times N} \) satisfies

\[
V_{q,j} = \sum_{i=1}^M a_{ij}(t_q) \phi_i(x_q)
\]

and the nonlinear part \( \varphi : \Theta \to \mathbb{R}^N \) is defined in (15). The matrix \( V \) can be constructed in advance, that is, before performing any measurements or optimization. On the other hand, the multivariate polynomials can be evaluated according to

\[
\varphi_j(\vartheta) = (\bar{\varphi}_{\vartheta_0})^{p_j - |L_j|} \prod_{p \in L_j} \bar{\varphi}_{\lambda_{p,j}}(\vartheta_p),
\]

where

\[
L_j = \{ p | \lambda_{j,p} = 0 \}
\]

contains the indices to nonconstant univariate polynomials. For a total degree space, the number of such indices must be \( |L_j| \leq n \) for each \( j = 1, \ldots, N \). In this case, and assuming \( n = O(1) \), the evaluation of \( \varphi \) requires \( O(N) \) floating point operations. The total complexity of evaluating \( U \) is thus determined by the matrix-vector multiplication (20), which takes \( O(NQ) \).

The Jacobian matrix \( J_U : \Theta \to \mathbb{R}^{Q \times P} \) can be written as

\[
J_U(\vartheta) = V J_\varphi(\vartheta),
\]

where the basis Jacobian \( J_\varphi : \Theta \to \mathbb{R}^{N \times P} \) contains the partial derivatives of the multivariate polynomials. Note that for a general argument \( \vartheta \in \Theta \), the basis Jacobian has exactly the same sparsity structure as the degree matrix \( A \) defining the underlying polynomial space. For that reason, the evaluation of \( J_\varphi \) becomes quite cheap. Indeed, for any \( 1 \leq j \leq N \) and \( p \in L_j \), we
have

\[ J(p)(\Theta)_{i,p} = (\varphi_0)^{p - |L_j|} \varphi^\prime_{\lambda_j}(\partial_p) \prod_{q \in L_j \backslash p} \varphi_{\lambda_q}(\partial_q), \]

and for \( p \notin L_j \), the corresponding entries in the basis Jacobian \( J_p \) vanish. Again, if \( |L_j| = O(1) \) for each \( j \), the evaluation of \( J_p \) requires \( O(N) \) operations. Due to the sparsity the matrix-matrix product in (21) takes only \( O(NQ) \).

From theorem 2.4 we immediately get the following:

**Corollary.** The computational cost of evaluating the parametric solution \( U \) and all its first-order partial derivatives is \( O(P^2) \), if the polynomial space of total degree \( n \) is used to discretize the parameter domain. In particular, choosing \( n = 2 \) yields the same complexity as the QR decomposition in a Gauss–Newton step.

In other words, a quadratic approximation of the parameter dependence does not pose a bottleneck for the efficiency of a Gauss–Newton or similar minimization scheme. The result is nontrivial in the sense that a naïve finite difference approximation of the Jacobian \( J_p \) would require \( P \) function evaluations and thus have a complexity of \( O(P^3) \). Let us also emphasize at this point that the complexity of the inverse problem is completely independent of spatial and temporal discretizations of the forward problem and it is also irrelevant whether the spectral approximation was obtained with the Galerkin or collocation method.

The linearized subproblems in a Gauss–Newton scheme can also be solved by, e.g., a conjugate gradient iteration instead of the QR decomposition. For general complexity analysis of such nested iterations, see [30]. Extending the analysis to include a polynomial surrogate model is beyond the scope of this article.

Notice that the forward parametric solution can be solved with a larger spectral basis than what is used when evaluating the solution and its Jacobian. This is useful, since it is difficult to choose in advance the optimal subset of multivariate polynomials. After having the parametric solution at hand, one can simply discard those polynomials that correspond to columns of the matrix \( V \) having smallest (Euclidean) norm. It is also possible to use different subsets of polynomials for \( \varphi \) and for \( J_p \). Having fewer polynomials for the Jacobian results in the so called perturbed Gauss–Newton method [21]. In particular, it is even possible to solve a single regular forward problem at each step while approximating the Jacobian with the pre-computed polynomials.

Finally, note that it is also possible to treat the inverse problem that is based on several different initial conditions, boundary fluxes or other components of the problem setting. This merely requires stacking the corresponding matrices \( V \) on top of each other.

### 4. Numerical examples

The proposed method is demonstrated with simulated boundary data. First, we solve a two-dimensional parametric forward problem corresponding to \( P = 14^2 = 196 \) bi-quadratic B-splines that are uniformly spaced so that they form a partition of unity on the unit square \( \Omega = (0, 1)^2 \). We choose the parameter interval \( E = (1/2, 2) \), constant weight \( w \) and employ polynomial space of total degree \( n = 2 \), resulting to \( N = 19,503 \) multivariate Legendre polynomials in accordance with part (a) of theorem 2.4. We assume zero initial condition \( u_0 \equiv 0 \) and also set \( f \equiv 0 \). For the horizontal boundaries, we assume homogeneous Neumann conditions, i.e., \( g|_{y \in (0,1)} \equiv 0 \), whereas the vertical boundaries satisfy \( g|_{u=0} = -20x \) and
This corresponds to the case where two sides of a square-shaped object are insulated and two sides are heated or cooled with a heat flux which is linear in time. The spatial discretization is performed with \( M = 37^2 = 1369 \) uniformly spaced piecewise linear FEM basis functions (corresponding to 2592 triangular elements) and for the semi-implicit Euler method we choose the time step \( \delta = 10^{-3} \).

The boundary data is generated by using \( \tilde{M} = 129^2 = 16,641 \) FEM basis functions and the second order Crank–Nicolson time integration method with a step length \( \tilde{\delta} = 10^{-3} \). The measurement consists of \( Q_s = 36 \) spatial points that are uniformly distributed across the boundary (including corners), and of \( Q_t = 13 \) time instances \( t^\theta \in \{0.01, 0.05, 0.09, \ldots, 0.49\} \). Thus, the measurement vector \( \tilde{U} \) has \( Q = Q_s Q_t = 468 \) elements. For each value, we add independent zero mean Gaussian noise realizations with standard deviation \( \tilde{s} = \sigma_0 \cdot \max_{1 \leq j \leq Q} \tilde{U}_j \) with some \( \sigma_0 > 0 \).

The reconstructions are computed by minimizing (19) with the \textit{lsqnonlin} function of Matlab. The default algorithm \textit{trust-region-reflective} handles bound constraints that are chosen to agree with the parameter domain \( \Theta = E^P \). The regularization function \( G \) is chosen to be a discretized Laplace operator (i.e., a matrix which in one dimension would be tridiagonal with values \(-1, 2 \) and \(-1\)) so that the minimization prefers smooth diffusivities. The regularization parameter \( \lambda \) is set such that the Morozov discrepancy principle

\[
\| U(\theta) - \tilde{U} \|_2 \approx \sqrt{Q} \sigma
\]

holds for the minimizing vector \( \theta \). Because the minimization process is very fast, the adjustment of \( \lambda \) can be done, for example, by trial-and-error.

**Figure 1.** Two-dimensional target diffusivity \( \tilde{a} \) (23) on the left and the reconstruction \( a \) on the right \((\sigma_0 = 0.001)\).

**Figure 2.** Two-dimensional piecewise constant target diffusivity \( \tilde{a} \) on the left and the reconstruction \( a \) on the right \((\sigma_0 = 0.001)\).
In the first two-dimensional example we consider the boundary data corresponding to a smooth diffusivity
\[
\tilde{a}(x) = 1.25 + \frac{\sin(6x_1) \cos(4x_2)}{2},
\]
which satisfies \(0.75 \leq \tilde{a}(x) \leq 1.75\) for all \(x \in \Omega\). Figure 1 shows the target diffusivity together with the reconstruction. Here, the noise parameter is \(\sigma_0 = 0.001\) and for the regularization we use \(\lambda = 0.025\). We see that the quality of the reconstruction is good. A piecewise constant diffusivity taking values \(1/2\) and \(3/2\) is reconstructed in figure 2 by using the aforementioned values for the noise and regularization. As expected, the reconstruction is far from exact in this nonsmooth case, because the regularization favors smooth diffusivities and such a piecewise constant diffusivity is also impossible to represent by using bi-quadratic B-splines.

Let us also consider the approximation error
\[
\|U(\theta) - \tilde{U}_\theta\|_2,
\]
where \(\theta\) is the minimizing vector as in (22) and \(\tilde{U}_\theta\) denotes the simulated high-accuracy noiseless \((\sigma_0 = 0)\) boundary data that is obtained by choosing the coefficient \(\tilde{a}\) to be the just reconstructed diffusivity. This error results from truncating the polynomial expansion (i.e., having a finite \(N\)) and also from the less accurate spatial and temporal discretizations of the parametric solution. In the examples shown in figures 1 and 2, the error values (24) are 0.11 and 0.10, respectively, which in these low-noise cases are slightly larger than \(\sqrt{Q}\sigma\) in (22).
Figure 5. Left: slices of the three-dimensional target diffusivity (25) in the unit cube. Right: corresponding slices of the reconstruction ($\sigma_0 = 0.01$).
However, although this approximation error is not accounted for in (22), no instability of the reconstructions was observed in any of the numerical tests.

Some other smooth target diffusivities and their reconstructions are shown in figures 3 and 4. Also in these cases the target diffusivity values lie between 1/2 and 2. Now the noise parameter is considerably larger, namely $\sigma_0 = 0.02$. In order to approximately satisfy the Morozov discrepancy principle (22), we set $\lambda = 0.4$. The reconstructions are still qualitatively correct. The approximation errors (24) related to figures 3 and 4 are 1.20 and 0.13, respectively, which are now smaller than the noise level in (22). The larger error in the third example may be explained by the target diffusivity values that are quite low; the parametric solution is usually most accurate when the diffusivity values are in the middle of the parameter interval $E$.

In our three-dimensional example we have $P = 6^3 = 216$ trilinear B-splines that form the partition of unity in the unit cube $\Omega = (0, 1)^3$. As in the two-dimensional case, we use $E = (1/2, 2)$, constant weight and $n = 2$, which now results in $N = 23,653$. The Neumann boundary conditions on two opposing sides are $g|_{y=0} = -40r$ and $g|_{y=1} = 40r$, whereas the remaining four faces have homogeneous Neumann boundary conditions. The initial value and the forcing term are set to zero as in the two-dimensional case. In three-dimensional case, we use a piecewise linear finite element mesh with $M = 26^3 = 17,576$ nodes and 93,750 tetrahedra. The time step of the semi-implicit Euler method is still $\delta = 10^{-3}$.

The three-dimensional measurement data is generated by using $M = 65^3 = 274,625$ piecewise linear basis functions, Crank–Nicolson time integration with a time step of $\delta = 10^{-3}$ and a diffusivity

$$\hat{a}(x) = 1.25 + (0.5 - x_3)\sin(6x_1)\cos(4x_2).$$ (25)

The measurement consists of $Q_s = 152$ spatial locations that are uniformly spaced across the boundary of the cube, and of $Q_t = 13$ time instances as before, so that $Q = Q_sQ_t = 1976$. The noise model is the same as in the two-dimensional case with $\sigma_0 = 0.01$ and for the regularization we choose $\lambda = 0.09$. Figure 5 shows that the reconstruction is qualitatively correct in this three-dimensional setting.

All computations were performed by using Matlab R2014b. Computationally the most demanding case is the three-dimensional parametric forward problem in which $MN \approx 4 \times 10^8$ and quite a large amount of memory is needed even to store the intermediate vector during the time integration. On the other hand, the measurement consists of only $Q \approx M/10$ values and thus the matrix $V \in \mathbb{R}^{Q \times N}$ can be handled easily even in the three-dimensional case. For a fixed regularization parameter, the reconstruction itself took at most a few seconds in all numerical experiments on an up-to-date desktop computer. This is the only step that cannot be performed offline, that is, prior to the measurements. Further improvements could still be expected by implementing a customized least squares solver and fine-tuning the stopping criterion for the iterative optimization. Moreover, combining our method with a state-of-the-art iteratively regularized Gauss–Newton method with inner conjugate gradient iteration is left for future studies.

5. Conclusions

We have studied a time-dependent parametric partial differential equation and the related inverse boundary value problem. The parametric forward problem was solved by using the spectral Galerkin method in the parameter domain, FEM in the spatial domain and a semi-
implicit Euler method in the time interval. The inverse problem was interpreted as a nonlinear least squares problem accompanied with a Tikhonov regularization.

It was shown that having a quadratic approximation for the parameter dependence results in an efficient inverse algorithm, where the evaluation of the multivariate polynomials and their derivatives does not constitute an essential bottleneck for a Gauss–Newtonian least squares method, if QR decompositions are used for solving the linear subproblems. In particular, the inverse problem can be solved independently of the physical discretization of the forward problem. The numerical results indicated that the quadratic approximation is indeed accurate enough for qualitatively correct reconstructions. See [30, 31] for alternative approaches that are more efficient if the number of parameters is large and the QR decomposition becomes too expensive.

The proposed method is versatile and can be applied to elliptic problems, including EIT, as well. The future research will concentrate on different regularization techniques such as sparsity promoting terms that do not satisfy the differentiability assumptions posed here. In addition, different parametrizations will be studied.

Acknowledgments

This work was supported by the Academy of Finland (decision 267789) and the Finnish Foundation for Technology Promotion TES. The author would also like to thank professor Nuutti Hyvönen and the anonymous referees for comments and suggestions.

References

[1] Ammari H and Kang H 2007 Polarization and Moment Tensors: With Applications to Inverse Problems and Effective Medium Theory (Berlin: Springer)
[2] Ascher U M, Ruuth S J and Wetton B T R 1995 Implicit–explicit methods for time-dependent partial differential equations SIAM J. Numer. Anal. 32 797–823
[3] Babuška I, Nobile F and Tempone R 2010 A stochastic collocation method for elliptic partial differential equations with random input data SIAM Rev. 52 317–55
[4] Babuška I, Tempone R and Zouraris G E 2004 Galerkin finite element approximations of stochastic elliptic partial differential equations SIAM J. Numer. Anal. 42 800–25
[5] Bäck J, Nobile F, Tamellini L and Tempone R 2011 Stochastic spectral Galerkin and collocation methods for PDEs with random coefficients: a numerical comparison Spectral and High Order Methods for Partial Differential Equations (Lecture Notes in Computational Science and Engineering vol 76) (Berlin: Springer) pp 43–62
[6] Bakirov V F and Kline R A 2005 Diffusion-based thermal tomography J. Heat Transfer 127 1276–9
[7] Bauer F, Hohage T and Munk A 2009 Iteratively regularized Gauss–Newton method for nonlinear inverse problems with random noise SIAM J. Numer. Anal. 47 1827–46
[8] Borcea L 2002 Electrical impedance tomography Inverse Problems 18 R99–136
[9] Canuto C, Hussaini M Y, Quarteroni A and Zang T A Jr 2006 Spectral Methods: Fundamentals in Single Domain (Berlin: Springer)
[10] Chapko R, Kress R and Yoon J-R 1998 On the numerical solution of an inverse boundary value problem for the heat equation Inverse Problems 14 853–67
[11] Chapko R, Kress R and Yoon J-R 1999 An inverse boundary value problem for the heat equation: the Neumann condition Inverse Problems 15 1033–46
[12] Chkifa A, Cohen A, DeVore R and Schwab C 2013 Sparse adaptive Taylor approximation algorithms for parametric and stochastic elliptic PDEs ESAIM: Math. Modelling Numer. Anal. 47 253–80
[13] Dahmen W, DeVore R and Scherer K 1980 Multi-dimensional spline approximation SIAM J. Numer. Anal. 17 380–402
[14] Dautray R and Lions J-L. 1992 Mathematical Analysis and Numerical Methods for Science and Technology, Vol 5: Evolution Problems I (Berlin: Springer)
[15] Ernst O G and Ullmann E 2010 Stochastic Galerkin matrices SIAM J. Matrix Anal. Appl. 31 1848–72
[16] Frauenfelder P, Schwab C and Todor R A 2005 Finite elements for elliptic problems with stochastic coefficients Comput. Methods Appl. Mech. Eng. 194 205–28
[17] Gaitan P, Isozaki H, Poisson O, Siltanen S and Tamminen J P 2015 Inverse problems for time-dependent singular heat conductivities: multi-dimensional case Commun. PDE 40 837–77
[18] Gautschi W 2004 Orthogonal Polynomials: Computation and Approximation (Oxford: Oxford University Press)
[19] Ghanem R G and Spanos P D 1991 Stochastic Finite Elements: A Spectral Approach (Berlin: Springer)
[20] Golub G H and Van Loan C F 2013 Matrix Computations (Baltimore, MD: John Hopkins University Press)
[21] Gratton S, Lawless A S and Nichols N K 2007 Approximate Gauss–Newton methods for nonlinear least squares problems SIAM J. Optim. 18 106–32
[22] Hakula H, Hyvönen N and Leinonen M 2014 Reconstruction algorithm based on stochastic Galerkin finite element method for electrical impedance tomography Inverse Problems 30 065006
[23] Hanke M 1997 Regularizing properties of a truncated Newton-CG algorithm for nonlinear inverse problems Numer. Funct. Anal. Optim. 18 971–93
[24] Harbrecht H and Tausch J 2013 On the numerical solution of a shape optimization problem for the heat equation SIAM J. Sci. Comput. 35 A104–21
[25] Hoang V H and Schwab C 2013 Sparse tensor Galerkin discretization of parametric and random parabolic PDEs—analytic regularity and generalized polynomial chaos approximation SIAM J. Math. Anal. 45 3050–83
[26] Höllig K 2003 Finite Element Methods with B-Splines (Philadelphia: SIAM)
[27] Ikehata M and Kawashita M 2010 On the reconstruction of inclusions in a heat conductive body from dynamical boundary data over a finite time interval Inverse Problems 26 095004
[28] Isakov V 2006 Inverse Problems for Partial Differential Equations (Berlin: Springer)
[29] Kolehmainen V, Kaipio J P and Orlande H R B 2008 Reconstruction of thermal conductivity and heat capacity using a tomographic approach Int. J. Heat Mass Transfer 51 1866–76
[30] Langer S 2009 Complexity analysis of the iteratively regularized Gauss–Newton method with inner CG-iteration J. Inverse Ill-Posed Problems 17 871–90
[31] Langer S and Hohage T 2007 Convergence analysis of an inexact iteratively regularized Gauss–Newton method under general source conditions J. Inverse Ill-Posed Problems 15 311–27
[32] Larsson S and Thomée V 2003 Partial Differential Equations with Numerical Methods (Berlin: Springer)
[33] Le Maître O P and Knio O M 2010 Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics (Berlin: Springer)
[34] Mustonen L 2014 Parametric differential equations and inverse diffusivity problem Master’s Thesis Aalto University
[35] Nobile F and Tempone R 2009 Analysis and implementation issues for the numerical approximation of parabolic equations with random coefficients Int. J. Numer. Methods Eng. 80 979–1006
[36] Nobile F, Tempone R and Webster C G 2008 A sparse grid stochastic collocation method for partial differential equations with random input data SIAM J. Numer. Anal. 46 2309–45
[37] Nocedal J and Wright S J 1999 Numerical Optimization (Berlin: Springer)
[38] Powell C E and Elman H C 2009 Block-diagonal preconditioning for spectral stochastic finite-element systemsIMA J. Numer. Anal. 29 350–75
[39] Saad Y, Yeung M, Erhel J and Guyomarck F 2000 A deflated version of the conjugate gradient algorithm SIAM J. Sci. Comput. 21 1909–26
[40] Schwab C and Gittelson C J 2011 Sparse tensor discretizations of high-dimensional parametric and stochastic PDEs Acta Numer. 20 291–467
[41] Soleimani M and Lionheart W R B 2005 Nonlinear image reconstruction for electrical capacitance tomography using experimental data Meas. Sci. Technol. 16 1987–96
[42] Toivanen J M, Kolehmainen V, Tarvainen T, Orlande H R B and Kaipio J P 2012 Simultaneous estimation of spatially distributed thermal conductivity, heat capacity and surface heat transfer coefficient in thermo tomography Int. J. Heat Mass Transfer 55 7958–68

[43] Toivanen J M, Tarvainen T, Huttunen J M J, Savolainen T, Orlande H R B, Kaipio J P and Kolehmainen V 2014 3D thermal tomography with experimental measurement data Int. J. Heat Mass Transfer 78 1126–34

[44] Ullmann E 2010 A Kronecker product preconditioner for stochastic Galerkin finite element discretizations SIAM J. Sci. Comput. 32 923–46

[45] Verwer J G 1996 Explicit Runge–Kutta methods for parabolic partial differential equations Appl. Numer. Math. 22 359–79

[46] Xiu D 2010 Numerical Methods for Stochastic Computations: A Spectral Method Approach (Princeton, NJ: Princeton University Press)

[47] Xiu D and Karniadakis G E 2003 A new stochastic approach to transient heat conduction modeling with uncertainty Int. J. Heat Mass Transfer 46 4681–93

[48] Xiu D and Shen J 2009 Efficient stochastic Galerkin methods for random diffusion equations J. Comput. Phys. 228 266–81

[49] Zhong X 1996 Additive semi-implicit Runge–Kutta methods for computing high-speed nonequilibrium reactive flows J. Comput. Phys. 128 19–31