

# A FRAMEWORK FOR THE ADAPTIVE FINITE ELEMENT SOLUTION OF LARGE INVERSE PROBLEMS

WOLFGANG BANGERTH\*

**Abstract.** Since problems involving the estimation of distributed coefficients in partial differential equations are numerically very challenging, efficient methods are indispensable. In this paper, we will introduce a framework for the efficient solution of such problems. This comprises the use of adaptive finite element schemes, solvers for the large linear systems arising from discretization, and methods to treat additional information in the form of inequality constraints on the parameter to be recovered. The methods to be developed will be based on an *all-at-once* approach, in which the inverse problem is solved through a Lagrangian formulation.

The main feature of the paper is the use of a continuous (function space) setting to formulate algorithms, in order to allow for discretizations that are adaptively refined as nonlinear iterations proceed. This entails that steps such as the description of a Newton step or a line search are first formulated on continuous functions and only then evaluated for discrete functions. On the other hand, this approach avoids the dependence of finite dimensional norms on the mesh size, making individual steps of the algorithm comparable even if they used differently refined meshes.

Numerical examples will demonstrate the applicability and efficiency of the method for problems with several million unknowns and more than 10,000 parameters.

**Key words.** Adaptive finite elements, inverse problems, Newton method on function spaces.

**AMS subject classifications.** 65N21,65K10,35R30,49M15,65N50

**1. Introduction.** Parameter estimation methods are important tools in cases where quantities we would like to know, such as material parameters, cannot be measured directly, but where only measurements of related quantities are available. In such cases one attempts to find a set of parameters for which the predictions of a mathematical model, the *equation of state*, best match what has actually been observed. Parameter estimation is therefore a problem that can be described as an optimization problem: minimize, by variation of the unknown parameter, the misfit between prediction and actual observation, subject to the constraint that the prediction satisfies the state equation.

If the state equation is a differential equation, such parameter estimation problems are commonly referred to as *inverse problems*. These problems have a vast number of applications. For example, identification of the underground structure (e.g. the elastic properties, density, electric or magnetic permeabilities of the earth) from measurements at the surface, or of the groundwater permeability of a soil from measurements of the hydraulic head fall in this class. Likewise, many biomedical imaging modalities, such as computer tomography, electrical impedance tomography, or several optical tomography modalities can be cast as inverse problems.

The case we are interested in here is recovering a *distributed*, i.e. spatially variable, coefficient. Oftentimes, such problems are found when trying to identify inhomogeneous material properties as in the examples mentioned above. In particular, we will consider cases where we make *many experiments* to identify the parameters. Here, by an experiment we mean subjecting the physical system to a certain forcing and measuring its response. For example, in computer tomography, a single experiment would be characterized by irradiating a body from a given angle and measuring the transmitted part of the radiation; the multiple experiment situation is characterized

---

\*Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA (bangerth@math.tamu.edu).

by using data from various incidence angles and trying to find a set of parameters that matches *all* the measurements *at the same time* (joint inversion). Likewise, in geophysics, a single experiment would be placing a seismic source somewhere and measuring reflection data at various receiver position; the multiple experiment case is taking into account data from more than one source position. We may also include entirely different kinds of data, e.g. use both magneto-telluric and gravimetry data for a joint, multi-physics inversion scenario.

This paper is devoted to the development of efficient techniques for the solution of such inverse problems where the state equation is a partial differential equation (PDE), and the parameters to be determined are one or several distributed functions. It is well-known that the numerical solution of PDE constrained inverse or optimization problems is significantly more challenging than that of a PDE alone (see, e.g., [13]), since the optimization procedure is usually iterative and in each iteration may need the numerical solution of a large number of partial differential equations. In some applications, several ten or hundred thousand solutions of linearized PDEs are required to solve the inverse problem, and each PDE may be discretized by up to several hundred thousand unknowns.

Although it is obvious that efficiency of solution is a major concern for this class of problems, efficient methods such as adaptive finite element techniques have not yet found widespread application to inverse problems and are only slowly adopted in the solution of PDE constrained optimization [9–11, 19, 20, 28, 28–30, 34, 41, 44, 45]. Rather, in most cases, the continuous inverse problem is first discretized on a predetermined mesh, and the resulting nonlinear problem is then solved using well-understood finite dimensional methods such as Newton’s method or a variant of it. On the other hand, discretizations can not be changed by adapting the mesh between nonlinear iterations, and the potential to significantly reduce the numerical cost by taking into account the spatial structure of solutions is lost. This is because a change in the discretization changes the size of finite dimensional problems, rendering finite-dimensional convergence criteria such as norms meaningless.

The goal of this paper is therefore to devise a framework for adaptive finite element techniques. By using such adapted meshes, we can not only significantly reduce the numerical effort needed to solve inverse parameter estimation problems, but the ability to choose a discretization mesh coarse where we lack information or where a fine mesh is not required also makes the inverse problem better posed. To achieve this goal, the main novel ingredients will be:

- *formulation of all algorithms in function spaces*, i.e. before rather than after discretization, since this gives us more flexibility in discretizing as iterations proceed, and resolves all scaling issues related to differing mesh sizes;
- the use of *adaptive finite element techniques* with mesh refinement based on a posteriori error estimates;
- the use of *different meshes for the discretization of different quantities*, for example of state variables and of parameters, in order to reflect their respective properties;
- the use of *Newton-type methods for the outer (nonlinear) iteration*, and of efficient linear solvers for the Newton steps;
- the use of approaches that allow for the *parallelization* of work, yielding sub-problems that are equivalent to only forward and adjoint problems;
- the *inclusion of pointwise bounds on the parameters* into the solution process.

Except for the derivation of error estimates, which we defer to future work (see also

[4]), we will discuss all these building blocks, and will show that these techniques allow us to solve problems of the size outlined above.

We envision that the techniques to be presented are used for relatively complex problems. Thus, we will state them in the setting of a generic inverse problem. In order to explain their concrete structure, we will define a model problem involving the Poisson equation and apply the framework to it. Numerical examples at the end of the paper will show this model problem as well as application of the framework to a significantly more complex case in optical tomography. Applications of the framework to other problems in acoustic scattering can be found in [4], and further work in optical tomography in biomedical imaging is also presented in [35,37].

Solving large-scale multiple-experiment inverse problems requires algorithms on several levels, all of which have to be tailored to high efficiency. In this article, we will review the building blocks of a framework for this:

- formulation as a Lagrangian optimization problem with PDE constraints (Section 2); a model problem is given in Section 3;
- outer nonlinear solution by a Gauss-Newton method posed in function spaces (Section 4);
- discretization of each Newton step by finite elements on independent meshes (Section 5);
- Schur complement solvers for the resulting linear systems (Section 6);
- methods to incorporate bound constraints on the parameters (Section 7).

In Section 8, we will present numerical examples. We will draw conclusions in the final section.

**2. General formulation and notation.** Let us begin by introducing some abstract notation, which we will use for the derivation of the entire scheme. This, above all, concerns the set of parameters, state equations, measurements, regularization, and the introduction of an abstract Lagrangian.

We note that some of the formulas below will become cumbersome to read because of the number of indices. To understand their meaning, it is often helpful to imagine we had only a single experiment (for example, only one incidence angle in tomography, or only one source position in seismic imaging). In this case, one may drop the index  $i$  on first reading, as well as all summations over  $i$ . In addition, the formulas of this section will be made concrete by introducing a model problem in Section 3.

**State equations.** Let the general setting of the problems we consider be as follows: assume that we subject a physical system to  $i = 1, \dots, N$  different external stimuli, and that we intend to learn about the system's material parameters by measuring how the system reacts. For the current purposes, we assume that the system's states can be described by (independent) partial differential equations posed on a domain  $\Omega \subset \mathbb{R}^d$ :

$$\mathcal{A}^i[q] u^i = f^i \quad \text{in } \Omega, \quad (2.1)$$

$$\mathcal{B}^i[q] u^i = h^i \quad \text{on } \Gamma_N^i \subset \partial\Omega, \quad (2.2)$$

$$u^i = g^i \quad \text{on } \Gamma_D^i = \partial\Omega \setminus \Gamma_N^i, \quad (2.3)$$

where  $\mathcal{A}^i[q]$  are partial differential operators all of which depend on a common set of a priori unknown distributed (i.e. spatially variable) coefficients  $q = q(\mathbf{x}) \in \mathcal{Q}$ ,  $\mathbf{x} \in \Omega$ , and  $\mathcal{B}^i[q]$  are boundary operators that may also depend on the coefficients.  $f^i, h^i$  and  $g^i$  are the known external forcing terms, and that are independent of the coefficients

$q$ . The functions  $u^i$  are the solutions of the partial differential equations, i.e. the physical outcomes (“states”) of our  $N$  experiments. These scalar or vector-valued solutions are assumed to be from spaces  $V_g^i = \{\varphi \in V^i : \varphi|_{\Gamma_D^i} = g^i\}$ . We assume that solutions  $u^i, u^j$  of different equations are independent except for their dependence on the common set of parameters  $q$ . We also assume that the solutions to each of the differential equations is unique for every given set of parameters  $q$  in a subset  $\mathcal{Q}_{ad} \subset \mathcal{Q}$  of physically meaningful values, for example  $\mathcal{Q}_{ad} = \{q \in L^\infty(\Omega) : q_0 \leq q(\mathbf{x}) \leq q_1\}$ .

Typical cases we have in mind would be a Laplace-type equation when we are considering electrical impedance tomography or gravimetry inversion, Helmholtz or wave equations for inverting seismic or magneto-telluric data, or diffusion-reaction equations for optical tomography applications. The set of parameters  $q$  would, in these cases, be electrical conductivities, densities, elasticity coefficients, or optical properties. The operators  $\mathcal{A}^i$  may be the same if we repeat the same kind of experiment multiple times with different forcings, but they will be different if we use different physical effects (for example gravimetry and seismic data) to identify a set of parameters.

The formulation above may easily be extended also to the case of time-dependent problems. Likewise, the case that the parameters are a finite number of scalar values instead of distributed functions is a simple special case, but we omit these cases for brevity.

For treatment in a Lagrangian setting in function spaces as well as for discretization by finite elements, it is necessary to formulate the state equations (2.1)–(2.3) in a variational form. For this we assume that the solutions  $u^i \in V_g^i$  are solutions of the following variational equalities:

$$A^i(q; u^i)(\varphi^i) = 0 \quad \forall \varphi^i \in V_0^i, \quad (2.4)$$

where  $V_0^i = \{\varphi^i \in V^i : \varphi^i|_{\Gamma_D^i} = 0\}$ . The semilinear form  $A^i : \mathcal{Q} \times V_g^i \times V_0^i \rightarrow \mathbb{R}$  may be nonlinear in its first set of arguments, but is linear in the test function, and includes the actions of domain and boundary operators  $\mathcal{A}^i$  and  $\mathcal{B}^i$ , as well as of inhomogeneous forcing terms. We will later have to assume that the  $A^i$  are differentiable.

As an example, we will introduce a model problem below for the Poisson equation. In that case,  $\mathcal{A}^i[q]u^i = -\nabla \cdot (q\nabla u^i)$ ,  $\mathcal{B}^i[q]u^i = q\partial_n u^i$ , and

$$A(q; u^i)(\varphi^i) = (q\nabla u^i, \nabla \varphi^i)_\Omega - (f, \varphi^i)_\Omega - (h, \varphi^i)_{\Gamma_N}.$$

**Measurements.** In order to determine the unknown quantities  $q$ , we measure how the physical system reacts to the external forcing, i.e. we measure (parts of) the states  $u^i$  or derived quantities. For example, we might have measurements of voltages, optical fluxes or stresses at certain points, averages on subdomains, or gradients. Let us denote the space of measurements of the  $i$ th state variable by  $\mathcal{Z}^i$ , and let  $M^i : V_g^i \rightarrow \mathcal{Z}^i$  be the measurement operator, i.e. the operator that extracts from physical state  $u^i$  that information that we measure.

If we knew the parameters  $q$ , we could use the state equation (2.4) to predict the state the system would be in, and  $M^i u^i$  would then be the predicted measurements. On the other hand, we don’t know  $q$ , but we have actual measurements that we denote by  $z^i \in \mathcal{Z}^i$ . Reconstruction of the coefficients  $q$  will be accomplished by finding that coefficient, for which the predicted measurements  $M^i u^i$  match the actual measurements  $z^i$  best. We will measure this comparison using a convex, differentiable functional  $m : \mathcal{Z}^i \rightarrow \mathbb{R}$ . In many cases,  $m$  will simply be an  $L^2$  norm on  $\mathcal{Z}^i$ , but more

general functionals are allowed, for example to suppress the effects of non-Gaussian noise [48].

Examples of common measurement situations are:

- *L<sup>2</sup> measurements of values*: If measurements on a set  $\Sigma \subset \Omega$  are available, then  $M^i$  is the embedding operator from  $V_g^i$  into  $\mathcal{Z}^i = L^2(\Sigma)$ , and we will try to find  $q$  by minimizing

$$m^i(M^i u^i - z^i) = \frac{1}{2} \|u^i - z^i\|_{L^2(\Sigma)}^2.$$

In many non-destructive testing or tomography applications, one has  $\Sigma \subset \partial\Omega$  because measurements in the interior are not possible. The case of distributed measurements occurs in situations where a measuring device can be moved around to every point of  $\Sigma$ , for example a laser scanning a membrane, or a camera is imaging a body.

- *Point measurements*: If we have  $S$  measurements of  $u(\mathbf{x})$  at positions  $x_s \in \Omega, s = 1, \dots, S$ , then  $\mathcal{Z}^i = \mathbb{R}^S$ , and  $(M^i u^i)_s = u(\mathbf{x}_s)$ . If we take again a quadratic norm on  $\mathcal{Z}^i$ , then for example

$$m^i(M^i u^i - z^i) = \frac{1}{2} \sum_{s=1}^S |u^i(\mathbf{x}_s) - z_s^i|^2$$

is a possible choice. The case of point measurements is frequent in applications where a small number of stationary measurement devices is used, for example seismometers in seismic data assimilation.

Other choices are of course possible, and are usually dictated by the type of available measurements.

We will in general assume that the  $M^i$  are linear, but there are applications where this is not the case. For example, in some applications only statistical correlations of  $u^i$  are known, or a power spectrum. Extending the algorithms below to nonlinear  $M^i$  is straightforward, but we omit this for brevity.

**Regularization.** Since inverse problems are often ill-posed, regularization is needed to suppress unwanted features in solutions  $q$ . In this work, we include it by using a Tikhonov regularization term involving a convex differentiable regularization functional  $r : \mathcal{Q} \rightarrow \mathbb{R}^+$ , see for example [25, 39]. Most frequently  $r(q) = \frac{1}{2} \|\nabla^t(q - \bar{q})\|_{L^2(\Omega)}^2$  with an a priori guess  $\bar{q}$  and some  $t \geq 0$ . Other popular choices are smoothed versions of bounded variation seminorms [22, 24, 27]. As above, the type of regularization is usually dictated by the application and insight into physical and unphysical features of solutions.

**Characterization of solutions.** The goal of the inverse problem is to find that set of physical parameters  $q \in \mathcal{Q}_{ad}$  for which the predictions  $M^i u^i$  match the actual observations  $z^i$  best. We formulate this as the following constrained minimization problem over  $u^i \in V_g^i, q \in \mathcal{Q}_{ad}$ :

$$\begin{aligned} \text{minimize} \quad & J(\{u^i\}, q) = \sum_{i=1}^N \sigma^i m^i(M^i u^i - z^i) + \beta r(q) & (2.5) \\ \text{such that} \quad & A^i(q; u^i)(\varphi^i) = 0 \quad \forall \varphi^i \in V_0^i, \quad 1 \leq i \leq N. \end{aligned}$$

Here,  $\sigma^i > 0$  are factors weighting the relative importance of individual measurements, and  $\beta > 0$  is a regularization parameter. As the choice of these constants is a topic of its own, we assume their values as given within the scope of this work.

In order to characterize solutions to (2.5), let us subsume the individual solutions  $u^i$  to a vector  $\mathbf{u}$ , and likewise  $\mathbf{V}_g = \{V_g^i\}$ ,  $\mathbf{V}_0 = \{V_0^i\}$ . Furthermore, we introduce a set of Lagrange multipliers  $\boldsymbol{\lambda} \in \mathbf{V}_0$ , and denote the joint set of variables by  $x = \{\mathbf{u}, \boldsymbol{\lambda}, q\} \in \mathcal{X}_g = \mathbf{V}_g \times \mathbf{V}_0 \times \mathcal{Q}$ .

Under appropriate conditions (see, e.g., [8]), solutions of problem (2.5) are stationary points of the following Lagrangian  $L : \mathcal{X}_g \rightarrow \mathbb{R}$ , which couples the functional  $J : \mathbf{V}_g \times \mathcal{Q} \rightarrow \mathbb{R}^+$  defined above to the state equation constraints through Lagrange multipliers  $\lambda^i \in V_0^i$ :

$$L(x) = J(\mathbf{u}, q) + \sum_{i=1}^N A^i(q; u^i)(\lambda^i). \quad (2.6)$$

The optimality conditions then read in abstract form

$$L_x(x)(y) = 0 \quad \forall y \in \mathcal{X}_0, \quad (2.7)$$

where the semilinear form  $L_x : \mathcal{X}_g \times \mathcal{X}_0 \rightarrow \mathbb{R}$  is the derivative of the Lagrangian  $L$ , and  $\mathcal{X}_0 = \mathbf{V}_0 \times \mathbf{V}_0 \times \mathcal{Q}$ . Indicating derivatives of functionals with respect to their arguments by subscripts, we can expand (2.7) to yield the following set of nonlinear equations:

$$L_{\lambda^i}(x; \varphi^i) \equiv A^i(q; u^i)(\varphi^i) = 0 \quad \forall \varphi^i \in V_0^i, \quad (2.8)$$

$$L_{u^i}(x; \psi^i) \equiv \sigma^i m_u^i(M^i u^i - z^i)(\psi^i) + A_u^i(q; u^i)(\psi^i, \lambda^i) = 0 \quad \forall \psi^i \in V_0^i, \quad (2.9)$$

$$L_q(x; \chi) \equiv \beta r_q(q)(\chi) + \sum_{i=1}^N A_q^i(q; u^i)(\chi, \lambda^i) = 0 \quad \forall \chi \in \partial \mathcal{Q}. \quad (2.10)$$

The first set of equations denotes the state equations for  $i = 1, \dots, N$ ; the second the adjoint equations defining the Lagrange multipliers  $\lambda^i$ ; finally, the third is the control equation holding for all functions from the tangent space  $\partial \mathcal{Q}$  to  $\mathcal{Q}$  at the solution  $q$ .

**3. A model problem.** As a simple model problem which we will use to give the abstract results of this work a more concrete form, we will consider the following situation: assume we intend to identify the coefficient  $q$  in the (single) elliptic PDE

$$-\nabla \cdot (q \nabla u) = f \quad \text{in } \Omega, \quad u = g \quad \text{on } \partial \Omega, \quad (3.1)$$

and that measurements are the values of the solution  $u$  everywhere in  $\Omega$ , i.e. we choose  $m(Mu - z) = \frac{1}{2} \|u - z\|_{L^2(\Omega)}^2$ . This situation can be considered as a mathematical description of a membrane with variable stiffness  $q(\mathbf{x})$ . We try to identify this coefficient by subjecting the membrane to a known force  $f$  and clamping it at the boundary with boundary values  $g$ . This results in displacements of which we obtain measurements  $z$  everywhere.

For this situation,  $V_g = \{u \in H^1 : u|_{\partial \Omega} = g\}$ ,  $\mathcal{Q} \subset L^\infty$ . Choosing  $\sigma = 1$ , the Lagrange functional has the form

$$L(x) = \frac{1}{2} \|u - z\|_{L^2(\Omega)}^2 + \beta r(q) + (q \nabla u, \nabla \lambda) - (f, \lambda).$$

With this, the optimality conditions (2.8)–(2.10) read in weak form

$$(q\nabla u, \nabla\varphi) = (f, \varphi), \quad (3.2)$$

$$(q\nabla\psi, \nabla\lambda) = -(u - z, \psi), \quad (3.3)$$

$$\beta r_q(q; \chi) = -(\chi\nabla u, \nabla\lambda), \quad (3.4)$$

and have to hold for all test functions  $\{\varphi, \psi, \chi\} \in H_0^1 \times H_0^1 \times \mathcal{Q}$ . Note that the first of these is the state equation, while the second is the adjoint equation.

**4. Nonlinear solvers.** The stationarity conditions (2.7) form a set of nonlinear partial differential equations that has to be solved iteratively, for example using Newton’s method, or a variant thereof. In this section, we will formulate the Gauss-Newton method in function spaces. The discretization of each step by adaptive finite elements will then be presented in the next section, followed by a discussion of solvers for the resulting linear systems.

Since there is no need to compute the initial nonlinear steps on a very fine grid when we are still far away from the solution, we will want to use successively finer meshes as we approach the solution. In order to make quantities computed on different meshes comparable, all of the following algorithms will be formulated in a continuous setting, and only then be discretized. This also answers once and for all questions about the correct scaling of weighting matrices in misfit and regularization functionals, as discussed for example in [2], even if we choose locally refined grids, as they will appear naturally upon discretization.

In this section, we indicate a Gauss-Newton procedure, i.e. determination of search direction and step length, in infinite dimensional spaces, and discuss in the next section its discretization by a finite element scheme. At least for finite-dimensional problems, there is a vast number of alternatives to the Gauss-Newton method, see for example [1, 21, 32, 33, 42, 43, 47]. However, we believe that the Gauss-Newton method is particularly suited since it allows for scalable algorithms even with large numbers of experiments, and large numbers of degrees of freedom both in the discretization of the state equations as well as of the parameter. Comparing this method to a pure Newton method, it allows for the use of more efficient linear solvers for the discretized problems, see Section 6. In addition, the Gauss-Newton method has been shown to have better stability properties for parameter estimation problems than the Newton method, see [16, 17].

**Search directions.** Given a current approximation  $x_k = \{\mathbf{u}_k, \boldsymbol{\lambda}_k, q_k\} \in \mathcal{X}$  after  $k$  iterations, the first task of any iterative nonlinear solver method is to compute a search direction  $\delta x_k = \{\delta\mathbf{u}_k, \delta\boldsymbol{\lambda}_k, \delta q_k\} \in \mathcal{X}_{\delta g}$ , in which we seek the next iterate  $x_{k+1}$ . The Dirichlet boundary values  $\delta g$  of this update are chosen as  $\delta u_k^i|_{\Gamma_D} = g^i - u_k^i|_{\Gamma_D}$ ,  $\delta\lambda_k^i|_{\Gamma_D} = 0$ , bringing us to the exact boundary values if we take a full step.

The Gauss-Newton method determines search directions  $\{\delta\mathbf{u}_k, \delta q_k\}$  by minimizing the following quadratic approximation to  $J(\cdot, \cdot)$  with linearized constraints:

$$\begin{aligned} \min_{\delta\mathbf{u}_k, \delta q_k} & J(\mathbf{u}_k, q_k) + J_u(\mathbf{u}_k, q_k)(\delta\mathbf{u}_k) + J_q(\mathbf{u}_k, q_k)(\delta q_k) \\ & + \frac{1}{2}J_{uu}(\mathbf{u}_k, q_k)(\delta\mathbf{u}_k, \delta\mathbf{u}_k) + \frac{1}{2}J_{qq}(\mathbf{u}_k, q_k)(\delta q_k, \delta q_k) \end{aligned} \quad (4.1)$$

$$\text{such that } A^i(q_k; u_k^i)(\varphi^i) + A_u^i(q_k; u_k^i)(\delta u_k^i, \varphi^i) + A_q^i(q_k; u_k^i)(\delta q_k, \varphi^i) = 0,$$

where the linearized constraints are understood to hold for  $1 \leq i \leq N$  and for all test functions  $\varphi^i \in V_0^i$ . The solution of this problem provides us with updates  $\delta\mathbf{u}_k, \delta q_k$

for the state variables and the parameters. The updates for the Lagrange multiplier  $\delta\lambda_k$  are not determined by the Gauss-Newton step at first. However, we can get updates  $\delta\lambda_k$  for the original problem, by using  $\lambda_k + \delta\lambda_k$  as Lagrange multiplier for the constraint of the Gauss-Newton step (4.1). Bringing the terms with  $\lambda_k$  to the right hand side, the updates are then characterized by the following system of linear equations:

$$\begin{aligned} \sigma^i m_{uu}^i (M^i u_k^i - z^i)(\delta u_k^i, \varphi^i) + A_u^i(q_k; u_k^i)(\varphi^i, \delta\lambda_k^i) &= -L_{u^i}(x_k)(\varphi^i), \\ A_u^i(q_k; u_k^i)(\delta u_k^i, \psi^i) + A_q^i(q_k; u_k^i)(\delta q_k^i, \psi^i) &= -L_{\lambda^i}(x_k)(\psi^i), \\ \sum_i A_q^i(q_k; u_k^i)(\chi, \delta\lambda_k^i) + \beta r_{qq}(q_k)(\delta q_k, \chi) &= -L_q(x_k)(\chi), \end{aligned} \quad (4.2)$$

for all test functions  $\{\varphi^i, \psi^i, \chi\}$ . The right hand side of these equations is the negative gradient of the original Lagrangian, given already in the optimality condition (2.8)–(2.10).

Note that the equations determining the updates for the  $i$ th experiment decouple from all other experiments, except for the last equation. This will allow us to solve them mostly separated, and in particular it allows for simple parallelization by placing the description of different experiments onto different machines. Furthermore, the first and second equations can be solved sequentially.

In order to illustrate these equations, we state their form for the model problem of Section 3. In this case, the above system reads

$$\begin{aligned} (\delta u_k, \varphi) &+ (\nabla\delta\lambda_k, q_k\nabla\varphi) &&= -L_u(x_k)(\varphi), \\ (\nabla\psi, q_k\nabla\delta u_k) &+ (\nabla\psi, \delta q_k\nabla u_k) &&= -L_\lambda(x_k)(\psi), \\ &(\nabla\delta\lambda_k, \chi\nabla u_k) + \beta r_{qq}(q_k)(\delta q_k, \chi) &&= -L_q(x_k)(\chi), \end{aligned}$$

with the right hand sides being the gradient of the Lagrangian given in Section 3.

In general, this continuous Gauss-Newton direction will not be computable analytically. We will therefore approximate it by a finite element function  $\delta x_{k,h}$ , as discussed in the next section.

As a final remark, let us note that the pure Newton's method would read

$$L_{xx}(x_k)(\delta x_k, y) = -L_x(x_k)(y) \quad \forall y \in \mathcal{X}_0, \quad (4.3)$$

where  $L_{xx}(x_k)(\cdot, \cdot)$  denotes the bilinear form of second variational derivatives of the Lagrangian  $L$  at position  $x_k$ . The Gauss-Newton method can alternatively be obtained from this by simply dropping all terms that are proportional to the Lagrange multiplier  $\lambda_k$ . This is based on considering (2.9) or (3.3):  $\lambda^i$  is proportional to  $M^i u^i - z^i$  and thus will be small if  $M^i u^i - z^i$  is small, assuming stability of the (linear) adjoint operator.

**Step lengths.** Once we have a search direction  $\delta x_k$ , we have to decide how far to go in this direction starting at  $x_k$  to obtain the next iterate  $x_{k+1} = x_k + \alpha_k \delta x_k$ . In constrained optimization, a merit function including the minimization functional  $J(\cdot)$  as well as the violation of the constraints is usually used for this [46].

One particular problem here is the infinite dimensional nature of the state equation constraint, with the residual of the state equation being in the dual space,  $V_0'$ , of  $V_0$  (which, for the model problem, is  $H^{-1}$ ). Consequently, it is unclear which norm to use and whether we need to weight a violation of the state equation in different parts



of the domain differently or not. Furthermore, the relative weighting of constraint and objective function is not obvious.

In order to avoid these problems, we propose to use the norm of the residual of the optimality condition (2.7) on the dual space of  $\mathcal{X}_0$  as merit function:

$$p(\alpha) = \frac{1}{2} \|L_x(x_k + \alpha \delta x_k)(\cdot)\|_{\mathcal{X}'_0}^2 \equiv \frac{1}{2} \sup_{y \in \mathcal{X}_0} \frac{[L_x(x_k + \alpha \delta x_k)(y)]^2}{\|y\|_{\mathcal{X}_0}^2}.$$

We will show in the next section that we can give a simple-to-compute lower bound for this norm using the discretization we already employ for the computation of the search direction.

The following lemma shows that this merit function is actually useful:

**LEMMA 4.1.** *The merit function  $p(\alpha)$  is valid, i.e. Newton directions are directions of descent,  $p'(0) < 0$ . Furthermore, if  $x_k = x$  is a solution of the parameter estimation problem, then  $p(0) = 0$ . Finally, in the vicinity of the solution, full steps are taken, i.e.  $\alpha = 1$  minimizes  $p$  as  $x_k \rightarrow x$ .*

*Proof.* We prove the lemma for the special case of only one experiment ( $N = 1$ ) and that  $\mathcal{X} = H_0^1 \times H_0^1 \times L^2$ , i.e. the situation of the model example. However, it is obvious how to extend the proof to the general case. In this simplified situation, by the Riesz theorem there is a representation  $g_u(x_k + \alpha \delta x_k) = L_u(x_k + \alpha \delta x_k)(\cdot) \in H^{-1}$ ,  $g_\lambda(x_k + \alpha \delta x_k) = L_\lambda(x_k + \alpha \delta x_k)(\cdot) \in H^{-1}$ , and  $g_q(x_k + \alpha \delta x_k) = L_q(x_k + \alpha \delta x_k)(\cdot) \in L^2$ . The dual norm of  $L_x$  can then be written as

$$\|L_x\|_{\mathcal{X}'_0}^2 = \langle g_u, (-\Delta)^{-1} g_u \rangle + \langle g_\lambda, (-\Delta)^{-1} g_\lambda \rangle + (g_q, g_q),$$

where  $(-\Delta)^{-1} : H^{-1} \rightarrow H_0^1$ , and  $\langle \cdot, \cdot \rangle$  indicates the duality pairing between  $H^{-1}$  and  $H_0^1$ . Then,

$$p'(0) = \langle g_{uu}(\delta u_k), (-\Delta)^{-1} g_u \rangle + \langle g_{\lambda\lambda}(\delta \lambda_k), (-\Delta)^{-1} g_\lambda \rangle + (g_{qq}(\delta q_k), g_q),$$

where  $g_{ux}(\delta x_k)$  is the derivative of  $g_u$  in direction  $\delta x_k$ , i.e. the functional of second derivatives of  $L$ . However, by definition of the Newton direction, (4.3), this is equal to the negative gradient, i.e.

$$p'(0) = -\|L_x(x_k)\|_{\mathcal{X}'_0}^2 = -2p(0) < 0.$$

Thus, Newton directions are directions of descent for this merit function.

The second part of the lemma is obvious by noting the optimality condition (2.7). The last part follows by noting that near the solution, the Lagrangian (and thus the function  $p(\alpha)$ ) is well approximated by a quadratic function if the various functionals involved in the Lagrangian are sufficiently smooth. As  $x_k \rightarrow x$ , Newton directions satisfy  $\delta x_k \rightarrow x - x_k$  and  $\min_\alpha p(\alpha) \rightarrow 0$ . On the other hand, it is easy to show that quadratic functions with  $p'(0) = -2p(0)$  and  $\min_\alpha p(\alpha) = 0$  have their minimum at  $\alpha = 1$ .  $\square$

**5. Discretization.** The goal of the preceding section was to provide the function space tools to find a solution  $x$  of the inverse problem. In order to actually compute finite-dimensional approximations to  $x$ , we have to discretize both the state and adjoint variables, as well as the parameters. In this section, we will introduce finite element schemes to do so. The main point is to be able to change meshes between Gauss-Newton iterations. This has at least three advantages over an a priori choice of a mesh: (i) it makes the initial iterations significantly cheaper when we are still

far away from the solution; (ii) coarser meshes act as an additional regularization, making the problem better posed; and (iii) it allows us to adapt the resolution of the mesh to the characteristics of the solution.

In each iteration, we define finite element spaces  $\mathcal{X}_h \subset \mathcal{X}$  over triangulations in the usual way. In particular, let  $\mathbb{T}_k^i$  bet the mesh on which to discretize state and adjoint variables  $u_k^i, \lambda_k^i$  of the  $i$ th experiment in the  $k$ th Gauss-Newton iteration. Independently, a mesh  $\mathbb{T}_k^q$  will be used to discretize the parameters  $q$  on step  $k$ . This reflects that the regions of missing regularity of parameters and state variables need not necessarily coincide. We may also use different discretization spaces for parameters and state/adjoint variables, for example spaces of discontinuous functions for quantities like density or elasticity coefficients. For these grids and and the finite element spaces defined on them, we assume the following requirements:

- *Nesting*: The mesh  $\mathbb{T}_k^i$  must be obtainable from  $\mathbb{T}_{k-1}^i$  by hierarchic coarsening and refinement. This greatly simplifies operations like evaluation of the right hand side of the Newton direction equation,  $L_x(x_k)(y_{k+1})$  for all discrete test functions  $y_{k+1}$ , but also the update operation  $x_{k+1} = x_k + \alpha_k \delta x_{k,h}$ .
- *State vs. parameter meshes*: Each of the ‘state meshes’  $\mathbb{T}_k^i$  can be obtained by hierarchical refinement from the ‘parameter mesh’  $\mathbb{T}_k^q$ .

Although obvious, the choice of entirely independent grids for state and parameter meshes has apparently not been used in the literature to the author’s best knowledge. On the other hand, this technique offers the prospect of greatly reducing the amount of numerical work. We will see that with the requirements on the meshes above, the additional work associated with using different meshes is in fact small.

Choosing different ‘state’ and ‘parameter meshes’ is also beneficial for problems where the parameters do not require high resolution, or only in certain areas of the domain, while the state equation does. A typical problem is high-frequency potential scattering, where the coefficient might be a function that is constant in large parts of the domain, while the high-frequency oscillations of state and adjoint variables require a fine grid everywhere.

In the next few paragraphs, we will briefly describe the process of discretizing the equations for the search directions and the choice of the step length. We will then give a brief note on the criteria for generating the meshes on which we discretize.

**Search directions.** By choosing a finite dimensional subspace  $\mathcal{X}_h = V_h \times V_h \times \mathcal{Q}_h \subset \mathcal{X}$ , we obtain a discrete counterpart for equation (4.2) describing the Gauss-Newton search direction:

$$\begin{aligned} \sigma^i m_{uu}^i (M^i u_k^i - z^i)(\delta u_{k,h}^i, \varphi_h^i) + A_u^i(q_k; u_k^i)(\varphi_h^i, \delta \lambda_{k,h}^i) &= -L_{u^i}(x_{k,h})(\varphi_h^i), \\ A_u^i(q_k; u_k^i)(\delta u_{k,h}^i, \psi_h^i) + A_q^i(q_k; u_k^i)(\delta q_{k,h}^i, \psi_h^i) &= -L_{\lambda^i}(x_{k,h})(\psi_h^i), \\ \sum_i A_q^i(q_k; u_k^i)(\chi_h, \delta \lambda_{k,h}^i) + \beta r_{qq}(q_k)(\delta q_{k,h}, \chi_h) &= -L_q(x_{k,h})(\chi_h). \end{aligned} \quad (5.1)$$

Choosing a basis for the space  $\mathcal{X}_h$ , we can write (5.1) in matrix form as follows:

$$\begin{pmatrix} \mathbf{M} & \mathbf{A}^T & \mathbf{0} \\ \mathbf{A} & \mathbf{0} & \mathbf{C} \\ \mathbf{0} & \mathbf{C}^T & \beta \mathbf{R} \end{pmatrix} \begin{pmatrix} \delta \mathbf{u}_{k,h} \\ \delta \boldsymbol{\lambda}_{k,h} \\ \delta q_{k,h} \end{pmatrix} = \begin{pmatrix} \mathbf{F}_u \\ \mathbf{F}_\lambda \\ \mathbf{F}_q \end{pmatrix}. \quad (5.2)$$

Since the individual state equations and variables do not couple across experiments,  $\mathbf{M} = \text{diag}(\mathbf{M}^i)$  and  $\mathbf{A} = \text{diag}(\mathbf{A}^i)$  are block diagonal matrices, with the diagonal

blocks stemming from second derivatives of the misfit functionals, and of the tangential operators of the state equations, respectively. They are equal to

$$(\mathbf{M}^i)_{kl} = m_{uu}^i (M^i u_k^i - z^i)(\varphi_k^i, \varphi_l^i), \quad (\mathbf{A}^i)_{kl} = A_u^i(x_k)(\varphi_l^i, \varphi_k^i),$$

where  $\varphi_l^i$  are test functions for the discretization of the  $i$ th state equation. Likewise,  $\mathbf{C} = [\mathbf{C}^1, \dots, \mathbf{C}^N]$  is defined by  $(\mathbf{C}^i)_{kl} = A_q^i(x_k)(\chi_l^q, \varphi_k^i)$ , with  $\chi_l^q$  being discrete test functions for the parameters  $q$ , and  $(\mathbf{R})_{kl} = r_{qq}(q_k)(\chi_k^q, \chi_l^q)$ .

The evaluation of  $\mathbf{C}^i$  may be difficult since it involves shape functions from different meshes and finite element spaces. However, since we have required that  $\mathbb{T}_k^i$  can be obtained from  $\mathbb{T}_k^q$  by hierarchical refinement, we can represent each shape function  $\chi_k^q$  on the parameter mesh as a sum over respective shape functions  $\chi_s^i$  on each of the state meshes:  $\chi_k^q = \sum_s \mathbf{X}_{ks}^i \chi_s^i$ . Thus,  $\mathbf{C}^i = \hat{\mathbf{C}}^i \mathbf{X}^i$ , with  $\hat{\mathbf{C}}^i$  built with shape functions from only one grid. The matrix  $\mathbf{X}^i$  is fairly simple to generate in practice because of the hierarchical structure of the meshes.

Solving equation (5.2) will give us an approximate search direction. The solution of this linear system will be discussed in Section 6.

**Step lengths.** Since step length selection is only a tool for seeking the exact solution, we may be content with approximating the merit function  $p(\alpha)$  introduced in Section 4. To this end, we use a lower bound  $\underline{p}(\alpha)$  for  $p(\alpha)$ , by restricting the set of possible test functions to the discrete space  $\mathcal{X}_h$  which we are already using for the discretization of the search direction:

$$\underline{p}(\alpha) = \frac{1}{2} \sup_{y \in \mathcal{X}_h} \frac{[L_x(x_k + \alpha \delta x_k)(y)]^2}{\|y\|_{\mathcal{X}_0}^2} \leq \frac{1}{2} \|L_x(x_k + \alpha \delta x_k)(\cdot)\|_{\mathcal{X}_0'}^2 = p(\alpha).$$

By selecting a basis of  $\mathcal{X}_h$ ,  $\underline{p}(\alpha)$  can be computed by linear algebra. For example, for the single experiment case ( $N = 1$ ) and if  $\mathcal{X} = H_0^1 \times H_0^1 \times L^2$ , we have that

$$\underline{p}(\alpha) = \frac{1}{2} [\langle g_u(\alpha), Y_1^{-1} g_u(\alpha) \rangle + \langle g_\lambda(\alpha), Y_1^{-1} g_\lambda(\alpha) \rangle + \langle g_q(\alpha), Y_0^{-1} g_q(\alpha) \rangle],$$

where  $(Y_0)_{kl} = (\chi_k, \chi_l)$ ,  $(Y_1)_{kl} = (\nabla \varphi_k, \nabla \varphi_l)$  are mass and Laplace matrices, respectively. The gradient vectors are  $(g_u)_l = L_u(x_k + \alpha \delta x_k)(\varphi_l)$ , and correspondingly for  $g_\lambda$  and  $g_q$ . Here,  $\varphi_l$  are again basis functions from the discrete approximation space to the state and adjoint variable, and  $\chi_l$  for the parameters.

The evaluation of  $\underline{p}(\alpha)$  therefore requires only two inversions of  $Y_1$  per experiment, and of one mass matrix for the parameters. Setting up the gradient vectors reuses operations that are also available from the generation of the linear system in each Gauss-Newton step. With this merit function, the computation of a step length is then done using the usual methods (see, e.g., [46]). Having to invert matrices for step length selection would seem like an expensive procedure. However, compared to the effort required for the solution (5.2), the effort is usually rather negligible. On the other hand, we note that  $\underline{p}(\cdot)$  correctly scales components of the residual according to the size of cells on our adaptive meshes, unlike the usual  $l_p$  norms of the residual vectors  $g_u, g_\lambda, g_q$ , and is therefore a robust indicator for progress of the nonlinear iteration.

**Mesh refinement.** The meshes we choose for discretization should share a minimum of characteristics as described above, but apart from that their generation is arbitrary. For the numerical examples presented in this paper, both state and parameter meshes are kept constant for a number of nonlinear iterations until we are satisfied

with the progress of the nonlinear iterations. They are then refined using indicators based on the smoothness of solutions or rigorous error estimates. The construction of such refinement indicators is beyond the scope of this paper, but a discussion of can, for example, be found in [4, 7, 41].

**6. Linear solvers.** The linear system (5.2) is hardly solvable as is, except for the most simple problems: its size is twice the number of variables in each discrete forward problem, summed up over all experiments plus the number of discretized parameters; for many applications this size easily reaches into the tens of millions. Furthermore, it is indefinite and often extremely ill-conditioned (see [4]): for the model problem with  $m(\varphi) = \frac{1}{2}\|\varphi\|^2$ , the condition number of the matrix grows with the mesh size  $h$  as  $\mathcal{O}(h^{-6})$ .

Several schemes have been devised in the literature to solve (5.2) [3, 23, 31]. Particularly noteworthy is the comparison of different methods by Biros and Ghattas [14]. Because it leads to an algorithm that is relatively simple to parallelize and because it allows for the inclusion of bound constraints (see Section 7), we prefer to re-state the system by block elimination and use of the sub-structure of the individual blocks to obtain the following Schur complement formulation:

$$\mathbf{S} \delta q_{k,h} = \mathbf{F}_q - \sum_{i=1}^N \mathbf{C}^{iT} \mathbf{A}^{i-T} (\mathbf{F}_u^i - \mathbf{M}^i \mathbf{A}^{i-1} \mathbf{F}_\lambda^i), \quad (6.1)$$

$$\mathbf{A}^i \delta u_{k,h}^i = \mathbf{F}_\lambda^i - \mathbf{C}^i \delta q_{k,h}, \quad (6.2)$$

$$\mathbf{A}^{iT} \delta \lambda_{k,h}^i = \mathbf{F}_u^i - \mathbf{M}^i \delta q_{k,h}. \quad (6.3)$$

Here  $\mathbf{S}$  denotes the Schur complement

$$\mathbf{S} = \left( \beta \mathbf{R} + \sum_{i=1}^N \mathbf{C}^{iT} \mathbf{A}^{i-T} \mathbf{M}^i \mathbf{A}^{i-1} \mathbf{C}^i \right). \quad (6.4)$$

These equations are much simpler to solve, mainly for their size and their structure: For the second and third equations, which are linearized forward and adjoint problems, efficient solvers are usually available. Since the equations for the individual experiments are independent, they can also be solved in parallel. The system in the first equation, (6.1), is small, its size being equal to the number  $\#\delta q_{k,h}$  of discretized parameters  $\delta q_{k,h}$ , which is *much* smaller than the total number of degrees of freedom and in particular independent of the number of experiments. Furthermore,  $\mathbf{S}$  has some nice properties:

LEMMA 6.1. *The Schur complement matrix  $\mathbf{S}$  is symmetric and positive definite if at least  $\beta \mathbf{R}$  as defined above is positive definite.*

*Proof.* The proof of symmetry is trivial, noting that both  $\mathbf{R}$  and  $\mathbf{M}$  stem from second derivatives and are therefore symmetric matrices. Because  $m(\cdot)$  and  $r(\cdot)$  were assumed to be convex,  $M$  and  $R$  are also at least positive semidefinite. Consequently,  $\mathbf{v}^T \mathbf{S} \mathbf{v} = \sum_{i=1}^N (\mathbf{A}^{i-1} \mathbf{C}^i \mathbf{v})^T M (\mathbf{A}^{i-1} \mathbf{C}^i \mathbf{v}) + \beta \mathbf{v}^T \mathbf{R} \mathbf{v} > 0$  for all vectors  $\mathbf{v}$  and  $\mathbf{S}$  is positive definite.  $\square$

By consequence of the lemma, we can use well-known and fast iterative methods for the solution of this equation, such as the Conjugate Gradient method. In each matrix-vector multiplication we have to perform one linearized forward and one backward step for each experiment for which, again, we can assume that efficient solvers are available. The other operations are comparably cheap.

Of crucial importance for the speed of convergence of the CG method is the condition number of the Schur complement matrix  $\mathbf{S}$ . Numerical experiments have shown that, in contrast to the original matrix (5.2), the condition number only grows as  $\mathcal{O}(h^{-4})$ , i.e. by two orders of  $h$  less than the full matrix [4]. Furthermore and even more importantly, the condition number improves if more experiments are available, i.e.  $N$  is higher, corresponding to the fact that more information reduces the ill-posedness of the problem [37]. In particular, it is not hard to show using Rayleigh quotients for the largest and smallest eigenvalues that the condition number of the Schur complement matrix is not greater than the maximal condition number of its building blocks, i.e. that

$$\kappa(\mathbf{S}) \leq \max \left\{ \kappa(\mathbf{R}), \max_{1 \leq i \leq N} \kappa \left( \mathbf{C}^i \mathbf{A}^{i-T} \mathbf{M}^i \mathbf{A}^{i-1} \mathbf{C}^i \right) \right\},$$

assuming that both  $\mathbf{R}$  and  $\mathbf{C}^i \mathbf{A}^{i-T} \mathbf{M}^i \mathbf{A}^{i-1} \mathbf{C}^i$  are regular. In practice, the condition number  $\kappa(\mathbf{S})$  of the joint inversion matrix is often significantly smaller than that of the single experiment inversion matrices [37].

Finally, the CG method allows us to terminate the iteration relatively early. This is important since high accuracy is not required in the computation of search directions. Experience shows that for typical cases, a good solution can be obtained with 10–30 iterations, even if the size of  $\mathbf{S}$ ,  $\#\delta q_{k,h}$ , is several hundred to a few thousand.

The solution of the Schur complement equation can be accelerated by preconditioning the matrix. Since one will not usually build up the matrix, a preconditioner cannot make use of the individual matrix elements. However, other approaches have been investigated in the literature, see for example [14, 49].

Finally, we note that the Schur complement formulation is simple to parallelize (see [4]): matrix-vector multiplications with  $\mathbf{S}$  are easily performed in parallel due to the sum structure of this matrix, and the remaining two equations defining the updates for the state and adjoint variables are independent anyway.

**7. Bound constraints.** In the previous sections, we have described an efficient scheme for the discretization and solution of the inverse problem (2.5). However, in practical applications, one often has more information on the parameter than included in the formulation so far. For example, lower and upper bounds  $q_0 \leq q(\mathbf{x}) \leq q_1$  may be known, possibly only in parts of the domain, or with spatially dependent bounds. Such inequalities typically denote prior physical knowledge about the material properties we would like to identify, but even if such knowledge is absent, we will often want to impose constraints of the form  $q \geq q_0 > 0$  if  $q$  appears as a coefficient in an elliptic operator (as in the model problem).

In this section, we will extend the scheme developed above to incorporate such bounds, and we will show that the inclusion of these bounds comes at essentially no additional cost, since it only reuses information that is already there. On the contrary, as it reduces the size of the problems, it makes its solution faster. We would also like to stress that the approach does not make use of the actual form of state equations, misfit or regularization functionals; it is therefore possible to implement it in a very generic way inside the Newton solver. The approach is based on the same ideas that active set methods use (see, e.g., [46]) and is similar to the gradient projection–reduced Newton method [50]. However, since we consider problems with several thousands or more parameters, some parts of the algorithm have to be devised differently. In particular, the determination of the active set has to happen on the continuous level, as discussed

in the Introduction. For related approaches to constrained optimization problems in partial differential equations, we refer to [12, 41, 52].

**Basic idea.** Since the method to be introduced is simple to extend to the more general case, let us describe the basic idea here for the special case that  $q$  is only one scalar parameter function, and that we only have lower bounds,  $q_0 \leq q(\mathbf{x})$ . The approach is then: before each step, identify those regions where the parameters are already at their bounds and we expect their values to move out of the feasible region. Let us denote this part of the domain, the so-called *active set*, by  $\mathcal{I} = \{\mathbf{x} \in \Omega : q_k(\mathbf{x}) = q_0, \delta q_k(\mathbf{x}) \text{ presumably } < 0\}$ . After discretization,  $\mathcal{I}$  will usually be the union of a number of cells from  $\mathbb{T}_k^q$ .

We then have to answer two questions: how do we identify  $\mathcal{I}$ , and once we have found it what do we do with the parameter degrees of freedom inside  $\mathcal{I}$ ? Let us start with the second question: In order to prevent these parameters from moving further outside, we simply set the respective updates to zero, and for this augment the definition (4.1) of the Gauss-Newton step by a corresponding equality condition:

$$\begin{aligned} \min_{\delta \mathbf{u}_k, \delta q_k} & J(\mathbf{u}_k, q_k) + J_u(\mathbf{u}_k, q_k)(\delta \mathbf{u}_k) + J_q(\mathbf{u}_k, q_k)(\delta q_k) \\ & + J_{uu}(\mathbf{u}_k, q_k)(\delta \mathbf{u}_k, \delta \mathbf{u}_k) + J_{qq}(\mathbf{u}_k, q_k)(\delta q_k, \delta q_k) \quad (7.1) \\ \text{such that} & A^i(q_k; u_k^i)(\varphi^i) + A_u^i(q_k; u_k^i)(\delta u_k^i, \varphi^i) + A_q^i(q_k; u_k^i)(\delta q_k, \varphi^i) = 0, \\ & (\delta q_k, \xi)_{\mathcal{I}} = 0, \end{aligned}$$

where the last constraint is to hold for all test functions  $\xi \in L^2(\mathcal{I})$ .

The optimality conditions for this minimization problem are then equal to the original ones stated in (4.2), except that the last equation has to be replaced by

$$\sum_i A_q^i(q_k; u_k^i)(\chi, \delta \lambda_k^i) + \beta r_{qq}(q_k)(\delta q_k, \chi) + (\mu, \chi)_{\mathcal{I}} = -L_q(x_k)(\chi), \quad (7.2)$$

where  $\mu$  is the Lagrange multiplier corresponding to the constraint  $\delta q_k|_{\mathcal{I}} = 0$ .

These equations can be discretized in the same way as before. In particular, we take the same space  $\mathcal{Q}_h$  for the discrete Lagrange multiplier  $\mu$  as for  $\delta q_k$ . After performing the same block elimination procedure we used for (5.2), we then get as matrix the following system to compute the Lagrange multipliers and parameter updates:

$$\begin{pmatrix} \mathbf{S} & \mathbf{B}_{\mathcal{I}}^T \\ \mathbf{B}_{\mathcal{I}} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \delta q_{k,h} \\ \mu_h \end{pmatrix} = \begin{pmatrix} \mathbf{F}_{red} \\ 0 \end{pmatrix}, \quad (7.3)$$

with the reduced right hand side  $\mathbf{F}_{red}$  equal to the right hand side of (6.1). The equations identifying  $\delta \mathbf{u}_{k,h}$  and  $\delta \boldsymbol{\lambda}_{k,h}$  are exactly as in (6.2) and (6.3), and are solved once  $\delta q_{k,h}$  is available.

The matrix  $\mathbf{B}_{\mathcal{I}}$  appearing in (7.3) is of mass matrix type. If we denote by  $\mathcal{I}_h$  the set of indices of those basis functions in  $\mathcal{Q}_h$  with a support that intersects  $\mathcal{I}$ , and  $\mathcal{I}_h(k)$  its  $k$ th element, then  $\mathbf{B}_{\mathcal{I}}$  is of size  $\#\mathcal{I}_h \times \#\delta q_{k,h}$ , and  $(\mathbf{B}_{\mathcal{I}})_{kl} = (\chi_{\mathcal{I}_h(k)}, \chi_l)_{\mathcal{I}}$ . In this way, the last row of the system,  $\mathbf{B}_{\mathcal{I}} \delta q_{k,h} = 0$ , simply sets parameter updates in the selected region to zero.

Let us now denote by  $\mathbf{Q}$  the projector onto the feasible set for  $\delta q_{k,h}$ , i.e. it is a rectangular matrix of size  $(\#\delta q_{k,h} - \#\mathcal{I}_h) \times \#\delta q_{k,h}$ , where we have a row for each degree of freedom  $i \notin \mathcal{I}_h$  with a 1 at position  $i$ , such that  $\mathbf{Q} \mathbf{B}_{\mathcal{I}}^T = 0$ . Elementary calculations then yield that the updates we seek satisfy

$$\begin{bmatrix} \mathbf{Q} \mathbf{S} \mathbf{Q}^T \end{bmatrix} (\mathbf{Q} \delta q_{k,h}) = \mathbf{Q} \mathbf{F}_{red}, \quad \mathbf{B}_{\mathcal{I}} \delta q_{k,h} = \mathbf{0},$$

which are conditions for disjoint subsets of parameter degrees of freedom. Besides being smaller, the reduced Schur complement  $\mathbf{Q}\mathbf{S}\mathbf{Q}^T$  inherits the following desirable properties from  $\mathbf{S}$ :

LEMMA 7.1. *The reduced Schur complement  $\mathbf{S}_{red} = \mathbf{Q}\mathbf{S}\mathbf{Q}^T$  is symmetric and positive definite. Its condition number satisfies  $\kappa(\mathbf{S}_{red}) \leq \kappa(\mathbf{S})$ .*

*Proof.* While symmetry is obvious, we inherit (positive) definiteness from  $\mathbf{S}$  by the fact that the matrix  $\mathbf{Q}$  has by construction full row rank. For the proof of the condition number estimate, let  $N^q = \#\delta q_{k,h}$ ,  $N_{red}^q = N^q - \#\mathcal{I}_h$ ; then we have for the maximal eigenvalue of  $\mathbf{S}_{red}$ :

$$\Lambda(\mathbf{S}_{red}) = \max_{\substack{\mathbf{v} \in \mathbb{R}^{N_{red}^q} \\ \|\mathbf{v}\|=1}} \mathbf{v}^T \mathbf{S}_{red} \mathbf{v} = \max_{\substack{\mathbf{w} \in \mathbb{R}^{N^q} \\ \|\mathbf{w}\|=1 \\ \mathbf{w}|_{\mathcal{I}_h}=0}} \mathbf{w}^T \mathbf{S} \mathbf{w} \leq \max_{\substack{\mathbf{w} \in \mathbb{R}^{N^q} \\ \|\mathbf{w}\|=1}} \mathbf{w}^T \mathbf{S} \mathbf{w} = \Lambda(\mathbf{S}).$$

Similarly, we get for the smallest eigenvalue  $\lambda(\mathbf{S}_{red}) \geq \lambda(\mathbf{S})$ .  $\square$

In practice,  $\mathbf{S}_{red}$  needs not be built up for use in a conjugate gradient method. Since application of  $\mathbf{Q}$  is essentially for free, the inversion of  $\mathbf{Q}\mathbf{S}\mathbf{Q}^T$  for the constrained updates is at most as expensive as that of  $\mathbf{S}$  for the unconstrained ones, and possibly cheaper if the condition number is indeed smaller. It is worth noting that treating constrained nodes in this way does not imply knowledge of the actual problem under consideration: if we have code to produce the matrix-vector product with  $S$ , then adding bound constraints is simple.

This approach has several advantages. First, in the implementation of solvers for the state equations, one does not have to care about constraints as one would need to if positivity of a parameter were enforced by replacing  $q$  by  $e^q$ . Secondly, it is simple to add bound constraints in the Schur complement formulation, while it would be more complicated to add them to a solver operating directly on (5.2).

**Determination of the active set.** There remains the question of how to determine the set of parameter updates we want to constrain to zero. For this, let us for a moment consider  $\mathcal{I}$  as an unknown set that is implicitly determined by the fact that the constraint is active there at the solution. The idea of active set methods is then the following: from (7.2), we see that at the optimum there holds  $(\mu, \chi)_{\mathcal{I}} = -L_q(x)(\chi)$  for all test functions  $\chi$ . Outside  $\mathcal{I}$ ,  $\mu$  should be zero, and optimization theory tells us that it must be negative inside. If we have not yet found the solution, these properties do not hold exactly, but as we approach the solution, the updates  $\delta\lambda_k, \delta q_k$  become small and we can use the identity to get an approximation  $\tilde{\mu}_k$  to the Lagrange multiplier defined on all of  $\Omega$ . If we discretize it using the same space  $\mathcal{Q}_h$  as for the parameters, then we can define  $\tilde{\mu}_{k,h}$  by

$$(\tilde{\mu}_{k,h}, \chi_h) = -L_q(x_{k,h})(\chi_h) \quad \forall \chi_h \in \mathcal{Q}_h.$$

We will then use  $\tilde{\mu}_{k,h}$  as an indicator whether a point lies inside the set where the constraint on  $q$  is active and define

$$\mathcal{I}_h = \{\mathbf{x} \in \Omega : q_{k,h}(\mathbf{x}) = q_0, \tilde{\mu}_{k,h}(\mathbf{x}) \leq -\varepsilon\},$$

with a small positive number  $\varepsilon$ . With the so fixed set  $\mathcal{I}_h$ , the algorithm proceeds as above. Since  $-L_q(x_{k,h})(\chi_h)$  is already available as the right hand side of the discretized Gauss-Newton step, computing  $\tilde{\mu}_{k,h}$  only requires the inversion of the mass matrix resulting from the left hand side term  $(\mu_{k,h}, \chi_h)$ . This is particularly cheap if  $\mathcal{Q}_h$  is made up of discontinuous shape functions.

Numerical experiments indicate that it is necessary to set up this scheme in a function space first, and only discretize afterwards. Enforcing bounds only after discretization would amount to replacing the mass matrix by the identity matrix. This would then lead to the elements of the Lagrange multiplier  $\tilde{\mu}_{k,h}$  having a size that scales with the size of the cell they are defined on, preventing us from comparing their size with a fixed number  $\epsilon$  in the definition of the set  $\mathcal{I}_h$ .

**8. Numerical examples.** In this section, let us give some examples of computations that have been performed with an implementation of the framework laid out above. The first two examples are applications of the model problem defined in Section 3, i.e. we want to recover the spatially dependent coefficient  $q(x)$  in a Laplace-type operator  $-\nabla \cdot (q\nabla)$  from measurements of the state variable. In one or two space dimensions, this is a model of a bar or membrane of variable stiffness that is subjected to a known force; the stiffness coefficient is then identified by measuring the deflection at every point. Similar applications arise also in groundwater management, where the hydraulic head satisfies a Poisson equation with  $q$  being the water permeability, as well as in biomedical imaging methodologies such as electric impedance tomography [18] or ultrasound-modulated optical tomography [51].

The third example deals with a parameter estimation problem in fluorescence enhanced optical tomography and will be explained in Section 8.3. Further examples of the present framework to Helmholtz-type equations with high wave numbers, as appearing in seismic imaging, can be found in [4].

In the examples below, mesh adaptation is performed by simply looking at the smoothness of the solution. In particular, we use an indicator for the size of the second derivative of  $u^i$  to refine the meshes  $\mathbb{T}^i$  as is appropriate when using first order continuous finite elements, and an indicator for the gradient of  $q$  to refine  $\mathbb{T}^q$  as we use piecewise constant shape functions. In both cases, approximations of these derivatives are scaled by appropriate powers of the mesh size  $h$ . More details on our mesh refinement strategies can be found in [4, 36].

The program used here is built on the open source finite element library `deal.II` [5, 6] and runs on multiprocessor machines or clusters of computers.

**8.1. Example 1: A single experiment.** In this first example, we consider the model problem introduced in Section 3 with  $N = 1$ , i.e. we attempt to identify a possibly discontinuous coefficient from a single global measurement of the solution of a Poisson equation. This corresponds to the situation

$$A(q; u)(\varphi) = (q\nabla u, \nabla \varphi) - (f, \varphi), \quad m(Mu - z) = \frac{1}{2} \|u - z\|_{\Omega}^2,$$

where  $\Omega = [-1, 1]^d, d = 2$ . Measurement data  $z$  was generated synthetically by solving  $-\nabla \cdot q^* \nabla u^* = f$  numerically for  $u^*$  using a higher order method (to avoid the inverse crime), and setting  $z(\mathbf{x}) = u^*(\mathbf{x}) + \varepsilon(\mathbf{x})$ , where  $\varepsilon$  is random noise with a fixed amplitude  $\|\varepsilon\|_{\infty}$ .

For this example, we choose  $q^*$  as

$$q^*(\mathbf{x}) = \begin{cases} 1 & \text{for } |\mathbf{x}| < \frac{1}{2}, \\ 8 & \text{otherwise,} \end{cases} \quad f(\mathbf{x}) = 2d,$$

which yields  $u^*(\mathbf{x}) = |\mathbf{x}|^2$  inside  $|\mathbf{x}| < \frac{1}{2}$ , and  $u^*(\mathbf{x}) = \frac{1}{8}|\mathbf{x}|^2 + \frac{7}{32}$  otherwise. Boundary conditions  $g$  for  $u$  are chosen accordingly. The circular jump in the coefficient is not



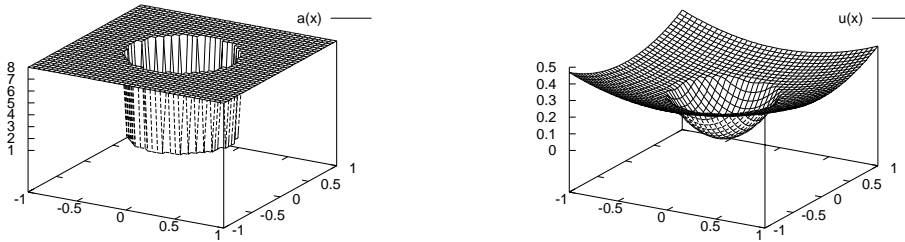


FIG. 8.1. *Example 1: Exact coefficient  $q^*$  (left) and displacement  $u^*$  (right).*

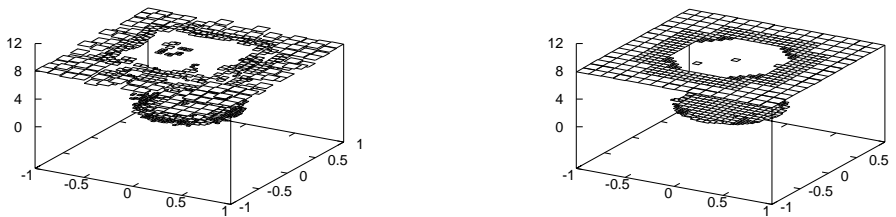


FIG. 8.2. *Example 1: Recovered coefficient with no noise, on grids  $\mathbb{T}^q$  with 800-900 degrees of freedom. Left: No bounds on  $q$  imposed. Right:  $1 \leq q \leq 8$  imposed.*

aligned with the mesh cells, and can only be resolved properly by mesh refinement.  $u^*$  and  $q^*$  are shown in Figure 8.1.

For the case of no noise, i.e. measurements can be made everywhere without error, Figure 8.2 shows the mesh  $\mathbb{T}^q$  and the identified parameter after some refinement steps. The left panel shows the reconstruction with no bounds on  $q$  imposed, whereas the right one shows results with tight bounds  $1 \leq q \leq 8$ . The latter case can be considered typical if one knows that a body is composed of two different materials, but their interface is unknown. In both cases, the accuracy of reconstruction is good, and it is clear that adding bound information stabilizes the process. No regularization is used for this experiment.

On the other hand, if  $\|\varepsilon\|_\infty/\|z\|_\infty = 2\%$  noise is present, Figure 8.3 shows the identified coefficient without and with bounds imposed on the parameter. Again, no regularization is used, and it is obvious that the additional information of bounds on the parameter improves the result significantly (quantitative results are given as part of the next section). Of course, adding a regularization term, for example of Bounded Variation-type [22, 24, 27], would also aid to get a better reconstruction. Instead of regularization, we will rather consider noise suppression by multiple measurements in the next section.

**8.2. Example 2: Multiple experiments.** Let us consider the same situation as in the previous section, but this time we perform multiple experiments with different

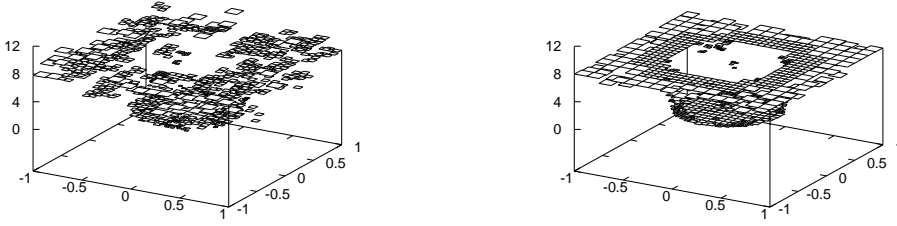


FIG. 8.3. *Example 1: Same as Fig. 8.2, but with 2% noise in the measurement.*

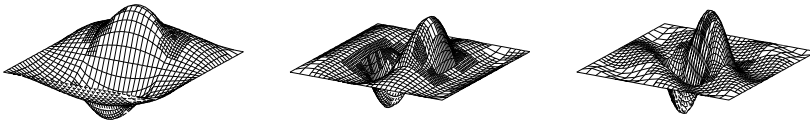


FIG. 8.4. *Example 2: Solutions of the state equations for experiments  $i = 2, 6, 12$ .*

forcing  $f^i$ , producing measurements  $z^i$ . Thus, for each experiment  $1 \leq i \leq N$ ,

$$A^i(q; u^i)(\varphi) = (q \nabla u^i, \nabla \varphi) - (f^i, \varphi), \quad m^i(M^i u^i - z^i) = \frac{1}{2} \|u^i - z^i\|_{\Omega}^2. \quad (8.1)$$

Our hope is that if each of these measurements is noisy, we can still recover the correct coefficient well if we only measure often enough. Since the measurements have independent noise, measuring more than once would already yields a gain even if we chose the right hand sides  $f^i$  identically. However, we expect to gain more if we use different forcing functions  $f^i$  in different experiments.

In addition to  $f^1(\mathbf{x}) = 2d$  already used in the last example, we use

$$f^i(\mathbf{x}) = \pi^2 \mathbf{k}_i^2 \sin(\pi \mathbf{k}_i \cdot \mathbf{x}), \quad 2 \leq i \leq N$$

as forcing terms for the rest of the state equations (8.1). The vectors  $\mathbf{k}_i$  are chosen as the first  $N$  elements of the integer lattice  $\{0, 1, 2, \dots\}^d$  when ordered by their  $l_2$ -norm and after eliminating collinear pairs in favor of the element of smaller magnitude. Numerical solutions for these right hand sides are shown in Fig. 8.4 for  $i = 2, 6, 12$ . Synthetic measurements  $z^i$  were obtained as in the first example.

Fig. 8.5 shows a quantitative comparison of the reconstruction error  $\|q_h - q^*\|_{L^2(\Omega)}$ , as we increase the number of experiments used for the reconstruction, and as Newton iterations proceed on successively finer grids. In most cases, we only perform one Newton iteration on each grid, but if we are not satisfied with the progress on this grid, more than one iteration will be done; in this case, curves in the charts have vertically stacked data points. The finest discretizations had 3-400,000 degrees of freedom for the discretization of state and adjoint variable in each experiment (i.e., up to a total of about 4 million unknowns in the examples shown), and about 10,000 degrees of freedom for the discretization of the parameter  $q_h$ . We only show the case

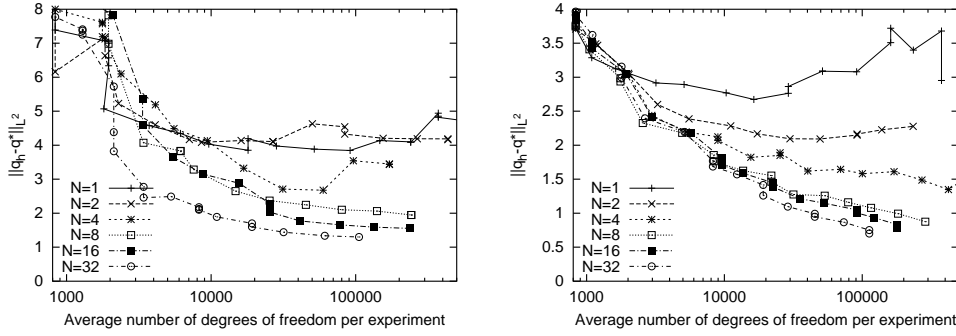


FIG. 8.5. Error  $\|q_h - q^*\|_{L^2(\Omega)}$  in the reconstructed coefficient as a function of the number  $N$  of experiments used in the reconstruction and the average number of degrees of freedom used in the discretization of each experiment. Left: No bounds imposed. Right:  $1 \leq q \leq 8$  imposed. 2% noise in both cases. Note the different scales.

of non-zero noise level, since otherwise the number of experiments was not relevant for the reconstruction error.

From these numerical results, several conclusions can be drawn. First, imposing bounds helps identify significantly more accurate reconstructions, but using more measurements also strongly reduces the effects of noise. Secondly, if noise is present, there is a limit for the amount of information that can be obtained; as can be seen from the erratic and growing behavior of curves for small  $N$  and large numbers of degrees of freedom, further refining meshes may deteriorate the result beyond a certain mesh size (the identified parameter deteriorates by high frequency oscillations). Finally, since the numerical effort required to solve the problem grows roughly linear with the number of experiments, using more experiments may be cheaper than using finer meshes in many cases: discretizing twice as many experiments yields better reconstructions of  $q$  than choosing meshes with twice as many unknowns.

**8.3. Example 3: Optical tomography.** The third and last application comes from a relatively recent biomedical imaging technique, fluorescent-enhanced optical tomography. The state equations in this case consist of two coupled equations

$$-\nabla \cdot [D_x \nabla w] + k_x w = 0, \quad -\nabla \cdot [D_m \nabla v] + k_m v = b_{xm} q w. \quad (8.2)$$

These equations describe the propagation of light in tissue and are the diffusion approximation of the full radiative transfer equation. Here,  $w$  is the light intensity at the wave length of a laser with which the skin is illuminated.  $v$  is the intensity of fluorescent light excited in the interior by the incident light in the presence of a fluorescent dye of unknown concentration  $q$ . If the incident light intensity is modulated at a frequency  $\omega$ , then both  $w$  and  $v$  are complex-valued functions and the various coefficients in the equations above are given by

$$D_x = \frac{1}{3(\mu_{axi} + q + \mu'_{sx})}, \quad k_x = \frac{i\omega}{c} + \mu_{axi} + q, \quad b_{xm} = \frac{\phi}{1 - i\omega\tau},$$

$$D_m = \frac{1}{3(\mu_{ami} + \mu_{amf} + \mu'_{sm})}, \quad k_m = \frac{i\omega}{c} + \mu_{ami} + \mu_{amf},$$

where  $\mu_{axi}, \mu_{ami}$  are intrinsic absorption coefficients at incident and fluorescent wave lengths,  $\mu'_{sx}, \mu'_{sm}$  are reduced scattering coefficients,  $\mu_{amf}$  absorption due fluorophore,

$\phi$  the fluorophore's quantum efficiency,  $\tau$  its half life, and  $c$  the speed of light. All of these coefficients are assumed known. More details about this model and the actual values of material parameters can be found in [35, 37].

In clinical applications, one injects a fluorescent dye into tissue that is suspected to have a tumor. Since certain dyes specifically bind to tumor cells while they are washed out from the rest of the tissue, their presence is considered to be a good indicator for the existence and location of a tumor. The goal of the inverse problem is therefore to identify the unknown concentration of fluorescent dye,  $q = q(\mathbf{x})$ , in the tissue, using above model. Note that  $q$  appears in the diffusion coefficient  $D_x$ , the absorption coefficient  $k_x$ , and the right hand side of the second equation. In order to identify  $q$  one illuminates the body at the incident wave length (but not at the fluorescent wave length) with a laser, which we can model using the boundary conditions

$$2D_x \frac{\partial w}{\partial n} + \gamma w + S = 0, \quad 2D_m \frac{\partial v}{\partial n} + \gamma v = 0, \quad (8.3)$$

where  $n$  denotes the outward normal to the surface and  $\gamma$  is a constant depending on the optical reflective index mismatch at the boundary [26], and  $S(\mathbf{x})$  is the intensity pattern of the incident laser light. We would then measure the fluorescent intensity  $v(\mathbf{x})$  on a part  $\Sigma$  of the boundary  $\partial\Omega$ . Intuitively, in areas of  $\Sigma$  where we see much fluorescent light, a fluorescent source must be close by, i.e. the dye concentration is large pointing to the presence of a tumor.

Given these explanations, we can define the inverse problem in the language of Section 2 by setting  $u^i = \{w^i, v^i\} \in V^i = [H^1(\Omega \rightarrow \mathbb{C})]^2$ , defining test functions  $\varphi^i = \{\zeta^i, \xi^i\} \in V^i$ , and using

$$\begin{aligned} A(q; u^i)(\varphi^i) &= (D_x \nabla u^i, \nabla \zeta^i)_\Omega + (k_x u^i, \zeta^i)_\Omega + \frac{\gamma}{2}(u^i, \zeta^i)_{\partial\Omega} + \frac{1}{2}(S^i, \zeta^i)_{\partial\Omega} \\ &\quad + (D_m \nabla v^i, \nabla \xi^i)_\Omega + (k_m v^i, \xi^i)_\Omega + \frac{\gamma}{2}(v^i, \xi^i)_{\partial\Omega} - (b_{xm} u^i, \xi^i)_\Omega \end{aligned}$$

as the bilinear form, where all scalar products apply to complex-valued quantities. The measurement operator is given by  $M^i : V^i \mapsto \mathcal{Z}^i = L^2(\Sigma \rightarrow \mathbb{C})$ ,  $M^i u^i = v^i|_\Sigma$  and we use  $m^i(M^i u^i - z^i) = \frac{1}{2}\|v^i - z^i\|_{L^2(\Sigma)}^2$ .

Figs 8.6 and 8.7 show results obtained with a program that implements the framework laid out before for these equations and operators. It shows a situation in which a simulated widened laser line is scanned in  $N = 8$  increments over an area of roughly  $8cm \times 8cm$  of the experimentally determined surface of a tissue sample (in this case the groin region of pig, see [38]). Synthetic data  $z^i$  is generated assuming that a spherical tumor of  $1cm$  diameter is located at the center of the scene some  $1.5cm$  below the surface. This data is then used to reconstruct the function  $q(\mathbf{x})$  which should ideally match the previously assumed size and location of the tumor.

Fig. 8.6 shows the real parts of the current iterates  $u_{25}^2, u_{25}^6$  after 25 Gauss-Newton iterations for experiments 2 and 6, along with the mesh  $\mathbb{T}_{25}^6$  used to discretize the latter. This mesh has approximately 22,900 cells, on which a total of some 270,000 primal and adjoint variables are discretized. The total number of unknowns involved in this inverse problem, added up over all  $N = 8$  experiments, is some 1.5 million. It is quite clear that a single mesh able to resolve the features of *all* forward solutions would have to have significantly more degrees of freedom since the areas where  $u^i(\mathbf{x})$  varies are different between experiments. A back of the envelope calculation shows that this

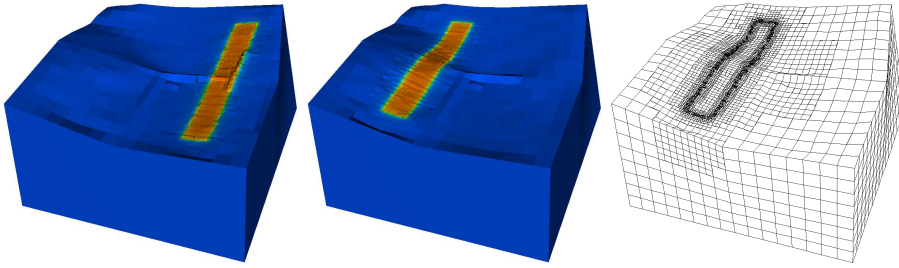


FIG. 8.6. *Example 3: Left and middle: Real part of the solution  $w$  of model (8.2)–(8.3) for experiments 2 and 6, characterized by different boundary sources  $S^2(\mathbf{x}), S^6(\mathbf{x})$ . Right: Mesh  $\mathbb{T}^6$  after 4 refinement cycles.*

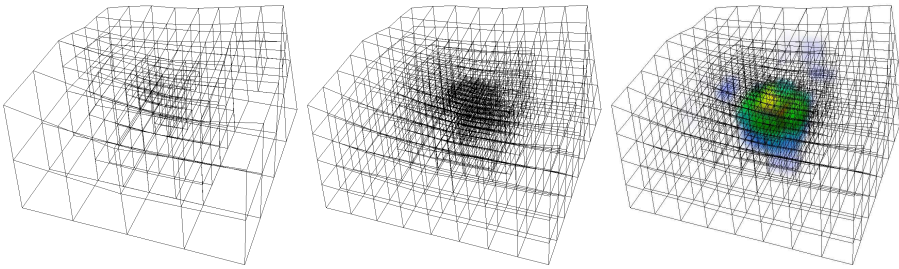


FIG. 8.7. *Example 3: Left and middle: Meshes  $\mathbb{T}^9$  used to discretize the parameter  $q(\mathbf{x})$  after 1 and 4 refinement cycles, respectively. The left mesh is used for Gauss-Newton iterations 8–11, the one in the middle for iterations 22–25. Right: Reconstructed parameter  $q_{25}$  after 25 Gauss-Newton iterations.*

single mesh would have to have on the order of 2 million cells, giving rise to a total number of degrees of freedom on the order of 10–12 million. Since the solution of the inverse problem is not dominated by generating meshes but by solving the linear systems (6.1)–(6.3), the savings resulting from using adaptive meshes are apparent.

Finally, Fig. 8.7 shows the meshes  $\mathbb{T}_{11}^q$  and  $\mathbb{T}_{25}^q$ , as well as a cloud image of the solution after 25 Gauss-Newton iterations. The reconstruction has correctly identified the location and size of the tumor, and the mesh is appropriately refined to resolve its features. The image does contain a few artifacts, mainly below the target and elsewhere deep inside the tissue; this is not overly surprising given that light does not penetrate very deep into tissue. However, the main features are clearly correct. (For similar reconstructions, in particular using experimentally measured instead of synthetically generated data, see also [35, 38].)

In the case shown, the final mesh has 977 unknowns, of which 438 are constrained by the condition  $0 \leq q \leq 2.5$  (the upper bound is, in fact, not attained here). Over the course of the entire 25 Gauss-Newton iterations, some 9,000 CG iterations were performed to solve the Schur complement systems. Since each iteration involves  $2N$  solves with  $A^i$  or  $A^{iT}$ , and we also need  $2N$  solves for equations (6.2)–(6.3) per Gauss-Newton step, this amounts to a total of some 150,000 solutions of the three-dimensional, coupled system (8.2)–(8.3) over the course of the entire computation which took some 6 hours on a 2.2GHz Opteron system; approximately two thirds of the total compute time were spent on the last 4 iterations on the finest grid, underlining the claim that the initial iterations on coarser meshes are relatively cheap.

**9. Conclusions.** Adaptive meshing strategies have become the state-of-the-art technique in solving partial differential equations. However, they are not yet widely used in solving inverse problems. This may be attributed in part to the fact that the numerical solution of inverse problems has, in the past, been considered more in the context of optimization than in the context of discretizations. Since practical optimization methods are mostly developed in finite dimensions, the notion that discretizations can change over the course of a computation therefore doesn't fit very well into existing algorithms.

To merge these two streams of research, optimization and discretization, we have presented a framework for the solution of large-scale multiple-experiment inverse problems that can deal with adaptively changing meshes. Its main features are:

- formulation in function spaces, allowing for different discretizations in subsequent steps of Newton-type nonlinear solvers;
- discretization by adaptive finite element schemes, with different meshes for state and adjoint variables on the one hand, and the parameters sought on the other;
- inclusion of bound constraints with a simple but efficient active set strategy;
- choice of a formulation that allows for efficient parallelization.

This framework has then been applied to some examples showing that inclusion of bounds can stabilize the identification of a coefficient from noisy data, as well as the (obvious) fact that measuring more than once can reduce the effects of noise. The last example also demonstrated that the framework is applicable also to problems of realistic complexity beyond mathematical model problems.

There are obviously many aspects of the framework laid out here that warrant further research. Among these are:

- Refinement indicators: For the numerical examples, only a rather simple criterion was used to drive mesh refinement. Ideally, refinement would be based on error estimates. There are some attempts in this direction [4, 7, 40, 41]. One problem is that traditionally, error estimates are derived using stability estimates; however, these rarely hold for inverse problems.
- More efficient linear solvers: The majority of the compute time is spent on inverting the Schur complement system (6.1) because constructing preconditioners for the matrix  $\mathbf{S}$  is complicated by the fact that its entries are not known. Attempts to use simpler versions of this matrix to precondition can be found in [14]. Another possibility are Krylov subspace recycling techniques using information from previous Gauss-Newton iterations. This, however, would have to deal with the fact that the matrix  $\mathbf{S}$  results from discretizations that may change between iterations.
- Very little theoretical justification, for example in terms of convergence proofs, has been presented for the viability of the individual steps of the framework laid out above.

We hope that future research will answer some of these open questions.

*Acknowledgments.* The author thanks Rolf Rannacher for the encouragement given for this research. It is also a pleasure to thank Amit Joshi for the collaboration on optical tomography, from which example 3 was generated.

#### REFERENCES

- [1] R. ACAR, *Identification of coefficients in elliptic equations*, SIAM J. Control Optim., 31 (1993), pp. 1221–1244.

- [2] U. M. ASCHER AND E. HABER, *Grid refinement and scaling for distributed parameter estimation problems*, Inverse Problems, 17 (2001), pp. 571–590.
- [3] ———, *A multigrid method for distributed parameter estimation problems*, tech. report, University of British Columbia, 2002.
- [4] W. BANGERTH, *Adaptive Finite Element Methods for the Identification of Distributed Parameters in Partial Differential Equations*, PhD thesis, University of Heidelberg, 2002.
- [5] W. BANGERTH, R. HARTMANN, AND G. KANSCHAT, *deal.II Differential Equations Analysis Library, Technical Reference*, 2007. <http://www.dealii.org/>.
- [6] ———, *deal.II – a general purpose object oriented finite element library*, ACM Trans. Math. Softw., 33 (2007, accepted).
- [7] W. BANGERTH AND R. RANNACHER, *Adaptive Finite Element Methods for Differential Equations*, Birkhäuser Verlag, 2003.
- [8] H. T. BANKS AND K. KUNISCH, *Estimation Techniques for Distributed Parameter Systems*, Birkhäuser, Basel–Boston–Berlin, 1989.
- [9] R. BECKER, *Adaptive finite elements for optimal control problems*. Habilitation thesis, University of Heidelberg, 2001.
- [10] R. BECKER, H. KAPP, AND R. RANNACHER, *Adaptive finite element methods for optimal control of partial differential equations: Basic concept*, SIAM J. Contr. Optim., 39 (2000), pp. 113–132.
- [11] H. BEN AMEUR, G. CHAVENT, AND J. JAFFRÉ, *Refinement and coarsening indicators for adaptive parametrization: application to the estimation of hydraulic transmissivities*, Inverse Problems, 18 (2002), pp. 775–794.
- [12] M. BERGOUNIOUX, K. ITO, AND K. KUNISCH, *Primal-dual strategy for constrained optimal control problems*, SIAM J. Control Optim., 37 (1999), pp. 1176–1194.
- [13] L. T. BIEGLER, O. GHATTAS, M. HEINKENSCHLOSS, AND B. VAN BLOEMEN WAANDERS, eds., *Large-Scale PDE-Constrained Optimization*, Springer, 2003.
- [14] G. BIROS AND O. GHATTAS, *Parallel Lagrange-Newton-Krylov-Schur methods for PDE-constrained optimization. Part I: The Krylov-Schur solver*, SIAM J. Sci. Comput., 27 (2005), pp. 687–713.
- [15] ———, *Parallel Lagrange-Newton-Krylov-Schur methods for PDE-constrained optimization. Part II: The Lagrange-Newton solver and its application to optimal control of steady viscous flow*, SIAM J. Sci. Comput., 27 (2005), pp. 714–739.
- [16] Å. BJÖRCK, *Numerical Methods for Least Squares Problems*, SIAM, 1996.
- [17] H. G. BOCK, *Randwertproblemmethoden zur Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen*, vol. 183 of Bonner Mathematische Schriften, University of Bonn, Bonn, 1987.
- [18] L. BORCEA, *Electrical impedance tomography*, Inverse Problems, 18 (2002), pp. R99–R136.
- [19] L. BORCEA AND V. DRUSKIN, *Optimal finite difference grids for direct and inverse Sturm Liouville problems*, Inverse Problems, 18 (2002), pp. 979–1001.
- [20] L. BORCEA, V. DRUSKIN, AND L. KNIZHNERMAN, *On the continuum limit of a discrete inverse spectral problem on optimal finite difference grids*, Comm. Pure. Appl. Math., 58 (2005), pp. 1231–1279.
- [21] J. CARTER AND C. ROMERO, *Using genetic algorithms to invert numerical simulations*, in ECMOR VIII: 8th European Conference on the Mathematics of Oil Recovery, Freiberg, Germany, European Association of Geoscientists and Engineers (EAGE), 2002, pp. E–45.
- [22] Z. CHEN AND J. ZOU, *An augmented Lagrangian method for identifying discontinuous parameters in elliptic systems*, SIAM J. Control Optim., 37 (1999), pp. 892–910.
- [23] M. J. DAYDÉ, J.-Y. L’EXCELLENT, AND N. I. M. GOULD, *Element-by-element preconditioners for large partially separable optimization problems*, SIAM J. Sci. Comp., 18 (1997), pp. 1767–1787.
- [24] F. DIBOS AND G. KOEPLER, *Global total variation minimization*, SIAM J. Numer. Anal., 37 (2000), pp. 646–664.
- [25] H. W. ENGL, M. HANKE, AND A. NEUBAUER, *Regularization of Inverse Problems*, Kluwer Academic Publishers, Dordrecht, 1996.
- [26] A. GODAVARTY, D. J. HAWRYSZ, R. ROY, AND E. M. SEVICK-MURACA, *The influence of the index-mismatch at the boundaries measured in fluorescence-enhanced frequency-domain photon migration imaging*, Optics Express, 10 (2002), pp. 653–662.
- [27] S. GUTMAN, *Identification of discontinuous parameters in flow equations*, SIAM J. Control Optimization, 28 (1990), pp. 1049–1060.
- [28] M. GUVEN, B. YAZICI, X. INTES, AND B. CHANCE, *An adaptive multigrid algorithm for region of interest diffuse optical tomography*, in Proceedings of the International Conference on Image Processing 2003, 2003, pp. II: 823–826.

- [29] M. GUVEN, B. YAZICI, K. KWON, E. GILADI, AND X. INTES, *Effect of discretization error and adaptive mesh generation in diffuse optical absorption imaging: I*, Inverse Problems, (2007, in print).
- [30] ———, *Effect of discretization error and adaptive mesh generation in diffuse optical absorption imaging: II*, Inverse Problems, (2007, in print).
- [31] E. HABER AND U. M. ASCHER, *Preconditioned all-at-once methods for large, sparse parameter estimation problems*, Inverse Problems, 17 (2001), pp. 1847–1864.
- [32] E. HABER, U. M. ASCHER, AND D. OLDENBURG, *On optimization techniques for solving non-linear inverse problems*, Inverse Problems, 16 (2000), pp. 1263–1280.
- [33] K. ITO, M. KROLLER, AND K. KUNISCH, *A numerical study of an augmented Lagrangian method for the estimation of parameters in elliptic systems*, SIAM J. Sci. Stat. Comput., 12 (1991), pp. 884–910.
- [34] C. R. JOHNSON, *Computational and numerical methods for bioelectric field problems*, Critical Reviews in BioMedical Engineering, 25 (1997), pp. 1–81.
- [35] A. JOSHI, W. BANGERTH, K. HWANG, J. C. RASMUSSEN, AND E. M. SEVICK-MURACA, *Fully adaptive fem based fluorescence optical tomography from time-dependent measurements with area illumination and detection*, Med. Phys., 33 (2006), pp. 1299–1310.
- [36] A. JOSHI, W. BANGERTH, AND E. M. SEVICK-MURACA, *Adaptive finite element modeling of optical fluorescence-enhanced tomography*, Optics Express, 12 (2004), pp. 5402–5417.
- [37] ———, *Non-contact fluorescence optical tomography with scanning patterned illumination*, Optics Express, 14 (2006), pp. 6516–6534.
- [38] AMIT JOSHI, WOLFGANG BANGERTH, RUCHI SHARMA, WEI WANG, AND EVA M. SEVICK-MURACA, *Molecular tomographic imaging of lymph nodes with NIR fluorescence*, in Proceedings of the IEEE International Symposium on Biomedical Imaging, Arlington, VA, 2007, IEEE, 2007, pp. 564–567.
- [39] C. KRAVARIS AND J. H. SEINFELD, *Identification of parameters in distributed parameter systems by regularization*, SIAM J. Control Optim., 23 (1985), pp. 217–241.
- [40] K. KUNISCH, W. LIU, AND N. YAN, *A posteriori error estimators for a model parameter estimation problem*, in Proceedings of the ENUMATH 2001 conference, 2002.
- [41] R. LI, W. LIU, H. MA, AND T. TANG, *Adaptive finite element approximation for distributed elliptic optimal control problems*, SIAM J. Control Optim., 41 (2002), pp. 1321–1349.
- [42] R. LUCE AND S. PEREZ, *Parameter identification for an elliptic partial differential equation with distributed noisy data*, Inverse Problems, 15 (1999), pp. 291–307.
- [43] X.-Q. MA, *A constrained global inversion method using an over-parameterised scheme: Application to poststack seismic data*. Geophysics Online, Aug. 2000.
- [44] M. MOLINARI, B. H. BLOTT, S. J. COX, AND G. J. DANIELL, *Optimal imaging with adaptive mesh refinement in electrical impedance tomography*, Physiological Measurement, 23 (2002), pp. 121–128.
- [45] M. MOLINARI, S. J. COX, B. H. BLOTT, AND G. J. DANIELL, *Adaptive mesh refinement techniques for electrical impedance tomography*, Physiological Measurement, 22 (2001), pp. 91–96.
- [46] J. NOCEDAL AND S. J. WRIGHT, *Numerical Optimization*, Springer Series in Operations Research, Springer, New York, 1999.
- [47] M. K. SEN, A. DATTA-GUPTA, P. L. STOFFA, L. W. LAKE, AND G. A. POPE, *Stochastic reservoir modeling using simulated annealing and genetic algorithms*, SPE Formation Evaluation, (1995), pp. 49–55.
- [48] A. TARANTOLA, *Inverse Problem Theory and Methods for Model Parameter Estimation*, SIAM, 2004.
- [49] C. R. VOGEL, *Sparse matrix computations arising in distributed parameter identification*, SIAM J. Matrix Anal. Appl., 20 (1999), pp. 1027–1037.
- [50] ———, *Computational Methods for Inverse Problems*, SIAM, 2002.
- [51] L. V. WANG, *Ultrasound-mediated biophotonic imaging: A review of acousto-optical tomography and photo-acoustic tomography*, Disease Markers, 19 (2003/2004), pp. 123–138.
- [52] W. H. YU, *Necessary conditions for optimality in the identification of elliptic systems with parameter constraints*, J. Optimization Theory Appl., 88 (1996), pp. 725–742.