

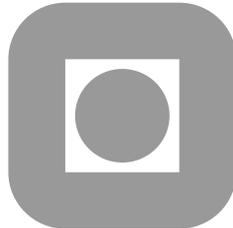
NORGES TEKNISK-NATURVITENSKAPELIGE  
UNIVERSITET

# Adaptive error control in inverse electromagnetic scattering

by

Larisa Beilina, Marte P. Hatlo, Harald E. Krogstad

PREPRINT  
NUMERICS NO. 4/2008



NORWEGIAN UNIVERSITY OF  
SCIENCE AND TECHNOLOGY  
TRONDHEIM, NORWAY

This report has URL

<http://www.math.ntnu.no/preprint/numerics/2008/N4-2008.pdf>

Address: Department of Mathematical Sciences, Norwegian University of Science and  
Technology, N-7491 Trondheim, Norway.



# Adaptive error control in inverse electromagnetic scattering

Larisa Beilina, Marte P. Hatlo, Harald E. Krogstad

May 20, 2008

In this paper we derive an a posteriori error estimate and present an adaptive algorithm for an inverse electromagnetic scattering problem.

The inverse problem is formulated as an optimal control problem, where we solve equations expressing stationarity of an associated Lagrangian. The a posteriori error estimate for the Lagrangian couples residuals of the computed solution to weights of the reconstruction. The performance of the adaptive finite element method and the usefulness of the a posteriori error estimate are illustrated in numerical examples.

## 1 Introduction

We apply a mesh-adaptive method, which was originally developed in [3, 5, 6, 7, 8], to an inverse electromagnetic scattering problem. The method is based on an a posteriori error estimate which couples residuals of the computed solution to weights in the reconstruction. A new element in the present work is the introduction of absorbing and Neumann boundary conditions in the formulation of the forward problem for PDE. Thus, the main contribution of this work is to derive an a posteriori error estimate for the Lagrangian in the presence of absorbing and Neumann boundary conditions, and apply this in an adaptive algorithm. The derivation of the a posteriori estimate follows the main approach to adaptive error control in computational differential equations, presented in [2, 15] and references therein.

The adaptivity technique for the classic forward problems is well known, see, e.g., [14, 15, 16, 17, 18]. A simplifying factor for a forward problem is that all coefficients of a PDE are known. However, an inverse problem is about approximating an unknown coefficient of a PDE. Thus, it seems to be, at the first glance, that in order to apply the adaptivity idea to an inverse problem, one needs to know its solution in advance, which is nonsense. Contrary to this, the idea of application of the adaptivity to inverse problems is to analyze the Lagrangian, to use this analysis in order to find spots, where the maximum error of the solution is and finally insert more finite elements in those spots, see Theorem 6.1. It is well known that it is hard to reconstruct correct values of the unknown coefficient inside of small inclusions. So, our numerical experiments demonstrate that these values can indeed be reconstructed accurately using the adaptivity technique, and this is one of advantages of this technique.

The inverse problem consists of reconstructing the dielectric permittivity,  $\epsilon(x)$ , from data measured on parts of the surface of the given domain, given the wave input on other parts. By solving the wave equation with the same input, the material variables are in principle obtained by fitting the computed solution to the measured data. The problem

is formulated as finding a stationary point of the Lagrangian, involving the forward wave equation (the state equation), the backward wave equation (the adjoint equation), and an equation expressing that the gradient with respect to the parameter vanishes. The optimum is found by a quasi-Newton iteration solving the forward and backward wave equations and updating the material coefficient for each step. In order to stabilize this ill-posed problem, we use Tikhonov regularization [12, 22]. We present an adaptive algorithm to solve the inverse problem where the space-mesh adaptivity is based only on the computation of the residuals, since they already give us enough information where to adapt the mesh.

It is well known that minimization of the residual least-squares functionals can give multiple local minima or ravines, see Test 5 in [9] and [19]. To treat this problem we employ an adaptive approach, where we first solve the inverse problem on a coarse mesh, then refine the mesh locally and use the results from previous iteration as an initial guess in our optimization algorithm on a new refined mesh. In fact, we believe that the adaptive control method is sort of mitigating the phenomenon of local minima and our numerical experiments confirm this thought.

We present numerical experiments where a periodic structure is reconstructed, showing the possibilities of using adaptive error control in computational inverse scattering. Testing the adaptive algorithm with different initial guess values in the optimization algorithm, we can find the neighborhood of the starting point on a coarse mesh where our adaptive algorithm will converge. Our numerical experiments show that the neighborhood of the initial guess in the adaptive optimization algorithm is bigger than in the usual optimization algorithms. Thus, using adaptivity in the optimization algorithms allow us to mitigate the problem of local minima appearing in the usual optimization algorithms.

## 2 The mathematical model

We shall restrict ourselves to the propagation of light in a mixed dielectric medium in a bounded domain  $\Omega \subset \mathbb{R}^2$  with boundary  $\Gamma$ , governed by Maxwell's equations:

$$\begin{aligned} \frac{\partial D}{\partial t} - \nabla \times H &= -J, & \text{in } \Omega \times (0, T], \\ \frac{\partial B}{\partial t} + \nabla \times E &= 0, & \text{in } \Omega \times (0, T], \\ \nabla \cdot D &= \rho, & \text{in } \Omega \times (0, T], \\ \nabla \cdot B &= 0, & \text{in } \Omega \times (0, T]. \end{aligned} \tag{1}$$

Here  $E(x, t)$  and  $H(x, t)$  are the electric and magnetic fields, whereas  $D(x, t)$  and  $B(x, t)$  are the electric and magnetic inductions, respectively. We assume that the dielectric permittivity,  $\epsilon(x)$ , is scalar and that the material is non-magnetic, so that  $\mu(x) = 1$ . Then  $D = \epsilon E$  and  $B = H$ . The current density,  $J$ , and charge density,  $\rho$ , are both assumed to be zero.

Let us consider the system of Maxwell's equations:

$$\nabla \times H = \epsilon \frac{\partial E}{\partial t}, \tag{2}$$

$$\nabla \times E = -\frac{\partial H}{\partial t}. \tag{3}$$

Assume that all functions in (2), (3) are independent of  $z$ , see [11]. Let  $E$  and  $H$  be vectors with components  $E = (E_1, E_2, E_3)$  and  $H = (H_1, H_2, H_3)$ . Then (2) is equivalent with

$$\mathbf{i}\frac{\partial H_3}{\partial y} - \mathbf{j}\frac{\partial H_3}{\partial x} + \mathbf{k}\left(\frac{\partial H_2}{\partial x} - \frac{\partial H_1}{\partial y}\right) = \mathbf{i}\epsilon\frac{\partial E_1}{\partial t} + \mathbf{j}\epsilon\frac{\partial E_2}{\partial t} + \mathbf{k}\epsilon\frac{\partial E_3}{\partial t}.$$

From this equation we have

$$\epsilon\frac{\partial E_3}{\partial t} = \frac{\partial H_2}{\partial x} - \frac{\partial H_1}{\partial y}. \quad (4)$$

Now, we use (3) taking into account the fact that  $E$  is independent of  $z$ :

$$\mathbf{i}\frac{\partial E_3}{\partial y} - \mathbf{j}\frac{\partial E_3}{\partial x} + \mathbf{k}\left(\frac{\partial E_2}{\partial x} - \frac{\partial E_1}{\partial y}\right) = -\mathbf{i}\frac{\partial H_1}{\partial t} - \mathbf{j}\frac{\partial H_2}{\partial t} - \mathbf{k}\frac{\partial H_3}{\partial t}. \quad (5)$$

Differentiating (4) with respect to  $t$ , we get

$$\epsilon\frac{\partial^2 E_3}{\partial t^2} = \frac{\partial^2 H_2}{\partial x \partial t} - \frac{\partial^2 H_1}{\partial y \partial t}. \quad (6)$$

Taking into account (5), we can write

$$\frac{\partial H_2}{\partial t} = \frac{\partial E_3}{\partial x}, \quad (7)$$

$$\frac{\partial H_1}{\partial t} = -\frac{\partial E_3}{\partial y}. \quad (8)$$

Substituting both expressions above into (5), we get

$$\epsilon\frac{\partial^2 E_3}{\partial t^2} = \Delta E_3. \quad (9)$$

Equations (7), (8), (9) contains  $H_1$ ,  $H_2$  and  $E_3$  variables and are called the transverse electric (TE) polarization [11]. In the rest of the paper we will use the notation  $E$  instead of  $E_3$ .

Split  $\Gamma$  into three disjoint parts,  $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3$ , and consider the forward problem consisting of (9) and the following initial and boundary conditions (here and below, we denote  $Dv = \frac{\partial v}{\partial t}$ )

$$\begin{aligned} E(\cdot, 0) &= 0, \quad \frac{\partial E}{\partial t}(\cdot, 0) = 0, \quad \text{in } \Omega, \\ \partial_n E|_{\Gamma_1} &= v_1, \quad \text{on } \Gamma_1 \times (0, t_1], \\ \partial_n E|_{\Gamma_1} &= DE, \quad \text{on } \Gamma_1 \times (t_1, T], \\ \partial_n E|_{\Gamma_2} &= DE, \quad \text{on } \Gamma_2 \times (0, T], \\ \partial_n E|_{\Gamma_3} &= 0, \quad \text{on } \Gamma_3 \times (0, T]. \end{aligned} \quad (10)$$

Thus,  $v_1$  is a pulse emitting from  $\Gamma_1$  and propagating into  $\Omega$  for  $t \in [0, t_1]$ . First order absorbing boundary conditions [13] are used on  $\Gamma_1 \times (t_1, T]$  and  $\Gamma_2 \times (0, T]$ , and Neumann boundary conditions on  $\Gamma_3$ .

### 3 A hybrid finite element/difference method

To solve equation (9)-(10) we use a hybrid FEM/FDM method developed in [10]. The method uses continuous, piecewise linear finite elements in space and time on a partially structured mesh in space. The computational space domain  $\Omega$  is decomposed into a finite element domain  $\Omega_{\text{FEM}}$ , with an unstructured mesh, and a finite difference domain  $\Omega_{\text{FDM}}$ , with a structured mesh. Typically,  $\Omega_{\text{FEM}}$  covers only a small part of  $\Omega$ . In  $\Omega_{\text{FDM}}$  we use quadrilateral elements in  $\mathbb{R}^2$  and hexahedra in  $\mathbb{R}^3$ . In  $\Omega_{\text{FEM}}$  we use a finite element mesh  $K_h = \{K\}$  with elements  $K$  consisting of triangles in  $\mathbb{R}^2$  and tetrahedra in  $\mathbb{R}^3$ . Let us associate with  $K_h$  a mesh function  $h_K(x) = \text{diam}(K)$ ,  $\forall x \in K$ , representing the diameter of the element  $K$ . For the time discretization, let  $J_k = \{J\}$  be a partition of the time interval  $I = (0, T]$  into time intervals  $J = (t_{k-1}, t_k]$  of uniform length  $\tau = t_k - t_{k-1}$ .

We define the following  $L_2$  inner products and norm

$$(p, q) = \int_{\Omega} pq \, dx, \quad ((p, q)) = \int_{\Omega} \int_0^T pq \, dt \, dx, \quad \|p\|^2 = ((p, p)).$$

We introduce the finite element trial space  $W_h^v$  defined by :

$$W_h^v := \{v \in W_1^v \cup W_2^v : v|_{K \times J} \in P_1(K) \times P_1(J), \forall K \in K_h, \forall J \in J_k\},$$

where

$$\begin{aligned} W_1^v &:= \{v \in H^1(\Omega \times J) : v(\cdot, 0) = 0, \partial_n v|_{\Gamma_1} = v_1, \partial_n v|_{\Gamma_2} = Dv, \partial_n v|_{\Gamma_3} = 0\}, \\ W_2^v &:= \{v \in H^1(\Omega \times J) : v(\cdot, 0) = 0, \partial_n v|_{\Gamma_1} = \partial_n v|_{\Gamma_2} = Dv, \partial_n v|_{\Gamma_3} = 0\}. \end{aligned}$$

Here  $P_1(K)$  and  $P_1(J)$  are the set of linear functions on  $K$  and  $J$ , respectively.

Furthermore, the finite element space  $W_h^\lambda$  for the costate  $\lambda$ , is defined by:

$$W_h^\lambda := \{\lambda \in W_1^\lambda \cup W_2^\lambda : \lambda|_{K \times J} \in P_1(K) \times P_1(J), \forall K \in K_h, \forall J \in J_k\},$$

where

$$\begin{aligned} W_1^\lambda &:= \{\lambda \in H^1(\Omega \times J) : \lambda(\cdot, T) = 0, \partial_n \lambda|_{\Gamma_1} = \partial_n \lambda|_{\Gamma_3} = 0, \partial_n \lambda|_{\Gamma_2} = D\lambda\}, \\ W_2^\lambda &:= \{\lambda \in H^1(\Omega \times J) : \lambda(\cdot, T) = 0, \partial_n \lambda|_{\Gamma_1} = \partial_n \lambda|_{\Gamma_2} = D\lambda, \partial_n \lambda|_{\Gamma_3} = 0\}. \end{aligned}$$

The finite element method for (9)-(10) now reads: Find  $E_h \in W_h^v$  such that  $\forall \bar{\lambda} \in W_h^\lambda$ ,

$$\begin{aligned} & - ((\epsilon DE_h, D\bar{\lambda})) + ((\nabla E_h, \nabla \bar{\lambda})) \\ & - ((DE_h, \bar{\lambda}))_{(t_1, T] \times \Gamma_1} - ((DE_h, \bar{\lambda}))_{(0, T] \times \Gamma_2} = ((v_1, \bar{\lambda}))_{(0, t_1] \times \Gamma_1}. \end{aligned} \tag{11}$$

Here, the initial condition  $DE(0) = 0$  is imposed in weak form through the variational formulation.

Expanding  $E$  and  $\lambda$  in terms of the standard continuous piecewise linear functions  $\varphi_i(x)$  in space and  $\psi_i(t)$  in time, and substituting this into (11), we obtain an explicit scheme for solving (11), see for example [4] where a similar system is obtained for an acoustic wave equation with homogeneous boundary conditions.

### 4 The inverse problem

We formulate the inverse problem for (9) and (10) as follows: given the function  $\partial_n E = v_1$  on  $\Gamma_1 \times (0, t_1]$ , determine the coefficient  $\epsilon(x)$  for  $x \in \Omega$ , which minimizes the quantity

$$J(E, \epsilon) = \frac{1}{2} \int_0^T \int_{\Omega} (E - \tilde{E})^2 \delta_{obs} \, dxdt + \frac{1}{2} \gamma \int_{\Omega} (\epsilon - \epsilon_0)^2 \, dx. \tag{12}$$

Here  $\tilde{E}$  is the data observed at a finite set of points  $x_{obs}$ ,  $E$  satisfies (9) and (10) and thus depends on  $\epsilon$ . Moreover,  $\delta_{obs} = \sum \delta(x - x_{obs})$  is a sum of delta-functions corresponding to the observation points,  $\gamma$  is a regularization parameter, and  $\epsilon_0$  is the initial guess value for the parameter we want to reconstruct.

To solve this minimization problem, we introduce the Lagrangian

$$L(u) = J(E, \epsilon) - ((\epsilon DE, D\lambda)) + ((\nabla E, \nabla \lambda)) - ((DE, \lambda))_{(t_1, T] \times \Gamma_1} - ((DE, \lambda))_{(0, T] \times \Gamma_2} - ((v_1, \lambda))_{(0, t_1] \times \Gamma_1}, \quad (13)$$

where  $u = (E, \lambda, \epsilon)$ , and search for a stationary point with respect to  $u$ , satisfying for all  $\bar{u} = (\bar{E}, \bar{\lambda}, \bar{\epsilon})$

$$L'(u; \bar{u}) = 0, \quad (14)$$

where  $L'$  is the gradient of  $L$ . Equation (14) expresses that for all  $\bar{u}$ ,

$$\begin{aligned} \frac{\partial L(u)}{\partial \lambda}(\bar{\lambda}) &= -((\epsilon D\bar{\lambda}, DE)) + ((\nabla E, \nabla \bar{\lambda})) - ((DE, \bar{\lambda}))_{(t_1, T] \times \Gamma_1} \\ &\quad - ((DE, \bar{\lambda}))_{(0, T] \times \Gamma_2} - ((2v_1, \bar{\lambda}))_{(0, t_1] \times \Gamma_1} = 0, \\ \frac{\partial L(u)}{\partial E}(\bar{E}) &= ((E - \tilde{E}, \bar{E}))_{\delta_{obs}} - ((\epsilon D\lambda, D\bar{E})) + ((\nabla \lambda, \nabla \bar{E})) \\ &\quad + ((D\lambda, \bar{E}))_{[0, T] \times \Gamma_1} + ((D\lambda, \bar{E}))_{[0, T] \times \Gamma_2} = 0, \\ \frac{\partial L(u)}{\partial \epsilon}(\bar{\epsilon}) &= -((D\lambda DE, \bar{\epsilon})) + \gamma(\epsilon - \epsilon_0, \bar{\epsilon}) = 0. \end{aligned}$$

The first equation in (15) is a weak form of the state equation (9) and (10), the second equation is a weak form of the adjoint state equation,

$$\begin{aligned} \epsilon \frac{\partial^2 \lambda}{\partial t^2} - \nabla \cdot (\nabla \lambda) &= -(E - \tilde{E})\delta_{obs}, \quad x \in \Omega, \quad 0 \leq t < T, \\ \partial_n \lambda &= 0 \text{ on } \Gamma_1 \times [0, t_1), \\ \partial_n \lambda &= D\lambda \text{ on } \Gamma_1 \times [0, T), \\ \partial_n \lambda &= D\lambda \text{ on } \Gamma_2 \times [0, T), \\ \partial_n \lambda &= 0 \text{ on } \Gamma_3 \times [0, T), \\ \lambda(\cdot, T) &= D\lambda(\cdot, T) = 0 \text{ in } \Omega, \end{aligned} \quad (15)$$

and the last equation expresses stationarity with respect to the parameter  $\epsilon$ .

## 5 A finite element method for inverse problem

To formulate a finite element method for (14) we introduce the finite element space  $V_h$  of piecewise constants for the coefficient  $\epsilon(x)$ , defined by :

$$V_h := \{v \in L_2(\Omega) : v \in P_0(K), \forall K \in K_h\}.$$

Recalling the definitions of  $W_h^v$  and  $W_h^\lambda$ , related to the state  $E$  and the costate  $\lambda$ , and defining  $U_h = W_h^v \times W_h^\lambda \times V_h$ , we formulate the finite element method for (14) as: Find  $u_h \in U_h$ , such that

$$L'(u_h; \bar{u}) = 0, \quad \forall \bar{u} \in U_h. \quad (16)$$

## 6 An a posteriori error estimate for the Lagrangian

**Theorem 6.1.** *Let  $L(u) = L(E, \lambda, \epsilon)$  be the Lagrangian as defined in (13), and let  $L(u_h) = L(E_h, \lambda_h, \epsilon_h)$  be the approximation of  $L(u)$ . Then the following representation holds for the error  $e = L(u) - L(u_h)$ :*

$$\begin{aligned}
|e| \leq & ((R_{E_1}, \sigma_\lambda))_{(0, t_1] \times \Gamma_1} + ((R_{E_2}, \sigma_\lambda)) + ((R_{E_3}, \sigma_\lambda)) \\
& + ((R_{E_4}, \sigma_\lambda))_{(t_1, T] \times \Gamma_1} + ((R_{E_5}, \sigma_\lambda))_{(0, T] \times \Gamma_2} \\
& + ((R_{\lambda_1}, \sigma_E))_{\delta_{obs}} + ((R_{\lambda_2}, \sigma_E)) + ((R_{\lambda_3}, \sigma_E)) \\
& + ((R_{\lambda_4}, \sigma_E))_{(0, T] \times \Gamma_1} + ((R_{\lambda_5}, \sigma_E))_{(0, T] \times \Gamma_2} \\
& + ((R_{\epsilon_1}, \sigma_\epsilon)) + (R_{\epsilon_2}, \sigma_\epsilon),
\end{aligned} \tag{17}$$

where the residuals are defined by

$$\begin{aligned}
R_{E_1} &= 2|v_1|, & R_{E_2} &= \max_{S \subset \partial K} h_K^{-1} |[\partial_s E_h]|, \\
R_{E_3} &= \epsilon_h \tau^{-1} |[\partial_t E_h]|, & R_{E_4} &= R_{E_5} = |DE_h|, \\
R_{\lambda_1} &= |E_h - \tilde{E}|, & R_{\lambda_2} &= \max_{S \subset \partial K} h_K^{-1} |[\partial_s \lambda_h]|, \\
R_{\lambda_3} &= \epsilon_h \tau^{-1} |[\partial_t \lambda_h]|, & R_{\lambda_4} &= R_{\lambda_5} = |D\lambda_h|, \\
R_{\epsilon_1} &= |D\lambda_h| \cdot |DE_h|, & R_{\epsilon_2} &= \gamma |\epsilon_h - \epsilon_0|,
\end{aligned}$$

and the interpolation errors are

$$\begin{aligned}
\sigma_\lambda &= C\tau | [D\lambda_h] | + Ch_K | [\partial_n \lambda_h] |, \\
\sigma_E &= C\tau | [DE_h] | + Ch_K | [\partial_n E_h] |, \\
\sigma_\epsilon &= C | [\epsilon_h] |,
\end{aligned}$$

Here,  $[v]$  denotes the maximum of the modulus of a jump of  $v$  across the face of an element  $K$  (or the boundary node of a time interval  $J$ ),  $\partial_s v$  denotes the normal derivative of  $v$  across a side of  $K$ ,  $\partial_n v$  denotes the derivative of  $v$  in the outward normal of an element  $K$ ,  $[\partial_t v]$  is the maximum modulus of the jump of the time derivative of  $v$  across a boundary node of  $J$ ,  $C$  is interpolation constants of moderate size.

*Proof.* Throughout the proof, let  $C$  denote different constants of a moderate size.

As in [1], we use the fundamental theorem of calculus to write

$$\begin{aligned}
e &= L(v) - L(v_h) \\
&= \int_0^1 \frac{d}{ds} L(v_h + s(v - v_h)) ds \\
&= \int_0^1 L'(v_h + s(v - v_h); v - v_h) ds \\
&= L'(v_h; v - v_h) + R,
\end{aligned}$$

where  $R$  denotes a second order term. For full details of the arguments we refer to [1] and [15].

Neglecting the term  $R$ , and using the Galerkin orthogonality (16) with the splitting

$$v - v_h = (v - v_h^I) + (v_h^I - v_h), \tag{18}$$

where  $v_h^I$  denotes an interpolant of  $v$ , leads to the following error representation:

$$e \approx L'(v_h; v - v_h^I) = I_1 + I_2 + I_3. \tag{19}$$

Here

$$\begin{aligned}
I_1 &= -((\epsilon_h DE_h, D(\lambda - \lambda_h^I))) + ((\nabla E_h, \nabla(\lambda - \lambda_h^I))) - ((2v_1, \lambda - \lambda_h^I))_{(0,t_1] \times \Gamma_1} \\
&\quad - ((DE_h, \lambda - \lambda_h^I))_{(t_1, T] \times \Gamma_1} - ((DE_h, \lambda - \lambda_h^I))_{(0, T] \times \Gamma_2}, \\
I_2 &= ((E_h - \tilde{E}, E - E_h^I)_{\delta_{obs}} - ((\epsilon_h D\lambda_h, D(E - E_h^I))) + ((\nabla \lambda_h, \nabla(E - E_h^I))) \\
&\quad + ((D\lambda_h, E - E_h^I))_{(0, T] \times \Gamma_1} + ((D\lambda_h, E - E_h^I))_{(0, T] \times \Gamma_2}, \\
I_3 &= -((D\lambda_h DE_h, \epsilon - \epsilon_h^I)) + \gamma(\epsilon_h - \epsilon_0, \epsilon - \epsilon_h^I),
\end{aligned}$$

To estimate  $I_1$ , we integrate by parts in the first and second terms to obtain:

$$\begin{aligned}
|I_1| &= |((\epsilon_h D^2 E_h, \lambda - \lambda_h^I)) - ((\Delta E_h, \lambda - \lambda_h^I)) \\
&\quad - ((2v_1, \lambda - \lambda_h^I))_{(0,t_1] \times \Gamma_1} - ((DE_h, \lambda - \lambda_h^I))_{(t_1, T] \times \Gamma_1} \\
&\quad - ((DE_h, \lambda - \lambda_h^I))_{(0, T] \times \Gamma_2} \\
&\quad - \sum_k \int_{\Omega} \epsilon_h [DE_h(t_k)] (\lambda - \lambda_h^I)(t_k) dx \\
&\quad + \sum_K \int_0^T \int_{\partial K} \partial_n E_h (\lambda - \lambda_h^I) ds dt|, \tag{20}
\end{aligned}$$

Here,  $[DE_h(t_k)]$  denote the jump of the derivative of  $E_h$  at time  $t_k$  (see Figure 2), and  $\partial_n E_h$  denote the derivative of  $E_h$  in the outward normal direction  $n$  of the boundary  $\partial K$  of element  $K$ .

Since  $E_h$  is a piecewise linear function, the terms  $D^2 E_h$  and  $\Delta E_h$  in (20) disappear, and we get:

$$\begin{aligned}
|I_1| &= | - ((2v_1, \lambda - \lambda_h^I))_{(0,t_1] \times \Gamma_1} - ((DE_h, \lambda - \lambda_h^I))_{(t_1, T] \times \Gamma_1} \\
&\quad - ((DE_h, \lambda - \lambda_h^I))_{(0, T] \times \Gamma_2} \\
&\quad - \sum_k \int_{\Omega} \epsilon_h [DE_h(t_k)] (\lambda - \lambda_h^I)(t_k) dx \\
&\quad + \sum_K \int_0^T \int_{\partial K} \partial_n E_h (\lambda - \lambda_h^I) ds dt|. \tag{21}
\end{aligned}$$

In the last term of equation (21) we sum over the element boundaries, where each interior side  $S \in S_h$  occurs twice, see Figure 1. Denoting by  $\partial_{n^\pm} E_h$  the derivative of  $E_h$  in the outward normal direction  $n^\pm$  to element  $K^\pm$ , and by  $\partial_s E_h$  the derivative of a function  $E_h$  in one of the normal directions,  $n^-$  and  $n^+$ , of each side  $S$ , we can write

$$\sum_K \int_{\partial K} \partial_n E_h (\lambda - \lambda_h^I) ds = \sum_S \int_S [\partial_s E_h] (\lambda - \lambda_h^I) ds, \tag{22}$$

where the jump  $[\partial_s E_h]$  is defined as

$$[\partial_s E_h] = \max_{S \in \partial K} \{\partial_{n^+} E_h, \partial_{n^-} E_h\}.$$

We distribute each jump equally to the two sharing elements and return to a sum of the element edges  $\partial K$  :

$$\sum_S \int_S [\partial_s E_h] (\lambda - \lambda_h^I) ds = \sum_K \frac{1}{2} \int_{\partial K} [\partial_s E_h] (\lambda - \lambda_h^I) ds. \tag{23}$$

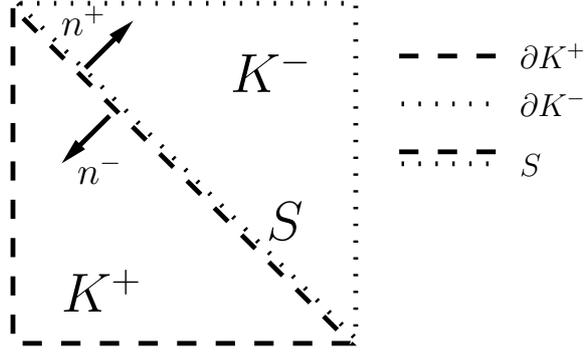


Figure 1: Two neighboring elements  $K^+$  and  $K^-$ , their boundaries,  $\partial K^+$  and  $\partial K^-$ , and the interior side  $S$ .

We multiply and divide by  $h_K$ , formally set  $dx = h_K ds$  and replace the integrals over the element boundaries  $\partial K$  by integrals over the elements  $K$ , to get:

$$\begin{aligned} & \left| \sum_K \frac{1}{2} h_K^{-1} \int_{\partial K} [\partial_s E_h] (\lambda - \lambda_h^I) h_K ds \right| \\ & \leq C \int_{\Omega} \max_{S \subset \partial K} h_K^{-1} |[\partial_s E_h]| |\lambda - \lambda_h^I| dx, \end{aligned} \quad (24)$$

where  $[\partial_s E_h] \Big|_K = \max_{S \subset \partial K} |[\partial_s E_h]|_S$ .

In a similar way we can estimate the jump in time in (21) by multiplying and dividing by  $\tau$ :

$$\begin{aligned} & \left| \sum_k \int_{\Omega} \epsilon_h [DE_h(t_k)] (\lambda - \lambda_h^I)(t_k) dx \right| \\ & \leq \sum_k \int_{\Omega} \epsilon_h \tau^{-1} | [DE_h(t_k)] | |(\lambda - \lambda_h^I)(t_k)| \tau dx \\ & \leq C \sum_k \int_{J_k} \int_{\Omega} \epsilon_h \tau^{-1} | [\partial_{t_k} E_h] | |\lambda - \lambda_h^I| dx dt \\ & = C ( (\epsilon_h \tau^{-1} | [\partial_t E_h] |, |(\lambda - \lambda_h^I)| ) ). \end{aligned} \quad (25)$$

Here, we have defined  $[\partial_{t_k} E_h]$  as the greatest of the two jumps on the interval  $(t_k, t_{k+1}]$ :

$$[\partial_{t_k} E_h] = \max_k ( [DE_h(t_k)], [DE_h(t_{k+1})] ),$$

$$[\partial_t E_h] = [\partial_{t_k} E_h] \text{ on } J_k.$$

where  $[DE_h(t_k)] = DE_h^+(t_k) - DE_h^-(t_k)$ . The time jumps are illustrated in Figure 2.

We substitute the expressions (24) and (25) in (21), to get:

$$\begin{aligned} |I_1| \leq & ((2|v_1|, |\lambda - \lambda_h^I|))_{(0, t_1] \times \Gamma_1} - ((|DE_h|, |\lambda - \lambda_h^I|))_{(t_1, T] \times \Gamma_1} \\ & - ((|DE_h|, |\lambda - \lambda_h^I|))_{(0, T] \times \Gamma_2} \\ & + C ( (\max_{S \subset \partial K} h_K^{-1} |[\partial_s E_h]|, |\lambda - \lambda_h^I|) ) \\ & + C ( (\epsilon_h \tau^{-1} |[\partial_t E_h]|, |\lambda - \lambda_h^I|) ). \end{aligned}$$

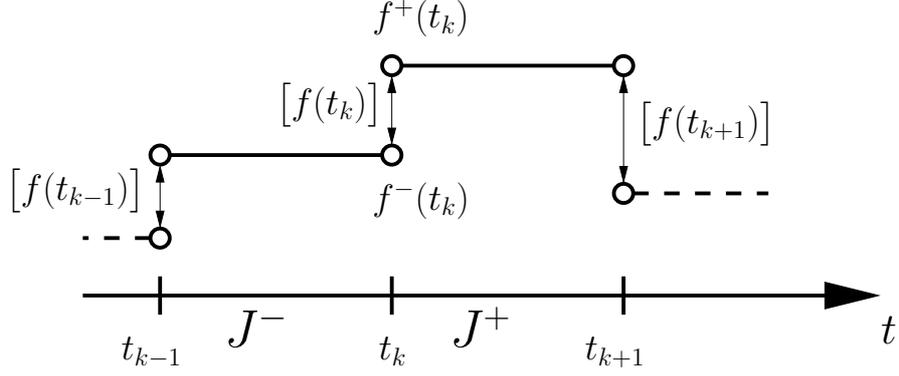


Figure 2: The jump of a function  $f$  on the time mesh.

Next, we use the following standard interpolation estimate

$$|\lambda - \lambda_h^I| \leq C(\tau^2 |D^2\lambda| + h_K^2 |D_x^2\lambda|), \quad (26)$$

where we approximate the second derivative in time as

$$D^2\lambda = \frac{\partial^2\lambda}{\partial t^2} = \frac{\partial(D\lambda)}{\partial t} \approx \frac{(D\lambda)^+ - (D\lambda)^-}{\tau} = \frac{[D\lambda_h]}{\tau}.$$

Here  $(\cdot)^+$  and  $(\cdot)^-$  represents values on two neighboring intervals  $J^+$  and  $J^-$ , see Figure 2. In the same way we approximate the second derivative in space:

$$D_x^2\lambda \approx \frac{[\partial_n \lambda_h]}{h}.$$

Substituting both expressions above in (26), we obtain

$$|\lambda - \lambda_h^I| \leq C(\tau |[D\lambda_h]| + h_K |[\partial_n \lambda_h]|) \quad (27)$$

and the estimate for  $I_1$  reduces to

$$\begin{aligned} |I_1| \leq & C((2|v_1|, \tau |[D\lambda_h]| + h_K |[\partial_n \lambda_h]|))_{(0, t_1] \times \Gamma_1} \\ & - C((|DE_h|, \tau |[D\lambda_h]| + h_K |[\partial_n \lambda_h]|))_{(t_1, T] \times \Gamma_1} \\ & - C((|DE_h|, \tau |[D\lambda_h]| + h_K |[\partial_n \lambda_h]|))_{(0, T] \times \Gamma_2} \\ & + C((\max_{S \subset \partial K} h_k^{-1} |[\partial_s E_h]|, \tau |[D\lambda_h]| + h_K |[\partial_n \lambda_h]|)) \\ & + C((\epsilon_h \tau^{-1} |[\partial_t E_h]|, \tau |[D\lambda_h]| + h_K |[\partial_n \lambda_h]|)). \end{aligned}$$

We estimate  $I_2$  similarly as  $I_1$ . First, we integrate by parts to obtain

$$\begin{aligned} |I_2| \leq & |((E_h - \tilde{E}, E - E_h^I))_{\delta_{obs}} + ((\epsilon_h D^2\lambda_h, E - E_h^I)) \\ & - ((\Delta\lambda_h, E - E_h^I)) + ((D\lambda_h, E - E_h^I))_{(0, T] \times \Gamma_1} \\ & + ((D\lambda_h, E - E_h^I))_{(0, T] \times \Gamma_2} \\ & + C((\max_{S \subset \partial K} h_K^{-1} |[\partial_s \lambda_h]|, |E - E_h^I|)) \\ & + C((\epsilon_h \tau^{-1} |[\partial_t \lambda_h]|, |E - E_h^I|)). \end{aligned}$$

Since  $\lambda_h$  is piecewise linear, the terms with  $\Delta\lambda_h$  and  $D^2\lambda_h$  will disappear:

$$\begin{aligned}
|I_2| \leq & \left( (|E_h - \tilde{E}|, |E - E_h^I|) \right)_{\delta_{obs}} \\
& + \left( (|D\lambda_h|, |E - E_h^I|) \right)_{(0,T] \times \Gamma_1} \\
& + \left( (|D\lambda_h|, |E - E_h^I|) \right)_{(0,T] \times \Gamma_2} \\
& + C \left( (\max_{S \subset \partial K} h_K^{-1} |\partial_s \lambda_h|, |E - E_h^I|) \right) \\
& + C \left( (\epsilon_h \tau^{-1} |\partial_t \lambda_h|, |E - E_h^I|) \right).
\end{aligned}$$

Next, we use the same kind of interpolation estimate for  $|E - E_h^I|$  as we found for  $|\lambda - \lambda_h^I|$  in equation (27), to get:

$$\begin{aligned}
|I_2| \leq & C \left( (|E_h - \tilde{E}|, \tau |DE_h| + h_K |\partial_n E_h|) \right)_{\delta_{obs}} \\
& + C \left( (|D\lambda_h|, \tau |DE_h| + h_K |\partial_n E_h|) \right)_{(0,T] \times \Gamma_1} \\
& + C \left( (|D\lambda_h|, \tau |DE_h| + h_K |\partial_n E_h|) \right)_{(0,T] \times \Gamma_2} \\
& + C \left( (\max_{S \subset \partial K} h_K^{-1} |\partial_s \lambda_h|, \tau |DE_h| + h_K |\partial_n E_h|) \right) \\
& + C \left( (\epsilon_h \tau^{-1} |\partial_t \lambda_h|, \tau |DE_h| + h_K |\partial_n E_h|) \right).
\end{aligned}$$

To estimate  $I_3$  we use the following approximation estimate for  $\epsilon - \epsilon_h^I$ :

$$|\epsilon - \epsilon_h^I| \leq Ch_K D_x \epsilon \leq Ch_K \left| \frac{[\epsilon_h]}{h_K} \right| \leq C |[\epsilon_h]|,$$

and we end up with

$$|I_3| \leq \left( (|D\lambda_h| |DE_h|, |[\epsilon_h]|) \right) + \gamma (|\epsilon_h - \epsilon_0|, |[\epsilon_h]|),$$

which completes the proof. □

## 7 An adaptive algorithm for solution of the inverse problem

In this section we present an adaptive algorithm for solution of the inverse problem defined in Section 4. In this algorithm the refinement is based on computations of the residuals for the parameter, since they already give a good indication where to refine the mesh. The interpolation errors, and thus the exact value of the computational error in the reconstructed parameter, can be obtained by computing the Hessian of the Lagrangian [8].

As we see from (17), the error in the Lagrangian consists of integrals in space and time of the different residuals multiplied by the interpolation errors. Thus, to estimate the error in the Lagrangian we need to compute the approximated values of  $(E_h, \lambda_h, \epsilon_h)$  together with residuals and interpolation errors. Since we want to control the error in the reconstructed parameter,  $\epsilon_h$ , we limit the computations to  $R_{\epsilon_1}$  and  $R_{\epsilon_2}$ , and neglect to compute the other residuals in the a posteriori estimate (17). Thus, the a posteriori error is calculated as

$$e(x) \approx \int_0^T R_{\alpha_1}(x, t) dt + R_{\alpha_2}(x). \quad (28)$$

### Algorithm

0. Choose an initial mesh  $K_h$  and an initial time partition  $J_0$  of the time interval  $(0, T]$ . Start with an initial guess  $\epsilon^0$ , and compute the sequence of  $\epsilon^n$  in the following steps:
1. Compute the solution  $E^n$  of the forward problem (9)-(10) on  $K_h$  and  $J_k$ , with  $\epsilon(x) = \epsilon^{(n)}$ .
2. Compute the solution  $\lambda^n$  of the adjoint problem (15) on  $K_h$  and  $J_k$ .
3. Update the parameter  $\epsilon$  on  $K_h$  and  $J_k$  using the quasi-Newton method

$$\epsilon^{n+1} = \epsilon^n + \alpha^n H^n g^n,$$

where  $H^n$  is an approximate Hessian, computed using the usual BFGS update formula for the Hessian, see [20]. Furthermore,  $g^n$  is the gradient of the Lagrangian (13) with respect to the parameter  $\epsilon$ ,

$$g^n = - \int_0^T D\lambda^n DE^n dt + \gamma(\epsilon^n - \epsilon_0), \quad (29)$$

where  $\alpha$  is the step length in the parameter upgrade computed using an one-dimensional search algorithm [21].

4. Stop computing  $\epsilon$  if the gradient  $g^n < \eta$ ; if not set  $n = n + 1$  and go to step 1. Here,  $\eta$  is the tolerance in the quasi-Newton update.
5. Compute the residuals,  $R_{\epsilon_1}, R_{\epsilon_2}$  and refine the mesh in all points where

$$\int_0^T R_{\alpha_1}(x, t) dt + R_{\alpha_2}(x) < tol \quad (30)$$

is violated. Here  $tol$  is a tolerance chosen by the user.

6. Construct a new mesh  $K_h$  and a new time partition  $J_k$ . Return to step 1 and perform all the steps of the optimization algorithm on the new mesh.

## 8 Numerical Results

In this section we present several numerical examples to show the performance of the adaptive hybrid method and the usefulness of the a posteriori error estimate (17). To solve the forward and adjoint problems, we use the hybrid FEM/FDM method described in [10].

The computational domain, is set as  $\Omega = [-4.0, 4.0] \times [-5.0, 5.0]$ . Next,  $\Omega$  is split into a finite element domain  $\Omega_{FEM} = [-3.0, 3.0] \times [-3.0, 3.0]$  with an unstructured mesh and a surrounding domain  $\Omega_{FDM}$  with a structured mesh, see Figure 3. Between  $\Omega_{FEM}$  and  $\Omega_{FDM}$  there is an overlapping layer consisting of structured elements. The space mesh consists of triangles in  $\Omega_{FEM}$ , and squares in  $\Omega_{FDM}$  with mesh size  $h = 0.125$  in the overlapping regions. At the top and bottom boundaries of  $\Omega$  we use first-order absorbing boundary conditions [13]. At the lateral boundaries, Neumann boundary conditions allow us to assume an infinite space-periodic structure in the lateral direction.

For simplicity, we assume that  $\epsilon = 1$  in  $\Omega_{FDM}$ . Thus, we only need to reconstruct the electric permittivity  $\epsilon$  in  $\Omega_{FEM}$ .

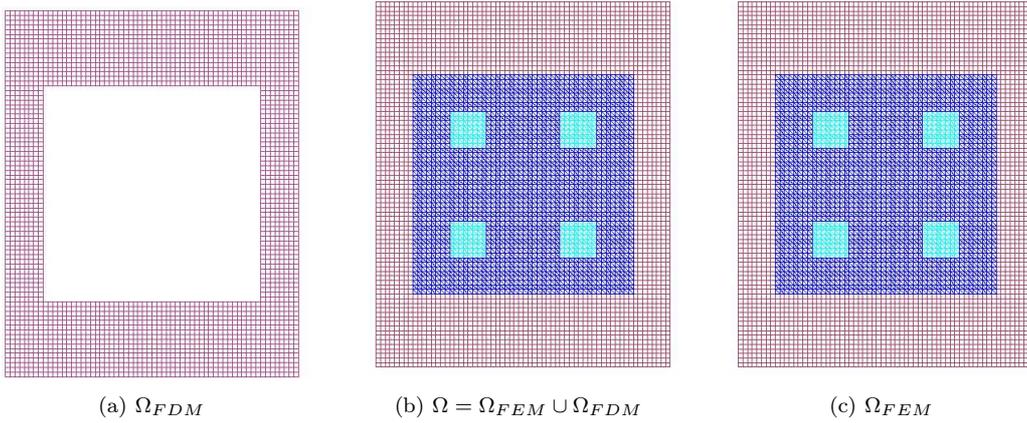


Figure 3: The hybrid mesh (b) is a combinations of a structured mesh (a), where FDM is applied, and a mesh (c), where we use FEM, with a thin overlapping of structured elements.

### 8.1 Example 1

We start to test our adaptive algorithm on the reconstruction of the periodic structure given in Figure 3-c).

To generate data at the observation points, we solve the forward problem (9)-(10) in the domain  $\Omega$  with a plane wave pulse given as

$$\partial_n E|_{\Gamma_1} = ((\sin(\omega t - \pi/2) + 1)/10), \quad 0 \leq t \leq \frac{2\pi}{\omega} = t_1. \quad (31)$$

The field initiates at the boundary  $\Gamma_1$ , in our examples this boundary represents the top boundary of the computational domain, and propagates in normal direction  $n$  into  $\Omega$  with  $\omega = 6$ . The trace of the forward problem is measured at the observation points, placed on the lower boundary of the computational domain  $\Omega_{FEM}$ . On  $\Gamma_1 \times (t_1, T]$  and  $\Gamma_2 \times (0, T]$  we use first order absorbing boundary conditions, [13]. Here,  $T = 12.0$  and the exact value of the parameter is  $\epsilon = 4.0$  inside the square lattices and  $\epsilon = 1.0$  everywhere else. Since an explicit method [4], is used to solve the forward and adjoint problems, we choose a time step  $\tau$  according to the Courant-Friedrich's-Levy (CFL) stability condition to provide a stable time discretization.

We start our adaptive algorithm with initial guess for the parameter being  $\epsilon = 1.0$  at all points in the computational domain  $\Omega_{FEM}$  and with regularization parameter  $\gamma = 0.1$ . We enforce that the parameter  $\epsilon(x)$  belongs to the set of admissible parameters  $C_M = \{\epsilon \in C(\Omega) | 1.0 < \epsilon(x) < 4.0\}$  as follows: if  $0 < \epsilon(x_0) < 1.0$  for some point  $x_0 \in \Omega$  then we set  $\epsilon(x_0) = 1.0$ .

To achieve better results in the reconstruction, we performed tests letting the incoming wave from the top boundary of  $\Omega_{FDM}$  be equal to the reflected non-plane wave measured on the lower boundary of  $\Omega_{FDM}$ . Thus, to generate data at the observation points, we first solve the forward problem (9)-(10) with a plane wave (31) in the time interval  $t = (0, T]$  with the exact value of the parameter being  $\epsilon = 4.0$  inside the square lattices and  $\epsilon = 1.0$  everywhere else, and registered the values of the solution of the forward problem at the lower boundary of  $\Omega_{FDM}$ . Then, using these registered values, a non-plane wave is initialized, starting at  $t = T$  and ending at  $t = 2T$ . In *Test 1* we describe computational results when the observation points are placed only at the top boundary of  $\Omega_{FEM}$ . In *Test 2* the computational tests was performed when the observation points are placed both on the lower and upper boundaries of  $\Omega_{FEM}$ .

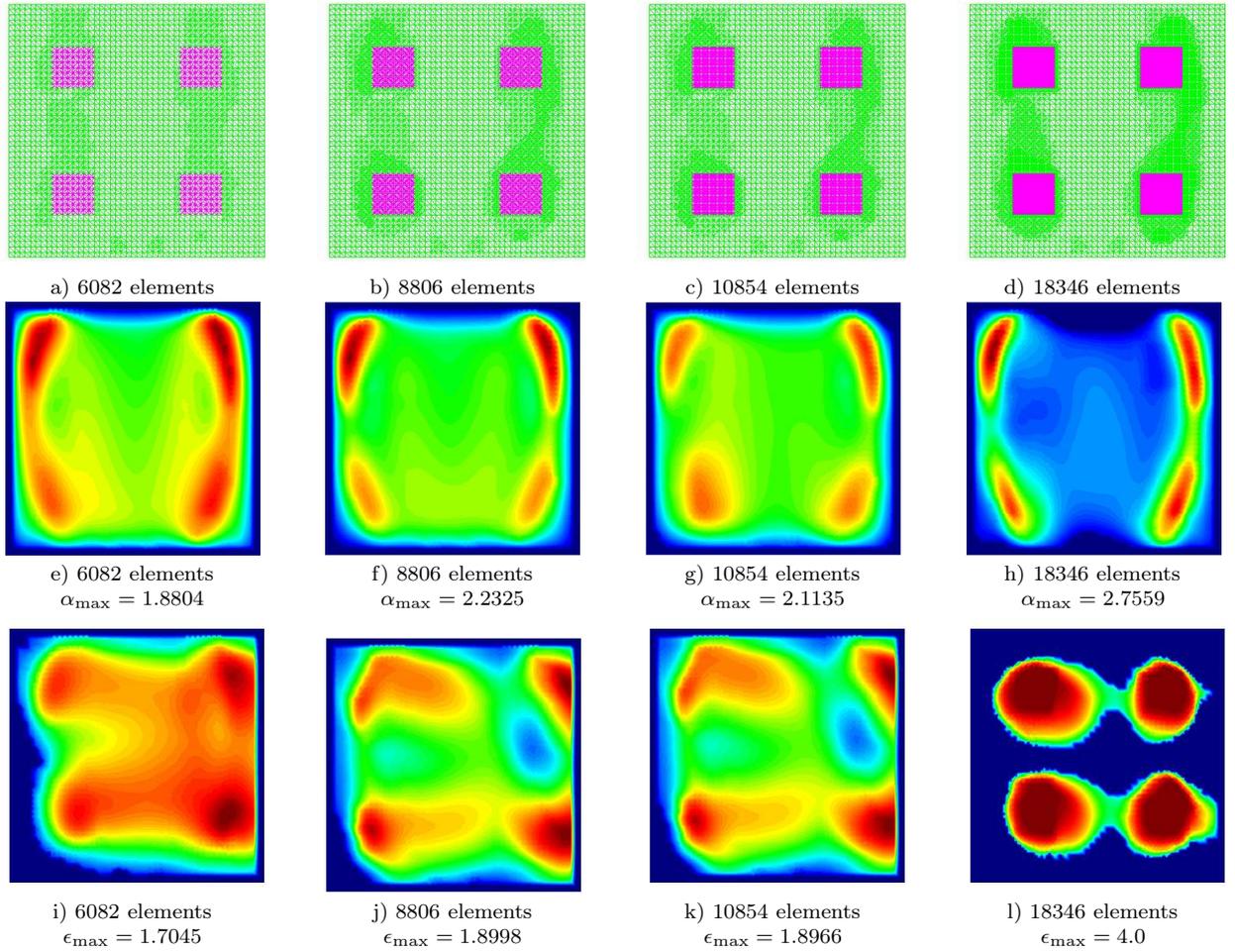


Figure 4: a)-d) Adaptively refined meshes ; Reconstructed parameter  $\epsilon(x)$ , indicating domains with a given parameter value: Test1 - Figures e)-h); Test 2 - Figures i)-l). Here, red color corresponds to the maximum parameter value on the corresponding meshes, and blue color - to the minimum.

We performed the same tests as above, but with adding relative noise to the observed data. The data with relative disturbance, or noise,  $E_\sigma$ , is computed by adding a relative error to the computed data  $E_{obs}$  using the expression

$$E_\sigma = E_{obs} + \alpha(E_{max} - E_{min})\sigma/100. \quad (32)$$

Here,  $\alpha$  is a random number on the interval  $[-1; 1]$ ,  $E_{max}$  and  $E_{min}$  are the maximal and minimal values of the computed data  $E_{obs}$ , and  $\sigma$  is the noise in percents.

### 8.1.1 Test 1

First we performed tests where the trace of the incoming wave was measured at the observation points at the lower boundary of  $\Omega_{FEM}$  in the time interval  $(0, T]$ , and then at the observation points at the top boundary in the time interval  $(T, 2T]$ .

In Figures 5-6 we present a comparison of the computed  $L_2$ -norms,  $\|E - E_{obs}\|_{L_2}$ , depending on the relative noise  $\sigma$  on the different adaptively refined meshes. The norms are plotted as long as they decrease. The relative noise  $\sigma$  in the data is computed using expression (32). From these results we conclude that the reconstruction is stable with

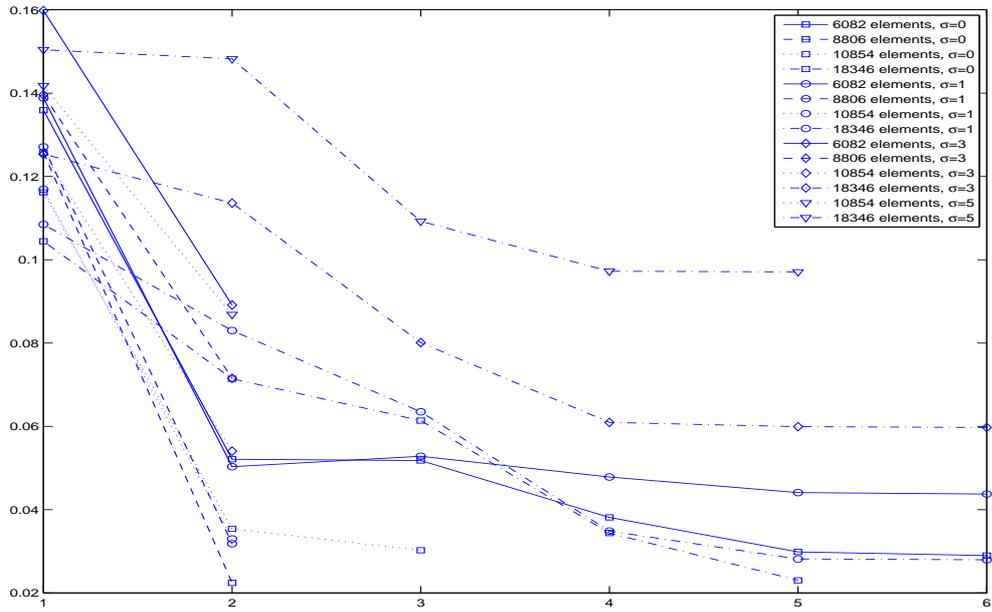


Figure 5: Tests 1:  $\|E - E_{obs}\|_{L_2}$  on adaptively refined meshes. Computations was performed with noise level  $\sigma = 0, 1, 3$  and 5% and with regularization parameter  $\gamma = 0.01$ . Here the  $x$ -axis denotes number of optimization iterations.

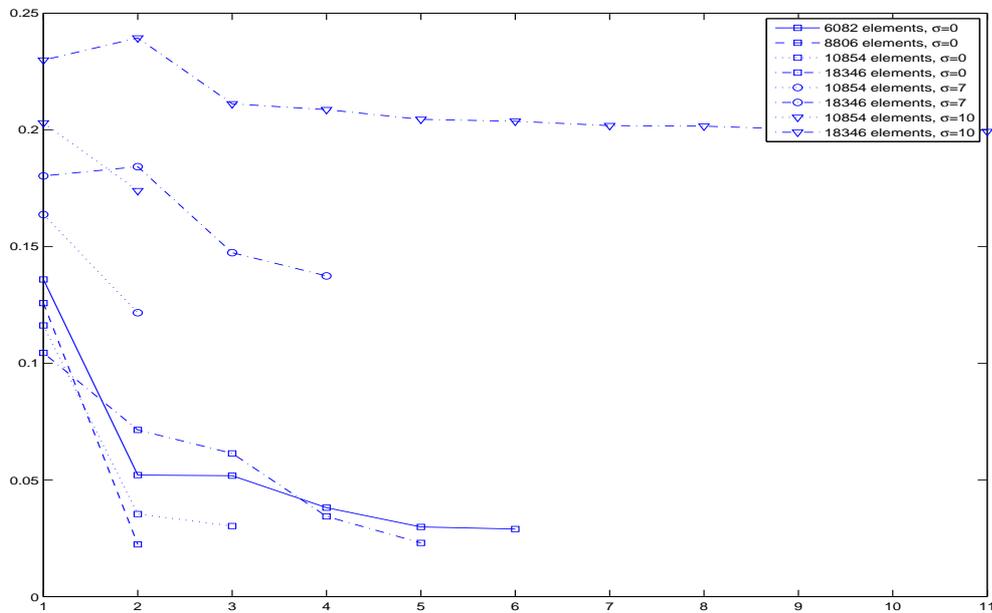


Figure 6: Test 1:  $\|E - E_{obs}\|_{L_2}$  on adaptively refined meshes. Computations was performed with noise level  $\sigma = 0, 7$  and 10% and with regularization parameter  $\gamma = 0.01$ . Here the  $x$ -axis denotes number of optimization iterations.

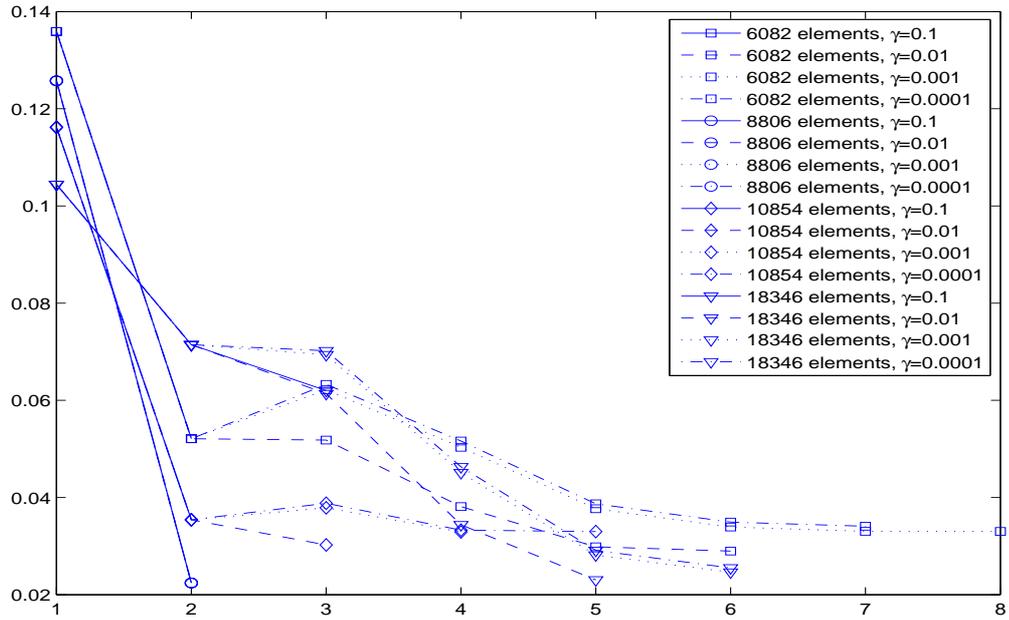


Figure 7: Test1:  $\|E - E_{obs}\|_{L_2}$  on adaptively refined meshes. Computations was performed with noise level  $\sigma = 0\%$ , and with regularization parameters  $\gamma = 0.1, 0.01, 0.001, 0.0001$ , Here the  $x$ -axis denotes number of optimization iterations.

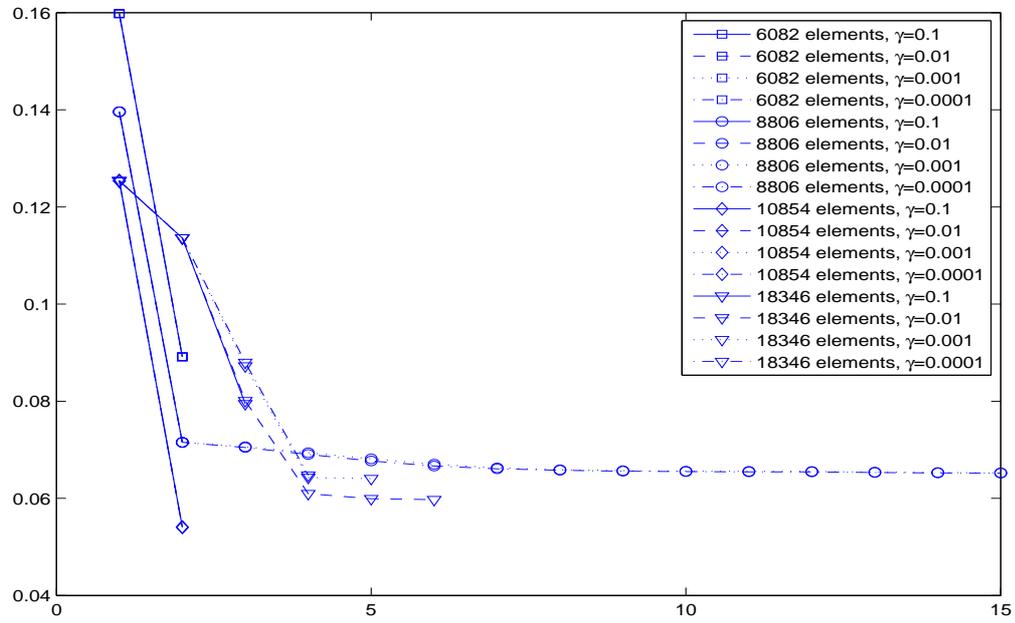


Figure 8: Test 1:  $\|E - E_{obs}\|_{L_2}$  on adaptively refined meshes. Computations was performed with noise level  $\sigma = 3\%$ , and with regularization parameters  $\gamma = 0.1, 0.01, 0.001, 0.0001$ , Here the  $x$ -axis denotes number of optimization iterations.

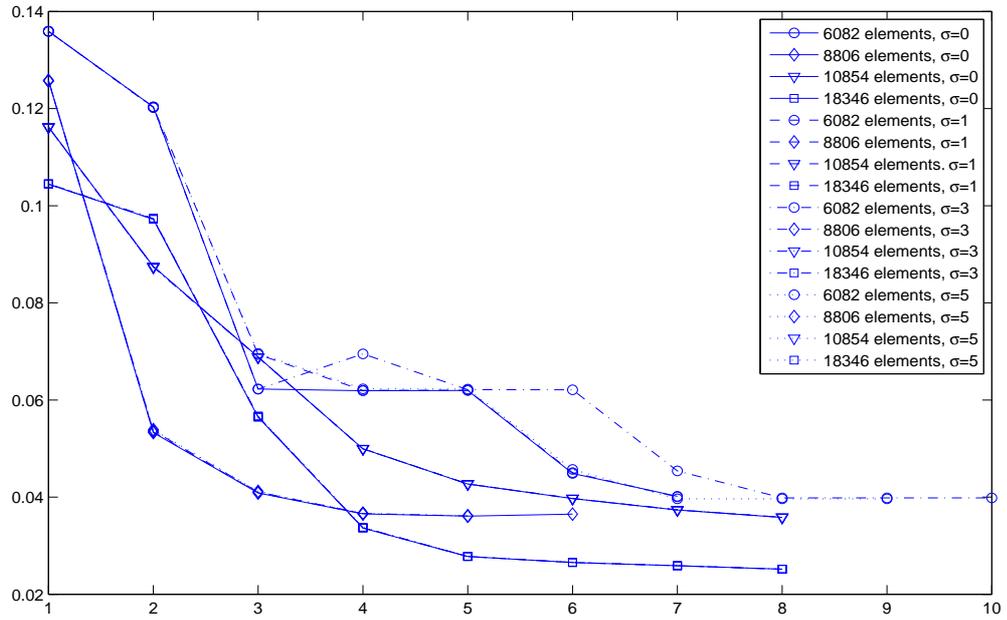


Figure 9: Test 2:  $\|E - E_{obs}\|_{L_2}$  on adaptively refined meshes. Computations was performed with noise level  $\sigma = 0, 1, 3$  and  $5\%$  and with regularization parameter  $\gamma = 0.01$ . Here the  $x$ -axis denotes number of optimization iterations.

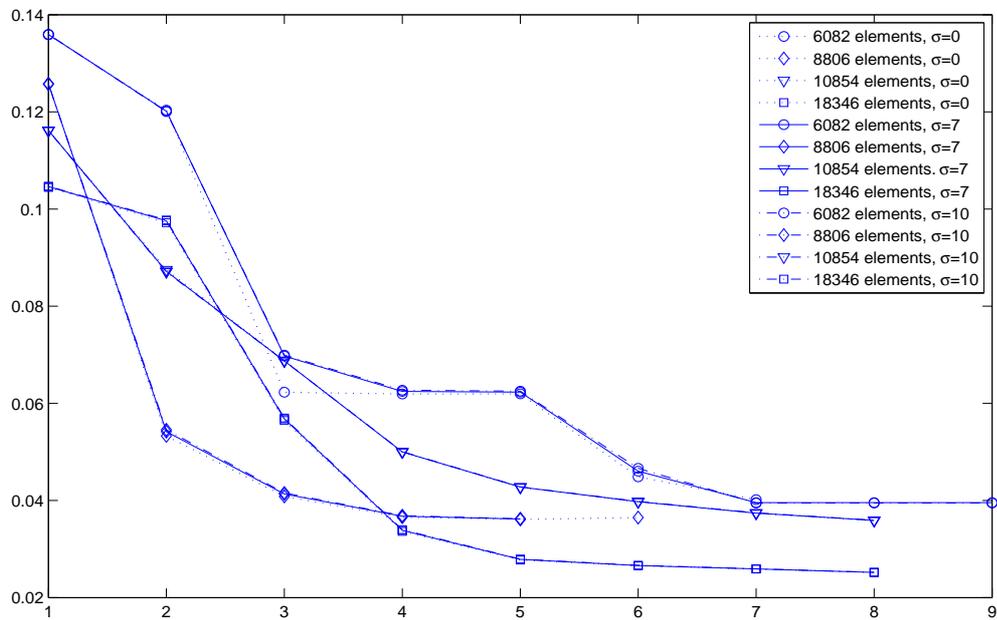


Figure 10: Test 2:  $\|E - E_{obs}\|_{L_2}$  on adaptively refined meshes. Computations was performed with noise level  $\sigma = 0, 7$  and  $10\%$  and with regularization parameter  $\gamma = 0.01$ . Here the  $x$ -axis denotes number of optimization iterations.

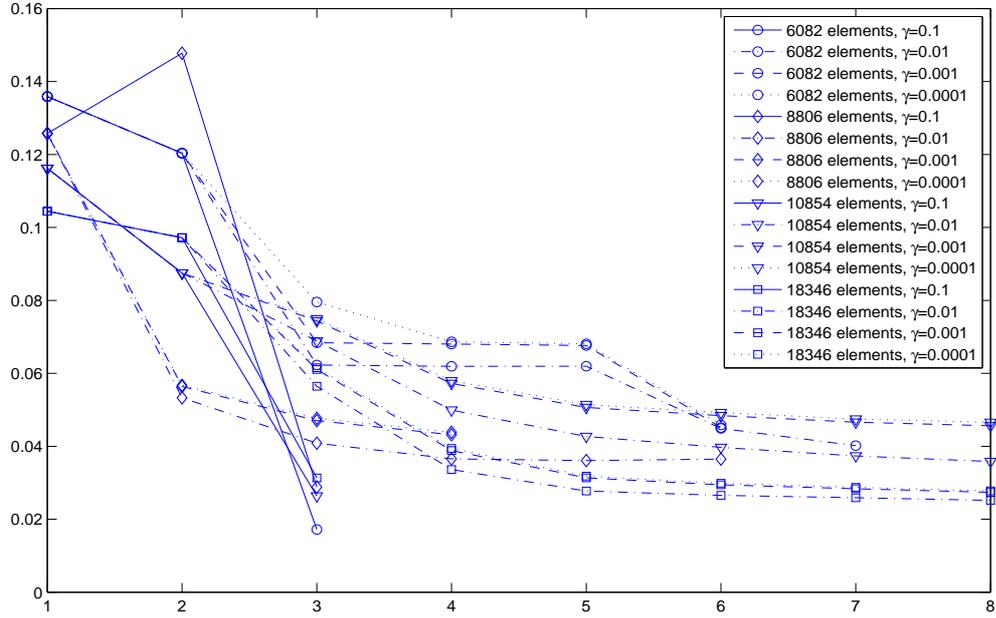


Figure 11: Test 2:  $\|E - E_{obs}\|_{L_2}$  on adaptively refined meshes. We show computational results with noise level  $\sigma = 1\%$  and with regularization parameters  $\gamma = 0.1, 0.01, 0.001, 0.0001$ . Here the  $x$ -axis denotes number of optimization iterations.

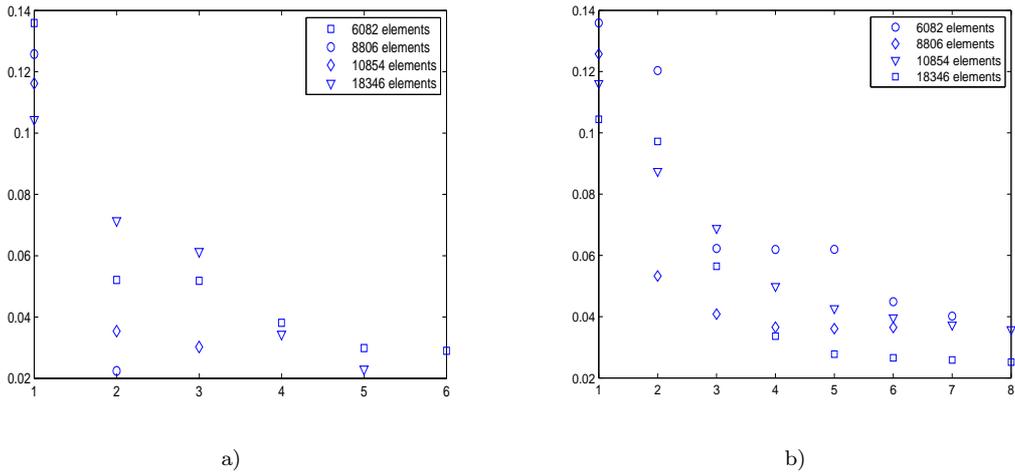


Figure 12: Test 2:  $\|E - E_{obs}\|_{L_2}$  on adaptively refined meshes. We show computations: on a) with noise level  $\sigma = 0\%$  and with regularization parameter  $\gamma = 0.01$  for Test 1; on b) with noise level  $\sigma = 1\%$  and with regularization parameter  $\gamma = 0.01$  for Test 2. Here the  $x$ -axis denotes number of optimization iterations.

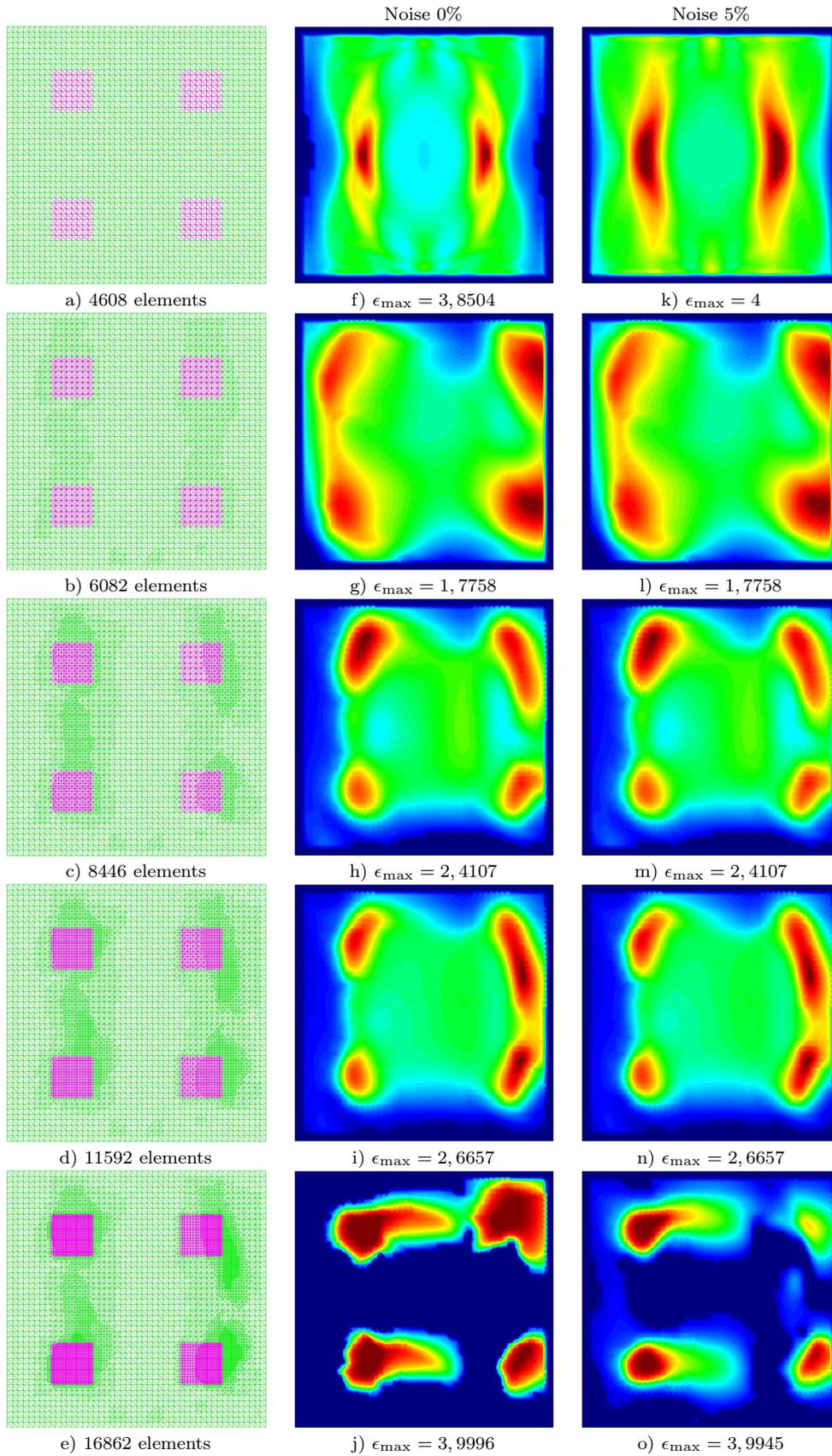


Figure 13: Example 2: the spatial distribution of  $\epsilon_h$  on different adaptively refined meshes. Here, red color corresponds to the maximum parameter value on the corresponding meshes, and blue color - to the minimum,  $\epsilon_{\min} = 1.0$  in all the plots.

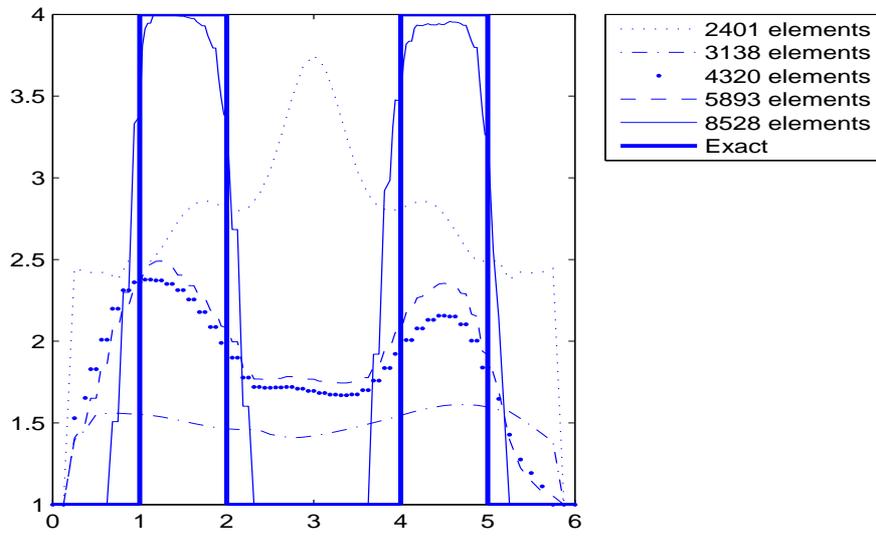


Figure 14: Example 2: the one-dimensional cross-sections of the image of function  $\epsilon_{comp}$  along the vertical line connecting the points  $(-1.5,-3.0)$  and  $(-1.5,3.0)$  computed for corresponding refined meshes with noise level 0 % on data.

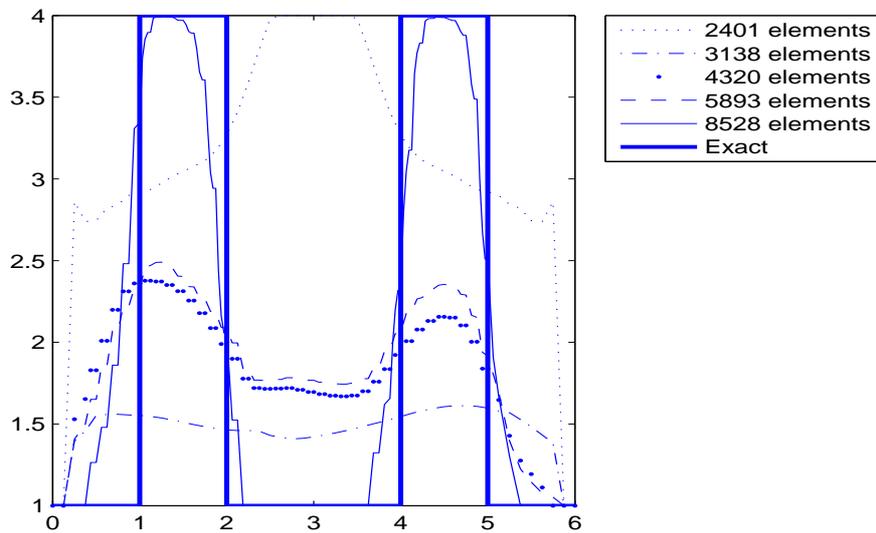


Figure 15: Example 2: the one-dimensional cross-sections of the image of function  $\epsilon_{comp}$  along the vertical line connecting the points  $(-1.5,-3.0)$  and  $(-1.5,3.0)$  computed for corresponding refined meshes with noise level 5 % in data.

small values of the noise (see Figure 5), and unstable when adding more than 5% noise to the data (Figure 6). We also used smoothness indicator in parameter update by local averaging over the neighboring elements.

In Figures 7-8 we show a comparison of the computed  $L_2$ -norms,  $\|E - E_{obs}\|_{L_2}$ , depending on the different regularization parameters  $\gamma$ . We see that we obtain the smallest value of  $\|E - E_{obs}\|_{L_2}$  with the regularization parameter  $\gamma = 0.01$ , while choosing  $\gamma = 0.1$  is too large and involve too much regularization. The computational tests show that the best results are obtained on the finest mesh, where  $\|E - E_{obs}\|_{L_2}$  is reduced by approximately a factor of seven between the first and last optimization iterations. Figure 4-e)-h) correspond to Figure 12-a) and show the reconstructed parameter field  $\epsilon(x)$  at the final optimization iteration.

### 8.1.2 Test 2

The tests described in this section, was performed by measuring the trace of the incoming wave at the observation points on both the lower and upper boundaries of the computational domain  $\Omega_{FEM}$ . Thus, we have twice as much information as in the previous test, and we expect to get a more quantitative reconstruction of the structure.

In Figures 9-10 we present a comparison of the computed  $L_2$ -norms,  $\|E - E_{obs}\|_{L_2}$ , depending on the relative noise  $\sigma$  on the different adaptively refined meshes. The norms are plotted as long as they decrease. The relative noise,  $\sigma$ , in the data is computed using expression (32). From these results we conclude that the reconstruction is stable on the two, three and four times refined meshes, even when 10% noise is added to the data.

In Figure 11 we show a comparison of the computed  $L_2$ -norms,  $\|E - E_{obs}\|_{L_2}$ , depending on the different regularization parameters  $\gamma$ . We see that the smallest value of  $\|E - E_{obs}\|_{L_2}$  is obtained with regularization parameter  $\gamma = 0.01$ , while  $\gamma = 0.1$  is again too large and involve too much regularization. The computational tests show that the best results are obtained on the finest mesh, where  $\|E - E_{obs}\|_{L_2}$  is reduced by approximately a factor of seven between the first and last optimization iterations, see Figure 12-b). Figure 4-i)-l) correspond to Figure 12-b), and show the reconstructed parameter field  $\epsilon(x)$  at the final optimization iteration.

## 8.2 Example 2

We continue to test our adaptive algorithm on the reconstruction of the same periodic structure as in Figure 3, but for the case where a good initial guess of the solution is unknown. As before, we assume that  $\epsilon = 1$  in  $\Omega_{FDM}$ . We measure the trace of the incoming wave at the observation points on both the lower and the upper boundaries of the computational domain  $\Omega_{FEM}$  as in Test 2. We start our adaptive algorithm with initial guess  $\epsilon = 1.5$  at the inner points of the computational domain  $\Omega_{FEM}$ , and with  $\epsilon = 1.0$  at the overlapping nodes of the domain  $\Omega_{FEM} \cap \Omega_{FDM}$ . We also enforce that the parameter  $\epsilon(x)$  belongs to the set of the admissible parameters  $C_M$  defined above. We stop the computations in the quasi-Newton procedure when the computed  $L_2$ -norms of  $\|E - E_{obs}\|_{L_2}$  are stabilized.

Figure 13 shows that the reconstruction of the parameter is improved as the meshes are refined. However, the locations of the imaged right squares are shifted slightly to the right. The meshes are refined using our mesh-adaptive algorithm. We note that the coarse and the one time refined meshes are the same as in the previous examples, while the two, three and four times refined meshes are different. We present the results of the reconstruction

with the noise level in the data being 0% and 5%. As before, the noise is computed using expression (32).

On Figures 14-15 we show the one-dimensional cross-sections of the image of the functions  $\epsilon^n$  along the vertical line passing through the middle of the right small square, superimposed with the correct  $\epsilon(x)$ . In Figure 14 the noise level is 0% and in Figure 15 it is 5%. We observe that the images deteriorate or achieve a local minima on the coarse mesh. The reconstruction is dramatically improved as the meshes are refined using the adaptive algorithm.

We also performed the same tests as above, but starting with  $\epsilon = 2.0$  at the inner points of  $\Omega_{FEM}$ , and  $\epsilon = 1.0$  at the overlapping nodes of  $\Omega_{FEM} \cap \Omega_{FDM}$ . The reconstructed parameter deteriorated not only on the coarse mesh, but also on the one and two times refined meshes. Thus, the adaptivity works in a neighborhood of an initial guess  $1 \leq \epsilon \leq 1.5$ . We note that the usual quasi-Newton algorithm without adaptivity works only with guess  $\epsilon = 1$  and deteriorate for  $\epsilon = 1.5$ , see Fig. 13-f),k) and Test 5 in [9].

## 9 Conclusions and Remarks

We have devised an explicit, adaptive hybrid FEM/FDM method for an inverse electromagnetic scattering problem. The method is hybrid in the sense that different numerical methods, finite elements and finite differences, are used in different parts of the computational domain. We derived an a posteriori estimate for the error in the Lagrangian in the case when we have first order absorbing [13] and Neumann boundary conditions in the formulation of the forward problem. The adaptivity is based on a posteriori error estimates for the associated Lagrangian in the form of space-time integrals of the residuals multiplied by the dual weights. We illustrated the usefulness of the adaptive error control on an inverse scattering problem for recovering the electric permittivity from boundary measured data. Our numerical experiments show that adaptivity can mitigate the problem of local minima in the usual optimization algorithms.

## References

## References

- [1] R. Becker. Adaptive finite elements for optimal control problems. *Habilitationsschrift*, 2001.
- [2] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica, Cambridge University Press*, pages 1–225, 2001.
- [3] L. Beilina. Adaptive FEM method for an inverse scattering problem. *J. Inverse Problems and Information Technologies*, 1, 2-nd issue, 2003.
- [4] L. Beilina. *Adaptive finite element/difference methods for time-dependent inverse scattering problems*. PhD thesis, Department of Computational Mathematics, Chalmers University of Technology, 2003.
- [5] L. Beilina. Adaptive Hybrid Finite Element/Difference methods: application to inverse elastic scattering. *J. Inverse Problems and Information Technologies*, 6, 2003.

- [6] L. Beilina and C. Clason. An adaptive hybrid fem/fdm method for an inverse scattering problem in scanning acoustic microscopy. *J. SIAM Sci.Comp.*, 28(1), 2006.
- [7] L. Beilina and C. Johnson. A Hybrid FEM/FDM method for an Inverse Scattering Problem. In *Numerical Mathematics and Advanced Applications - ENUMATH 2001*. Springer-Verlag, 2001.
- [8] L. Beilina and C. Johnson. A posteriori error estimation in computational inverse scattering. *J. Mathematical models and methods in applied sciences*, 15(1):23–37, 2005.
- [9] L. Beilina and M. V. Klibanov. A globally convergent numerical method for some coefficient inverse problems with resulting second order elliptic equations. Technical Report No.11, NTNU, Norway, 2007. Available online at [http://www.ma.utexas.edu/mp\\_arc/index-07.html](http://www.ma.utexas.edu/mp_arc/index-07.html), preprint number 07-311, and at <http://www.math.uncc.edu/people/research/mklibanv.php>.
- [10] L. Beilina, K. Samuelsson, and K. Åhlander. Efficiency of a hybrid method for the wave equation. In *International Conference on Finite Element Methods*, Gakuto International Series Mathematical Sciences and Applications. Gakkotosho CO.,LTD, 2001.
- [11] M. Cheney and D. Isaacson. Inverse problems for a perturbed dissipative half-space. *Inverse Problems*, 11:865–888, 1995.
- [12] H. W. Engl, M. Hanke, and A. Neubauer. *Regularization of Inverse Problems*. Dordrecht, Kluwer, 1996.
- [13] B. Engquist and A. Majda. Absorbing boundary conditions for the numerical simulation of waves. *Math. Comp.*, 31:629–651, 1977.
- [14] K. Eriksson, D. Estep, and C. Johnson. Introduction to adaptive methods for differential equations. *Acta Numerica 1995 Cambridge University Press*, pages 105 –158, 1995.
- [15] K. Eriksson, D. Estep, and C. Johnson. *Computational Differential Equations*. Studentlitteratur, Lund, 1996.
- [16] P. Hansbo and C. Johnson. Adaptive finite element methods for elastostatic contact problems. *IMA Volumes in Mathematics and its Applications*, 113:135 –150, 1998.
- [17] J. Hoffman and C. Johnson. *Dynamic computational subgrid modelling*. Lecture Notes in Computational Science and Engineering, Springer, 2003.
- [18] C. Johnson. Adaptive computational methods for differential equations. In *ICIAM99*, pages 96–104. Oxford University Press, 2000.
- [19] M. Klibanov and A. A. Timonov. *Carleman estimates for coefficient inverse problems and numerical applications*. VSP Publishing, Utrecht, 2004.
- [20] J. Nocedal. Updating quasi-newton matrices with limited storage. *J. Mathematical of Comp.*, 35, N.151:773–782, 1991.
- [21] O. Pironneau. *Optimal shape design for elliptic systems*. Springer Verlag, Berlin, 1984.
- [22] A. N. Tikhonov and V. Y. Arsenin. *Solution of Ill-posed Problems*. Wiley, 1977.