CHALMERS
FINITE ELEMENT CENTER

PREPRINT 2001-04

A Posteriori Error Estimation of Functionals in Elliptic Problems: Experiments

Mats G. Larson and A. Jonas Niklasson

Chalmers Finite Element Center
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg Sweden 2001

CHALMERS FINITE ELEMENT CENTER

Preprint 2001-04

A Posteriori Error Estimation of Functionals in Elliptic Problems: Experiments

Mats G. Larson and A. Jonas Niklasson

CHALMERS

Chalmers Finite Element Center
Chalmers University of Technology
SE–412 96 Göteborg Sweden
Göteborg, January 2001

A Posteriori Error Estimation of Functionals in Elliptic Problems: Experiments
Mats G. Larson and A. Jonas Niklasson
NO 2001-04
ISSN 1404–4382

Chalmers Finite Element Center
Chalmers University of Technology
SE–412 96 Göteborg
Sweden
Telephone: +46 (0)31 772 1000
Fax: +46 (0)31 772 3595
www.phi.chalmers.se

Printed in Sweden
Chalmers University of Technology
Göteborg, Sweden 2001

A Posteriori Error Estimation of Functionals in Elliptic Problems: Experiments*

Mats G. Larson † and A. Jonas Niklasson ‡

January 31, 2001

Abstract

In many applications the main goal when numerically solving a differential equation is to calculate a quantity of particular interest rather than the solution itself. Efficiency can be gained by adapting the numerical method to efficiently approximate the particular quantity. Such adaption may be based on an a posteriori estimate of the error in the quantity. The error is estimated in terms of a sum of local contributions estimated by a local computable residual and a weight which is estimated by solving an associated dual problem. The success of such a method depends on the accuracy of the a posteriori estimate and the design of the adaptive mesh refinement strategy. We examine three different levels of analytical estimates of the residual and corresponding weights numerically in the case of pointwise error estimates. We also discuss aspects of implementation of the methodology.

1 Introduction

Adaptive finite element methods for approximation of solutions to differential equations are a fundamental tool in many areas of engineering and science. An adaptive finite element method seeks to realize a computational goal with as few computational degrees of freedom as possible. The computational goal may take the abstract form: compute an approximate solution u_h such that the error $e = u - u_h$, where u is the exact solution, satisfies

$$|\lambda(e)| < tol, \tag{1.1}$$

where $\lambda(e)$ is a functional of the error and tol is a tolerance provided by the user. The functional $\lambda(e)$ could represent various quantities of interest in a particular application,

*This research was supported by The Swedish Foundation for International Cooperation in Research and Higher Education.

†Corresponding author, Department of Mathematics, Chalmers University of Technology, Göteborg, S-412 96, Sweden, mgl@math.chalmers.se, Supported by the Swedish Foundation for Strategic Research

‡Department of Mechanics, Chalmers University of Technology, Göteborg, S-412 96, Sweden, jnik@mec.chalmers.se

for instance, the heat flux through a part of the boundary in a heat conduction problem or the stress intensity factor in an elasticity problem, or a standard norm, such as the L^2 or energy norm, of the error in a subdomain or the domain itsef. The computational goal may have a large impact on the optimal discretization and it is therefore important that the design of the adaptive algorithm reflects the computational goal.

Such adaptive algorithms are usually based on an a posteriori error estimate which estimate the error in terms of a residual quantity measuring how well the approximate solution u_h satisfies the differential equation and a weight obtained from solving a dual problem numerically. The a posteriori error estimates for the functional $\lambda(e)$ are based on a representation formula of the error

$$|\lambda(e)| \leq \sum_T (R_T, \phi)_T, \tag{1.2}$$

where we sum over the elements T in a partion of the computational domain. R_T is the element contribution to the residual, obtained by inserting the computed solution into the differential equation. Further ϕ is the solution to an associated dual problem connecting the residual to the error in the functional. A posteriori error estimates can now be obtained in a various ways by estimating the right hand side of (1.2). We consider three simple estimates and investigate the efficiency of these different estimates numerically on several test cases. The a posteriori error estimates are valid for continuous piecewise polynomial spaces with variable meshsize and degree of polynomials. See, for instance [1] and [9], for details on such spaces. Further, it is not difficult to extend the estimates to classical nonconforming elemnts such as the Crouzeix-Raviart element.

Based on the a posteriori error estimates we design an adaptive method seeking to realize the desired error control using as few computational degrees of freedom as possible. Here we restrict our attention to h-adaptivity for continuous piecewise bilinear and biquadratic polynomials on rectangular elements allowing hanging nodes. In the adaptive method we solve the dual problem numerically, evaluate the a posteriori error estimates, and adapt the mesh. This process is repeated until the stopping criteria (1.1) is satisfied.

Two important questions arise:

- How accurate is the a posteriori error estimate ?
- Is the adaptive meshrefinement process and its convergence sensitive to the choice a posteriori estimate ?

In this paper we study these two questions numerically for an estimate of the pointwise error for a variety of test cases. It is found that in some cases the accuracy of the a posteriori estimates can detoriate significantly due to the analytical estimates of the right hand side in (1.2) because of loss of cancellation. This fact is indeed a serious problem if the estimate is to be used as a basis for a stopping criterion. For the second question we found that although different estimates may create visually different meshes no significant difference was found in the rate of convergence. Thus the adaptive refinement process was rather stable to different estimates. Similar investigations was presented by Barth and Larson [2] in the context of compressible fluid flow.

A posteriori error estimates related to ours appears, for instance, in Becker and Rannacher [3], Giles et al [6], and Prudhomme and Oden [8]. Weighted a posteriori error estimates are natural generalizations of the global norm a posteriori estimates presented in [5]. For a general introduction to a posteriori error estimates we refer to [4] and [10].

The reminder of this paper is organized as follows: in Section 2 we introduce a model problem and the finite element method, in Section 3 we derive three a posteriori error estimates. In section 4 we discuss practical aspects of the implementation of the a posteriori error estimates, including computation of the residual estimators and weights. We describe the adaptive algorithm. We also present comparisons of efficiency indices for the different estimates on several different examples.

2 Model problem and the finite element method

2.1 The model problem

Throughout this work Ω denotes a bounded domain in $\mathbf{R}^d$, $d = 1, 2$, or 3, with boundary Γ. We let $(\cdot,\cdot)$ be the scalar product in $L^2 = L^2(\Omega)$ and $\|\cdot\|$ denotes the corresponding L^2 norm. Further $H^s = H^s(\Omega)$ denote the standard Sobolev spaces.

We consider the following boundary value problem: find $u : \Omega \to \mathbf{R}$ such that

$$\begin{aligned} -\nabla \cdot \sigma(u) &= f \quad \text{in } \Omega, \\ n \cdot \sigma(u) &= g \quad \text{on } \Gamma_N, \\ u &= 0 \quad \text{on } \Gamma_D, \end{aligned} \tag{2.1}$$

where $\Gamma = \Gamma_D \cup \Gamma_N$ is a partion of Γ, the flux $\sigma(u) = A\nabla u$ with $A = A(x) \in C^1(\overline{\Omega})$ a symmetric uniformly positive matrix. Using the notation

$$a(v,w) = (\sigma(v), \nabla w), \tag{2.2}$$

$$l(v) = (f,v) + \int_{\Gamma_N} gv\, ds, \tag{2.3}$$

for all $v, w \in \mathcal{V} = \{v \in H^1 : v = 0 \text{ on } \Gamma_D\}$, we may formulate the weak version of (2.1): find $u \in \mathcal{V}$ such that

$$a(u,v) = l(v) \quad \text{for all } v \in \mathcal{V}. \tag{2.4}$$

As is well known, if $f \in H^{-1}$ and $g \in H^{1/2}(\Gamma_D)$ there exist a unique solution in $\mathcal{V}$ for $\Gamma_D \neq \emptyset$ and a solution in $\mathcal{V}$, which is unique up to a constant, when $\Gamma_D = \emptyset$. A basic example is the Poisson equation obtained by taking A equal to the identity matrix and we will return to this problem in the numerical examples.

2.2 The finite element method

To define the finite element method we introduce a partion $\mathcal{T}_h = \{T\}$ of Ω, with piecewise constant mesh functions h defined by $h|_T = \mathrm{diam}(T)$. We let $\mathcal{V}_h \subset \mathcal{V}$ be a space of continuous piecewise polynomials of degree p defined on $\mathcal{T}_h$, where p is allowed to be different at each element T. The degree of polynomials on each element is defined to be the maximal degree so that we have complete set of polynomials. Note that there are several different families of spaces which satisfy this condition, see Szabo and Babuska [9], and Ainsworth and Senior [1], for details on the construction and implementation of such spaces. The standard finite element method reads: find $u_h \in \mathcal{V}_h$ such that

$$a(u_h, v) = l(v) \quad \text{for all } v \in \mathcal{V}_h. \tag{2.5}$$

Note that the meshsize and the degree of polynomials used are allowed to vary throughout the domain.

3 A posteriori error estimation

3.1 Error representation by duality

We shall now derive an estimate for the error $\lambda(u) - \lambda(u_h)$ in a given linear functional $\lambda(\cdot)$. To derive a representation formula for the error in the functional we introduce the dual problem

$$a(v, \phi) = \lambda(v) \quad \text{for all } v \in \mathcal{V}. \tag{3.1}$$

Setting $v = u - u_h =: e$ in (3.1) we obtain:

$$\begin{aligned}
\lambda(u) - \lambda(u_h) &= \lambda(e) && (3.2)\\
&= a(e, \phi) && (3.3)\\
&= a(e, \phi - \pi\phi) && (3.4)\\
&= l(\phi - \pi\phi) - a(u_h, \phi - \pi\phi) && (3.5)\\
&= \sum_{T \in \mathcal{T}_h} (f + \nabla \cdot \sigma(u_h), \phi - \pi\phi)_T && (3.6)\\
&\qquad + (g - \sigma_n(u_h), \phi - \pi\phi)_{\partial T \cap \Gamma_N} - (\sigma_n(u_h) - \sigma_{n,h}(u_h), \phi - \pi\phi)_{\partial T \setminus \Gamma},
\end{aligned}$$

where we used Galerkin orthogonality (2.5) to insert an interpolant $\pi\phi \in \mathcal{V}_h$ of ϕ in equality (3.5), and split the integral into a sum of integrals over the elements, and integrated by parts to obtain (3.6). The flux $\sigma_{n,h}(u_h)$ is a numerical approximation of the true flux $\sigma_n(u)$, which we choose to be the average $\sigma_{n,h}(u_h) = (\sigma_n(u_h^+) + \sigma_n(u_h^-))/2$, where $v^{\pm}(x) = \lim_{s \to 0, s > 0} v(x + sn)$, for $x \in \partial T$.

Introducing the element residual $R_T(u_h) \in H^{-1}(T)$ by

$$\begin{aligned}
(R_T(u_h), v)_T = (f + \nabla \cdot \sigma(u_h), v)_T \qquad\qquad\qquad\qquad\qquad\qquad\qquad (3.7)\\
+ (g - \sigma_n(u_h), v)_{\partial T \cap \Gamma_N} - (\sigma_n(u_h) - \sigma_{n,h}(u_h), v)_{\partial T \setminus \Gamma},
\end{aligned}$$

for all $v \in H^1(T)$, we finally obtain the error representation formula

$$\lambda(u) - \lambda(u_h) = \sum_{T \in \mathcal{T}_h} (R_T(u_h), \phi - \pi\phi)_T. \tag{3.8}$$

3.2 Error estimates

Starting from (3.8) and estimating the right hand side using the triangle inequality followed by the Cauchy-Schwarz inequality on an element level we obtain the following three estimates of the error

$$\begin{aligned} |\lambda(e)| &= \Big| \sum_{T \in \mathcal{T}_h} (R_T(u_h), \phi - \pi\phi)_T \Big| && (3.9) \\ &\leq \sum_{T \in \mathcal{T}_h} |(R_T(u_h), \phi - \pi\phi)_T| && (3.10) \\ &\leq \sum_{T \in \mathcal{T}_h} \mathcal{R}_T(u_h) \cdot \mathcal{W}_T(\phi). && (3.11) \end{aligned}$$

Here $\mathcal{R}_T(u_h)$ and the weight $\mathcal{W}_T(\phi)$ are estimates of the residual and the local interpolation error in the solution ϕ to the dual problem defined by

$$\mathcal{R}_T(u_h) = \begin{bmatrix} \|f + \nabla \cdot \sigma(u_h)\|_T \\ h\|g - \sigma_{n,h}(u_h)\|_{\partial T \cap \Gamma_N} + h\|\sigma_{n,h}(u_h) - \sigma_n(u_h)\|_{\partial T \setminus \Gamma} \end{bmatrix}, \tag{3.12}$$

$$\mathcal{W}_T(\phi) = \begin{bmatrix} \|\phi - \pi\phi\|_T \\ h^{-1}\|\phi - \pi\phi\|_{\partial T \cap \Gamma_N} + h^{-1}\|\phi - \pi\phi\|_{\partial T \setminus \Gamma} \end{bmatrix}. \tag{3.13}$$

Note that the first estimate (3.9) is an identity, the second (3.10) does not admit cancellation between different elements, and in the third (3.11) cancellation on an element level is also excluded.

Remark 3.1 In several works on a posteriori error estimates, for instance, [3] and [4], an interpolation error estimate is used to estimate the weight $\mathcal{W}_T(\phi)$ by $ch^\alpha|\phi|_{T,\alpha}$, for suitable α. Here the constant c is in general unknown and the computation of $|\phi|_{T,\alpha}$ is probably not simpler than actually trying to directly compute an approximation of the local interpolation error $\|\phi - \pi\phi\|_T$. Therefore we have chosen to avoid using the interpolation error estimate.

Finally, we introduce some convenient notation. We denote the element indicators (or contributions) $E_T^i, i = 1, 2, 3$, corresponding to estimates (3.9–3.11) by

$$\begin{aligned} E_T^1 &= (R_T(u_h), \phi - \pi\phi)_T, && (3.14) \\ E_T^2 &= |(R_T(u_h), \phi - \pi\phi)_T|, && (3.15) \\ E_T^3 &= \mathcal{R}_T(u_h) \cdot \mathcal{W}_T(\phi). && (3.16) \end{aligned}$$

Note that the last two indicators are positive and may thus be used as a basis for an adaptive refinement algorithm. Furthermore, for brevity, we denote the global estimates (3.9–3.11) by

$$E^i = \sum_{T \in \mathcal{T}_h} E_T^i. \tag{3.17}$$

4 Numerical results

4.1 Test problem

In this section we consider the following model problem

$$-\Delta u = f \quad \text{in } \Omega, \tag{4.1}$$

$$u = 0 \quad \text{on } \Gamma, \tag{4.2}$$

where $\Omega = [0,1] \times [0,1]$, $\Gamma = \Gamma_D$ Further f is chosen so that the exact solution of (4.1) is given by

$$u(x,y) = \sin \pi x \sin \pi y e^{-(r/R)^2}, \tag{4.3}$$

with $r = \sqrt{(x-0.5)^2 + (y-0.5)^2}$ and $R = 0.1$, or

$$u(x,y) = \sin \pi m x \sin n \pi y, \tag{4.4}$$

for various integer values of m and n. We consider an a posteriori estimate of the error in a local average centred at a given point (x_0, y_0) of the solution, i.e., $\lambda(e) = (e, \psi)$ with ψ given by

$$\psi = c e^{-(r/R)^4}, \tag{4.5}$$

with $r = \sqrt{(x-x_0)^2 + (y-y_0)^2}$, $R = 0.05$, and the constant c is chosen such that $\int_\Omega \psi \, dx = 1$. For each test case we calculate and plot: (1) the three error estimates E^i, (2) the efficiency indices,

$$I^i = \left| \frac{E^i}{e} \right|, \tag{4.6}$$

$i = 1, 2, 3$, (3) the final refined mesh, and (4) the base 10 logarithm of the quotient of the element indicators

$$\log_{10} \left(\frac{E_T^3}{E_T^2} \right), \tag{4.7}$$

of estimates three and two.

The finite element method uses tensor product elements on rectangles allowing hanging nodes, see [1]. In order to eliminate questions on the accuracy of the solution to the dual problem we actually solve the dual problem using polynomials of degree two orders higher than the primal problem on the same mesh as we solve the primal problem. Of course this is not a practical scheme but here our focus is to explore properties and limitations of the analytical estimates. The interpolant $\pi\phi$ is the usual Lagrange nodal interpolant. That sufficient accuracy in the dual problem is obtained is clear from the fact that the efficiency index I^1 for the first estimate E^1 is very close to one in all computations.

4.2 Adaptive algorithm

The adaptive algorithm seeks to realize the stopping criterion

$$|E^1| \leq tol, \tag{4.8}$$

for a given tolerance tol. The refinement of the mesh is based either on E_T^2 or E_T^3 and aims at reducing the global residual E^i by a factor γ, with $0 < \gamma < 1$, i.e.,

$$E_{new}^i \leq \gamma E_{old}^i, \tag{4.9}$$

with $i = 2$ or 3. To determine how many elements to refine we assume that the element contributions satisfies $E_T^i \sim c_T h_T^{\alpha_T}$, for constants c_T, α_T and h_T the size of T. Under this assumption we have $E_{T,new}^i = E_{T,old}^i 2^{-\alpha_T}$. The constant α_T can be estimated using a least squares fit from values on previous meshes. Given α_T we can estimate the effect of refinement of an element can and we can simply start by refining elements with large residuals until we expect $E_{new}^i \leq \gamma E_{old}^i$ to hold. In addition, if an element has three refined neighbors the element is refined and, furthermore, only one hanging node on each edge is allowed.

4.3 Conclusions

Our numerical results presented below, Figures 1–10, indicate that:

- The use of analytical estimates in the a posteriori estimates can create bad accuracy. Note that there are no unknown constants in the experiments and thus the loss in accuracy is only a consequence of the analytical estimates.
- The element indicators (or contributions) are very sensitive to the indicator used, i.e., there are large local differences between E_T^2 and E_T^3.
- The adaptive meshrefinement algorithm creates meshes which are visually different when based on different indicators but the convergence of the goal functional is not very different.

Based on these observations we propose that an adaptive mesh refinement algorithm is based on E_T^2 while the stopping criterion is based on E^1. Mesh refinement could also be based on E^3. However, we believe it may actually be more efficient to calculate E_T^2 instead of E_T^3.

4.4 Future work

To turn these techniques into a mainstream tool used in engineering computations and in commercial finite element software an efficient and sufficiently accurate way of calculating the local contributions E_T^1 needs to be developed. This is a challenging problem since the orthogonality properties of the Galerkin method implies that one needs a rather accurate solution to the dual problem to calculate E_T^1. For work in this direction we refer to Larson and Samuelsson [7], where adaptive meshrefinement and solution of the primal and dual problems are combined in a multigrid method.

References

[1] M. Ainsworth and B. Senior. An adaptive refinement for *hp*-finite element computations. *Appl. Numer. Math.*, 26:165–178, 1998.

[2] T.J. Barth and M.G. Larson. A posteriori error estimates for functionals in compressible flow. In B. Cockburn, K.E. Karniadakis, and C.-W. Shu, editors, *Discontinuous Galerkin Methods: Theory, Computation, and Applications*, Lecture Notes in Computational Science and Engineering. Springer Verlag, 1999.

[3] R. Becker and R. Rannacher. A feed-back approach to error control in finite element methods: basic analysis and examples. *East-West J. Numer. Math*, 4(4):237–264, 1996.

[4] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. *Computational Differential Equations*. Cambridge University Press, 1996.

[5] K. Eriksson and C. Johnson. Adaptive finite element methods for parabolic problems I: A linear model problem. *SIAM J. Numer. Anal*, 28:43–77, 1991.

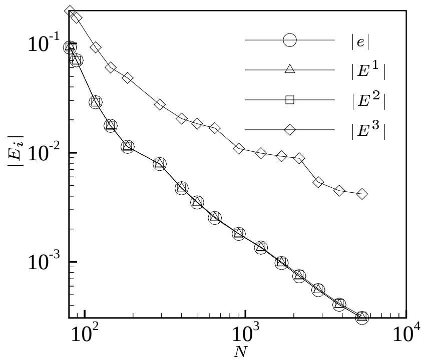
[6] M. Giles, M.G. Larson, J.M. Levenstam, and E. Süli. Adaptive error control for finite element approximations of the lift and drag coefficients in viscous flow. preprint NA-97/06, Comlab, Oxford University, 1997.

[7] M.G. Larson and K. Samuelsson. A posteriori error estimation for functionals. Preprint, Chalmers Finite Element Center, Chalmers University of Technology, Sweden, 2001.

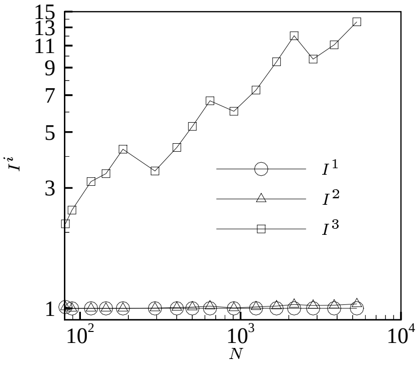
[8] S. Prudhomme and J.T. Oden. On goal-oriented error estimation for elliptic problems: application to the control of pointwise errors. *Comput. Methods Appl. Mech. Engr.*, 159(2):313–331, 1999.

[9] B. Szabo and I. Babuska. *Finite Element Analysis*. John Wiley and Sons, Inc., 1991.

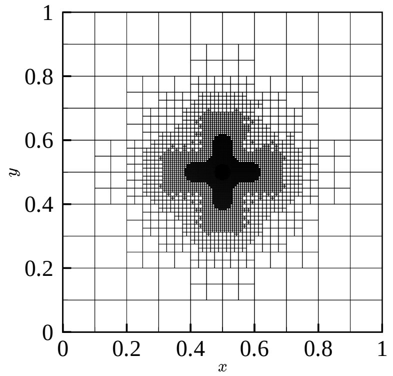
[10] R. Verfürth. *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*. Wiley and Teubner, 1996.



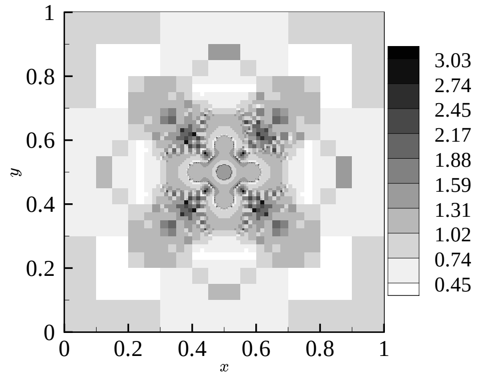
(a) The error.



(b) The efficiency indices.

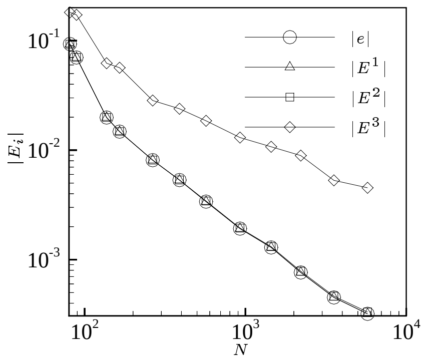


(c) The final mesh.

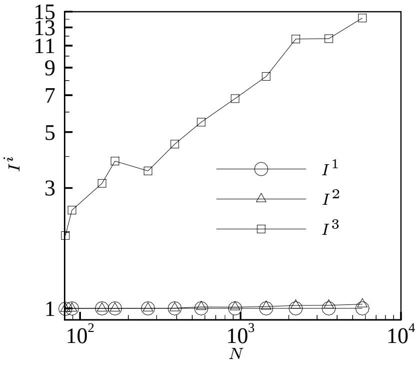


(d) $\log_{10}(E_T^3/E_T^2)$.

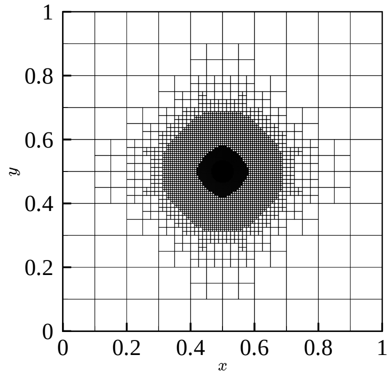
Figure 1: $u = \sin \pi x \sin \pi y e^{-(r/R)^2}$, $r = \sqrt{(x-0.5)^2 + (y-0.5)^2}$, $R = 0.1$. The error is measured in $(x_0, y_0) = (0.5, 0.5)$, $p = 1$, and the adaptive algorithm is based on E_T^2.



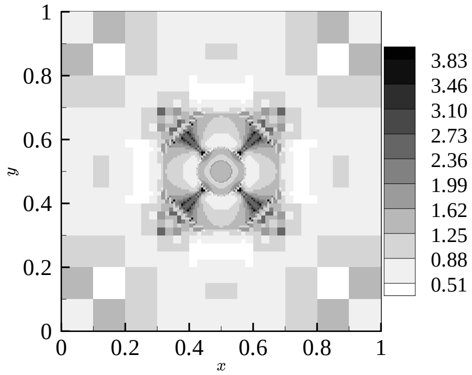
(a) The error.



(b) The efficiency indices.

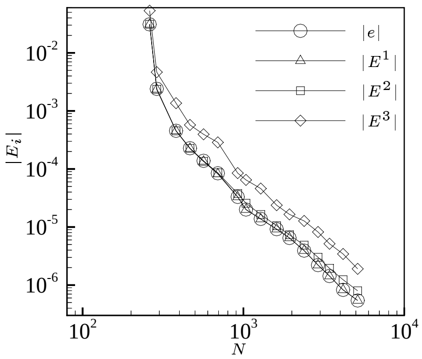


(c) The final mesh.

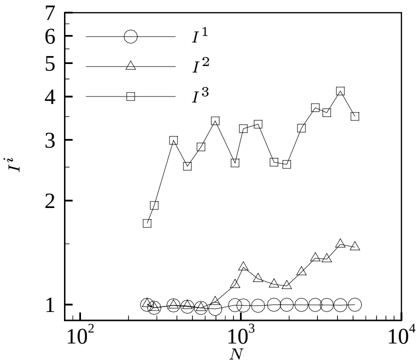


(d) $\log_{10}(E_T^3/E_T^2)$.

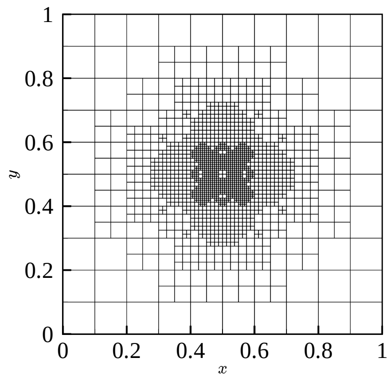
Figure 2: $u = \sin \pi x \sin \pi y e^{-(r/R)^2}$, $r = \sqrt{(x-0.5)^2 + (y-0.5)^2}$, $R = 0.1$. The error is measured in $(x_0, y_0) = (0.5, 0.5)$, $p = 1$, and the adaptive algorithm is based on E_T^3.



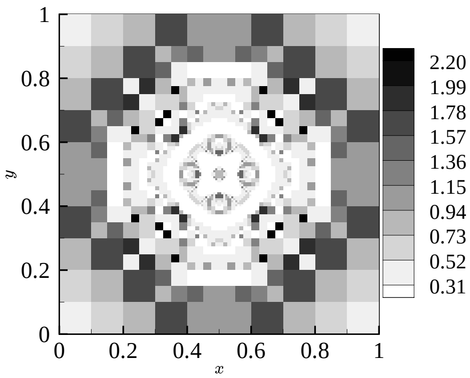
(a) The error.



(b) The efficiency indices.

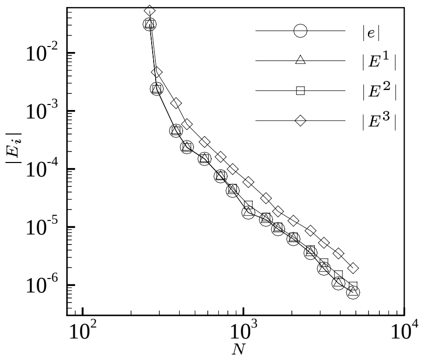


(c) The final mesh.

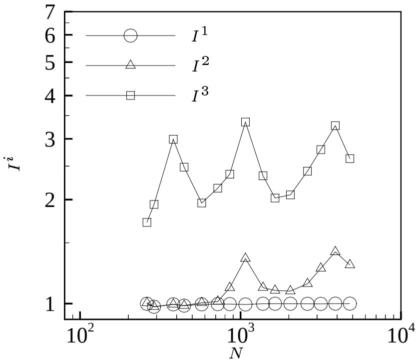


(d) $\log_{10}(E_T^3/E_T^2)$.

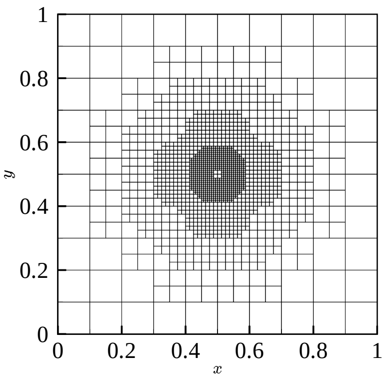
Figure 3: $u = \sin \pi x \sin \pi y e^{-(r/R)^2}$, $r = \sqrt{(x-0.5)^2 + (y-0.5)^2}$, $R = 0.1$. The error is measured in $(x_0, y_0) = (0.5, 0.5)$, $p = 2$, and the adaptive algorithm is based on E_T^2.



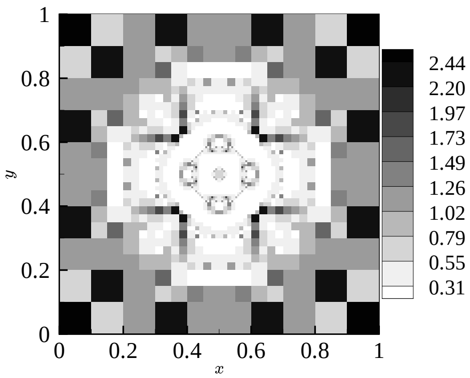
(a) The error.



(b) The efficiency indices.

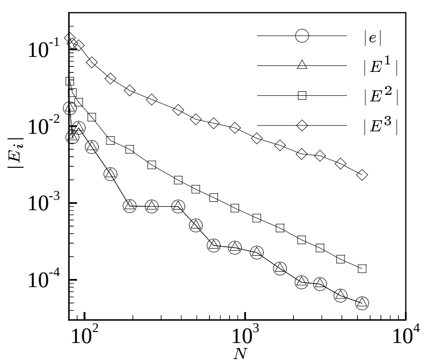


(c) The final mesh.

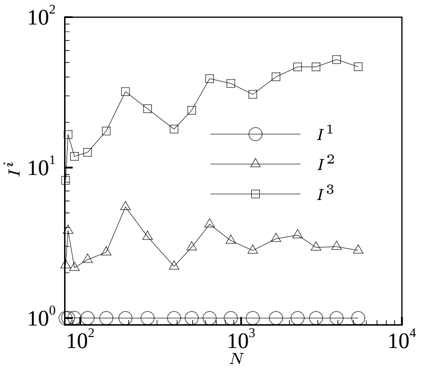


(d) $\log_{10}(E_T^3/E_T^2)$.

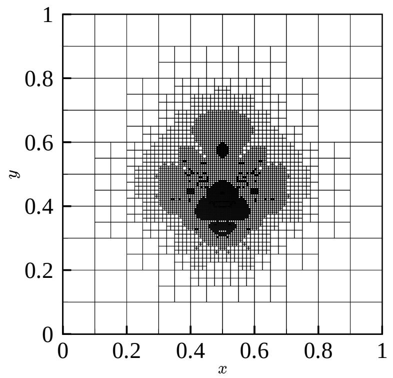
Figure 4: $u = \sin \pi x \sin \pi y e^{-(r/R)^2}$, $r = \sqrt{(x-0.5)^2 + (y-0.5)^2}$, $R = 0.1$. The error is measured in $(x_0, y_0) = (0.5, 0.5)$, $p = 2$, and the adaptive algorithm is based on estimate E_T^3.



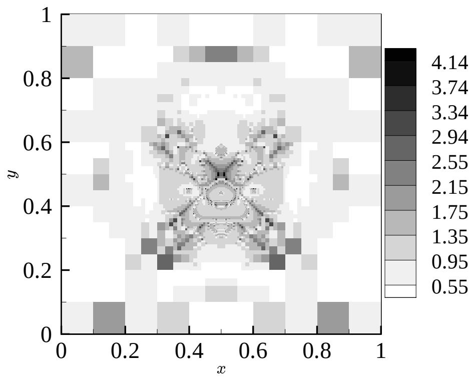
(a) The error.



(b) The efficiency indices.

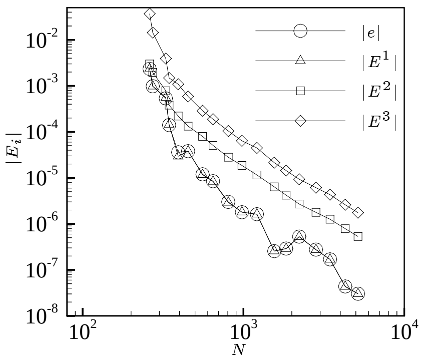


(c) The final mesh.

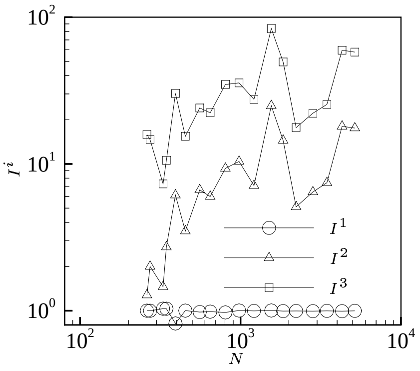


(d) $\log_{10}(E_T^3/E_T^2)$.

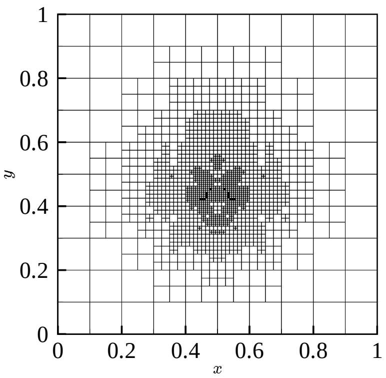
Figure 5: $u = \sin \pi x \sin \pi y e^{-(r/R)^2}$, $r = \sqrt{(x-0.5)^2 + (y-0.5)^2}$, $R = 0.1$. The error is measured in $(x_0, y_0) = (0.5, 0.4)$, $p = 1$, and the adaptive algorithm is based on E_T^2.



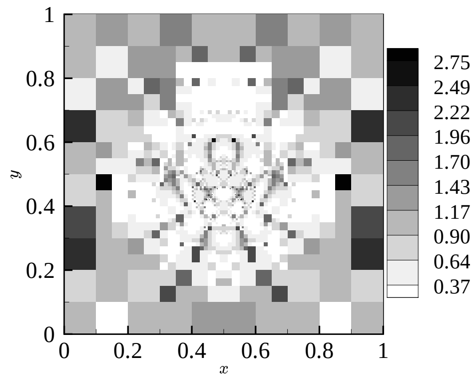
(a) The error.



(b) The efficiency indices.

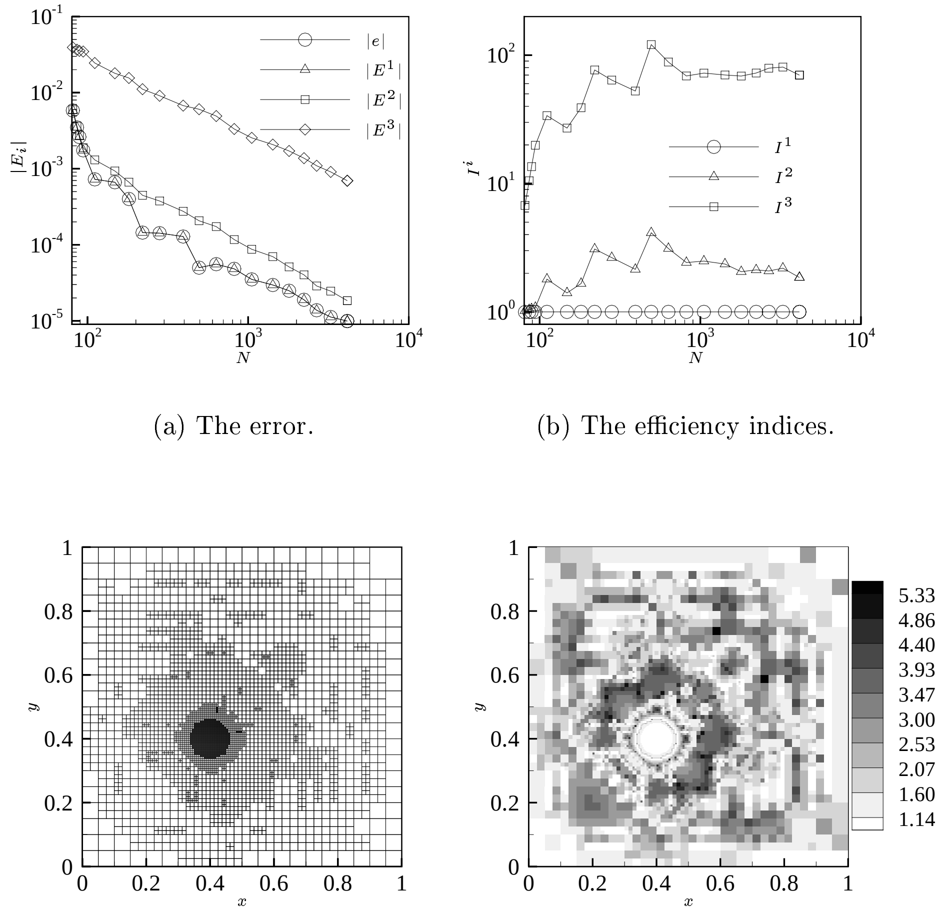


(c) The final mesh.



(d) $\log_{10}(E_T^3/E_T^2)$.

Figure 6: $u = \sin \pi x \sin \pi y e^{-(r/R)^2}$, $r = \sqrt{(x-0.5)^2 + (y-0.5)^2}$, $R = 0.1$. The error is measured in $(x_0, y_0) = (0.5, 0.4)$, $p = 2$, and the adaptive algorithm is based on E_T^2.



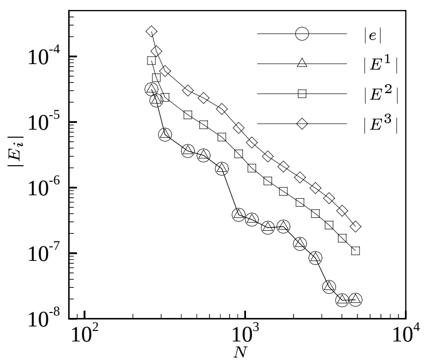
(a) The error.

(b) The efficiency indices.

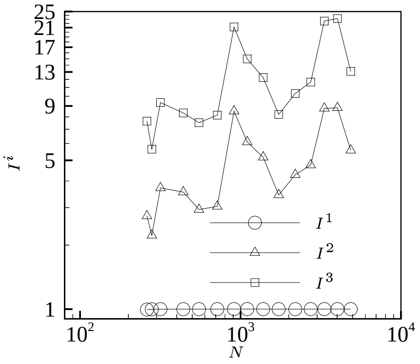
(c) The final mesh.

(d) $\log_{10}(E_T^3/E_T^2)$.

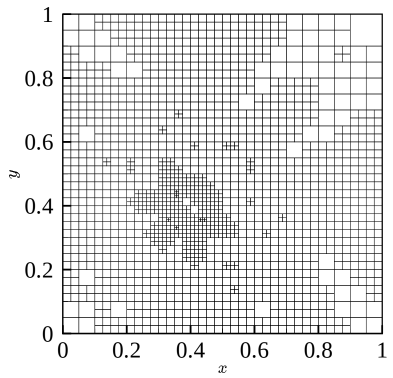
Figure 7: $u = \sin \pi x \sin \pi y$. The error is measured in $(x_0, y_0) = (0.4, 0.4)$, $p = 1$, and the adaptive algorithm is based on E_T^2.



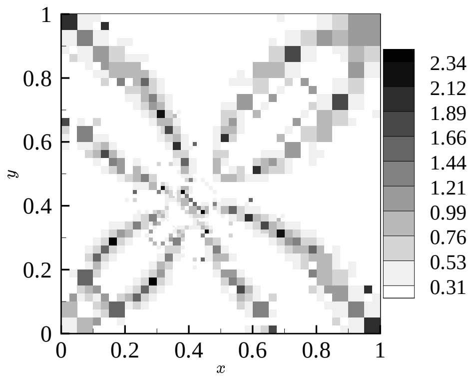
(a) The error.



(b) The efficiency indices.

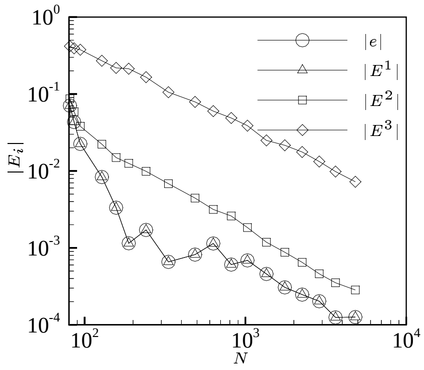


(c) The final mesh.

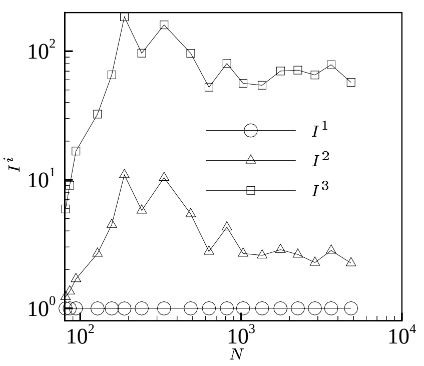


(d) $\log_{10}(E_T^3/E_T^2)$.

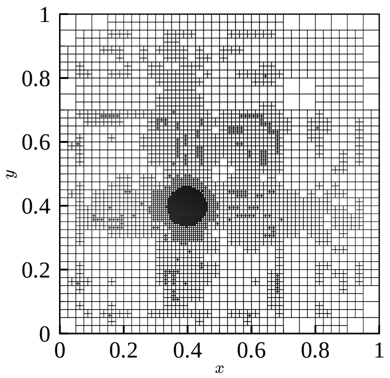
Figure 8: $u = \sin \pi x \sin \pi y$. The error is measured in $(x_0, y_0) = (0.4, 0.4)$, $p = 2$, and the adaptive algorithm is based on E_T^2.



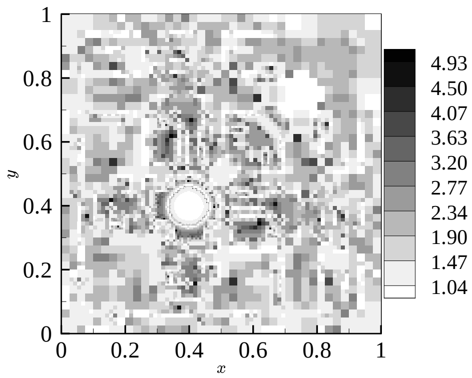
(a) The error.



(b) The efficiency indices.

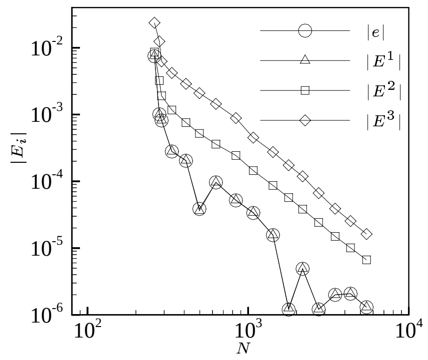


(c) The final mesh.

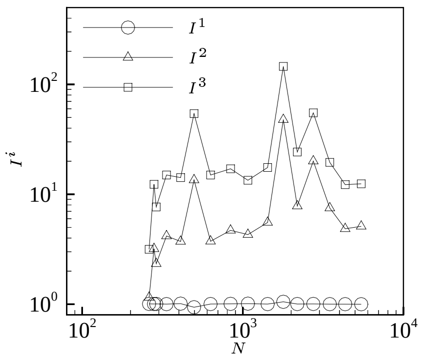


(d) $\log_{10}(E_T^3/E_T^2)$.

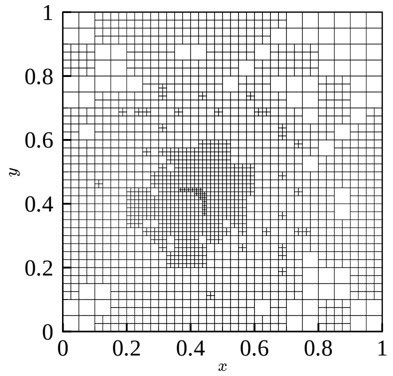
Figure 9: $u = \sin 4\pi x \sin 4\pi y$. The error is measured in $(x_0, y_0) = (0.4, 0.4)$, $p = 1$, and the adaptive algorithm is based on E_T^2.



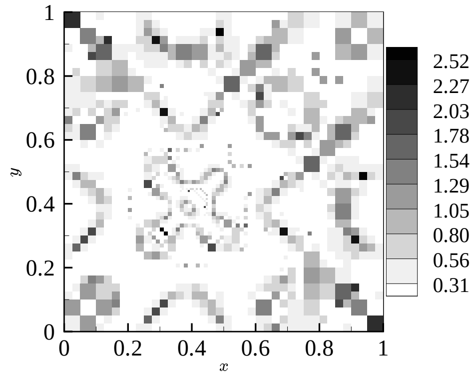
(a) The error.



(b) The efficiency indices.



(c) The final mesh.



(d) $\log_{10}(E_T^3/E_T^2)$.

Figure 10: $u = \sin 4\pi x \sin 4\pi y$. The error is measured in $(x_0, y_0) = (0.4, 0.4)$, $p = 2$, and the adaptive algorithm is based on E_T^2.

Chalmers Finite Element Center Preprints

2000–01 *Adaptive Finite Element Methods for the Unsteady Maxwell's Equations*
Johan Hoffman

2000–02 *A Multi-Adaptive ODE-Solver*
Anders Logg

2000–03 *Multi-Adaptive Error Control for ODEs*
Anders Logg

2000–04 *Dynamic Computational Subgrid Modeling* (Licentiate Thesis)
Johan Hoffman

2000–05 *Least-Squares Finite Element Methods for Electromagnetic Applications* (Licentiate Thesis)
Rickard Bergström

2000–06 *Discontinuous Galerkin Methods for Incompressible and Nearly Incompressible Elasticity by Nitsche's Method*
Peter Hansbo and Mats G. Larson

2000–07 *A Discountinuous Galerkin Method for the Plate Equation*
Peter Hansbo and Mats G. Larson

2000–08 *Conservation Properties for the Continuous and Discontinuous Galerkin Methods*
Mats G. Larson and A. Jonas Niklasson

2000–09 *Discontinuous Galerkin and the Crouzeix-Raviart element: Application to elasticity*
Peter Hansbo and Mats G. Larson

2000–10 *Pointwise A Posteriori Error Analysis for an Adaptive Penalty Finite Element Method for the Obstacle Problem*
Donald A. French, Stig Larson and Ricardo H. Nochetto

2000–11 *Global and Localised A Posteriori Error Analysis in the Maximum Norm for Finite Element Approximations of a Convection-Diffusion Problem*
Mats Boman

2000–12 *A Posteriori Error Analysis in the Maximum Norm for a Penalty Finite Element Method for the Time-Dependent Obstacle Problem*
Mats Boman

2000–13 *A Posteriori Error Analysis in the Maximum Norm for Finite Element Approximations of a Time-Dependent Convection-Diffusion Problem*
Mats Boman

2001–01 *A Simple Nonconforming Bilinear Element for the Elasticity Problem*
Peter Hansbo and Mats G. Larson

2001–02 *The $\mathcal{LL}^*$ Finite Element Method and Multigrid for the Magnetostatic Problem*
Rickard Bergström, Mats G. Larson, and Klas Samuelsson

2001–03 *The Fokker-Planck Operator as an Asymptotic Limit in Anisotropic Media*
Mohammad Asadzadeh

2001–04 *A Posteriori Error Estimation of Functionals in Elliptic Problems: Experiments*
Mats G. Larson and A. Jonas Niklasson

These preprints can be obtained from

`www.phi.chalmers.se/preprints`