Numerical analysis of depenence between adapted mesh and assumed error indicator

Jan Kucwaj
Institute for Informatics, Department of Physics, Mathematics and Informatics,
Cracow University of Technology, Warszawska 24, 31-155 Cracow, Poland,
e-mail: jkucwaj@pk.edu.pl,

Abstract

The paper consiedrs the influence of the assumed error indicator on the final adapted mesh. Provided that error threshold values of error are increased by applying the adaptive procedure it turns that final mesh depends on the assumed error indicator. In the paper there were used standard error estimates and the error indicator proposed by the author. The proposed error indicator is based on applying hierachically generalized finite diferrence method (FDM). In case of the proposed eror indicator the final one is more "suitable" to the strict solution.

Keywords: mesh generation, mesh size function, mesh adaptation, remeshing, error indicator

1. Introduction

The subject of the paper is an analysis of the impact of final adapted mesh on the error indicator assumed by the author. The applied method is of remeshing type and for every adaptation step a completely new mesh is generated. For mesh generation the computer code GRADMESH [3, 4] was applied. The generator uses mesh size function [6] defined in the whole domain including the boundary.

For the sake of the numerical simulations the infinite space is approximated by a finite dimensional space generated by a given set of basis functions, the approximated solution to the problem is equal to a linear combination of the basis functions. The coeffiecients of the linear combination are found from the solution of a nonliner algebraic system of equations. The system is carried out from stationarity conditions. The system of nonlinear algebraic equations is solved by the Newton-Raphson method.

In the case of standard error indicators [2, 8] the values of erros at elements are led to nodes as weighted averages taken at elements. In consequtive steps of the remeshing algorithm the values of the mesh size function at the nodes are modified in the way that at the points with greatest values of the error indicators the values of mesh size function are the most reduced. Having the values of the mesh size function at nodes the new mesh size function is defined in the whole computational domain by the linear interpolation. The process is performed till the error indicator attains assumed value. In case of the error indicator proposed by the author values of the errors are found directly at every node by applying the finite difference method in the hierarchical way.

2. Error indicators

The applied indicators [1] are calculated directly for every node, not in elements like in [5, 7]:
Let e_i for $i = 1, \ldots, n_\nu$ be an error indicator at $i\text{-}th$ apex of the mesh $\mathcal{T}_\nu$, and $\mathcal{P}_\nu$={ $P_i,\ i = 1\ ,\ldots, n_P$} $-$ set of nodes. We define a set of numbers of a patch of elements for every node P_i as:

$$L_i = \{k : node P_i \in \overline{T}_k\} \text{ for } i = 1, \ldots, n_P \text{ (number of points).} \quad (1)$$

1. The first proposed error indicator is based on the discretized form of the considered equation. At every node partial derivatives are found $\frac{\partial u_h}{\partial x}, \frac{\partial u_h}{\partial y}, \frac{\partial^2 u_h}{\partial x^2}, \ldots$ according to the folowing recipe:
Having found $u_h(P_i)$ for $i = 1, \ldots, N_P$, the hierachical formula is applied:

$$\frac{\partial u_h}{\partial x}(P_i) = \frac{\sum_{k\in L_i} \frac{\partial u_h^k}{\partial x}(\mathbf{P}_i)\text{area}(T_k)}{\sum_{k\in L_i} \text{area}(T_k)} \quad (2)$$

where u_h^k is the restriction of the approximate solution to the k-th element. As the restriction of u_h^k of the solution u_h to the k-th element is a linear combination of shape functions of the k-th element, then:

$$u_h^k = \sum_{j=0}^{n_e} \lambda_j N_j^k, \quad \text{what gives} \quad \frac{\partial u_h^k}{\partial x} = \sum_{j=0}^{n_e} \lambda_j \frac{\partial(N_j^k)}{\partial x} \quad (3)$$

where N_j^k is a shape function of the k-th element and λ_j are the the coefficient of the linear combination with n_e degrees of fredom of the e-th element. Formula (3) is applied at nodal points. The derivatives found in that way $\frac{\partial u_h}{\partial x}(P_i),\ i = 1, \ldots N_P$ are used for calculation of the second order derivatives at the nodes in the similar way by using the recurrent formulas:

$$\frac{\partial^2 u_h}{\partial x^2}(P_i) = \frac{\partial}{\partial x}(\frac{\partial u_h}{\partial x})(P_i) \quad (4)$$

2. In the case of the below error indicator it is suggested to evaluate directlty the values of the derivatives of the error indicator at every node of the mesh (in the case of first order approximtion) in the following way:

$$e_i = \sqrt{\sum_{k\in L_i,\ l\in L_i,\ l\neq k} (\frac{\partial u_i}{\partial x} - \frac{\partial u_k}{\partial x})^2 + (\frac{\partial u_i}{\partial y} - \frac{\partial u_k}{\partial y})^2}, \quad (5)$$

where L_i is the set of numbers of elements meeting at $i\text{-}th$ node.

3. The algorithm of remeshing based on grid generator

The algorithm of remeshing can be divided into the following steps:

1. Preparation of the information about the geometry and boundary conditions of the problem,
2. Fixing an initial mesh size function,
3. Mesh generation with the mesh size function,
4. Solution to the considered problem on the previously generated mesh,
5. Evaluation of error indicator in every element,
6. Calculation of nodal error indicator values by using average method,
7. Definition of the new mesh size function using the errors computed at every point,
8. If error not satisfactory go to point 3.,
9. End of computations.

The applied indicators are calculated for every element or directly at the nodes [3, 4].
Let e_i for $i = 1, \ldots, n_0$ be an error indicator at i-th apex of the initial mesh $\mathcal{T}_0$, with $\mathcal{P}_0$={ P_i, $i = 1$,. . . , $n_P\}$ − set of nodes.

The modification of the mesh size function is performed at every adaptation step to perform the upcoming one. The main idea of this part of the algorithm relies on reduction of the values of the mesh size function by an appropriately chosen function. The chosen function is continuous, linear and its smallest value is at the node of the maximum error indicator and the greatest value where the error is minimum. The function increases due to the error decrement.

4. Modification of the mesh size function

Having got the values of the erors at the nodes of the mesh on which the considered approximate $\tilde{e}_i$ solution is obtained we define:

$$\alpha = \min_{k=1,2,\ldots,N_{NOD}} \tilde{e}_k, \qquad \beta = \max_{k=1,2,\ldots,N_{NOD}} \tilde{e}_k, \tag{6}$$

where N_{NOD} is the number of nodes.

Then the folowing values are introduced:

λ − a value indicating the greatest mesh size function reduction,

μ − a value indicating the smallest mesh size function reduction.

The following transformation is defined

$$l : [\alpha, \beta] \mapsto [\mu, \lambda] \tag{7}$$

which satisfies the conditions: $l(\alpha) = \lambda$ and $l(\beta) = \mu$. Provided that

$$Q_i = l(\tilde{e}_i) \text{ for } i = 1, \ldots, N_{NOD}, \tag{8}$$

then one has: $\min_{i=1,2,\ldots,N_{NOD}} Q_i = \mu$,
$\max_{i=1,2,\ldots,N_{NOD}} Q_i = \lambda$.

Introducing the function $r : \overline{D} \mapsto I\!R$ as follows: $r(\overline{x}) = \Pi(\overline{x})$, if $\overline{x} \in \overline{T}_s$, where Π is an affine mapping of two variables, satisfying the following equalities:

$$\Pi(P_i) = Q_i \text{ for } i = 1, 2, 3, \tag{9}$$

where P_1, P_2, P_3 are the vertices of the triangle T_s of the triangulation of Ω, and appropriately Q_1, Q_2, Q_3 are the values defined by the formula (8). A bar means closure of the domain. The function $r(\overline{x})$ is defined in the whole domain because the triangles $\{\overline{T}_s\}_{s=1}^{n_e}$ cover it. The new mesh size function is defined as follows:

$$\gamma_{i+1}(\overline{x}) = \gamma_i(\overline{x}) r(\overline{x}). \tag{10}$$

4.1. *Numerical Examples*

Final mesh for example problem with known solution: $u(x, y) = x(1 - x)y(1 - y) \arctan(a(\frac{x+y}{\sqrt{2}} - \xi))$, with parameters a, ξ for the Laplace equation.

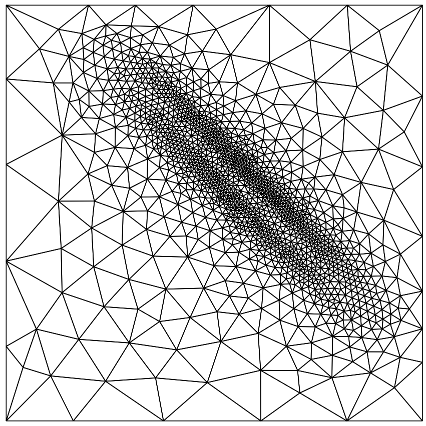

Figure 1: The final adapted mesh for the examplary problem.

5. Conlusions

Numerical experiments show that the final mesh generated in adaptation process obtained by using error estimates based on finite difference method is more optimal and better matches to the exact solution.

References

[1] Ainsworth, M. and Oden, J.T., A posteriori error estimation in finite element analysis, *Comput. Methods Appl. Mech. Engng*, 142, 1-88, 1997.

[2] Huerta A., Diez P., Error estimation including pollution assessment for nonlinear finite element analysis, *Comp. Meth. Appl. Mech. Engng*, 181, 21-41, 2000.

[3] Kucwaj J., The Algorithm of Adaptation by Using Graded Meshes Generator, *Computer Assisted Mechanics and Engineering Sciences*, 7, 615-624, 2000.

[4] J. Kucwaj, The analysis of the effieciency of an adaptation method based on the grid generator, Proceedings of the WCCM XI, ECCM V and ECFD VI Barcelona 2014 : 11 World Congress on Computational Mechanics (WCCM XI), 20-25.07.2014, 2624-2636.

[5] Kucwaj J., The influence of different equivalent boundary conditions on approximate solution to a potential problem, vol. 23, 69-81, 2016.

[6] Thompson J.F., Soni B.K., Weatherwill N.P., *Handbook of Grid Generation,* CRC Press, Boca Raton, 1999.

[7] Lo S.H., Finite element mesh generation and adaptive meshing , *Progress in Structural Engineering and Materials*, 4, 381-399, 2002.

[8] Desheng Wang, Oubay Hassan, Kenneth Morgan, Nigel Weatherill, Enhanced remeshing from STL files with applications to surface grid generation, *Comm. in Num. Meth. in Engrg.*, 65, 734-751, 2006.