INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2001; **51**:253–273

A combined *rh*-adaptive scheme based on domain subdivision. Formulation and linear examples

Harm Askes*,[1],† and Antonio Rodríguez-Ferran[2],‡

[1] *Faculty of Civil Engineering and Geosciences, Koiter Institute Delft/Delft University of Technology, P.O. Box 5048, 2600 GA Delft, The Netherlands*
[2] *Departament de Matemàtica Aplicada III, Universitat Politècnica de Catalunya, E.T.S. d'Enginyers de Camins, Canals i Ports, Jordi Girona 1 i 3, 08304 Barcelona, Spain*

SUMMARY

An adaptive scheme is proposed in which the domain is split into two subdomains. One subdomain consists of regions where the discretization is refined with an *h*-adaptive approach, whereas in the other subdomain node relocation or *r*-adaptivity is used. Through this subdivision the advantageous properties of both remeshing strategies (accuracy and low computer costs, respectively) can be exploited in greater depth. The subdivision of the domain is based on the formulation of a desired element size, which renders the approach suitable for coupling with various error assessment tools. Two-dimensional linear examples where the analytical solution is known illustrate the approach. It is shown that the combined *rh*-adaptive approach is superior to its components *r*- and *h*-adaptivity, in that higher accuracies can be obtained compared to a purely *r*-adaptive approach, while the computational costs are lower than that of a purely *h*-adaptive approach. As such, a more flexible formulation of adaptive strategies is given, in which the relative importance of attaining a pre-set accuracy and speeding-up the computational process can be set by the user. Copyright © 2001 John Wiley & Sons, Ltd.

KEY WORDS: mesh adaptivity; remeshing; Arbitrary Lagrangian–Eulerian; *r*-adaptivity; *h*-adaptivity; *rh*-adaptivity

1. INTRODUCTION

Mesh-adaptive strategies are a necessary tool to make (non-linear) finite element analysis applicable to engineering practice [1–5]. Without mesh-adaptive strategies the quality of the finite element solution cannot be assessed objectively. Moreover, mesh-adaptive strategies are indispensable to limit the computer costs needed to obtain finite element solutions for the large-scale structures of engineering practice. In this paper, we distinguish between two goals for using mesh adaptivity in finite element analysis.

*Correspondence to: Harm Askes, Faculty of Civil Engineering and Geosciences, Koiter Institute Delft/Delft University of Technology, P.O. Box 5048, 2600 GA Delft, The Netherlands
†E-mail: h.askes@citg.tudelft.nl
‡E-mail: antonio.rodriguez-ferran@upc.es

Received 25 March 1999
Revised 18 January 2000

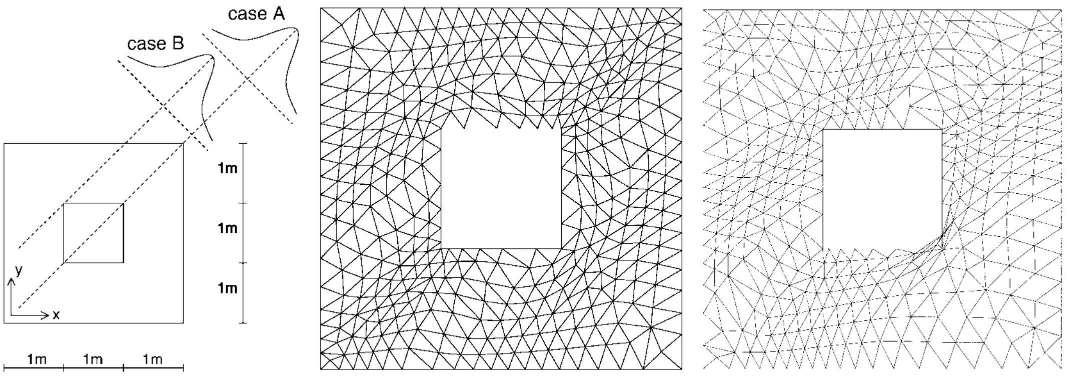


Figure 1. Square with a hole—geometry and temperature fields (left), r-adapted mesh for case A (centre) and r-adapted mesh for case B (right).

Firstly, adaptive strategies can be used to *attain a prescribed accuracy*. This goal is best met with an h-, p- or hp-adaptive scheme in combination with a sound error estimator. These schemes enable the addition and deletion of degrees of freedom, and the mesh connectivity is allowed to change. As a consequence, these schemes offer topological flexibility, but the expenses in terms of computer time associated to these adaptive techniques are relatively high [2].

Secondly, adaptive strategies may be used in order to *limit the computer costs*. For large-scale three-dimensional analyses mesh optimization can reduce the computer time and memory requirements of an analysis by several orders of magnitude. For instance, a significant *speed-up of the computational process* can be obtained by an r-adaptive scheme such as the Arbitrary Lagrangian–Eulerian (ALE) method [6, 7]. The constant number of degrees of freedom and the constant element connectivity make these methods inexpensive. Since the computational costs involved with ALE remeshing can be made as low as $\mathcal{O}(N)$, with N the number of finite elements, the overhead of r-adaptive remeshing included in a finite element implementation can be made negligible [8, 9]. However, the fixed number of degrees of freedom and mesh connectivity limit the applicability of r-adaptive remeshing. This is illustrated in Figure 1, where r-adaptivity is used to optimize a finite element discretization of a thermal problem. (Details of this problem can be found in Section 5.) The solution for the temperature follows a bell-shaped profile along the main diagonal from upper left to lower right. If the bell-shape is centred at the main diagonal from lower left to upper right (cf. case A in Figure 1, left), then r-adaptivity is an effective tool to improve the discretization (Figure 1, centre). On the other hand, if the centre of the bell-shape is shifted upwards (cf. case B in Figure 1, left), the non-convex corners of the hole lead to element entanglement when r-adaptivity is used (Figure 1, right). Thus, the applicability of r-adaptivity is limited.

The properties and the application fields of both types of adaptive schemes (sophisticated, expensive schemes versus inexpensive schemes of limited applicability) are thus complementary. This has led to the idea that the advantageous properties of both schemes could be combined. Each scheme should be used where and when it is suited most. Combinations of h- and r-adaptivity have already been reported for the boundary element method in two spatial dimensions [10–12]. The focus of this paper will be on combined *rh*-adaptive schemes for finite element analysis.

The approach followed is to decompose the domain into two subdomains. The first subdomain consists of all regions in the domain where the element sizes are close (to a certain extent) to the desired element sizes. In these regions, node relocation (*r*-adaptivity) based on the ALE technique is carried out. The second subdomain contains the remainder of the domain, where element sizes differ significantly from the desired element sizes. Here, *h*-adaptive mesh refinement or mesh de-refinement is applied. The subdivision of the domain is based on the concept of a *desired element size*, which is a common ingredient in most error assessment procedures. Thus, the approach can be generalized to various fields of application. A numerical parameter can be provided by the user which sets the bounds of the criterion with which an element is assigned to the *r*-adaptive subdomain or to the *h*-adaptive subdomain. When this parameter favours larger *r*-adaptive subdomains, consequently more emphasis is put on computational speed-up and less on numerical accuracy. Conversely, other values for this parameter lead to a larger relative importance of numerical accuracy as compared to computational speed-up. Thus, the user can control the relative importance of attaining a prescribed accuracy and speeding-up the computational process, and a more flexible formulation of adaptive strategies is obtained.

In the present study, only linear examples are treated where the exact error is known. If the approach is to be extended to other application fields, then error estimators and error indicators must be included [2]. However, the complete proposed algorithm is based on the formulation of a desired element size, by which extension to other error assessment tools is straightforward. In case of a non-linear analysis also an algorithm for the transfer of the state variables must be provided [8, 13]. The focus of this paper is on the subdivision of the domain and the construction of a new discretization. Application of error estimators and error indicators and the inclusion of a transfer algorithm will be dealt with in a forthcoming contribution.

The present paper is organised as follows. First, error assessment is discussed, as well as how an optimal element size can be derived. Next, the subdivision of the domain into an *h*-adaptive subdomain and an *r*-adaptive subdomain is treated. This subdivision is based on error assessment. Then, the generation of a new discretization for the two subdomains is treated. Two-dimensional examples are presented to illustrate the approach. Open questions and future developments are addressed next.

2. ERROR ASSESSMENT AND OPTIMAL ELEMENT SIZES

The basis of any adaptive strategy is error assessment. It must be known in which regions of the domain the discretization is fine enough, in which regions refinement is needed, and in which regions de-refinement can be accepted. In general, three methods of assessing the error exist, where we use the classification of Huerta and coworkers [2]:

- For a small subclass of (academic) problems, the exact solution is known. Then, it is possible to compute the *exact error* as the difference of the exact solution and the approximate (numerical) solution. Although the applicability of this type of error assessment is limited, its merits are that it supplies a general tool for the validation of other error assessment approaches and adaptive strategies.

- The so-called *error estimators* approximate the exact error. Since they provide *quantitative* information about the exact error, error estimators can be used to reach a prescribed accuracy. Error estimators must have a solid mathematical basis [2, 14]. The computation of error estimators is relatively expensive.
- Finally, *error indicators* only give relative information about the exact error. An error indicator is based on heuristic considerations and denotes where the error is large and where it is small, but not *how* large or small. A sound mathematical basis is lacking, and no quantitative measure of the exact error is provided. However, an error indicator is cheap in terms of computational costs. Error indicators can be based on variations of the solution, strain projection norms, element distortion and jump or average of state variables, e.g., each of which are readily available in a finite element analysis. Contrary to error estimators, error indicators do not necessarily have an upper bound.

Once an error tolerance is set, an *optimality criterion* can be used to translate the (exact or estimated) error distribution into a field of desired element size. If an error indicator is used, the user has to specify a coupling between error indicator and desired element size. In this paper, only exact errors are used, therefore a mathematically based *optimality criterion* will be used.

Several methods exist to relate the exact error or the estimated error to an element size that is expected to meet the accuracy requirements [15–19]. It has been shown that the optimality criterion proposed in References [15, 16] leads to meshes with the lowest number of elements. The desired size h^{des} for element i is related to the current size h^{cur} of element i according to [15, 16, 19]

$$h_i^{\mathrm{des}} = h_i^{\mathrm{cur}} \left(\frac{\eta \|u\|}{\sqrt{N^{\mathrm{des}}}\|e\|_i} \right)^{1/(p+d/2)} \tag{1}$$

where h is a characteristic element size such as the diameter, η is the prescribed relative error (prescribed accuracy), p is the maximum degree of the complete interpolation polynomials, d is the number of spatial dimensions, $\|u\|$ and $\|e\|_i$ are the energy norm of the solution field u for the whole domain and the local error e for element i, respectively, defined via

$$\|a\|_i^2 = \int_{\Omega_i} \nabla a C \nabla a \, \mathrm{d}\Omega \quad \text{and} \quad \|a\|^2 = \sum_{i=1}^{N^{\mathrm{cur}}} \|a\|_i^2 \tag{2}$$

with C the discretised Jacobian of the problem (the tangent stiffness matrix) and N^{cur} the current number of elements. In Equation (2) it is assumed that C is positive definite. It can be seen from Equation (1) that a larger error tolerance η leads to larger values for the desired element size. On the other hand, a larger local error $\|e\|_i^2$ leads to a smaller desired element size. In Equation (1), N^{des} is the desired number of elements, given by [15]

$$N^{\mathrm{des}} = \left(\sum_{i=1}^{N^{\mathrm{cur}}} \left(\frac{\|e\|_i}{\eta\|u\|} \right)^{d/(p+d/2)} \right)^{1+d/2p} \tag{3}$$

Note that the desired number of elements increases with a decreasing error tolerance and with increasing local errors, while it decreases for increasing values of the polynomial degree p. A closer investigation of Equations (1) and (3) reveals that the desired element size decreases with increasing values for p.

Remark 1. Equation (1) relies on the assumption that the order of convergence of the finite element approximation coincides with the polynomial degree p. It is well known that this does not hold near a singularity [20], where convergence is controlled by the strength of the singularity. However, as pointed out in Reference [21], the influence of the singularity on the solution decreases during the adaptive refinement. For this reason, Equation (1) is used all over the domain to keep the remeshing process as simple as possible.

3. SUBDIVISION OF THE DOMAIN

When the desired element sizes have been computed, basically three types of regions in the computational domain can be distinguished:

- Regions where the current element size does not differ too much from the desired element size.
- Regions where the current element size is significantly smaller than the desired element size.
- Regions where the current element size is significantly larger than the desired element size.

We will define the ratio of current element size over desired element size as the *R*efinement *R*atio:

$$RR = \frac{\text{current element size}}{\text{desired element size}} \tag{4}$$

where current element size and desired element size are taken from Equation (1). The refinement ratio denotes whether an element has a size which is close to optimal ($RR \approx 1$), too small ($RR < 1$) or too large ($RR > 1$). In the remainder of this paper the following idea is applied: if the current element size is more or less optimal ($RR \approx 1$), then r-adaptivity will be applied. If the current element size differs significantly from the desired element size, then locally a new discretization will be constructed to meet the desired element sizes, which is basically an h-adaptive approach. We adopt

$$\text{if } 0 < RR < 1/\varepsilon \quad \text{allow } h\text{-de-refinement} \tag{5}$$

$$\text{if } 1/\varepsilon \leqslant RR \leqslant \varepsilon \quad \text{relocate the nodes} \tag{6}$$

$$\text{if } \varepsilon < RR < \infty \quad \text{apply } h\text{-refinement} \tag{7}$$

with $\varepsilon > 1$ a scalar parameter that can be provided by the user. Based on the distribution of the refinement ratio, the domain can be subdivided into h-adaptive regions (the h-adaptive subdomain) and r-adaptive regions (the r-adaptive subdomain). By assigning specific values to ε, the user can put a relative importance to either subdomain. For instance, taking relatively large values for ε will lead to a relatively large r-adaptive subdomain. As such, the properties of r-adaptivity (significant computational speed-up, limited capabilities to decrease the error) will have a larger impact on the over-all process. On the other hand, values for ε closer to 1

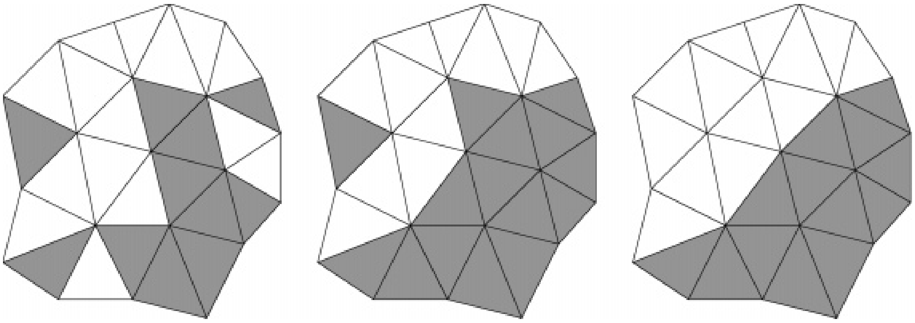


Figure 2. Smoothening of the h-adaptive (gray) and r-adaptive (white) subdomains: initial configuration (left), configuration after smoothening of the h-adaptive subdomain (centre) and configuration after smoothening of the r-adaptive subdomain (right).

will result in larger h-adaptive subdomains, so that computational speed-up is less pronounced but higher accuracies can be attained.

Remark 2. Note that the subdivision of the domain is based on the computed desired element size. As such, it is independent of the error measure that is used.

The subdomains that result from Equations (5)–(7) can yield a scattered pattern of small r-adaptive zones and small h-adaptive zones. Obviously, it is not desirable to assign a single element with $RR>\varepsilon$ to the h-adaptive subdomain if this element is surrounded by elements where $1/\varepsilon\leqslant RR\leqslant\varepsilon$. Therefore, both subdomains are smoothened. For the triangular elements adopted here, this means that each element is assigned to a certain zone when at least two neighbouring elements (i.e. sharing one edge) belong to this zone. To facilitate the adjustment of the number of elements, the h-adaptive subdomain is given preferential treatment in the smoothening procedure. First, r-adaptive elements that have at least two h-adaptive neighbours are added to the h-adaptive subdomain. When this is finished, the process is reversed: each h-adaptive element with at least two r-adaptive neighbours becomes an r-adaptive element. In this manner, more regular subdomains are obtained, see Figure 2 for an illustration of this approach.

4. REMESHING

After the domain has been split, a new discretization must be supplied for each subdomain. Since both subdomains are disjoint, the best available remeshing tools can be applied in each case. In the current study, the interface between the two subdomains remains fixed, i.e. nodes from the intersection between the r-adaptive subdomain and the h-adaptive subdomain are not allowed to move (no r-adaptivity), neither is it allowed that new nodes are added on the interface (no h-adaptivity). This treatment of the interface is a first attempt, and other options are possible.

4.1. *h-adaptive subdomain*

For the *h*-adaptive subdomain we directly apply the desired element sizes that emerge from the Li–Bettess optimality criterion [16] (cf. Equation (1)). This distribution of desired element sizes is used as input for the mesh generation module, by which the *h*-adaptive process is finished. Consequently, the quality of the mesh generator is decisive for a successfull application of *h*-adaptivity. Currently, good mesh generators are widely available for various two-dimensional element types. Three-dimensional mesh generators that are capable of meeting prescribed desired element sizes throughout the domain are becoming also available.

4.2. *r-adaptive subdomain*

In the *r*-adaptive subdomain a different strategy is followed. Since no elements are added to this subdomain, relative information is needed, that is, in which parts of the subdomain nodes should be concentrated, and from which parts of the subdomain nodes can be taken away [2]. The strategy employed here is the equidistribution of a relocation indicator. The relocation indicator, denoted K, takes large values where small elements are desired and vice versa. Equidistribution of this relocation indicator is stated as [22, 23]

$$K_i\Omega_i = K_j\Omega_j \quad \text{or} \quad \int_{\Omega_i} K\,\mathrm{d}\Omega = \int_{\Omega_j} K\,\mathrm{d}\Omega \quad \forall i,j \tag{8}$$

with Ω_i the volume of element i. Equation (8) can be rewritten in a differential format as [23]

$$\frac{\partial}{\partial\chi}\left(K(x)\frac{\partial x}{\partial\chi}\right) = 0 \tag{9}$$

which is used to solve for the nodal positions x. Equation (9) can be repeated for each spatial co-ordinate, so that generalization towards two and three-dimensional problems is straightforward. The co-ordinates χ in Equation (9) denote a reference system with respect to which the node relocation is carried out. This co-ordinate system must be chosen independent of the spatial co-ordinates, and remains fixed throughout the analysis [22–25]. Normally, the initial nodal positions are taken for χ. Then, the best choice for K would be the inverse of the desired element size:

$$K = \frac{1}{\text{desired element size}} \tag{10}$$

It can be seen directly from Equation (8) that equidistribution then leads to an optimal mesh, i.e. where each element takes its desired size, provided that the total number of elements is large enough.

It is also possible to take the current nodal positions x^{cur} instead of χ in Equation (9) if the change of the reference system is accounted for. This is highly advantageous, since then the initial co-ordinates need not be stored. Furthermore, in a combined *rh*-adaptive approach it can be a cumbersome task to assign a referential co-ordinate (which is needed for *r*-adaptivity) to a node that has been added (by means of *h*-adaptivity) in an intermediate stage of the analysis. Transition to current nodal co-ordinates by applying the chain rule to each $\partial/\partial\chi$

term in Equation (9) gives

$$\frac{\partial}{\partial x^{\text{cur}}}\left(K(x)\frac{\partial x}{\partial x^{\text{cur}}}\frac{\partial x^{\text{cur}}}{\partial \chi}\right)\frac{\partial x^{\text{cur}}}{\partial \chi}=0 \tag{11}$$

The factor $\partial x^{\text{cur}}/\partial \chi$ is proportional to the current element size and it is non-zero. If we substitute Equation (10) into Equation (11) and define a new relocation indicator $\tilde{K}$ as

$$\tilde{K}=\frac{\text{current element size}}{\text{desired element size}}=RR \tag{12}$$

then a new equidistribution differential equation is found with a format similar to that of Equation (9):

$$\frac{\partial}{\partial x^{\text{cur}}}\left(RR(x)\frac{\partial x}{\partial x^{\text{cur}}}\right)=0 \tag{13}$$

Note that Equation (13) only contains quantities of the current configuration, and is based on the computation of a desired element size. Thus, generality is preserved.

Remark 3. Equation (13) represents a non-linear system of equations, which has to be solved in the *r*-adaptive subdomain. Explicit algorithms have been devised to solve this system with the order of the computational costs $\mathcal{O}(N)$, where N is the number of elements involved. Thus, the remeshing in the *r*-adaptive subdomain is highly efficient [9, 22].

Remark 4. Equation (9) as well as Equation (13) must be solved together with a set of boundary conditions. These boundary conditions are normally taken as prescribing for all boundary nodes a zero displacement normal to the boundary [24].

Remark 5. Since different stages in the analysis may lead to different subdivisions of the domain, a varying *r*-adaptive subdomain is taken as the basis of solving Equation (13). For a straightforward implementation Equation (13) can also be solved on the complete domain, provided that all nodes of the *h*-adaptive subdomain are considered to be boundary nodes and are, thus, fixed (in all spatial directions).

Remark 6. In Equation (13), no assumptions have been made on the element type that is used. Indeed, the remeshing strategy of the *r*-adaptive subdomain is valid for any element type (in contrast, mesh generators for *h*-adaptive remeshing do depend on the applied element type).

5. EXAMPLES

The algorithm is summarised in Figure 3. This algorithm has been implemented in the object-oriented finite element code Castem 2000 [26]. Linear triangular elements have been taken. Linear thermal problems with known analytical solution have been used to test the algorithm. In Section 5.1, we have taken the subdivision parameter ε constant, while the effect of a varying ε is studied in Section 5.2.

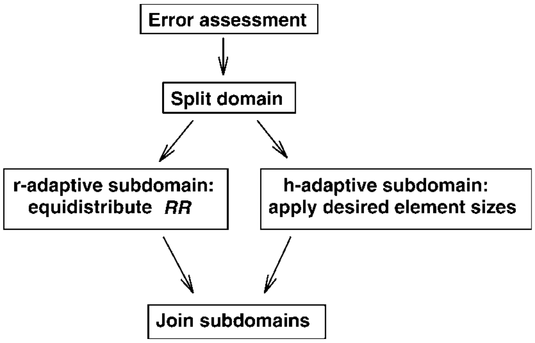

Error assessment
Split domain
r-adaptive subdomain: equidistribute *RR*
h-adaptive subdomain: apply desired element sizes
Join subdomains

Figure 3. Algorithm.

5.1. *Fixed subdivision parameter ε*

The first example deals with a rectangular strip with height $=4\,\text{m}$ and width $=1\,\text{m}$. The conductivity of the material $c=1\,\text{W/mK}$. If the origin of the axes is located at the centre of the strip with the y-axis parallel to the longer side of the specimen, the exact temperature field u_{exact} is expressed as $u_{\text{exact}}=5\exp(-2y^2)$ K. The numerical solution u_{num} follows from the partial differential equation

$$c\nabla^2 u_{\text{num}} = -q \tag{14}$$

with $-q$, a source term which is taken as the second derivative of u_{exact} times c. Essential boundary conditions $\bar{u}$ are imposed as $\bar{u}=u_{\text{exact}}$ on the boundary of the domain. A relative error tolerance $\eta=0.05$ has been prescribed (cf. Equation (1)). As an illustration of the *rh*-adaptive algorithm an analysis has been performed with $\varepsilon=2.5$.

The distribution of the energy norm of the error (cf. Equation (2) with $e=u_{\text{exact}}-u_{\text{num}}$) is depicted in Figure 4a. From this error distribution the refinement ratio has been computed (Figure 4b), after which the domain is split (Figure 4c). A new discretization is constructed based on h-adaptivity in the central part and based on r-adaptivity in the outermost parts (see Figure 4d). The two subdomains are joined and taken as input again for the same analysis. This procedure is repeated four times. The consecutive error distributions are plotted in Figure 5. After the first remeshing step all elements' refinement ratios fall within the interval $[1/\varepsilon,\varepsilon]$, so that only node relocation is performed in the later stages. Thus, the topology of the initial mesh is preserved in the outermost parts of the domain throughout the analysis.

The development of the energy norm of the error $\|e\|$, given by (cf. Equation (2))

$$\|e\|^2=\sum_{i=1}^{N^{\text{cur}}}\|e\|_i^2 \tag{15}$$

is shown in Table I. The table gives the development of the error and the number of elements for different values of ε. In this table, the integer column headers denote the analysis index, i.e. a '0' denotes the initial computation, a '1' denotes the computation after one step remeshing, etc. A full h-adaptive analysis ($\varepsilon=1$) and a full r-adaptive analysis ($\varepsilon=10^2$) have been added for comparison. The full h-adaptive analysis reaches the desired accuracy after the second

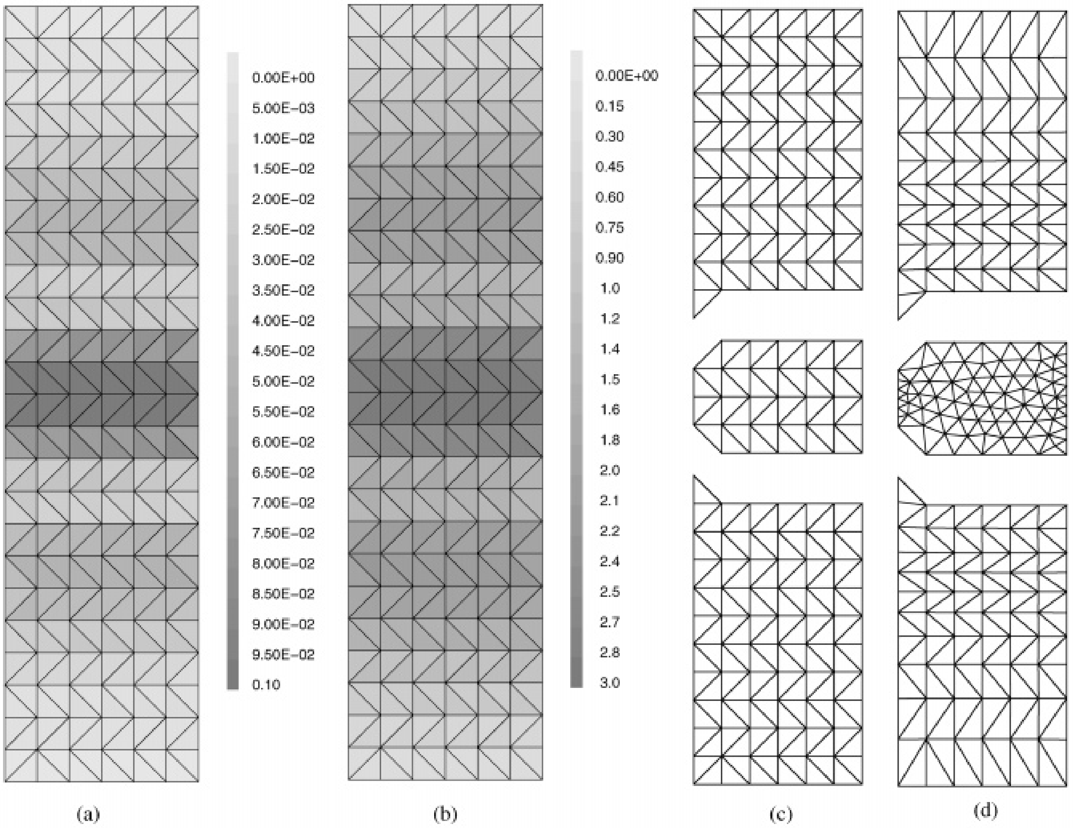


Figure 4. Example of a strip—energy norm of the error (a), refinement ratio (b), splitting the domain (c) and remeshing (d).

remeshing step. The relative errors and numbers of elements are summarized in Figure 6, where the mesh convergence behaviour for different values of ε is depicted.

Examining Table I and Figure 6, it can be concluded that lower values of ε (relatively much *h*-adaptivity, relatively little *r*-adaptivity) lead to higher accuracies. The lower the value of ε, the lower the relative error becomes. For these lower values of ε more elements are added during the analysis. For higher values of ε the effect of *r*-adaptivity is more pronounced, which can be seen from the longer vertical line segments in Figure 6. Although for higher values of ε, the number of elements is invariant during the later stage of computation, still a significant increase in accuracy can be obtained. Indeed, the mesh convergence of the *rh*-approach is higher than that of the pure *h*-adaptive technique, and for the higher values of ε the mesh convergence improves.

Discrepancies from the general tendency (higher accuracy for lower values of ε, higher efficiency for higher values of ε) are caused by a more scattered subdivision of the domain. For instance, the overall performance of $\varepsilon = 2.0$ is worse than that of $\varepsilon = 2.5$. In the first case the *h*-adaptive subdomain and the *r*-adaptive subdomain consist of more parts than for $\varepsilon = 2.5$. Consequently, the interface between the two subdomains has a more complicated

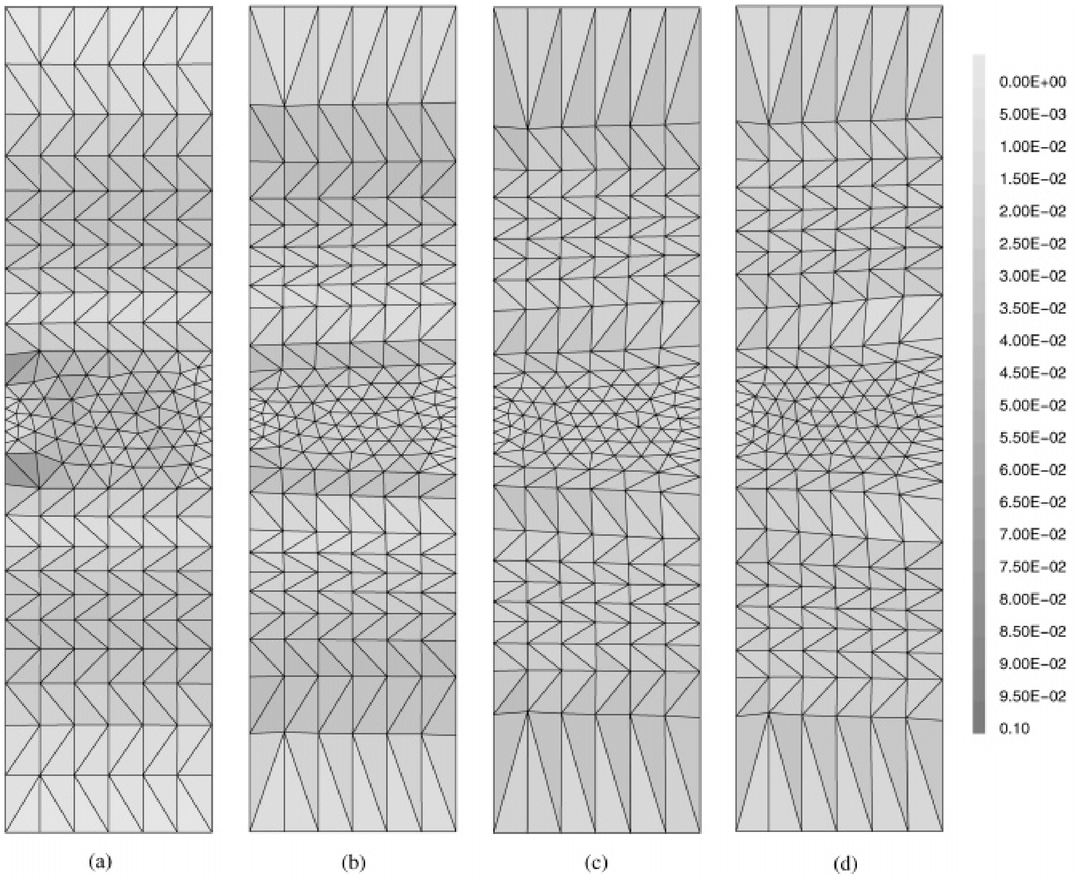


Figure 5. Example of a strip—error distribution after one, two, three and four steps remeshing (a–d, respectively).

form, which poses more constraints on the *h*-adaptive remeshing as well as on the *r*-adaptive remeshing. The same reasoning holds for the analysis with $\varepsilon = 3.0$, where a temporary decrease of accuracy is followed by an increase of accuracy (see Table I). The obtained accuracy can be improved by artificially decreasing the prescribed relative error. Thus, while we still aim at a relative error of 5 per cent, we impose a lower value for η. Figure 7 shows the mesh convergence behaviour for analyses with $\varepsilon = 1.5$ when η is varied. We observe a gradual decrease of the final error, while the mesh convergence behaviour is still better than that of the full *h*-adaptive analysis. For $\eta = 0.04$ the relative error is below the desired value of 5 per cent.

Figure 8 shows the mesh developments for two different prescribed relative errors, $\eta = 0.05$ and 0.025, respectively, and $\varepsilon = 2$. In the first remeshing step with $\eta = 0.05$ four *r*-adaptive zones can be recognized, separated by three *h*-refinement zones in the centre and two *h*-derefinement zones at the extremes of the domain. On the other hand, in the $\eta = 0.025$ analysis only two *r*-adaptive zones appear in the first remeshing step, located at the outer parts of the

Table I. Example of a strip—relative error $\|e\|/\|u\|$ and number of elements N^{cur} for different values of ε.

ε		0	1	2	3	4	5	6
1.00	$\|e\|/\|u\|$	0.1172	0.0702	0.0488	—	—	—	—
	N^{cur}	288	537	994				
1.25	$\|e\|/\|u\|$	0.1172	0.0774	0.0607	0.0561	0.0537	0.0524	0.0514
	N^{cur}	288	435	646	694	736	740	738
1.50	$\|e\|/\|u\|$	0.1172	0.0726	0.0598	0.0567	0.0563	0.0560	0.0558
	N^{cur}	288	502	560	556	556	556	556
2.00	$\|e\|/\|u\|$	0.1172	0.0805	0.0738	0.0688	0.0681	0.0678	0.0676
	N^{cur}	288	390	390	390	390	390	390
2.50	$\|e\|/\|u\|$	0.1172	0.0780	0.0709	0.0668	0.0661	0.0648	0.0638
	N^{cur}	288	370	370	370	370	370	370
3.00	$\|e\|/\|u\|$	0.1172	0.0888	0.0776	0.0791	0.0813	0.0711	0.0699
	N^{cur}	288	326	326	326	326	326	326
10^2	$\|e\|/\|u\|$	0.1172	0.0912	0.0859	0.0763	0.0771	0.0776	0.0778
	N^{cur}	288	288	288	288	288	288	288

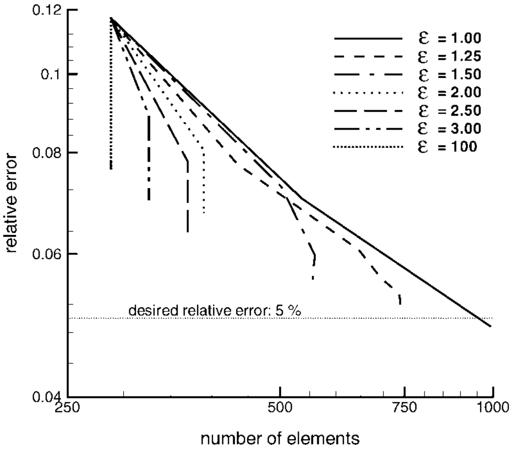


Figure 6. Example of a strip—mesh convergence for different values of ε.

domain. For both analyses most of the h-adaptation takes place in the first remeshing step. In subsequent steps, the majority of the domain belongs to the r-adaptive subdomain.

In a second example, we study the evolution of the error in the two subdomains. The geometry is shown in Figure 9. We follow the same approach as in the previous example, now with $u_{\mathrm{exact}} = 5\exp(-5y^2)\,\mathrm{K}$ and $c = 1\,\mathrm{W/mK}$. The desired relative error is $\eta = 0.05$, so that $\eta\|u\| = 0.497$. Two values for ε have been compared, namely $\varepsilon = 1.5$ and $\varepsilon = 2.0$. In Figure 10, the subdivision of the domain into an h-adaptive subdomain and an r-adaptive subdomain is depicted for subsequent stages.

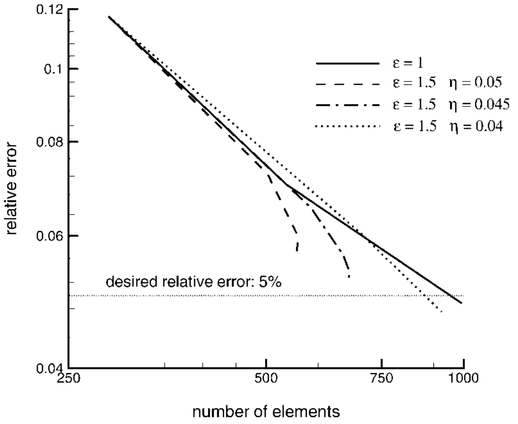


Figure 7. Example of a strip—mesh convergence for different values of η.

We compute the norm of the error in the h- and the r-adaptive zones separately in order to illustrate how the error evolves in each subdomain. This cannot be done in a straightforward manner, since the h-adaptive zones and the r-adaptive zones themselves evolve during the adaptive process. Thus, the error in each subdomain changes from one mesh to another due to (i) remeshing and (ii) a changing subdivision of the domain. Therefore, the subdomains must be 'fixed' for a correct investigation of the error development in each subdomain. We have projected the error distribution onto the subdomains of the *previous* mesh. For instance, the error in the r-adaptive subdomain of the initial mesh (Figure 10, top row) is compared to *exactly the same area* of the domain, but now taken from the mesh after remeshing (Figure 10, middle row). In this manner, the error development in both subdomains can be measured in an objective sense. Table II shows the energy norm of the error in the whole domain, the r- and the h-adaptive subdomain, as well as the number of elements of each of these domains. Note that $\|e\|^2 = \|e\|_r^2 + \|e\|_h^2$ (cf. Equation (2)). From the number of elements as well as from Figure 10 it can be seen that the r-adaptive subdomain grows with respect to the h-adaptive subdomain for ongoing remeshing steps. For larger ε the r-adaptive subdomain is larger. The error norms show that large decreases in error can be obtained in the h-adaptive subdomain, whereas the accuracy gain in the r-adaptive subdomain is smaller.

Thirdly, the example of Section 1 is analysed again, but now with the combined rh-adaptive approach. The geometry can be found in Figure 1, while we take the loading conditions of case B. As such, the exact temperature field $u_{\text{exact}} = 3\exp(-1.5s^2)$ whereby $s = \frac{1}{2}\sqrt{2}(x - y - 1)$. Furthermore, the conductivity $c = 1\,\text{W/mK}$. Again, the desired relative error $\eta = 0.05$. Figure 11 shows the relative error as a function of the number of elements for various values of ε. Basically, the same trends as with the other examples are found. Larger values for ε lead to higher rates of mesh convergence, while lower values for ε lead to higher final accuracies. In Figure 12, the final meshes for the various values of ε have been plotted. Note that the purely r-adaptive approach ($\varepsilon = 100$) leads to element entanglement for this example (see also Figure 1, right). Furthermore, the analysis with $\varepsilon = 2$ leads to some very badly shaped elements around the lower-right corner of the hole. This demonstrates the need for a certain degree of h-adaptivity to avoid problems around the hole.

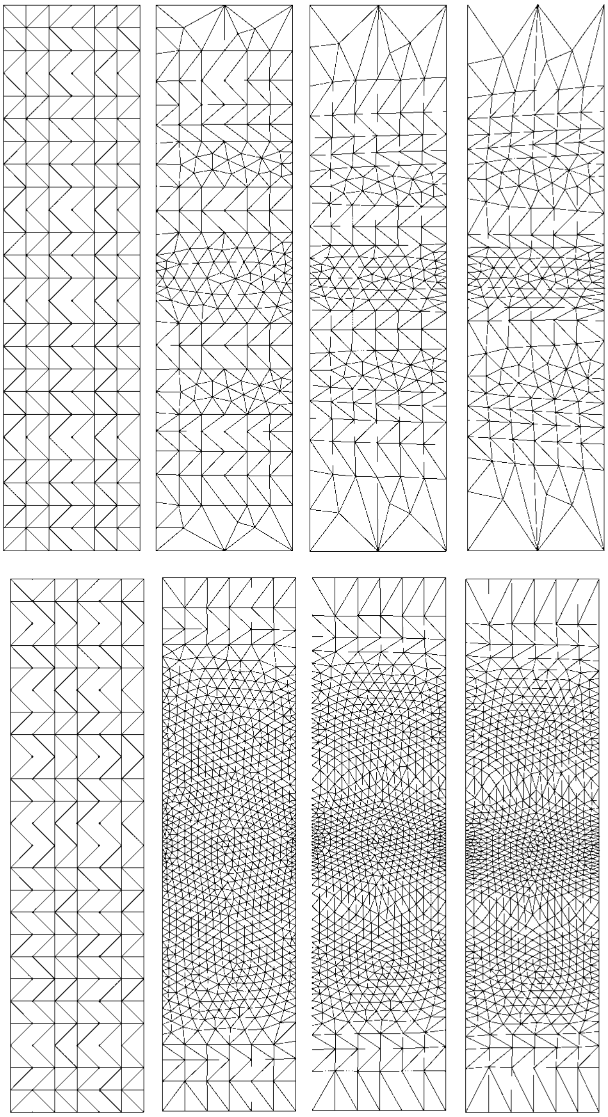


Figure 8. Example of a strip—mesh development for $\varepsilon = 2$, prescribed relative error $\eta = 0.05$ (top row) and $\eta = 0.025$ (bottom row).

5.2. Varying subdivision parameter ε

In the previous section, the subdivision parameter ε has been fixed. In this section, the effect of a varying ε is studied. Two strategies are proposed, which follow different reasonings. The first alternative aims to facilitate the adjustment of the number of elements in all stages of the computation, whereas the second alternative attempts to meet the desired number of elements in an early stage of the analysis.

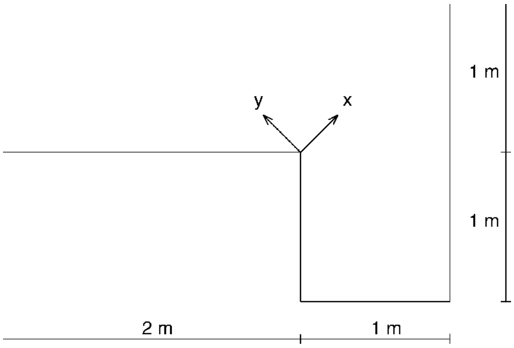


Figure 9. L-shape example—geometry.

For the first approach we have taken an expression for ε that *decreases* with a decreasing error as

$$\varepsilon = \frac{\|e\|}{\eta\|u\|} \tag{16}$$

In the limit state, i.e. as $\|e\|$ approaches $\eta\|u\|$, ε goes to one. Consequently, the interval $[1/\varepsilon, \varepsilon]$ on basis of which elements are assigned to the *r*-adaptive subdomain becomes smaller and smaller. In other words, a significant part of the domain will remain *h*-adaptive (contrary to the case of a fixed value for ε, see Section 5.1). Thus, the number of elements can easily be adjusted in all remeshing steps. The example of the strip is analysed again but now with ε according to Equation (16). The desired relative error $\eta = 0.05$. The mesh convergence behaviour is depicted in Figure 13. As a reference also the case of a fixed $\varepsilon = 1.0$ (full *h*-adaptive) is plotted in this figure. It can be seen that the mesh convergence behaviour is not an improvement of the results obtained with a fixed ε (cf. Figure 6). Although elements are added in all remeshing steps, the numerical error arrests at a certain stage and does not decrease anymore. The reason for this behaviour is the relatively complex form of the interface between the *h*-adaptive subdomain and the *r*-adaptive subdomain. Whereas for a *fixed* ε this interface becomes more and more simple and eventually disappears (see Figure 10, for instance), a *decreasing* ε maintains a significant *h*-adaptive subdomain even in later remeshing steps. As a consequence, the interface between the *r*-adaptive subdomain and the *h*-adaptive subdomain does not disappear but remains of relatively complex form. This poses topological constraints on the *h*-adaptive remeshing as well as on the *r*-adaptive remeshing, by which the decrease in error is severely reduced. To mitigate this deficiency, more flexible formulations of the interface should be considered. However, this falls beyond the scope of the present study.

A totally different approach is to let ε *increase* during the analysis. This is motivated by the fact that a small ε (much *h*-adaptivity) in the first remeshing steps is more suitable to meet the desired number of elements more precisely in an early stage of the analysis, whereas a larger ε (much *r*-adaptivity) in later remeshing steps facilitates the fine-tuning of the nodal co-ordinates. Thus, the interface between the two subdomains has a less complicated form compared to the case of a *decreasing* ε, by which less topological constraints are imposed on

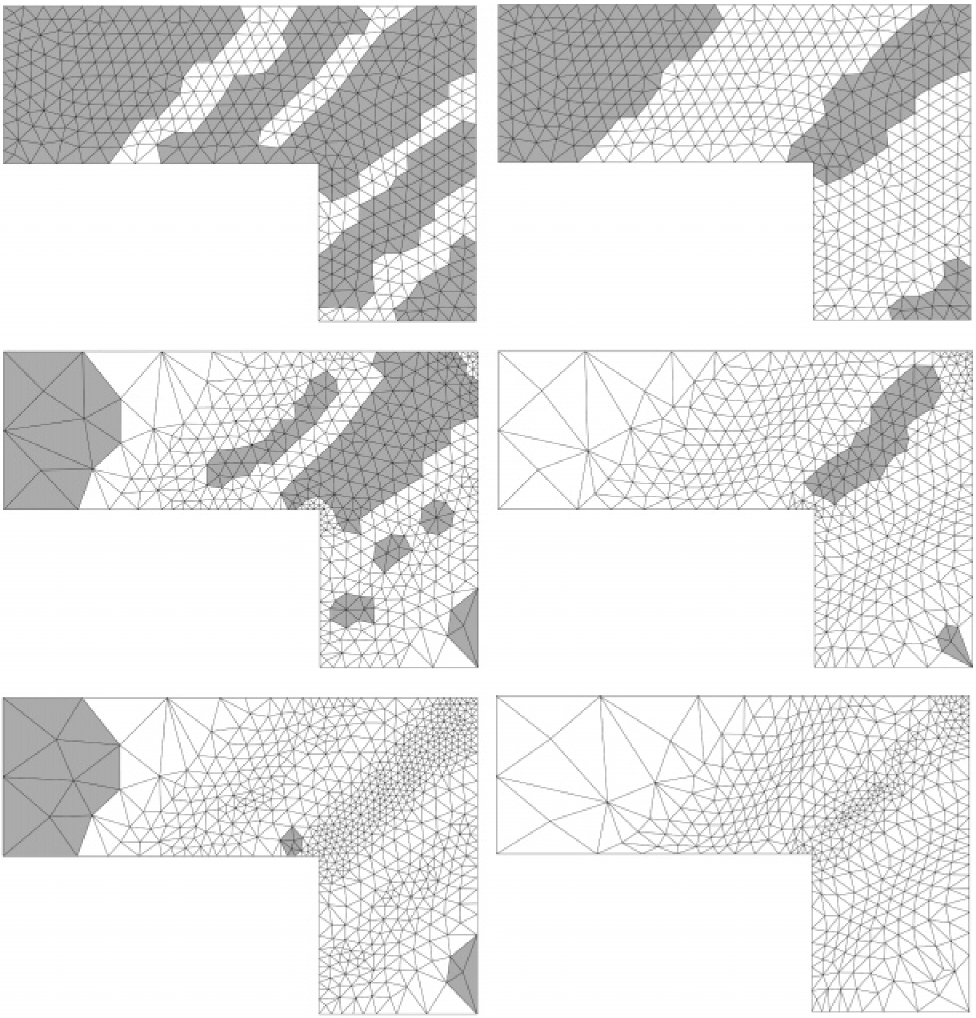


Figure 10. L-shape example—subdivision into h-adaptive subdomain (gray) and r-adaptive subdomain (white) for $\varepsilon = 1.5$ (left column) and $\varepsilon = 2.0$ (right column); initial mesh (mesh 0—top row), mesh after one remeshing step (mesh 1—middle row) and mesh after two remeshing steps (mesh 2—bottom row).

the remeshing process. We have adopted the following definition of ε:

$$\varepsilon = \frac{\eta\|u\|}{\beta\|e\|} + 1 \tag{17}$$

where β is a numerical parameter. Note that $\varepsilon \approx 1$ for relatively large values of $\|e\|$, whereas $\varepsilon \to 1/\beta + 1$ if $\|e\|$ approaches $\eta\|u\|$. Thus, ε goes to 2 if $\beta = 1$, ε goes to 1.5 if $\beta = 2$, etc. The mesh convergence behaviour of the strip example with $\eta = 0.05$ for an increasing ε is shown in Figure 14. Similar to the case of a fixed ε, the mesh convergence behaviour for

Table II. L-shape example—error development and number of elements.

		Step 1 Mesh 0 → Mesh 1		Step 2 Mesh 1 → Mesh 2		Step 3 Mesh 2 → Mesh 3	
$\varepsilon = 1.5$	$\|\|e\|\|$	1.02592	0.92105	0.92105	0.59855	0.59855	0.57943
	N^{cur}	852	821	821	1245	1245	1247
	$\|\|e\|\|_r$	0.25229	0.24051	0.43249	0.40300	0.59222	0.57271
	N_r^{cur}	230	230	538	538	1221	1221
	$\|\|e\|\|_h$	0.99442	0.88909	0.81319	0.44255	0.08683	0.08799
	N_h^{cur}	622	591	283	707	24	26
ε=2.0	$\|\|e\|\|$	1.02592	0.85628	0.85628	0.75951	0.75921	0.69872
	N^{cur}	852	740	740	844	844	844
	$\|\|e\|\|_r$	0.59277	0.51634	0.64836	0.64987	0.75921	0.69872
	N_r^{cur}	487	487	665	665	844	844
	$\|\|e\|\|_h$	0.83734	0.68309	0.55932	0.39310	0	0
	N_h^{cur}	365	253	75	179	0	0

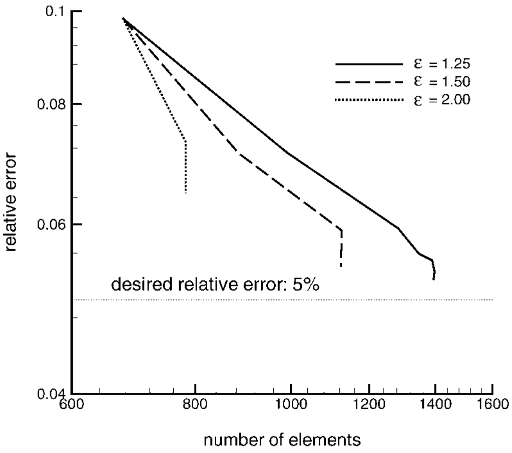


Figure 11. Square with a hole—mesh convergence for fixed values of ε.

for increasing ε is better than that of a full *h*-adaptive analysis. For the case $\varepsilon \rightarrow 1.25$ ($\beta = 4$) the desired relative error is met. Compared to the fixed ε case, more elements are added in an early stage of the analysis, so that the final error is lower.

Also, the example of the square with a hole is analysed with the increasing ε according to Equation (17). Figure 15 shows the mesh convergence behaviour for various values of β, while $\eta = 0.05$. For the case $\varepsilon \rightarrow 1.25$ the error tolerance is met. Compared to the corresponding analyses with fixed values of ε (see Figure 11), the mesh convergence rate is somewhat lower while the final accuracy has improved. This is in accordance with the higher number of elements that is added in the earlier stages of the analyses.

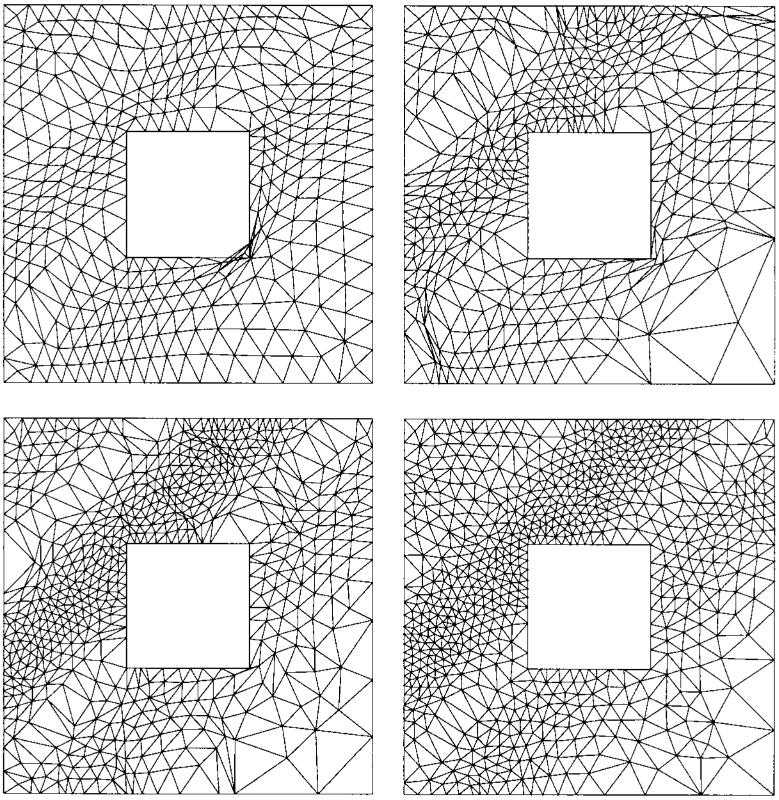


Figure 12. Square with a hole—final meshes for $\varepsilon = 100$ (top left), $\varepsilon = 2$ (top right), $\varepsilon = 1.5$ (bottom left) and $\varepsilon = 1.25$ (bottom right).

6. DISCUSSION

The paper aims to combine the advantageous properties of two adaptive strategies, namely, node relocation (r-adaptivity) and adjusting element sizes and mesh topology (h-adaptivity). Basically, r-adaptive strategies are recognized to be computationally inexpensive but with limited applicability, whereas h-adaptive strategies are suitable for meeting a prescribed accuracy at the cost of a higher computational effort.

In the current study, the domain under consideration is split into two subdomains. In the regions where the element sizes do not differ too much from the desired element sizes node relocation is applied. In the remainder of the domain, an h-adaptive approach is chosen. The subdivision of the domain as well as the construction of a new discretization in both subdomains is based on the computation of a desired element size, so that generality is preserved. Through this domain subdivision, each remeshing technique (r-adaptivity, h-adaptivity) can be used in the subdomain where it is suited most.

The present approach offers a more diffuse transition from adaptive strategies where meeting a prescribed accuracy is the goal towards adaptive strategies where computational speed-up

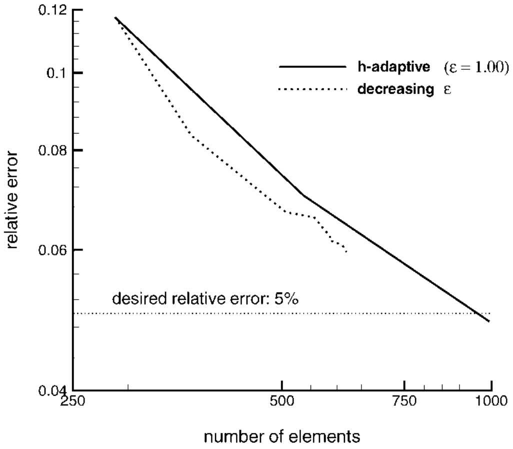
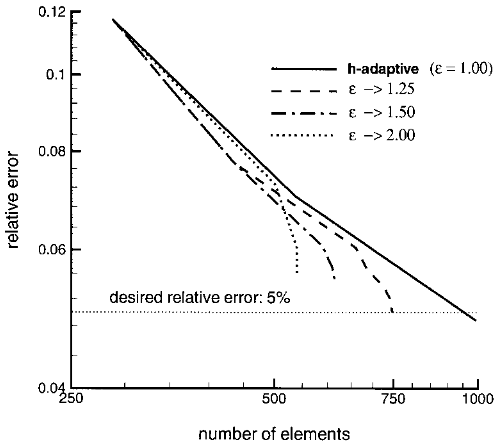


Figure 13. Example of a strip—mesh convergence for a decreasing ε according to Equation (16).



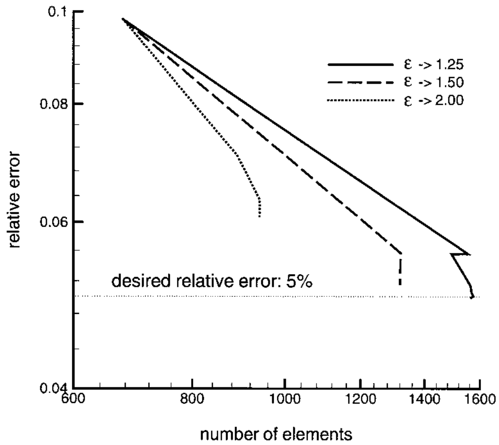


Figure 14. Example of a strip—mesh convergence for an increasing ε according to Equation (17).

Figure 15. Square with a hole—mesh convergence for an increasing ε according to Equation (17).

is desired. As such, the present approach can be regarded as an enhancement of the *h*-adaptive finite element method in which parts of the domain are remeshed with the cheaper *r*-adaptive technique. Conversely, the proposed combined *rh*-adaptive strategy may be seen as an extension of *r*-adaptive approaches, namely where *h*-adaptivity is used if *r*-adaptivity is deemed unsuitable.

A numerical parameter has been introduced that can be set by the user in order to attach more importance to accuracy or to computational speed-up. Alternatively, this parameter can be computed as a function of the error level. If more weight is put on *h*-adaptivity, then the final relative error is smaller, but the computational costs increase accordingly.

Conversely, if r-adaptivity is emphasized, then the mesh convergence behaviour is better while attaining a prescribed error level may become impossible. Compared to a purely h-adaptive technique, the computational costs are lower in the combined rh-adaptive approach. With a suitable choice for the above-mentioned numerical parameter the same accuracy can be obtained with a smaller number of finite elements. On the other hand, the proposed combined rh-adaptive strategy offers better accuracies than a purely r-adaptive approach.

Features that will be addressed in the future include the following:

- The treatment of the intersection between the r- and the h-adaptive subdomain has not been investigated in depth. In the present study, only a fixed interface is considered, which can have an interfering effect on the adaptivity process in the r-adaptive subdomain as well as the h-adaptive subdomain. Future investigations should include r- and/or h-adaptivity at the intersection.
- Using the numerical parameter ε as a criterion to split the domain has been a first attempt. More sophisticated algorithms may be used and compared. Furthermore, the case of a decreasing value for the subdivision parameter ε should be studied in combination with a less rigid treatment of the interface between the two subdomains. It is expected that through this combination the mesh convergence behaviour of rh-adaptive schemes can be improved.
- In the present study, only exact errors have been used. The performance of the proposed rh-adaptive strategy should also be assessed with error estimators and error indicators.
- The extension towards non-linear problems requires an algorithm to transfer the state variables. As with the adaptation of the mesh different stress transfer algorithms can be used that suit the two subdomains best.

ACKNOWLEDGEMENTS

We wish to thank Pedro Díez and Antonio Huerta for the various technical discussions. Furthermore, we gratefully acknowledge the Commissariat à l'Énergie Atomique, France, for the availability of the finite element code CASTEM 2000.

REFERENCES

1. Zienkiewicz OC, Zhu JZ. Adaptivity and mesh generation. *International Journal for Numerical Methods in Engineering* 1991; **32**:783–810.
2. Huerta A, Rodríguez-Ferran A, Díez P, Sarrate J. Adaptive finite element strategies based on error assessment. *International Journal for Numerical Methods in Engineering* 1999; **46**:1803–1818.
3. Ladevèze P, Rougeot Ph, Blanchard P, Moreau JP. Local error estimators for finite element linear analysis. *Computer Methods in Applied Mechanics and Engineering* 1999; **176**:231–246.
4. Babuška I, Strouboulis T, Gangaraj SK, Upadhyay CS. Pollution error in the h-version of the finite-element method and the local quality of the recovered derivatives. *Computer Methods in Applied Mechanics and Engineering* 1997; **140**:1–37.
5. Hager P, Wiberg N-E. Adaptive eigenfrequency analysis by superconvergent patch recovery. *Computer Methods in Applied Mechanics and Engineering* 1999; **176**:441–462.
6. Hughes TJR, Liu WK, Zimmermann TK. Lagrangian–Eulerian finite element formulation for incompressible viscous flows. *Computer Methods in Applied Mechanics and Engineering* 1981; **29**:329–349.
7. Donéa, J. Arbitrary Lagrangian–Eulerian finite element methods. In *Computational Methods for Transient Analysis* Chapter 10, Belytschko T, Hughes TJR (eds). Elsevier: Amsterdam, 1983; 474–516.
8. Rodríguez-Ferran A, Casadei F, Huerta A. ALE stress update for transient and quasistatic processes. *International Journal for Numerical Methods in Engineering* 1998; **43**:241–262.

9. Askes H, Sluys LJ. Remeshing strategies for adaptive ALE analysis of strain localisation. *European Journal of Mechanics A/Solids* 2000; **19**:447–467.
10. Cao T. Adaptive H- and H-R methods for Symm's integral equation. *Computer Methods in Applied Mechanics and Engineering* 1998; **162**:1–17.
11. Ammons BA, Vable M. An *hr*-method of mesh refinement for boundary element method. *International Journal for Numerical Methods in Engineering* 1998; **43**:979–996.
12. Kita E, Higuchi K, Kamiya N. *r*- and *hr*-adaptive boundary element method for two-dimensional potential problem. *Computers and Structures* 2000; **74**:11–19.
13. Ortiz M, Quigley JJ. Adaptive mesh refinement in strain localization problems. *Computer Methods in Applied Mechanics and Engineering* 1991; **90**:781–804.
14. Díez P, Egozcue JJ, Huerta A. A posteriori error estimation for standard finite element analysis. *Computer Methods in Applied Mechanics and Engineering* 1998; **163**:141–157.
15. Li LY, Bettess P. Notes on mesh optimal criteria in adaptive finite element computations. *Communications in Numerical Methods in Engineering* 1995; **11**:911–915.
16. Li LY, Bettess P, Bull JW, Bond T, Applegarth I. Theoretical formulations for adaptive finite element computations. *Communications in Numerical Methods in Engineering* 1995; **11**:857–868.
17. Zienkiewicz OC, Zhu JZ. A simple error estimator and adaptive procedure for practical engineering analysis. *International Journal for Numerical Methods in Engineering* 1987; **24**:337–357.
18. Oñate E, Bugeda G. A study of mesh optimality criteria in adaptive finite element analysis. *Engineering Computations* 1993; **10**:307–321.
19. Díez P, Huerta A. A unified approach to remeshing strategies for finite element *h*-adaptivity. *Computer Methods in Applied Mechanics and Engineering* 1999; **176**:215–229.
20. Zhu JZ, Zienkiewicz OC. Adaptive techniques in the finite element method. *Communications in Applied Numerical Methods* 1988; **4**:197–204.
21. Zienkiewicz OC, Zhu JZ. Error estimates and adaptive refinement for plate bending problems. *International Journal for Numerical Methods in Engineering* 1989; **28**:2839–2853.
22. Bodé L. Stratégies numériques pour la prévision de la ruine des structures du génie civil. *Dissertation* E.N.S. de Cachan/CNRS/Université Paris 6, 1994.
23. Pijaudier-Cabot G, Bodé L, Huerta A. Arbitrary Lagrangian–Eulerian finite element analysis of strain localization in transient problems. *International Journal for Numerical Methods in Engineering* 1995; **38**:4171–4191.
24. Huerta A, Casadei F. New ALE applications in non-linear fast-transient solid dynamics. *Engineering Computations* 1994; **11**:317–345.
25. Huerta A, Liu WK. Viscous flow with large free surface motion. *Computer Methods in Applied Mechanics and Engineering* 1988; **69**:277–324.
26. CASTEM 2000. Manuel d'utilisation. *Technical Report* 88/176, Laboratoire d'Analyse Mécanique des Structures, Commissariat à l'Énergie Atomique, Saclay, France, 1988.