

ADAPTIVE GRID-DESIGN METHODS FOR FINITE ELEMENT ANALYSIS

Noboru KIKUCHI

*Department of Mechanical Engineering and Applied Mechanics, The University of Michigan,
Ann Arbor, MI 48109, U.S.A.*

Received 31 July 1985

This paper is concerned with an introduction of a concept of adaptive grid design for finite element analysis by combining numerical grid-generation methods and adaptive finite element methods. Development of a finite model is considered as a design problem similar to structural optimization problems.

1. Introduction

After a quarter century of development finite element methods have reached that point of maturity at which nobody doubts their effectiveness and power to solve various kinds of engineering problems with a variety of aspects, both very theoretical and extremely practical. Based on the construction of theories of finite element methods during the 1960s, many general-purpose codes have been developed which are commercially available and have become standard means to analyze stresses, strains, deformations, velocity, and reactive forces of solid structures and machine components. Furthermore, due to the extensive advancement of computer hardware even these large-size finite element programs are executed not only by the so-called superminicomputers but also by the micro/personal computers such as the IBM PCs. Another very important new aspect in computer-aided engineering is the concept of networking of several levels of hardware which enables the linking of several software systems for computer-aided design and manufacturing. In this new context finite element methods are considered an extremely powerful means for analysis, and they have become part of a system of design/manufacturing activities.

In this paper we will develop a theory of adaptive finite element grid-design methods for finite element approximations, which can assure that the approximated solution to a given problem is accurate enough to be adequate for quality control of products, and which is consistently applicable in the network environment for design/manufacturing activities. To this end we will study methods of numerical grid generation and adaptive finite elements in order to design a better finite element grid, in the sense that it reflects the solution characteristics, and to improve the accuracy of finite element approximations at the same time. The basic idea in the present approach is to set up the grid-generation problem as a design problem which is similar to the optimal design of structures. That is, starting from a given finite element grid which is already generated by the application of numerical grid-generation methods, the optimum grid will be obtained by a certain iterative manner following, for example, the

optimality-criteria method in structural optimization. This is only possible when an optimal grid-design problem is formulated mathematically and a necessary condition for optimality is obtained. It is certain that the formulation of a grid-design problem is not unique. For example, if a grid-design problem is defined to minimize the maximum value of the error of a finite element approximation, different choices of error measures yield different problems and possibly different optimal grids. Therefore, it will be very important to know various kinds of error measures of finite element approximations, which reflect the nature of the solution and the designer's desire for analysis.

The necessity of mixing numerical grid-generation and adaptive methods is also recognized through the present study. For example, for extremely large deformation analysis of metal forming, no matter how existing adaptive methods such as the r -, h -, and p -methods are applied, it is impossible to treat adequately finite element solutions at nonconvex corner points, where deformation is localized and material points turn their position from one boundary segment to another. In this case, we will have a negative determinant of either the deformation gradient or the Jacobian transformation matrix from the physical and isoparametric finite element coordinate systems. A negative determinant may be avoided in very lucky cases, but the quality of finite element approximations is definitely poor at these points despite the application of adaptive methods. One of the methods is remeshing, that is, after a certain deformation a new finite element grid is defined to avoid very distorted finite elements for the updated Lagrangian approach, or to maintain a reasonably small error amount accumulated in certain specific finite elements for the total Lagrangian approach. For remeshing two aspects are important. An adequate measure must be introduced to decide whether remeshing is required and the logic for automatic remeshing must be defined for practical applications.

It is noted that the present study largely owes to previous extensive and thorough studies on numerical grid-generation methods by Thompson [1] and Thompson et al. [2]; on finite element grid-optimization and adaptive methods given by Shephard [3], Babuška et al. [4], and Oliveira et al. [5]; and on adaptive methods for finite difference methods in Ghia and Ghia [6], although the spirit for synthesis of these studies is based on the memorial works [7, 8] by Southwell in which many grid-generation methods as well as the origin of the h -adaptive method were already discussed.

2. General procedure for adaptive grid design

Roughly speaking, a procedure for grid design for finite element, finite difference, and boundary element methods can be defined by the following flow chart:

- Step 1.* Geometric modeling of a given structure.
- Step 2.* Modeling for mechanics.
- Step 3.* Rough characterization of the solution.
- Step 4.* Development of the initial grid.
- Step 5.* FEM, FDM, BEM analysis.
- Step 6.* Computation of error measures for adaption and remeshing.
- Step 7.* Remeshing.
- Step 8.* Adaptation.
- Step 9.* Postprocedure.

The first step is geometric modeling of a given structure. Especially boundary segments and surfaces are represented by appropriate spline functions which will also be used in numerical grid generation. In many cases the boundary of a given structure is decomposed into several or many segments and surfaces so that they can be approximated by reasonably low-order spline or equivalent polynomial functions.

Step 2, modeling for mechanics, involves the specification of loading and kinematical boundary conditions and material characterization. It is then necessary to develop an image of the solution in order to grasp the qualitative behavior of the solution of the problem setup in the first two procedures. Based on the nature of the imagined solution, the initial grid for analysis by finite element, finite difference, and boundary element methods must be defined using numerical grid-generation methods.

According to Thompson et al. [2] numerical grid-generation methods can be classified into three groups:

- (1) conformal mapping methods;
- (2) differential equation (elliptic or hyperbolic type) methods;
- (3) algebraic methods.

Conformal mapping methods are based on the flow net in a two-dimensional potential flowfield. That is, the families of curves of the stream function, $\xi = \text{constant}$, and the potential function, $\eta = \text{constant}$, form an orthogonal system, and when the intervals between consecutive values of ξ and η are equal and sufficiently small, they form an orthogonal net. Elliptic differential equation methods are a generalization of conformal mapping methods in order to generate grids for complicated, possibly curved, domains. That is, since both stream and potential functions satisfy the Laplace equations

$$-\Delta\xi = 0, \quad -\Delta\eta = 0, \quad (1)$$

the physical coordinates (x, y) are determined by the second-order elliptic-type differential equations

$$-\nabla \cdot (D(\xi, \eta)\nabla\xi) = P(\xi, \eta) \quad \text{and} \quad -\nabla \cdot (D(\xi, \eta)\nabla\eta) = Q(\xi, \eta), \quad (2)$$

where ∇ is the gradient operator in the physical coordinate system

$$\nabla = i\partial/\partial x + j\partial/\partial y.$$

In this case, (ξ, η) should be considered as the grid coordinates rather than the stream and potential functions. The existence of the terms D , P , and Q will provide the possibility to combine adaptive methods. Indeed, the analogy to heat-conduction problems yields that smaller grids are obtained in the portion where D , P , and/or Q are larger than in other portions. Thus, the adaptation will be performed by adjusting either the amount of heat sources P and Q , or the heat conductivity D to generate the desired grids. One of the annoying tasks in this approach is to set up boundary conditions which define the boundary location in the physical coordinate system. To do this, as mentioned earlier, expressing the boundary segments by some spline functions is very helpful.

Although the first and second approach are very similar, the third one is, relatively speaking, more independent from the others. Within this category, algebraic serendipity-mapping and

algebraic integer methods are the most popular for numerical grid generation. A number of engineers, especially, adopt the serendipity-mapping method to generate a finite element grid since the same serendipity mappings are applied to define the shape functions of the serendipity elements for finite element approximations. This method first decomposes a given structure into many blocks which can be identified with either triangular or quadrilateral serendipity elements. Each block, then, is decomposed into a given number of finite elements by specifying the number of grids and the grid gradient in each coordinate axis of the normalized coordinate system in the master element of the finite elements. More precisely, if (ξ, η) are the normalized coordinates for the square master element defined by the region $\{(\xi, \eta): -1 < \xi < 1, -1 < \eta < 1\}$, then a rectangular net constructed in the master element is mapped into a block in the physical coordinate system by the algebraic transformations

$$x = \sum x_{\alpha} N_{\alpha}(\xi, \eta), \quad y = \sum y_{\alpha} N_{\alpha}(\xi, \eta), \quad (3)$$

where (x_{α}, y_{α}) are the coordinates of the nodes characterizing a block, and N_{α} are the shape functions of the serendipity element. This method is based on the simplest concept, although it needs very annoying work to maintain continuity along the interfaces of blocks forming a given domain, especially for very complicated structures. Furthermore, this approach may yield distorted grids. The algebraic integer method first spans either a triangular or rectangular net which covers a given domain completely. Every grid point identified by the integers (i, j) is labeled by the location index L_{ij} such that $L_{ij} = 1$ if (i, j) is inside the domain and $L_{ij} = 0$ if (i, j) is on the outside. If all the grid points whose location indices are equal to one are accumulated, a zigzag domain is obtained which is contained in the given domain. Then the grid points on the edges of the zigzag domain are projected on the boundary of the original domain. This procedure may yield very distorted and irregular elements near the boundary, but these can be modified by applying grid-smoothing schemes.

Now, after generating a grid, the next step is analysis by, for example, finite element methods. An approximated solution is computed. For *the adaptive grid design* which is introduced in this paper, we cannot terminate the execution of programs as usual. It is necessary to compute error measures that quantify the amount of error of the approximation, although the exact amount may not be computed. Using the computed error measures, it is determined whether remeshing and adapting grids are necessary to maintain the level of error in the approximation. The question is how error measures are defined for a given problem. To answer this question it must be noted that extensive work on this issue has been done by Babuška and his co-workers, and that very clear ideas for error measures and indicators have been obtained. In this paper, despite some disagreement with Babuška's and other mathematicians' definitions of error indicators, we will compute an error measure based on the so-called interpolation error of the solution, which bounds the error of the finite element approximations from above. As far as our experience is concerned, this approach would not imply large disagreement with others on the adapted grids. More precisely, the adapted grids based on the interpolation error always yield improved solutions.

If it is necessary to do, remeshing for large-deformation nonlinear or free boundary problems must be completely automatic. Otherwise, it is not attractive at all. A possible approach for automatic remeshing is the application of grid-generation methods such as elliptic differential equation and algebraic integer methods. Conformal mapping and algebraic

serendipity mapping methods may not be applicable for this purpose, since decomposed blocks must be assigned at the initial stage that does not contain any information on how grids will be distorted during the process of analysis. Even differential equation methods are considerably more difficult to apply, because the boundary segments must be set up at the initial stage. Algebraic integer methods seem the most promising for remeshing at this moment.

The main process of the adaptive grid design is the application of adaptive methods based on the optimal grid-design problem defined by

$$\begin{array}{l} \text{Minimize Maximum } E_e, \\ \text{by grid design } \quad e = 1, \dots, \text{NE} \end{array} \quad (4)$$

where E_e is an error measure of e th finite element and NE is the total number of elements in finite element approximations. So far, three adaptive methods have been introduced. The oldest one is the r -method that relocates nodes (that is, grid points) so that smaller-sized elements are accumulated at the portion where the approximation error is large. The total number of degrees of freedom and element connectivities are unchanged during this adaptation. Only the nodal coordinates are changed. The h -method refines an element into several elements if it contains a large amount of error, that is, the total number of elements (and thus the total number of degrees of freedom) increases during this adaptation. Very refined finite elements will be generated around singular points. The p -method introduces higher-order polynomials for the shape functions of elements in which a relatively large amount of error is accumulated. If the error level is low in certain elements, low-order polynomials are sufficient for the shape functions. This method also increases the total number of degrees of freedom. Furthermore, this requires that finite element analysis programs be capable of handling various types of shape functions, whereas for the h -method a single choice of shape functions is sufficient for all the elements. By deciding which adaptation is applied, we redesign the initial grid so that an optimality condition to the optimization problem (4),

$$E_e = \text{constant} \quad \text{for } e = 1, \dots, \text{NE}, \quad (5)$$

is more likely satisfied by the adapted grid. Then, if it is necessary to go back to the analysis again, we simply return to the analysis process and repeat this until a satisfactory approximation is obtained.

As described above the adaptive grid design is a combination of numerical grid-generation and adaptive finite element (or possibly finite difference and boundary element) methods. According to the approximated solution, the grid is adapted or remeshed completely in order to maintain or improve the accuracy of the solution.

3. Numerical grid generations

As shown in the previous section, numerical grid-generation methods can be classified into three groups: conformal mapping, differential equation, and algebraic methods as Thompson et al. [2] described in their very thorough survey on this issue for finite difference methods. In

this work, we will follow Thompson and review some grid-generation methods, especially toward adaptivity and automatic remeshing of a given irregular domain. It is certain that covering all available grid-generation methods is impossible. Here, we will present those we have already applied to solve linear and nonlinear mechanics problems at the University of Michigan.

3.1. Conformal mapping methods

These methods are based on the fact that the families of curves of the constant-stream and constant-potential functions form an orthogonal system, the so-called flow net in the potential flow field. If the flow net is identified with the grid for finite element approximations, this yields a means to generate a family of quadrilateral elements. The coordinates (ξ, η) consisting of the stream and potential functions ξ and η , respectively, are identified with the grid coordinate system. For example, the conformal mapping

$$w = \xi + i\eta = ar^n, \quad r = x + iy \quad (6)$$

is defined, where (x, y) are the coordinates in the physical coordinate system, and the grid coordinates are computed by the inverse relation

$$r = x + iy = (w/a)^{1/n}, \quad \text{i.e.,} \quad x = (R/a)^{1/n} \cos(\theta/n), \quad y = (R/a)^{1/n} \sin(\theta/n),$$

where $R = \sqrt{\xi^2 + \eta^2}$ and $\theta = \text{Arctan}(\xi/\eta)$. This mapping may provide grids for a portion of a given domain around a corner point. For example, for an L-shaped domain or a domain containing a crack inside, the conformal mapping (6) generates the grid shown in Fig. 1. It is clear that this grid represents a singular behavior near the corner point where the solution, in general, has a singularity, that is, elements of very small size are generated there. However, it is noted that the number of elements in the vicinity of a singular point is limited, for example, only one for $n = 2$ and three for $n = 4$. This leaves a somewhat unsatisfactory feeling to the grid generated, although refined finite elements are obtained in the neighborhood of a singular point. This nature is quite different from the algebraic grid generation using the idea of serendipity transformation of finite element methods as shown later. Roughly speaking, if the number of grids in the circumference is specified, the algebraic serendipity-mapping method can produce very refined elements at a singular point. That is, the number of elements can be specified in the vicinity of a singular point, while the conformal mapping method automatically defines the number of elements at a singular point despite the arbitrariness of the total number of elements.

As in [2], the conformal mapping method may be classified into two main subgroups: the series-transformation method and the Schwarz–Christoffel transformation method. The first one is nothing but a method that generates a grid using a linear combination of known conformal mappings. For example, Gnoffo [9] introduced a finite series expansion

$$\begin{aligned} x &= (-B \sinh \eta + C \cosh \eta) \cos \xi - \sum_{n=2} A_n e^{n\eta} \cos(n\eta), \\ y &= (B \cosh \eta - C \sinh \eta) \sin \xi + \sum_{n=2} A_n e^{n\eta} \sin(n\eta). \end{aligned} \quad (7)$$

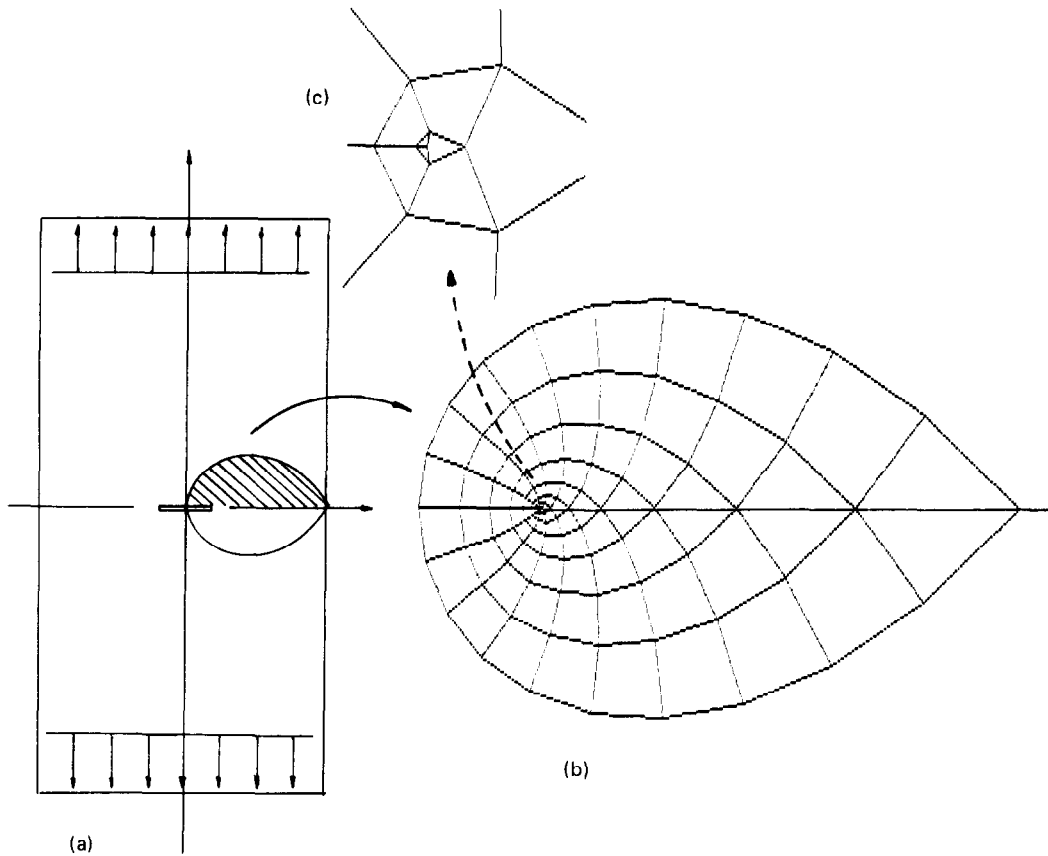


Fig. 1. An application of a conformal mapping for fracture mechanics. (a) Elastic thin plate subjected to a tension field. (b) Grid containing a crack. (c) Enlargement of the grid in the vicinity of a crack tip.

By defining coefficients appropriately an arbitrary domain is discretized into finite elements. However, it is clear that a given complex structure may not be discretized by this method. In fluid mechanics, the domain boundary is often defined according to the analyst's desire without destroying the nature of a given problem. But in solid and structural mechanics their configuration is a priori given and there is no chance to modify it. Thus this method may not be applicable for discretization of solids and structures in general.

The Schwarz–Christoffel transformation method possesses more flexibility than the series-transformation method. It can generate grids for an arbitrary polygon, even for a curved domain after certain modifications. The application of this approach in solid mechanics can be found in the work by Brown and Hayhurst [10] for fracture mechanics, whereas many applications for fluid mechanics can be found in [11–13]. Anderson [12] applied a third-order Runge–Kutta method to integrate the derivative form of the Schwarz–Christoffel transformation.

3.2. Elliptic differential equation methods

As a generalization of the conformal mapping method, the elliptic differential equation

method is the one most extensively studied so far. Noting that any conformal mapping satisfies the Laplace equation, which is a particular form of elliptic second-order differential equations, an extension to the use of elliptic differential equations is natural for grid generation. Indeed, Winslow [14] was the first to clearly state the use of elliptic equations to define a nonuniform triangular grid for an arbitrarily shaped region of a nonuniform material with a smoothly varying grid spacing. For the potential and stream functions ξ and η , respectively, which are identified by the grid coordinates, the Laplace equations

$$\Delta\xi = 0, \quad \Delta\eta = 0 \quad (8)$$

are considered over each region with boundary conditions determined by the interface and boundary zoning. Winslow thought that “because of the well-known averaging property of solutions to Laplace’s equation, we might expect a mesh constructed in this way to be, in some sense, smooth.” Equations (8) are solved numerically by inverting them and writing them in terms of the physical coordinates x and y which are functions of ξ and η . Using the relations

$$\eta_x = -y_\xi/J, \quad \eta_y = x_\xi/J, \quad \xi_x = y_\eta/J, \quad \xi_y = -x_\eta/J, \quad (9)$$

where J is the Jacobian defined by $J = x_\xi y_\eta - x_\eta y_\xi$, equations (8) can be transformed into the inverse Laplace equations

$$L(x) = \alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} = 0, \quad L(y) = \alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} = 0, \quad (10)$$

under the assumption $J \neq 0$, see [14, Appendix]. Here

$$\alpha = x_\eta^2 + y_\eta^2, \quad \beta = x_\xi x_\eta + y_\xi y_\eta, \quad \gamma = x_\xi^2 + y_\xi^2. \quad (11)$$

Subscripts indicate partial derivatives, for example, $x_\xi = \partial x / \partial \xi$. Later, this idea was extended to control the spacing of the coordinate lines. Thompson et al. [15] modified the Laplace equations (8) by adding the source terms P and Q

$$\Delta\xi = P(\xi, \eta), \quad \Delta\eta = Q(\xi, \eta). \quad (12)$$

Then (10) becomes

$$L(x) = -J(Px_\xi + Qx_\eta), \quad L(y) = -J(Py_\xi + Qy_\eta). \quad (13)$$

Another type of modification for the spacing of grids is given again by Winslow [16]. Modifying the Laplace equations (8) by

$$\nabla \cdot (D\nabla\xi) = 0, \quad \nabla \cdot (D\nabla\eta) = 0, \quad (14)$$

introducing a function $D(\xi, \eta)$, the elliptic equations (10) for grid generation are changed to

$$L(x) = -\sqrt{J}(D_\xi y_\eta - D_\eta y_\xi)/D, \quad L(y) = -\sqrt{J}(D_\xi x_\eta - D_\eta x_\xi). \quad (15)$$

It is now clear that if the grid generation follows the equations

$$\nabla \cdot (D\nabla\xi) = P, \quad \nabla \cdot (D\nabla\eta) = Q, \quad (16)$$

the transformed equations become

$$\begin{aligned} L(x) &= -J(Px_\xi + Qx_\eta) - \sqrt{J}(D_\xi y_\eta - D_\eta y_\xi)/D, \\ L(y) &= -J(Py_\eta + Qy_\xi) - \sqrt{J}(D_\xi x_\eta - D_\eta x_\xi)/D. \end{aligned} \quad (17)$$

Brackbill and Saltzman [17] derived similar elliptic differential equations for grid generation by using variational methods. Their approach has a very strong advantage: the variational problem can be set up to minimize the global smoothness of the mapping measured by the integral

$$I = \int_{\Omega} \{\nabla\xi \cdot \nabla\xi + \nabla\eta \cdot \nabla\eta\} d\Omega, \quad (18)$$

together with constraints such as the orthogonality constraint

$$\nabla\xi \cdot \nabla\eta = 0 \quad (19)$$

and the constant-area (or constant-volume for three-dimensional problems) variation

$$J = \text{constant} . \quad (20)$$

It is easy to derive the same elliptic equations (10) as Euler's equations of the minimization problem

$$\text{Min } I \quad (21)$$

without constraints. Constraints may be imposed by applying penalty methods.

It is clear that the extension of grid generation by elliptic differential equations to three-dimensional domains is very straightforward as Mastin and Thompson [18] demonstrated. A similar three-dimensional extension is given by Ghia and Ghia [19] using the cylindrical coordinate system. Another advantage of the present approach is that the concept of adaptivity to increase accuracy or to capture discontinuous lines/surfaces such as shock fronts, is easily applicable to the grid-generation method using elliptic equations. Many beautiful results of adaptive grid generation for finite difference methods in computational fluid mechanics can be found in the two proceedings edited by Thompson [1] and Ghia and Ghia [6]. However, it is not clear whether the adaptive methods described in these proceedings are related to some of the mathematical studies on numerical error analysis. The adaptive methods for finite elements, such as provided by Babuška, have a rather complete theoretical background based on error analysis by either refinement of elements or increase of the degrees of polynomials for interpolation. The adaptive methods used in these two proceedings

seem to be more primitive. Roughly speaking, after finding the high-gradient portion of the pressure distribution, adaptation schemes are introduced by defining appropriately the source terms P and Q , or the function D in the elliptic equations stated above. It seems to be true that there is no established form for adaptation. Therefore, in this paper, it is intended to consider this elliptic differential equation method as *just one of grid-generation methods* independent of the adaptive methods discussed later. That is, grids generated by this method would be regarded as initial grids on which adaptive methods are applied while stress/flow analysis is performed.

Difficulties in elliptic grid generation can be found when the boundary or interface condition is set up for solving the differential equations. If the outside boundary of a structure is already specified, the choice of boundary conditions is sometimes not quite obvious. This yields a possible unpopularity of the elliptic differential equation method for grid generation. For rather simple domains often found in fluid-mechanics problems, specification of the boundary condition may not be difficult. But, for example, the domain for heat-conduction analysis of the head part of a piston of an engine shown in Fig. 2 (in which the grid is obtained

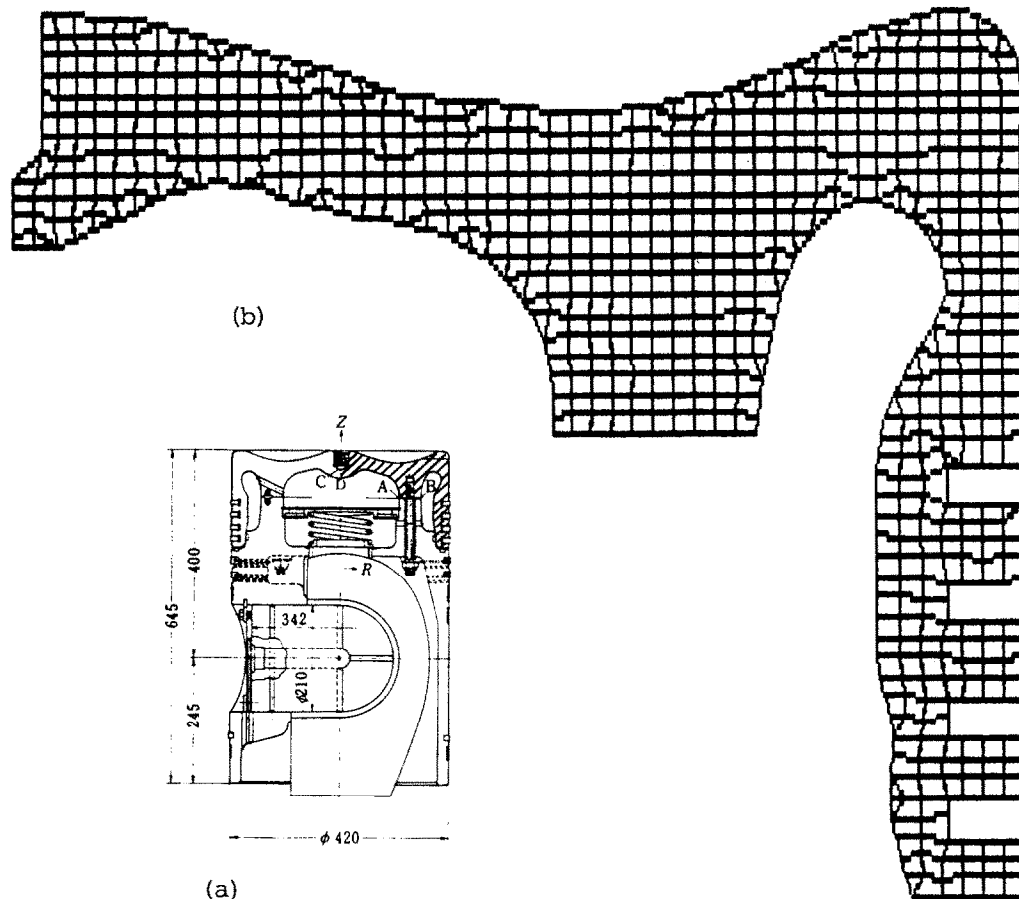


Fig. 2. A grid generated by the algebraic integer method (by T. Kotil). (a) Piston of an engine. (b) Head part discretized into finite elements.

by the algebraic integer method discussed in the next subsection) would yield very difficultly to the definition of an adequate boundary condition for the elliptic differential equations. Furthermore, *different boundary conditions yield different grids*. This means that the method requires a high level of understanding and experience. Thus, the method of elliptic differential equations cannot be used as a general-purpose grid-generation method by less experienced engineers and students although, if the appropriate boundary conditions are specified, grids generated are rather artistic and beautiful.

The elliptic differential equations in, e.g., (10) are first discretized by finite difference methods since the domain for the equations is, in general, regular. That is, the domain is rectangular, triangular, or circular. In any case, it is possible to set up a grid-generation domain simple enough for application of finite difference methods without any boundary adjustment/interpolation. For simplicity, let us assume that, in this paper, the domain for grid generation is rectangular. The first and second derivatives are approximated by the central difference scheme

$$x_{\xi} = (x_{i+1,j} - x_{i-1,j})/2 \Delta \xi, \quad x_{\xi\xi} = (x_{i+1,j} - 2x_{i,j} + x_{i-1,j})/(\Delta \xi)^2, \text{ etc.}$$

Here, $x_{i,j}$ means the value of x at the point $(i \Delta \xi, j \Delta \eta)$, $\Delta \xi$ is the increment in the ξ -direction, and $\Delta \eta$ is the increment in the η -direction. After the discretization of the elliptic differential equations, these discrete equations are then solved by an iterative method such as the successive overrelaxation (SOR) method. Since the differential equations are elliptic, convergence of the SOR method is, in general, assured. It is also noted that fast convergence of the iterative scheme applied is not required when the grids are reasonably smooth. Thus, the shortcoming of the SOR method that the speed of convergence becomes very slow near the solution, is no longer critical. In the SOR method the expected rapid convergence in the early stages of the iteration process will be almost sufficient to solve the discrete equations for grid generation. For (12), the SOR method becomes as follows:

$$\begin{aligned} x_{i,j}^{n+1/2} = & (1 - \omega)x_{i,j}^n + \omega\{(A/E)_{i,j}(x_{i+1,j}^n + x_{i-1,j}^{n+1}) \\ & - (B/E)_{i,j}(x_{i+1,j+1}^n - x_{i+1,j-1}^n - x_{i-1,j+1}^n + x_{i-1,j-1}^n) \\ & + (C/E)_{i,j}(x_{i,j+1}^n + x_{i,j-1}^n) + (F/E)_{i,j}\}, \end{aligned} \quad (22)$$

$$x_{i,j}^{n+1} = \text{projection of } x_{i,j}^{n+1/2}, \quad (23)$$

where the superscript indicates the number of the iteration, and

$$\begin{aligned} A_{i,j} = \frac{\alpha_{i,j}}{\Delta \xi^2}, \quad B_{i,j} = \frac{\beta_{i,j}}{2 \Delta \xi \Delta \eta}, \quad C_{i,j} = \frac{\gamma_{i,j}}{\Delta \eta^2}, \quad E_{i,j} = 2 \left(\frac{\alpha_{i,j}}{\Delta \xi^2} + \frac{\gamma_{i,j}}{\Delta \eta^2} \right), \\ F_{i,j} = J_{i,j} \left\{ P_{i,j} \frac{(x_{i+1,j}^n - x_{i-1,j}^{n+1})}{2 \Delta \xi} + Q_{i,j} \frac{(x_{i,j+1}^n - x_{i,j-1}^{n+1})}{2 \Delta \eta} \right\}, \quad (24) \\ \alpha_{i,j} = \left(\frac{x_{i,j+1}^n - x_{i,j-1}^{n+1}}{2 \Delta \eta} \right)^2 + \left(\frac{y_{i,j+1}^n - y_{i,j-1}^{n+1}}{2 \Delta \eta} \right)^2, \end{aligned}$$

$$\beta_{i,j} = \frac{(x_{i+1,j}^n - x_{i-1,j}^{n+1})(y_{i+1,j}^n - y_{i-1,j}^{n+1}) - (x_{i,j+1}^n - x_{i,j-1}^{n+1})(y_{i,j+1}^n - y_{i,j-1}^{n+1})}{4 \Delta \xi \Delta \eta},$$

$$\gamma_{i,j} = \left(\frac{x_{i+1,j}^n - x_{i-1,j}^{n+1}}{2 \Delta \xi} \right)^2 + \left(\frac{y_{i+1,j}^n - y_{i-1,j}^{n+1}}{2 \Delta \xi} \right)^2.$$

For y , a similar iterative scheme can be obtained. Here projection means that if either the orthogonality condition or the constant-volume condition is imposed, then the SOR method must be modified by applying the concept of projection to satisfy given constraints.

Now let us look at several examples of grid generation by the elliptic differential equation method. The first example is the discretization of the left half of a notched thin elastic plate. Since stress singularity can be expected at the corner point of the notch, refined elements must be obtained in its neighborhood, as shown in Fig. 3. It is noted, however, that only one element is connected to the singular point, as was the case in the grid obtained by the conformal mapping method. Fig. 4 shows the grids computed by the elliptic differential equation method for an analysis of large-deformation elastic-plastic metal-forming processes. Since the deformation is extremely large, remeshing is performed at three stages. In each grid, since corner singularity is expected, very refined elements are set up near the corners. In the finite element approximation of the elastic-plastic deformation, rectangular elements are desirable in order to avoid the so-called locking phenomenon and to obtain soft enough results. Thus, the orthogonality condition of the grid is imposed in this example when possible. Note that, if exact orthogonality is required, the iterative scheme used by the SOR method may not converge. Thus, we only expect orthogonality when we can impose it. Fig. 5

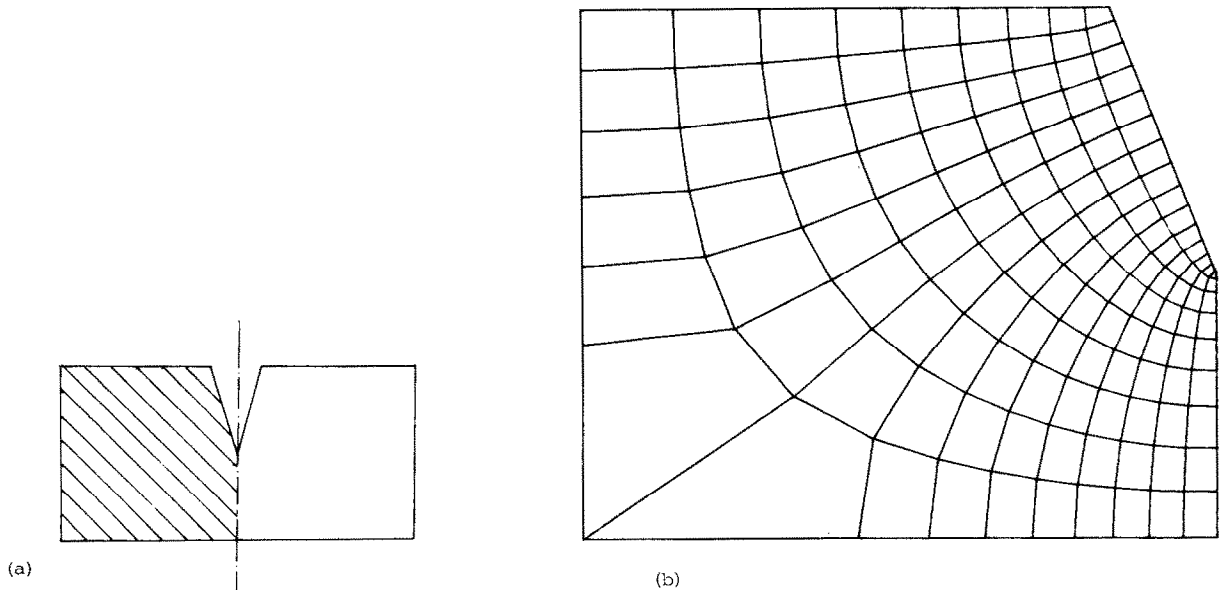


Fig. 3. A grid generated by the elliptic differential equation method (by H.C. Rodrigues). (a) Notched thin elastic plate. (b) Grid generated.

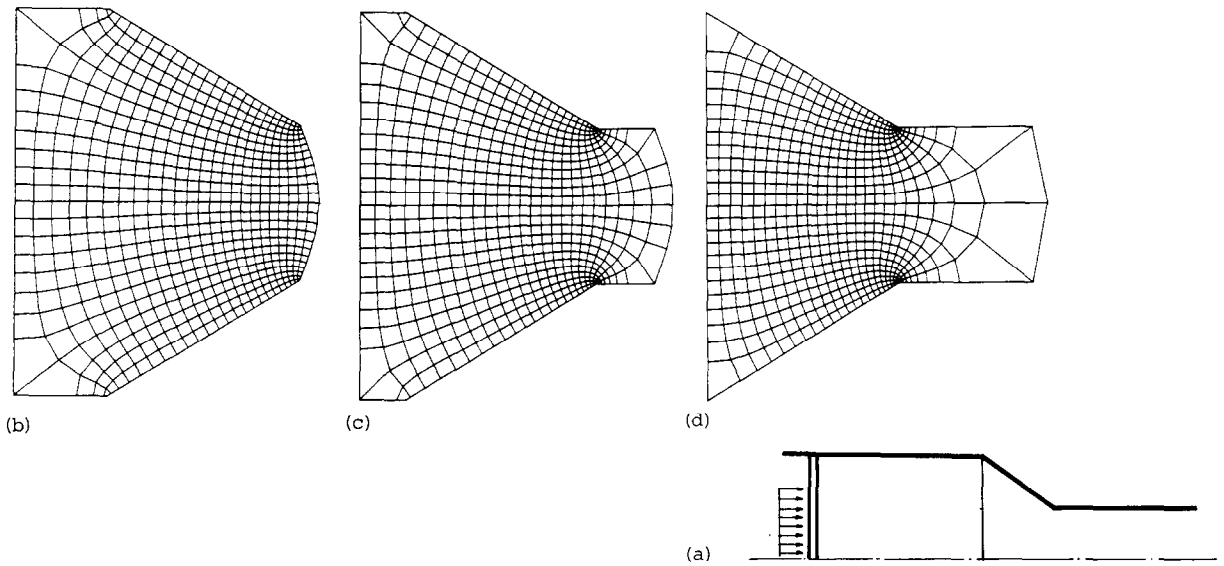


Fig. 4. Remeshed grids at three deformation stages by the elliptic differential equation method (by H.C. Rodrigues). (a) Domain of an extrusion process. (b) First remeshed grid. (c) Second remeshed grid. (d) Third remeshed grid.

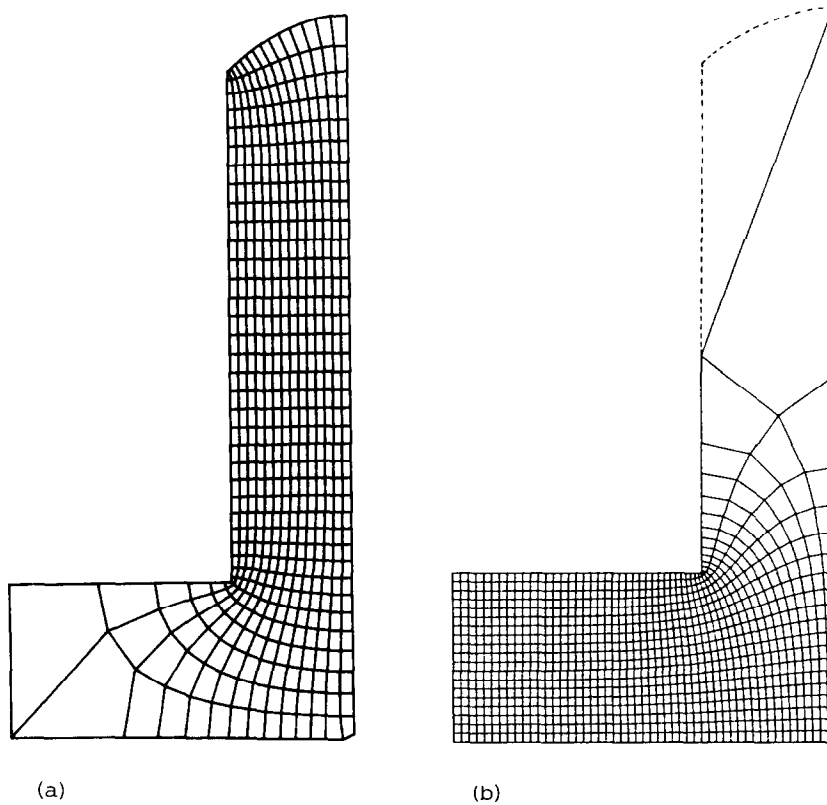


Fig. 5. Grids generated for a reverse extrusion problem by the elliptic different equation method with different boundary conditions (by H.C. Rodrigues). (a) Grid with boundary condition 1. (b) Grid with boundary condition 2.

shows two different grids computed by using two different boundary conditions on the same domain for elastic-plastic analysis of a reverse extrusion process. This example clearly shows that different boundary conditions yield different grids. Furthermore, it is very difficult to judge which one is better than the other. This characteristic makes the elliptic differential equation method difficult to apply to automatic remeshing of a domain under large deformation in the analysis of metal-forming processes. Although the grids obtained are extremely smooth, this method may not be an appropriate general-purpose method.

3.3. Algebraic methods

Although there are many algebraic grid-generation methods, only two will be discussed in this paper. The first one is the algebraic serendipity-mapping method, and the other is the algebraic integer method.

The algebraic serendipity-mapping method seems to be the oldest and the most popular among users of finite element methods in both areas of solid/structural mechanics and fluid mechanics. The basic idea is that a given structure is first decomposed into several blocks which will be identified with possibly curved-edge triangular or quadrilateral serendipity elements whose geometry can be characterized by the nodal points on their boundaries. Since any finite element can be generated by a mapping from the master element, its size is normalized by $(-1, 1) \times (-1, 1)$ in the coordinate system (ξ, η) for two-dimensional problems. Any point (x, y) in an element in the physical coordinate system can be mapped from the corresponding point (ξ, η) in the master element by the following algebraic equations

$$x = \sum x_{\alpha} N_{\alpha}(\xi, \eta), \quad y = \sum y_{\alpha} N_{\alpha}(\xi, \eta). \quad (25)$$

Here (x_{α}, y_{α}) are the coordinates of nodes in a serendipity element in the physical coordinate system, and $N_{\alpha}(\xi, \eta)$ are the shape functions of the serendipity element in the master element. The idea of setting up two coordinate systems for physical description and grid generation is similar to that of the elliptic differential equation method. But in this case no differential equations need to be solved to define grids. Mere performance of the algebraic mapping (25) provides the grid coordinates. Because of this, the resulting grids may not be very smooth. Anyhow, after producing grids in the master element, these grid points are transformed into the physical plane. The element connectivities of the finite elements generated in the master element are not even changed after the transformation to the physical plane. Defining a grid in the master element is an easy matter, since it is a square domain.

This is a very simple method to generate grids, and thus, it is necessary to pay a certain cost. As mentioned above, the resulting grids may not be smooth. For a complex domain, we may need to introduce many serendipity blocks unless various higher-order serendipity master elements are applied. This leads to many input data which must be supplied by the users. A possible remedy is to apply grid-smoothing and boundary-projection schemes. A very popular grid smoothing can be obtained using the idea of Winslow [14]. If a node, say n , is shared with elements $\Omega_1^n, \Omega_2^n, \dots, \Omega_N^n$, then the coordinates (x_n, y_n) are modified by

$$x_n = \left(\sum x_{c\alpha} w_{\alpha} \right) / \sum w_{\alpha}, \quad y_n = \left(\sum y_{c\alpha} w_{\alpha} \right) / \sum w_{\alpha}, \quad (26)$$

where $(x_{c\alpha}, y_{c\alpha})$ are the coordinates of the centroid of element E_α^n , $\alpha = 1, 2, \dots, N$, and w_α are weights appropriately defined for averaging. If $w_\alpha = \text{area of } \Omega_\alpha^n$, then the repeated application of (26) yields almost a uniform grid. If refined elements are desired in a certain region, then larger values for the weighting factors w_α must be given in the corresponding elements. It is also noted that after applying the smoothing scheme to the grid generated by the algebraic serendipity-mapping method, the smoothed grid becomes very similar to the one obtained by the elliptic differential equation method, because (26) is a very rough approximation of the discrete Laplacian. Fig. 6 presents a comparison between two grids generated by the elliptic method and the algebraic serendipity-mapping method with smoothing.

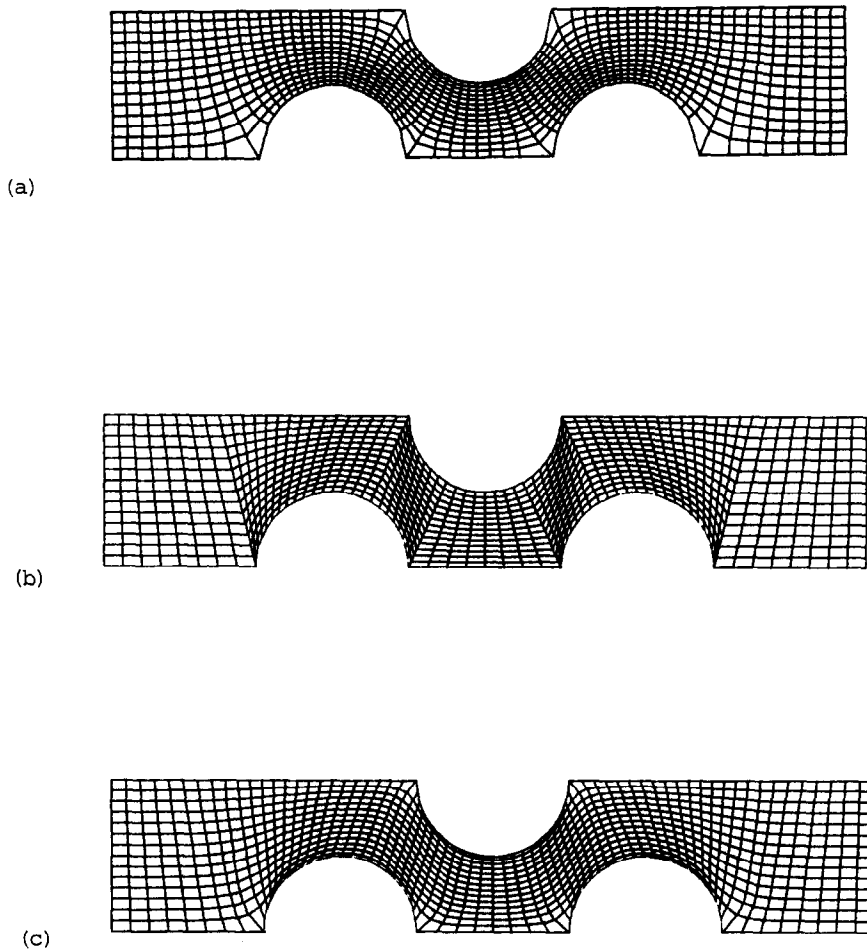


Fig. 6. Comparison between the grids obtained by the elliptic differential equation method and the algebraic serendipity mapping method with a smoothing scheme (by H.C. Rodrigues and T. Torigaki). (a) Grid obtained by the elliptic differential equation method. (b) Grid obtained by the algebraic serendipity-mapping method. (c) Grid obtained after applying a smoothing scheme to (b).

A boundary-projection scheme is introduced to project nodes onto a given curve identified with, for example, a spline. As mentioned above, the algebraic serendipity-mapping method may require many input data to characterize curved blocks which would be identified with higher-order serendipity elements. To avoid this, we first approximately decompose a given complex curved domain by several straight-edge blocks, i.e., by four-node serendipity elements. After generating a grid, nodes on the grid boundary are projected onto the originally curved boundary. A boundary-projection scheme can be found in [20]. Fig. 7 gives an example of boundary projection. It is worthwhile to mention that the grid in Fig. 7 is obtained by a grid generator, ASGRID, which is based on the algebraic serendipity-mapping method with smoothing and boundary projection, suitable for IBM PCs.

Because of the decomposition of a given domain into several blocks, this method may not be suitable for automatic remeshing for large-deformation analysis. In the process of deformation these blocks, defined at the initial undeformed configuration, may become nonconvex or too much distorted so that their shape yields an inadequate grid. This is the third shortcoming of the algebraic serendipity-mapping method for adaptive grid generation.

Now we will discuss the last grid-generation method, the algebraic integer method, based on the work by Thacker, Gonzalez, and Putland [20], that uses a triangular grid. The basic procedure of this method is as follows:

Step 1. Define an integer net in which a given domain is contained. The net may be triangular or rectangular as shown in Fig. 8. Grid points in the net are identified with a pair of integers (i, j) which define the location.

Step 2. Define the boundary of the domain and the interfaces of different materials by, for example, splines.

Step 3. Determine a zigzag grid, which contains the given domain, by using the indices $L_{i,j}$ defined by

$$L_{i,j} = \begin{cases} 1, & \text{if point } (i, j) \text{ is in the given domain,} \\ 0, & \text{if point } (i, j) \text{ is outside the domain.} \end{cases} \quad (27)$$

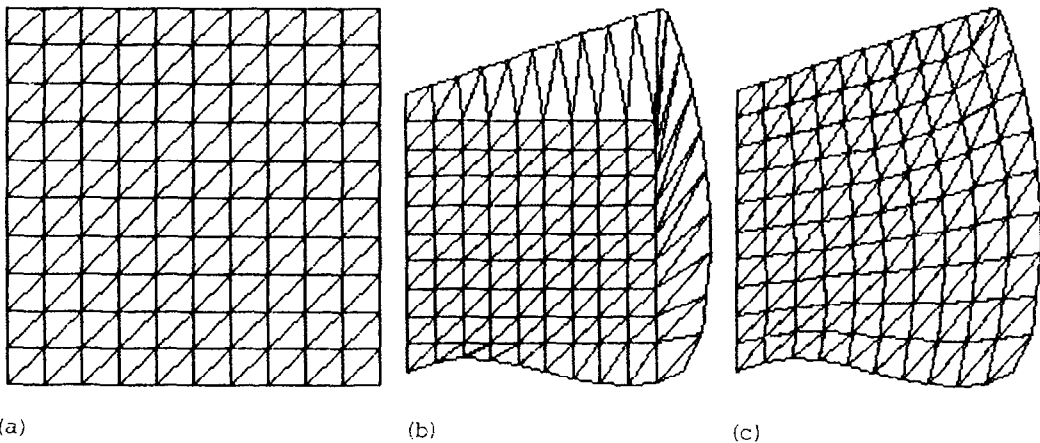


Fig. 7. Boundary projection method. (a) Originally assumed square domain. (b) Boundary projection of boundary nodes to given curves. (c) Final grid after applying a smoothing scheme.

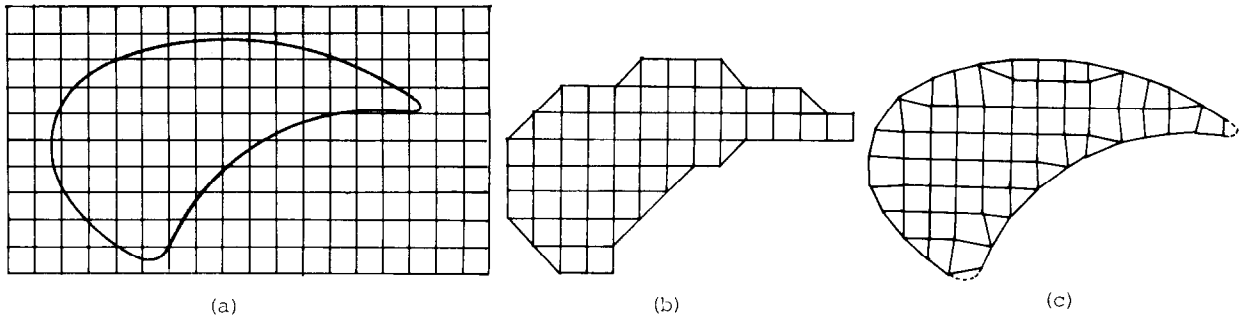


Fig. 8. Basic processes of the algebraic integer method. (a) Set-up of a given domain on an integer net. (b) Elimination of unnecessary portion. (c) Boundary projection to the given boundary.

This process automatically defines the grid points on the boundary. Then find nodes which represent the interfaces of different materials if necessary.

Step 4. Project the grid points of the zigzag boundary and the interfaces onto the real boundaries by, for example, the bisection method.

Step 5. Adapt and smooth the generated grid. Adaptation may be performed by coarsening or refining elements. Smoothing is done in a similar way as discussed above.

Thacker et al. [20] give a fine example of grid generation by this algebraic integer method. The method Shephard [21] uses to generate grids may be considered as a variation of the one introduced by Thacker et al. Thacker's method has the following advantages.

(1) It is extremely general and extendable to three-dimensional problems as well as to the elliptic differential equation method and the algebraic serendipity-mapping method. However, it is not necessarily bothered with defining appropriate boundary conditions as is the elliptic differential equation method. Once the boundary and interface of a given body are defined by some representative nodes for the spline definition, this method can generate essentially uniform grids without any difficulty. Thus it is applicable, having automatic remeshing capabilities, for large-deformation analysis of metal-forming processes and for shape optimization of structures in which the domain is subjected to large remodeling during the process of optimization. Grid adaptation is also possible in this method.

(2) The grids generated by this method are basically the same as the ones generated by other persons if the same number of grid points and spacing are used. In this sense, it is universal.

One of the possible disadvantages is that this scheme requires many integers such as $L_{i,j}$ and (i, j) for defining grid points. Since many IF-statements are applied, the speed of grid generation may be slower. However, programs for this method are, at least for two-dimensional problems, considerably smaller.

It is worthwhile to mention that this method provides the best grid for large-deformation elastic-plastic problems in which a plastic locking phenomenon can be observed for certain finite elements, such as 4CST elements (triangular elements generated by connecting the diagonals of quadrilaterals), since very uniform, almost rectangular, quadrilaterals can be generated by this method in the plastic zone. In shape-optimization problems, this method must be accompanied by smoothing schemes which imply almost orthogonal elements to the

boundary optimized. Otherwise, some oscillatory behavior can be observed in the design boundary, see [22].

In Fig. 9 several examples of grids generated by the algebraic integer method are given. These are obtained by using AIGRID developed for IBM PCs. Note that AIGRID can also be executed by other, larger computers, but it is intended that this method be implemented even in microcomputers.

3.4. Additional functions for grid generation

In solids and structural mechanics, domains are, in general, quite complex. Thus, mere application of the grid-generation methods discussed above may not provide grids for finite element approximations. To overcome this difficulty, we need to introduce other functions to compensate the capability of the above grid-generation methods. A practical approach may be the combination of all the grid-generation methods after decomposing a given domain into several blocks as in the method of algebraic serendipity mapping. Each subdivided block is then discretized into finite elements by applying an appropriate grid-generation method independently of other blocks. After this process, the next step is to connect disjoint blocks discretized into finite elements. A typical setting is shown in Fig. 10 in which two disjoint blocks must be connected. This can be done by applying a function to connect two blocks. If there are functions to shift the location of nodes, to delete nodes and elements, to add nodes and elements, and to modify element connectivities, then the connection of two blocks would

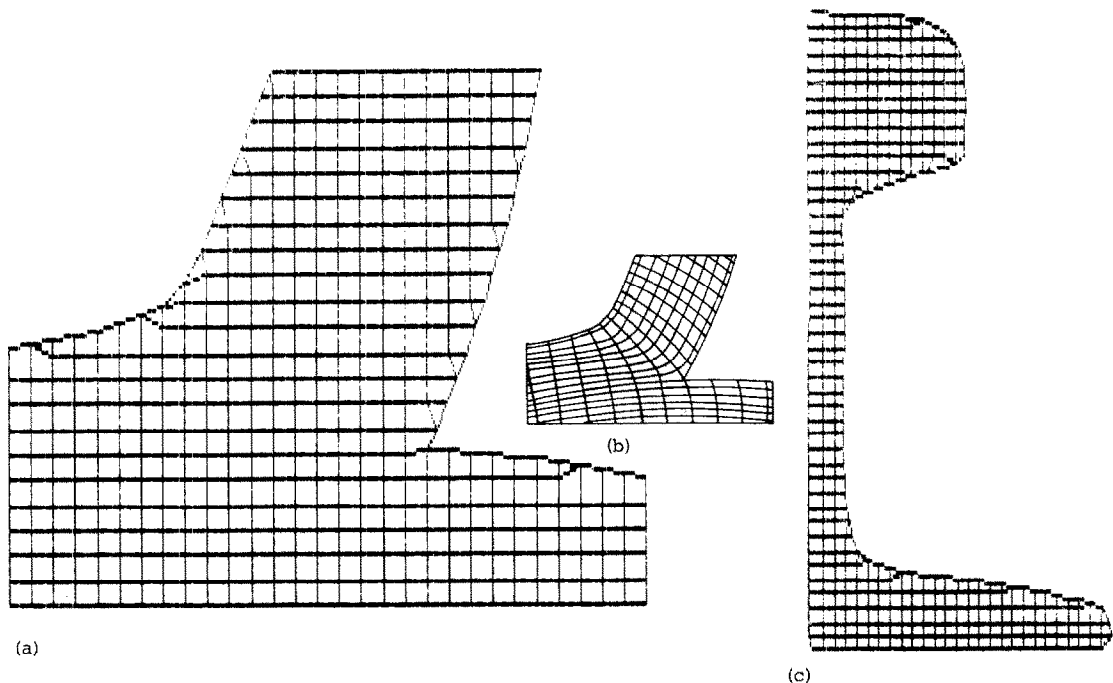


Fig. 9. Examples of grids obtained by the algebraic integer method (by T. Kotil). (a) Discretized domain for metal-cutting analysis. (b) An example of the metal-cutting process. (c) Right half of the rail road cross-section.

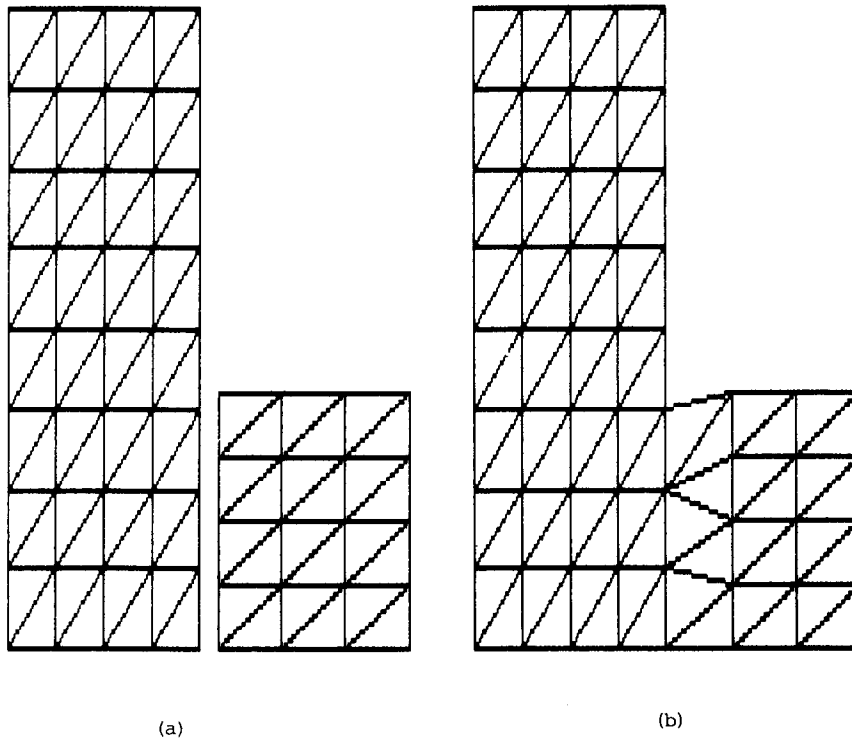


Fig. 10. Connection of two disjoint blocks. (a) Two disjoint blocks. (b) After connection.

be easy. This must be done interactively, while users look at the disjoint blocks on a graphic terminal. If nodes can be projected onto a specified curve/surface, making holes with an arbitrary shape becomes easy. Shifting and rotation of a block are also useful as well as enlargement and reduction of the size of a block. A renumbering scheme that minimizes the bandwidth is also helpful. Many other functions can be introduced for grid generation. To develop a user-friendly grid-generation capability effort should be endless.

4. Adaptive finite element methods

It is now desirable to control the approximation error in the finite element methods. Therefore, Babuška and his colleagues have introduced the concept of adaptive finite element methods which enable to predict the total amount of the finite element error and to provide information on how a finite element model must be refined to reduce the error to the level of a specified value. They have established various error indicators for both linear and nonlinear problems and have given a strong impact to practical engineers by showing how the mathematical theory of finite element methods can be applied to solve engineering problems. Here, details of the mathematical theory of adaptive finite element methods are not discussed. These can be found in a series of papers published by Babuška and his colleagues [23–27]. We will describe how adaptive methods are implemented to solve problems based on the idea of structural

optimization. As stated in Section 2, adaptive methods are characterized by the optimal design problem

$$\text{Minimize } (\text{Max } E_e), \quad (28)$$

grid design by e
 r -, h -, p -methods

where E_e is an error measure of finite element approximations in an finite element Ω_e ; the r -method means the node-relocation method which moves the node toward the optimal location; the h -method means that after finding elements suffused by a large amount of error these are refined into much smaller elements; and the p -method introduces higher-order polynomials for the shape functions of elements containing a large amount of finite element approximation error. The necessary condition for the grid optimization problem (28) is

$$E_e = \text{constant} \quad \text{for } e = 1, \dots, \text{NE}, \quad (29)$$

where NE is the total number of elements. In the r - and p -method, NE is fixed, while it increases in the h -method. The most interesting question in the practical application of adaptive methods is how an appropriate error measure E_e is defined in each finite element. Here, we basically follow to the discussion in [28–31]. For simplicity, let us consider the solution of linear plane-elasticity finite element approximation problems which uses four-node quadrilateral isoparametric elements. More precisely, let us consider the problem

$$\mathbf{u} \in \mathbf{V}: \quad a(\mathbf{u}, \mathbf{v}) = f(\mathbf{v}), \quad \mathbf{v} \in \mathbf{V}_0, \quad (30)$$

where

$$a(\mathbf{u}, \mathbf{v}) = \sum a_e(\mathbf{u}, \mathbf{v}), \quad f(\mathbf{v}) = \sum f_e(\mathbf{v}),$$

$$a_e(\mathbf{u}, \mathbf{v}) = \int_{\Omega_e} e_{ij}(\mathbf{v}) D_{ijkl} e_{kl}(\mathbf{u}) \, d\Omega, \quad f_e(\mathbf{v}) = \int_{\Omega_e} \mathbf{v} \cdot \mathbf{f} \, d\Omega + \int_{\Gamma \cap \partial\Omega_e} \mathbf{v} \cdot \mathbf{t} \, d\Gamma. \quad (31)$$

\mathbf{V} is the admissible set of displacements satisfying the essential boundary condition; the linear space \mathbf{V}_0 is the corresponding homogeneous version; $a(\cdot, \cdot)$ is the internal virtual work in a given structure; $a_e(\cdot, \cdot)$ is the corresponding value in each finite element; $f(\cdot)$ is the potential due to the applied body force \mathbf{f} and the traction \mathbf{t} on a part Γ of the boundary $\partial\Omega$ of the structure Ω ; $f_e(\cdot)$ is the corresponding value in each finite element; and $\partial\Omega_e$ is the boundary of an element Ω_e . Suppose that \mathbf{V}_h and \mathbf{V}_{0h} are the finite element approximations of \mathbf{V} and \mathbf{V}_0 using four-node quadrilateral elements. Then, the finite element approximation of (30) becomes

$$\mathbf{u}_h \in \mathbf{V}_h: \quad a(\mathbf{u}_h, \mathbf{v}_h) = f(\mathbf{v}_h), \quad \mathbf{v}_h \in \mathbf{V}_{0h}. \quad (32)$$

It follows from (30) and (32) that

$$a(\mathbf{u}_h - \mathbf{u}, \mathbf{u}_h - \mathbf{u}) \leq a(\mathbf{u}_h - \mathbf{u}, \mathbf{v}_h - \mathbf{u}), \quad \mathbf{v}_h \in \mathbf{V}_h. \quad (33)$$

If the error of the finite element approximation is defined by

$$\varepsilon = \sqrt{a(\mathbf{u}_h - \mathbf{u}, \mathbf{u}_h - \mathbf{u})},$$

and if the interpolation error in V_h is defined by

$$\varepsilon_I = \sqrt{a(v_h - u, v_h - u)},$$

where v_h is the interpolation of the solution u , then the approximation error of the finite element method is bounded by the interpolation error, that is, the following inequality holds:

$$\varepsilon \leq \varepsilon_I. \quad (34)$$

This suggests an error measure in each finite element Ω_e defined by

$$E_e = \sqrt{a_e(v_h - u, v_h - u)}, \quad (35)$$

the physical meaning of which is the square of twice of the strain energy in a finite element Ω_e due to the difference of the true displacement and its finite element approximation. Note that

$$\varepsilon_I = \sqrt{\sum_e E_e^2} \leq \sqrt{NE} \cdot \text{Max}_e E_e. \quad (36)$$

Now, the question is how the difference between the solution and its interpolation is estimated for the unknown solution. To do this, it is worthwhile to mention that only the first derivatives of the difference are required to compute the error measure defined in (35).

Suppose that *the second derivatives of all the components of the true displacement u are constant in each finite element*. Their values need not be the same in different elements. Define

$$\begin{aligned} A &= \partial^2 w / \partial x^2, & B &= \partial^2 w / \partial x \partial y, & C &= \partial^2 w / \partial y^2, \\ x^i &= \{x_1, x_2, x_3, x_4\}, & y^i &= \{y_1, y_2, y_3, y_4\}, \\ X^i &= \{X_1, X_2, X_3, X_4\}, & Y^i &= \{Y_1, Y_2, Y_3, Y_4\}, \\ L_s^i &= \frac{1}{4}\{-1, 1, 1, -1\}, & L_t^i &= \frac{1}{4}\{-1, -1, 1, 1\}, & h^i &= \frac{1}{4}\{1, -1, 1, -1\}, \\ J_{11} &= (L_s \cdot x) + (h \cdot x)\eta, & J_{12} &= (L_s \cdot y) + (h \cdot y)\eta, \\ J_{21} &= (L_t \cdot x) + (h \cdot x)\xi, & J_{22} &= (L_t \cdot y) + (h \cdot y)\xi, & J &= J_{11}J_{22} - J_{12}J_{21}, \end{aligned} \quad (37)$$

where (x_α, y_α) are the coordinates of the four corner nodes of a quadratic element in the physical coordinate system, and (X_α, Y_α) are four nodes inside of the element, say, the four nodes corresponding to the 2×2 Gaussian integration points in the master element. It is noted that if an element is a parallelogram, the terms $h \cdot x$ and $h \cdot y$ become identically zero. Under the assumption stated above, the first derivatives of the difference of the function w and its interpolation w_h by the four-node quadrilateral element can be written as

$$\begin{aligned} \partial(w_h - w) / \partial x &= (1/J)[-J_{22}(J_{11}^2 A + 2J_{11}J_{12}B + J_{12}^2 C)\xi + J_{12}(J_{21}^2 A + 2J_{21}J_{22}B + J_{22}^2 C)\eta \\ &\quad + \{(1 - \eta^2)L_t \cdot y(L_t - \xi h) + (1 - \xi^2)L_s \cdot y(-L_s + \eta h)\} \\ &\quad \cdot \{h \cdot x(AX + BY) + h \cdot y(BX + CY)\} \\ &\quad + \{\xi(1 - \eta^2)(L_t - \xi h) + \eta(1 - \xi^2)(-L_s + \eta h)\} \\ &\quad \cdot \{h \cdot x h \cdot y(AX + BY) + h \cdot y h \cdot y(BX + CY)\}], \end{aligned}$$

$$\begin{aligned}
\partial(w_h - w)/\partial y = & (1/J)[-J_{21}(J_{11}^2 A + 2J_{11}J_{12}B + J_{12}^2 C)\xi + J_{11}(J_{21}^2 A + 2J_{21}J_{22}B + J_{22}^2 C)\eta \\
& + \{(1 - \eta^2)L_t \cdot \mathbf{x}(-L_t + \xi \mathbf{h}) + (1 - \xi^2)L_s \cdot \mathbf{x}(L_s - \eta \mathbf{h})\} \\
& \cdot \{\mathbf{h} \cdot \mathbf{x}(A\mathbf{X} + B\mathbf{Y}) + \mathbf{h} \cdot \mathbf{y}(B\mathbf{X} + C\mathbf{Y})\} \\
& + \{\xi(1 - \eta^2)(-L_t + \xi \mathbf{h}) + \eta(1 - \xi^2)(L_s - \eta \mathbf{h})\} \\
& \cdot \{\mathbf{h} \cdot \mathbf{x} \mathbf{h} \cdot \mathbf{x}(A\mathbf{X} + B\mathbf{Y}) + \mathbf{h} \cdot \mathbf{x} \mathbf{h} \cdot \mathbf{y}(B\mathbf{X} + C\mathbf{Y})\}, \tag{38}
\end{aligned}$$

where (ξ, η) are the normalized coordinates in the master element. This means that if the second derivatives of the solution are known, the interpolation error can be expressed explicitly in terms of (ξ, η) . Thus, the error measure defined above can be computed. If an element is a parallelogram, then the computation of the interpolation error becomes very simple since $\mathbf{h} \cdot \mathbf{x} = 0$ and $\mathbf{h} \cdot \mathbf{y} = 0$, that is, the last two lines are identically zero in each partial derivative. On the other hand, if a parallelogram is considerably distorted, then the terms in the second and third lines in the interpolation error become large in the region where large strain is expected, since $(A\mathbf{X} + B\mathbf{Y})$ and other similar terms are basically strain components in an element. This suggests that regular refined finite elements must be set up in the neighborhood of singular points. Here regularity means that an element is close to a rectangle or parallelogram. Otherwise, the error contributions from the terms in the second and third lines would become quite large. This means that grids generated by conformal mappings are very appropriate. Similarly, grids generated by the elliptic differential equation method with the orthogonality condition and by the algebraic integer method are suitable in the sense that the contributions from the grid distortion and high strain (i.e., stress) can be restricted to be small enough.

The last question in the evaluation of the error is how the second derivatives of the solution are estimated. One method is the application of the least-squares method to obtain continuous first derivatives of strain components from the ones computed from the finite element approximation at the Gauss integration points. Once the strain components are continuous, their derivatives can be obtained very easily using the routine to form element stiffness matrices.

In this paper, let us define an error indicator which indicates the amount of finite element approximation error

$$EI = \sqrt{\sum_e E_e^2 / a(\mathbf{u}_h, \mathbf{u}_h)}. \tag{39}$$

If EI becomes small enough, it may be possible to say that the approximation error is small enough.

Fig. 11 and Table 1 show the distribution and the amount of error, respectively, for a model linear elasticity problem. Some finite elements are deliberately distorted in order to see the effect of the distortion.

Now, let us briefly review the three adaptive methods, r -, h -, and p -methods, together with several applications.

4.1. Node-relocation (r -) method

This method has the longest history of all possible adaptive methods. Oliveira [32] studied

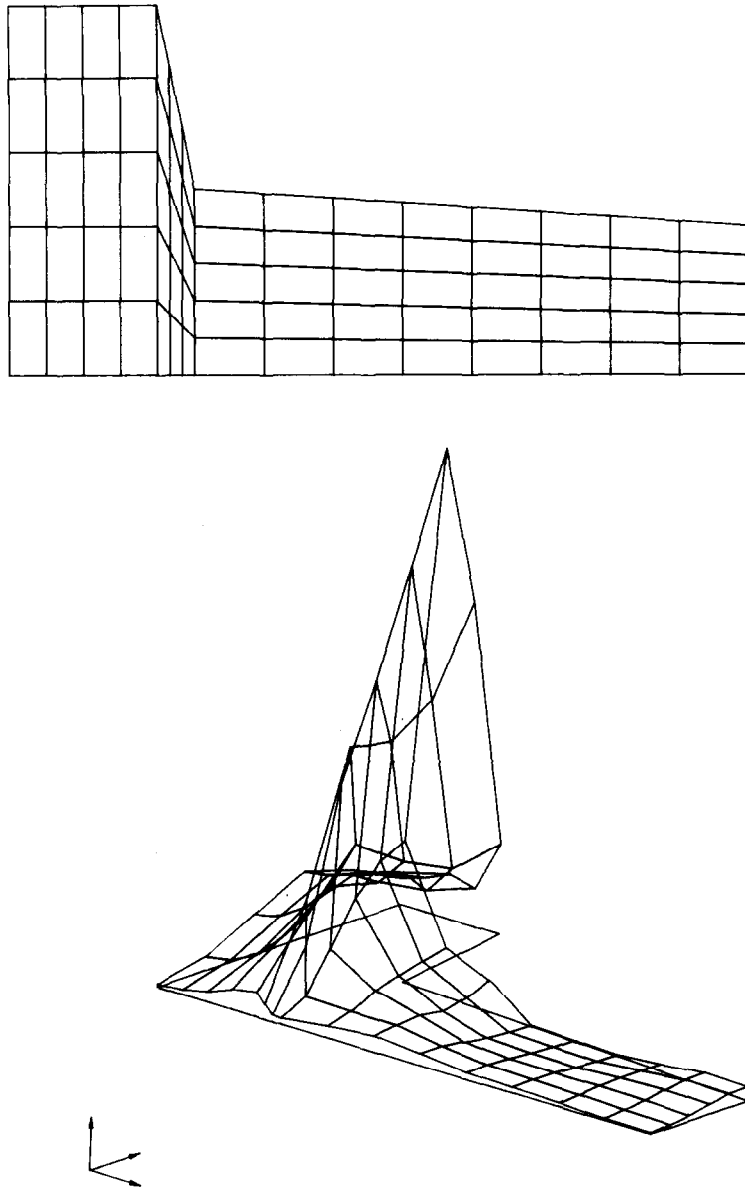


Fig. 11. Distribution of error measures in finite elements for an application of stress analysis of a fillet (by B. Koh). (a) Grid initially distorted on purpose. (b) Distribution of error measures.

the optimal location problem of a grid by considering nodal coordinates and seeking variables as well as the function satisfying a boundary value problem. After this, studies by Turcke and McNeice [33], Felippa [34, 35], Melosh and Marcal [36], and Shephard, Gallagher, and Abel [37] have been published. These are based on either the idea that the nodal points of a finite element model must be on the isothermal lines for heat-conduction problems, or the idea that each finite element must share an equal amount of the strain energy for linear elasticity problems. However, this approach does not yield the minimum amount of finite element approximation error, as shown by Babuška. Furthermore, if a linear fracture-mechanics

Table 1

Pointwise error value $\partial u/\partial x$ ($\times 10^{-5}$) and integrated error measure in the upper boundary elements ($|(e_{xx})_{e,i}|$ = absolute value of pointwise error of $\partial u/\partial x$ due to the line i in (38), E_e = integrated error measure in each element)

NE	$ (e_{xx})_{e,1} $	$ (e_{xx})_{e,2} $	$ (e_{xx})_{e,3} $	E_e
5	0.0	0.0	0.0	0.224
10	0.0	0.0	0.0	0.354
15	0.0	0.0	0.0	0.412
20	0.0	0.0	0.0	0.253
25	2.85	0.08	1.01	2.445
30	0.25	0.59	2.81	3.656
35	5.84	0.87	2.96	5.426
40	0.02	0.19	0.29	0.326
45	0.05	0.53	0.70	0.521
50	0.002	0.013	0.02	0.017
55	0.007	0.012	0.11	0.010
60	0.015	0.031	0.17	0.020
65	0.010	0.050	0.13	0.012
70	0.165	0.062	0.255	0.117
75	0.038	0.030	0.605	0.123

problem involving a crack is solved using the idea of equal distribution of the strain energy in each finite element, this adaptive method implies that zero area elements be connected to the crack tip. That is, all the nodes around the crack tip move to the crack tip and finally they share the same coordinates. Another difficulty involved in this approach is to find a moving-grid scheme which would not destroy the convexity of finite elements and which is free from element collapse.

For stationary heat-conduction problems and elastostic problems, the smoothing scheme (26) works reasonably well by taking the weights $w_e = E_e/\text{area of } \Omega_e$, where E_e is an error measure of an element Ω_e . For one-dimensional problems, this smoothing scheme yields the satisfaction of the necessary condition, $E_e = \text{constant}$, by repeated application. For two-dimensional problems, it can only be said that if all elements are rectangular and if the necessary condition is satisfied, the application of the smoothing scheme would not change the location of nodes by its application. That is, repeated application may not yield exact satisfaction of the necessary optimality condition for the grid-design problem. But it always reduces the maximum value of E_e at the beginning and does converge to a fixed grid which is very close to the optimal one. It is noted, however, that the smoothing scheme changes the location of nodes of irregular grids even though the necessary optimality condition, $E_e = \text{constant}$, is satisfied. This always allocates refined elements in the neighborhood of corner points whose domain angle is larger than 180 degrees, since the smoothing scheme is essentially an approximation of the discrete Laplacian. Because of this characteristic of smoothing, node relocation by (26) yields reasonable adaptivity for reducing the finite element error and for smoothing a given irregular grid, at the same time.

As stated above, the error measure introduced here is *the strain energy in an element due to the difference between the solution and its interpolation, but not the strain energy due to the finite*

element solution as used in other works. This choice assures reduction of the finite element error defined in this paper. Only if an improved grid is expected containing less error than the original grid, then there are many other error measures. For example, the L_2 -norm of the gradient of the stress tensor in an element can be a good choice for an error measure as well as the L_2 -norm of the gradient of the von Mises equivalent stress. For large-deformation elastic-plastic analysis, the L_2 -norm of the total plastic strain (for necking problems) and the L_2 -norm of the gradient of the total plastic strain (for extrusion problems) are meaningful error measures, see [38]. For shape-optimization problems of linearly elastic structures the L_2 -norm of the gradient of local criteria is applicable as an error measure for adaptation, see [22].

4.2. Element refinement (h -) method

The h -method, originally studied by Babuška, is the method in which, after finding elements in which a large amount of error is accumulated, refinement is performed by subdividing these elements into much small ones. For example, a square domain is divided into four elements, and an adequate error measure is computed as given in Table 2. The average error measure is 0.04. In this case, if the third element is divided into four elements of equal size, then each element generated must share a quarter of the error measure, i.e., 0.025. Since the third element is refined, four nodes are added to the model. Thus, the second and the fourth elements must be divided into three triangular elements, see Fig. 12. This yields that a triangular element generated from the second element has an error measure of 0.007 and that a triangular element from the fourth element has error measure 0.01. Therefore, the average error measure becomes 0.015, and the maximum error measure is reduced to 0.025 from 0.10. Using the adapted grid, analysis is performed again to compute the error measure in each finite element, and the subdivision process of finite elements is repeated. The question is how to decide which elements must be refined. One method is to find elements whose error measure is larger than kE_{av} , where k is a scale factor and E_{av} is average error measure. Another way is to pick up elements whose error measure is larger than the $M\%$ of the maximum value of error measure in the whole domain, or $M\%$ elements will be refined. In any case, after the h -adaptation the necessary optimality condition is never satisfied, but the absolute value of the difference of the maximum and minimum error measure decreases considerably as well as the value of the maximum error measure. That is, at the limit of the h -adaptation the necessary optimality condition is satisfied, and furthermore, the amount of finite element approximation error becomes zero!

Table 2
 h -method error measure

Element	Error measure	Area
1	0.01	0.25
2	0.02	0.25
3	0.10	0.25
4	0.03	0.25

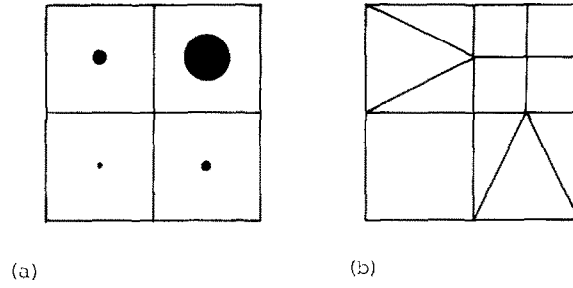


Fig. 12. An example of h -adaptation. (a) Original grid and distribution of the error measure. (b) Refined grid.

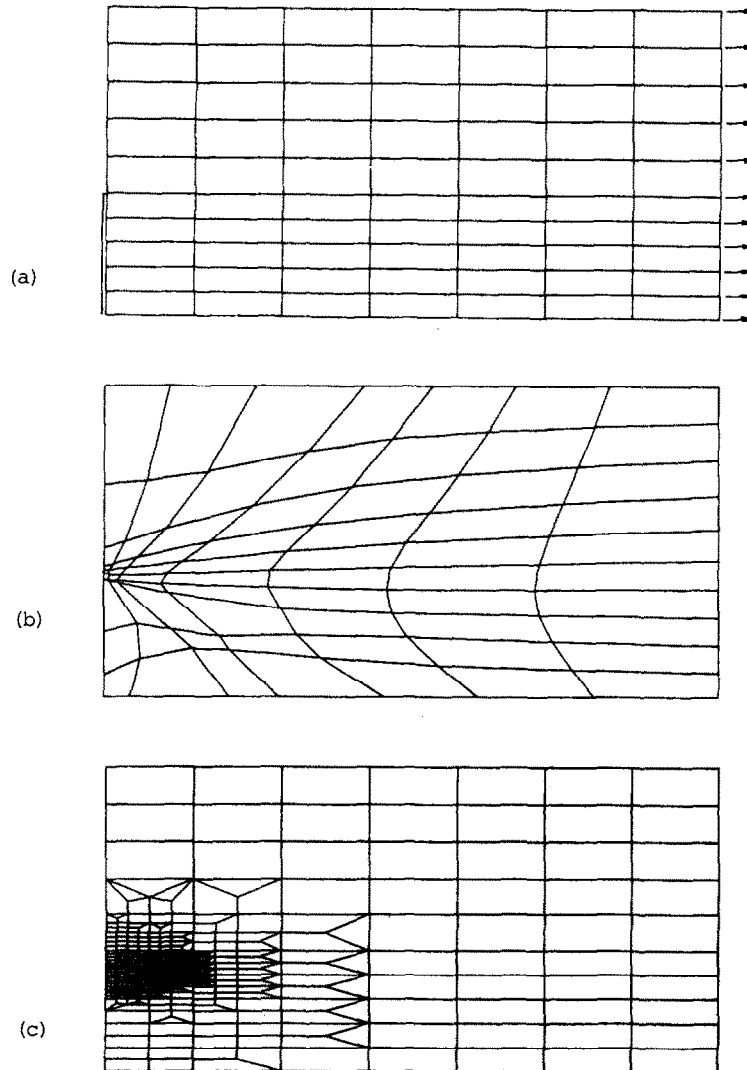


Fig. 13. Application of the r - and h -methods to fracture mechanics (by T. Torigaki). (a) Quarter part of a thin elastic plate containing a crack. (b) Grid obtained by the r -method. (c) Grid obtained by the h -method.

Fig. 13 shows an example of the application of the r - and h -methods to solve a tension problem of a thin linearly elastic plate with a crack subjected to the applied tensile traction at the end surfaces. Starting from the uniform grid for a quarter of the plate, the adapted grids obtained by the r - and h -methods are as shown in Fig. 13. It is clear that very refined grids are accumulated in the vicinity of the crack tip. Fig. 14 represents the distribution of the stress along the center line computed for the original, r -adapted, and h -adapted grids. The results for both adaptations are almost the same. This means that the r -method provides a result superior to the h -method in this problem, since the total number of degrees of freedom in the r -adapted grid is much less than in the h -adapted grid. Fig. 15 shows another application of the r - and h -adapted grid. Fig. 15 shows another application of the r - and h -methods to a problem of stress analysis of a gear tooth. Fig. 16 is an application to stress analysis of a thin shell structure.

It is noted that the above methods are applicable not only to solids but also to flows. Several applications to flow problems are presented in [30, 31]. Applications to free boundary problems of seepage flow and to shape-optimization problems are presented in [39] and [22] respectively. Extension to nonlinear problems, especially to metal-forming analysis can be found in [38, 40] together with a remeshing method.

4.3. Higher-order interpolation (p -) method

This method is based on the idea that higher-order polynomials are applied to interpolate a function in the elements whose error measure is large. That is, the grid itself is not modified,

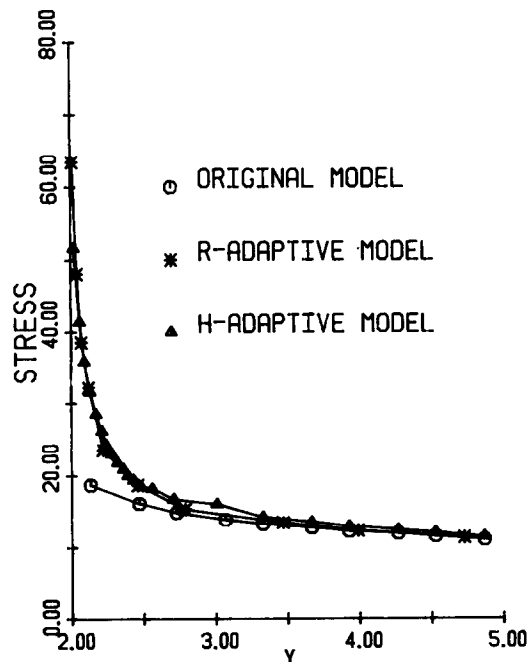


Fig. 14. Comparison of the stress distribution by the r - and h -methods.

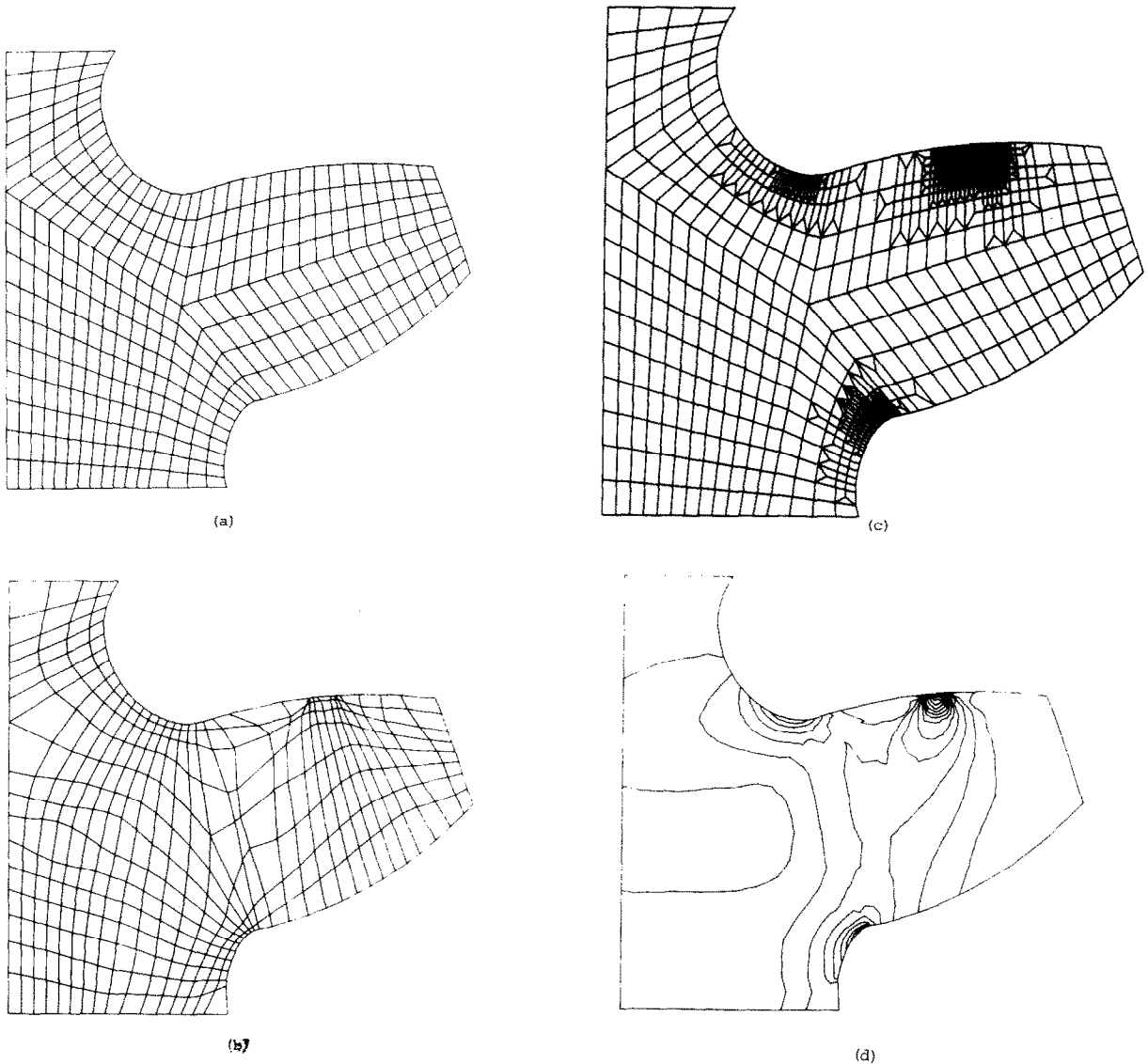


Fig. 15. Application of the r - and h -methods to the stress analysis of a gear tooth (by T. Torigaki). (a) Original grid. (b) Grid obtained by the r -method. (c) Grid obtained by the h -method. (d) Contours of the equivalent stress for the grid obtained by the r -method.

but the order of polynomials applied. This method have been studied mainly by Babuška, but also by Szabó and Mehta [41], Zienkiewicz, Kelly, Gago, and Babuška [42], and Oden et al. [43, 44]. In this paper, we shall only present an example, solved by ourselves, of a heat-conduction problem (see Fig. 17). Both r - and p -method are applied at the same time to improve the result. For the p -method, only linear and quadratic polynomials are applied. It is certain that quadratic polynomials are assigned in the elements near the two singular points. The amount of increase in the maximum value of the heat flux was more than 56% in this case, although the mere application of the r -method implies only 28% increase.

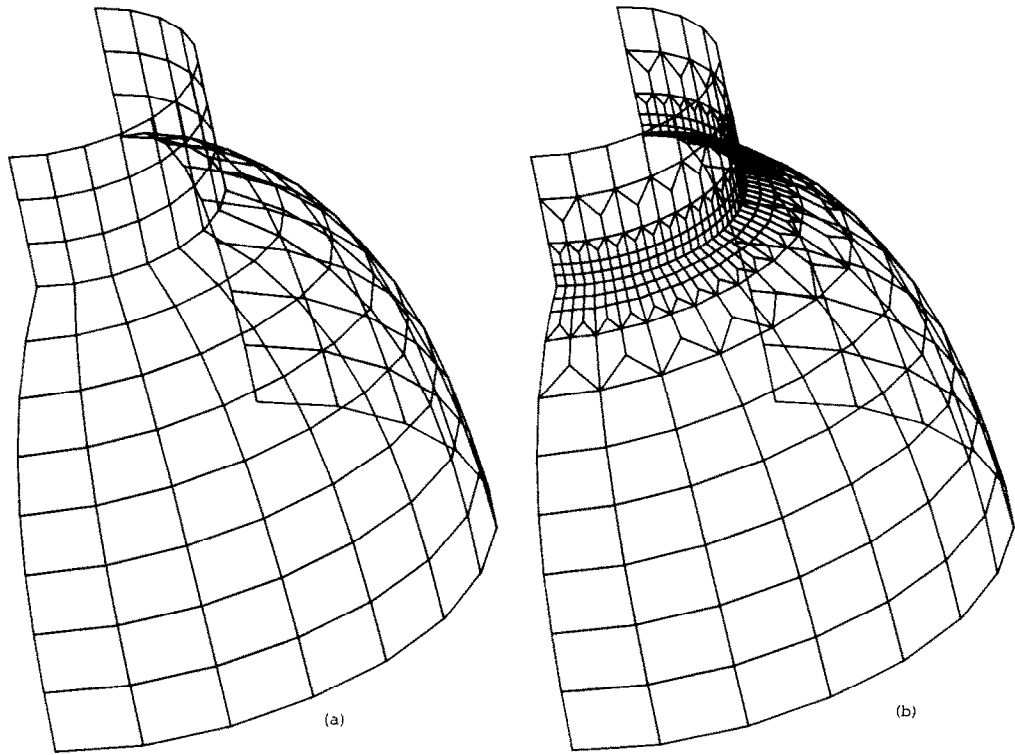


Fig. 16. Application of the h -method for shell stress analysis (by T. Torigaki). (a) Original grid. (b) Adapted grid.

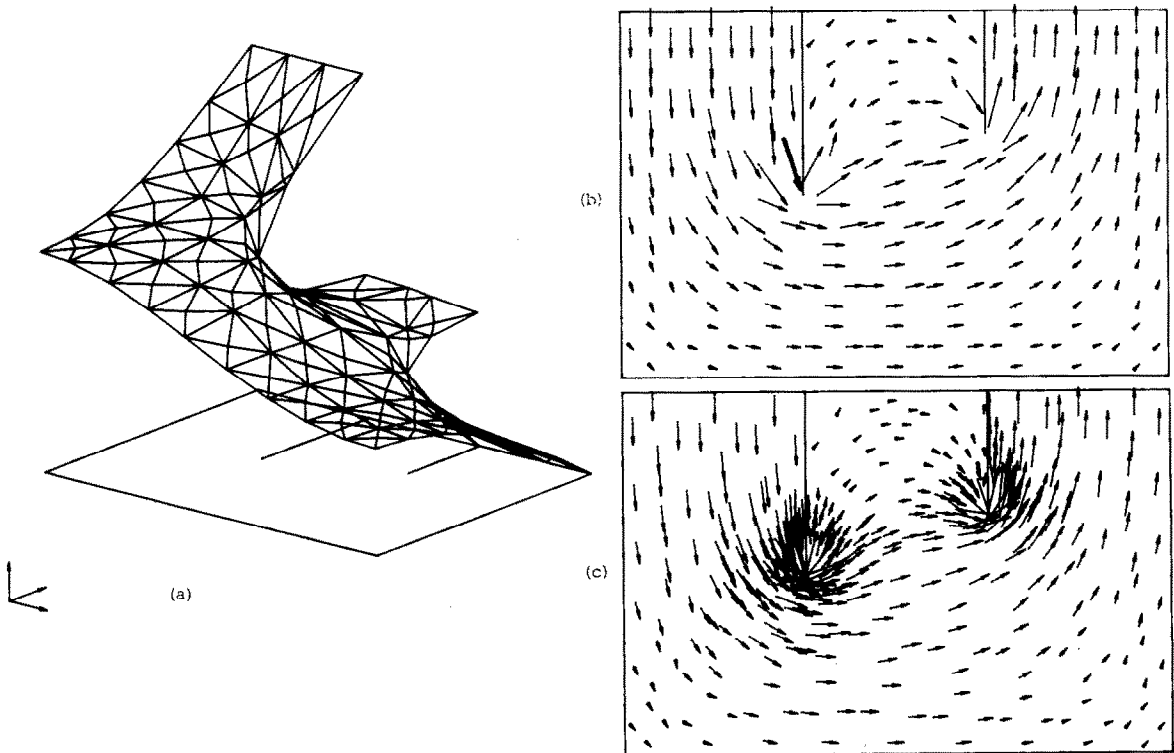


Fig. 17. Application of the r - together with the p -method to a heat-conduction problem (by T. Kishimoto). (a) Temperature distribution on the adapted grid. (b) Heat-flux vectors on the original uniform grid. (c) Heat-flux vectors on the adapted grid.

5. Conclusion

We have introduced the concept of adaptive grid design which requires the combination of numerical grid-generation methods and adaptive finite element methods. Applications of this idea to real problems have been described in order to demonstrate its usefulness. For two-dimensional problems this is almost at the stage of completion, but for three-dimensional problems it will take considerably more work and time.

Acknowledgment

The author is grateful to Toshikazu Torigaki, Helder C. Rodrigues, Temel Kotil, Byeong C. Koh, and Toshikazu Kishimoto who provided the computational results and developed the computer program ADGRID for adaptive grid design which includes both grid-generation methods described above and adaptive finite element methods. The author has been partially supported during the present work by Daewoo Heavy Industries and NASA Lewis Research Center through Grant NAG3-388. He expresses his appreciation for these supports.

References

- [1] J.F. Thompson, ed., *Numerical Grid Generation* (Elsevier, Amsterdam, 1982).
- [2] J.F. Thompson, Z.U.A. Warsi and C.W. Mastin, Boundary-fitted coordinate systems for numerical solution of partial differential equations—a review, *J. Comput. Phys.* 47 (1982) 1–108.
- [3] M.S. Shephard, ed., *Finite Element Grid Optimization*, ASME Special Publication PVP-38 (ASME, New York, 1979).
- [4] I. Babuška, J. Chandra and J.E. Flaherty, eds., *Adaptive Computational Methods for Partial Differential Equations* (SIAM, Philadelphia, PA, 1983).
- [5] E.R. Arantes e Oliveira, I. Babuška, O.C. Zienkiewicz and J.P. Gago, eds., *Proceedings International Conference on Accuracy Estimates and Adaptive Refinements in Finite Element Computations*, Lisbon, 1984.
- [6] K. Ghia and U. Ghia, eds., *Advances in Grid Generation*, FED-Vol. 5 (ASME, New York, 1983).
- [7] R.V. Southwell, *Relaxation Methods in Theoretical Physics* (Clarendon Press, Oxford, 1946).
- [8] R.V. Southwell, *Relaxation Methods in Theoretical Physics*, Vol II (Clarendon Press, Oxford, 1956).
- [9] P.A. Gnoffo, A generalized orthogonal coordinate system for describing families of axisymmetric and two-dimensional bodies, NASA TMX-3468, NASA Langley Research Center, 1977.
- [10] P.R. Brown and R.R. Hayhurst, The use of the Schwarz–Christoffel transformation in mesh generation for the solution of two-dimensional problems, in: *Computers in Engineering*, Vol. 3 (ASME, New York, 1982) 1–8.
- [11] L.N. Trefethen, *SIAM J. Sci. Stat. Comput.* 1 (1980) 82.
- [12] O.L. Anderson, *Comput. & Fluids* 8 (1980) 391.
- [13] R.T. Davis, Numerical methods for coordinate generation based on Schwarz–Christoffel transformations, in: *Proceedings AIAA Fourth Computational Fluid Dynamics Conference*, Williamsburg, VA, 1979.
- [14] A.M. Winslow, Numerical solution of the quasilinear Poisson equation in a nonuniform triangular mesh, *J. Comput. Phys.* 2 (1967) 149–172.
- [15] J.F. Thompson, F.C. Thames and C.W. Mastin, TOMCAT—a code for numerical generation of boundary-fitted curvilinear coordinate system on fields containing any number of arbitrary two-dimensional bodies, *J. Comput. Phys.* 24 (1977) 274–302.
- [16] A.M. Winslow, *Adaptive mesh zoning by the equipotential method*, UCID-19062, University of California at Lawrence Livermore, CA, 1981.

- [17] J.U. Brackbill and J.S. Saltzman, Adaptive zoning for singular problems in two dimensions, *J. Comput. Phys.* 46 (1982) 342–368.
- [18] C.W. Mastin and J.F. Thompson, Transformation of three-dimensional regions onto rectangular regions by elliptic systems, *Numer. Math.* 29 (1978) 397.
- [19] K.N. Ghia and U. Ghia, Numerical generation of a system of curvilinear coordinates for turbine cascade flow analysis, Aerospace Engineering and Applied Mechanics Rept. 75-4-17, University of Cincinnati, OH, 1975.
- [20] W.C. Thacker, A. Gonzalez and G.E. Putland, A method for automating the construction of irregular computational grids for storm surge forecast models, *J. Comput. Phys.* 37 (1980) 371–387.
- [21] M.S. Shephard and M.A. Yerry, An approach to automatic finite element mesh generation, in: *Computers in Engineering*, Vol. 3 (ASME, New York, 1982) 21–28.
- [22] K.Y. Chung, N. Kikuchi and J.E. Taylor, Application of adaptive finite element methods to shape optimization problems, Mechanical Engineering and Applied Mechanics Rept. 85-9-1, University of Michigan, East Lansing, MI, 1985.
- [23] I. Babuška and W.C. Rheinboldt, Reliable error estimation and mesh adaptation for the finite element method, in: J.T. Oden, ed., *Computational Methods in Nonlinear Mechanics* (North-Holland, Amsterdam, 1980) 67–109.
- [24] I. Babuška and W.C. Rheinboldt, Error estimates for adaptive finite element computations, *SIAM J. Numer. Anal.* 15 (1978) 736–754.
- [25] I. Babuška and M.R. Dorr, Error estimates for the combined h and p version of the finite element method, *Numer. Math.* 25 (1981) 257–277.
- [26] I. Babuška and B. Szabó, On the rates of convergence of the finite element method, *Internat. J. Numer. Meths. Engrg.* 18 (1982) 323–341.
- [27] I. Babuška, A. Miller and M. Vogelius, Adaptive methods and error estimation for elliptic problems of structural mechanics, in: I. Babuška, J. Chandra and J.E. Flaherty, eds., *Adaptive Computational Methods for Partial Differential Equations* (SIAM, Philadelphia, PA, 1983) 57–73.
- [28] A.R. Diaz, N. Kikuchi, P. Papalambros and J.E. Taylor, Design of an optimal grid for finite element methods, *J. Structural Mech.* 11(2) (1983) 215–230.
- [29] A.R. Diaz, N. Kikuchi and J.E. Taylor, A method of grid optimization for finite element methods, *Comput. Meths. Appl. Mech. Engrg.* 37 (1983) 29–46.
- [30] A.R. Diaz and N. Kikuchi, Remodeling of finite element grids based on interpolation errors, in: E.R. Arantes e Oliveira, I. Babuška, O.C. Zienkiewicz and J.P. Gago, eds., *Proceedings International Conference on Accuracy Estimates and Adaptive Refinements in Finite Element Computation*, Lisbon, 1984.
- [31] A.R. Diaz, N. Kikuchi and J.E. Taylor, Optimal design formulations for finite element grid adaptation, in: V. Komkov, ed., *Sensitivity of Functionals with Applications to Engineering Science. Lecture Notes in Mathematics 1086* (Springer, Berlin 1984) 56–76.
- [32] E.R. Arantes e Oliveira, Optimization of finite element solution, in: *Proceedings Third Conference of Matrix Methods in Structural Mechanics*, Wright-Patterson Air Force Base, Dayton, OH, 1971.
- [33] D.J. Turcke and G.M. McNeice, Guidelines for selecting finite element grids based on an optimization study, *Comput. & Structures* 4 (1974) 499–519.
- [34] C.A. Felippa, Optimization of finite element grids by direct energy search, *Appl. Math. Modelling* 1 (1976) 93–96.
- [35] C.A. Felippa, Numerical experiments in finite element grid optimization by direct energy search, *Appl. Math. Modelling* 1 (1977) 239–244.
- [36] R.J. Melosh and P.V. Marcal, An energy basis for mesh refinement of structural continua, *Internat. J. Numer. Meths. Engrg.* 11 (1977) 1083–1091.
- [37] M.S. Shephard, R.H. Gallagher and J.F. Abel, The synthesis of near-optimum finite element meshes with interactive computer graphics, *Internat. J. Numer. Meths. Engrg.* 15 (1980) 1021–1039.
- [38] N. Kikuchi and J.H. Cheng, Adaptive remodeling of finite element grids for large-deformation elastoplasticity in metal-forming analysis, *Comput. Meths. Appl. Mech. Engrg.* (1986) to appear.
- [39] K.Y. Chung and N. Kikuchi, Adaptive methods to solve free boundary problems of flow through porous media, *Internat. J. Numer. Analyt. Meths. Geomech.* (1986) to appear.
- [40] J.H. Cheng and N. Kikuchi, A mesh rezoning technique for finite element simulations of metal forming processes, *Internat. J. Numer. Meth. Engrg.* (1986) to appear.

- [41] B.A. Szabó and A.K. Mehta, p -convergent finite element approximations in fracture mechanics, *Internat. J. Numcr. Meths. Engrg.* 12 (1978) 551–560.
- [42] O.C. Zienkiewicz, J.P. Gago and D.W. Kelly, The hierarchical concept in finite element analysis, *Comput. & Structures* 16 (1983) 53–65.
- [43] J.T. Oden, Adaptive methods for incompressible viscous flow problems with moving boundaries, in: E.R. Arantes e Oliveira, I. Babuška, O.C. Zienkiewicz and J.P. Gago, eds., *Proceedings International Conference on Accuracy Estimates and Adaptive Refinements in Finite Element Computations*, Lisbon, 1984.
- [44] L. Demkowicz, J.T. Oden and T. Strouboulis, Adaptive finite elements for flow problems with moving boundaries. Part I: Variational principles and a posteriori estimates, *Comput. Meths. Appl. Mech. Engrg.* 46 (1984) 217–251.