INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 18, 747–760 (1994)

ADAPTIVE PANEL METHODS AND THEIR INTEGRATION WITH CAD

TIM DAVID AND ROBERT LEWIS

Department of Mechanical Engineering, University of Leeds, Leeds, L52 9JT, U.K.

SUMMARY

The work outlined below presents simple but effective adaptive meshing algorithms for boundary integral methods modelling inviscid flows (panel method) using the IGES standard for describing geometry. By using certain IGES entities in describing the boundary, CAD-derived geometry may be used such that the geometric integrity of the boundary is maintained after an adaptive redistribution of the mesh. Three types of error estimators are tested and all are shown to produce a more accurate representation of the flow phenomena for the same number of panels as compared with a uniform mesh distribution.

KEY WORDS Computer-aided design Adaptive methods Panel methods Computational fluid dynamics

1. INTRODUCTION

Although panel methods rely on a simple inviscid fluid model, they still have an important role to play, particularly in the initial design phase and in providing starting solutions for more complex numerical fluid models. Additionally, the integration of computer-aided-design (CAD) generated complex geometry data with the numerical solution of fluid models is becoming realizable within the world of increasingly powerful computational hardware and emerging CAD standards such as the Initial Graphics Exchange Specification (IGES)[1] and STEP.[2] Shorter design iteration times are now possible with the full integration of design stages and analysis software.

Adaptive grid algorithms using *a posteriori* error indicators have been used extensively for finite difference and finite element methods for some time. It is only in the past few years that work has been undertaken on boundary elements. Three basic methods have been advocated, namely h-, p- and r-methods: h-methods essentially refine in areas of high error value, p-methods vary the order of the polynomial approximating the solution across each boundary element and finally r-methods keep both the polynomial order and the number of boundary elements constant but redistribute the mesh to reduce the global error.

Ingber and Mitra[3] presented a redistribution (r-method) for both harmonic and biharmonic equations, including the lid-driven cavity (for small Re) as a relevant example. Although improvements over uniform grids were small, the modelling of recirculation zones available from the biharmonic equation were significantly better. Rencis and Mullen[4] used an h-method to provide self-adaptive meshes for an elasticity problem. Solutions for the current and previous meshes were used to provide error estimators for each element, which was subsequently subdivided into a number of divisions when the error became large enough. Carey and Kennon[5] realized the importance of maintaining the geometry of the problem as well as providing a better solution in redistributing the mesh. Their method relied on a composite error indicator taking

CCC 0271–2091/94/080747–14 *Received February 1993*

into account both the solution and the geometry in the mesh redistribution. The number of elements remained constant. Rank[6] combined both h- and p-methods in solving problems where singularities existed in the solution. In this case convergence to a mesh was obtained by using the p-method refinement far from singularities and the h-method refinement close to singularities. Additionally, residuals had to be evaluated at two non-collocation points for each element to provide numerical approximations to the error estimators. Guiggiani[7] presented an h-adaptive strategy together with an algorithm for truncating the mesh refinement procedure. The error estimator was based on the fundamental property for boundary elements that two independent problem definitions can be evaluated for the same model problem. Collocations were evaluated at two differing position sets and an error estimator was found from the difference between the two solutions. Clearly the computational work could be extensive if the problem size increased. Finally Sun and Zamani[8] have presented an adaptive hr-method with error indicators modelled by residuals formed from the interpolation operator across the boundary element. They show that the method is only applicable in 2D owing to the weak singularities and cannot be easily applicable to 3D.

For the most part grid strategies have concentrated on classical stress modelling problems. In addition, r-methods have been chosen for their advantages in storage and CPU time. However, it has been shown[5] that the geometry of the problem is just as important as a solution and mesh redistribution and, if not carried out properly, can cause geometry degradation. Parallel algorithms for boundary element problem formulations[9] have also shown that both computationally intensive schemes and storage problems can be alleviated and are not such a disadvantage as first presumed. P-methods also have the disadvantage that for relatively high polynomial orders the numerical integration becomes particularly difficult.[5]

It is proposed that an integrated fluid/design tool should have the following characteristics. Firstly it should be able to use, with relative ease, the geometry emanating from generally available CAD design tools. Secondly it should be reasonably interactive so that the computational cost will not be high. Thirdly it should model the flow with the greatest possible accuracy for that computational cost.

The following work is based on the integration of a CAD-derived geometry and a simple fluid model along with an *a posteriori* mesh refinement technique with the ability to provide a mesh distribution with a *reduced*, *constant* or *increased* number of elements. As an initial model the element mesh length has a linear relationship with the estimated error rather than the more common use of binary subdivision. These error estimators are evaluated relatively cheaply by using heuristic arguments for their definitions.

For modelling fluid flow, inviscid conditions are assumed and we use the well-known panel method[10] as the underlying solution mechanism. The work also shows that through the use of IGES data standardization the burden of integration is eased between differing software tools, thus providing a firm framework for progression towards more complex applications.

Conjugate gradient (CG) algorithms have been shown to have advantages for the solution of dense linear matrix equations over other iterative methods such as Gauss–Seidel and direct methods, especially in the area of parallelization.[9] It is in this light that CG methods have been chosen to solve the dense matrix equations formed from panel methods.

The case study used in this work was for the initial design of prosthetic heart valves.[11,12] Basic 2D flow patterns and pressure contours were needed for various positions during the cardiac cycle of the leaflets inside a duct modelling the artery. At this stage quick results from the fluid model were essential, since a variety of shapes could be tried and analysed, each design emanating from CAD software, without the need for the designer to leave the workstation.

2. INITIAL GRAPHICS EXCHANGE SPECIFICATION (IGES)

The Initial Graphics Exchange Specification has as its basic definition a set of file output formats representing product data. As set out in the version 4.0 IGES specification,[1] the file format establishes structures to be used for the digital representation and communication of product definition data. Thus it permits the compatible exchange of product definition data used by various computer-aided-design and computer-aided-manufacturing (CAD/CAM) systems. Some of these product definition data specify the geometry, in particular the bounding surface of a part or assembly of parts in or around which fluid may flow. This geometry is represented as a non-ordered listed set of predefined geometric entities. Examples of these entities are arcs, lines, points and B-spline curves/surfaces. Both 2D and 3D geometries may be represented in this way. The specification of the IGES output is such that it produces an ASCII text file. It is clear from the above definition that IGES (or any other standard for geometric data storage and output such as STEP) could provide a well-defined specification for the integration of boundary element methods in general.

For the presented work only straight line and arc entities were used to describe the geometry of the design shape. It is important to note that other entities used in the CAD drawing, e.g. lines for the use of dimensioning arrows, are indistinguishable from the required geometric entities. In normal circumstances, however, this problem can easily be overcome, since the CAD user would provide dimensioning details on another layer (recognizable by the IGES file format) from the main drawing, thus enabling the IGES interface to recognize automatically the bounding surface of the design.

3. PANEL METHOD THEORY AND SOLUTION METHOD

No attempt will be made here to completely describe the panel method, since it has been fully documented elsewhere.[9,10] The adaptive algorithm is independent of the order of approximation of the singularity and hence for simplicity the test cases and examples are modelled by a non-lifting source distribution of constant strength over each panel.

Let $\mathbf{q}_{ij}$ be the velocity induced at the control point of the ith panel by a unit source density on the jth panel. This induced velocity may be evaluated by using the normal derivative of the perturbation potential and integrating over the panel surface S. For 2D the integral can be evaluated analytically; for axisymmetric and 3D geometries the integral is determined numerically using only the spatial and geometrical information on the panels i and j, which is readily available. The normal velocity induced at the ith control point by the singularity distribution at the jth panel is thus

$$A_{ij} = \mathbf{n}_i \cdot \mathbf{q}_{ij}, \tag{1}$$

where $\mathbf{n}_i$ is the unit normal vector of the ith panel and A_{ij} is termed the influence coefficient. From the boundary condition of a prescribed normal velocity on the body surface at each panel i we obtain the set of simultaneous equations

$$\sum_{j=1}^{N} A_{ij}\sigma_j = -\mathbf{V}_{i\infty} \cdot \mathbf{n}_i + F_i, \tag{2}$$

i.e. the matrix equation

$$A\sigma = b, \tag{3}$$

where $b_i = -\mathbf{V}_{i\infty} \cdot \mathbf{n}_i + F_i$, σ_j is the source density at the control point of panel j and F_i is a known transpiration velocity at the control point of panel i. Clearly, from the form of the integral defining the perturbation velocity, the matrix A is dense and non-symmetric. Once a solution has been obtained, the velocity at each control point on the solid surface is evaluated by

$$\mathbf{q}_i = \sum_{j=1}^{N} \mathbf{q}_{ij}\sigma_j + \mathbf{q}_{i\infty}. \tag{4}$$

Velocities may now be evaluated at positions within the fluid surrounding the solid surface by using similar algorithms to that used in the influence matrix evaluation. The off-body velocity is evaluated with the use of the equation

$$\mathbf{V}_i = \sum_{j=1}^{N} \mathbf{V}_{ij}\sigma_j + \mathbf{V}_{i\infty}. \tag{5}$$

Here the matrix $\mathbf{V}_{ij}$ is evaluated by the same method as for $\mathbf{q}_{ij}$, but in this case the panel centroid position i is substituted for the spatial position at which the off-body velocity is to be calculated. Since each perturbation velocity is evaluated from every other panel (including itself) so that the matrix A_{ij} is dense, the method chosen for solution in this case is a conjugate gradient algorithm owing to its efficient parallelization.[9] For purposes of experiment the test cases were two-dimensional and used constant source distributions so that each panel had a single collocation point located at its centre. We note here that for both 2D and 3D cases the diagonal entries of A, i.e. a_{ii}, are 2π and all other entries a_{ij} are such that $a_{ij} < 2\pi$. Since conjugate gradient methods are minimization algorithms, in order to increase the efficiency of the algorithm, it is useful to obtain a good starting vector for σ, i.e. as close to the minimum of the quadratic functional as possible. We assume that initially the matrix A can be approximated by the diagonal matrix $2\pi\mathbf{I}$, where $\mathbf{I}$ is the identity matrix. Thus the first approximation to the solution is given by

$$\sigma^0 = \mathbf{b}/2\pi.$$

For its robustness we choose the conjugate gradient squared algorithm[13] which is applicable to non-symmetric matrices. The algorithm is shown below.

Defining the residual $\mathbf{r}$ as $\mathbf{r} = \mathbf{b} - A\mathbf{x}$, with $\mathbf{x}_i$ as the ith iterate, then

$$\mathbf{r}(\mathbf{x}_i) = \mathbf{r}_i = \mathbf{b} - A\mathbf{x}_i. \tag{6}$$

We choose an initial value $\mathbf{x}_0$ for $\mathbf{x}$, and set $\mathbf{r}_0 = \mathbf{k}_0 = \mathbf{p}_0$. Thus for $\mathbf{i} = 0, 1, \ldots, N$

$$\alpha_i = \langle \mathbf{r}_0^{\mathrm{T}}, \mathbf{k}_i \rangle / \langle \mathbf{r}_0 A\mathbf{p}_0 \rangle, \tag{7}$$

$$\mathbf{h}_i = \mathbf{k}_i - \alpha_i A\mathbf{p}_i, \tag{8}$$

$$\mathbf{r}_{i+1} = \mathbf{r}_i - \alpha_i A(\mathbf{h}_i + \mathbf{k}_i), \tag{9}$$

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha_i(\mathbf{h}_i + \mathbf{k}_i), \tag{10}$$

$$\beta_k = \langle \mathbf{r}_0^{\mathrm{T}}, \mathbf{r}_{i+1} \rangle / \langle \mathbf{r}_0^{\mathrm{T}}, \mathbf{r}_i \rangle, \tag{11}$$

$$\mathbf{p}_{i+1} = \mathbf{r}_{1+1} + 2\beta_i\mathbf{h}_i + \beta_i^2\mathbf{p}_i, \tag{12}$$

$$\mathbf{k}_{i+1} = \mathbf{r}_{i+1} + \beta_i\mathbf{h}_i. \tag{13}$$

The iteration is stopped when $|\mathbf{r}_{1+1}| < \varepsilon$, with ε specified *a priori*.

4. INITIAL DISCRETIZATION

Once an initial design has been defined on a CAD system, the geometry and other data are output in the IGES format. The discretization algorithm searches the IGES file for each entity describing the geometry. These entities are initially discretized uniformly into panel segments. The number of panels per unit arc length (or unit area for 3D) of each entity is specified *a priori* such that they describe the geometry sufficiently.

The class of problems chosen for this work requires

(i) that all bodies be closed
(ii) normal vectors for each panel point into the fluid.

It is assumed that the first condition is met by the CAD user. The second requirement is met *automatically* by the software and is transparent from the user.

The reordering of the panels is continuous around each body and specified to be anticlockwise. A user input of nominal panel length for straight line entities and nominal angle for arc entities is all that is required.

5. ADAPTIVE PANEL DISTRIBUTION ALGORITHM

It is a common occurrence in numerical simulations of fluid problems that the uniform discretization of either the boundary surface or the entire fluid domain may not be an optimal one. This is certainly the case for panel methods with arbitrary geometry. The algorithm provides that the zero (or prescribed) normal flow boundary condition is imposed only at the collocation point on each panel. This discrete condition allows the panel to 'leak' at positions other than the panel control points and therefore will not necessarily provide a good representation of a streamline along the body surface other than at the collocation position. Normally an adaptive discretization will be based on some local error as has been reviewed in Section 1. An objective of any adaptive mesh or boundary discretization will be to reduce that error over the full domain. The authors have taken the same approach with the present work; however, rather than try to find an actual error value, the philosophy has been to use the knowledge obtained of panel methods and the associated solution to obtain parameters which are in some sense proportional to the local error but computationally cheap. In addition, there is the important constraint that the adaptive redistribution of the panels should maintain the geometric integrity of the body derived from the CAD system. In all cases the guiding principle has been to adapt such that the flow is better represented close to the body surface (i.e, the body's 'leakage' has been reduced). In the work presented here three different formulations emerged.

1. Large fluid accelerations associated with sharp convex corners provide difficulties in modelling the fluid and cannot normally be guaranteed to provide a solution accurately modelling the flow phenomena.[10] An error parameter is formed by using the magnitude of the gradient of tangential velocity, ∇v_T, at the body surface. The error $\varepsilon_T(m)$ associated with the mth panel centroid is defined as

$$\varepsilon_T(m) = \nabla v_T(m). \tag{14}$$

Here ∇v_T is the magnitude of the tangential acceleration evaluated as a first-order approximation. Thus

$$\nabla v_T(x_m, y_m) = \left(\frac{(v_{(m+1)x} - v_{mx})^2 + (v_{(m+1)y} - v_{my})^2}{(x_{m+1} - x_m)^2 + (y_{m+1} - y_m)^2} \right)^{1/2}, \tag{15}$$

where v_{mx} and v_{my} are the x- and y-components of the tangential velocity at the collocation point of the mth panel respectively and (x_m, y_m) are the co-ordinates of the mth panel. The first-order approximation used here is for simplicity and can be of a higher order if necessary. It should be noted that the index $m + 1$ denotes the 'nearest' panel geometrically rather than ordinally.

2. Following on from formulation 1, it is presumed that a function representing the variation in source strength with respect to the boundary arc length should be as smooth as possible. Hunt[14] has already alluded to this but only provides an algorithm for special aerodynamic cases. The error algorithm presented here has been constructed completely independently from Hunt. Under this hypothesis an error parameter $\varepsilon_s(m)$ associated with the mth panel centroid is defined to be directly proportional to the magnitude of the gradient of source strength with respect to the arc length s measured along the boundary:

$$\varepsilon_s = \frac{\partial \sigma}{\partial s} = \left| \frac{\sigma_{m+1} - \sigma_m}{[(x_{m+1} - x_m)^2 + (y_{m+1} - y_m)^2]^{1/2}} \right|. \tag{16}$$

3. It is assumed that a 'large' value of average normal velocity across the entire panel, L_i given by

$$L_i = \frac{1}{\delta s_i} \int_0^{\delta s} |\mathbf{V}_i \cdot \mathbf{n}_i| \, ds - |F_i|, \tag{17}$$

where $\mathbf{V}_i$ is the sum of the perturbation velocity due to all other panels and the onset velocity, will correspond to a 'large' error in some neighbourhood of the panel. Initial tests with single closed bodies have shown that increasing the number of panels will always reduce the total leakage defined as

$$L_{\text{total}} = \sum_{i=1}^{N} \left(\int_0^{\delta s} |\mathbf{V}_i \cdot \mathbf{n}_i| \, ds - |F_i \delta s_i| \right). \tag{18}$$

This is based on the principle that in the limit as N, the number of panels, approaches infinity, the solution approaches a continuous distribution of singularities, which is an 'ideal' condition. An error parameter $\varepsilon_L(m)$ is therefore defined by evaluation of the total leakage along the panel m. In order to provide a panel-independent error parameter, the average normal velocity across the panel is defined per unit panel length and given by

$$\varepsilon_L(m) = \frac{1}{\delta s_m^2} \int_0^{\delta s_m} |\mathbf{V}_m \cdot \mathbf{n}_m| \, ds - \left| \frac{F_i}{\delta s_i} \right| = \frac{L_m}{\delta s_m}. \tag{19}$$

In order to reduce the computational work at this point, the normal velocity $\mathbf{V}_m \cdot \mathbf{n}_m$ is evaluated at only two points, each a symmetrical distance from the collocation point to the panel ends. The sum of these velocities is then equated to the normal velocity L_i given in equation (17)

From the error evaluation described above (either ε_T, ε_s or ε_L) an ordered set $[\varepsilon_i]$ of N error values is obtained, with each error value associated with the ith panel collocation point. In order to redistribute panels, there must be a relationship between error and panel length. A linear function is chosen to initially model the variation in panel length as a function of error values, the argument for this being that it is the most basic. The linear function is derived using the

maximum and minimum values of ε_i and the user-specified maximum and minimum values of panel length, P_{max} and P_{min}, so that a new panel length $P(\varepsilon)$ is found from

$$P(\varepsilon) = \frac{P_{max} - P_{min}}{\varepsilon_{min} - \varepsilon_{max}} \varepsilon + c. \tag{20}$$

In order for the integrity of the CAD-generated geometry data to be maintained, the IGES file is now reread and for each entity the following discretization algorithm is performed.

Given some co-ordinate (x_i, y_i) lying on the boundary, we associate an error value $[\varepsilon_i]$, which implies a panel arc length p_i at this point. The algorithm provides a predictor point also lying on the boundary,

$$(x_i + p_i \cos\theta,\ y_i + p_i \sin\theta) = (x_p, y_p),$$

where θ is the angle that the entity curve makes with the x-axis at the point (x_i, y_i). Using the point (x_p, y_p), a second panel arc length p_p is found by a similar association with an error value $[\varepsilon_p]$. The co-ordinates

$$(x_i, y_i) \quad \text{and} \quad \left(x_i + \frac{p_p + p_i}{2} \cos\theta,\ y_i + \frac{p_p + p_i}{2} \sin\theta\right)$$

are used as the start and end points of a single panel lying along the entity curve. The panel redistribution, with a prudent choice of maximum and minimum panel lengths, can give rise to either a *reduced, constant* or *increased* number of panels used to define the body shape. However, it will normally be the case that an increase in the number of panels will occur, resulting in a better representation of the flow.

6. RESULTS

Three test cases are presented (two of which have analytical solutions) which represent a cross-section of general 2D problems. In all the figures shown the adaptive mesh distributions have the same number of panels as the uniform distribution, thereby enabling direct comparisons between boundary discretizations.

We first consider the problem of modelling the flow past an infinitely long rectangular body whose width of two units is immersed in a unit uniform flow. Large changes in tangential velocity are found at the upstream and downstream corners of the body.[10] The geometry was modelled using four straight line entities making up a rectangular body of aspect ratio 20. A uniform mesh of 84 panels was generated to provide the initial solution. The large number of constant length panels shows the sensitivity to the corner point singularity in obtaining even a crude solution for the fluid flow. Figures 1(a)–1(d) show the meshes for the area surrounding the upstream stagnation point for all four discretizations. The asymmetry in some of these meshes is caused by the restriction of the final panel length needed to fit the associated IGES entity. Figures 2(a)–2(c) show the tangential surface speed V_T as a function of the arc length measured clockwise from the upstream stagnation point for each of the error estimators compared with the speed obtained from both a uniform mesh solution and the exact solution. In all three cases the adaptive mesh provides a better representation of the flow, with a significant increase in mesh density close to the singular point at the edge of the rectangular body. The mesh obtained from the error estimator ε_s seems to have obtained the 'best' approximation given the restricted number of panels. Note should be taken here that the uniform distribution, even using a considerable number of panels, does not model the velocity increase around the corner at all.

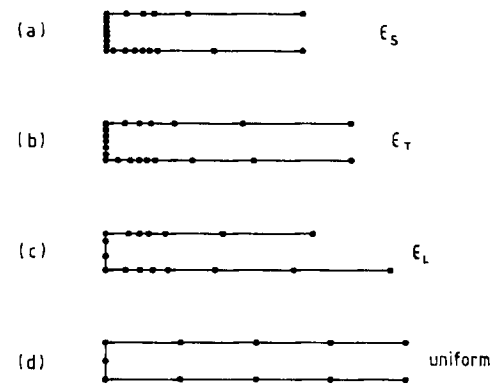
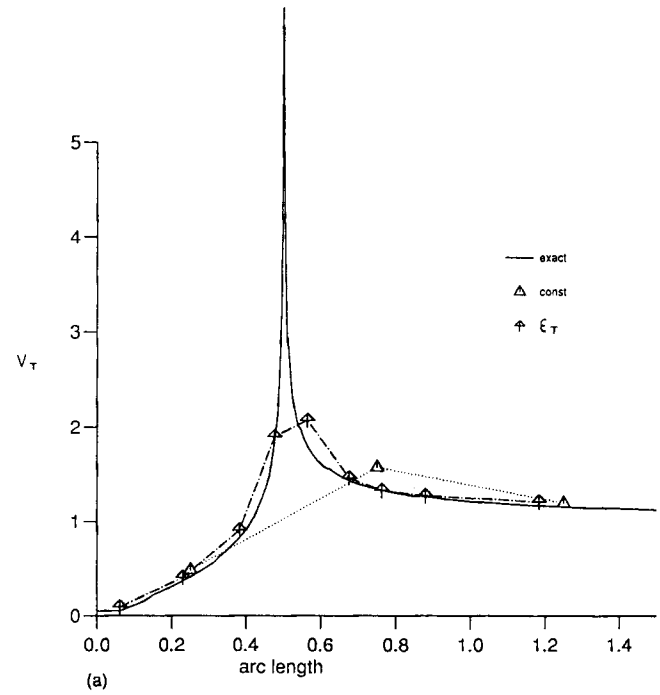
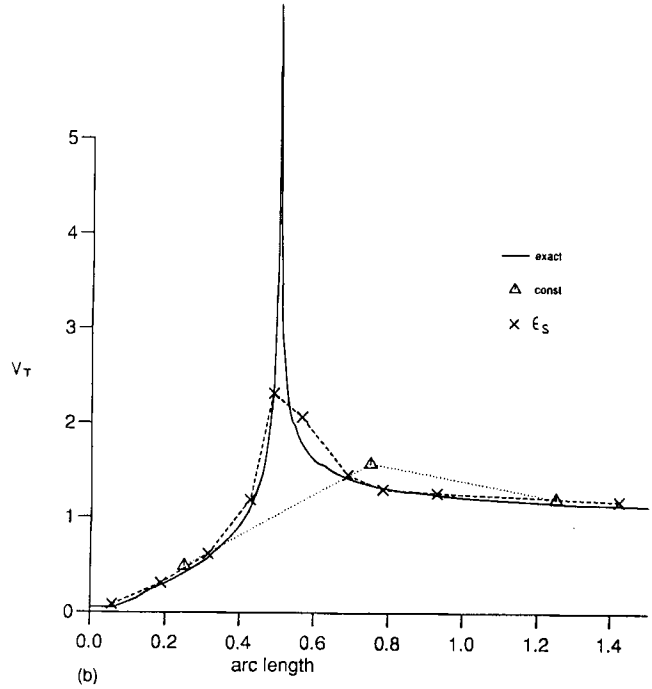


Figure 1. Mesh distributions for an infinite rectangular body: (a) ε_s, (b) ε_T, (c) ε_L, (d) uniform

The second test case, again taken from Reference 10, shows how the adaptive meshing algorithm copes with discontinuous boundary conditions. Here the flow modelled is that of uniform distributed suction over the left-hand side of a unit-radius cylinder. The cylinder geometry consists of a single arc entity and is made up from 30 panels. Figures 3(a)–3(c) show the analytical solution of the tangential surface speed V_T compared with both uniform and adaptive meshes concentrated around the boundary condition singularity where the velocity varies discontinuously for the three adaptive mesh distributions. All three error estimators



2. Tangential surface speed V_T versus boundary arc length (rectangular body): (a) ε_T, (b) ε_s, (c) ε_L



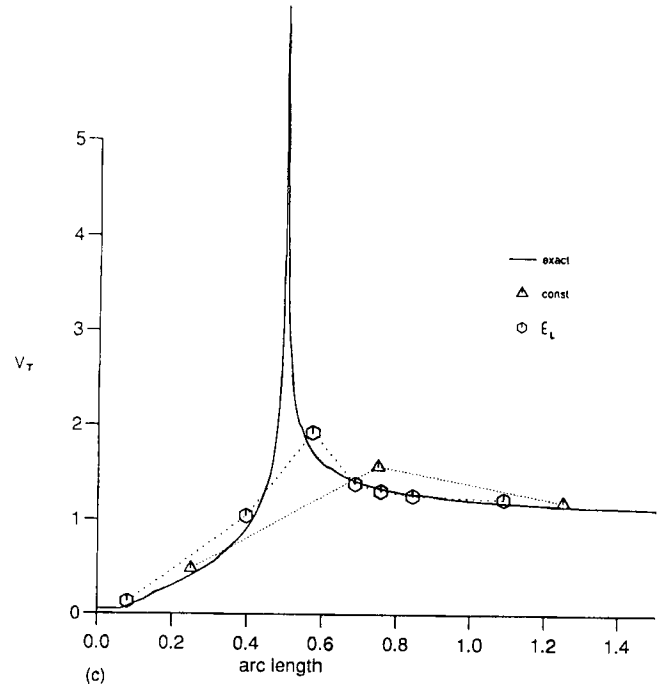


Figure 2 (*Continued*)

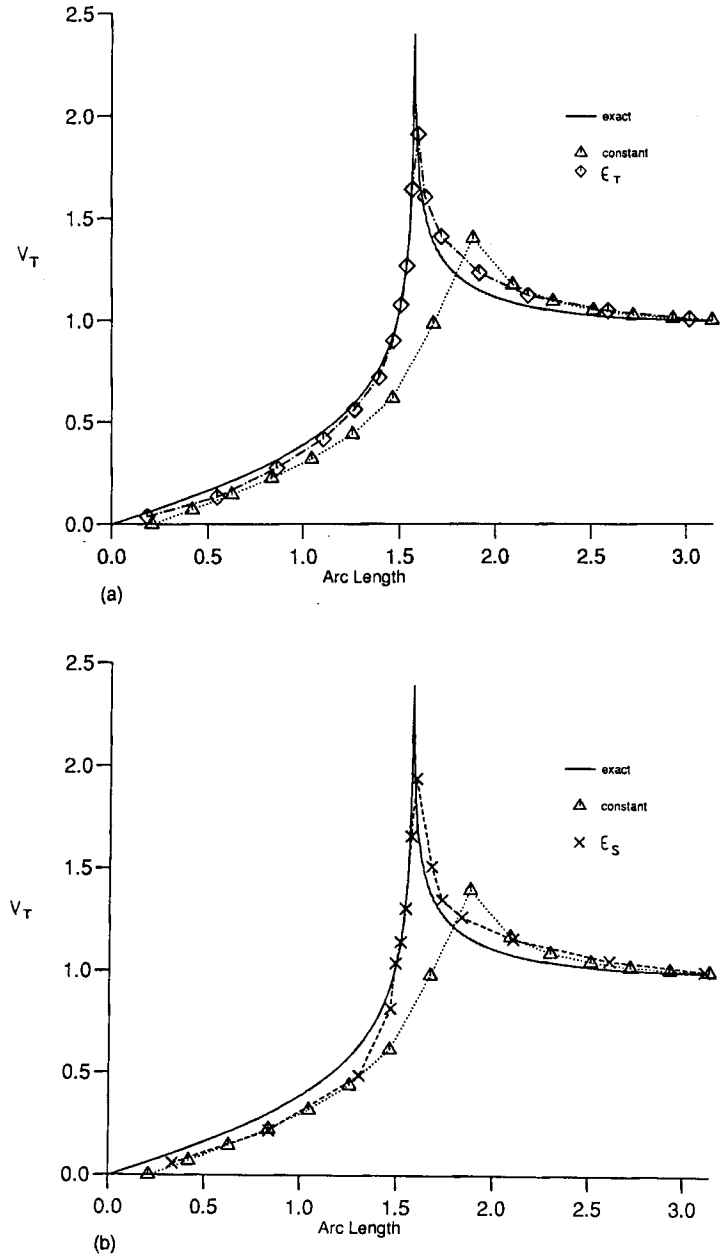


Figure 3. Tangential surface speed V_T versus boundary arc length (suction cylinder): (a) ε_T, (b) ε_s, (c) ε_L

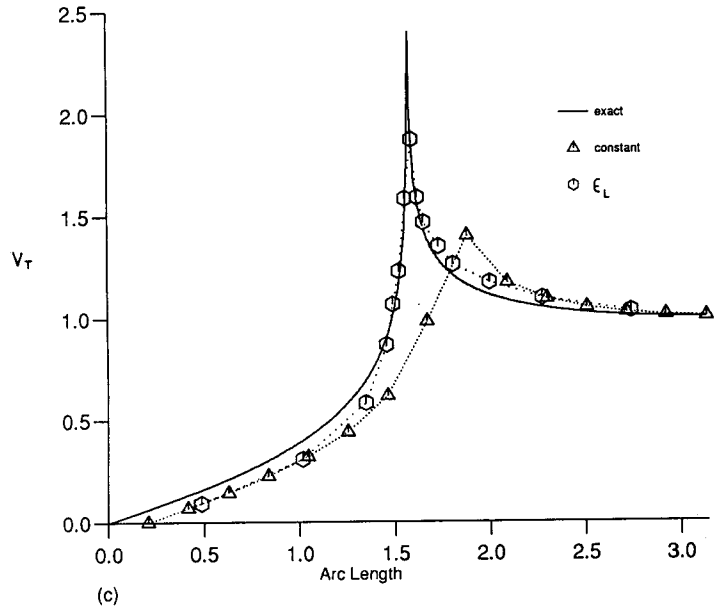


Figure 3 (*Continued*)

Figure 4. Mesh distributions for a pair of heart valve leaflets: (a) uniform, (b) ε_T

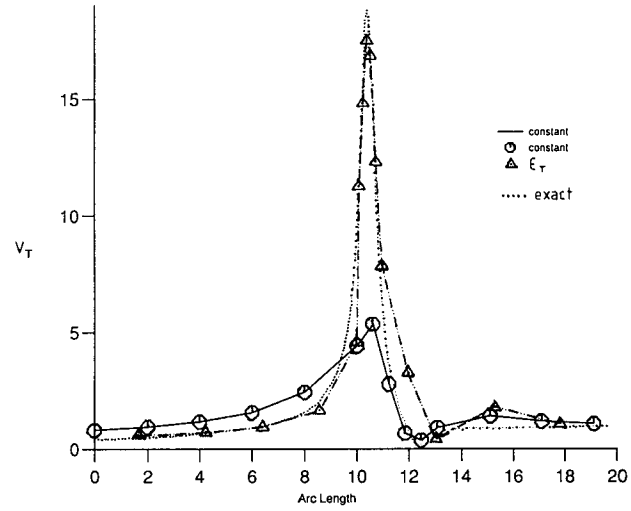


Figure 5. Tangential surface speed V_T versus boundary arc length using ε_T.

provide an excellent approximation to the solution, modelling the peak with surprising accuracy compared with the uniform mesh.

The third test case models the first-stage design of a pair of prosthetic mechanical heart valve leaflets. Here the relative positioning of the pivot points of the leaflets is an important design parameter; in this case they lie close together, with the fluid accelerating to extreme values when passing through the gap. Figure 4 shows the mesh distribution (30 panels) for both the uniform case and the adaptive algorithm for ε_T. Figure 5 shows the tangential surface speed V_T as a function of the arc length measured along the leaflet boundary, comparing the error estimator ε_T with the uniform mesh and an 'exact' solution. The 'exact' solution was evaluated using 800 panels. It can be seen that the high-velocity peak is well approximated by the adaptive mesh and is a significant improvement over the uniform mesh for the same number of panels. There was little difference in the results from the estimators of ε_s and ε_L compared with that of ε_T and those results are therefore not shown here.

7. EXTENSIONS TO 3D

Extensions to 3D provide no severe difficulties, since all three error parameters may be similarly defined in three dimensions.

1. First we consider the tangential velocity gradient. In 3D we may define the error parameter $\varepsilon(m)$ as

$$\varepsilon_T(m) = \max_k [\nabla \mathbf{v}_T(k, m)]. \tag{21}$$

where k ranges over the n neighbours of a general n-sided polygonal panel (modelling the body surface) with $\nabla \mathbf{v}_T(k, m)$ defined as

$$\nabla \mathbf{v}_T(k, m) = \left(\frac{(v_{kx} - v_{mx})^2 + (v_{ky} - v_{my})^2 + (v_{kz} - v_{mz})^2}{(x_k - x_m)^2 + (y_k - y_m)^2 + (z_k - z_m)^2} \right)^{1/2}. \tag{22}$$

In this definition the error is associated with the maximum gradient, since averaging may cancel out gradients of opposite sign.

2. Similarly the source function error may be defined as

$$\varepsilon_s(m) = \max_k \nabla\sigma \tag{23}$$

and again k ranges over the n neighbours of the mth polygonal panel, with

$$\nabla\sigma = \frac{\sigma_{m+1} - \sigma_m}{[(x_{m+1} - x_m)^2 + (y_{m+1} - y_m)^2]^{1/2}} \tag{24}$$

From a computational cost viewpoint it is best if the panels are planar. This can easily be achieved by the triangulation of each panel and is a relatively simple task.[15]

3. Finally the error estimated from the normal velocity can be defined as follows. We assume initially that the 3D panel is either a quadrilateral or a triangle. The panel may be divided into four or three equal-area subpanels respectively and their corresponding centroids

found. The normal velocity as given in equation (17) is evaluated as these centroids and summed to find the total. Thus the error estimator is given as

$$\varepsilon_L = \frac{1}{a_i^2} \sum_{j=1}^{k} [(\mathbf{V}_j \cdot \mathbf{n}_j) a_j - F_i a_i] \tag{25}$$

for all panels i and $k = 3, 4$.

8. CONCLUSIONS

It has been shown that the format of the IGES permits the compatible exchange of product definition data used by various computer-aided design and computer-aided manufacturing (CAD/CAM) systems with fluid analysis tools. By using these definition data, engineers may interact with their designs utilizing both CAD and CFD tools without leaving the workstation, thereby reducing design cycle times.

An adaptive meshing algorithm has been presented which allows an initial simple fluid solution to be obtained from the CAD geometry data. The data are structured such that any redistribution of the mesh maintains the integrity of the geometry. Three error estimators have been defined using heuristic arguments from the flow solutions. They are based upon evaluations of

(i) gradients of tangential surface velocity
(ii) gradients of source strength
(iii) average normal flow across the panel.

Using the above error estimates, an adaptive boundary discretization may be defined providing a better approximation to the fluid problem. Results have been presented for 2D flows and compared with analytical solutions. It is shown that even with the same number of panels approximating the solid body a more accurate representation of the flow is obtained. The algorithm permits a reduced, constant or increased number of panels approximating the boundary.

Extensions to 3D of all three error estimators ε_T, ε_s and ε_L have been defined and briefly discussed.

REFERENCES

1. B. Smith, G. Rindaudot *et al.*, *Initial Graphics Exchange Specification (IGES), version 4.0, NSBIR 88-3813*, NIST, Gaitherburg, MD, 1988.
2. *STEP. Industrial Automation Systems—Product Data Representation and Exchange—Part I 'Overview of Fundamental Principles', ISO 10303-1*, Committee Draft, 19XX.
3. M. S. Ingber and A. K. Mitra, 'Grid optimization for the boundary element mthod', *Int. j. numer. methods eng.*, **23**, 2121–2136 (1986).
4. J. J. Rencis and R. L. Mullen, 'Solution of elasticity problems by a self-adaptive mesh refinement technique for boundary element computation', *Int. j. numer. methods eng.*, **23**, 1509–1527 (1986).
5. G. F. Carey and S. Kennon, 'Adaptive mesh redistribution for a boundary element (panel) method', *Int. j. numer. methods eng.*, **24**, 2315–2325 (1987).
6. E. Rank, 'Adaptive *h*-, *p*- and *hp* versions for boundary integral element methods', *Int. j numer. methods eng.*, **28**, 1335–1349, (1989).
7. M. Guiggiani, 'Error indicators for adaptive mesh refinement in the boundary element method—a new approach', *Int. j. numer. methods eng.*, **29**, 1247–1269 (1990).
8. W. Sun and N. G. Zamani, 'An adaptive h–r boundary element method algorithm for $\nabla^2\psi = 0$', *Int. j. numer. methods eng.*, **33**, 537–552 (1992).
9. T. David and G. Blyth, 'Parallel algorithms for panel methods', *Int. j. numer. methods fluids*, **14**, 95–108 (1992).

10. Hess and A. M. O. Smith, 'Calculation of potential flow about arbitrary bodies', *Prog. Aeronaut. Sci.*, **8**, 1–138 (1967).
11. R. Lewis, 'Integrating CAD software with irrotational fluid flow models (a CAE application)', *Final Year Report*, Department of Mechanical Engineering, Leeds University, 1991.
12. M. King 'A novel design of a bi-leaflet mechanical heart valve', *Final Year Report*, Department of Mechanical Engineering, Leeds University, 1991.
13. D. Howard, W. M. Connolley and J. S. Rollett, 'Unsymmetric conjugate gradient methods and sparse direct methods in finite element flow simulation', *Int. j. numer. methods fluids*, **10**, 925–945 (1990).
14. B. Hunt, 'The panel method for subsonic aerodynamic flows', in W. Kollmann (ed.), *Computational Fluid Dynamics*, McGraw-Hill, New York, 1980.
15. T. David, P. H. G. Gaskell and A. Saia, 'Integrating sculptured surface design with the panel method for flow visualisation', in D. C. Handscomb (ed.), *Mathematics of Surfaces III*, Clarendon, Oxford, 1989, pp. 301–315.