

THE USE OF THE SCHWARZ-CHRISTOFFEL TRANSFORMATION
IN FINITE ELEMENT MESH GENERATION

A thesis submitted for the degree of
Doctor of Philosophy
at the University of Leicester

by

Philip Raymond Brown B.A. (Oxford)
Department of Engineering
University of Leicester

February 1990

UMI Number: U496990

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.

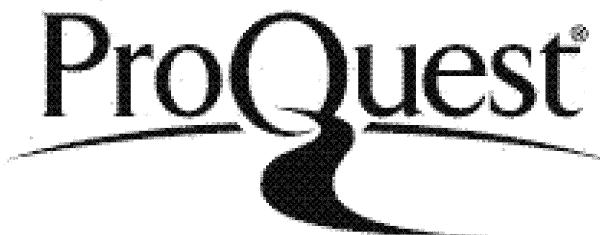


UMI U496990

Published by ProQuest LLC 2015. Copyright in the Dissertation held by the Author.

Microform Edition © ProQuest LLC.

All rights reserved. This work is protected against unauthorized copying under Title 17, United States Code.



ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346



X75169431
75000975X

1346169431 X

THE USE OF THE SCHWARZ-CHRISTOFFEL TRANSFORMATION
IN FINITE ELEMENT MESH GENERATION

A thesis submitted for the degree of
Doctor of Philosophy
at the University of Leicester

by

Philip Raymond Brown B.A. (Oxford)
Department of Engineering
University of Leicester

ABSTRACT

This thesis describes a new computer-based method for the generation of finite element meshes. It relies upon the Schwarz-Christoffel transformation, a conformal mapping from complex variable theory. This mapping is defined and some examples of its use in classical fluid dynamics are given. A practical method for evaluating the parameters defining this transformation is described and emphasis is placed on the efficiency of the solution process in order that computer run times may be kept as short as possible. A theorem in Euclidean geometry is stated and proved which links the theory of the Schwarz-Christoffel mapping and the geometrical use to which it is put here. Two such Schwarz-Christoffel transformations are used to construct a mapping between any two polygons. The desirable properties of a finite element mesh are stated and a method is described which attempts to generate such a mesh in any simply-connected two-dimensional region. Numbering of the nodes is an inherent part of the generation scheme, thus ensuring that the optimum bandwidth of the resulting system of linear equations in the analysis phase is obtained. In order to be able to present example meshes, a particular element type, the three-noded triangle, is used and a section describing the enumeration of hexagons, all of whose internal angles are $2\pi/3$, is included. The thesis includes a brief survey of existing methods of two-dimensional mesh generation as well as several example meshes.

February 1990

ACKNOWLEDGEMENTS

I wish to thank the two people who have acted as research supervisors for their helpful advice in the preparation of this thesis: Professor David Hayhurst and Professor Alan Ponter.

Special thanks go to Monica Dowson, Mark Brock, Bill Cutteridge and Keith Carter for their constant encouragement over the years.

I wish to thank Lloyd Trefethen, Thomas Higgins and Mark Shephard for their welcome correspondence.

PRB

CONTENTS

	Page
CHAPTER 1. Introduction. Existing methods in mesh generation.	
1.1 Introduction	1
1.2 Isoparametric mapping method	2
1.3 Transfinite mapping method	7
1.4 Triangulation	13
1.5 Laplacian grid generation	19
1.6 Modified quadtree method	22
CHAPTER 2. Complex variable theory and the Schwarz-Christoffel transformation.	
2.1 Background	26
2.2 Complex variable theory	30
2.3 Setting up the problem of finding the defining parameters of the Schwarz-Christoffel transformation	37
2.4 Geometrical theorem	49
2.5 Example showing ω equivalent to θ	52
CHAPTER 3. Numerical methods for functions of a complex variable.	
3.1 Equation to be solved	59
3.2 Evaluation of the integrals	59
3.3 Numerical evaluation of the Schwarz-Christoffel transformation	61
3.4 Calculation of a_2 and b_2	65
3.5 Example of complete Schwarz-Christoffel transformation	66
3.6 Inversion of the Schwarz-Christoffel transformation	69
3.7 Example of inversion of the conformal mapping	72
3.8 Conformal mapping between two polygons	75
CHAPTER 4. Finite element mesh generation.	
4.1 Desirable properties of a finite element mesh	79
4.2 Aims of this method	81
4.3 Mesh generation in a simply-connected two-dimensional region	82

	Page
4.4 Choice of r_i for mapping f	84
4.5 Alterations to the generated mesh	92
CHAPTER 5. Examples. Choice of a class of Q.	
5.1 Choice of polygon Q	94
5.2 Enumerating $2\pi/3$ hexagons	96
5.3 Example of the variation of function ϕ with X	104
5.4 Numbering the nodes of a $2\pi/3$ hexagon	104
5.5 Description of a computer program	108
5.6 Example meshes	112
CHAPTER 6. Conclusions and further work.	
6.1 Conclusions	125
6.2 Further work	126
APPENDIX 1. Paper "A non-interactive method for the automatic generation of finite element meshes using the Schwarz-Christoffel transformation" by P.R. Brown, from Computer Methods in Applied Mechanics and Engineering, volume 25, pp 101-126 (1981).	128
APPENDIX 2. Paper "The use of the Schwarz-Christoffel transformation in mesh generation for the solution of two-dimensional problems" by P.R. Brown and D.R. Hayhurst, from "Computers in Mechanical Engineering", ASME, New York, Volume 1, No. 1, August 1982. Originally appeared in L.E. Hulbert (Ed.), "Computers in Engineering", Proceedings of International Conference, San Diego, California, 15-19 August, 1982, pp 1-8, ASME, New York (1982).	155
APPENDIX 3. Derivation of the expressions for a_2 and b_2	163
REFERENCES	166
APPENDIX 4. Microfiche inside back cover. Computer program listing.	

CHAPTER 1

Introduction. Existing methods in mesh generation

1.1 Introduction

The method of finite element analysis has become a well known one in various fields of engineering since the use of computers has become widespread. It has meant that complicated artifacts can be represented by an assemblage of elements of a finite size and their behaviour modelled theoretically without recourse to expensive laboratory testing. However, the creation of complicated finite element meshes can in itself be very time consuming. Drawing by hand is one way of defining a mesh and this is clearly the most flexible method, albeit a time consuming one. The use of a digitising table to quantify the information in a drawing can, however, lessen the burden. Defining a mesh manually is prone to errors and has created a need for simple computer algorithms which check that the details of the mesh have been input correctly: even when a mesh is displayed on a graphical device, there may well be errors which cannot be seen to the observer but which would invalidate any subsequent analysis. As the computational efficiency of finite element programs has increased, the process of accurate data preparation is still a burden for most analysts. There is a need for improved input procedures to parallel the advances made in computational methods.

For the past twenty years, investigators have been developing finite element mesh generating techniques and designing pre-analysis (pre-processing) systems to reduce the effort required to create finite element models. An early attempt at mesh generation was in the LUPIM system developed by the computer aided design group at Leicester

University and described by Nazlawy [reference 1]. This was an interactive computer program and relied on the user to judge by eye whether the mesh generated would be a good model for subsequent analysis. As well as relieving the engineer of the burden of data preparation, such a software tool should also ensure that the generated data are free of errors. Since the time of LUPIM, different methods of mesh generation have been developed by researchers, automatic with the minimum of user input or interactive with the user's judgement necessary to decide when a satisfactory mesh has been obtained, but with a subsequent increase in user input and time. The remainder of this chapter is devoted to a review of methods of mesh generation currently available.

1.2 Isoparametric mapping method

Zienkiewicz and Phillips [2] show in their paper how to construct curvilinear co-ordinate systems on regions whose boundaries are defined as low order parametric polynomials. Their scheme uses isoparametric mappings to generate generalised co-ordinate systems in four-sided regions and presumes that the area in which the mesh is to be generated has already been sub-divided into four-sided patches. As an example, let the case be considered where each of the four sides is represented by a quadratic curve so that the whole quadrilateral is defined by co-ordinates of the four corners and four mid-sided points (figure 1-1). Let a curvilinear co-ordinate system (u,v) be defined inside and on the boundary of the patch so that $(u,v) \in [0,1] \times [0,1]$. A general point (x,y) is given by

$$x = \sum_{i=1}^8 N_i x_i, \quad y = \sum_{i=1}^8 N_i y_i \quad (1-1)$$

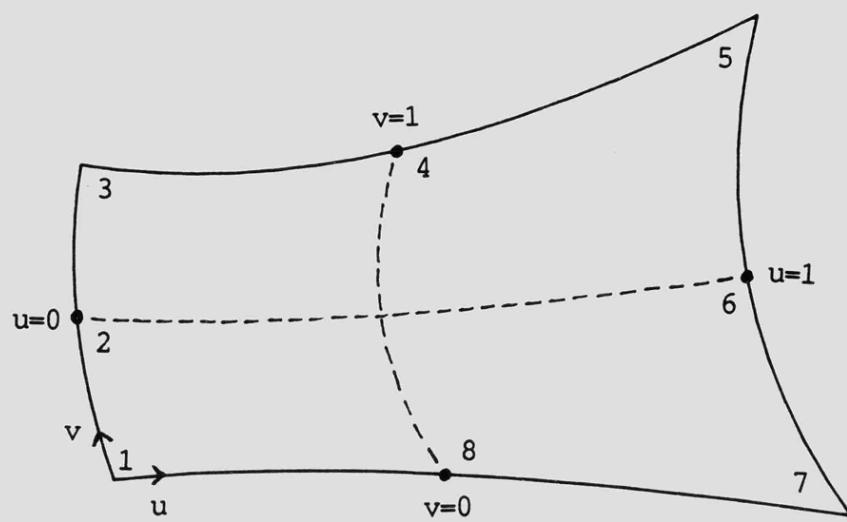


Figure 1-1. Isoparametric co-ordinates in a four-sided region with four mid-side nodes.

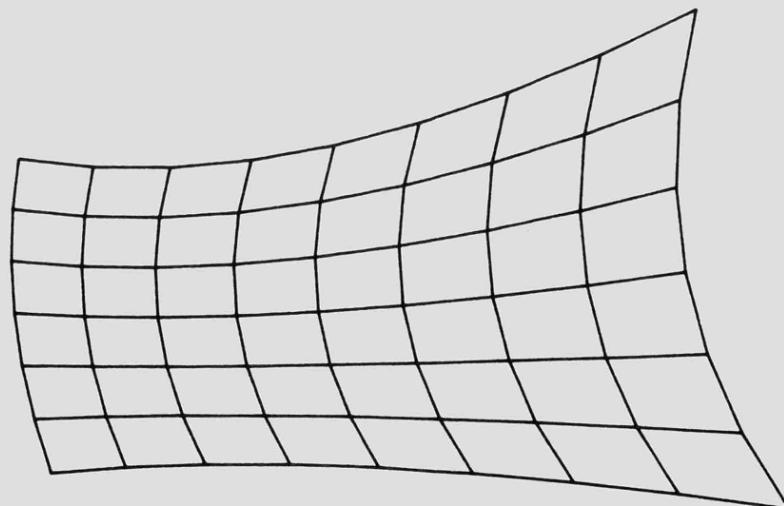


Figure 1-2. Typical mesh in four-sided patch. Nodal points given by equation (1-1).

where N_i is a shape function associated with each node and defined in terms of (u,v) . The numbered points in the figure have co-ordinates (x_i, y_i) . For this particular case,

$$\begin{aligned}
 N_1 &= -(2u + 2v - 1) (u - 1) (v - 1), \\
 N_2 &= 4 (u - 1) (v - 1) v, \\
 N_3 &= (2u - 2v + 1) (u - 1) v, \\
 N_4 &= -4 (u - 1) u v, \\
 N_5 &= (2u + 2v - 3) u v, \\
 N_6 &= -4 (v - 1) u v, \\
 N_7 &= -(2u - 2v - 1) (v - 1) u, \\
 N_8 &= 4 (u - 1) (v - 1) u.
 \end{aligned} \tag{1-2}$$

For any given (u,v) , N_i may be found by (1-2) and (x,y) by (1-1). Figure 1-2 shows a grid of lines of constant u and v . In order to generate a grid like this, all that need be specified are the co-ordinates of the 8 selected points (4 corner and 4 mid-side) and the number of sub-divisions of each side. Grading of inner elements may be obtained by appropriate selection of the mid-side points (figure 1-3) although care must be taken in the selection of points 2, 4, 6 and 8 that the map from $[0,1] \times [0,1]$ remains one-one. An example where the mapping is not one-one is given in figure 1-4. The resultant mesh could easily be made to consist of triangular elements by joining opposite quadrilateral corners, typically by selecting the shorter diagonal to achieve a mesh of elements as well proportioned as possible or by applying the "max-min" angle criterion, described in section 1.4. The quadrilateral shape can be distorted, even to the point where two adjacent sides degenerate into one line. An example of the sort of mesh which can be obtained by this method is given in figure 1-6.

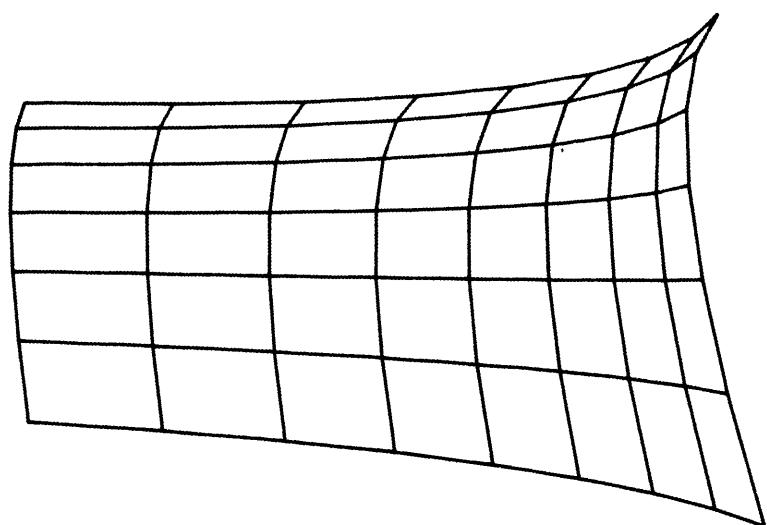


Figure 1-3. Graded isoparametric mesh.

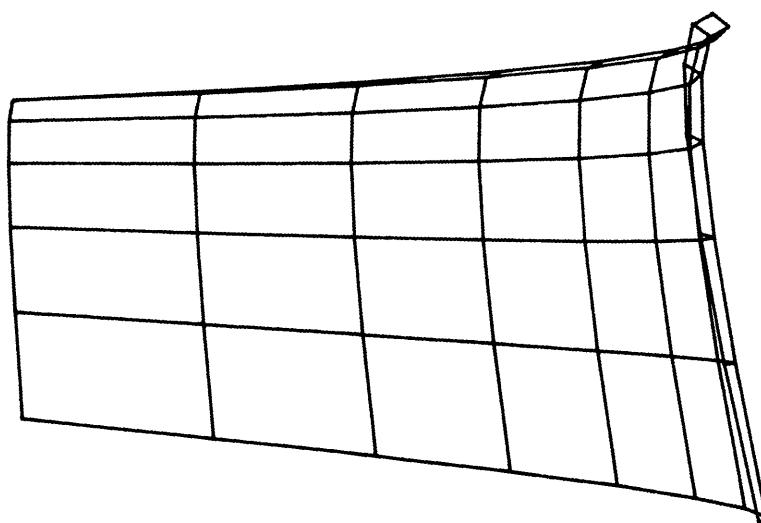


Figure 1-4. Overspill in isoparametric patch where map is not one-one.

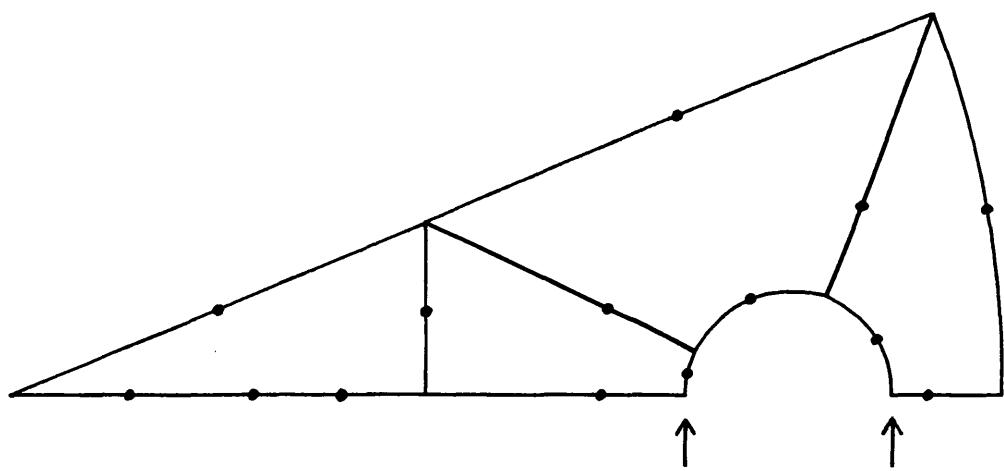


Figure 1-5. Sub-division of region into four-sided patches.

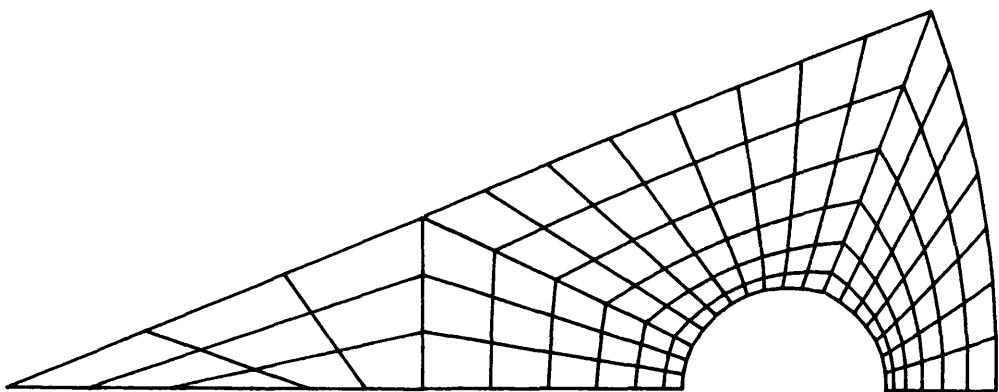


Figure 1-6. Mesh generated in region sub-divided in figure 1-5.

Figure 1-5 shows the initial sub-division into patches. This method may easily be extended to three dimensions by adding a third component to (u,v) and (x,y) in equations (1-1) and (1-2).

1.3 Transfinite mapping methods

Continuous form. A disadvantage of the method of Zienkiewicz and Phillips is that the boundary of the generated mesh is guaranteed to coincide with the boundary of the target quadrilateral patch only at the specified nodal points, in the example in equations (1-1) and (1-2) at the points (x_i, y_i) . The boundaries are represented by parametric quadratics which in general will not coincide with the given boundary of the patch. For example, the arrowed boundary in figure 1-5 is represented by three parametric quadratics whereas the target boundary was a semi-circle. This method is satisfactory only if the boundaries of the patch can be reasonably represented by a parametric polynomial. In their paper [3], Gordon and Hall approach the same problem of generating a grid inside a quadrilateral patch from a slightly different standpoint: their method, using a continuous transfinite mapping, ensures that the boundaries are mapped exactly. Moreover, each boundary does not need to be smooth: it is allowed to be piecewise smooth. Their method, which, like that of Zienkiewicz and Phillips, assumes that the domain has already been sub-divided into four-sided patches, is as follows. Let F be the patch where the mesh is to be generated. A vector-valued mapping W will be defined which maps $[0,1] \times [0,1]$ onto F . The mapping is constrained by ensuring that the grid of $M + N + 2$ lines of constant generalised co-ordinates

$$\begin{aligned} (u_j, v) & \quad 0 \leq j \leq M, \quad 0 \leq v \leq 1, \\ (u, v_1) & \quad 0 \leq l \leq N, \quad 0 \leq u \leq 1, \end{aligned}$$

where u_j and v_1 satisfy

$$\begin{aligned} 0 = u_0 < u_1 < \dots < u_M = 1, \\ 0 = v_0 < v_1 < \dots < v_N = 1, \end{aligned}$$

is mapped onto given lines in F :

$$\underline{W}(u_j, v) = \underline{F}(u_j, v) \quad 0 \leq j \leq M$$

and

$$\underline{W}(u, v_1) = \underline{F}(u, v_1) \quad 0 \leq l \leq N. \quad (1-3)$$

$\underline{F}(u, v)$ denotes a vector-valued parametric representation of the interior and boundary of F . Let ϕ_i , ψ_k be functions which satisfy the conditions

$$\begin{aligned} \phi_i(u_j) = \delta_{ij} & \equiv \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \quad 0 \leq i, j \leq M, \\ \psi_k(v_1) = \delta_{kl} & \quad 0 \leq k, l \leq N. \end{aligned} \quad (1-4)$$

Then the desired function \underline{W} is defined by

$$\begin{aligned} \underline{W}(u, v) &= \sum_{i=0}^M \phi_i(u) \underline{F}(u_i, v) + \sum_{k=0}^N \psi_k(v) \underline{F}(u, v_k) \\ &\quad - \sum_{i=0}^M \sum_{k=0}^N \phi_i(u) \psi_k(v) \underline{F}(u_i, v_k). \end{aligned} \quad (1-5)$$

It may easily be verified that equations (1-3) hold for this definition of \underline{W} . The functions ϕ_i and ψ_k , undefined so far except that they satisfy the conditions in (1-4), are called blending functions. The

simplest choice for \underline{w} is when $M = N = 1$ and

$$\begin{aligned}\phi_0(u) &= 1 - u, & \phi_1(u) &= u, \\ \psi_0(v) &= 1 - v, & \psi_1(v) &= v.\end{aligned}$$

Equation (1-5) then becomes explicit as

$$\begin{aligned}\underline{w}(u,v) &= (1-u) \underline{F}(0,v) + u \underline{F}(1,v) + (1-v) \underline{F}(u,0) + v \underline{F}(u,1) \\ &\quad - (1-u)(1-v) \underline{F}(0,0) - (1-u)v \underline{F}(0,1) \\ &\quad - u(1-v) \underline{F}(1,0) - uv \underline{F}(1,1).\end{aligned}$$

As well as having the desired property of (1-3), this equation defines a map from all of $[0,1] \times [0,1]$ and by drawing images of $u = \text{constant}$, $v = \text{constant}$, a mesh may be obtained in F .

Gordon and Hall give as an example the case where F is a quarter annulus whose boundary is given by

$$\begin{aligned}\underline{F}(u,0) &= \begin{bmatrix} 1 + u\sqrt{2} \\ 0 \end{bmatrix}, & \underline{F}(u,1) &= \begin{bmatrix} 0 \\ 1 + u\sqrt{2} \end{bmatrix}, \\ \underline{F}(0,v) &= \begin{bmatrix} \cos v\pi/2 \\ \sin v\pi/2 \end{bmatrix}, & \underline{F}(1,v) &= \begin{bmatrix} (1+\sqrt{2}) \cos v\pi/2 \\ (1+\sqrt{2}) \sin v\pi/2 \end{bmatrix}.\end{aligned}$$

The resulting mesh is given in figure 1-7.

As an example of the variation in element size which may be obtained by this method, let the variation in the u -direction remain linear but let $N = 2$ with $v_0 = 0$, $v_1 = 1/2$, $v_2 = 1$. This means that three ψ_k functions need to be defined to satisfy (1-4), for example by

$$\begin{aligned}\psi_0(v) &= (2v - 1)(v - 1), \\ \psi_1(v) &= 4v(1 - v), \\ \psi_2(v) &= v(2v - 1).\end{aligned}$$

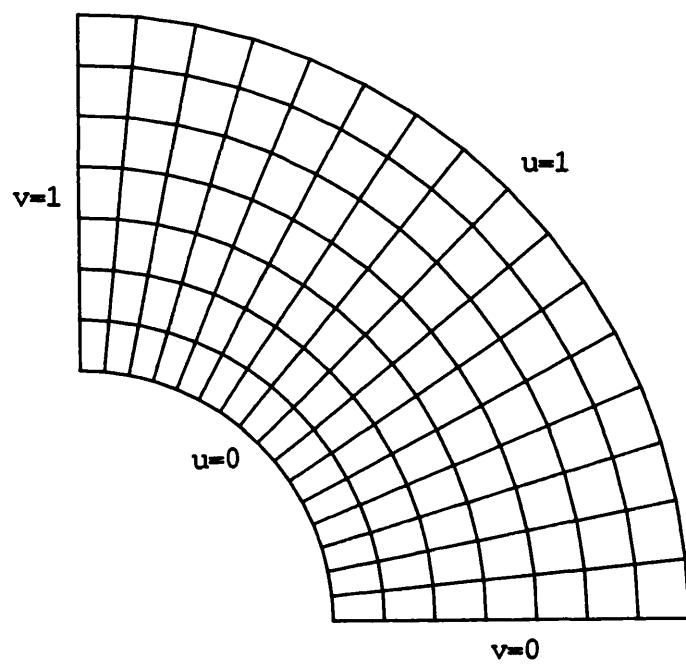


Figure 1-7. Mesh by continuous transfinite mapping in a quarter annulus. $M = N = 1$.

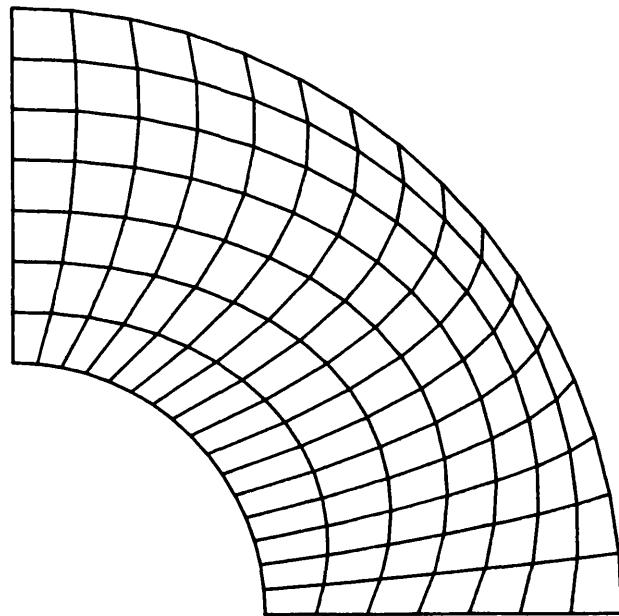


Figure 1-8. Graded mesh by continuous transfinite mapping. $M = 1$, $N = 2$.

When the extra internal constraint line is

$$\underline{F}(u, 1/2) = \begin{bmatrix} 5/4 u (1 - u) + (1 + u/2) \cos \pi/4 \\ (1 + u/2) \sin \pi/4 \end{bmatrix},$$

then the resulting mesh is graded as shown in figure 1-8.

The method of the continuous transfinite mapping can also produce a map which is not one-one. Gordon and Hall give as an example the following situation when $M = N = 1$:

$$\begin{aligned}\underline{F}(u, 0) &= \begin{bmatrix} u \\ 0 \end{bmatrix}, & \underline{F}(u, 1) &= \begin{bmatrix} u \\ 3.4 (u - 1/2)^2 + 0.15 \end{bmatrix}, \\ \underline{F}(0, v) &= \begin{bmatrix} 0 \\ v \end{bmatrix}, & \underline{F}(1, v) &= \begin{bmatrix} 1.5 - 2 (v - 1/2)^2 \\ v \end{bmatrix}.\end{aligned}$$

The resulting mesh, which displays overspill, is shown in figure 1-9. In this case, this can be avoided by inserting an extra constraint at $u = 1/2$:

$$\underline{F}(1/2, v) = \begin{bmatrix} 0.5 \\ 0.15 v \end{bmatrix}$$

to ensure that the mesh passes through the minimum section, as shown in figure 1-10.

Discrete form. Haber et al. [4] offer an alternative to this method by representing patch boundaries in a discrete form by a list of points rather than by specified continuous descriptions. They point out that boundaries which have no exact mathematical representation present no difficulties when presented in discrete form. Complex curves or a union of piecewise continuous curves may be represented by a series of

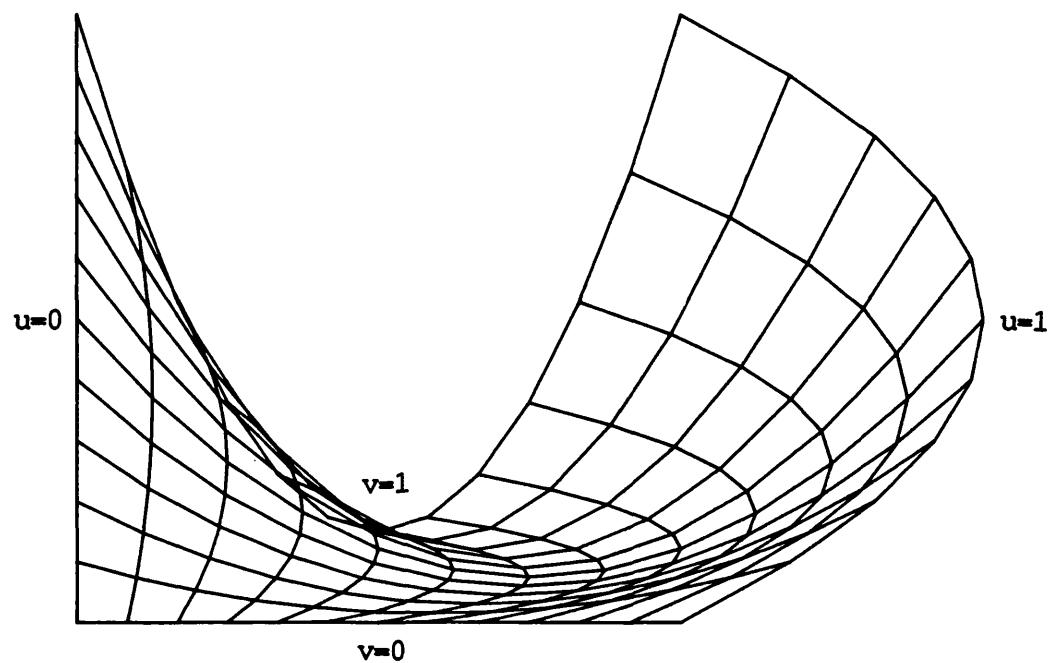


Figure 1-9. Mesh by continuous transfinite mapping. Overspill with $M = 1, N = 1$.

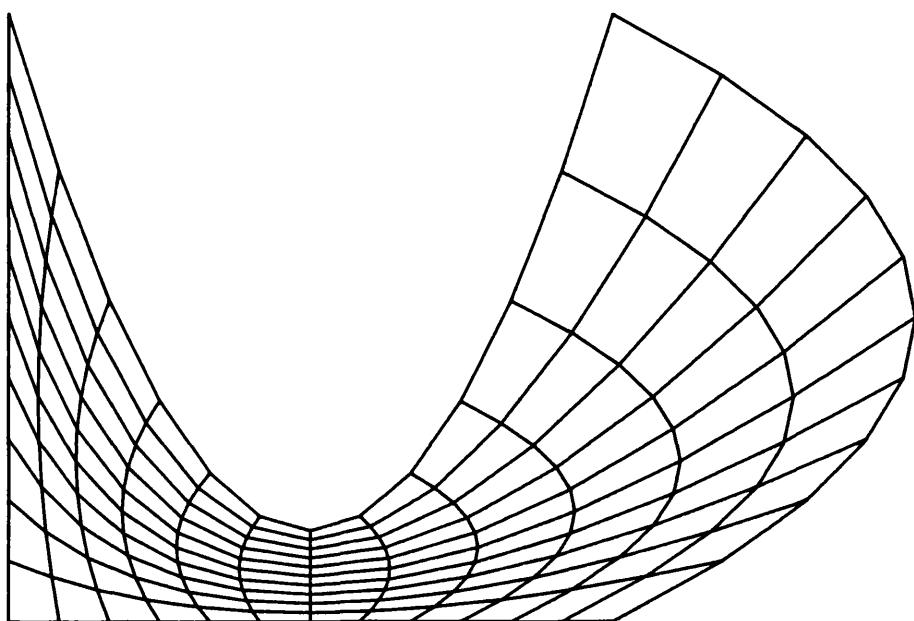


Figure 1-10. Mesh by continuous transfinite mapping. Extra constraint line inserted with $M = 2, N = 1$ to prevent overspill.

simple curves passing through the given points.

1.4 Triangulation

The method of defining a mesh by triangulation has demanded its own place in the literature for at least twenty years. For these methods, triangular elements and two-dimensional regions only are considered but the region in which the mesh is to be defined is allowed to be multiply connected. This brief survey begins with the method described by Frederick, Wong and Edge [5]. Here the positions of the mesh nodes are defined manually and are input using a digitising table. Since node numbers are assigned at the same time as the nodal points are digitised, care must be taken that the order is an efficient one so as to keep the bandwidth as small as possible. There is a full discussion of bandwidth later. The automatic part of their method is in the construction of the elements by a process with attempts to ensure that they are as well shaped (as nearly equilateral) as possible.

The method of Cavendish [6], improving on that suggested by Fukuda and Suhara [7], requires that all the boundary nodes' positions be specified but selects the positions of the interior nodes automatically by superimposing over the domain a grid of squares and randomly generating a node in each sub-square. The random position of the generated node is not universally accepted - for example no node is allowed to be placed too close to the boundary, anticipating that the resulting element should be as well shaped as possible. By alternatively defining the domain as a disjoint union of sub-regions, he allows the node density to vary from one sub-region to the next. Having chosen the positions of all the nodes, his method continues by deciding how best

they should be joined to form the triangular elements, in a similar way to that of Frederick, Wong and Edge. He defines a quantity which is a measure of how close to equilateral is each element. Having defined a mesh which fully covers the domain, the method then attempts to improve on its quality by redefining the x- and y-coordinates of each interior node as the average of the x- and y-coordinates of all the adjacent nodal points, a process he calls "smoothing". He suggests that several passes of the smoothing process should be completed. The nodes of the mesh then have to be numbered efficiently.

The methods of Bykat [8] and Sadek [9] differ from those in references [5, 6 and 7] in that the nodes and elements are generated simultaneously working inwards from the boundary, the nodes of which are specified initially.

The significant difference between the work of Cavendish and that described by Shaw and Pitchen [10] is that the latter workers have abandoned the idea of randomly locating the interior nodes. Their method places the interior nodes - again the boundary nodes' positions are specified to be compatible with the desired density within the sub-region - on a regular grid which, when connected, form equilateral triangles.

One feature which all of these element generation algorithms have in common is that they spend most of their time ensuring that the domain is covered and that no two triangles overlap. Lo [11] has put his method of element formation into a mathematically theoretical context by proving that his triangulation is the Delaunay triangulation. By doing so, he has devised an algorithm which is more efficient than those already covered in this section. There follows a parenthesis in the discussion to define what is meant by Delaunay triangulation. The nomenclature is that used by Sibson [12].

Interpolation and the Delaunay triangulation. The Dirichlet tessellation of a finite set of distinct data points is obtained by associating with each data point a region ("tile") consisting of that part of the plane strictly closer to its generating data point than to any other. Thus the tile of the data point A is bounded by the perpendicular bisector of lines AB for all other data points B. Tiles are open, convex polygonal regions and each tile contains precisely one data point. Two data points are said to be contiguous if their tiles share a common edge. Normally tiles meet in threes and if contiguous data points are joined by edges, a triangulation results. This is the Delaunay triangulation and by its manner of construction it is unique: it is the dual of the Dirichlet tessellation. An example is shown in figure 1-11. Each point where three tiles meet is the circumcentre of the corresponding Delaunay triangle. Occasionally more than three tiles meet at a common point. In this case, the generating data points must be concyclic and the Delaunay triangle degenerates into a cyclic polygon. It is here, and only here, that any ambiguity can arise. Sibson calls this construction a pretriangulation. In practice the triangulation can be completed by arbitrarily triangulating each cyclic polygon containing more than three sides. This is called the completion of the pretriangulation.

Delaunay triangulation finds useful applications in interpolation over a two-dimensional domain and in the drawing of contours. In this field, the quality of the interpolation or contours is governed by the quality of the triangulation. Unlike in finite element mesh generation where the mesh quality may be improved by changing the nodal positions by a technique like Cavendish's "smoothing", for interpolation by this method all that can change is the triangulation. If a triangulation can be formed to cover the domain of the data points, linear (or more

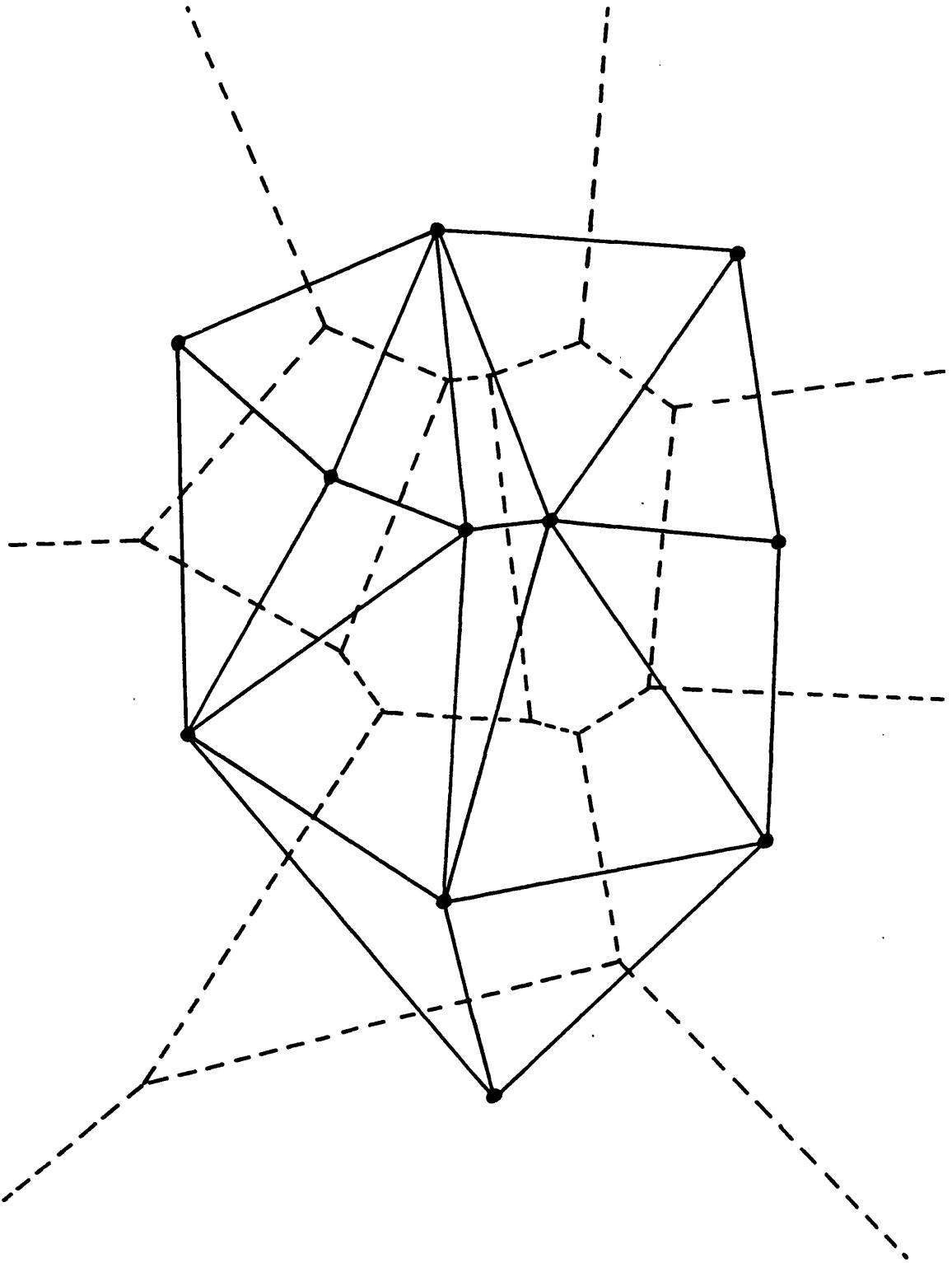


Figure 1-11. Example of the Dirichlet tessellation (dotted lines) and Delaunay triangulation (solid lines) for a set of data points.

complex) interpolation may then be used for any point which lies in the convex hull of the set. Sibson proves that when one such method of improvement, the "max-min" angle criterion, is applied consistently, the final triangulation obtained is the Delaunay triangulation (or the completion of the pretriangulation in a degenerate case). This method - and it can be used equally well when triangulating quadrilateral finite element meshes - is used for selecting the better diagonal of the quadrilateral formed by two adjacent triangles. It can be clearly seen that its aim is to make each triangle as nearly equilateral as possible. Stated formally, this selection criterion is as follows.

If two triangles in a triangulation have a common edge and the quadrilateral which they define is strictly convex, then the diagonal should be selected which maximises the minimum of the six angles of the two triangles formed.

It is not immediately obvious that this algorithm is guaranteed to terminate but this is the case if diagonals are changed only when strictly required.

A property of Delaunay triangulation is that three data points form a Delaunay triangle if, and only if, the circumcircle defined by these three points contains no other data points. This is a direct consequence of the definition of the Delaunay triangle. In practice, this is the property which is used to construct the Delaunay triangulation of a set of data points. Starting at the boundary - boundary edges are assured to be part of the triangulation - an edge CD is considered. Of all the candidates for the third vertex of the triangle of which CD is a side, the data point E for which angle \hat{CED} is the largest should be selected. In this incremental method, the triangulation advances inwards until all data points become part of the triangulation and the whole

region is covered.

With this theoretical background, Lo's method may now be described. Firstly, the boundary nodes are selected, then the interior nodes defined along suitably spaced horizontal lines. As with Cavendish's method, no node is allowed to be positioned too close to the boundary. To check for this, for each candidate interior node X close to the boundary, the nearest boundary element edge $P_m Q_m$ is found: that is

$$XP_m + XQ_m \leq XP_i + XQ_i$$

for all boundary edges $P_i Q_i$. A quantity κ is then calculated for triangle $XP_m Q_m$ which is a measure of its closeness to being equilateral. κ is defined by

$$\kappa = 4\sqrt{3} \frac{\Delta}{XP_m^2 + P_m Q_m^2 + Q_m X^2} \quad (1-6)$$

where Δ denotes the area of triangle $XP_m Q_m$ and the factor $4\sqrt{3}$ is introduced so that κ attains its maximum value of unity when $XP_m Q_m$ is equilateral. If the calculated value of κ exceeds an arbitrary value - Lo suggests $1/2$ - then the interior point X is retained. Otherwise it is discarded as a mesh node. Lo's method continues by defining the mesh elements to be the Delaunay triangulation of the nodal points. Since the concept of Delaunay triangulation described earlier is for convex hulls only, this procedure involves some extra complication but this is overcome by keeping careful check of direction within the mesh: on which side of the considered element edge should the next node lie?

Having defined the triangulation, Lo calculates κ_i for each triangle by equation (1-6) and then $\bar{\kappa}$ as the geometric mean of all the calculated κ_i ,

$$\bar{\kappa} = \left[\prod_{i=1}^{n_e} \kappa_i \right]^{1/n_e}, \quad (1-7)$$

as a global measure of the mesh's quality. n_e denotes the number of elements in the mesh. He then applies Cavendish's smoothing formula to all the interior nodes but accepts the newly calculated nodal position only if $\bar{\kappa}$ has increased from the previous arrangement. Figure 1-12 shows an attempt to reproduce an example given in Lo's paper to indicate the sort of mesh which may be obtained by this method. Two passes of Cavendish's smoothing process have been completed and $\bar{\kappa}$ has risen from its initial value of 0.9520 to 0.9654.

Lo's method produces meshes with good quality elements in multiply-connected domains. Although he says different grading of elements can be obtained by suitable initial sub-division of the domain, as with all these methods, obtaining a smooth transition between large and small elements can be difficult.

1.5 Laplacian grid generation

Herrmann [13] has suggested an alternative to Cavendish's smoothing technique for the re-positioning of interior nodal points. It assumes that the mesh topology, that is the way the elements connecting the nodes are defined, has already been selected. Cavendish's formula may be written as

$$\underline{x}_i = \sum_{n=1}^{N_i} (\underline{x}_{nj} + \underline{x}_{nk}) \quad (1-8)$$

where \underline{x}_i denotes the position vector of the i th interior node to be

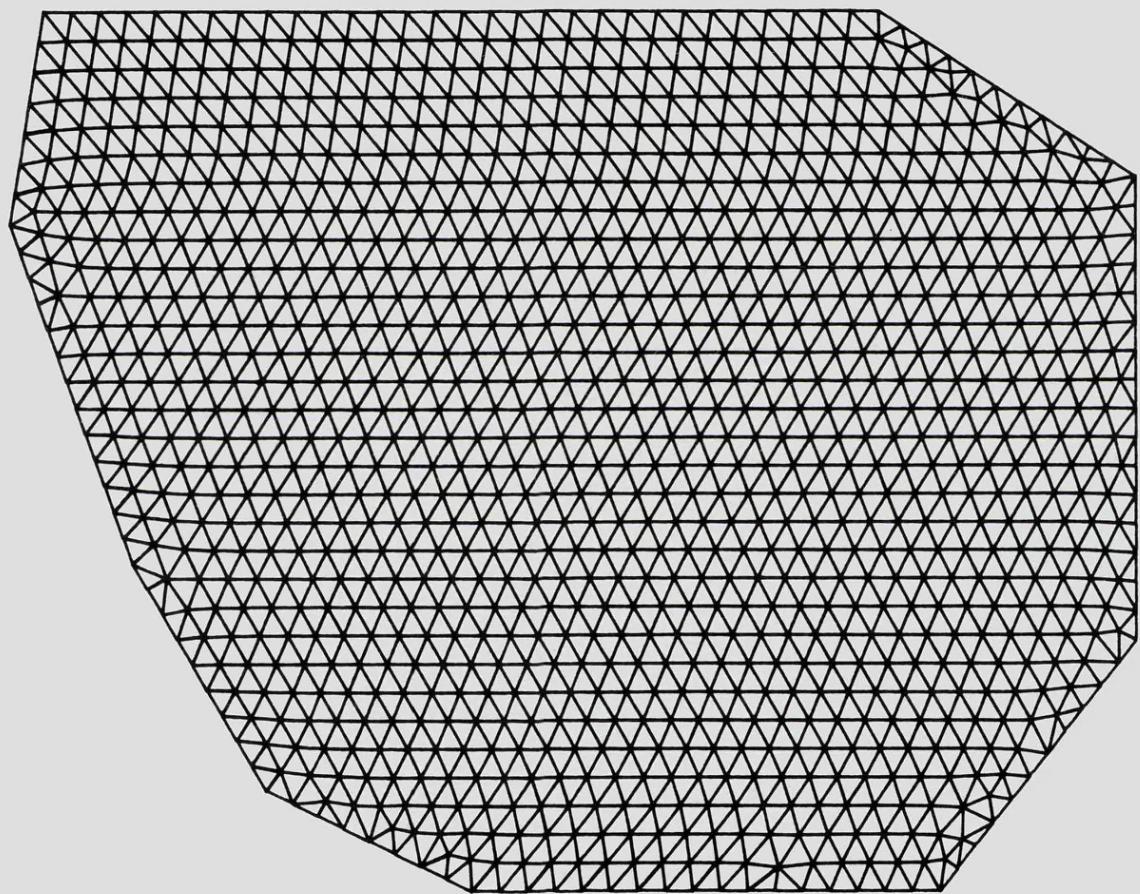


Figure 1-12. Mesh in a simply connected region by Lo's method of triangulation after two passes of smoothing. Node density is constant over the whole region.

positioned, N_i denotes the number of elements which include x_i as a node and x_{ij} and x_{ik} the position vectors of the two adjacent nodes in the n th neighbouring element. Two nodes are said to be adjacent if they share a common edge.

For a rectangular grid where every interior node is at the junction of four elements, equation (1-8) is equivalent to the finite difference relaxation formula for solving Laplace's equation in a two-dimensional region, hence the name of this method. Equation (1-8) is satisfactory if all the elements are of about the same size. However, Herrmann has found that it does not reflect the spacing of boundary nodes when this is irregular. He proposes the alternative formula, effective for quadrilateral meshes only, of

$$x_i = \frac{1}{4(2-w)} \sum_{n=1}^4 (x_{nj} + x_{nk} - w x_{nl}) \quad (1-9)$$

where x_{nl} represents the position vector of the fourth quadrilateral node, the one which is not adjacent to x_i . w is a constant and $0 \leq w \leq 1$. When w is set to zero, equations (1-8) and (1-9) are equivalent. $w = 1$ produces the best meshes and Herrmann actually solves the linear system of equations in (1-9) which is not the same as Cavendish's recommendation of applying a few passes of equation (1-8) to smooth the nodal positions. Unfortunately, when an iterative method is used for solving (1-9), the number of iterations required rises sharply as w attains its optimum value.

1.6 Modified quadtree method

Shephard and Yerry [14] have adapted a technique from three dimensional solid modelling, "octree encoding", to two dimensional finite element mesh generation and have called it the modified quadtree method. They say their method can be extended to three dimensional problems.

In quadtree encoding, the domain to be modelled is placed inside a suitably sized square. The square is subdivided into four quadrants and each quadrant is tested to see whether it lies completely inside the domain, completely outside the domain or partially inside the domain. Quadrants lying wholly inside or outside the domain are appropriately flagged and partial quadrants are further subdivided into four more quadrants which themselves are tested. This process is repeated until a satisfactory resolution has been obtained. The result of this testing may be represented in a tree structure or quadtree. Moreover, because each sub-square may be in one of only three states, computationally this representation is very efficient: integer arithmetic may be used for hardware manipulation to speed up the modelling process. Figure 1-13 shows a quadtree representation of a quarter of an elliptical region.

By this method, any two dimensional region may be represented by the union of a set of squares to any desired resolution. For finite element work, however, Shephard and Yerry report the following drawbacks.

- (i) The interior may be represented by a small number of large elements.
- (ii) Neighbouring elements may be divided to a different resolution, introducing difficulties of continuity between elements.
- (iii) Boundaries which are not parallel to the x- and y-axes are represented by jagged edges.

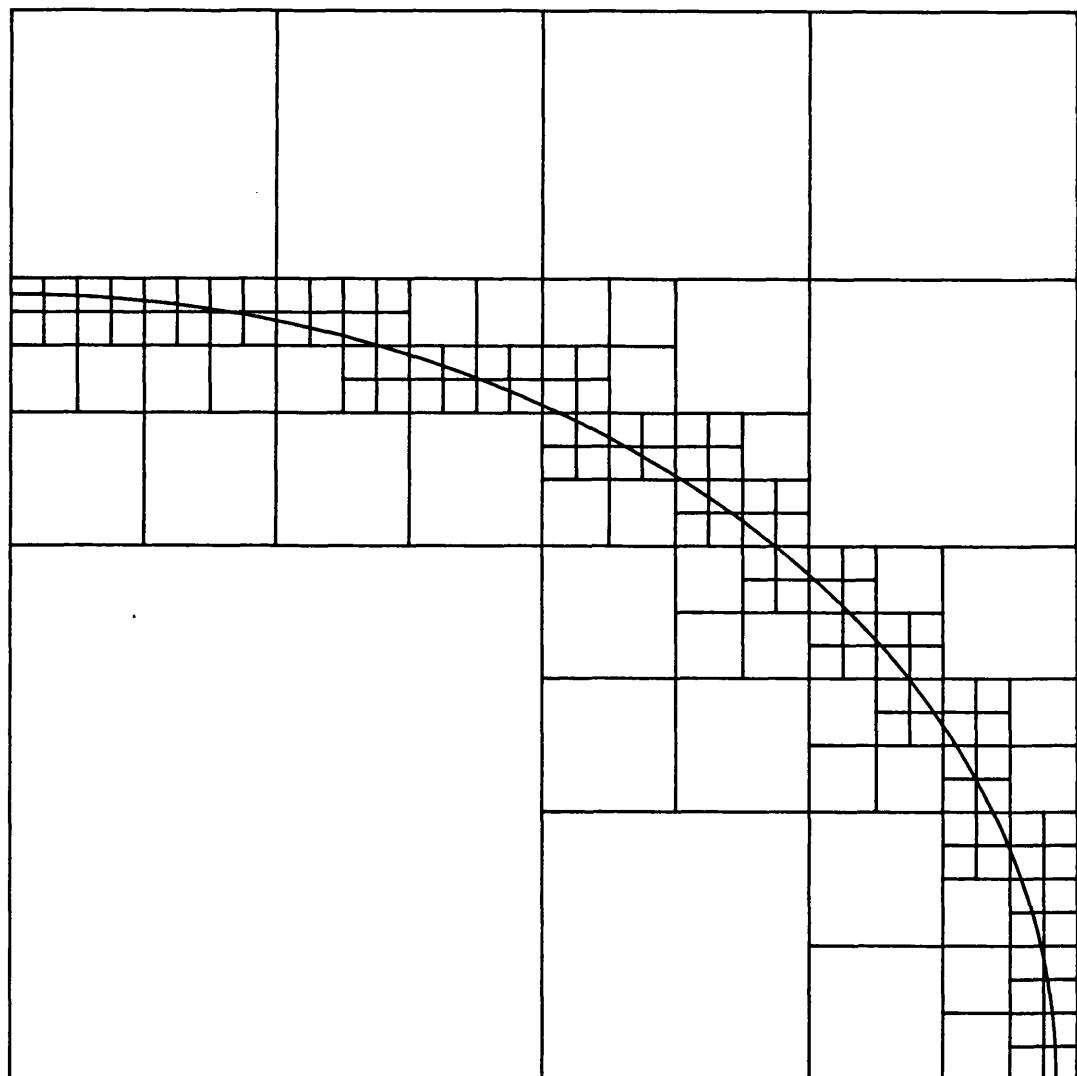


Figure 1-13. Representation of a quarter of an ellipse by quadtree encoding.

(iv) More elements than are usually desired are needed for areas near boundaries.

Shephard and Yerry have overcome these difficulties by (i) sub-dividing even wholly contained sub-squares to a minimum resolution; (ii) introducing transition elements between squares of a different size. For (iii) and (iv), the quadtree encoding technique is modified in the neighbourhood of the boundary. Intersection points between the domain boundary edges and the edges of the sub-squares are calculated and the boundary sub-squares' corners are cut off to represent the shape of the boundary in a better way, while retaining the integer quadtree structure for efficiency. Finally to obtain a mesh with more satisfactorily shaped elements, Cavendish's smoothing technique (equation (1-8)) is applied to all interior nodes. Figure 1-14 shows a mesh drawn inside a quarter of an ellipse to illustrate this method. Figure 1-15 shows the same mesh after one and five passes of Cavendish's smoothing process. This figure illustrates that smoothing a mesh does not inevitably improve the shape of all elements: one small but well shaped element in the original mesh has been transformed into a long, thin, badly shaped element in figure 1-15(b).

By this method it follows that areas near to intricately shaped boundaries are automatically modelled by small elements: it is not immediately clear how to introduce small elements in regions which do not have intricate boundaries except by manual intervention.

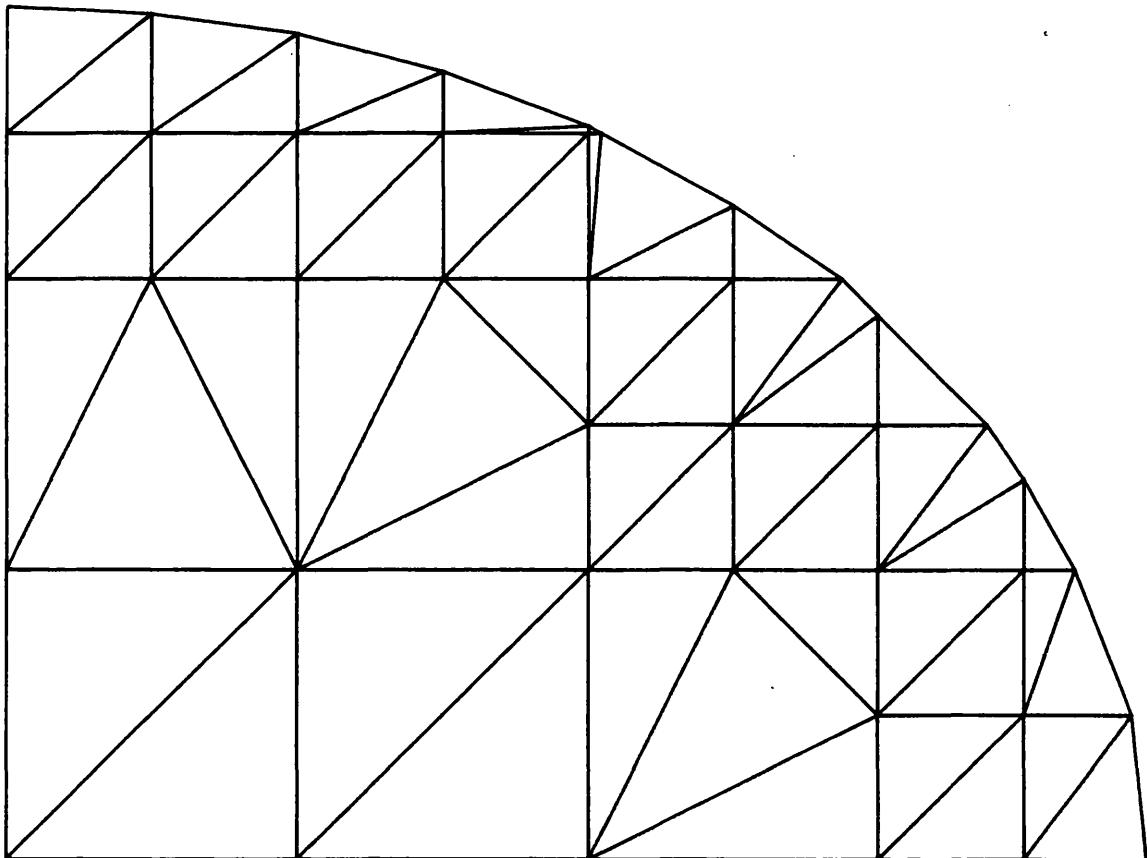


Figure 1-14. Finite element mesh in a quarter of an ellipse by the modified quadtree encoding method of Shephard and Yerry.

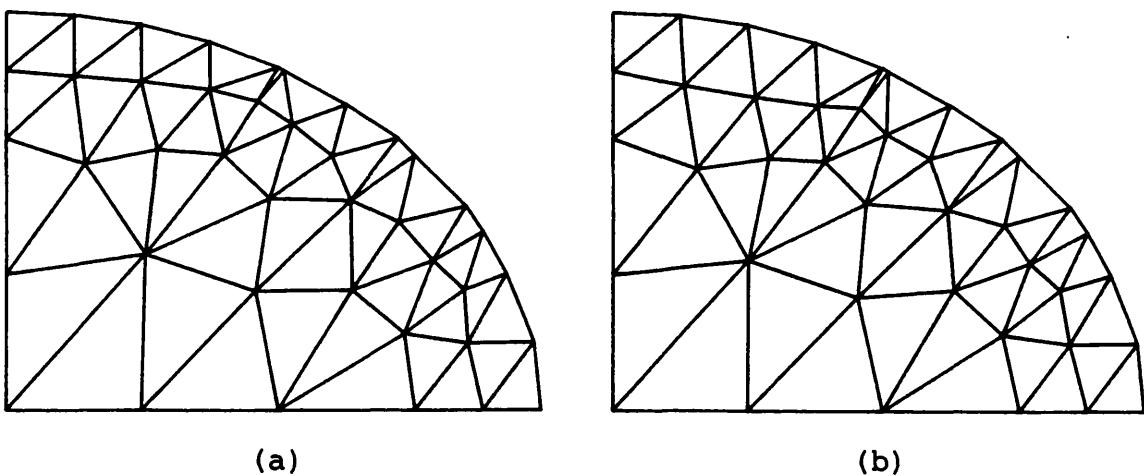


Figure 1-15. Mesh of the modified quadtree encoding method after
(a) one smoothing pass, (b) five smoothing passes.

CHAPTER 2

Complex variable theory and the Schwarz-Christoffel transformation

2.1 Background

The motivation for the work described in this thesis came at about the same time as the author assisted in the presentation of an undergraduate course given in the Department of Engineering at Leicester University in 1977 [15]. The aim of the short course was to give final year mechanical engineering students some practical experience of using a finite element computer program, filling the gap between theoretical paper calculations (necessarily very limited in size) and access to a large industrial package (often intimidating to the beginner). The program was specially written by the author to be used as a "black box", inputting a data file defining a mesh and boundary conditions and outputting calculated displacements and stresses. Substantial error checking was built in to the program to ensure that the mesh was correctly specified: any mistake in the data file generated a helpful message to point out where the error had probably occurred.

The most straightforward problem set was to find the stress distribution in a thin plate containing a central circular hole (figure 2-1). A uniform tension stress was specified as acting on the two faces AB and CD. Conditions of symmetry mean that, in solving a problem of this sort by the finite element method, only one quarter of the plate need be considered. The problem facing the students came down to defining a finite element mesh in the two dimensional region shown in figure 2-2. The undergraduate class was divided into groups of two with each pair being asked to draw a mesh composed of three-noded

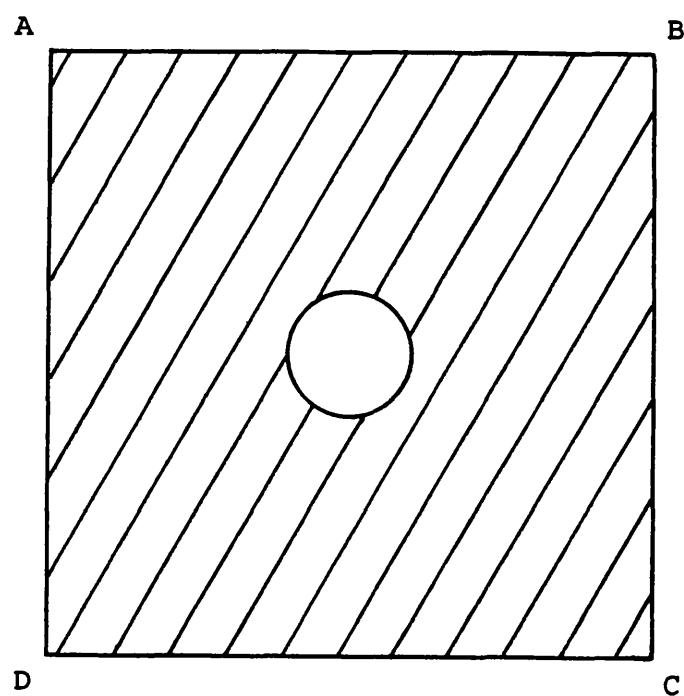


Figure 2-1. Student exercise: thin square plate with central circular hole.

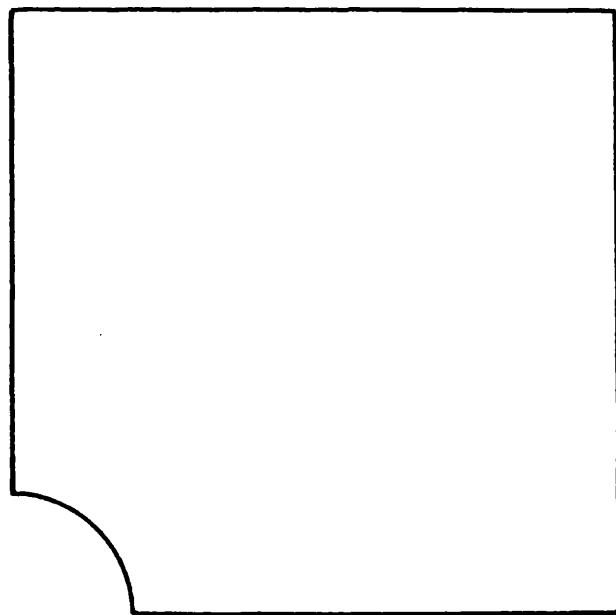


Figure 2-2. Quarter of plate in which mesh was to be drawn.

triangular elements. The total number of nodes of the mesh was set according to each group's ability. The advice on the construction of the mesh ran along the following lines.

There should be small elements in the area where the stress is expected to vary most rapidly (i.e. near the hole) and large elements away from the hole.

Each element should be as nearly equilateral as possible.

The change in element size from small to large should be as gradual as possible.

In practice, for this particular shape, having decided how many nodes there should be around the circular arc and having decided that they should be equally spaced, if the student follows the advice on making each triangle as nearly equilateral as possible then the mesh more or less draws itself, as shown in figure 2-3: by choosing the positions of some of the boundary nodes and applying systematically the equilateral triangle rule, the interior of the mesh is defined as well. If one could generalise this principle and choose just the positions of the boundary nodes and let the interior nodes take care of themselves, while ensuring that the elements were well shaped, then one would have the beginnings of a mesh generator. The size of the elements would be controlled by there being large elements where the boundary nodes were widely spaced and small elements where they were more closely grouped. Moreover since just the boundary of the shape was being considered, one would not need to be concerned about intermediate discretisation into patches, upon which other mapping methods depend.

All that is required is a device for ensuring that the elements of the mesh are as well shaped as possible. Could one find a mapping from a region where a mesh is composed entirely of regular triangles to the target region which has the special property of preserving element shape?

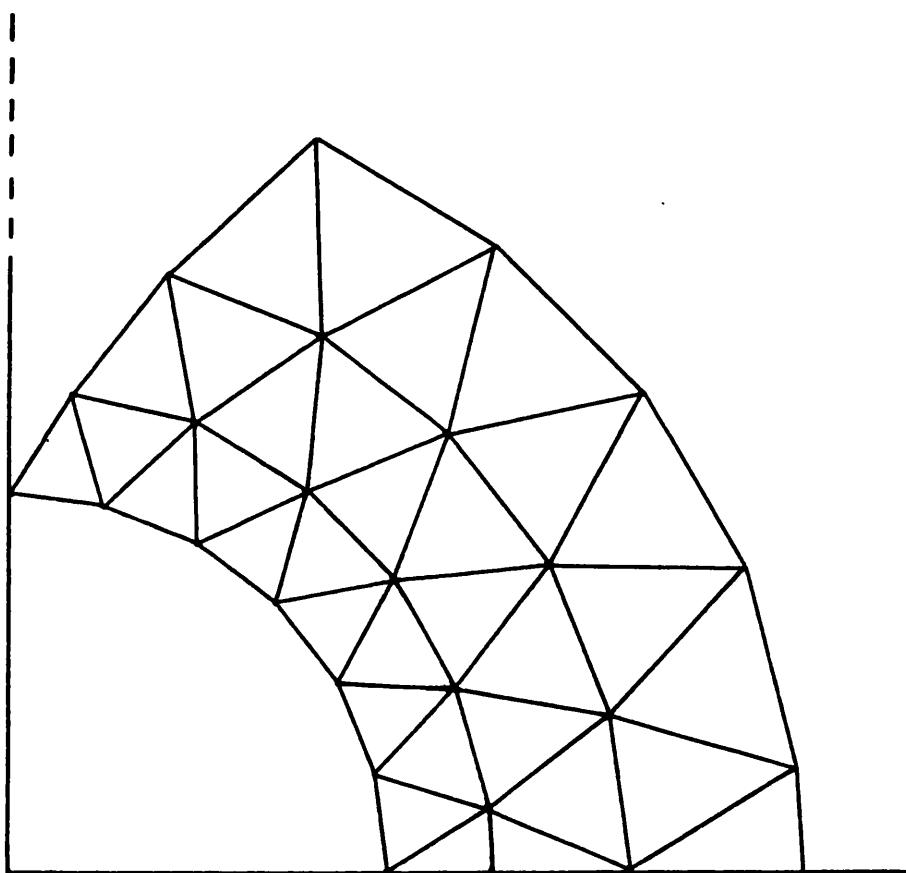


Figure 2-3. Typical triangular mesh with well shaped elements.

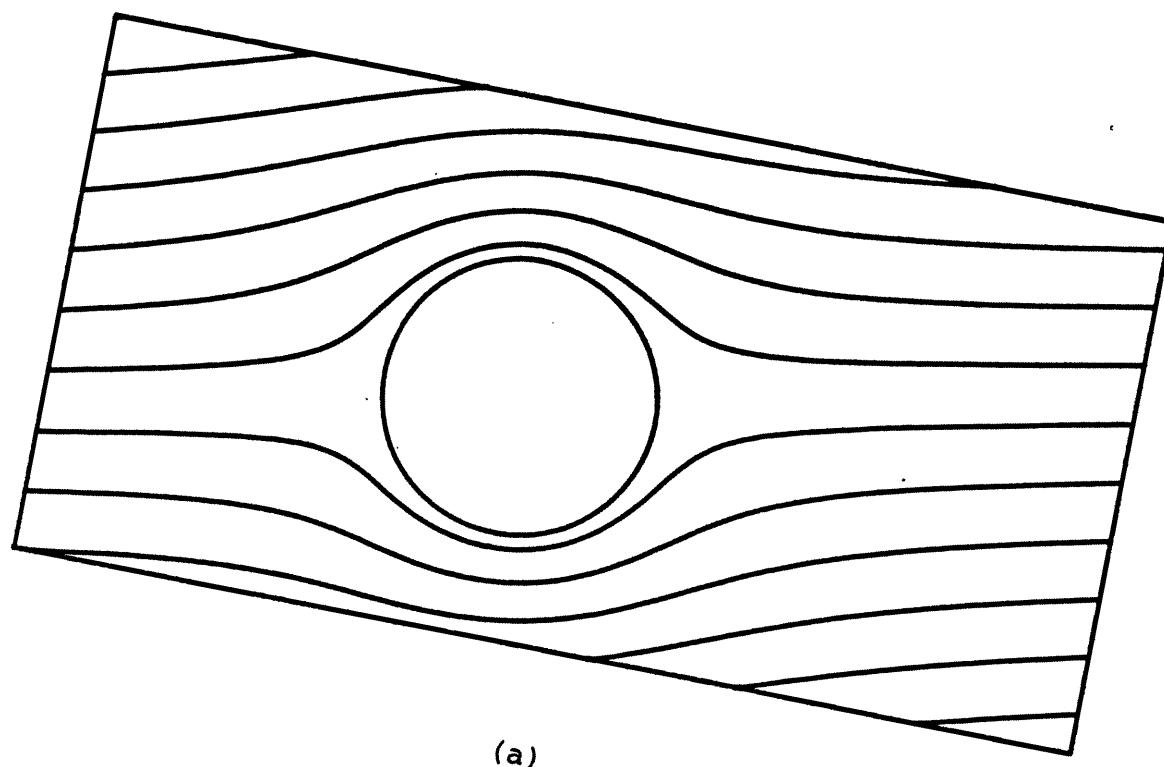
A powerful tool at the disposal of the researcher is the conformal mapping of complex variable theory. Two of its most well known uses are for solving potential problems and in the classical theory of hydrodynamics for finding the streamlines and velocities of fluid flow. Figure 2-4 shows examples from this latter field. The curves indicate the streamlines of idealised, inviscid fluid flow [16, 17]. Conformal mappings will be used here to supply the missing ingredient of the method briefly sketched out above. It should be noted immediately that by going to complex variable theory for suitable mappings which can be used in finite element mesh generation, a necessary and unfortunate restriction is that only two-dimensional situations may be considered. Nevertheless, much valuable work can be done in two dimensions and many three-dimensional situations can be represented by two-dimensional models.

Notwithstanding this restriction, the remainder of this chapter describes the theory upon which the practical work of this thesis depends.

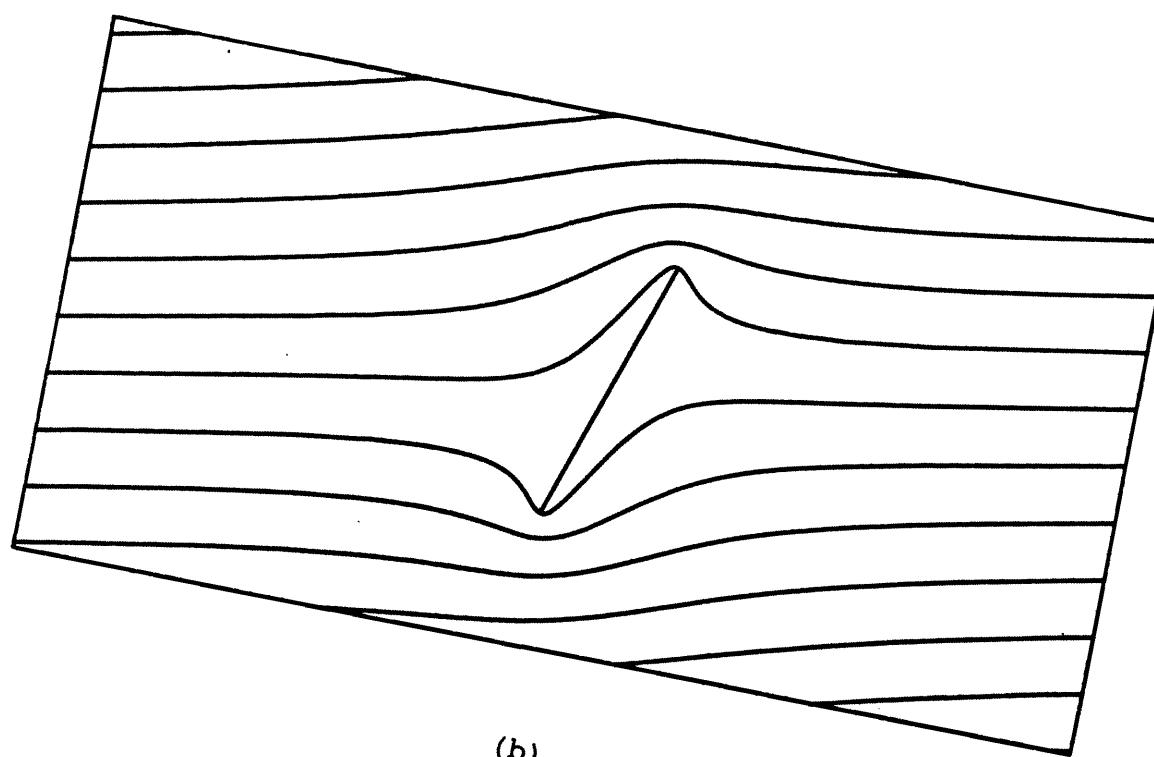
2.2 Complex variable theory

Some of what follows in this section also appears in appendices 1 and 2. A complete description is given here for continuity, completeness and because some notation has been slightly changed.

Let the finite n-sided polygon P with vertices P_0, P_1, \dots, P_{n-1} be considered. The vertices may be represented by complex numbers w_0, w_1, \dots, w_{n-1} . The mapping between a standard domain and P is considered. The two standard domains considered here are the unit circle $|z| < 1$ and the upper half-plane $\text{Im } \zeta > 0$. If the interior angles of

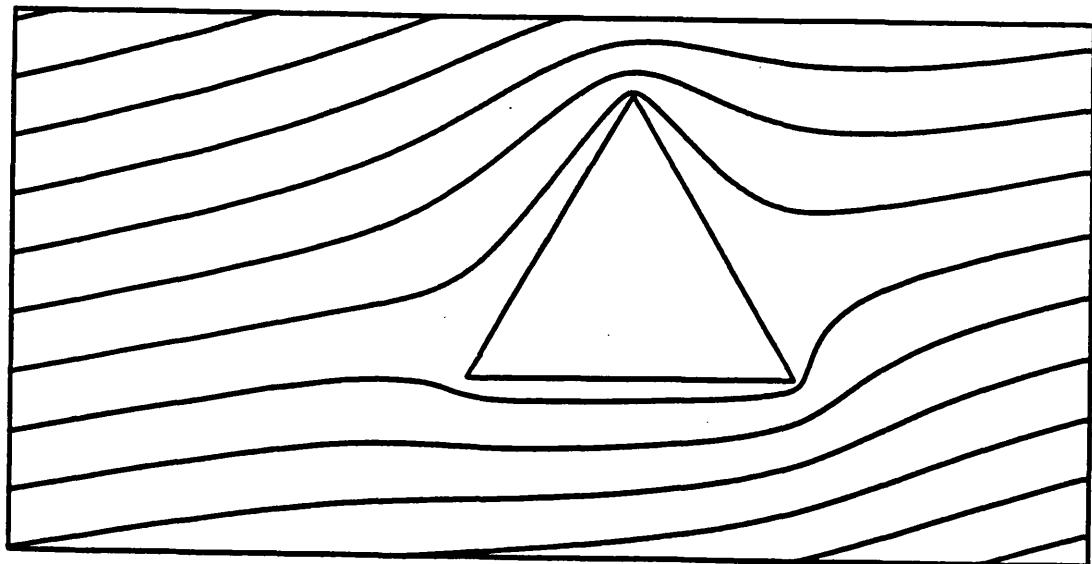


(a)

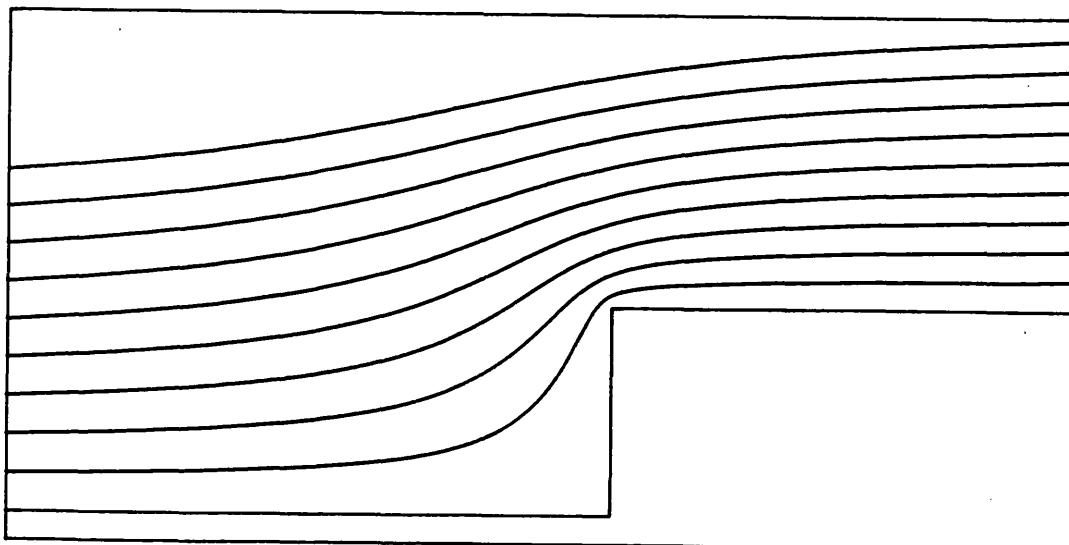


(b)

Figure 2-4. Inviscid two-dimension fluid flow (a) past a circular cylinder, (b) past a thin plate.



(c)



(d)

Figure 2-4 continued. Inviscid two-dimension fluid flow (c) past a triangular prism, (d) in a semi-infinite region bounded by a 90 degree stepped wall.

P are $\alpha_0, \alpha_1, \dots, \alpha_{n-1}$ and if δ_k is defined by

$$\delta_k = 1 - \alpha_k/\pi \quad (2-1)$$

for $k = 0, 1, \dots, n-1$, then P will be supposed to be non-degenerate, namely $\alpha_k < 2\pi$ for all k . The condition that P is finite gives $\alpha_k > 0$ so that

$$|\delta_k| < 1 \quad (2-2)$$

for all k . In addition, the usual condition for a polygon,

$$\sum_{k=0}^{n-1} \delta_k = 2 \quad (2-3)$$

holds for δ_k . Then a standard text on complex variable theory (e.g. [18]) states that the Schwarz-Christoffel equation

$$w = f_1(\zeta) = a_1 \int_{\zeta_0}^{\zeta} \frac{d\zeta}{\prod_{k=1}^{n-1} (\zeta - \xi_k)^{\delta_k}} + b_1 \quad (2-4)$$

describes a mapping between the upper half ζ -plane and the interior of P . a_1, b_1 and ζ_0 are complex constants with $\operatorname{Im} \zeta_0 \geq 0$. Once ζ_0 has been chosen (this may be done arbitrarily), the choice of b_1 determines the position of the polygon in w -space. The value of a_1 controls the size of the polygon by $|a_1|$ and its orientation by $\arg a_1$. Although α_0 does not appear explicitly in (2-4), it is included implicitly by equation (2-3). The mapping has the property that angles are preserved - that is the angle of intersection of any two smooth curves in one region is equal, in both magnitude and sense of rotation, to the angle of intersection of the image curves in the mapped region (conformal mapping.) Moreover if the domain of f_1 is extended

to include the real ζ -axis, then P_0 is the image of infinity and the other vertices of P , P_1, P_2, \dots, P_{n-1} are the images of the points $\xi_1, \xi_2, \dots, \xi_{n-1}$ all of which satisfy $\operatorname{Im} \xi_k = 0$ and

$$\xi_1 < \xi_2 < \dots < \xi_{n-1}.$$

The sides of P , $\Gamma_1, \Gamma_2, \dots, \Gamma_n$ (figure 2-5) are the images of the segments of the real ζ -axis $\Gamma_1^1, \Gamma_2^1, \dots, \Gamma_n^1$ under f_1 (figure 2-6). f_1 is conformal and analytic in the whole region $\operatorname{Im} \zeta \geq 0$ except at the critical points $\xi_1, \xi_2, \dots, \xi_{n-1}$.

The domain of the mapping may be changed by a Moebius transformation. A Moebius transformation is one which maps a generalised circle into a generalised circle, a generalised circle being an ordinary circle or a straight line. Such a transformation has the form

$$z = \frac{c_1\zeta + c_2}{c_3\zeta + c_4}$$

where c_1, c_2, c_3 and c_4 are complex numbers with $c_1c_4 - c_2c_3 \neq 0$. In order to select the Moebius transformation which changes the real ζ -axis into the unit circle $|z| = 1$, while mapping the infinity-point and zero into $+z_0$ and $-z_0$ respectively with $|z_0| = 1$, the defining constants may be chosen as

$$\frac{z}{z_0} = \frac{\zeta - i}{\zeta + i} \quad (2-5)$$

Insertion of (2-5) into (2-4) gives [18]

$$w = f_2(z) = a_2 \int_0^z \frac{dz}{\prod_{k=0}^{n-1} \left(1 - \frac{z}{z_k}\right)^{\delta_k}} + b_2 \quad (2-6)$$

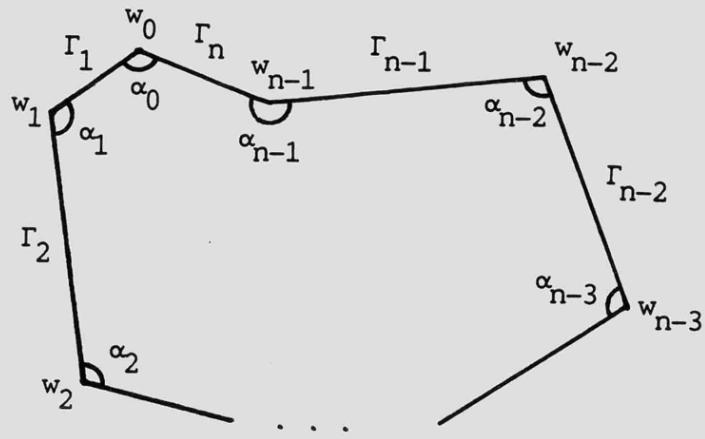


Figure 2-5. Polygon P with vertices at w_0, w_1, \dots, w_{n-1} and sides $\Gamma_1, \Gamma_2, \dots, \Gamma_n$ in the w -plane.



Figure 2-6. The real points $\xi_1, \xi_2, \dots, \xi_{n-1}$ and the line segments $\Gamma_1^1, \Gamma_2^1, \dots, \Gamma_n^1$ which map onto w_1, w_2, \dots, w_{n-1} and $\Gamma_1, \Gamma_2, \dots, \Gamma_n$ respectively under f_1 .

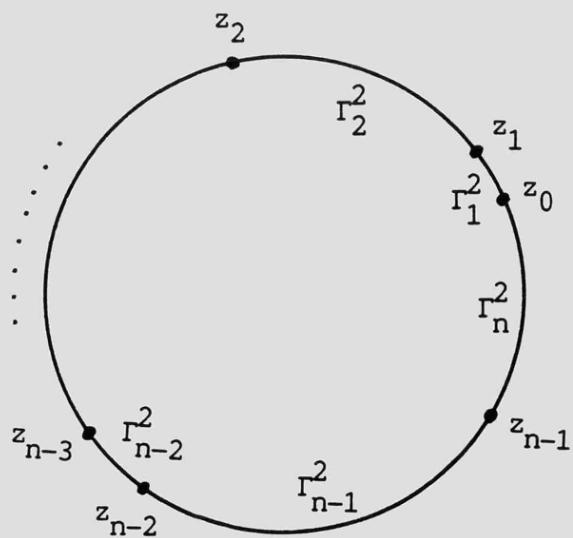


Figure 2-7. The points z_0, z_1, \dots, z_{n-1} and arc segments $\Gamma_1^2, \Gamma_2^2, \dots, \Gamma_n^2$ which map onto w_0, w_1, \dots, w_{n-1} and $\Gamma_1, \Gamma_2, \dots, \Gamma_n$ respectively under f_2 .

Equation (2-6) has a similar form to (2-4). Here though the product runs between 0 and $n-1$, all the δ_k values being included explicitly. Equation (2-6) describes a conformal map from the unit circle $|z| < 1$ to the interior of the polygon P . a_2 and b_2 are complex constants. If the domain of f_2 is extended to include the circle $|z| = 1$, then p_0, p_1, \dots, p_{n-1} are the images of z_0, z_1, \dots, z_{n-1} , points in anti-clockwise order on the unit circle. The points z_0, z_1, \dots, z_{n-1} are called the pre-vertices of mapping f_2 . f_2 is conformal and analytic in the whole region $|z| \leq 1$ except at the critical points z_0, z_1, \dots, z_{n-1} . The sides of P , $\Gamma_1, \Gamma_2, \dots, \Gamma_n$ (figure 2-5) are the images of the arc segments $r_1^2, r_2^2, \dots, r_n^2$ (figure 2-7). Equation (2-5) may be inverted to give ζ in terms of z -

$$\zeta = \frac{i(1 + z/z_0)}{1 - z/z_0} \quad (2-7)$$

To observe how points on the unit circle $|z| = 1$ are mapped by (2-7), let z be defined by

$$z/z_0 = e^{i(\theta-\theta_0)} = \cos(\theta-\theta_0) + i \cdot \sin(\theta-\theta_0)$$

Under the Moebius transformation (2-7), z maps onto the point

$$\begin{aligned} \zeta &= \frac{i[1 + e^{i(\theta-\theta_0)}]}{1 - e^{i(\theta-\theta_0)}} \\ &= \frac{-i[e^{i(\theta-\theta_0)/2} + e^{-i(\theta-\theta_0)/2}]}{e^{i(\theta-\theta_0)/2} - e^{-i(\theta-\theta_0)/2}} \\ &= -\cot \frac{\theta - \theta_0}{2} \end{aligned} \quad (2-8)$$

and it is this equation which links ξ_k in equation (2-4) with z_k in equation (2-6). (2-8) may be represented geometrically by considering figure 2-8. Here O is the centre of a circle of unit diameter AB. The point $z = e^{i\theta}$ is represented by the point M where $\hat{AO}M = \theta - \theta_0$. If AM is produced to meet the tangent at B at X and if BX is considered to be negative upwards, then a simple geometrical argument ($\hat{BXA} = (\theta - \theta_0)/2$) gives

$$BX = -AB \cdot \tan \hat{BAX} = -\cot \frac{\theta - \theta_0}{2} \quad (2-9)$$

Equations (2-8) and (2-9) being identical, this gives a straightforward way of switching between the mapping equation (2-4) between the upper half ζ -plane and P and equation (2-6) which describes the mapping between the unit circle in the z-plane and P.

2.3 Setting up the problem of finding the defining parameters of the Schwarz-Christoffel transformation

This section describes a way of posing the problem of finding the parameters which define equation (2-6), the Schwarz-Christoffel formula for the conformal mapping between the unit circle $|z| \leq 1$ and P.

Earlier attempts by the author to use Schwarz-Christoffel transformations in a practical way - and it is these which are described in appendices 1 and 2 - used equation (2-4) instead of (2-6). The equation to be used was changed for the following reasons: firstly, it is easier to visualise the three available degrees of freedom which allow equivalent sets of z_k points to be solutions of (2-6); secondly, it allows a more efficient solution process to be used to solve

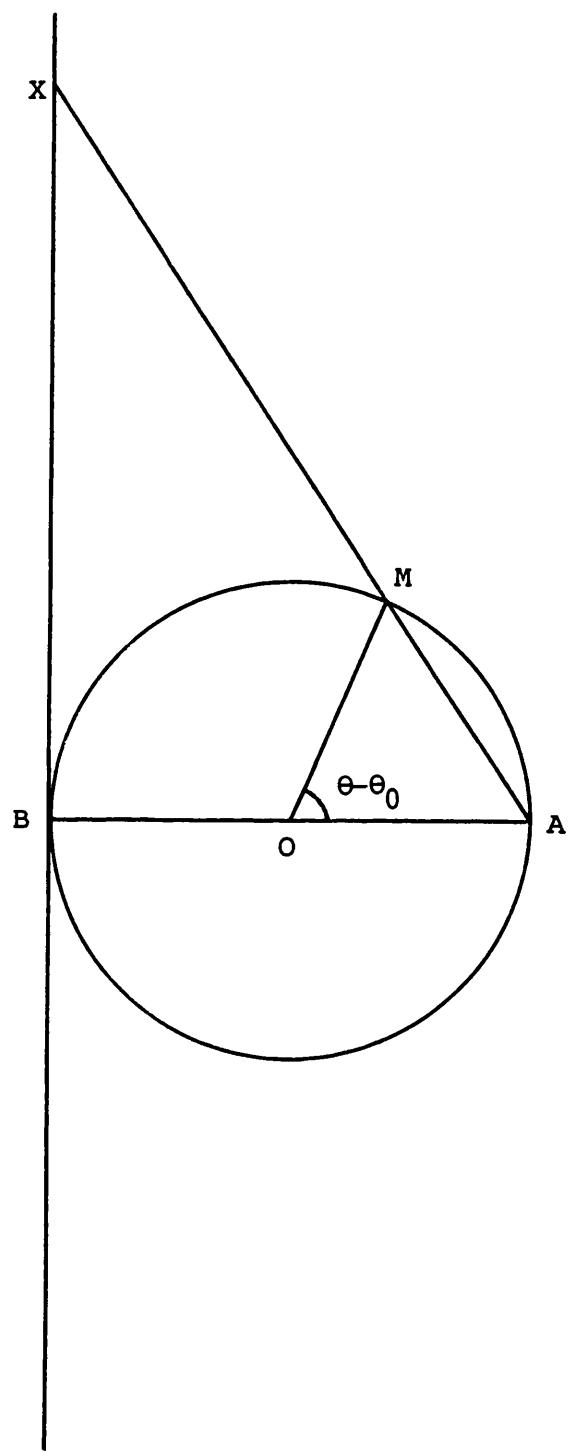


Figure 2-8. Equation (2-8) represented geometrically.

(2-6), described in detail in a later section; and thirdly, although the infinite definite integral (2-4) when $\zeta = \infty$ exists and can be evaluated numerically, using equation (2-6) avoids making a special case in the evaluation thereof.

The unknowns of equation (2-6) are $a_2, b_2, z_0, z_1, \dots, z_{n-1}$. This problem is a non-trivial one with very few analytic solutions.

Indeed even for the case of a general rectangle, z_k can be written only in terms of elliptical functions. In general, then, the problem can be solved only by numerical methods. It will be shown that the case $n = 3$ has a trivial solution so it will be supposed that $n > 3$. The form of the integrand in (2-6) ensures that any set of complex numbers z_k spread around the unit circle in anti-clockwise order define a mapping onto a polygon with the same angles as P . What such a mapping will in general fail to do, though, is get the shape of the polygon right - the relative lengths of the sides will in general be wrong even though the vertex angles are correct.

The difficult part of the problem is finding the z_k points which do get the relative lengths correct. Once this has been achieved, the evaluation of a_2 and b_2 is a trivial task. These are the constants which scale and rotate the obtained polygon, similar to P , onto P itself. Almost all, then, of this section will be devoted to describing a method for obtaining the z_k points. Rather than attempt to find z_k directly, it is natural to find the arguments of z_k, θ_k , where

$$z_k = e^{i\theta_k} \quad \text{and} \quad \theta_0 < \theta_1 < \theta_2 < \dots < \theta_{n-1} < \theta_0 + 2\pi. \quad (2-10)$$

In order to eliminate a_2 and b_2 from consideration, the shape of polygon P is characterised by length ratios β_j , $n-3$ in number, defined by

$$\beta_j = \frac{P_0 P_1 + P_1 P_2 + \dots + P_{j-1} P_j}{P_0 P_1 + P_1 P_2 + \dots + P_{n-3} P_{n-2}} \quad (2-11)$$

for $j = 1, 2, \dots, n-3$. This description of polygon P in terms of the β_j quantities may be thought of as an alternative to the more usual one where the co-ordinates of the vertices are given. The n vertex co-ordinates require $2n$ real quantities to describe P . The alternative description requires $n-3$ β_j length ratios together with $n-1$ angles $\alpha_0, \alpha_1, \dots, \alpha_{n-2}$. The co-ordinates of precisely two distinct vertices are all that remain to complete the description.

As an example, a regular hexagon would be described by the co-ordinates of any two of its vertices together with the following eight quantities

$$\alpha_0 = \alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \frac{2}{3}\pi;$$

$$\beta_1 = \frac{1}{4}, \quad \beta_2 = \frac{1}{2}, \quad \beta_3 = \frac{3}{4}.$$

The numbers $\alpha_1, \alpha_2, \dots, \alpha_{n-2}$ and $\beta_1, \beta_2, \dots, \beta_{n-3}$ define the relative positions of the vertices P_0, P_1, \dots, P_{n-2} . α_0 and α_{n-2} define the remaining two edges and so the final vertex P_{n-1} at their intersection (figure 2-9). The co-ordinates of the two distinct vertices fix the polygon in cartesian space. For P to be defined unambiguously in this way, the assumption is made that $\alpha_{n-1} \neq \pi$. This is not seen as a serious drawback, however, as there would be no purpose in choosing such a point as a vertex. If necessary, though, the vertices could be re-numbered to avoid difficulties.

As a convenient alternative choice of notation, let w_n, z_n, θ_n and δ_n be defined by

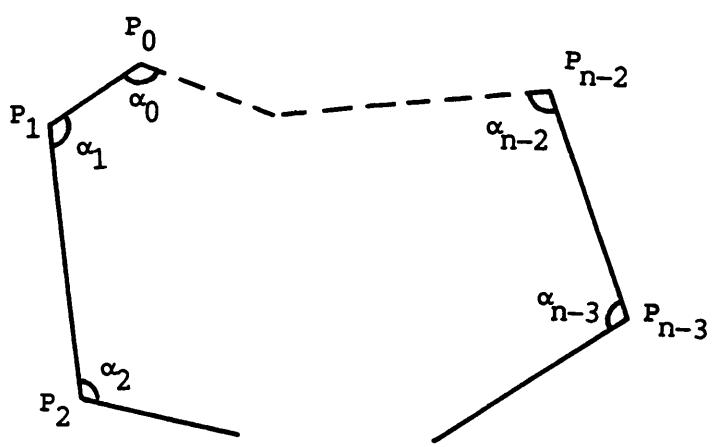


Figure 2-9. Vertex p_{n-1} in P defined at intersection of two implicitly defined edges.

$$w_n = w_0, \quad z_n = z_0, \quad \theta_n = \theta_0 + 2\pi \quad \text{and} \quad \delta_n = \delta_0.$$

Inequality (2-10) becomes

$$\theta_0 < \theta_1 < \theta_2 < \dots < \theta_{n-1} < \theta_n = \theta_0 + 2\pi. \quad (2-12)$$

When w_0, w_1, \dots, w_{n-1} are used to represent the vertices of P , (2-11) becomes

$$\beta_j = \frac{\sum_{k=1}^j |w_k - w_{k-1}|}{\sum_{k=1}^{n-2} |w_k - w_{k-1}|}. \quad (2-13)$$

Then for any z_k in anti-clockwise order around the unit circle, v_k can be calculated by

$$v_k = \int_0^{z_k} \frac{dz}{\prod_{l=0}^{n-1} \left(1 - \frac{z}{z_l}\right)^{\delta_l}} = \int_0^{e^{i\theta_k}} \frac{dz}{\prod_{l=1}^n \left(1 - \frac{z}{e^{i\theta_l}}\right)^{\delta_l}} \quad (2-14)$$

for $k = 0, 1, \dots, n-2$ and γ_j defined by

$$\gamma_j = \frac{\sum_{k=1}^j |v_k - v_{k-1}|}{\sum_{k=1}^{n-2} |v_k - v_{k-1}|} \quad (2-15)$$

for $j = 1, 2, \dots, n-3$. If the vectors $\underline{\gamma}$, $\underline{\beta}$ and $\underline{\theta}$ are defined by

$$\underline{\gamma} = \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \vdots \\ \gamma_{n-3} \end{bmatrix}, \quad \underline{\beta} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_{n-3} \end{bmatrix}, \quad \underline{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{bmatrix}$$

then γ_j has been defined as a function of v_k , v_k as a function of θ and

$$\underline{\gamma} = \underline{\gamma}(\underline{\theta}).$$

The problem of finding $\underline{\theta}$ such that function f_2 maps the unit circle onto the interior of P is equivalent to solving the vector equation

$$\underline{\gamma}(\underline{\theta}) = \underline{\beta}. \quad (2-16)$$

Being a mapping from the space of n -dimensional vectors R^n to the space of $(n-3)$ -dimensional vectors R^{n-3} , $\underline{\gamma}$ is clearly many-one - equation (2-16) does not have a unique solution for $\underline{\theta}$. That there are just three degrees of freedom available is in agreement with the Riemann mapping theorem [18]. This states that a conformal mapping from any simply-connected domain D in the w -plane onto the unit circle $|z| < 1$ is made unique by selecting three arbitrary points on the boundary of D and the three points onto which they map on the boundary of the unit circle. The two sets of points must be in the same order on both boundaries. It is for this reason that the solution of (2-6) is trivial for the case $n = 3$. Any vector $\underline{\theta}$ in R^3 is a solution as long as the three points $e^{i\theta_k}$ are in anti-clockwise order around the unit circle's boundary.

Thus if one solution $\underline{\theta}$ to equation (2-16) can be found, there exist three independent ways in which $\underline{\theta}$ may be transformed so that it remains a solution. These three degrees of freedom will be measured by

real numbers r_1 , r_2 and r_3 .

Let vector $\underline{\theta}'$ be defined by

$$\underline{\theta}' = \begin{bmatrix} \theta_1' \\ \theta_2' \\ \vdots \\ \theta_n' \end{bmatrix} = \begin{bmatrix} \theta_1 + 2r_1\pi \\ \theta_2 + 2r_1\pi \\ \vdots \\ \theta_n + 2r_1\pi \end{bmatrix}.$$

Then it will be shown that $\underline{\gamma}(\underline{\theta}') = \underline{\gamma}(\underline{\theta})$. To prove this, let z_a and z_b be any two points in $|z| \leq 1$ and let their images under the mapping

$$w = \int_0^z \frac{dz}{\prod_{k=1}^n \left(1 - \frac{z}{e^{i\theta_k}}\right)^{\delta_k}} \quad (2-17)$$

be w_a and w_b . The distance between w_a and w_b is

$$\begin{aligned} |w_b - w_a| &= \left| \int_0^{z_b} \frac{dz}{\prod_{k=1}^n \left(1 - \frac{z}{e^{i\theta_k}}\right)^{\delta_k}} - \int_0^{z_a} \frac{dz}{\prod_{k=1}^n \left(1 - \frac{z}{e^{i\theta_k}}\right)^{\delta_k}} \right| \\ &= \left| \int_{z_a}^{z_b} \frac{dz}{\prod_{k=1}^n \left(1 - \frac{z}{e^{i\theta_k}}\right)^{\delta_k}} \right|. \end{aligned}$$

Now let the alternative mapping

$$w = \int_0^z \frac{dz}{\prod_{k=1}^n \left(1 - \frac{z}{e^{i\theta_k'}}\right)^{\delta_k}} \quad (2-18)$$

be considered. The distance between the images of the points
 $z_a' = z_a \cdot e^{2r_1 \pi i}$ and $z_b' = z_b \cdot e^{2r_1 \pi i}$, w_a' and w_b' , is

$$\begin{aligned}
|w_b' - w_a'| &= \left| \int_0^{z_b'} \frac{dz}{\prod_{k=1}^n \left(1 - \frac{z}{e^{i\theta_k'}}\right)^{\delta_k}} - \int_0^{z_a'} \frac{dz}{\prod_{k=1}^n \left(1 - \frac{z}{e^{i\theta_k'}}\right)^{\delta_k}} \right| \\
&= \left| \int_{z_a'}^{z_b'} \frac{dz}{\prod_{k=1}^n \left(1 - \frac{z}{e^{i\theta_k'}}\right)^{\delta_k}} \right| \\
&= \left| \int_{z_a e^{2r_1 \pi i}}^{z_b e^{2r_1 \pi i}} \frac{dz}{\prod_{k=1}^n \left(1 - \frac{z}{e^{i(\theta_k + 2r_1 \pi)}}\right)^{\delta_k}} \right| \\
&= \left| \int_{z_a}^{z_b} \frac{du \cdot e^{2r_1 \pi i}}{\prod_{k=1}^n \left(1 - \frac{ue^{2r_1 \pi i}}{e^{i(\theta_k + 2r_1 \pi)}}\right)^{\delta_k}} \right| \\
&= \left| e^{2r_1 \pi i} \int_{z_a}^{z_b} \frac{dz}{\prod_{k=1}^n \left(1 - \frac{z}{e^{i\theta_k}}\right)^{\delta_k}} \right| \\
&= |w_b - w_a| .
\end{aligned}$$

Thus it can be seen that the distance between the images of any two points z_a and z_b under mapping (2-17) is the same as the distance between the images of the two points z_a' and z_b' - in the same

positions relative to $e^{i\theta_k'}$ as z_a and z_b are to $e^{i\theta_k}$ - under mapping (2-18). Hence the ranges of $|z| \leq 1$ under (2-17) and (2-18) are congruent. This gives the required relationship $\underline{\gamma}(\underline{\theta}') = \underline{\gamma}(\underline{\theta})$.

It is now shown how $\underline{\theta}'$ may be transformed into another vector in R^n , $\underline{\omega}$, so that the condition $\underline{\gamma}(\underline{\theta}') = \underline{\gamma}(\underline{\omega})$ holds. To do this, it is more convenient to use the alternative Schwarz-Christoffel formula (2-4). The ξ_k values which define

$$w = \int_{\zeta_0}^{\zeta} \frac{d\zeta}{\prod_{k=1}^{n-1} (\zeta - \xi_k)^{\delta_k}} \quad (2-19)$$

may be calculated from $\underline{\theta}'$ by

$$\xi_k = -\cot \frac{\theta_k' - \theta_0'}{2} \quad (2-20)$$

for $k = 1, 2, \dots, n-1$ and where $\theta_0' = \theta_n' - 2\pi$. Let ξ_k be transformed by defining

$$\xi_k' = e^{r_2} \cdot \xi_k + r_3 = v \cdot \xi_k + r_3 \quad (2-21)$$

for $k = 1, 2, \dots, n-1$. r_2 and r_3 are real, the exponential being used in (2-21) merely as a convenient way of ensuring that v is positive.

Let ζ_a and ζ_b be any two points in the domain $\text{Im } \zeta > 0$ of the mapping in (2-19) and let their images be w_a and w_b . Let the alternative mapping

$$w = \int_{\zeta_0}^{\zeta} \frac{d\zeta}{\prod_{k=1}^{n-1} (\zeta - \xi_k')^{\delta_k}} \quad (2-22)$$

be considered. Let the points ζ_a' and ζ_b' be defined by

$\zeta_a' = v\zeta_a + r_3$ and $\zeta_b' = v\zeta_b + r_3$. Since $v > 0$, ζ_a' and ζ_b' also lie in the domain of the mapping in (2-22), $\operatorname{Im} \zeta > 0$. Then the distance between the images of ζ_a' and ζ_b' under (2-22), w_a' and w_b' , is

$$|w_b' - w_a'| = \left| \int_{\zeta_a'}^{\zeta_b'} \frac{d\zeta}{\prod_{k=1}^{n-1} (\zeta - \xi_k')^{\delta_k}} \right|$$

$$= \left| \int_{v\zeta_a + r_3}^{v\zeta_b + r_3} \frac{d\zeta}{\prod_{k=1}^{n-1} [\zeta - (v\xi_k + r_3)]^{\delta_k}} \right|$$

$$= \left| \int_{\zeta_a}^{\zeta_b} \frac{vd\zeta}{\prod_{k=1}^{n-1} [v(\zeta - \xi_k)]^{\delta_k}} \right|$$

$$= v \left(1 - \sum_{k=1}^{n-1} \delta_k \right) \left| \int_{\zeta_a}^{\zeta_b} \frac{d\zeta}{\prod_{k=1}^{n-1} (\zeta - \xi_k)^{\delta_k}} \right| = v^{(\delta_0 - 1)} |w_b - w_a|$$

using relationship (2-3). Thus the distance between the images of any two points ζ_a and ζ_b under the mapping (2-19) is a constant multiple of the distance between the images of the two points ζ_a' and ζ_b' - in the same positions relative to ξ_k' as ζ_a and ζ_b are to ξ_k - under mapping (2-22). Hence the ranges of $\operatorname{Im} \zeta > 0$ under

(2-19) and (2-22) are similar.

Finally, $\underline{\omega}$ is defined from ξ_k' by

$$\omega_0 = \theta_0', \quad \omega_k = \omega_0 + \pi + 2 \cdot \tan^{-1} \xi_k'$$

for $k = 1, 2, \dots, n-1$. This equation is effectively the inverse of (2-20) but where ω_k is written for θ_k' and ξ_k' for ξ_k . The appropriate value of arctangent is selected to ensure that

$$\omega_0 < \omega_1 < \dots < \omega_{n-1} < \omega_n = \omega_0 + 2\pi.$$

Similarity of the ranges of $\operatorname{Im} \zeta \geq 0$ under mappings (2-19) and (2-22) ensures that $\underline{\gamma}(\underline{\omega}) = \underline{\gamma}(\underline{\theta}')$.

In summary, it has been shown that if vector $\underline{\omega}$ is defined by

$$\begin{aligned} \theta_k' &= \theta_k + 2\pi r_1 & k &= 0, 1, \dots, n \\ \xi_k &= -\cot [(\theta_k' - \theta_0')/2] & k &= 1, 2, \dots, n-1 \\ \xi_k' &= e^{r_2} \cdot \xi_k + r_3 & k &= 1, 2, \dots, n-1 \\ \omega_k &= \theta_0' + \pi + 2 \cdot \tan^{-1} \xi_k' & k &= 1, 2, \dots, n-1 \\ \omega_0 &= \theta_0' , \quad \omega_n = \theta_0' + 2\pi & & \end{aligned} \tag{2-23}$$

then

$$\underline{\gamma}(\underline{\omega}) = \underline{\gamma}(\underline{\theta}). \tag{2-24}$$

Equation (2-24) means that if one solution $\underline{\theta}$ can be found of equation (2-16) then $\underline{\omega}$ is also a solution. In addition, however, it means that in an iterative scheme to find $\underline{\theta}$ numerically, the current value may be transformed by (2-23) without affecting the current $\underline{\gamma}$ value. This will be shown to be important in solving (2-16) efficiently.

2.4 Geometrical theorem

Notwithstanding all the complex variable theory, in particular the Riemann mapping theorem, upon which much of this work depends, it may be thought that in the transformation from $\underline{\theta}$ to $\underline{\omega}$, symmetry is lost when selecting one particular polygon vertex, θ_0' , about which to project the other polygon vertices onto the diametrically opposite tangent. In this section, a proof is given, from an entirely geometrical standpoint, that this is not the case. The notation $W, X, Y, \dots, \alpha, \beta, \gamma, \dots$ and A, B, C, \dots is used in this section only.

Theorem. Let $WXYZ$ be a cyclic quadrilateral with W, X, Y, Z lying on the circle with centre O . Let WOW' , XOX' be diameters of the circle and let the tangents at W', X' be constructed as in figure 2-10. Let WX, WY, WZ be produced to meet the tangent at W' at L, M, N respectively and let XY, XZ, XW be produced to meet the tangent at X' at P, Q, R respectively. Then

$$\frac{LM}{MN} \cdot \frac{PQ}{QR} = 1$$

Proof. Let the angles $\hat{\angle WOX}, \hat{\angle XOY}, \hat{\angle YOZ}, \hat{\angle ZOW}$ be $\alpha, \beta, \gamma, \delta$.

Then

$$\begin{aligned} LM &= LW' + W'M = d \tan \hat{\angle OWX} + d \tan \hat{\angle OYW} \\ &= d \cot \alpha/2 + d \cot [(\gamma+\delta)/2] \end{aligned}$$

and

$$\begin{aligned} MN &= W'N - W'M = d \tan \hat{\angle OWZ} - d \tan \hat{\angle OYW} \\ &= d \cot \delta/2 - d \cot [(\gamma+\delta)/2]. \end{aligned}$$

Hence

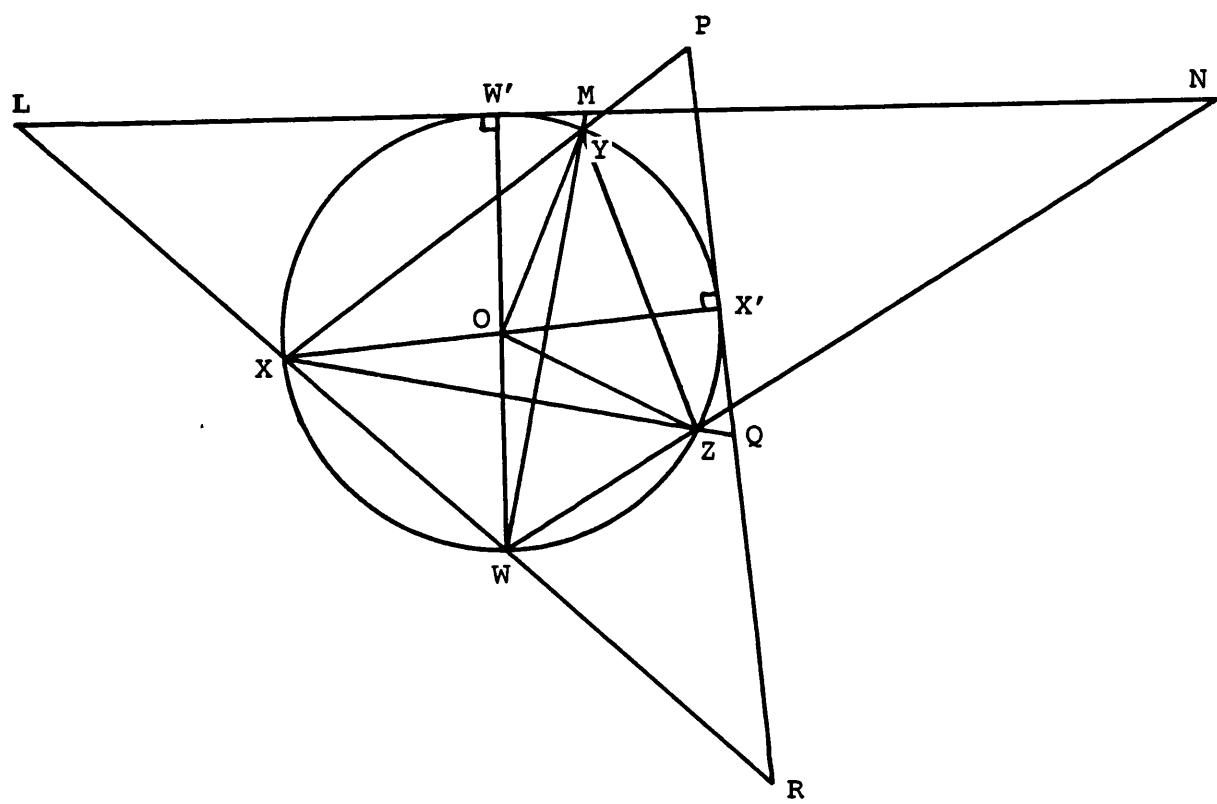


Figure 2-10. Cyclic quadrilateral $WXYZ$ inscribed in circle,
centre O .

$$\frac{LM}{MN} = \frac{\cot \alpha/2 + \cot [(\gamma+\delta)/2]}{\cot \delta/2 - \cot [(\gamma+\delta)/2]} . \quad (2-25)$$

Similarly

$$\frac{PQ}{QR} = \frac{\cot \beta/2 + \cot [(\delta+\alpha)/2]}{\cot \alpha/2 - \cot [(\delta+\alpha)/2]} . \quad (2-26)$$

Expanding $\cot [(\gamma+\delta)/2]$ and writing A for $\cot \alpha/2$, B for $\cot \beta/2$ etc., (2-25) becomes

$$\frac{LM}{MN} = \frac{A + (CD-1)/(C+D)}{D - (CD-1)/(C+D)} = \frac{AC + AD + CD - 1}{D^2 + 1} .$$

But $\alpha + \beta + \gamma + \delta = 2\pi$ so in (2-26), $\cot \beta/2$ becomes

$$\begin{aligned} \cot \left(\frac{\beta}{2} \right) &= \cot \left(\pi - \frac{\alpha+\gamma+\delta}{2} \right) = -\cot \left(\frac{\alpha+\gamma+\delta}{2} \right) \\ &= \frac{ACD - A - C - D}{CD + AD + AC - 1} \end{aligned}$$

and (2-26) may be re-written as

$$\begin{aligned} \frac{PQ}{QR} &= \frac{-\frac{ACD - A - C - D}{CD + AD + AC - 1} + \frac{DA - 1}{D + A}}{A - \frac{DA - 1}{D + A}} \\ &= \frac{\frac{D^2 + 1}{CD + AD + AC - 1}}{\frac{CD + AD + AC - 1}{D + A}} \\ &= \left(\frac{LM}{MN} \right)^{-1} \end{aligned}$$

Hence

$$\frac{LM}{MN} \cdot \frac{PQ}{QR} = 1$$

as required.

What this theorem means is that there is nothing special about choosing vertex W as the point, onto the diametrically opposite tangent of which, to project the three other vertices of the cyclic quadrilateral. If the points W, X, Y and Z are moved around the circumference of the circle but the positions of X, Y and Z are kept in the same positions relative to W by ensuring that the ratio $LM:MN$ remain constant then, as the ratio $PQ:QR$ also remains constant, the positions of W, Y and Z also remain in the same positions relative to vertex X . The theorem could trivially be extended to a cyclic polygon and from it the conclusion drawn that, in section 2.3, there is no loss of generality in selecting vertex $e^{i\theta_0}$ as the point from which to project the remaining polygonal vertices. The condition that $LM:MN$ remains constant is analogous to the transformation in equation (2-21).

2.5 Example showing $\underline{\omega}$ equivalent to $\underline{\theta}$

Before describing a method for solving equation (2-16), this section gives an example of how $\underline{\omega}$ is equivalent to $\underline{\theta}$. To do this, let P be one of the few polygons for which an analytic solution of equation (2-16) is known, a regular polygon. For such a case, arguments of symmetry [18] show that z_k values spread equally around the unit circle's boundary define θ_k correctly. If the case $n = 6$ is considered, then $\underline{\theta}$ defined by

$$\theta_0 = 0, \quad \theta_1 = \pi/3, \quad \theta_2 = 2\pi/3, \quad \theta_3 = \pi, \quad \theta_4 = 4\pi/3, \quad \theta_5 = 5\pi/3$$

is a solution of equation (2-16). The corresponding z_k points are shown in figure 2-11. As an illustration of how vector ω is an equally valid solution of (2-16), let $r_1 = 0.1$, $r_2 = -0.8$ and $r_3 = 1.0$. The formulae in (2-23) give values of θ_k' of

$$\theta_0' = 0.6283, \quad \theta_1' = 1.6755, \quad \theta_2' = 2.7227, \\ \theta_3' = 3.7699, \quad \theta_4' = 4.8171, \quad \theta_5' = 5.8643$$

and the corresponding $e^{i\theta_k'}$ points are shown in figure 2-12. In figure 2-13, the lines through $e^{i\theta_0'}$ and $e^{i\theta_k'}$ are drawn to meet the tangent to the circle opposite $e^{i\theta_0'}$ at ξ_k . These ξ_k have values

$$\xi_1 = -1.7321, \quad \xi_2 = -0.5774, \quad \xi_3 = 0, \quad \xi_4 = 0.5774, \quad \xi_5 = 1.7321.$$

The ξ_k are transformed by $\xi_k' = e^{r_3^2} \cdot \xi_k + r_3$ to give (figure 2-14)

$$\xi_1' = -3.0241, \quad \xi_2' = -1.5414, \quad \xi_3' = -0.8, \\ \xi_4' = -0.0586, \quad \xi_5' = 1.4241.$$

The ξ_k' points are projected back onto the circle's circumference in figure 2-15 to give the $e^{i\omega_k}$ points where

$$\omega_0 = 0.6283, \quad \omega_1 = 1.2670, \quad \omega_2 = 1.7793, \\ \omega_3 = 2.4204, \quad \omega_4 = 3.6528, \quad \omega_5 = 5.6871.$$

The $e^{i\omega_k}$ points are shown in figure 2-16.

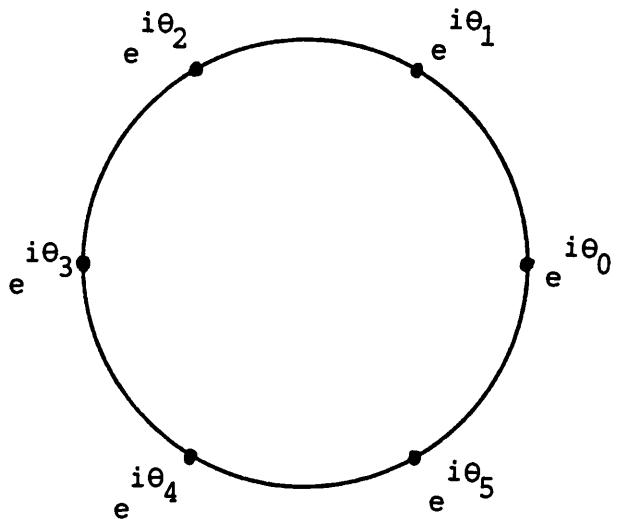


Figure 2-11. Points $z_k = e^{i\theta_k}$ equally spaced around circumference of circle.

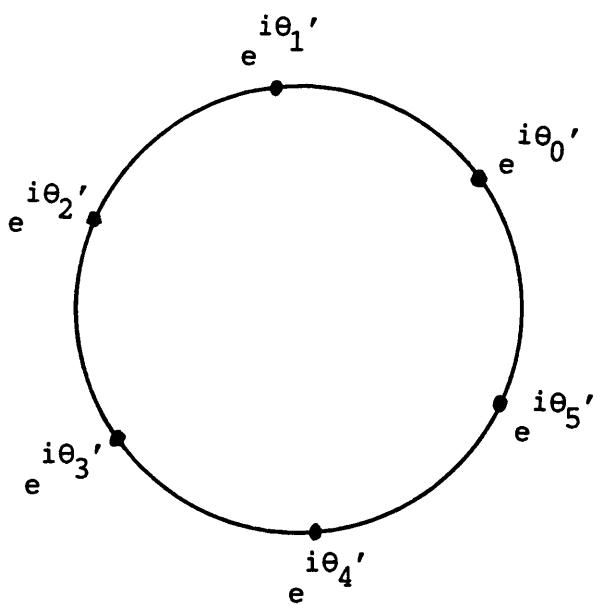


Figure 2-12. Points $e^{i\theta'_k}$ after receiving a rotation of $2\pi r_1$.

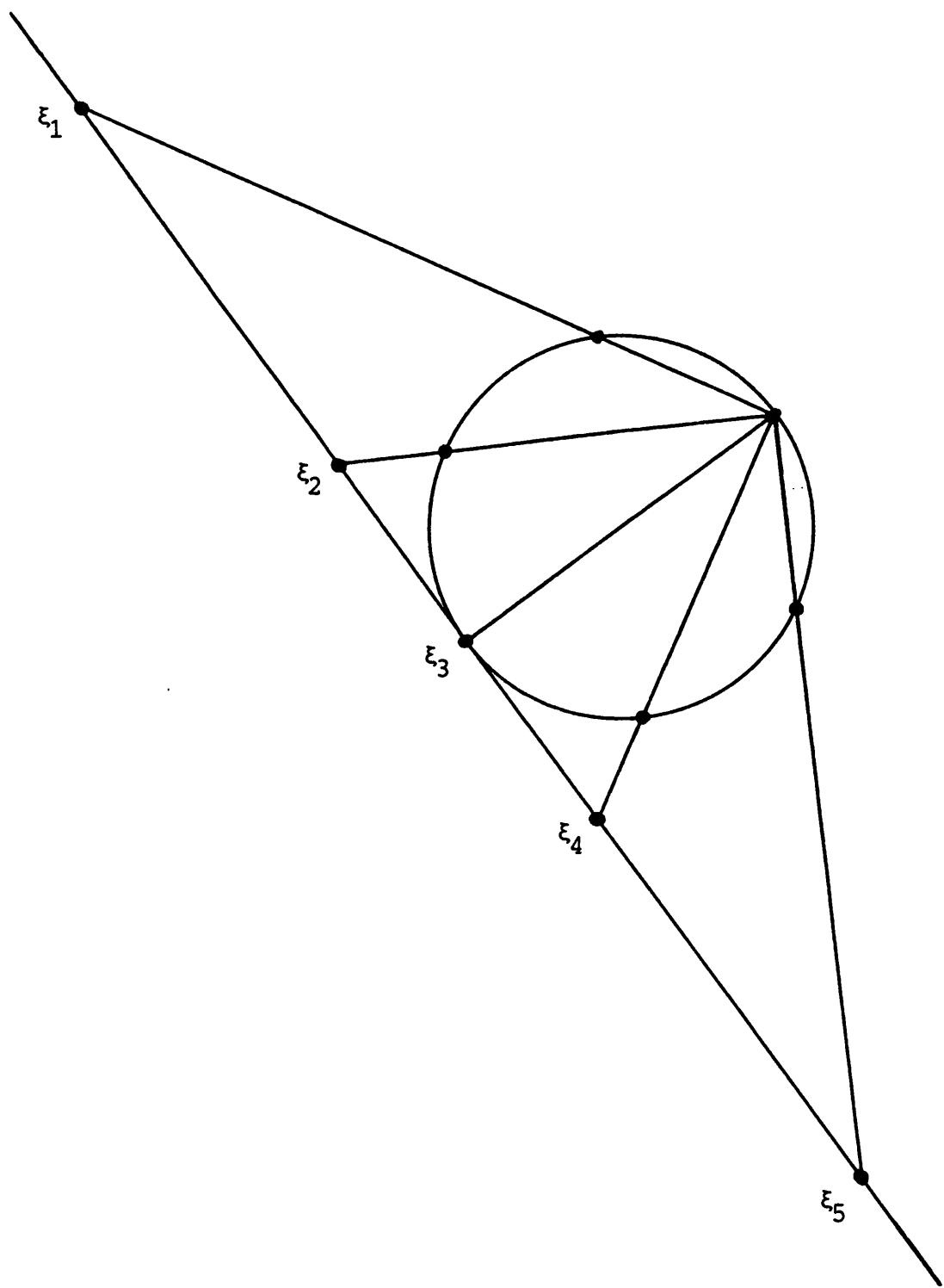


Figure 2-13. $e^{i\theta_k'}$ projected onto tangent opposite $e^{i\theta_0'}$ to define ξ_k .

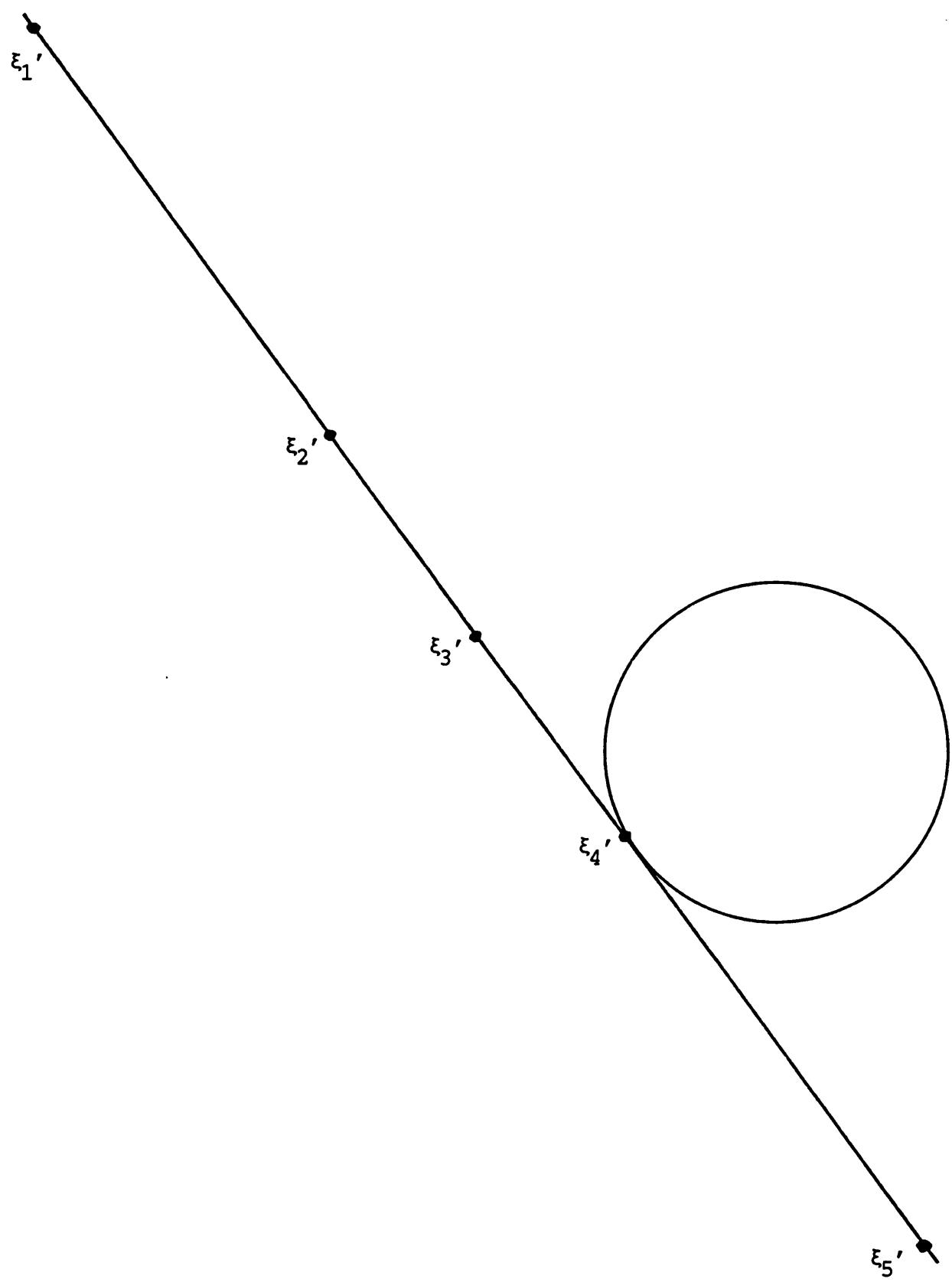


Figure 2-14. ξ_k transformed to define ξ'_k .

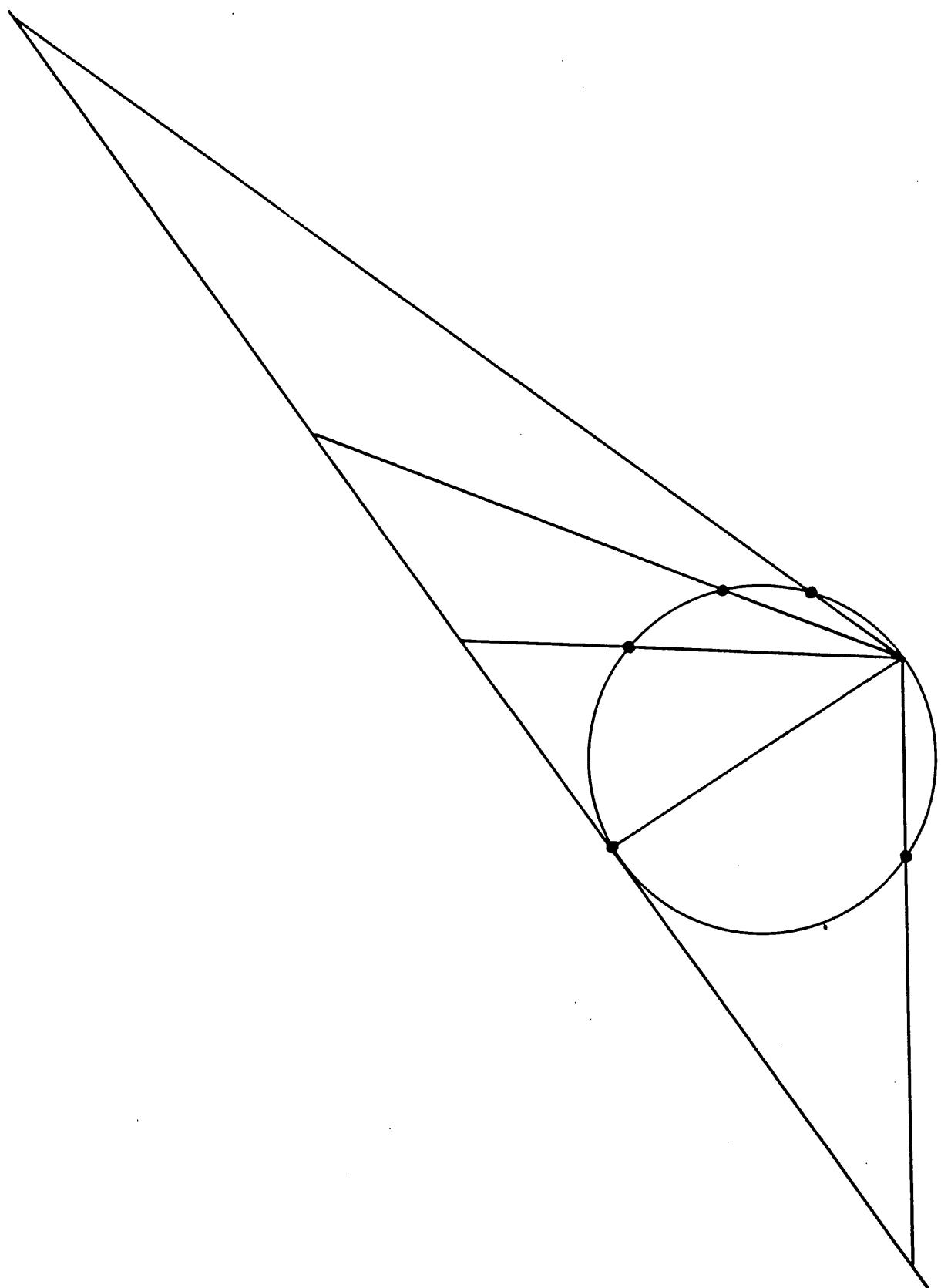


Figure 2-15. E_k' projected back onto circumference of circle.

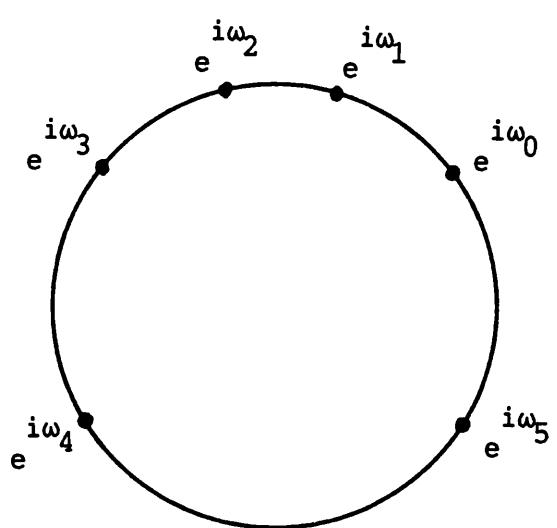


Figure 2-16. $e^{i\omega_k}$ defined at intersection points.

CHAPTER 3

Numerical methods for functions of a complex variable

3.1 Equation to be solved

It has been shown in an earlier section that the equation to be solved, (2-16), does not have a unique solution $\underline{\theta}$: there are three degrees of freedom available in how $\underline{\theta}$ may be selected. In order to solve (2-16) uniquely, three degrees of freedom must be eliminated. One way in which this could be done would be to fix three of the θ_k values, say θ_0 , θ_{n-2} and θ_{n-1} throughout the solution process and solve for θ_1 , θ_2 , ..., θ_{n-3} . This is almost the path followed here. Not being able to find $\underline{\theta}$ uniquely may at first appear to be a nuisance. However being able to vary $\underline{\theta}$ during the solution process aids considerably the efficiency of the iterative procedure. Being able to vary the solution $\underline{\theta}$ is vitally important in the way the Schwarz-Christoffel maps will be used in this study. Why the efficiency of the procedure is helped by adjusting $\underline{\theta}$ by relations (2-23) is discussed in the next section.

3.2 Evaluation of the integrals

In any numerical procedure to solve equation (2-16), the expression which defines θ_k in expression (2-14),

$$v_k = \int_0^{z_k} \frac{dz}{\prod_{l=0}^{n-1} \left(1 - \frac{z}{z_l}\right)^{\delta_l}}$$

has to be evaluated many times. If $\delta_k > 0$, this integral is singular at $z = z_k$ but since $\delta_k < 1$, (inequality (2-2)), v_k is always well defined.

The integral in (2-14) could be evaluated numerically immediately by using a quadrature rule which does not sample the end-points of the interval of integration. Gaussian quadrature is one such method. However a far more computationally efficient method to evaluate v_k is to remove the singularity of the integrand. This may be done by making a change of variable in (2-14)

$$(1 - z/z_k)^{1-\delta_k} = u$$

so that v_k becomes

$$v_k = \int_0^1 \frac{z_k \ du}{(1 - \delta_k) \prod_{\substack{l=1 \\ l \neq k}}^n \left(1 - \frac{z_k}{z_l} \left[1 - u^{1/(1-\delta_k)} \right] \right)^{\delta_l}} . \quad (3-1)$$

However it is not only the singularity in (2-14) which makes it expensive to evaluate. It is often the case that the z_k points are very close together - both at and on the way to the solution. Suppose, for example, that $\delta_k > 0$ and $\delta_{k+1} > 0$ and that z_k and z_{k+1} are very close. The singularity at $z = z_k$ may be eliminated but at $u = 0$ in (3-1), the integrand is very close to another singularity. Any quadrature method samples the integrand at points along the line of integration and the integrand will vary hugely at $u \sim 0$. In order for the integral to be evaluated with accuracy, many samples will have to be taken. It is to make this integral evaluation as efficient as possible that use is made of the transformation developed in an earlier section. There, it was shown that if ω is defined by (2-23), then equation

(2-24) holds. By suitable choice of r_2 and r_3 in (2-23), the θ_k values upon which v_k depends may be distributed around the unit circle's circumference so that they are as evenly spread as possible. r_1 is set arbitrarily to zero as, being merely a rotation, it makes no difference to the relative spacing of the ω_k points. However r_2 and r_3 are selected, y remains unaltered as does the stage reached in the iterative process.

The way in which the ω_k points are spread may be defined quantitatively as follows - let S be defined by

$$S = \ln \prod_{k=1}^n (\omega_k - \omega_{k-1}) \quad (3-2)$$

For any given θ_k , S is defined as a function of r_2 and r_3 . Admittedly this function is defined rather arbitrarily but it has been selected in this way as it is continuously differentiable and can be maximised relatively easily. The ω_k points are optimally spread when S attains its maximum. The logarithm merely ensures that S remains of about the same order of magnitude as otherwise scaling becomes a problem. Spreading the ω_k points around the circumference as evenly as possible also has the advantage that if P has any symmetry, this is reflected in the symmetrical positioning of the ω_k points. This is a consequence of the symmetrical definition of the function S of ω_k in (3-2)

3.3 Numerical evaluation of the Schwarz-Christoffel transformation

Nearly all the constituent parts of the process of solving equation (2-16) have been assembled. The solution vector $\underline{\theta}$ satisfies the inequality (2-10) and from the computational point of view, it is more

convenient to solve instead an unconstrained equation. It is for this reason that one final transformation is made,

$$y_k = \ln \frac{\theta_k - \theta_{k-1}}{\theta_{k+1} - \theta_k} \quad (3-3)$$

for $k = 1, 2, \dots, n-3$. Transformation (3-3) is due to Trefethen [19]. It supposes that θ_0 , θ_{n-2} and θ_{n-1} are held fixed for the duration of one iteration. Let vector \underline{y} be defined by

$$\underline{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{n-3} \end{bmatrix}.$$

The complete numerical scheme can now be given:

1. Given w_0, w_1, \dots, w_{n-1} , calculate δ_k by (2-1) and β by (2-11).
2. Select starting value of $\underline{\theta}$ by $\theta_0 = 0$, $\theta_k = \theta_0 + 2\pi k/n$ for $k = 1, 2, \dots, n$, i.e. z_k points evenly spaced around the unit circle's circumference.
3. Calculate y_k from $\underline{\theta}$ by (3-3).
4. Calculate v_k by (2-14).
5. Calculate \underline{v} by (2-15).
6. If $||\underline{v} - \beta||$ is small, stop.
7. Calculate $\partial v_j / \partial y_k$.
8. Solve the system of equations

$$\sum_{k=1}^{n-3} \frac{\partial v_j}{\partial y_k} \delta y_k = \beta_j - v_j$$

9. Increment y_k by δy_k .
10. Calculate new $\theta_1, \theta_2, \dots, \theta_{n-3}$ from y_k .
11. Select best values of r_2 and r_3 which define ω spread as evenly as possible around the circumference of the unit circle - i.e. those r_2 and r_3 which maximise S in (3-2). Reset θ to be this ω .
12. Repeat from step 3.

Details. Step 10 is not obvious because of the way in which y_k and θ_k are interlinked. In fact, θ_k can be recovered from y_k by

$$\theta_k = \theta_0 + (\theta_{n-2} - \theta_0) \frac{\left[\begin{array}{c} 1 \\ 1 + \sum_{l=1}^{k-1} e^{-\sum_{j=1}^l y_j} \end{array} \right]}{\left[\begin{array}{c} 1 \\ 1 + \sum_{l=1}^{n-3} e^{-\sum_{j=1}^l y_j} \end{array} \right]} \quad (3-4)$$

This equation holds for $k = 2, 3, \dots, n-2$. It holds for $k = 1$ as well if the convention

$$\sum_{l=1}^0 = 0$$

is followed.

Step 2 is equivalent to setting $\theta_0 = 0$, $\theta_{n-1} = 2\pi - 2\pi/n$, $\theta_{n-2} = 2\pi - 4\pi/n$ and $y_1 = y_2 = \dots = y_{n-3} = 0$.

All the integration in this study (e.g. at step 4) is done by 16 point Gaussian quadrature. The routine used is self-checking to a given tolerance. Along an interval, it compares the value obtained using a 16 point rule with that obtained using an 8 point rule. If agreement is not

satisfactory, the interval is halved and the two rules' values compared in each half-interval. This halving is repeated as necessary.

Only approximate values for the Jacobian matrix in step 7 are calculated. This is done by differencing: to estimate $\partial r_j / \partial y_k$, increment y_k by a small amount, say ϵ ; calculate the altered $\theta_1, \theta_2, \dots, \theta_{n-3}$; evaluate r_j and hence $r_j^+ (= r_j^+)$ at this amended θ . Approximate $\partial r_j / \partial y_k$ by $(r_j^+ - r_j) / \epsilon$.

In step 8, a library routine employing Crout's factorisation method is used.

The 2-norm

$$||\underline{y} - \underline{\beta}|| = \left[\sum_{j=1}^{n-3} (r_j - \beta_j)^2 \right]^{1/2}$$

is used at step 6.

Step 9 is written in its simplest form. The aim at each iteration (one iteration comprises steps 3 to 11) is to get nearer to the solution, that is to reduce $||\underline{y} - \underline{\beta}||$. Occasionally the straightforward Newton increment δy does not do this. This has been noted to happen in difficult examples where the θ_k points are exceptionally closely grouped or where the Jacobian matrix is nearly singular. In this case, a limited linear search along the direction of the increment vector, δy , is performed: only a fraction of the full Newton increment, $\mu \cdot \delta y$ is added to y . If \underline{y}^0 is the current value of the \underline{y} function and \underline{y}^+ is its value at $\underline{y} + \delta y$, and if it is assumed that \underline{y} varies linearly between \underline{y}^0 and \underline{y}^+ , then μ is chosen to be that factor of δy which minimises $||\underline{y} - \underline{\beta}||$. The value of μ which does this is

$$\mu = \frac{\sum_{j=1}^{n-3} (\beta_j - \gamma_j^0)(\gamma_j^+ - \gamma_j^0)}{\sum_{j=1}^{n-3} (\gamma_j^+ - \gamma_j^0)^2}.$$

Large values of μ are not used in order to prevent oscillation in the solution process. If μ is computed to exceed unity, the value 0.75 is used instead; if computed to be less than -1, the value -1 is used.

3.4 Calculation of a_2 and b_2

Finally in order to define the function f_2 completely, the two constants a_2 and b_2 in mapping (2-6) need to be calculated. If v_k is defined by formula (2-14) as a function of the solution vector $\underline{\theta}$, then a_2 and b_2 are those complex constants which satisfy

$$w_k = a_2 v_k + b_2 \quad (3-5)$$

for $k = 0, 1, \dots, n-1$. If equation (2-16) could be solved exactly and if all the integrals could be evaluated exactly, it would be enough to ensure that just two distinct v_k points are mapped onto the corresponding w_k points by (3-5). The remaining $n-2$ equations would be automatically satisfied. Since the equation is solved and the integrals are evaluated only by numerical methods, instead a_2 and b_2 are selected to be those that do the best job: they are chosen to minimise the penalty function

$$\sum_{k=0}^{n-1} |a_2 v_k + b_2 - w_k|^2. \quad (3-6)$$

By posing this as a problem in four variables, the real and imaginary parts of a_2 and b_2 , this reduces to solving a set of four linear simultaneous equations. The solution is as follows:

$$a_2 = \frac{n \sum_{k=0}^{n-1} w_k v_k^* - \sum_{k=0}^{n-1} v_k^* \sum_{k=0}^{n-1} w_k}{n \sum_{k=0}^{n-1} |v_k|^2 - |\sum_{k=0}^{n-1} v_k|^2},$$

$$b_2 = \frac{\sum_{k=0}^{n-1} |v_k|^2 \sum_{k=0}^{n-1} w_k - \sum_{k=0}^{n-1} v_k \sum_{k=0}^{n-1} w_k v_k^*}{n \sum_{k=0}^{n-1} |v_k|^2 - |\sum_{k=0}^{n-1} v_k|^2}. \quad (3-7)$$

* denotes the complex conjugate. The derivation of these expressions appears in appendix 3.

3.5 Examples of complete Schwarz-Christoffel transformations

The first example in this section has been selected to demonstrate just how closely the θ_k points may be grouped.

Here, P is chosen to be a hexagon with all interior angles equal to $2\pi/3$. The opposite sides are equal and have values 1, 1 and x where x is allowed to vary. The Schwarz-Christoffel transformation was found for P for a range of values of $x = 0, 1, 2, \dots, 10$. When x is zero, P degenerates into a rhombus. If the vertices of P are numbered as shown in figure 3-1 with the lengths

$P_0P_1 = P_2P_3 = P_3P_4 = P_5P_0 = 1$, $P_1P_2 = P_4P_5 = x$, then the symmetry of P dictates that, once the pre-vertices defined by the solution vector $\underline{\theta}$

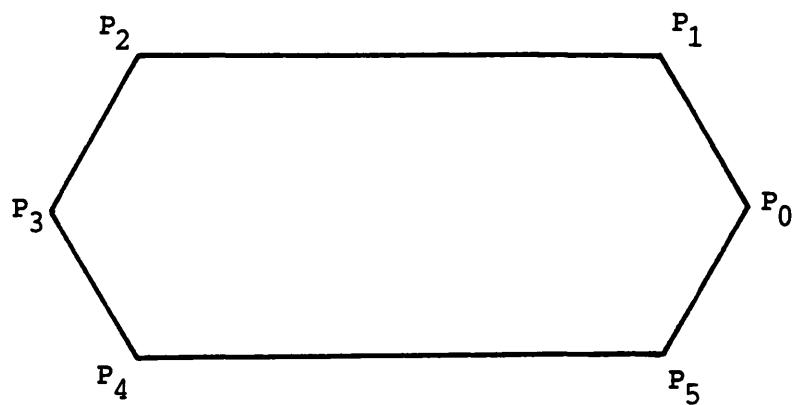


Figure 3-1. Example polygon P with internal angles all $2\pi/3$.

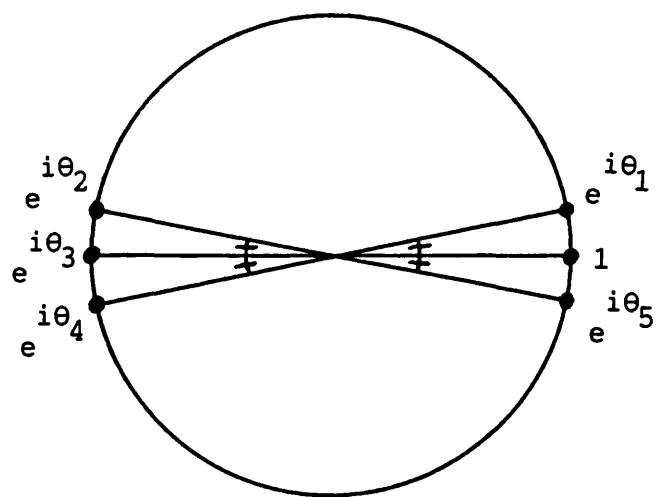


Figure 3-2. Pre-vertices of mapping to polygon in previous figure.

have been optimally spread around the circle's circumference as earlier described,

$$\theta_3 = \pi \quad \text{and} \quad \theta_1 = \pi - \theta_2 = \theta_4 - \pi = 2\pi - \theta_5$$

as shown in figure 3-2. In addition, for the special cases of $x = 0$ and 1 , $\theta_1 = \pi/2$ and $\pi/3$ respectively. The calculated values of θ_1 are shown in the following table.

x	θ_1
0	1.570
1	1.047
2	0.4631
3	0.1914
4	0.07710
5	0.03117
6	0.01269
7	0.005057
8	0.002042
9	0.0008325
10	0.0003410

Table 1

For $x = 5$, the two groups of pre-vertices span just 0.020 of the circumference; when $x = 10$, this figure has decreased to 0.00022. It should be borne in mind that the pre-vertices are this close even after they have been optimally spread around the circumference by the algorithm described in section 3.2.

Finally in this section, an example of a complete Schwarz-Christoffel transformation is given which possesses none of the symmetry of the above examples. It is the mapping between the unit circle and the interior of the septagon, most of which is drawn in figure 2-5. n has the value 7 so that, for example, w_{n-3} in the figure is w_4 in the example. The 7 vertices are given by

$$w_0 = 0.60 + 2.15 i, \quad w_1 = 0.12 + 1.82 i, \quad w_2 = 0.25 + 0.40 i,$$

$$w_3 = 1.84 - 0.40 i, \quad w_4 = 3.11 + 0.73 i, \quad w_5 = 2.72 + 1.95 i,$$

$$w_6 = 1.26 + 1.85 i.$$

The parameters which define the mapping in (2-6) are given below. δ_k is defined in (2-1) and θ_k in (2-10).

k	θ_k	δ_k
0	0.0000	0.3275
1	0.2254	0.3373
2	1.3930	0.3226
3	2.7323	0.3798
4	3.3760	0.3670
5	3.7004	0.4233
6	5.3480	-0.1576

The two constants are

$$a_2 = -0.8826 + 0.6276 i, \quad b_2 = 1.2489 + 1.1453 i.$$

Just two iterations were needed to find these θ_k values: $||y - \beta||$ was equal to 7.83×10^{-3} at the algorithm's starting point, $y = 0$. It had dropped to 1.05×10^{-3} after the first iteration and to a sufficiently small value of 5.07×10^{-7} after the second iteration. μ had a value of unity for both iterations.

3.6 Inversion of the Schwarz-Christoffel transformation

The Schwarz-Christoffel mapping in (2-6) is one-one and onto. These are precisely the conditions needed to ensure that f_2 can be inverted and that f_2^{-1} is well defined. In this section, details are given about the numerical inversion of the Schwarz-Christoffel formula in equation (2-6), that is, given a point $w = w_x$ in the interior or on the

boundary of polygon P , how to find the point z_x which maps onto it. The general scheme for doing this, but not the details which follow, is the same as that given by Trefethen [19]. In what follows, it will be assumed that the w_x point is not a vertex of polygon P ; if it is a vertex, then the point which maps onto it by (2-6) is already known.

Presented here are two distinct methods by which the value $z = z_x$ can be found such that

$$w_x = f_2(z). \quad (3-8)$$

(2-6) can be thought of as a non-linear equation in one variable, z , and a Newton method used to solve it. This method depends on having an approximation to the solution and systematically improving it: if z is an approximation to the solution, then $z + \delta z$ is a better approximation, where

$$\delta z = \frac{w_x - f_2(z)}{f_2'(z)} = \frac{w_x - f_2(z)}{a_2 \prod_{k=0}^{n-1} \left(1 - \frac{z}{z_k}\right)^{-\delta_k}}. \quad (3-9)$$

It is a typical feature of Newton's method that it converges very quickly once near to the solution. On the other hand, oscillation and divergence can occur if the current point is not near to the solution. The value δz given by (3-9) is well defined as long as z never attains the value of one of the z_k points. This is true at the solution and it will be assumed that it is true also en route to the solution.

An alternative way of solving (3-8) for z is to re-write the equation as follows:

$$\frac{dz}{dw} = \left(\frac{dw}{dz}\right)^{-1} = \frac{\prod_{k=0}^{n-1} \left(1 - \frac{z}{z_k}\right)^{\delta_k}}{a_2}. \quad (3-10)$$

Written like this, it can be thought of as an ordinary differential equation in one variable and solved as an initial value problem. A starting point, say z_c , and its known image under f_2 , w_c , must be given. Either a previously obtained $\{z, w\}$ pair or $\{0, b_2\}$ may be used as this $\{z_c, w_c\}$ initial value. A vertex whose internal angle is not π radians and its pre-image (say w_{k_e} , z_{k_e}) should not be used as an initial point. Although the z point and its image are accurately known, the value of the derivative dz/dw at $z = z_{k_e}$ is either zero if $\delta_{k_e} > 0$ or infinite if $\delta_{k_e} < 0$. Either case is useless in a numerical solution.

As an illustration of how the differential equation in (3-10) may be solved numerically, the simplest of all such numerical methods, that of Euler, is used: starting from $\{z_c, w_c\}$, an approximation of the value which maps onto $w + \delta w$ is

$$z_c + \delta w \frac{dz}{dw} \Big|_{z=z_c} = z_c + \delta w \frac{\prod_{k=0}^{n-1} \left(1 - \frac{z_c}{z_k}\right)^{\delta_k}}{a_2} .$$

This iterative procedure is continued until the w_x point is reached. The path of integration along which successive $w + \delta w$ points lie must be selected to be completely within or on the boundary of P , for otherwise z would go outside the range for which dz/dw has a meaningful value. If P is a convex polygon, the straight line segment joining w_c and w_x is acceptable. If P is not convex, two or more straight line segments connecting w_c and w_x may be used.

If the differential equation (3-10) with initial conditions $z = z_c$, $w = w_c$ could be solved analytically between $w = w_c$ and w_x , the value z_x satisfying $w_x = f_2(z_x)$ would be obtained immediately. However no numerical method can be expected to remain error free, even a

more sophisticated one than that of Euler. Again this is especially true for w_x near to a vertex of P . It has been decided in this study to use both available methods, using the obtained value of z from the second method as an initial estimate for the first. In this way, use is made of the better features of each method. The good point about the Newton method is its rapid convergence once near to the solution. However it is an expensive method with one evaluation of a complex integral every iteration. This involves sampling the integrand of mapping (2-6) at several positions along the length of the interval 0 to z . Conversely, the Euler method is cheap as each increment of δw requires just one evaluation of dz/dw , the reciprocal of the integrand of (2-6). However, its convergence cannot be guaranteed.

3.7 Example of inversion of the conformal mapping

As an example of how the z points which map onto given w points may be found, let P be the polygon with all interior angles $2\pi/3$ and sides, in anti-clockwise order, 2, 6, 3, 4, 4 and 5. The θ_k values which define the Schwarz-Christoffel transformation are

$$\begin{aligned}\theta_0 &= 0, & \theta_1 &= 0.1978, & \theta_2 &= 2.1272, \\ \theta_3 &= 2.7142, & \theta_4 &= 3.7264, & \theta_5 &= 4.9227.\end{aligned}$$

These are unique if $r_1 = 0$ and r_2 and r_3 are selected to maximise S in (3-2). This P is shown in figure 3-3. A mesh of interior lines has been drawn inside P so that it is filled with equilateral triangles of unit side length. Let M denote the set of all these mesh lines and the polygon perimeter. If the mapping is $w = f_2(z)$, then figure 3-4 shows the curves which map onto M in P , that is

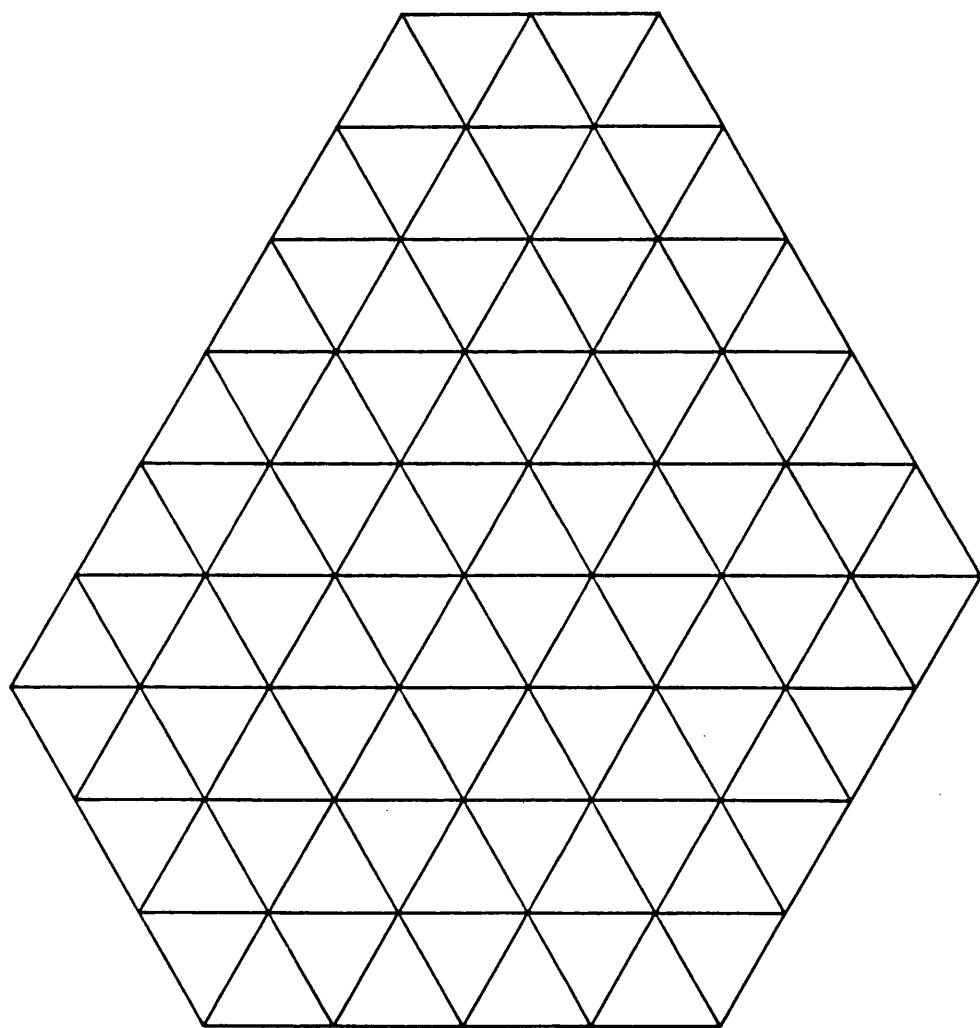


Figure 3-3. Example polygon P with internal grid of lines
intersecting at angles of $\pi/3$.

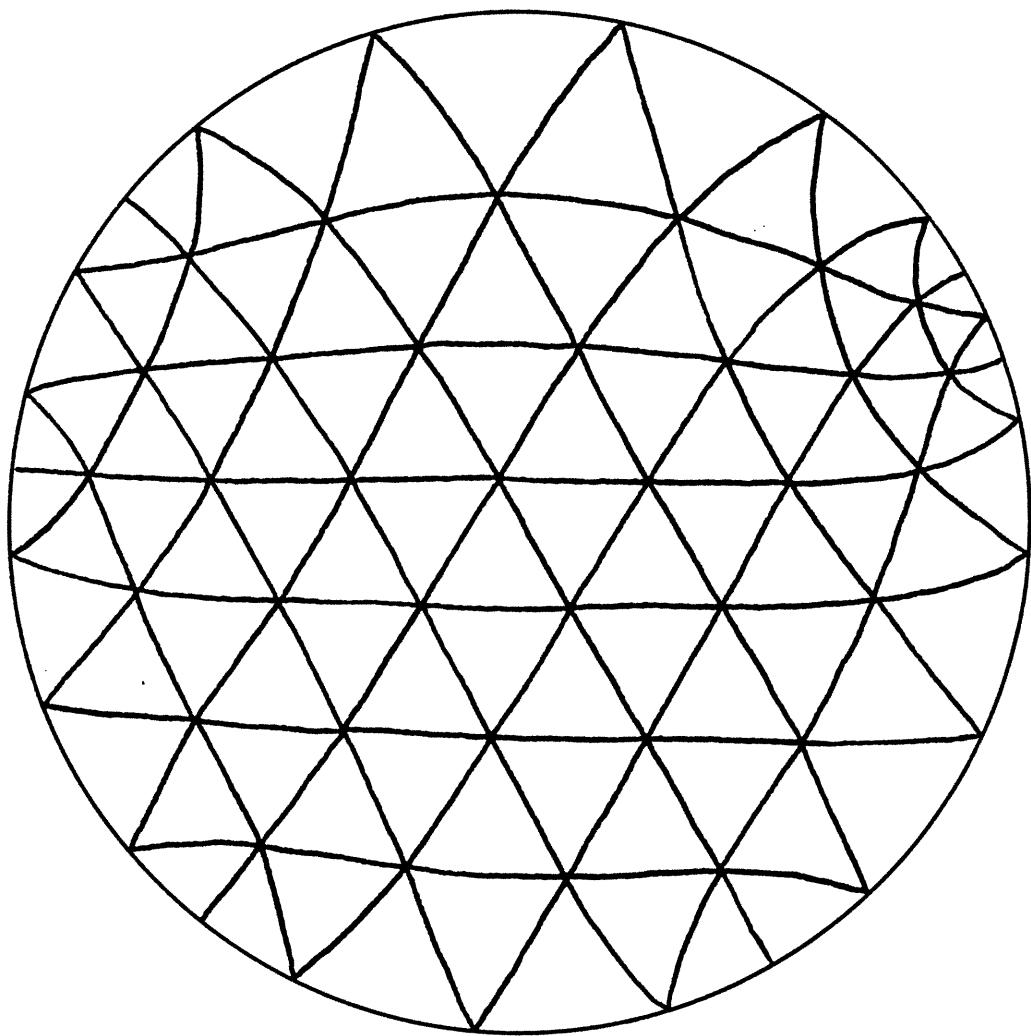


Figure 3-4. Curves which map onto defined grid in P under f_2 .

$f_2^{-1}(M)$. This figure was drawn by finding the inverses of a series of points along the mesh lines in P . It illustrates the nature of a typical Schwarz-Christoffel transformation: it is angle-preserving at all but a finite number of points. The mesh lines in P all intersect at angles of $2\pi/3$. The inverses of the mesh lines, in the unit circle in the z -plane, which in general are not straight lines, intersect at angles of $2\pi/3$ except at the pre-vertices $e^{i\theta_k}$: it is at these points, and these points only, where the conformality of the mapping breaks down.

3.8 Conformal mapping between two polygons

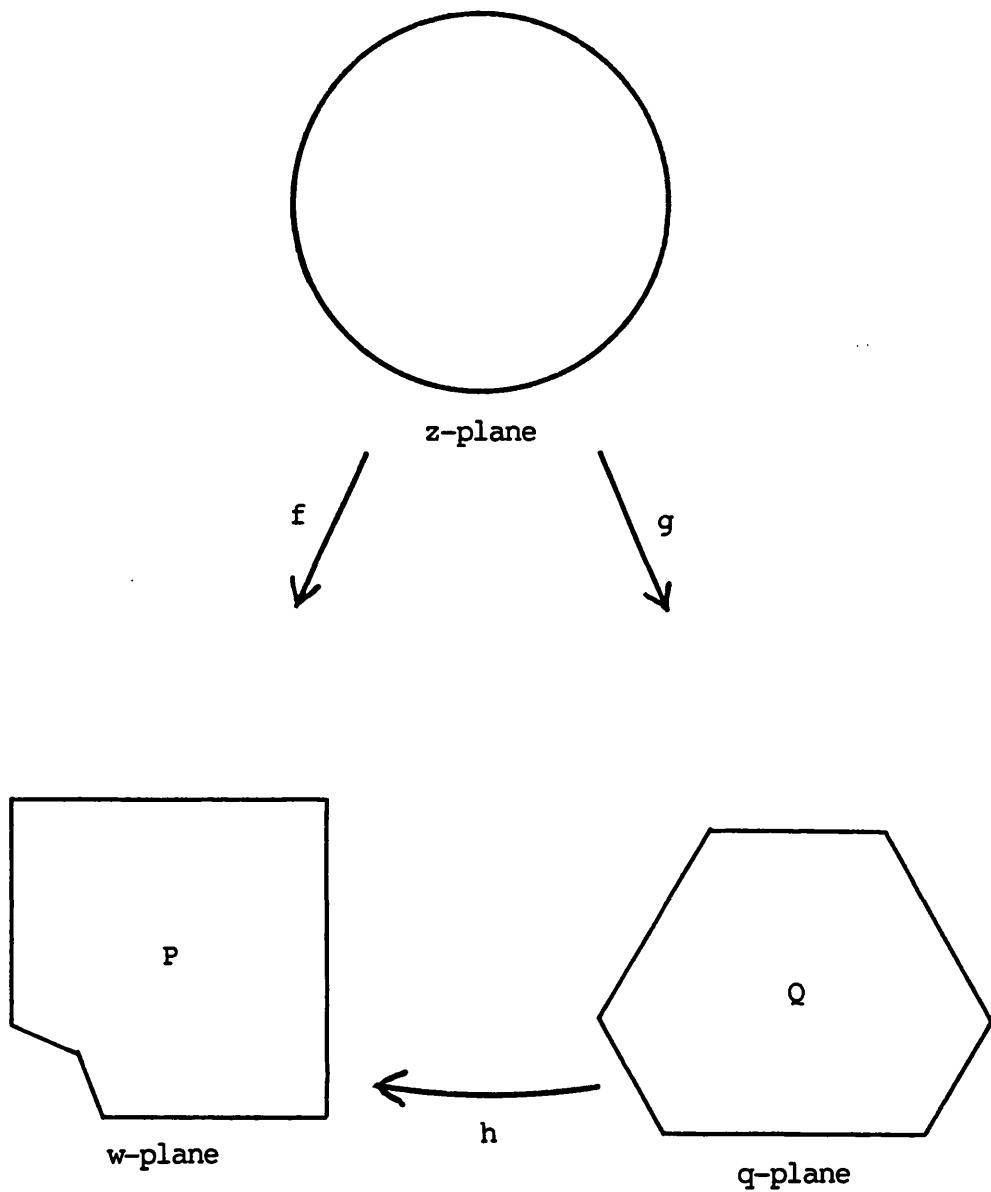
Earlier sections have shown how it is possible to evaluate the parameters which define the Schwarz-Christoffel transformation between the unit disc and the interior of a general finite polygon P . It has also been shown how to invert such a mapping by numerical methods. It will now be shown how to use two such mappings to create a composite mapping between two general polygons P and Q . Let f and g denote the two Schwarz-Christoffel transformations between $|z| \leq 1$ and the interior and boundary of the two polygons P and Q . Let Q be situated in the q -plane. f is conformal everywhere except at the pre-vertices of P . Similarly g is conformal everywhere except at the pre-vertices of Q . Then a composite mapping

$$w = h(q) = f(g^{-1}(q)) = (f \circ g^{-1})(q)$$

can be created. The angle-preserving nature of g is clearly one which applies to g^{-1} also. g^{-1} is conformal everywhere except at the vertices of Q . The composite mapping h is therefore conformal

everywhere except at the vertices of Q and at the points on the boundary of Q which map onto the pre-vertices of P . Figure 3-5 gives a pictorial representation of the three mappings, f , g and h , and figure 3-6 shows an example of such a mapping from the same hexagon used in an earlier example mapped onto a rectangle. The places where the conformality of the mapping breaks down are the vertices of the hexagon (these map onto the points marked * in the figure) together with the points on the hexagon's boundary which map onto the vertices of the rectangle.

In the next chapter, it will be described how such a composite mapping h can be used to generate a finite element mesh in a two-dimensional region.



$$w = h(q) = f(g^{-1}(q)) = (f \circ g^{-1})(q)$$

Figure 3-5. Pictorial representation of composite mapping between two polygons.

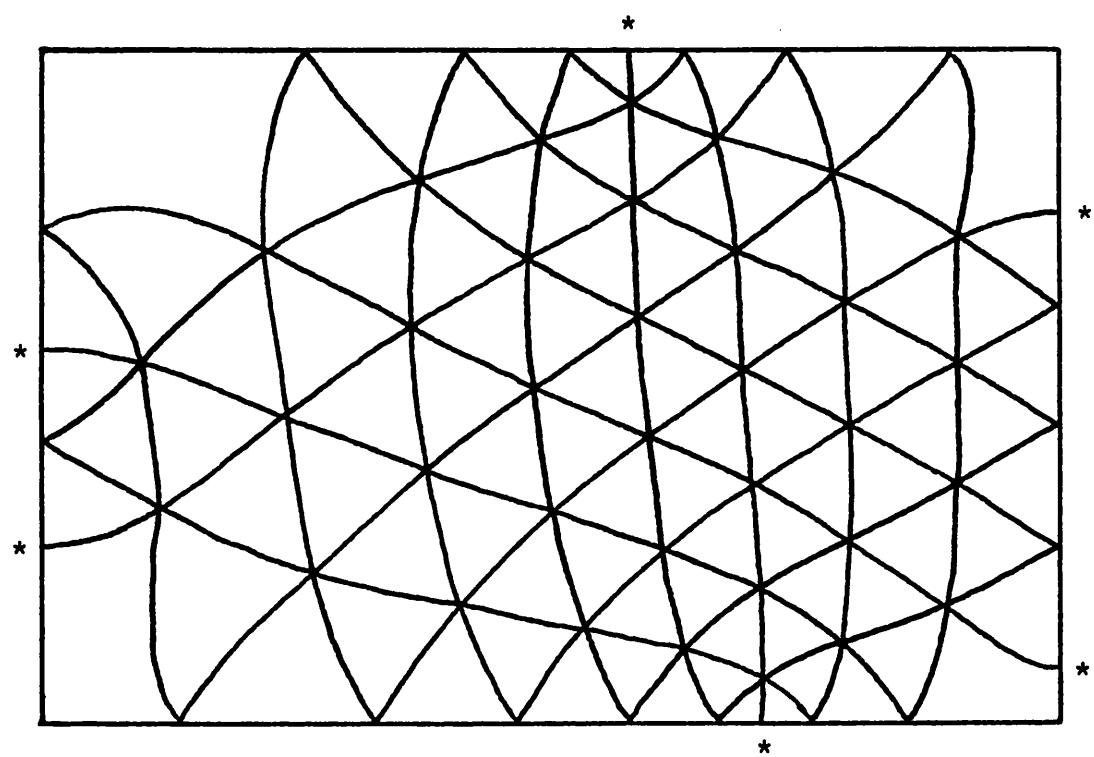


Figure 3-6. Regular grid in figure 3-4 mapped onto a rectangle by a composite conformal mapping.

CHAPTER 4

Finite element mesh generation

4.1 Desirable properties of a finite element mesh

In this section, the desirable properties of a finite element mesh are stated and it is shown how the described generation scheme attempts to define a mesh having these properties.

Best element shape. Meshes for two-dimensional regions (and it is two-dimensional meshes only which will be considered here) are usually composed of quadrilateral or triangular elements. All authorities agree that it is the symmetrical element which best models the variation of the quantity (be it displacement, temperature, velocity, etc.) over the finite element mesh: the best shaped quadrilateral element is a square and the best triangular element is equilateral. For a general region it is impossible to define a mesh composed entirely of symmetrically shaped elements but a mesh containing elements as near to the ideal as possible is the most desirable.

It is important that any change in element size should be gradual over the mesh. (For definiteness, it will be assumed that the problem being tackled is one of stress analysis where the quantity varying over the mesh area is displacement. From these calculated displacements, components of stress may be calculated. Clearly the generated mesh could be used to tackle any other problem in finite element analysis: the described method has no such restrictions.) Areas of the mesh where stresses are expected to be changing rapidly need to be modelled by many small elements but economy of computer time and program size prohibit meshes from having equally small elements everywhere. Areas where

stresses are expected to be spatially constant may be modelled by larger elements. Since the variation of displacement over the mesh is invariably continuous, so the finite elements which model a structure too should vary gradually in size.

Bandwidth. Once the mesh has been defined, the nodes, that is the vertices of the elements, must be assigned numbers. The finite element method then consists of assembling a square matrix, the matrix of the system of linear equations which must be solved [20]. The matrix is banded, that is all elements of the matrix beyond a certain distance from the main diagonal are zero. By making the width of this band as small as possible, the amount of computer time and storage space needed is also minimised and the equation solving process is made as efficient as possible. The bandwidth of the matrix is a function of the maximum of d_j where j ranges over the numbers of all the elements of the mesh. d_j is the maximum difference of the numbers of those nodes which define element j . Optimally numbering the nodes of a mesh is, in general, not straightforward except for regular meshes. The term "bandwidth of the mesh" will be used for this maximum d_j . Although not strictly a property of a finite element mesh, the size of the bandwidth of the matrix of the resulting system of linear equations is closely connected to mesh generation. If an optimal numbering system can be an inherent part of mesh generation, this is an obvious advantage. Alternative solution schemes such as the frontal solution are available. If such a scheme is used, the numbering of the elements becomes an important consideration. Considering an efficient numbering scheme for the elements could easily be done but, for simplicity, it will be assumed here that the banded nodal matrix method is the one being used.

4.2 Aims of this method

An automatic method of generating a mesh in a simply-connected region will now be described which attempts to satisfy the following conditions.

- (a) Regions of greatest interest where the stresses are expected to vary most rapidly are modelled by the smallest elements.
- (b) The change of size of the elements is gradual throughout the region except possibly in places where the stresses are expected to be spatially constant.
- (c) The shape of the majority of elements is good, especially in regions of greatest interest.
- (d) The nodes are numbered in an efficient way to make the bandwidth of the mesh as small as possible for a given number of elements.

The method described attempts to accomplish these aims by generating a mesh in any region which may be represented by a polygon. A second polygon is chosen in which a regular mesh of any density, composed entirely of ideally shaped elements, can easily be generated. The composite conformal transformation is found which maps this polygon onto the modelling polygon. There are three degrees of freedom in the choice of this transformation which determine that the smallest elements be placed where highest stress gradients are expected.

It will be shown that the angle-preserving nature of the conformal mappings is a useful tool in determining the shape of the elements in the generated mesh.

4.3 Mesh generation in a simply-connected two-dimensional region

In earlier sections, it has been shown how to find a mapping from a polygon Q to a polygon P . The mapping is conformal everywhere except at the vertices of Q and at those points on the boundary of Q which map onto the vertices of P . Let R be a two-dimensional simply-connected region in which a finite element mesh is to be drawn. It is proposed that P be so defined to model the region R . If the boundary of R is composed entirely of straight lines this is readily achieved. If part of the boundary of R is a curved arc, this may be represented by a suitable number of straight line segments. As R is to be represented by a polygon, it is for this reason that the generation scheme described here is restricted to simply-connected regions. It is conceivable that the described method could be applied to regions which are not simply-connected by dividing such a region into more than one modelling polygon and finding transformations to each component part. However this is beyond the scope of the research described here.

Polygon Q is now defined. Let Q be a polygon in which a good finite element mesh may easily be generated automatically, that is one composed entirely of ideally shaped elements, elements of the same type as that which it is desired should define the mesh in R . For example, if the mesh in P is to be composed of quadrilateral elements, Q may be defined as a rectangle, the ratio of whose sides is rational, and the mesh within Q may comprise any desired number of square elements. The nodes within Q can readily be numbered to minimise the mesh's bandwidth. This is easily achieved in a regular mesh.

The edges which make up the mesh in Q are now mapped into curves in P by the mapping $h = f \circ g^{-1}$. These curves intersect at the same angles as the corresponding edges in Q except at a finite number of

points on the boundary of P . This breakdown of conformality at some points of Q means that the curves into which the boundary edges of some elements map are not smooth. To produce a finite element mesh in P , the curves must be changed so that they are valid elemental edges of whatever element type it is desired should define the mesh in R . This usually means that the mapped element edges are approximated by straight lines joining the nodal points p_j . This is the case in the next chapter when exemplary polygons Q filled with three-noded triangles are selected. p_j is defined by

$$p_j = h(q_j) = f(g^{-1}(q_j)) = f(z_j)$$

where q_j is the complex number representing the j th node of the mesh in Q . If the nodes of the generated mesh are numbered in the natural way, that is node j is defined to be that node whose co-ordinates are given by p_j , then the topology of the mesh in P , the arrangement by which the edges join the nodes, is identical to that of the mesh in Q - hence the bandwidths of the two meshes are the same subject to the proviso discussed in section 4.5. Moreover in regions of P which are of greatest interest and where therefore the elements are smallest, this straight line approximation is least noticeable. In such a region, then, the edges intersect at very nearly the same angles as the corresponding edges of the mesh in Q . Conversely it is only in regions of least interest where element size is relatively large that the intersection angles differ significantly from those in polygon Q .

4.4 Choice of r_i for mapping f

A composite Schwarz-Christoffel transformation $h = f \circ g^{-1}$ has been described which may be used to generate a finite element mesh in a polygon P representing a general simply-connected region R . What have yet to be defined are the polygon Q containing a mesh of ideally shaped elements and the three degrees of freedom r_1 , r_2 and r_3 which are available to define f uniquely. It will be assumed that the corresponding quantities for the mapping g have been set so as to spread the pre-vertices of Q as evenly as possible around the circumference of the unit circle. In this section, it will be assumed that Q has already been chosen and the method used to select the r_i quantities will be described. It is these r_i which will determine the distribution of the boundary nodes of the mesh and it is their distribution which will determine where should be the smallest elements and where the largest. In order to arrange the boundary nodes in a desired way, let r positive integers t_1, t_2, \dots, t_r be defined together with vertex numbers of P , u_1, u_2, \dots, u_{r-1} where $r > 2$ and

$$0 < u_1 < u_2 < \dots < u_{r-1} < n.$$

It is these quantities which will determine how many boundary nodes there should be in the mesh in P , as well as specifying their distribution. They are defined as follows.

Let t_1 be the number of boundary edges which would ideally lie between vertex 0 and vertex u_1 .

For $j = 2, 3, \dots, r-1$, let t_j be the number of boundary edges which would ideally lie between vertex u_{j-1} and vertex u_j .

Let t_r be the number of boundary edges which would ideally lie between vertex u_{r-1} and vertex u_n (\equiv vertex 0).

The concept of boundary edges has been introduced in the definition of t_j because, although the desired number of edges between vertices of P is always integral, it will be necessary to consider fractions of boundary edges in order to select the best values of r_i .

At this point, there is a recapitulation of some of the notation used and some more is introduced. Polygon P has vertices

w_0, w_1, \dots, w_{n-1} , and $w_n = w_0$. $\underline{\theta}$ is the solution of equation (2-16) and for any $r_1, r_2, r_3, \underline{\omega}$ is also a solution of (2-16) where $\underline{\omega}$ is defined in (2-23). w_i is the image of $e^{i\omega_i}$ under mapping f . g is a Schwarz-Christoffel mapping between the unit circle in the z -plane and Q . The nodes of a finite element mesh in polygon Q are denoted by q_j and $q_j = g(z_j)$. Let $\psi_1, \psi_2, \dots, \psi_{N_Q}$ be defined so that the images of $e^{i\psi_k}$ under g are the boundary nodes of Q : that is the set $\{e^{i\psi_k}\}$ consists of those z_j points for which $|z_j| = 1$. Moreover let the ψ_k be ordered so that

$$\psi_0 < \psi_1 < \psi_2 < \dots < \psi_{N_Q-1} < \psi_{N_Q} = \psi_0 + 2\pi.$$

N_Q is the number of boundary nodes of Q .

What is needed is some mechanism for selecting the best values of r_i so as to position the $e^{i\omega_i}$ points in the most suitable positions with respect to the $e^{i\psi_k}$ points: a device for measuring how many boundary edges there would be between w_i points in P is required. In order to solve what is basically an integer-valued functional problem uniquely and by a well-tried numerical method, a real-valued function of a real-valued variable is introduced. $\phi(x)$ is required to be a monotonically increasing, continuously differentiable function so that its integral value is the number of complete boundary edges between the boundary node of Q , $g(e^{i\psi_0})$, and the general point of the boundary of Q , $g(e^{ix})$, where the range of x is restricted, for the moment, to

$$\psi_0 < x \leq \psi_{N_Q}. \quad (4-1)$$

Immediately the condition

$$\phi(\psi_k) = k \quad (4-2)$$

must be satisfied for $k = 1, 2, \dots, N_Q$. There is a natural and obvious choice for function ϕ to define $\phi(x)$: it is to apply the g mapping to e^{ix} and count the number of element edges, both complete and fractional, between the image point and $g(e^{ix_0})$. For example, if $g(e^{ix})$ happened to lie midway between $g(e^{i\psi_4})$ and $g(e^{i\psi_5})$, $\phi(x)$ would be assigned the value 4.5. Such a definition would satisfy the required conditions of ϕ being monotonically increasing and continuously differentiable. However, it was decided not to define ϕ in this way. As a numerical method would have to be used to select the best r_i values, very many evaluations of ϕ would have to be made. It would have been a considerable overhead to have had to evaluate a definite integral - implied in the evaluation of mapping g - every time this was done. Instead it was decided to fit a sequence of cubic polynomials, one cubic for each interval (ψ_k, ψ_{k+1}) , selected to satisfy equation (4-2). By using polynomials of degree three and by carefully choosing ϕ' at $x = \psi_k$, it can be ensured that function ϕ and its first derivative are continuous. In fact, in the interval

$$\psi_k < x < \psi_{k+1} \quad (1 \leq k \leq N_Q - 2),$$

the end point derivatives ρ_k and ρ_{k+1} were chosen as

$$\rho_k = \frac{2}{\psi_{k+1} - \psi_{k-1}}, \quad \rho_{k+1} = \frac{2}{\psi_{k+2} - \psi_k}.$$

The cubic ϕ is, then,

$$\begin{aligned}\phi(x) = k + \left[\frac{\rho_k + \rho_{k+1}}{\Delta\psi^2} - \frac{2}{\Delta\psi^3} \right] (x - \psi_k)^3 + \\ \left[\frac{3}{\Delta\psi^2} - \frac{2\rho_k + \rho_{k+1}}{\Delta\psi} \right] (x - \psi_k)^2 + \rho_k(x - \psi_k) \quad (4-3)\end{aligned}$$

where $\Delta\psi$ is written for $\psi_{k+1} - \psi_k$, and its derivative is

$$\begin{aligned}\phi'(x) = 3 \left[\frac{\rho_k + \rho_{k+1}}{\Delta\psi^2} - \frac{2}{\Delta\psi^3} \right] (x - \psi_k)^2 + \\ 2 \left[\frac{3}{\Delta\psi^2} - \frac{2\rho_k + \rho_{k+1}}{\Delta\psi} \right] (x - \psi_k) + \rho_k. \quad (4-4)\end{aligned}$$

It can immediately be seen that (4-3) satisfies (4-2) at $x = \psi_k$ and $x = \psi_{k+1}$ and that (4-4) satisfies the end condition $\phi'(\psi_k) = \rho_k$ and $\phi'(\psi_{k+1}) = \rho_{k+1}$. To prove that ϕ is a monotonically increasing function of x , let (4-4) be re-arranged as

$$\begin{aligned}\phi'(x) = \rho_k \left[1 - 3 \frac{x - \psi_k}{\Delta\psi} \right] \left[1 - \frac{x - \psi_k}{\Delta\psi} \right] + \\ \frac{x - \psi_k}{\Delta\psi} \left[\rho_{k+1} + 3 \left(1 - \frac{x - \psi_k}{\Delta\psi} \right) \left(\frac{2}{\Delta\psi} - \rho_{k+1} \right) \right]. \quad (4-5)\end{aligned}$$

By definition, ρ_k and ρ_{k+1} are each subject to the constraints

$$0 < \rho_k < 2/\Delta\psi, \quad 0 < \rho_{k+1} < 2/\Delta\psi \quad (4-6)$$

so that, when

$$0 \leq \frac{x - \psi_k}{\Delta\psi} \leq \frac{1}{3},$$

each part of (4-5) may be seen to be positive. In a similar way, (4-4) may be re-arranged as

$$\begin{aligned}\phi'(x) = \rho_{k+1} \frac{x - \psi_k}{\Delta\psi} & \left[3 \frac{x - \psi_k}{\Delta\psi} - 2 \right] + \\ & \left[1 - \frac{x - \psi_k}{\Delta\psi} \right] \left[\rho_k + 3 \frac{x - \psi_k}{\Delta\psi} \left(\frac{2}{\Delta\psi} - \rho_k \right) \right].\end{aligned}\quad (4-7)$$

When x lies in the range defined by

$$\frac{2}{3} \leq \frac{x - \psi_k}{\Delta\psi} \leq 1,$$

each part of (4-7) may be seen to be positive. Finally, to prove that

$\phi'(x) > 0$ when

$$\frac{1}{3} < \frac{x - \psi_k}{\Delta\psi} < \frac{2}{3},$$

(4-5) may be re-arranged as

$$\begin{aligned}\phi'(x) = \left(1 - \frac{x - \psi_k}{\Delta\psi} \right) & \left[\rho_k \left(1 - \frac{3}{2} \frac{x - \psi_k}{\Delta\psi} \right) + \frac{3}{2} \frac{x - \psi_k}{\Delta\psi} \left(\frac{2}{\Delta\psi} - \rho_k \right) \right] + \\ & \frac{x - \psi_k}{\Delta\psi} \left[\frac{1}{2} \rho_{k+1} \left(3 \frac{x - \psi_k}{\Delta\psi} - 1 \right) + \frac{3}{2} \left(1 - \frac{x - \psi_k}{\Delta\psi} \right) \left(\frac{2}{\Delta\psi} - \rho_{k+1} \right) \right]\end{aligned}\quad (4-8)$$

Once more, each part of (4-8) may be seen to be positive.

The domain of function ϕ may be extended to the entire real line by defining $\phi(x)$, when x is outside the range of (4-1), by

$$\phi(x) = xN_Q + \phi(x - 2x\pi),$$

where x is defined to be the largest integer not exceeding $x/2\pi$. By this definition, ϕ satisfies

$$\phi(x + 2\pi) - \phi(x) = N_Q$$

for all real x . Moreover, ϕ is continuously differentiable for all x . ρ_0 and ρ_{N_Q} may now be defined as

$$\rho_0 = \frac{2}{\psi_1 - \psi_{N_Q-1} + 2\pi}, \quad \rho_{N_Q} = \frac{2}{\psi_1 + 2\pi - \psi_{N_Q-1}}.$$

The cubic and its derivative defined in (4-3) and (4-4) now hold good for $\psi_0 < x < \psi_1$ and $\psi_{N_Q-1} < x < \psi_{N_Q}$ as well as for the range $\psi_1 < x < \psi_{N_Q-1}$ for which they were previously defined.

All the necessary machinery has now been assembled for tackling the problem of finding the r_i which best arrange the boundary nodes of the finite element mesh in P . It will be recalled that, for $j = 2, 3, \dots, r-1$, t_j was defined as the desired number of elemental edges between vertex u_{j-1} and u_j : the number of edges actually obtained is

$$\phi(\omega_{u_j}) - \phi(\omega_{u_{j-1}}),$$

or at least as nearly as the cubic ϕ models the actual number of boundary edges in P . In order to choose the best r_i , the difference between all the $\phi(\omega_{u_j}) - \phi(\omega_{u_{j-1}})$ and t_j needs to be minimised: S needs to be minimised where

$$S = [\phi(\omega_{u_1}) - \phi(\omega_0) - t_1]^2 + \sum_{j=2}^{r-1} [\phi(\omega_{u_j}) - \phi(\omega_{u_{j-1}}) - t_j]^2 + [\phi(\omega_n) - \phi(\omega_{u_{r-1}}) - t_r]^2. \quad (4-9)$$

A Newton method with linear search has been used to minimise S . This consists of finding r_i such that $\partial S / \partial r_i = 0$ for $i = 1, 2$ and 3 . This numerical method depends on having an approximation to the solution

and systematically improving it: if the set $\{r_i\}$ is an approximation to the solution, then $\{r_i + \mu \delta r_i\}$ is a better approximation where

$$0 = \frac{\partial S}{\partial r_i} + \sum_{j=1}^3 \frac{\partial^2 S}{\partial r_i \partial r_j} \delta r_j \quad (4-10)$$

and μ is a scaling factor selected to ensure that S decreases at each iteration. Equation (4-10) may be solved to give

$$\begin{bmatrix} \delta r_1 \\ \delta r_2 \\ \delta r_3 \end{bmatrix} = -H^{-1} \begin{bmatrix} \partial S / \partial r_1 \\ \partial S / \partial r_2 \\ \partial S / \partial r_3 \end{bmatrix}$$

where H is the hessian matrix

$$\left[\begin{array}{ccc} \frac{\partial^2 S}{\partial r_1 \partial r_1} & \frac{\partial^2 S}{\partial r_1 \partial r_2} & \frac{\partial^2 S}{\partial r_1 \partial r_3} \\ \frac{\partial^2 S}{\partial r_2 \partial r_1} & \frac{\partial^2 S}{\partial r_2 \partial r_2} & \frac{\partial^2 S}{\partial r_2 \partial r_3} \\ \frac{\partial^2 S}{\partial r_3 \partial r_1} & \frac{\partial^2 S}{\partial r_3 \partial r_2} & \frac{\partial^2 S}{\partial r_3 \partial r_3} \end{array} \right].$$

For completeness, the expressions for the first and second derivatives of ω , upon which $\partial S / \partial r_i$ and $\partial^2 S / \partial r_i \partial r_j$ depend, are included here. They are as follows:

$$\begin{aligned} \frac{\partial \omega}{\partial r_1} &= 2\pi, & \frac{\partial \omega}{\partial r_2} &= \frac{2e^{r_2} \xi}{\xi'^2 + 1}, & \frac{\partial \omega}{\partial r_3} &= \frac{2}{\xi'^2 + 1}; \\ \frac{\partial^2 \omega}{\partial r_1^2} &= 0, & \frac{\partial^2 \omega}{\partial r_1 \partial r_2} &= 0, & \frac{\partial^2 \omega}{\partial r_1 \partial r_3} &= 0; \end{aligned}$$

$$\frac{\partial^2 \omega}{\partial r_2^2} = - \frac{2e^{r_2} \xi (e^{2r_2} \xi^2 - r_3^2 - 1)}{(\xi'^2 + 1)^2},$$

$$\frac{\partial^2 \omega}{\partial r_2 \partial r_3} = - \frac{4e^{r_2} \xi \xi'}{(\xi'^2 + 1)^2},$$

$$\frac{\partial^2 \omega}{\partial r_3^2} = - \frac{4\xi'}{(\xi'^2 + 1)^2}.$$

Unfortunately, in many cases, there are several local minima of S as well as the single, desired, global minimum. How many local minima there are depends on both P and Q . It may be also that the global minimum of S is obtained for various sets of r_i . A trivial example of this latter case is if S is minimised at $\{r_1, r_2, r_3\}$, it will also be minimised at $\{r_1+1, r_2, r_3\}$ by definition of r_1 . The arrangement of the $e^{i\omega_i}$ points is identical, however. A less trivial example is if polygon Q is symmetrical. For example, suppose Q is a regular hexagon and S is minimised at $\{r_1, r_2, r_3\}$. Then S will also be minimised at $\{r_1+1/6, r_2, r_3\}$, $\{r_1+1/3, r_2, r_3\}$, etc. The symmetry of Q means that the $e^{i\psi_k}$ points are spaced around the circumference of the unit circle symmetrically and so the $e^{i\psi_k}$ points will be in an equivalent position relative to the $e^{i\omega_j}$ points whether they receive a rotation of r_1 or $r_1+1/6$ or $r_1+1/3$, etc. In order to ensure that the global minimum is obtained, it was decided to start the Newton minimisation procedure at a number of initial points and, having obtained the (possibly several different) $\{r_i\}$ at which $\partial S / \partial r_i = 0$, to select the $\{r_i\}$ which gave the smallest S . For the class of polygons described in the next chapter, six initial points are used: $r_1 = 0, 1/6, 1/3, 1/2, 2/3, 5/6, r_2 = r_3 = 0$.

It should be pointed out that there is no guarantee that the

attained minimum value of S will be zero. This depends on the selection of r_i , t_j , u_j and Q . In practical terms, what is happening is that, in the mapping from Q to P , the boundary of Q is being continuously deformed into that of polygon P . If P and Q are very dissimilar in shape, then whatever values are selected for r_i , the nodes may not be located in the desired positions. However the choice for Q is open and it is hoped that a suitably selected Q will ensure that a desired small value for S will be attained.

4.5 Alterations to the generated mesh

Function h has now been completely defined to map the z_j points onto the nodes p_j on the boundary and interior of polygon P in the w -plane. However there remain two further operations to be performed on the created mesh in P before the represented structure may be analysed by a finite element computer program.

Any node p_j which lies on a polygonal edge modelling a curved arc in the region R must be re-defined so that it lies on the nearest point of that arc to its calculated position. If too few polygonal edges were chosen to model such a polygonal arc, this re-definition of the boundary node co-ordinates may upset the desired element shape in such regions so this is a point to be borne in mind when choosing P to model R .

The second point is as follows. Let a particular vertex w_i be considered. Then unless by coincidence in the choice of r_i , w_i happens to be the image under h of some point q_j - that is unless $w_i = h(q_j)$ for some j or, equivalently, $w_i - \psi_k$ is a multiple of 2π for some k - then no node of the mesh in Q will be mapped onto w_i . It is true, however, that w_i lies on the image of one of the boundary

element edges of the mesh in Q but, since h is not conformal at $g^{-1}(e^{iw_i})$, this image edge is not smooth. Hence when this edge is approximated by an acceptable element edge (typically a straight line), the effect is to cut off the corner at w_i . Clearly it is one of the most fundamental requirements of a mesh that there should be a node at all vertices w_i of P . The possible exception to this is where w_i is at the junction of two polygonal edges, each of which is modelling the same curved arc.

Assuming that there should be a node at w_i , there are two ways of achieving this end. Firstly the co-ordinates of the nearest boundary node to w_i could be re-defined to those of w_i . This would mean that the topology of the mesh would remain unchanged and would be an acceptable move if the shapes of the neighbouring elements were not significantly altered.

The second alternative is to add a new node at w_i and re-define the surrounding elements, or add a new element, to contain it. The new node must be carefully numbered, however, so as not to increase the bandwidth of the mesh unnecessarily.

It has not proved difficult to accomplish either of these alternatives or indeed to establish a condition so that one or other may be chosen automatically from within a computer program.

CHAPTER 5

Examples. Choice of a class of Q

5.1 Choice of polygon Q

Until now, the choice of polygon Q has been left unspecified, only that it should be completely filled with ideally shaped elements. In this section, this choice is restricted so that Q is of a particular shape. The general method described is not restricted in this way: limiting the shape of Q is done merely for definiteness and so that examples may be given. For definiteness, let the type of finite element be a three-noded triangle and let the question be posed "How best may Q be selected so that it is filled entirely with ideally shaped elements?"

Firstly, what is an ideally shaped element? It has already been stated that a symmetrical element best models the variation of the quantity (be it displacement, temperature, velocity, etc.) over the finite element mesh: the best triangular element is equilateral. A polygon filled with equilateral triangles has a general form exemplified by that in figure 5-1. The interior angles at the vertices of such a polygon are all multiples of $\pi/3$. However, anticipating how such a polygon would be deformed by a composite Schwarz-Christoffel mapping onto the target polygon P , vertices with angle $\pi/3$ (like that marked * in the figure) must be rejected immediately. This is because the vertex * along with the two nodes adjacent to it, all of them boundary nodes, would be mapped onto the boundary of P . When the mapped element edges were replaced by straight lines, the resulting element in P would have three collinear nodes and zero area, clearly unacceptable. The effect of mapping a vertex whose interior angle is $2\pi/3$ onto the straight

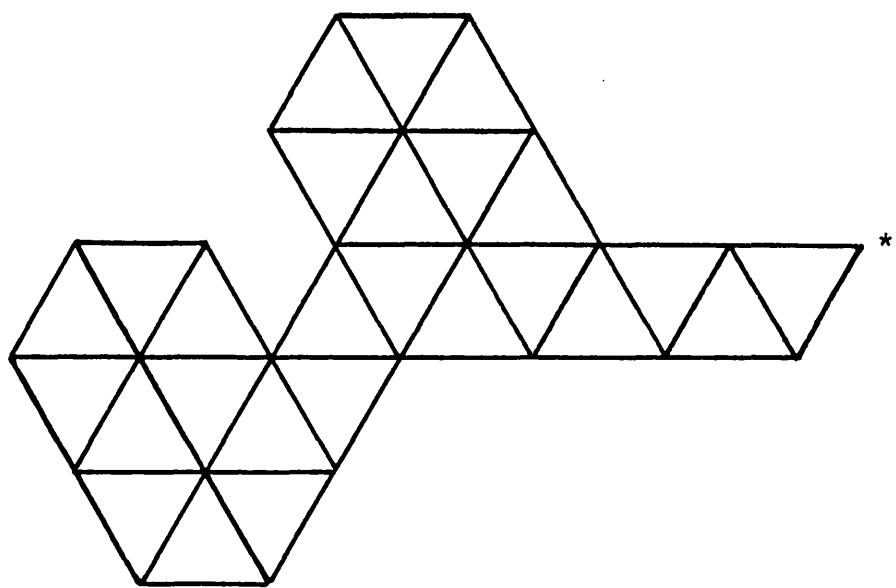


Figure 5-1. General polygon filled with equilateral triangles.

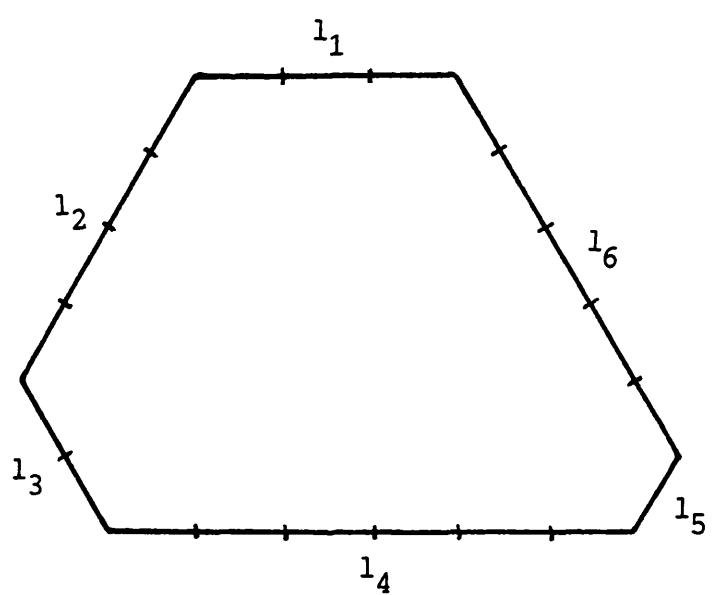


Figure 5-2. General $2\pi/3$ hexagon with sides l_i .

boundary of the target polygon would be to "open out" the angle and increase the original element angles of $\pi/3$ to approximately $\pi/2$. The word "approximately" is used here because of the way in which the mapped element edges are approximated by straight lines to produce a mesh in P . Similarly, the 4 or 5 finite elements which are adjacent to vertices of Q whose interior angles are $4\pi/3$ or $5\pi/3$, would have their angles of $\pi/3$ reduced to approximately $\pi/4$ or $\pi/5$ respectively. Polygons Q which have vertices with interior angles of $4\pi/3$ or $5\pi/3$ are not considered for the rest of this study. This is not for the reason that they would not produce good finite element meshes but more for the reason that it is easier to enumerate polygons which have interior angles of $2\pi/3$ only. Using the more general polygon with angles of $2\pi/3$, $4\pi/3$ and $5\pi/3$ remains an active research area of the writer.

5.2 Enumerating $2\pi/3$ hexagons

The question which has been posed is "How many polygons are there, all of whose interior angles are precisely $2\pi/3$, which have a given perimeter?" Such a polygon can immediately be seen to be a hexagon. Moreover interest is restricted to hexagons whose side lengths are multiples of the side of the fundamental equilateral triangle which fills P . For convenience, let the length of this fundamental triangle side be unity. The question can then be re-written as "How many $2\pi/3$ hexagons with a given integral perimeter are there which have positive integer side lengths?" where for convenience, the term " $2\pi/3$ hexagon" will be used to mean a hexagon, all of whose interior angles are equal to $2\pi/3$. A typical hexagon satisfying these conditions is shown in figure 5-2.

The lengths of the sides have been designated as l_1, l_2, l_3, l_4, l_5 and l_6 in anti-clockwise order. Clearly these are not independent as they satisfy the equations

$$\begin{aligned} l_1 + l_2 &= l_4 + l_5, & l_2 + l_3 &= l_5 + l_6, \\ l_3 + l_4 &= l_6 + l_1, & p = l_1 + l_2 + l_3 + l_4 + l_5 + l_6, \end{aligned} \quad (5-1)$$

where p is the perimeter of the hexagon. All the l_i must be positive integers.

In order to solve the combinatorial problem of how many different hexagons with a given integral perimeter p there are, the following procedure was adopted. If all the arrangements with perimeter $p-2$ and $p-1$ are already known, hexagons with perimeter p may be constructed as follows: if a hexagon with ordered sides $\{l_1, l_2, l_3, l_4, l_5, l_6\}$ has perimeter $p-2$, then the hexagons with sides

$\{l_1+1, l_2, l_3, l_4+1, l_5, l_6\}$, $\{l_1, l_2+1, l_3, l_4, l_5+1, l_6\}$ and $\{l_1, l_2, l_3+1, l_4, l_5, l_6+1\}$ satisfy the equations in (5-1). This generation of new hexagons is shown in figure 5-3. Alternatively if a hexagon with sides $\{l_1, l_2, l_3, l_4, l_5, l_6\}$ has perimeter $p-1$ and if one side (say l_2) is not less than 2 units in length, then the hexagon with sides $\{l_1+1, l_2-1, l_3+1, l_4, l_5, l_6\}$ satisfies equations (5-1) and has perimeter p . As many new hexagons of perimeter p as there are sides $l_i \geq 2$ may be generated in this way as shown in figure 5-4.

It can readily be seen that, if all the hexagons of perimeter $p-2$ and $p-1$ are known, then these two methods generate all the hexagons of perimeter p . So starting the procedure with the hexagon of minimum perimeter (6), all required hexagons can be generated. Of course, using a procedure like this produces many repetitions, including those by rotations, for example figure 5-3(c) and figure 5-4(c). In order

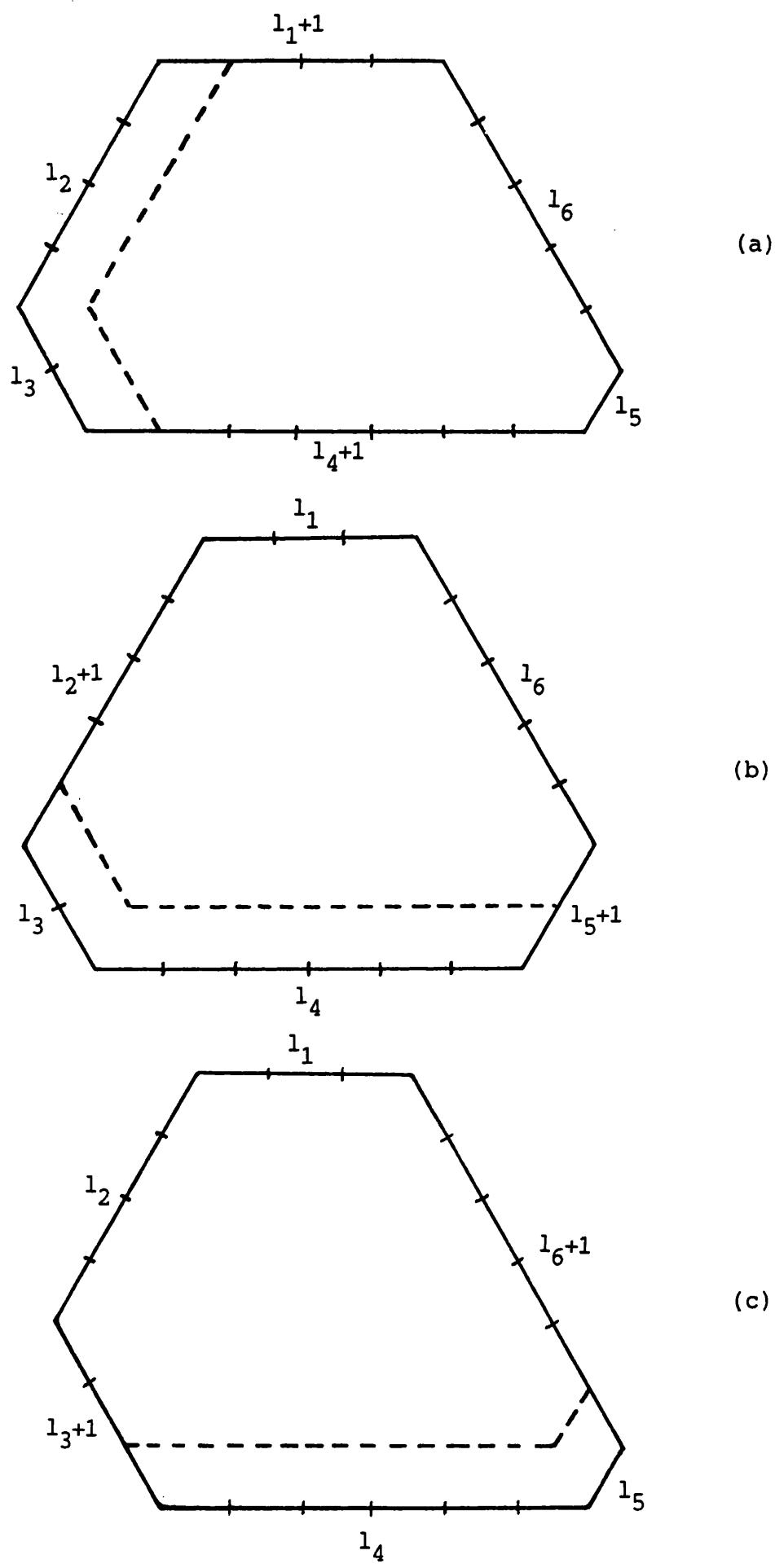
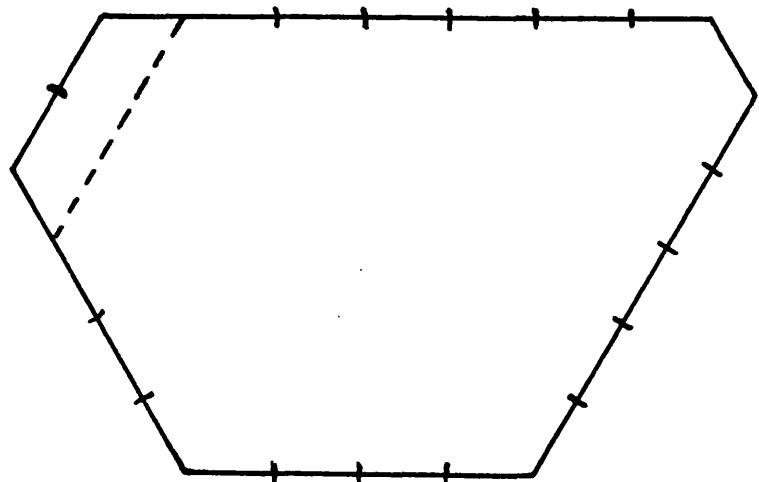
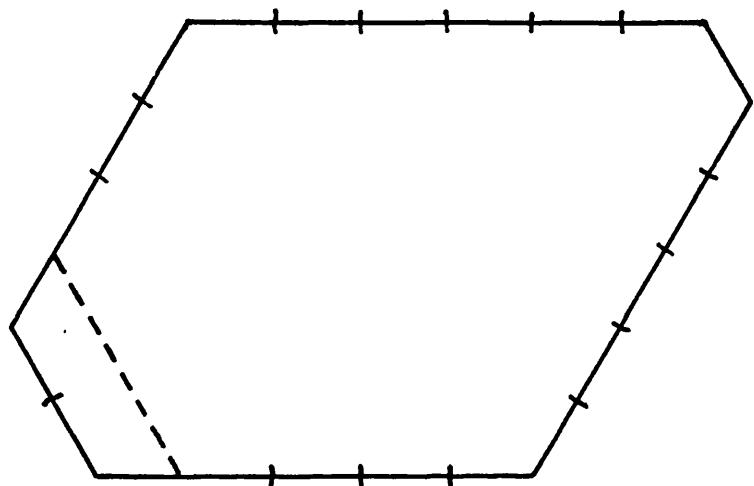


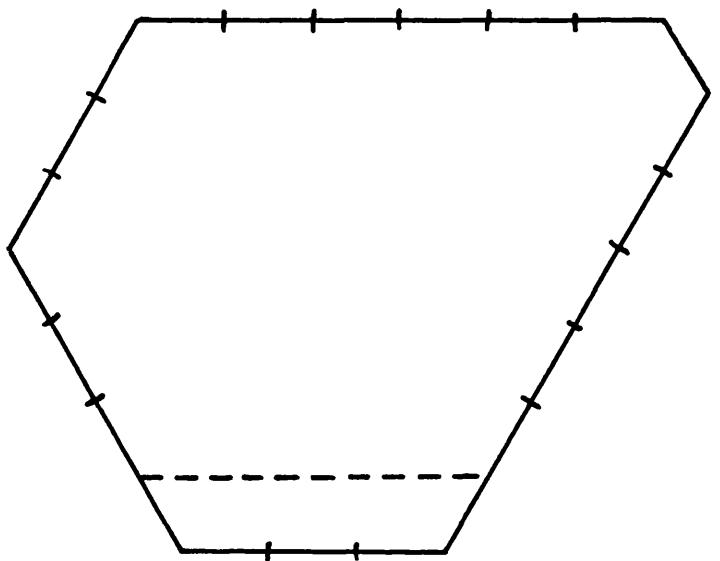
Figure 5-3. Newly generated hexagons obtained by extending two opposite sides.



(a)

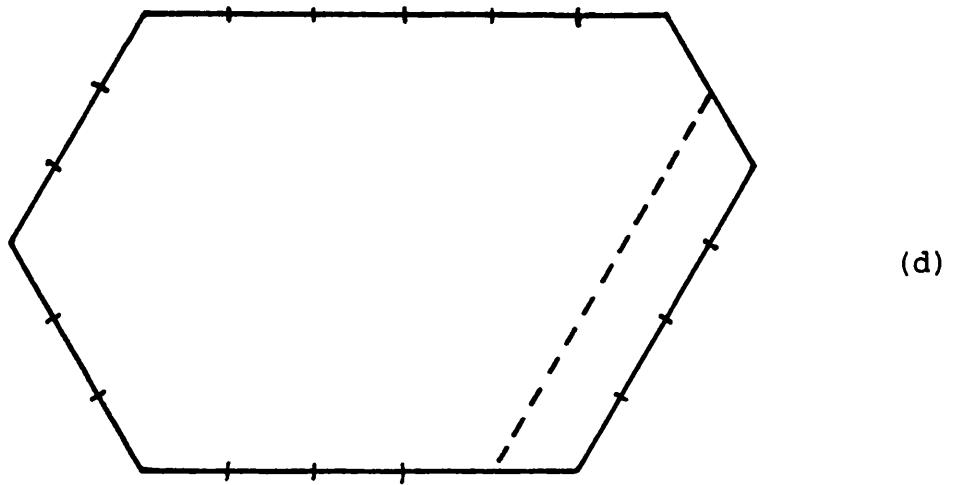


(b)

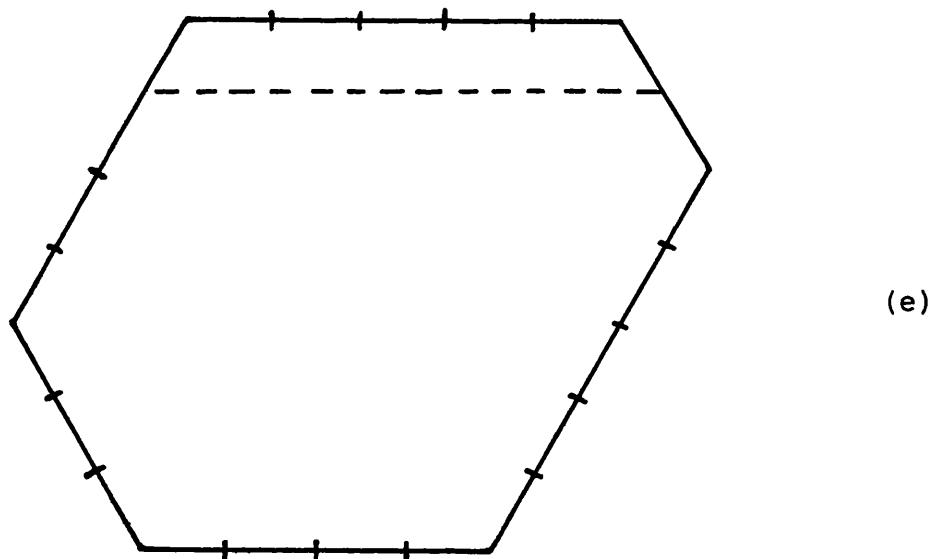


(c)

Figure 5-4. Newly generated hexagons obtained by increasing the lengths of two sides and decreasing the length of the intervening side.



(d)



(e)

Figure 5-4 continued.

to ensure that all the required hexagons of perimeter p were found but that no repetitions were included, the following procedure was followed.

For each hexagon with sides $\{l_1, l_2, l_3, l_4, l_5, l_6\}$, the six integers

$$l_1 p^5 + l_2 p^4 + l_3 p^3 + l_4 p^2 + l_5 p + l_6,$$

$$l_2 p^5 + l_3 p^4 + l_4 p^3 + l_5 p^2 + l_6 p + l_1, \text{ etc.}$$

(i.e. with cyclic rotation of the sides) were computed. The maximum was calculated and stored as a unique record of the hexagon's side lengths.

Two or more of these integers' being equal indicates some symmetry in the hexagon. Being able to record the geometry of each hexagon as a unique single integer enabled checking of repetitions to be easily performed. A table of the numbers of different arrangements of sides for a given hexagonal perimeter follows.

p	Number of $2\pi/3$ hexagons with perimeter p	(a)	(b)
6	1	1	0
7	0		
8	1	1	1
9	1		
10	2	2	3
11	1		
12	5	5	1
13	2		
14	6	6	3
15	5		
16	9	9	6
17	6		
18	15	15	3
19	9		
20	18	18	6
21	15		
22	24	24	10
23	18		
24	34	34	6
25	24		
26	40	40	10
27	34		
28	50	50	15
29	40		
30	65	65	10
31	50		
32	75	75	15
33	65		

34	90	90	21
35	75		
36	111	111	15
37	90		
38	126	126	21
39	111		
40	147	147	28
41	126		
42	175	175	21
43	147		
44	196	196	28
45	175		
46	224	224	36
47	196		
48	260	260	28
49	224		
50	288	288	36
51	260		
52	324	324	45
53	288		
54	369	369	36
55	324		
56	405	405	45
57	369		
58	450	450	55
59	405		
60	505	505	

Table 2

As an example, the 18 different hexagons for $p = 23$ are drawn in figure 5-5.

Some observations can be made from the table without further comment: if $M(p)$ denotes the number of unique hexagons with perimeter p , then for any positive integer m , the following relationships are true:

$$M(6m) = M(6m+3),$$

$$M(6m+2) = M(6m+5),$$

$$M(6m+1) = M(6m-2).$$

The $M(p)$ values which are different are highlighted in column (a) of the table and their forward differences shown in column (b). The differences can be seen to be triangular numbers.

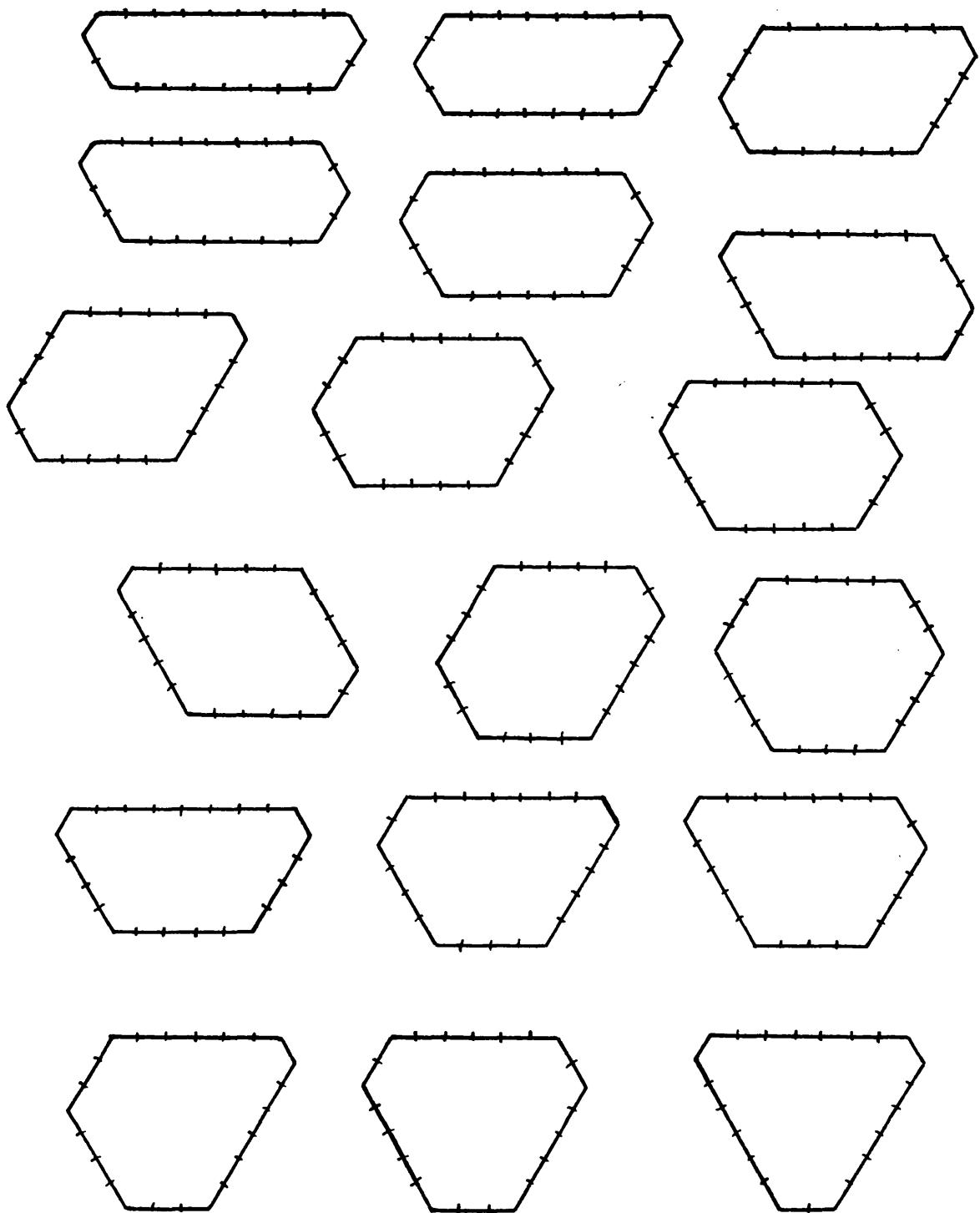


Figure 5-5. The 18 different $2\pi/3$ hexagons with perimeter 23.

5.3 Example of the variation of function ϕ with X

Now that a class of polygons Q has been selected, an example of the variation of function ϕ with X may be given. ϕ was defined in section 4.4. The Q chosen is the hexagon illustrated in figure 3-3. This has 24 element edges and the $e^{\imath\psi_k}$ are shown as the points on the circumference of the circle in figure 3-4, with $\psi_0 = 0$. The variation of ϕ and ϕ' with X in the range $0 \leq X \leq 2\pi$ is shown in figure 5-6. Table 3 shows the ψ_k values at which ϕ is integral.

k	ψ_k	k	ψ_k
0	0.0000	13	3.1915
1	0.0983	14	3.5214
2	0.1978	15	3.7264
3	0.3235	16	3.9385
4	0.6104	17	4.3113
5	1.0456	18	4.6948
6	1.5349	19	4.9227
7	1.9292	20	5.1495
8	2.1272	21	5.5276
9	2.3028	22	5.8985
10	2.5526	23	6.1622
11	2.7142	24	6.2832
12	2.8844		

Table 3

5.4 Numbering the nodes of a $2\pi/3$ hexagon

In this section the topic of numbering the nodes of an equilateral triangular finite element mesh in a $2\pi/3$ hexagon is covered. It has already been stated that the numbering of a regular mesh is straightforward. The object is to ensure that the bandwidth of the mesh (section 4.1) is as small as possible. The way to number the nodes of

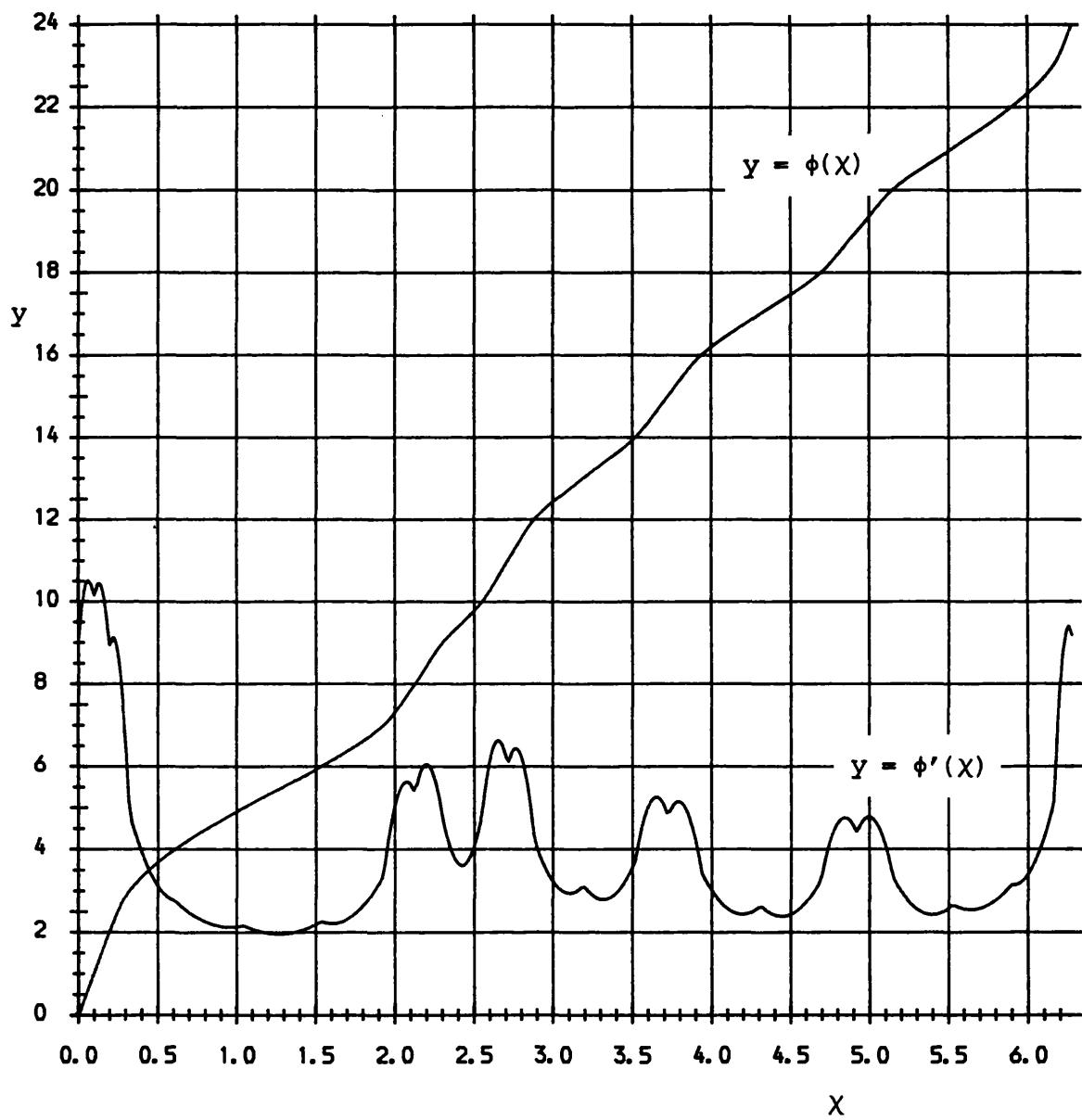


Figure 5-6. Example graphs of $y = \phi(x)$ and $y = \phi'(x)$ where ϕ is defined from the ψ_k values in table 3.

a mesh like that in figure 5-7 is in the same way as the words on the page of a book are ordered. However, there still remains the problem of which of the six possible orientations of the hexagon should be selected: which should be the top left-hand corner? Should it be at the intersection of sides l_6 and l_1 or at the intersection of l_1 and l_2 , etc? It is soon seen to be apparent that if side l_1 is selected to be the topmost side, then the bandwidth is given by

$$v_1 = l_1 + \min \{l_6, l_2\} + 1 \quad \text{if } l_2 \geq l_6,$$

or

$$v_1 = l_1 + \min \{l_6, l_2\} + 2 \quad \text{if } l_2 < l_6.$$

If l_2 is the topmost side, the bandwidth is

$$v_2 = l_2 + \min \{l_1, l_3\} + 1 \quad \text{if } l_3 \geq l_1,$$

or

$$v_2 = l_2 + \min \{l_1, l_3\} + 2 \quad \text{if } l_3 < l_1,$$

and so on for the remaining four sides. However, from (5-1),

$$l_1 + \min \{l_6, l_2\} = l_4 + \min \{l_3, l_5\},$$

$$l_2 + \min \{l_1, l_3\} = l_5 + \min \{l_4, l_6\},$$

$$l_3 + \min \{l_2, l_4\} = l_6 + \min \{l_5, l_1\}.$$

Moreover equations (5-1) imply that $l_2 < l_6$ if and only if $l_3 < l_5$ so that $v_1 = v_4$. Similarly $v_2 = v_5$ and $v_3 = v_6$. This means that three of the six allowable orientations may be discarded as they are repetitious. All that is required is to select the orientation whereby

$$\min \{v_1, v_2, v_3\}$$

attains its minimum value.

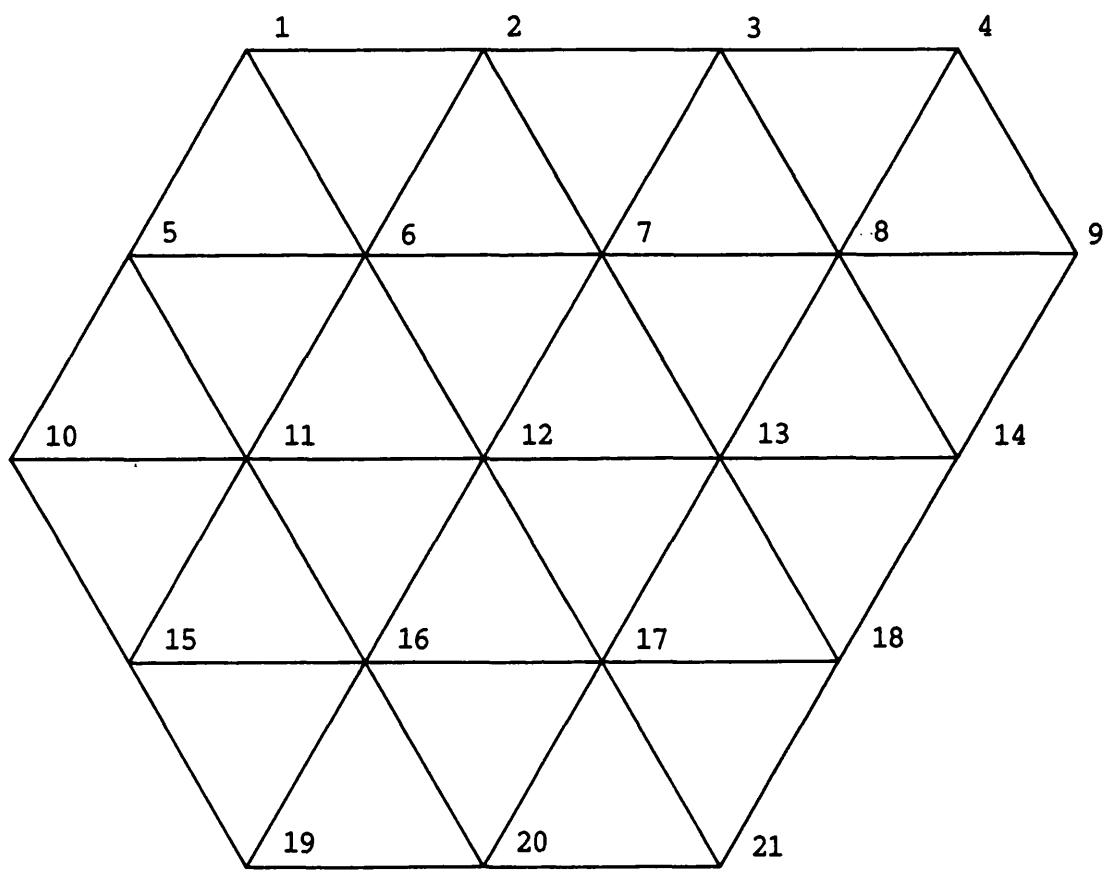


Figure 5-7. A numbered triangular mesh in a $2\pi/3$ hexagon.

5.5 Description of a computer program

The mesh generation scheme which has been described is general for meshes in simply-connected two-dimensional domains. In order to be able to present example meshes and to describe a complete computer program, in this section the type of mesh is selected and the class of polygons Q is specified. In fact only a very small part of the program is specific for three-noded triangles. The complete program is listed in appendix 4. The following is a brief description of the main steps in the program:

1. Input the definition of R , the region in which the mesh is to be generated.
2. Select polygon P to model R and define the complex numbers representing the vertices of P .
3. Find the Schwarz-Christoffel transformation f between the unit circle and P to define ω_i .
4. Input boundary information to describe the ideal distribution of the boundary element edges.
5. Generate all the $2\pi/3$ hexagons with required perimeter.
6. For each generated $2\pi/3$ hexagon ($= Q$), complete steps 7 to 11.
7. Find the Schwarz-Christoffel transformation g between the unit circle and Q .
8. Define the boundary nodes of Q .
9. Find the inverses of the points in step 8 under g . Order these points to define ψ_k .
10. Find r_1 , r_2 and r_3 which best arrange ω_i amongst ψ_k to minimise the penalty function S in equation (4-9).
11. Store the lowest S found so far and the $2\pi/3$ hexagon which

produced this S .

12. For the $2\pi/3$ hexagon which produced the lowest S , select the orientation so that the bandwidth of the mesh will be the smallest.
13. Generate the complete equilateral triangular mesh in the selected Q .
14. Find the inverses of the nodes of Q under g .
15. Apply f to the points in step 14 to produce the mesh in P .
16. Make adjustments to the generated mesh.

Details. The particulars of most of the above steps have already been covered. However, for those steps which are peculiar to the three-noded triangular element, details are given here.

In step 6, in fact a limit is put on the number of generated hexagons used. As can be seen from the table 2, this number increases rapidly with the perimeter p . In order to save computer time, a limit of 40 is placed on the number of trial hexagons used. If the generated number exceeds 40, a carefully selected sample of 40 is used.

In step 16, alterations to the generated mesh fall into three categories.

16(a). Correcting numerical errors. When calculating the positions of nodes which should be on a straight boundary edge, because of numerical inaccuracies in evaluating the integrals in equation (2-6), there is inevitably a slight error. The co-ordinates of such nodes are amended so that they accurately lie on a polygonal side.

16(b). Moving a node to a curved arc. Those nodes which lie on a side of P which represents a curved boundary of R must be moved. As well as storing the co-ordinates of the vertices of P , the program also stores the geometry of R which P is representing. Each

node to which this applies is moved to the nearest point of the boundary of R .

16(c). Ensuring that there is a node at the vertices of P . As mentioned earlier, in general there will not be a node at each vertex of P in the generated mesh. The program ensures that, at each vertex of P which is not at the junction of two polygonal sides both modelling the same curved arc, either a node is moved to the vertex or else a new node is added. The implemented algorithm is as follows.

(i) At the vertex w_i , find the nearest node and the next nearest node in the same direction.

(ii) If the distance from w_i to the nearest node is less than one quarter the distance from w_i to the next nearest node, and both nodes are on the same polygonal side adjacent to w_i , then move the nearest node to w_i . (The factor one quarter is selected so that the shapes of generated elements is not overly spoiled.)

If the nearest node is not sufficiently close to w_i , a new node is added there: since this operation alters the topology of the mesh, care must be taken when numbering the added node that the bandwidth of the mesh is not increased unnecessarily. Let the two nodes straddling vertex w_i be numbered i_1 and i_2 . Let the third node of the element be i_3 (figure 5-8). Increment by one all nodes numbered $\geq \min\{i_1, i_2\} + 1$. Add a new node, number $j_4 = \min\{i_1, i_2\} + 1$, at w_i . Suppose that the node numbers of the element are now $\{j_1, j_2, j_3\}$. Replace this element by two elements with nodes $\{j_1, j_3, j_4\}$ and $\{j_4, j_3, j_2\}$ (figure 5-9). In addition, the newly defined elements must be numbered. However, this process is not critical like the re-numbering of the nodes, having accepted the assumptions of section 4.1.

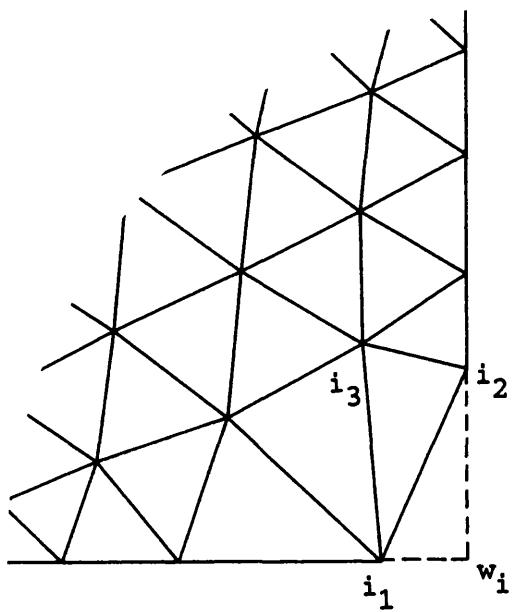


Figure 5-8. Original numbering of nodes near a vertex (w_i) where a node is to be added.

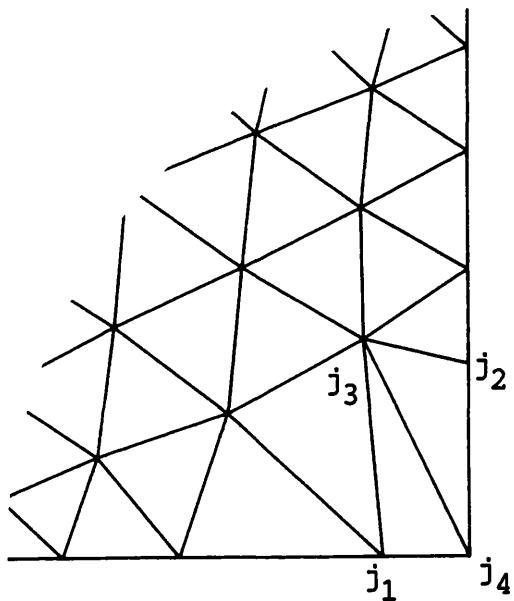


Figure 5-9. New node added and node numbering amended.

5.6 Example meshes

This section shows several example meshes generated using the program described in the previous section. Some examples are from problems with which the author has been involved: others are generated in regions considered by other investigators and reported in the literature.

Figure 5-10 shows a mesh generated in a simple L-shaped region.

Figure 5-11 shows three meshes generated in the region shown in figure 2-2. (c) has double the number of elements of the mesh in (a) and (b) shows that the elements need not be arranged symmetrically if the loading conditions are known to produce large stresses in just one part of the mesh.

Figure 5-12 shows a mesh generated in a region representing a dam and foundation which Zienkiewicz uses as an analysis example [21].

Figure 5-13 shows a mesh generated in a region which represents the axi-symmetric cross-section of a flywheel with fine element grading around a small circular arc.

Figure 5-14 shows a mesh generated in a rectangular region representing one quarter of a cracked plate. It was this type of mesh which was used for the calculations described in the paper in appendix 2 although at the time that paper was written, only regular hexagons were used as polygon Q. Very small elements may be generated around the area close to the singularity.

Figure 5-15 shows a mesh generated in a region representing a test specimen used by Felce [22]. Small elements are placed in that part of the model where high stresses (and, in due course, failure) are known to occur. Because of the symmetry of both geometry and loading, only one eighth of the specimen needs to be modelled.

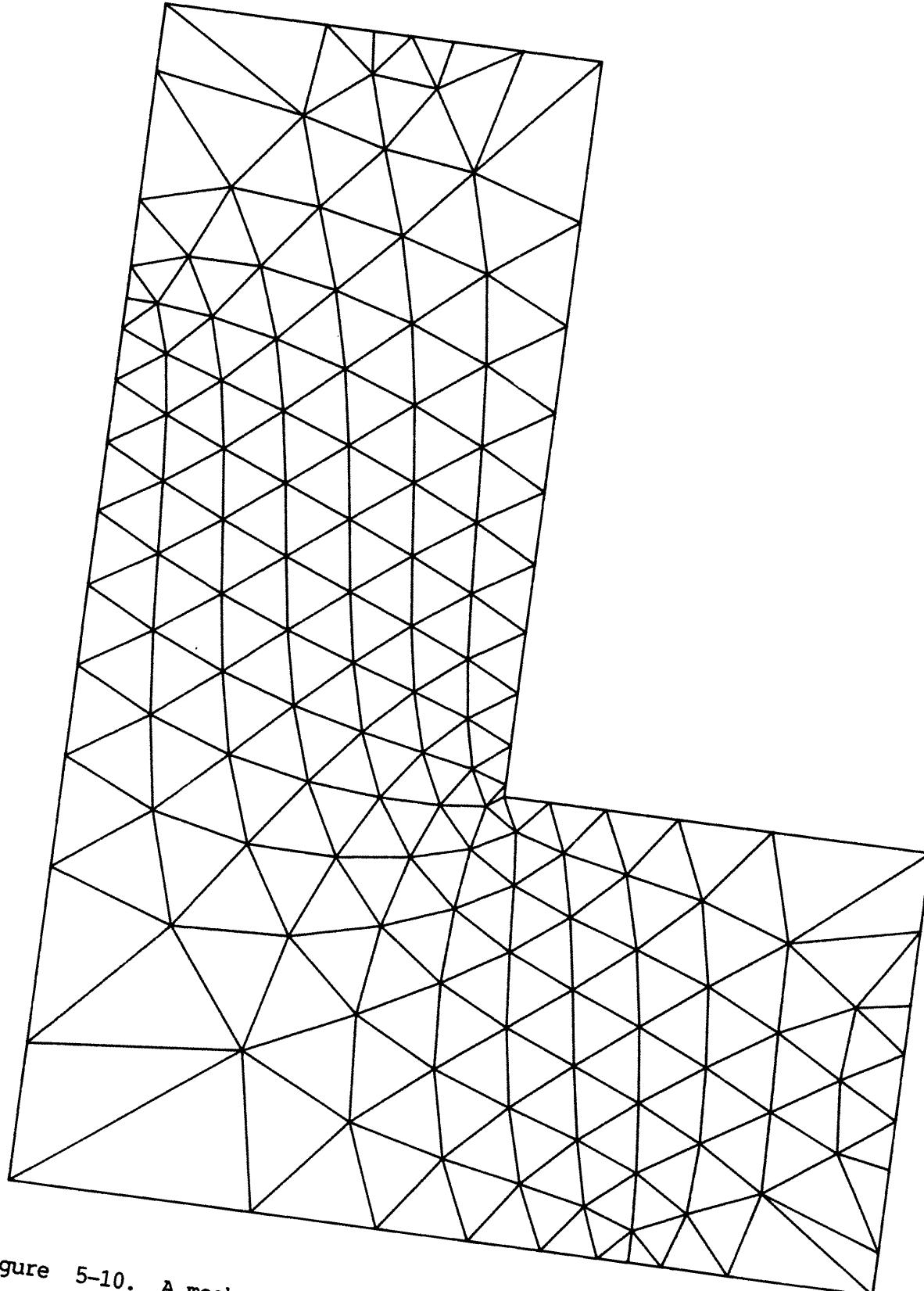
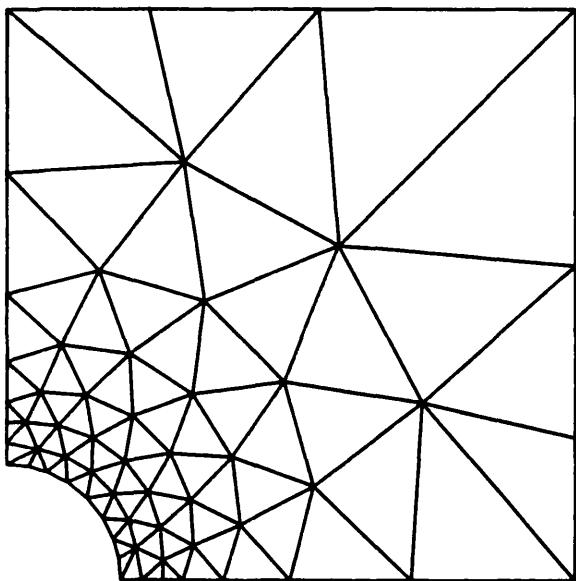
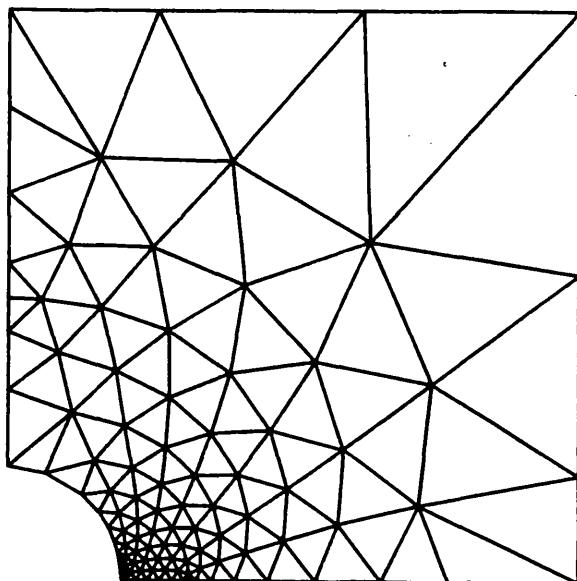


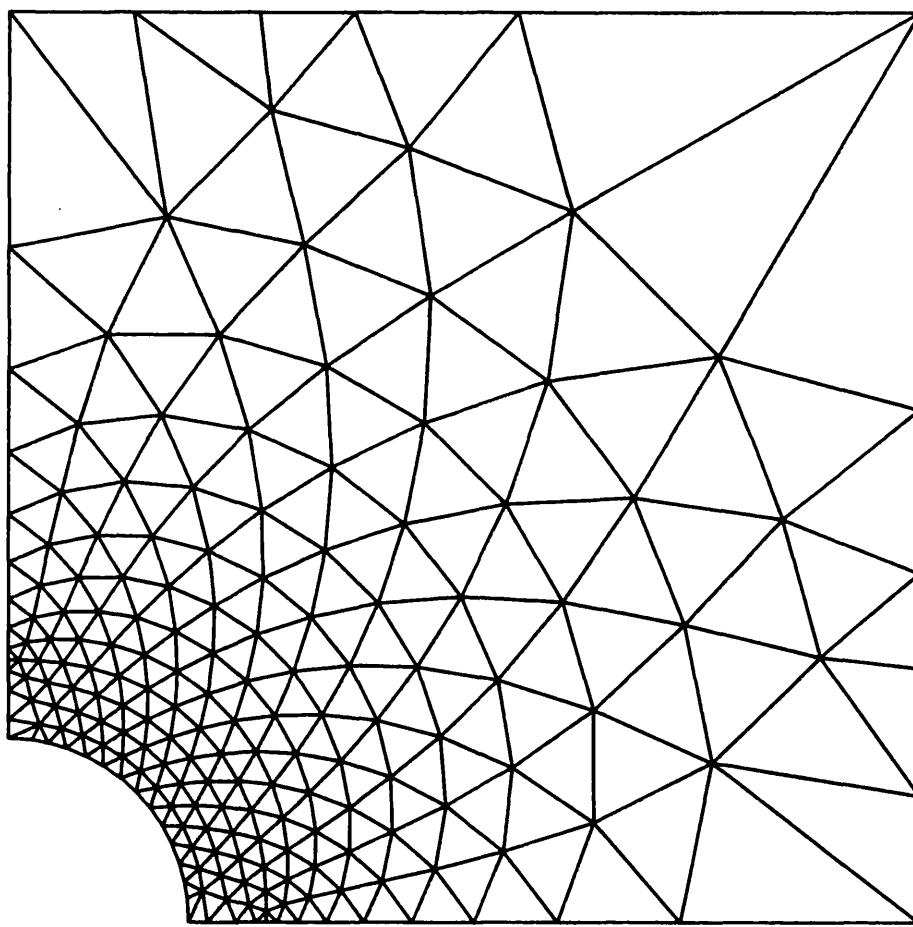
Figure 5-10. A mesh generated in a simple L-shaped region.



(a)

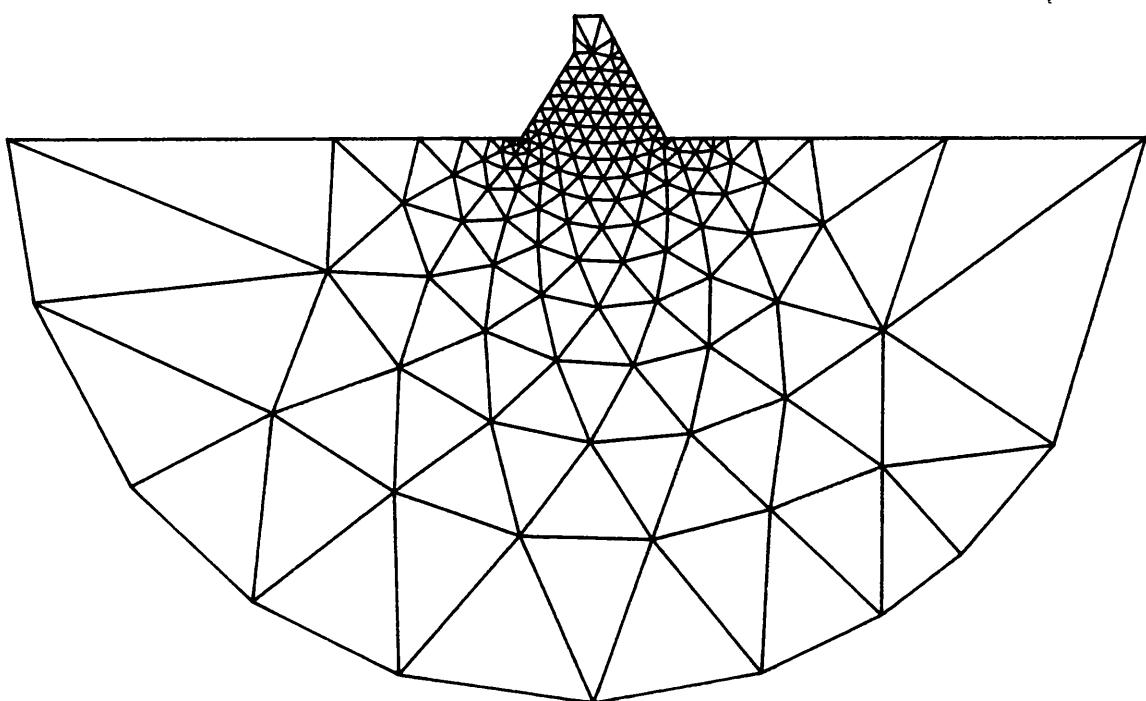


(b)

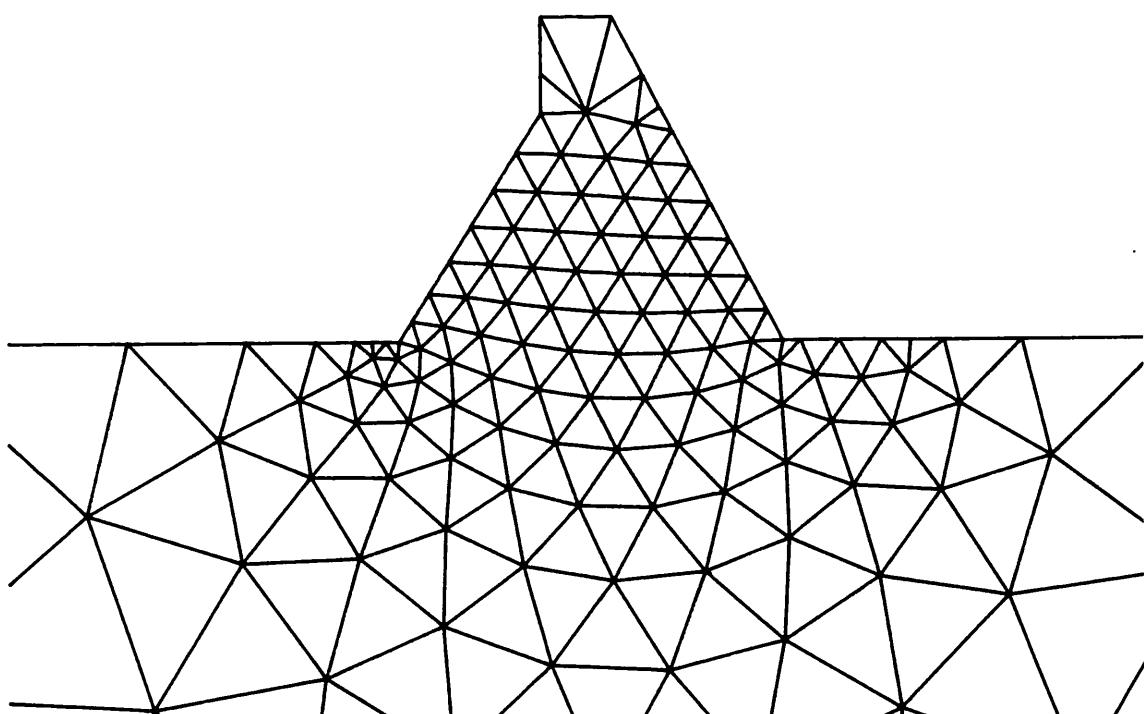


(c)

Figure 5-11. Three meshes generated in the region shown in figure 2-2:
(b) elements arranged non-symmetrically; (c) double
density of (a).



(a)



(b)

Figure 5-12. (a) an example mesh generated in a region representing a dam and foundation from Zienkiewicz [21]; (b) detail.

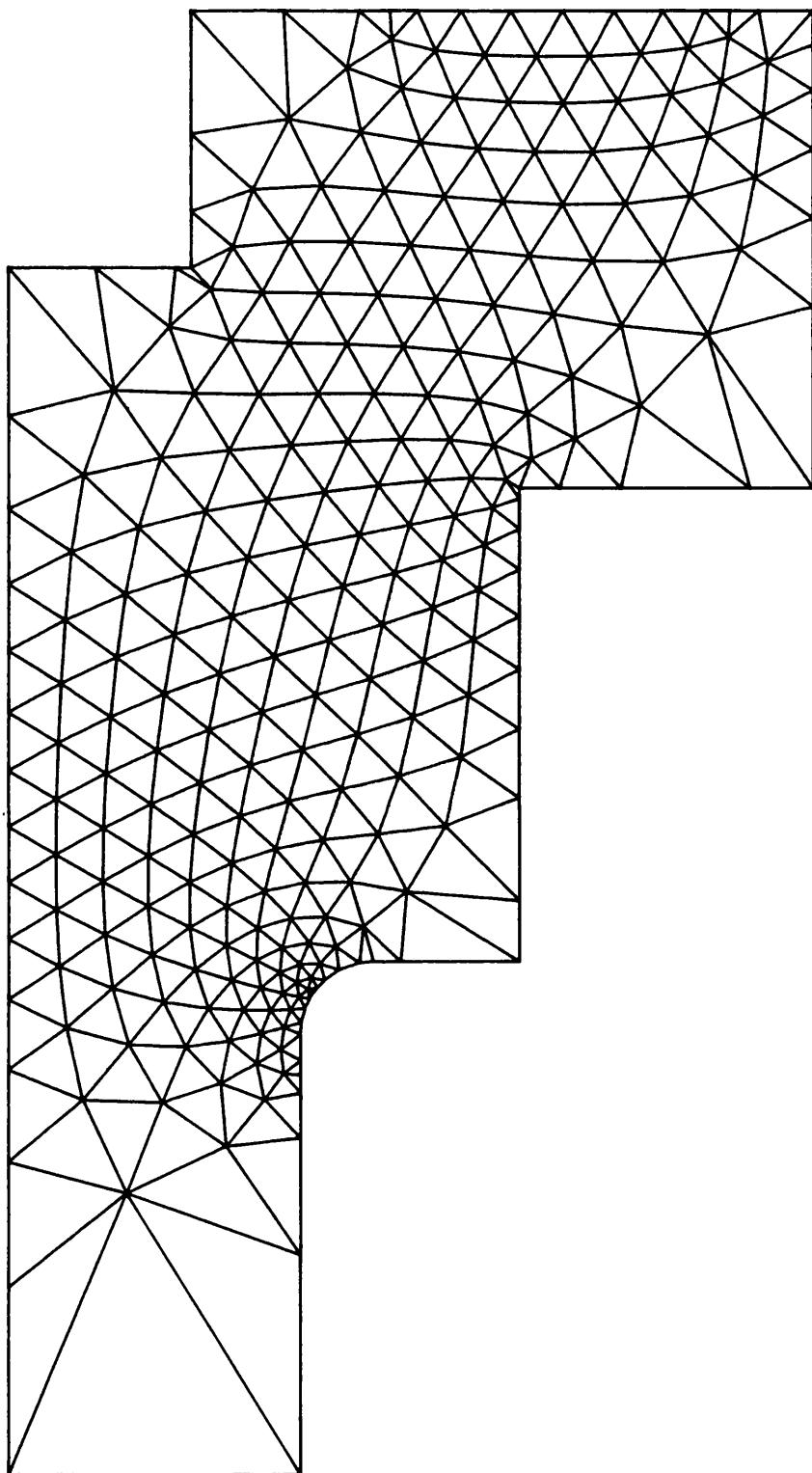


Figure 5-13. Mesh generated in a region which represents the axi-symmetric cross-section of a flywheel.

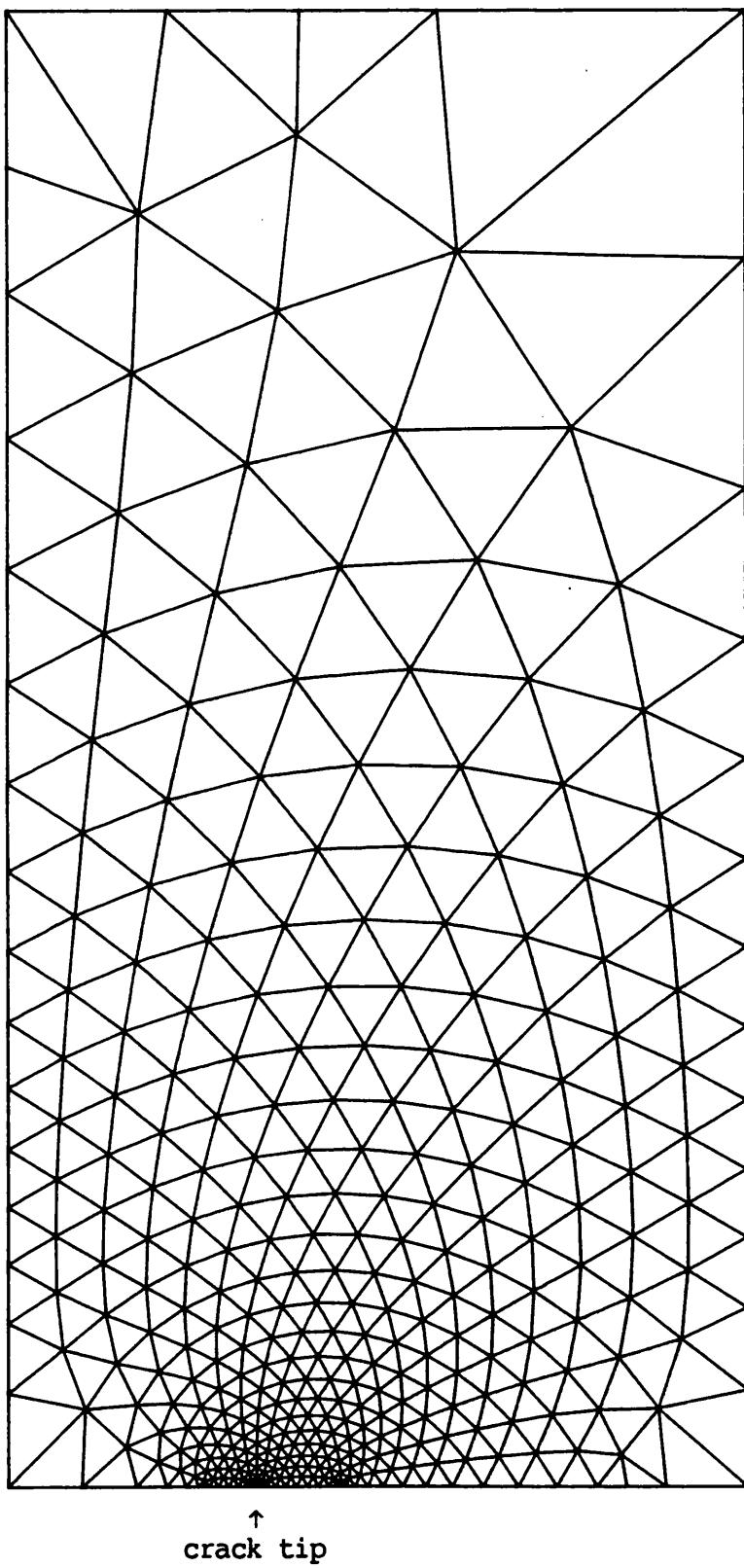


Figure 5-14. Mesh generated in a rectangular region representing one quarter of a cracked plate.

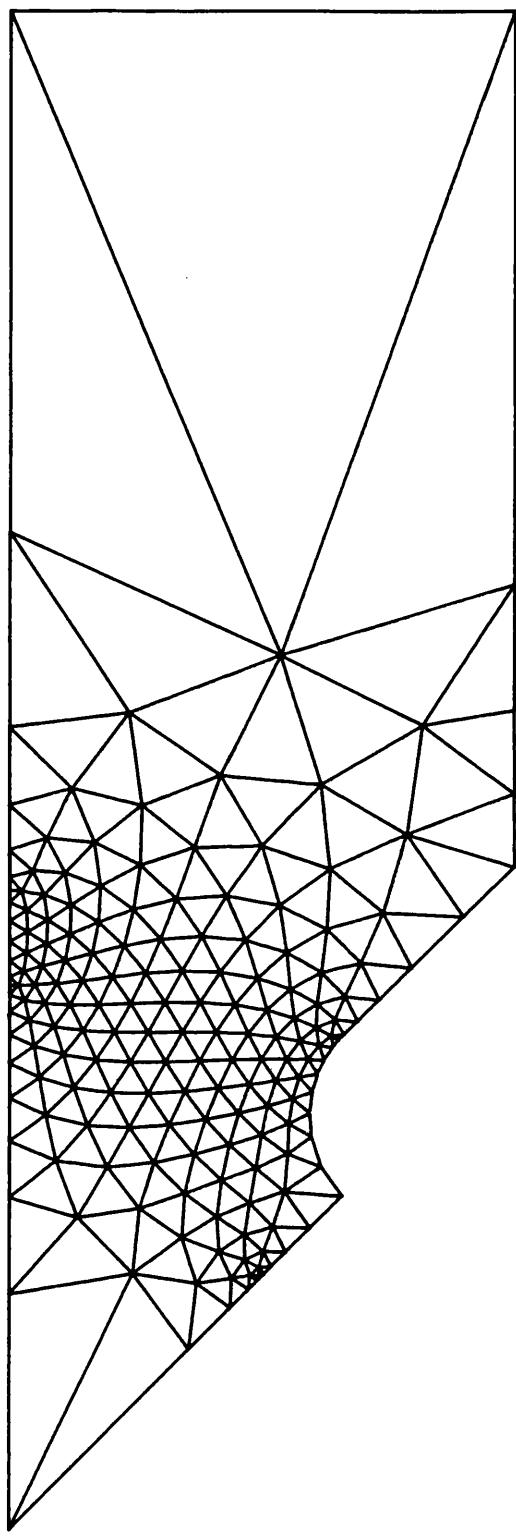


Figure 5-15. Mesh generated in a region representing one eighth of a test specimen used by Felce [22].

Figure 5-16 shows a mesh generated in a region typical of those analysed in ice studies described by Ponter and Brown [23]. It must be pointed out that the meshes figures 5-15 and 5-16 are examples only and were not in fact those used in references 22 and 23. For these steady state plane strain studies, straightforward three-noded triangular meshes were not suitable.

Figure 5-17 and 5-18 show example meshes in regions studied by Zienkiewicz, Liu and Huang [24]. These are suitable meshes for plane and axi-symmetric extrusion.

Figure 5-19 shows a mesh generated in a region analysed by Yokoyama [25]. This region represents a rack tooth where the loading is not symmetrical, hence the whole region needs to be modelled.

Figure 5-20 shows an example mesh for the modelling of incompressible fluid flow past an aerofoil. This problem is described by Jiang and Carey [26].

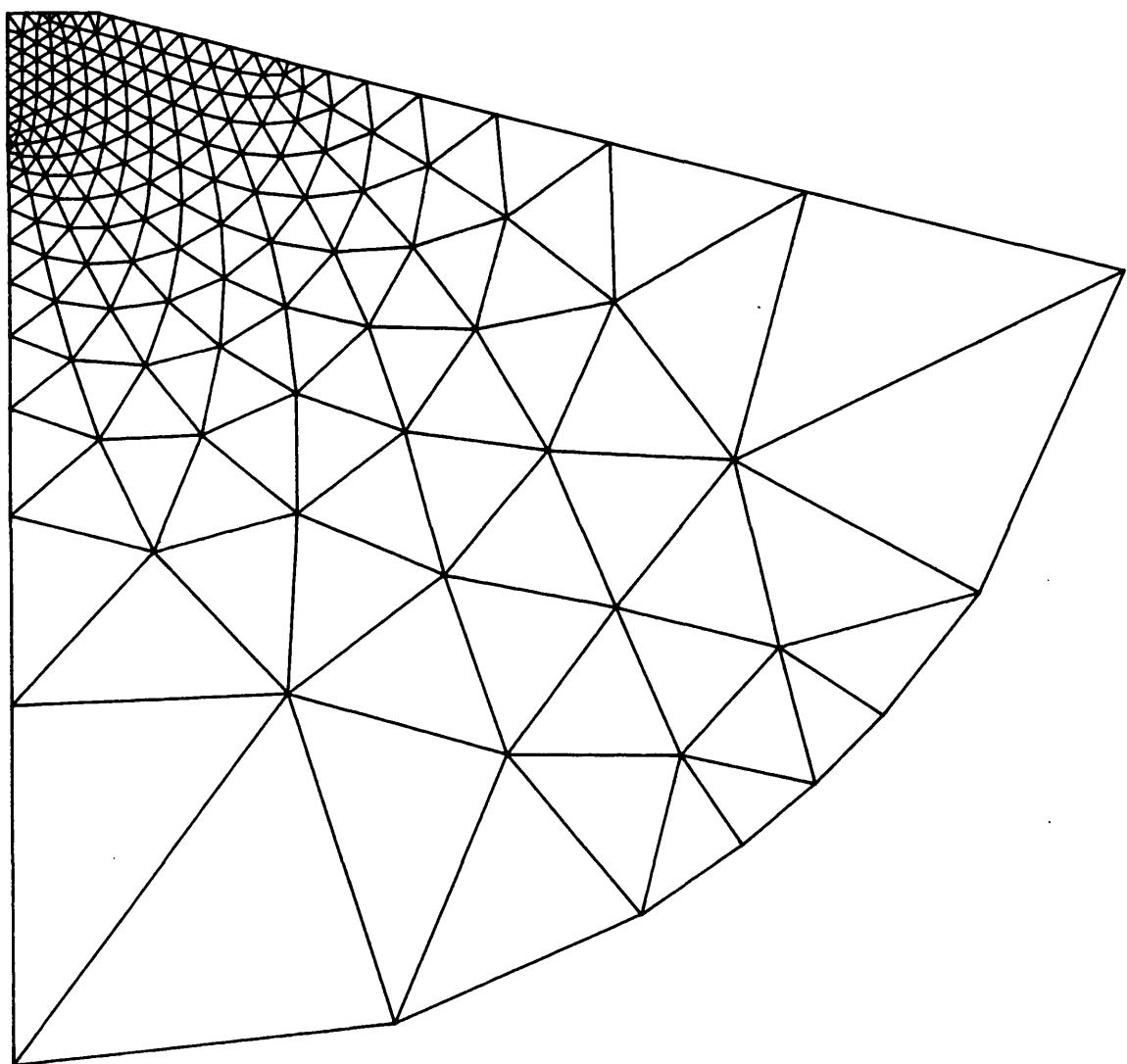
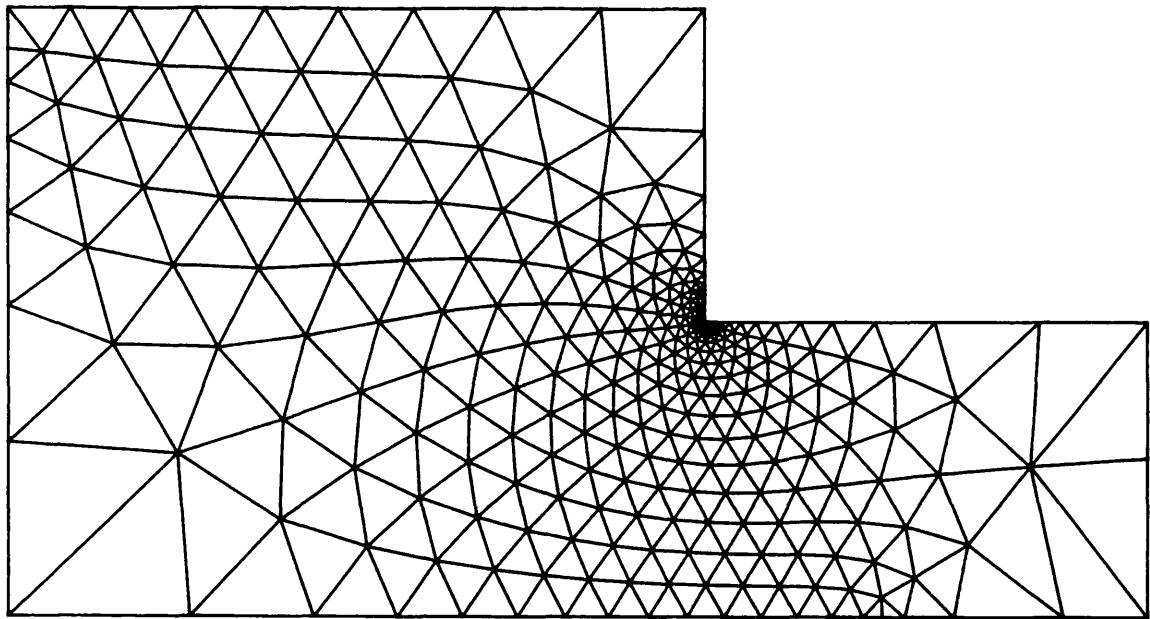
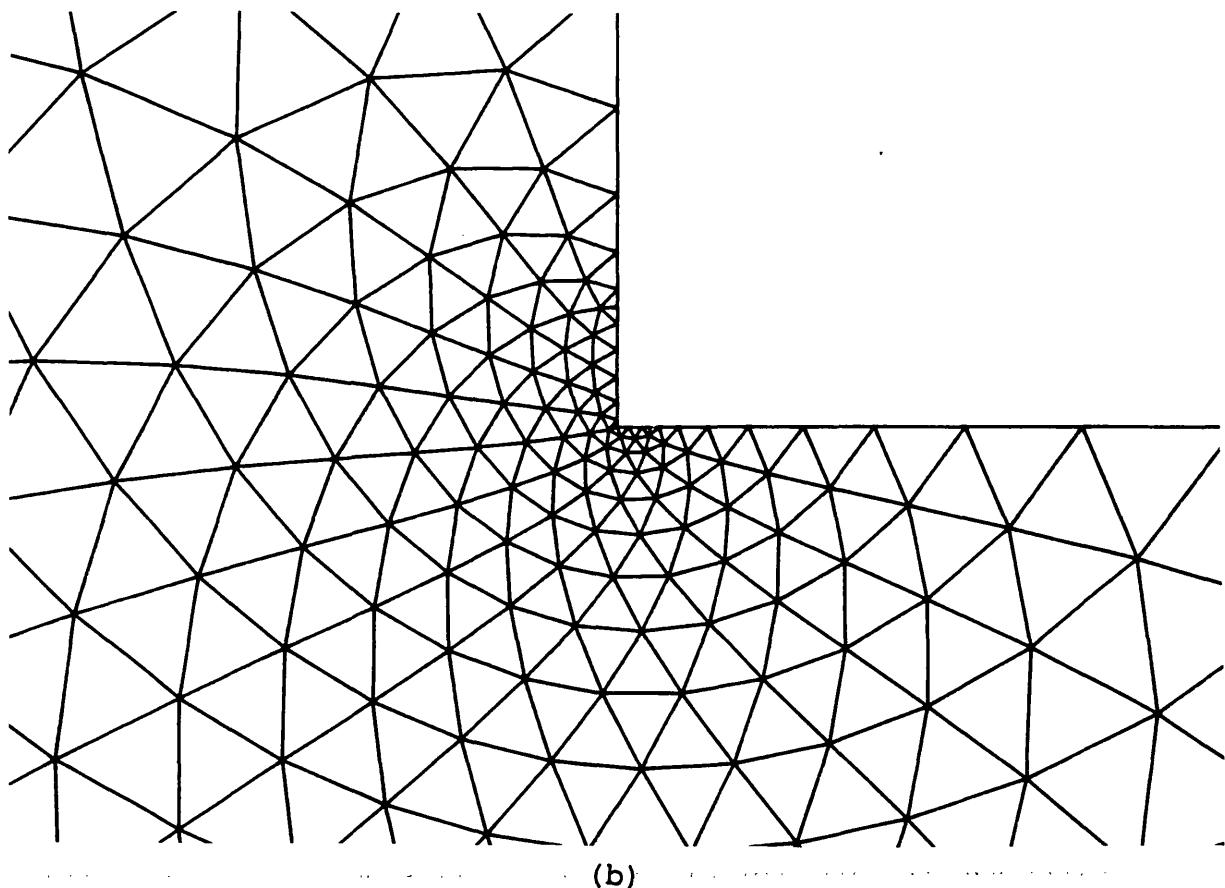


Figure 5-16. Mesh generated in a region typical of those analysed in ice studies described by Ponter and Brown [23] where an indentor is pressing down on the upper flat section.



(a)



(b)

Figure 5-17. (a) an example mesh in a region studied by Zienkiewicz, Liu and Huang [24] for plane extrusion; (b) detail.

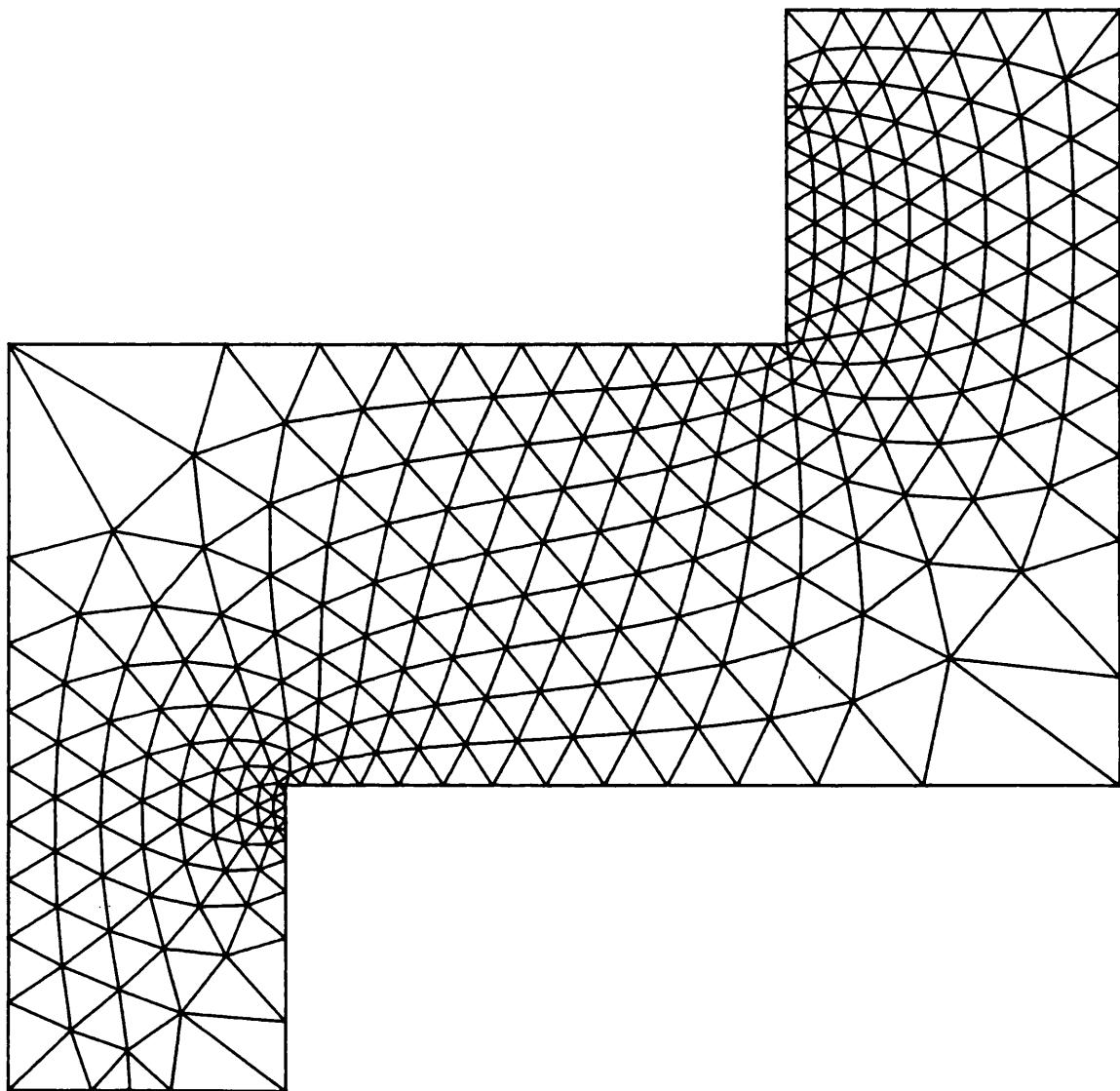


Figure 5-18. An example mesh in a region studied by Zienkiewicz, Liu and Huang [24] for axi-symmetric extrusion.

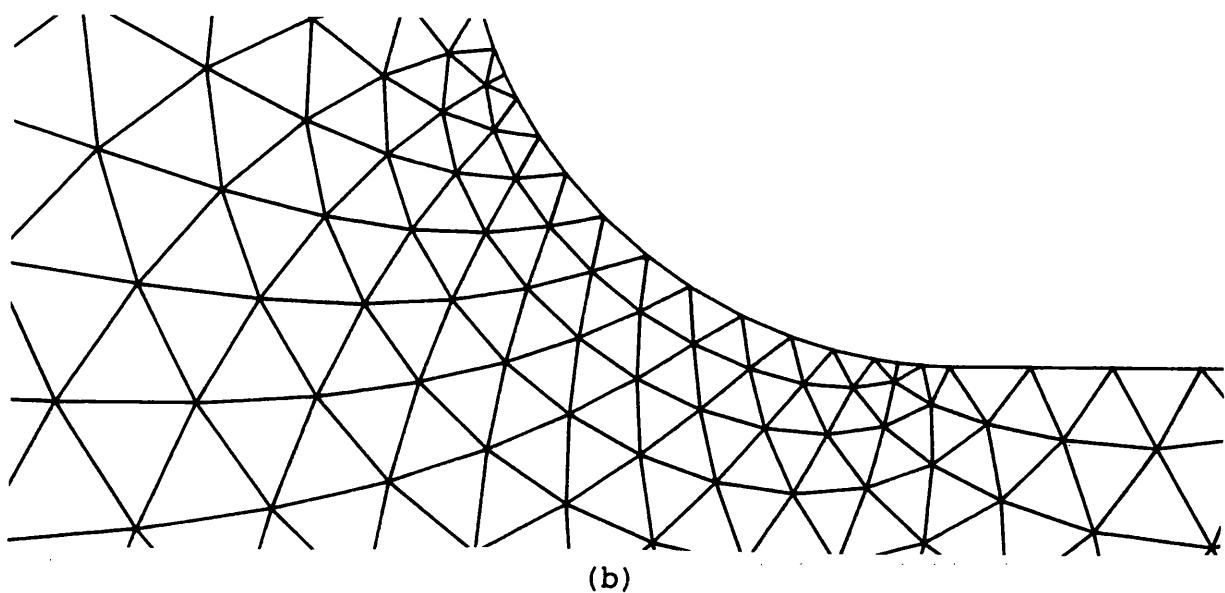
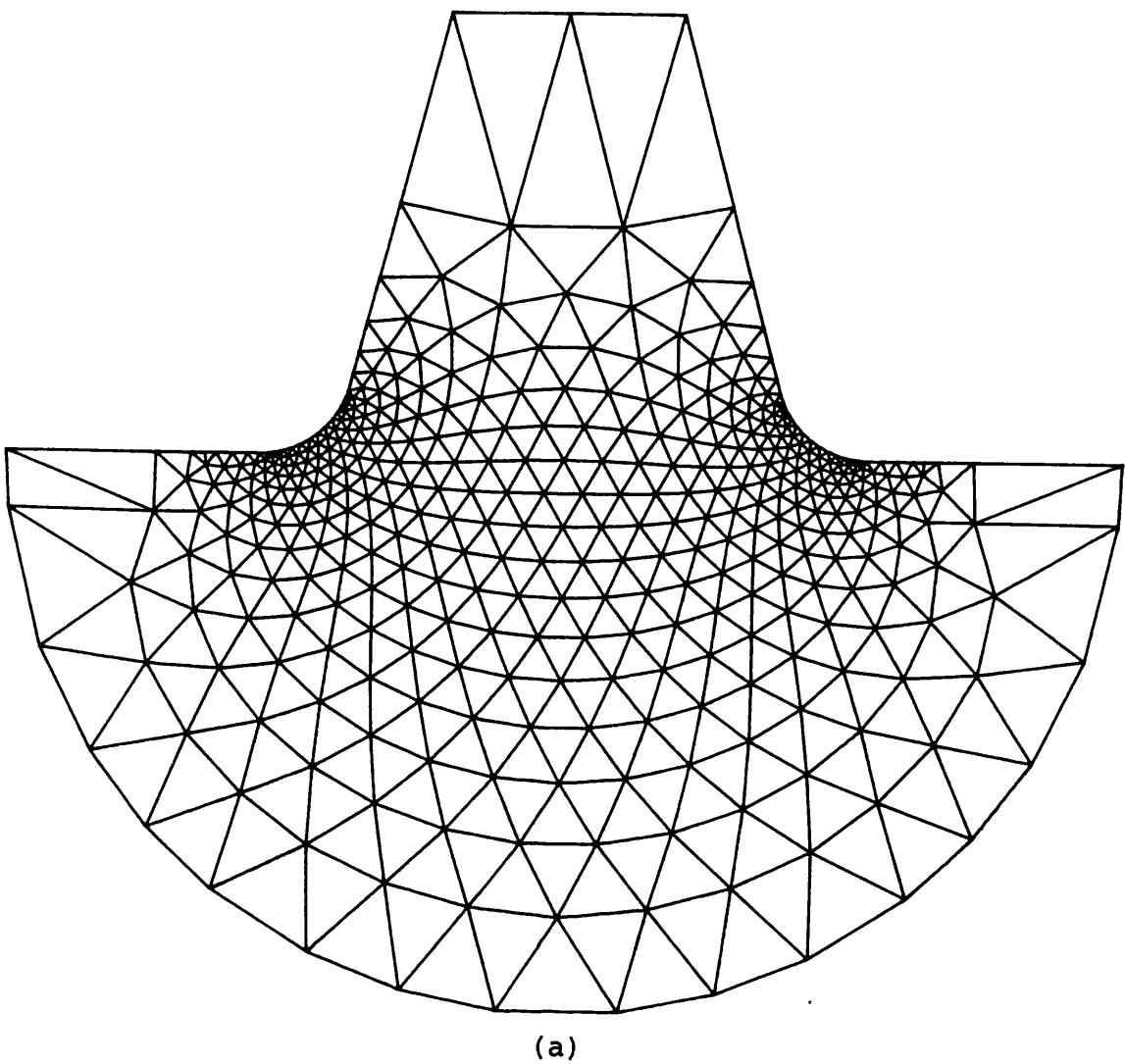
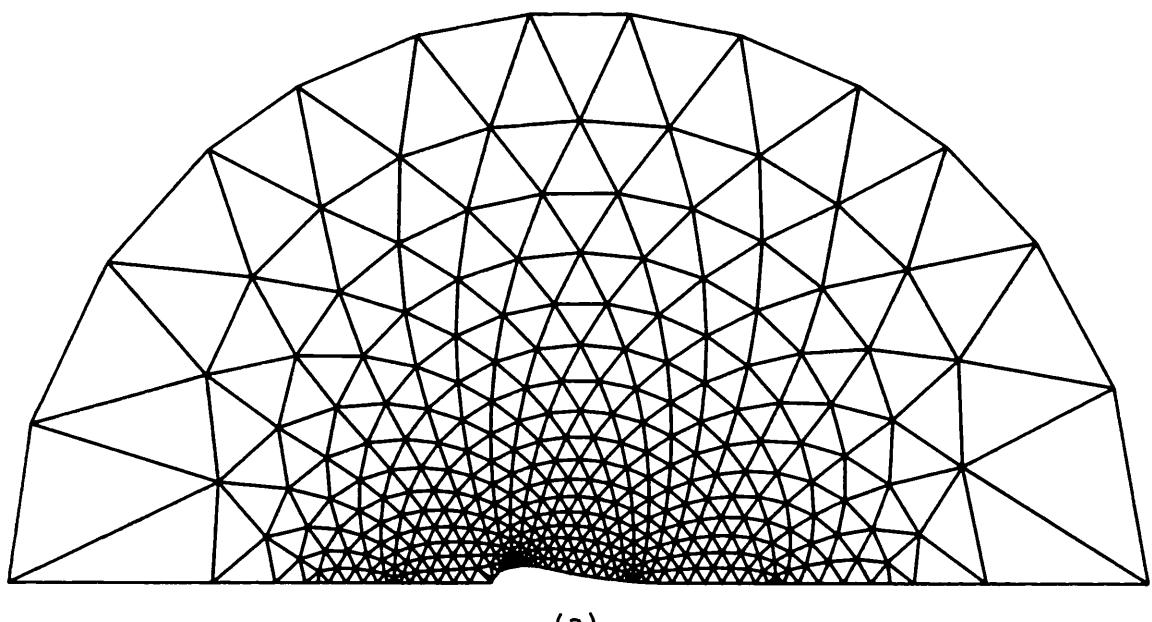
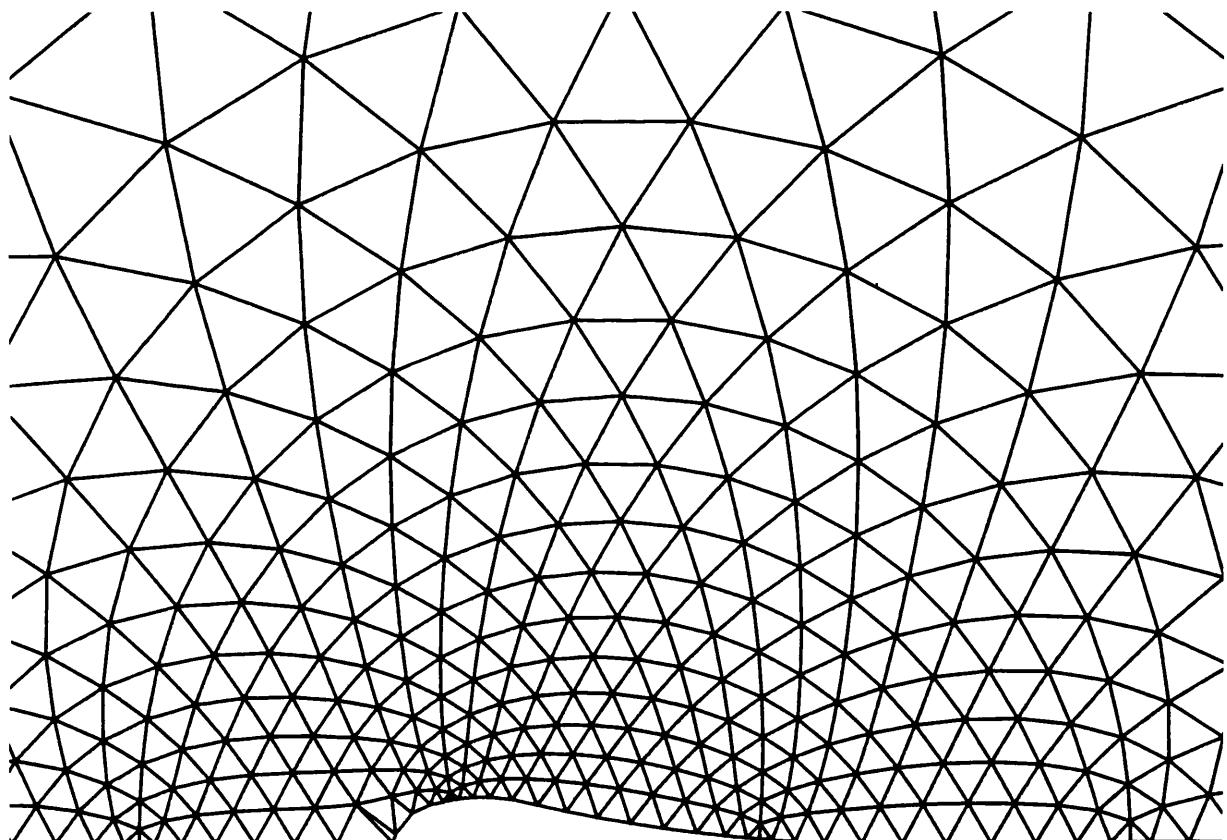


Figure 5-19. (a) an example mesh generated in a rack tooth region analysed by Yokoyama [25]; (b) detail.



(a)



(b)

Figure 5-20. (a) an example mesh for the modelling of incompressible fluid flow past an aerofoil [26]; (b) detail.

CHAPTER 6

Conclusions and further work

6.1 Conclusions

This thesis began with a survey of existing methods of mesh generation for two-dimensional regions. In the "patch" mapping methods, the boundary lines between patches are fixed and there is often an abrupt change in element size across these boundaries. In the triangulation methods, in order to achieve different nodal densities, the domain where the mesh is to be generated has first to be sub-divided into a disjoint union of sub-regions, the union of which covers the domain. This introduces inner boundaries over the area of the domain over which element density can change abruptly, even after several passes of the smoothing process have been applied.

Throughout all reported work, researchers place an emphasis on producing meshes which have well shaped elements but in none of the existing methods is there a guarantee that the change from small to large element should be gradual. This is an important consideration too and a new method has been presented which attempts to remedy this deficiency.

Because a single continuously differentiable transformation is used to map onto the whole of the target domain, the change of element size from small to large is sure to be gradual. The map also has the conformal property which ensures that the good element shape is retained, except possibly in regions adjacent to boundaries where the largest elements are located. In order that the method is computationally efficient, the Schwarz-Christoffel transformations which define the composite map are evaluated by a very efficient process. However, the

mesh generation computer program is not designed to be run interactively: all that is specified are the shape of the boundary and the desired arrangement of the boundary nodes - a bare minimum of information - and no interaction is needed. Example meshes illustrate that in many cases, good meshes may be generated efficiently. Although all of these meshes are composed of three-noded triangular elements, this is not a restriction of the general method: it is merely that in order to present examples, the selected polygon filled with ideally shaped elements was selected to contain triangles.

It is probably true that all workers in this field would agree that no single method can be used for all cases. This is certainly true of the method described in this thesis because only simply-connected two-dimensional domains may be considered. However it is hoped that the several example meshes have demonstrated that complicated geometries can be covered and that the original aims have been fulfilled.

6.2 Further work

Currently the computer program which generates finite element meshes using the Schwarz-Christoffel transformation can handle only three-noded triangular elements and it does this by using as the set of polygons Q , $2\pi/3$ hexagons. It would be more general if the library of available polygons could be extended to include those with interior angles of $4\pi/3$ and $5\pi/3$ in addition, as discussed in section 5.1. A classification scheme for the available polygons would be a useful addition to the mesh generation method. Although the way the program works at the moment - finding all those polygons Q of a given perimeter p and running through all of them (or a representative sample for large p) to

minimise a penalty function - is effective, an improvement would be to devise an iterative scheme so as to move towards that Q which produced the smallest value of the penalty function in a systematic way, rather than find it merely by a number of trials.

A second extension would be to allow quadrilateral in addition to triangular elements. Extending the program beyond three-noded triangles and four-noded quadrilaterals would not, then, be a significant task.

APPENDIX 1

Paper "A non-interactive method for the automatic generation of finite element meshes using the Schwarz-Christoffel transformation" by P.R. Brown, from Computer Methods in Applied Mechanics and Engineering, volume 25, pp 101-126 (1981). Reproduced by kind permission of the publishers.

A NON-INTERACTIVE METHOD FOR THE AUTOMATIC GENERATION OF FINITE ELEMENT MESHES USING THE SCHWARZ-CHRISTOFFEL TRANSFORMATION

P.R. BROWN

Dept. of Engineering, The University of Leicester, Leicester LE1 7RH, England

Received 18 February 1980

This paper describes a particular conformal mapping, the Schwarz-Christoffel transformation. A practical method for evaluating the parameters defining this transformation is described and two such transformations are used to construct a mapping between any two polygons. The desirable properties of a finite element mesh are stated and a method is described which attempts to generate such a mesh in any simply-connected region, using conformal mappings. This computer method is non-interactive and attempts to produce a good mesh with the minimum of input data. Some examples are included.

1. Introduction

The method of finite element analysis is a well-known one in mechanics since the use of computers has become widespread. It has meant that the stress distribution in a complicated structure can be analysed theoretically by modelling the structure as a number of elements of finite size in each of which a simple mechanical law is held to be true. Areas in a structure where stresses are changing rapidly can be modelled by many small elements but economy of computer program size and central processor time prohibit finite element meshes from having equally small elements everywhere. Areas where stresses are expected to be spatially constant may be modelled by larger elements and it is this gradual transition from small to large elements which is difficult to achieve, even in a hand-drawn mesh.

Drawing by hand is one way of defining a mesh and this is clearly the most flexible method, albeit a time-consuming one. The use of a digitising table to quantify the information in a drawing can, however, lessen the burden.

A second method of defining a finite element mesh is to use an automatic process as this can save much time. Interaction with a computer program is valuable here because, after a certain amount of experience has been gained, one can judge by eye whether or not a mesh will be a good model for a structure. One such interactive program is the PAMGEN system developed by the Computer Aided Design Group at Leicester University [1]. To use this system, the two-dimensional region modelling the structure must first be divided into a number of four-sided "patches". Each patch is then filled with any desired number of elements and the program ensures that adjacent patches are compatible. The mesh may be modified and refined until the most satisfactory one is obtained, although the fact that the edges of the patches are fixed can mean an abrupt change in element size from one patch to the next.

The disadvantage of an interactive mesh generation system is that the more flexible it is, the longer it takes to learn how to use it. Often an engineer, with a limited amount of time at his disposal, finds he is spending proportionately less time studying the results of an analysis program than he would like.

The method described here attempts to solve these problems by using conformal transformations to generate a mesh in any region which may be modelled by a polygon. The nature of such transformations ensures that the change in element size is gradual except possibly near part of the boundary of the region where stress gradients are expected to be lowest.

The system described is non-interactive and attempts to provide a good finite element mesh with the minimum of input data.

The first sections of this paper describe the numerical procedure by which the parameters defining the conformal mappings are found. The later sections state the desirable properties of a finite element mesh and show how, using conformal mappings, these are achieved. Some examples of the method are included.

2. Conformal mappings

The use of conformal mappings is a well-known technique in the solution of boundary value problems and in fluid mechanics. One particular conformal mapping is examined in this paper, the Schwarz–Christoffel transformation which maps the upper half-plane onto the interior of a general polygon. The mapping has the property that angles are preserved – that is the angle of intersection of any two smooth curves through all but a finite number of points in one region is equal, in both magnitude and sense of rotation, to the angle of intersection of the image curves in the mapped region. The parameters which define this mapping are presented and a practical method is described which evaluates these parameters numerically. The efficiency of the method is carefully considered as a system of non-linear simultaneous equations needs to be solved. The number of unknowns in this system of equations is kept to a minimum.

Most of the applications of the Schwarz–Christoffel theorem in fluid mechanics are concerned with degenerate polygons with two or more sides extending to infinity [2]. It is envisaged, however, that the method herein described would not have to be extended very far to be capable of finding transformations which would solve such problems.

A direct application of the present work is, however, in the field of structural mechanics. This is concerned with the automatic generation of a finite element mesh in any simply-connected two-dimensional domain. Two Schwarz–Christoffel transformations are used, the composite mapping thus obtained being between two general polygons with the upper half-plane used as a convenient intermediate region. The angle-preserving nature of this conformal mapping will be shown to be an important tool in the generation of a finite element mesh.

3. Schwarz–Christoffel transformation

One particular complex transformation is now considered, that which maps the upper half complex ζ -plane ($\zeta = \xi + i\eta$) onto the interior of a polygon. If the function which performs this

transformation is continuous, then the real ζ -axis will be mapped onto the boundary of the polygon. It will be assumed this is the case.

Consider an $(n+1)$ -sided polygon in complex w -space. The problem of finding the transformation which maps the upper half-plane onto the interior of a triangle will later be shown to be trivially soluble. It will be assumed, then, that n is not less than 3. Let the vertices of the polygon be represented by complex numbers w_0, w_1, \dots, w_n and let the interior angles at the vertices w_1, w_2, \dots, w_n be $\alpha_1, \alpha_2, \dots, \alpha_n$. Suppose the transformation which maps the polygon into the upper half-plane is $\zeta = f(w)$. Then let the images of w_1, w_2, \dots, w_n be $\xi_1, \xi_2, \dots, \xi_n$ and let the image of w_0 be ∞ . Without loss of generality suppose that $\xi_1 < \xi_2 < \dots < \xi_n$.

Let the $n+1$ sides of the polygon be $\Gamma_0, \Gamma_1, \dots, \Gamma_n$ as shown in fig. 1. Their images are denoted by $\Gamma'_0, \Gamma'_1, \dots, \Gamma'_n$ (see fig. 2).

It can readily be shown [2] that the Schwarz–Christoffel formula

$$w = F(\zeta) = a \int_{\zeta_0}^{\zeta} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \xi_k)^{1-\alpha_k/\pi}} + b$$

is the relationship between ζ and w by which this conformal mapping is achieved. a, b and ζ_0 are constants with $\text{Im } \zeta_0 \geq 0$. Once ζ_0 has been chosen (this may be done arbitrarily), the choice of b determines the position of the polygon in w -space. The value of a controls the size of the polygon by $|a|$ and its orientation by $\arg a$. The real power k of a complex number ζ is defined uniquely here by

$$\zeta^k = \exp[k \ln \zeta].$$

$\ln \zeta$ is the principal logarithm of ζ , that is

$$\ln \zeta = \ln |\zeta| + i \arg \zeta, \quad -\pi < \arg \zeta \leq \pi.$$

Although F is not conformal at the points ξ_k , it is well-defined there, albeit by a singular integral.

The Schwarz–Christoffel transformation maps the upper half ζ -plane onto the interior of an

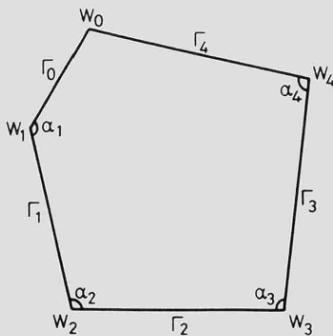


Fig. 1. w -plane.

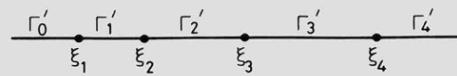


Fig. 2. ζ -plane.

$(n+1)$ -sided polygon. The points $\infty, \xi_1, \xi_2, \dots, \xi_n$ map onto the vertices of the polygon w_0, w_1, \dots, w_n respectively. The length of side w_j to w_{j+1} is, then

$$|F(\xi_{j+1}) - F(\xi_j)| = \left| a \int_{\xi_j}^{\xi_{j+1}} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \xi_k)^{1-\alpha_k/\pi}} \right|.$$

Since for all values of k

$$1 - \alpha_k/\pi < 1,$$

improper integrals such as the one above are well-defined and are no more difficult to evaluate numerically than real integrals like

$$\int_0^1 x^{-k} dx, \quad k < 1.$$

4. Numerical evaluation of ξ_k, a, b

A numerical procedure is now described which shows how to find the quantities ξ_k, a, b which define the Schwarz–Christoffel transformation.

Given a polygon P with its $n+1$ distinct vertices represented as complex numbers in the w -plane, the problem of finding transformation $w = F(\zeta)$ is as follows:

(1) Choose one vertex of P , w_0 , to be the image of ∞ under F . The angle at w_0 must not be π . Denote the remaining vertices by w_1, w_2, \dots, w_n ordered in a counter-clockwise direction and the interior angles at these vertices by $\alpha_1, \alpha_2, \dots, \alpha_n$, respectively.

(2) Arbitrarily fix ζ_0 as a complex quantity satisfying $\text{Im } \zeta_0 \geq 0$.

Rather than attempt to solve directly a set of equations in the n unknowns ξ_i and two complex quantities a and b , it is easier to proceed in two steps as follows. Let ξ be regarded as a variable n -dimensional vector whose components satisfy $\xi_1 < \xi_2 < \dots < \xi_n$.

Then let F_1 be defined as a function of ζ and ξ by

$$F_1(\zeta, \xi) = \int_{\zeta_0}^{\zeta} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \xi_k)^{1-\alpha_k/\pi}}.$$

(3) Find the particular ξ such that the transformation $w = F_1(\zeta, \xi)$ maps the upper half ζ -plane into the interior of a polygon P' similar to P . Polygons P and P' are similar if the ratio of the lengths of sides of P to corresponding sides of P' is constant and if corresponding interior angles are equal.

(4) With this value of ξ , choose a and b such that the transformation

$$w = F(\zeta) = a \int_{\zeta_0}^{\zeta} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \xi_k)^{1-\alpha_k/\pi}} + b$$

maps the upper half ζ -plane onto P . Having found the ξ_i points, it is sufficient to define a, b so that two ξ_i should be mapped onto corresponding w_i points.

The choice of transformation in step (3) ensures that, given any vector ξ whose components satisfy $\xi_1 < \xi_2 < \dots < \xi_n$, the polygon with vertices at the points $F_1(\infty, \xi), F_1(\xi_1, \xi), \dots, F_1(\xi_n, \xi)$ will have the required interior angles, as has already been shown. Finding the particular ξ so that the resulting polygon is similar to P is more difficult however. Indeed, even for the case of a general rectangle, ξ_i can be expressed only in terms of elliptic functions. For a general polygon, ξ must be found numerically and for this reason, quantities are introduced here which describe P uniquely.

An $(n+1)$ -sided polygon (fig. 3) may be described by its $n+1$ ordered vertices in some cartesian system – $2(n+1)$ independent variables. It is convenient here to describe a polygon by a different set of $2(n+1)$ variables. The quantities chosen are

- n interior angles $\alpha_1, \alpha_2, \dots, \alpha_n$ at the vertices P_1, P_2, \dots, P_n ;

- the coordinates of two arbitrarily chosen vertices;

- $n-2$ quantities $\beta_2, \beta_3, \dots, \beta_{n-1}$ to describe the relative lengths of the sides of the polygon.

If P_0, P_1, \dots, P_n are the vertices of the polygon as shown in fig. 3, β_j is defined by

$$\beta_j = \frac{P_1 P_2 + P_2 P_3 + \dots + P_{j-1} P_j}{P_1 P_2 + P_2 P_3 + \dots + P_{n-1} P_n}.$$

As an example a regular hexagon would be described by the coordinates of any two of its vertices together with the following eight quantities

$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = \frac{2}{3}\pi;$$

$$\beta_2 = \frac{1}{4}, \quad \beta_3 = \frac{1}{2}, \quad \beta_4 = \frac{3}{4}.$$

The numbers $\alpha_2, \dots, \alpha_{n-1}, \beta_2, \dots, \beta_{n-1}$ describe the relative position of the vertices P_1, \dots, P_n . α_1 and α_n define the remaining two edges and so the remaining vertex P_0 at their intersection. The coordinates of any two vertices fix the polygon in cartesian space.

If the points P_0, P_1, \dots, P_n are represented by complex numbers w_0, w_1, \dots, w_n , then the

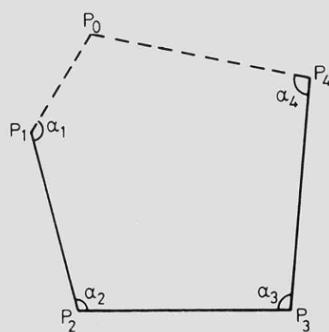


Fig. 3. General polygon fully defined by the coordinates of two vertices, interior angles α_i and length ratios β_j .

definition of β_j becomes

$$\beta_j = \sum_{k=2}^j |w_k - w_{k-1}| / \sum_{k=2}^n |w_k - w_{k-1}|$$

for $j = 2, 3, \dots, n-1$ so that $0 < \beta_2 < \beta_3 < \dots < \beta_{n-1} < 1$. β will be written for the $(n-2)$ -dimensional vector

$$\begin{bmatrix} \beta_2 \\ \vdots \\ \beta_{n-1} \end{bmatrix}.$$

Now suppose that, for any vector ξ , its components satisfying $\xi_1 < \xi_2 < \dots < \xi_n$, w'_j is defined by

$$w'_j = F_j(\xi_j, \xi) = \int_{\xi_0}^{\xi_j} \frac{d\xi}{\prod_{k=1}^n (\zeta - \xi_k)^{1-\alpha_k/\pi}}$$

for $j = 1, 2, \dots, n$ and γ_j by

$$\gamma_j = \sum_{k=2}^j |w'_k - w'_{k-1}| / \sum_{k=2}^n |w'_k - w'_{k-1}|$$

for $j = 2, 3, \dots, n-1$. Let the vector-valued function γ be defined by

$$\gamma(\xi) = \begin{bmatrix} \gamma_2 \\ \gamma_3 \\ \vdots \\ \gamma_{n-1} \end{bmatrix}.$$

The problem of finding ξ such that the transformation maps the upper half ζ -plane onto the interior of a polygon similar to P is equivalent to solving the vector equation

$$\gamma(\xi) = \beta.$$

Being a mapping from the space of n -dimensional vectors R^n to the space of $(n-2)$ -dimensional vectors R^{n-2} , γ is clearly many-one. It is proved in the Appendix that if

$$\xi'_j = p\xi_j + q$$

for any real scalars p, q with p positive, then

$$\gamma(\xi') = \gamma(\xi)$$

so that the solution to the equation $\gamma(\xi) = \beta$ is unique only up to a linear transformation on ξ .

Thus, in order to solve the equation uniquely, the domain of ξ must be restricted so that ξ has only $n - 2$ degrees of freedom. This is done by fixing two components of ξ at constant values – arbitrarily ξ_1 and ξ_n are set to -1 and 1 respectively. Let vector χ denote this constrained ξ , that is,

$$\chi = \begin{bmatrix} \chi_1 \\ \chi_2 \\ \vdots \\ \chi_{n-1} \\ \chi_n \end{bmatrix} = \begin{bmatrix} -1 \\ \chi_2 \\ \vdots \\ \chi_{n-1} \\ 1 \end{bmatrix}$$

where the χ_i satisfy $-1 = \chi_1 < \chi_2 < \dots < \chi_{n-1} < \chi_n = 1$. The vector equation to be solved is now symbolically

$$\gamma(\chi) = \beta$$

or

$$\begin{aligned} \gamma_2(\chi_1, \chi_2, \dots, \chi_n) &= \beta_2 \\ &\vdots \\ \gamma_{n-1}(\chi_1, \chi_2, \dots, \chi_n) &= \beta_{n-1} \end{aligned}$$

where γ is a mapping from a subset of R^n to R^{n-2} . Having found the solution to this equation, χ , the general solution is ξ where the components of ξ satisfy

$$\xi_j = p\chi_j + q \quad (p > 0).$$

5. Numerical solution of the equation $\gamma(\chi) = \beta$

The system of equations is solved numerically by a Newton–Raphson iterative method which, given an estimate of the solution, predicts a better estimate. Given a current estimate χ , the non-zero values of the predicted increment

$$\delta\chi = \begin{bmatrix} 0 \\ \delta\chi_2 \\ \vdots \\ \delta\chi_{n-1} \\ 0 \end{bmatrix}$$

to be added to χ are the solutions of the set of $n - 2$ equations

$$\begin{aligned} \frac{\partial\gamma_2}{\partial\chi_2}\delta\chi_2 + \frac{\partial\gamma_2}{\partial\chi_3}\delta\chi_3 + \dots + \frac{\partial\gamma_2}{\partial\chi_{n-1}}\delta\chi_{n-1} &= \beta_2 - \gamma_2(\chi) \\ &\vdots \\ \frac{\partial\gamma_{n-1}}{\partial\chi_2}\delta\chi_2 + \frac{\partial\gamma_{n-1}}{\partial\chi_3}\delta\chi_3 + \dots + \frac{\partial\gamma_{n-1}}{\partial\chi_{n-1}}\delta\chi_{n-1} &= \beta_{n-1} - \gamma_{n-1}(\chi). \end{aligned}$$

As an initial estimate to the solution, χ is defined by

$$\chi_1 = -1, \quad \chi_n = 1, \quad \chi_j = 2\beta_j - 1 \quad \text{for } j = 2, 3, \dots, n-1.$$

Since the domain of function γ is only a subset of R^n , the incremented vector must not be allowed to assume a value outside the domain – it may be that for some i, j with $\chi_i < \chi_j$, the calculated increments $\delta\chi_i, \delta\chi_j$ are such that

$$\chi_i + \delta\chi_i \geq \chi_j + \delta\chi_j.$$

For this reason, for every $i < j$, a quantity λ_{ij} is introduced which is a measure of by how much the incremented χ values overlap. λ_{ij} is defined by

$$\lambda_{ij} = \frac{\chi_j - \chi_i}{\delta\chi_i - \delta\chi_j} \quad \text{for all } i < j, \quad i, j = 1, 2, \dots, n.$$

For simplicity of notation, $\delta\chi_1$ and $\delta\chi_n$ are assumed to be zero. If, for some i and j , λ_{ij} is positive and not greater than unity this indicates that

$$\chi_i + \delta\chi_i \geq \chi_j + \delta\chi_j.$$

If one or more of the λ_{ij} is positive and not greater than unity, the addition of the full Newton–Raphson increment $\delta\chi$ to χ would send the predicted estimate outside the γ -domain. If this is the case, $\delta\chi$ is weighted by an amount W – that is, χ is incremented by an amount $W \delta\chi$ where W is defined by

$$W = \theta \min\{\lambda_{ij} : \lambda_{ij} > 0\}.$$

θ is a positive constant < 1 , typically 0.9.

The Newton–Raphson iterative procedure is repeated until $\gamma(\chi)$, the vector-valued function of the latest value of χ , differs from β by a small amount, that is until $\max_k |\gamma_k - \beta_k|$ is sufficiently small.

For completeness, the expressions for $\partial\gamma_k/\partial\chi_j$ are included here. If v_l is written for the expression

$$\int_{\zeta_0}^{\chi_l} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \chi_k)^{1-\alpha_k/\pi}},$$

then $\partial\gamma_k/\partial\chi_j$ is given by

$$\begin{aligned} \frac{\partial\gamma_k}{\partial\chi_j} = & \left(\sum_{l=2}^n |v_l - v_{l-1}| \sum_{l=2}^k \frac{\operatorname{Re} \left\{ (v_l - v_{l-1}) \left[\left(\frac{\partial v_l}{\partial\chi_j} \right)^* - \left(\frac{\partial v_{l-1}}{\partial\chi_j} \right)^* \right] \right\}}{|v_l - v_{l-1}|} \right. \\ & \left. - \sum_{l=2}^k |v_l - v_{l-1}| \sum_{l=2}^n \frac{\operatorname{Re} \left\{ (v_l - v_{l-1}) \left[\left(\frac{\partial v_l}{\partial\chi_j} \right)^* - \left(\frac{\partial v_{l-1}}{\partial\chi_j} \right)^* \right] \right\}}{|v_l - v_{l-1}|} \right) \Bigg/ \left[\sum_{l=2}^n |v_l - v_{l-1}| \right]^2 \end{aligned}$$

where * denotes complex conjugate.

When $l \neq j$, $\partial v_l / \partial \chi_j$ is given by

$$\frac{\partial v_l}{\partial \chi_j} = \int_{\zeta_0}^{\chi_l} \frac{(1 - \alpha_j/\pi) d\zeta}{(\zeta - \chi_j) \prod_{k=1}^n (\zeta - \chi_k)^{1-\alpha_k/\pi}}.$$

Otherwise,

$$\frac{\partial v_l}{\partial \chi_j} = \frac{1}{\prod_{k=1}^n (\zeta_0 - \chi_k)^{1-\alpha_k/\pi}} - \int_{\zeta_0}^{\chi_l} \frac{\sum_{k=1, k \neq j}^n \frac{1 - \alpha_k/\pi}{(\zeta - \chi_k)}}{\prod_{k=1}^n (\zeta - \chi_k)^{1-\alpha_k/\pi}} d\zeta.$$

6. Numerical evaluation of the integrals

The method used to evaluate the complex integrals associated with the Schwarz–Christoffel formula is that of Gaussian quadrature. As well as being more accurate than the more straightforward Newton–Cotes methods for the same amount of work done, a formula like that of Gauss is essential in evaluating improper integrals where the integrand is infinite at the endpoints of the path of integration. One of the properties of the integral of an analytic function between two points is that it is independent of the path joining the points. The path of integration used by the Gaussian quadrature algorithm used here is the straight line between the specified endpoints. It is with the practical difficulties of evaluating the complex integrals borne in mind that the quantity ζ_0 , unspecified so far except that its imaginary part be non-negative, is chosen. It may be defined to lie on the real axis but this would unnecessarily introduce further singularities into some integrands. Although such integrals would be well defined and could be calculated, their evaluation would introduce unnecessary complications which would lead to the consumption of more computer time. It is appropriate to take $\text{Im } \zeta_0 > 0$ and since the χ_j quantities, always the upper limit of integration in solving the equations for χ_j , satisfy $-1 \leq \chi_j \leq 1$, the symmetrical choice of $\zeta_0 = i$ is made.

7. Complete specification of the Schwarz–Christoffel transformation

Once χ , the solution to the equation $\gamma(\chi) = \beta$, has been found, the general solution is ξ where the components satisfy $\xi_j = p\chi_j + q$ for any real scalars p, q with p positive. These two quantities, p and q , leave two degrees of freedom in the choice of mapping from the half-infinite plane to the interior of polygon P . Whatever the choice of p and q , the mapping is defined completely once the two quantities a and b have been found, the quantities which locate the scale polygon P' , similar to P , into P itself.

Suppose v_0, v_1, \dots, v_n are the complex numbers which represent the $n + 1$ vertices of polygon P' , that is,

$$v_0 = \int_{\zeta_0}^{\infty} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \xi_k)^{1-\alpha_k/\pi}}, \quad v_j = \int_{\zeta_0}^{\xi_j} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \xi_k)^{1-\alpha_k/\pi}} \quad \text{for } j = 1, 2, \dots, n.$$

In theory, a and b are defined by specifying that any two vertices of P' , say v_{k_1} and v_{k_2} are mapped into the corresponding vertices of P , w_{k_1} and w_{k_2} by $w_k = av_k + b$ for $k = k_1$ and k_2 . In practice, because of numerical errors in finding χ and in evaluating the definite integrals, the vertices v_k are not exactly the vertices of a polygon similar to P . With this practical point in mind, the two numbers k_1 and k_2 are chosen so that w_{k_1} and w_{k_2} are the two vertices of P which are furthest apart. a and b are given by

$$a = \frac{w_{k_1} - w_{k_2}}{v_{k_1} - v_{k_2}}, \quad b = \frac{v_{k_1}w_{k_2} - v_{k_2}w_{k_1}}{v_{k_1} - v_{k_2}}.$$

8. Example

To exemplify the method described, the transformation $w = F(\zeta)$ which maps the upper half ζ -plane onto the interior of a regular hexagon will be found. The hexagon (fig. 4) has vertices at the points represented by complex numbers $-1 + i\sqrt{3}$, -2 , $-1 - i\sqrt{3}$, $1 - i\sqrt{3}$, 2 and $1 + i\sqrt{3}$. $-1 + i\sqrt{3}$ is to be the image of ∞ under F and w_0 is accordingly defined to be this point. w_1 , w_2 , w_3 , w_4 and w_5 are defined as the other vertices in counter-clockwise order. It is further required that the points -1 and $+1$ be mapped onto w_2 and w_4 respectively.

The 12 real quantities which define the hexagon in the w -plane are:

– the coordinates of the vertices w_1 and w_4 , i.e.

$$w_1 = -2, \quad w_4 = 2;$$

– the interior angles at w_1 , w_2 , w_3 , w_4 and w_5 , i.e.

$$\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = \frac{2}{3}\pi;$$

– the length ratios $\beta_2 = \frac{1}{4}$, $\beta_3 = \frac{1}{2}$, $\beta_4 = \frac{3}{4}$.

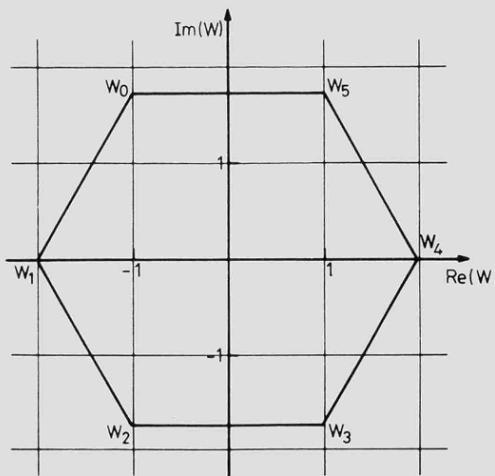


Fig. 4. Example polygon, a regular hexagon in the w -plane.

In solving the equation $\gamma(\chi) = \beta$, the initial values chosen for the components of χ are given by $\chi_1 = -1$, $\chi_5 = 1$ and for $j = 2, 3$ and 4 ,

$$x_j = 2\beta_j - 1,$$

i.e., $\chi_2 = -\frac{1}{2}$, $\chi_3 = 0$, $\chi_4 = \frac{1}{2}$. Now any χ can be said to generate a polygon, the vertices of which are

$$v_0 = \int_i^{\infty} \frac{d\zeta}{\prod_{k=1}^5 (\zeta - \chi_k)^{1/3}},$$

$$v_j = \int_i^{\chi_j} \frac{d\zeta}{\prod_{k=1}^5 (\zeta - \chi_k)^{1/3}} \quad \text{for } j = 1, 2, 3, 4 \text{ and } 5.$$

The polygon generated by this first χ is shown in fig. 5. The method necessitates that the angles of this polygon are all $\frac{2}{3}\pi$ but the ratios of the lengths of the sides are incorrect. In fact for this χ

$$\gamma(\chi) = \begin{bmatrix} 0.22 \\ 0.50 \\ 0.78 \end{bmatrix}.$$

After the system of equations for the 3 increments $\delta\chi_2$, $\delta\chi_3$ and $\delta\chi_4$ has been solved, the new value of χ is

$$\begin{bmatrix} -1 \\ -0.33 \\ 0 \\ 0.33 \\ 1 \end{bmatrix},$$

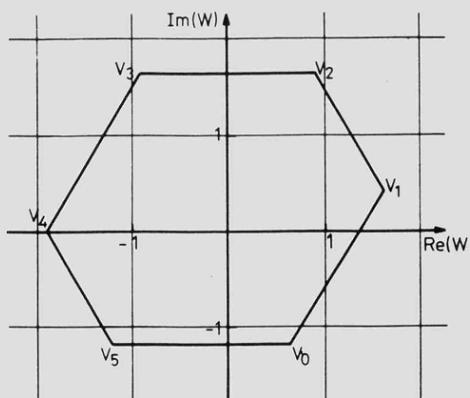


Fig. 5. Polygon generated by the initial choice of χ .

and the polygon generated by this χ is shown in fig. 6. For this second χ ,

$$\gamma(\chi) = \begin{bmatrix} 0.25 \\ 0.50 \\ 0.25 \end{bmatrix}.$$

Thus after only one iteration in this particular example, the equation has been solved. The transformation maps ∞ into the point v_0 in fig. 6, -1 into v_1 , -0.33 into v_2 , 0 into v_3 , 0.33 into v_4 and 1 into v_5 .

The stipulation that -1 and 1 be mapped into w_2 and w_4 defines ξ_2 as -1 and ξ_4 as $+1$ making the values of p and q to be 3.0 and 0.0 respectively. The polygon generated by the vector

$$\xi = \begin{bmatrix} -3.0 \\ -1.0 \\ 0 \\ 1.0 \\ 3.0 \end{bmatrix}$$

is shown in fig. 7, the six vertices being at the points

$$\begin{aligned} v_0 &= 0.54 - 0.94i, & v_1 &= 0.99 - 0.18i, & v_2 &= 0.54 + 0.58i, \\ v_3 &= -0.33 + 0.58i, & v_4 &= -0.77 - 0.18i, & v_5 &= -0.33 - 0.94i. \end{aligned}$$

Finally the values of a and b , the constants which rotate, magnify and position this hexagon into the required hexagon are found by solving the equations

$$av_2 + b = w_2, \quad av_4 + b = w_4,$$

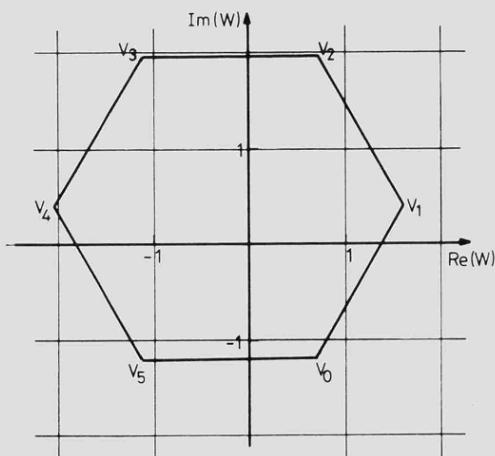


Fig. 6. Polygon generated by the second value of χ .

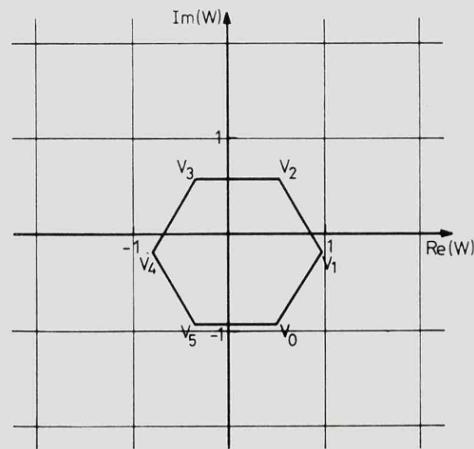


Fig. 7. Polygon generated by ξ .

i.e.,

$$a(0.54 + 0.58i) + b = -2, \quad a(-0.33 - 0.94i) + b = 2,$$

giving values of $a = -2.27$, $b = 0.25 - 0.41i$. Thus the required transformation is

$$w = F(\zeta) = -2.27 \int_i^{\zeta} \frac{d\zeta}{(\zeta + 3.0)^{1/3}(\zeta + 1.0)^{1/3}\zeta^{1/3}(\zeta - 1.0)^{1/3}(\zeta - 3.0)^{1/3}} + 0.25 - 0.41i.$$

This completes the sections which describe the mathematics behind the Schwarz–Christoffel transformation and a numerical method of finding the defining parameters. The remaining sections of the paper are a direct application of this work for which two such mappings are used to generate a finite element mesh in any simply-connected region which may be modelled by a polygon.

9. Desirable properties of a finite element mesh

9.1. Best element shape

Meshes for two-dimensional representations of engineering structures (and it is two-dimensional meshes only which will be considered here) are usually composed of quadrilateral or triangular elements. The best shaped element is that which will best model the finite change in dimensions of a structure put under load. Zienkiewicz [3] recommends that the best triangular element is equilateral and it is a small generalisation to state that the best general element shape is symmetrical. For example, the best quadrilateral element is square.

It is usually impossible to fill an irregularly-shaped region entirely with symmetrically-shaped elements but a mesh which contains elements as near to the ideal as possible is the most desirable.

9.2. Bandwidth

Once the mesh has been defined, the finite element method consists of the following operations. Boundary conditions are imposed on those nodes (that is the vertices of the elements) which are loaded or whose movement is constrained in some way. A system of linear simultaneous equations is then solved, the unknowns being the displacements at the nodal points [3]. From this displacement field, the strains in each element may be calculated and thence the stresses.

It is the solution of the set of linear equations which is at the centre of the method since it is this operation which takes most computer time. Anything which can make this solution more efficient is a worthwhile gain.

The square matrix of the system of linear equations is banded – that is, all elements of the matrix beyond a certain distance from the main diagonal are zero. By making the width of this band as small as possible, not only can the solution be obtained more quickly, but also less numbers need to be held in the computer's store, so reducing program size. What determines the bandwidth of the matrix is the way in which the nodes of the modelling mesh are

numbered. The bandwidth of the matrix is a function of the maximum of d_j where j ranges over the numbers of all the elements in the mesh. d_j is the maximum difference in the numbers of those nodes which define element j . The term "bandwidth of a mesh" will be used as a synonym for this maximum d_j . It is often difficult to number the nodes of a mesh in such a way that its bandwidth is as small as it might be. This is especially true when elements vary considerably in size.

9.3. Aims of this method

An automatic method of generating a mesh in a simply-connected region is now described which attempts to satisfy the following conditions:

- (1) Regions of greatest interest where the stresses are expected to vary most rapidly are modelled by the smallest elements.
- (2) The change of size of the elements is gradual throughout the region except possibly in places where the stresses are expected to be spatially constant.
- (3) The shape of the majority of elements is good, especially in regions of greatest interest.
- (4) The nodes are numbered in an efficient way to make the bandwidth of the mesh as small as possible for a given number of elements.

The method described attempts to accomplish these aims by generating a mesh in any region which may be modelled by a polygon. A second polygon is defined in which a regular mesh of any desired density, composed entirely of ideally-shaped elements, can easily be generated. Two conformal transformations are found which firstly map this polygon onto an intermediate region and thence onto the modelling polygon. There are two degrees of freedom in the choice of transformation which determine that the smallest elements be placed where highest stress gradients are expected.

It will be shown that the angle-preserving nature of the conformal mappings is a useful tool in determining the shape of elements in the generated mesh.

10. Generation of a finite element mesh in a general polygon

A practical method of finding a transformation from the upper half complex ζ -plane onto the interior of any $(n+1)$ -sided polygon P in the complex w -space has been described in earlier sections. This transformation

$$w = a \int_{\zeta_0}^{\zeta} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \xi_k)^{1-\alpha_k/\pi}} + b$$

is conformal everywhere except at ∞ and the points ξ_k , those points which map into the vertices, w_0, w_1, \dots, w_n , of polygon P . α_k is the interior angle at the vertex w_k .

Suppose R is a two-dimensional simply-connected region in which a finite element mesh is to be drawn. It is proposed that polygon P be so defined to model R . If the boundary of R is composed entirely of straight lines, this is readily achieved. If part of the boundary of R is a curved arc, this may be split into a suitable number of straight line segments.

A polygon Q situated in the complex z -plane is now defined. Q is chosen to be a polygon in which a good finite element mesh may easily be generated automatically, that is a mesh composed entirely of ideally-shaped elements. For example, if the mesh for P is to be composed of rectangles, Q may be defined as a rectangle and the mesh within Q may comprise any desired number of square elements. The nodes defining the mesh in Q can readily be numbered to minimise the mesh's bandwidth. This is easily achieved in a regular mesh.

Let the mapping

$$z = G(\zeta) = c \int_{\zeta_0}^{\zeta} g(\zeta) d\zeta + d$$

be the Schwarz-Christoffel transformation from the upper half ζ -plane onto the interior of Q . c and d are complex constants defining the location, magnification and orientation of Q in the z -plane. The quantities corresponding to p and q are chosen arbitrarily. The mappings F and G are represented pictorially in fig. 8. In this figure, the polygon Q is represented by the regular hexagon. Let the set S designate those ζ -points which map into the vertices of Q .

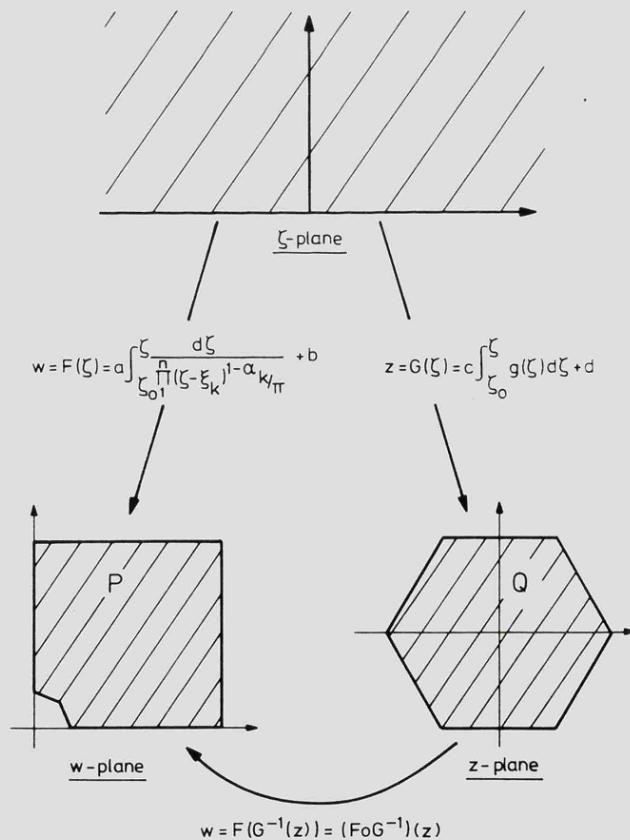


Fig. 8. Pictorial representation of the different mappings used.

Let the nodes of the mesh in Q be at the points in the z -plane represented by the complex numbers z_j . Those points ζ_j which map onto z_j under G may be found numerically by a Newton-Raphson procedure which, given an estimate for ζ_j , predicts a more accurate one. To find ζ_j , suppose the current estimate is ζ . It is supposed that z_j is not a vertex of Q since those ζ -points, elements of the set S , which map onto the vertices of Q are already known. Then a better estimate for ζ_j is $\zeta + \delta\zeta$, where $\delta\zeta$ is given by

$$\delta\zeta = z_j - \frac{G(\zeta)}{G'(\zeta)} = z_j - \frac{G(\zeta)}{cg(\zeta)}.$$

$g(\zeta)$ is defined since ζ is not an element of the set S . This procedure is repeated until the difference between z_j and $G(\zeta)$ is sufficiently small. Those ζ_j points which map onto the boundary of Q are real. Define these real ζ_j , n_Ω in number, to be the quantities Ω_k where $\Omega_1, \Omega_2, \dots, \Omega_{n_\Omega}$ are in ascending algebraic order.

Now suppose that, as well as finding those ζ_j which map onto z_j under G , the curves in ζ -space which map onto all the edges of the finite element mesh in Q are also found. Then, except at a finite number of ζ -points, those in S , the conformality of mapping G ensures that these curves intersect at the same angles as corresponding edges of the mesh in Q .

Now let these points and curves which map onto the finite element mesh in Q be mapped into w -space under F . Conformality of the mapping F except at the set of points $\{\xi_1, \xi_2, \dots, \xi_n\}$ ensures that the composite mapping $F \circ G^{-1}$ from the z -plane to the w -plane,

$$w = (F \circ G^{-1})(z) = F(G^{-1}(z))$$

is conformal except at the vertices of the polygon Q and at those points on the boundary of Q which map onto the points $\{\xi_i\}$ under G^{-1} . (G^{-1} is the inverse mapping of G .)

Hence if the edges which make up the mesh in Q are mapped into curves in P by the mapping $F \circ G^{-1}$, these curves intersect at the same angles as the corresponding edges in Q , except at a finite number of points on the boundary of P . This breakdown of conformality at some points of Q means that the curves into which the boundary edges of some elements map are not smooth.

To produce a finite element mesh in P , the curves, mapped from Q under $F \circ G^{-1}$, are approximated by straight lines joining the nodal points v_j . v_j is defined by the equation

$$v_j = F(\zeta_j) = F[G^{-1}(z_j)].$$

If the nodes of this mesh are numbered in the natural way, that is, node j is defined to be that node whose coordinates are given by v_j , then the topology of the mesh in P , the arrangement by which the edges join the nodes, is identical to that of the mesh in Q – hence the bandwidths are the same. The mesh thus created is of the same type as that in Q (for example composed of triangles or rectangles); moreover in regions of P which are of the greatest interest and where therefore the elements are smallest, this straight line approximation is least noticeable. In such a region, then, the edges intersect at very nearly the same angles as the corresponding edges of the mesh in Q . Conversely it is only in regions of least interest where element size is relatively large that the intersection angles differ significantly from those in polygon Q .

Thus in those places of P where stresses are expected to be changing most rapidly and where element size is smallest, the mesh is "good" in the same sense as was the mesh in Q .

11. Choice of p and q

There remain two variables or degrees of freedom undefined which will determine the number of boundary nodes in P and their distribution. It will be the distribution of the boundary nodes alone which will determine where should be the smallest elements and where the largest.

p and q are real scalars ($p > 0$) which may be chosen at will to define the n real numbers ξ_j by

$$\xi_j = p\chi_j + q \quad \text{for } j = 1, 2, \dots, n.$$

The χ_j satisfy the condition

$$-1 = \chi_1 < \chi_2 < \dots < \chi_{n-1} < \chi_n = 1.$$

The quantities ξ_j are those points in the ζ -plane which map into the vertices of P , $\{w_j\}$, under mapping F . The quantities $\{\Omega_j\}$, n_Ω in number, have been defined as those ζ_j points which are real and are arranged in ascending algebraic order.

It is a consequence of the Schwarz–Christoffel transformation F from the upper half ζ -plane to polygon P that any point ζ lying on the real axis between consecutive points ξ_j and ξ_{j+1} , that is, $\xi_j < \zeta < \xi_{j+1}$, is mapped onto a point on that part of the boundary of P between vertices w_j and w_{j+1} . The number of nodes in the generated finite element mesh between vertices w_j and w_{j+1} is, then, the number of Ω_k points which lie between ξ_j and ξ_{j+1} .

It is here that the r integers t_1, t_2, \dots, t_r are defined together with vertex numbers of P , u_1, u_2, \dots, u_{r-1} where

$$0 < u_1 < u_2 < \dots < u_{r-1} < n + 1 \quad (r > 2).$$

It is these quantities which determine how many boundary nodes there should be on the mesh in P , as well as specifying their distribution. They are defined as follows:

– t_1 is the number of boundary nodes which would ideally lie between vertex w_0 and vertex w_{u_1} ;

– (for $j = 2, 3, \dots, r-1$); t_j is the number of boundary nodes which would ideally lie between vertex $w_{u_{j-1}}$ and vertex w_{u_j} ;

– t_r is the number of boundary nodes which would ideally lie between vertex $w_{u_{r-1}}$ and vertex w_0 .

Ideally the generated mesh should have $\sum t_i$ boundary nodes. To achieve this ideal distribution of boundary nodes, p and q should be chosen so that there are $t_1 \Omega_k$ points less than ξ_{u_1} , $t_r \Omega_k$ points greater than $\xi_{u_{r-1}}$ and, for $j = 2, 3, \dots, r-1$, $t_j \Omega_k$ points between $\xi_{u_{j-1}}$ and ξ_{u_j} .

In general this is an over-specified problem since there are only two variables to define, p

and q , and it is solved by minimising a sum-of-squares function. In order to solve what is basically an integer-valued functional problem uniquely and by a well-tried numerical method, a real-valued function of a real variable is introduced. $\phi(x)$ is defined to be a monotonically-increasing, continuously-differentiable function so that its integral value is the number of Ω_k points less than or equal to x . In particular, the relationship $\phi(\Omega_j) = j$ holds. For reasons of symmetry, $\phi(-\infty)$ and $\phi(\infty)$ are defined by

$$\phi(-\infty) = 0, \quad \phi(\infty) = n_\Omega + 1.$$

p and q are chosen to be those values which minimise the sum-of-squares function S_r , where S_r is defined by

$$\begin{aligned} S_r &= \{\phi(\xi_{u_1}) - \phi(-\infty) - t_1\}^2 + \sum_{j=2}^{r-1} \{\phi(\xi_{u_j}) - \phi(\xi_{u_{j-1}}) - t_j\}^2 + \{\phi(\infty) - \phi(\xi_{u_{r-1}}) - t_r\}^2 \\ &= \{\phi(\xi_{u_1}) - t_1\}^2 + \sum_{j=2}^{r-1} \{\phi(\xi_{u_j}) - \phi(\xi_{u_{j-1}}) - t_j\}^2 + \{n_\Omega + 1 - \phi(\xi_{u_{r-1}}) - t_r\}^2. \end{aligned}$$

Once again the Newton-Raphson algorithm is used to solve the two equations

$$\frac{\partial S_r}{\partial p} = \frac{\partial S_r}{\partial q} = 0$$

which minimise S_r with respect to p and q .

A detailed example is included in a later section, but first some alterations must be made to the generated mesh so this it is a form suitable for input to an analysis program.

12. Alterations to the generated mesh

Function F has now been completely defined to map the ζ_j points onto the nodes v_j on the boundary and interior of polygon P in the w -plane. However there remain two further operations on the created mesh in P before the represented structure may be analysed by a finite element computer program.

Any node v_j which lies on a polygon edge modelling a curved arc in the region R must be re-defined so that it lies on the nearest point of that arc to its calculated position. If too few polygonal edges were chosen to model such a curved arc, this re-definition of boundary node coordinates may upset the desired element shape in such regions.

The second point is as follows. Consider a particular vertex of the polygon P , say w_j where $j \neq 0$. Then unless by coincidence in the choice of p and q , w_j is the image under F of some point ζ_k , that is, unless $\xi_j = \zeta_k$ for some k , then no node of the mesh in Q will be mapped onto w_j . It is true, however, that w_j lies on the image of one of the boundary element edges of the mesh in Q but, since F is not conformal at ξ_j , this image edge is not smooth. Hence when this edge is approximated by a straight line, the effect is to cut off the corner at w_j . Clearly it is the most fundamental requirement of a mesh that there should be a node at all vertices w_j of P .

The possible exception to this is where w_j is at the junction of two polygonal edges each of which is modelling the same curved arc.

Assuming there should be a node at w_j , there are two ways to achieve this end. Firstly, the coordinates of the nearest boundary node to w_j could be re-defined to those of w_j . This would mean that the topology of the mesh would remain unchanged and would be an acceptable move if the shape of the neighbouring elements was not significantly altered.

The second alternative is to add a new node at w_j and re-define the surrounding elements, or add a new element, to contain it. The new node must be carefully numbered, however, so as not to increase the bandwidth of the mesh unnecessarily.

It has not proved difficult to accomplish either of these alternatives or indeed to establish a condition so one or the other may be chosen automatically within a computer program.

13. Examples of generated finite element meshes

As an example to demonstrate the method by which a finite element mesh may be generated in any two-dimensional simply-connected region, a triangular mesh is constructed within the region R . R is defined to be one quarter of a square plate with a central circular hole within it. The diameter of the hole is one-fifth the length of the side of the square. P is defined to be the polygon modelling R , two straight lines representing the circular arc. The six vertices are numbered as shown in fig. 9.

In this example and in the others included, the polygon Q is a constant figure for which mapping G has been calculated beforehand. Q is defined to be a regular hexagon in the z -plane as shown in fig. 10. The mapping $z = G(\zeta)$ which maps the upper half ζ -plane into Q is

$$G(\zeta) = -2.27 \int_i^\zeta \frac{d\zeta}{(\zeta + 3)^{1/3}(\zeta + 1)^{1/3}\zeta^{1/3}(\zeta - 1)^{1/3}(\zeta - 3)^{1/3}} + (0.25 - 0.41 i)$$

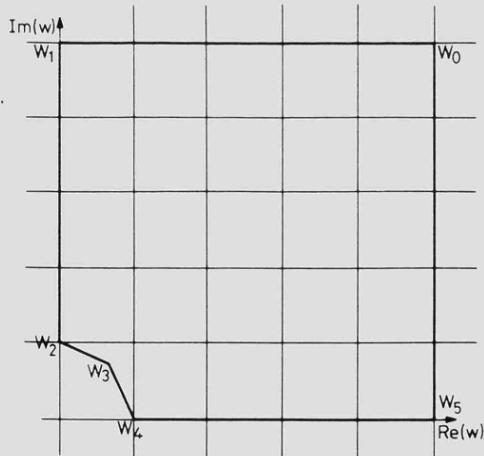


Fig. 9. Modelling polygon P in the w -plane.

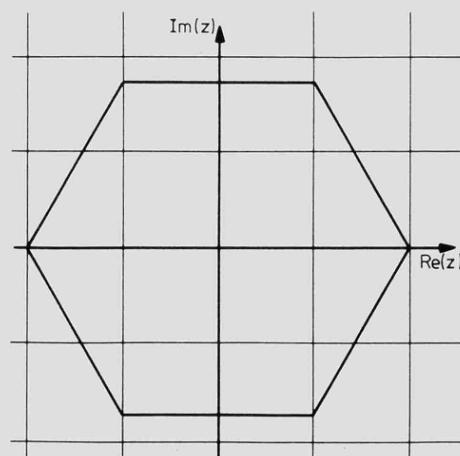


Fig. 10. Polygon Q in the z -plane.

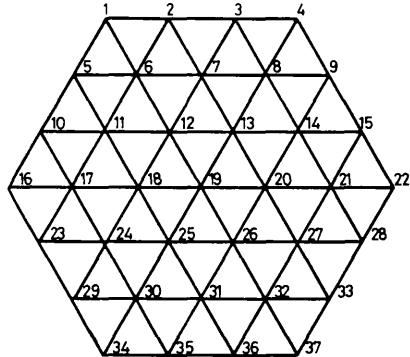
Fig. 11. The mesh drawn in Q .

Table 1
Values of t_j and u_j

j	t_j	u_j
1	1	1
2	5	2
3	5	4
4	5	5
5	1	

so that the ζ -points ∞ , -3 , -1 , 0 , 1 and 3 map into the vertices of $Q - 1 + i\sqrt{3}$, -2 , $-1 - i\sqrt{3}$, $1 - i\sqrt{3}$, 2 and $1 + i\sqrt{3}$, respectively.

The mesh within Q is defined to be composed entirely of equilateral triangles, the particular one in fig. 11 having 18 boundary nodes. (In general, if the mesh in Q has $6m$ boundary nodes, then it has a total of $3m(m+1)+1$ nodes and $6m^2$ elements.) The nodes are numbered in such a way as to minimise the mesh's bandwidth, that is, the maximum difference in the node numbers which define any one element. In this case, this maximum number is 7.

Having evaluated the parameters defining the Schwarz-Christoffel transformation from the upper ζ -plane onto P , the values for χ_i are as follows:

$$\chi_1 = -1, \quad \chi_2 = -0.12, \quad \chi_3 = 0, \quad \chi_4 = 0.12, \quad \chi_5 = 1,$$

the symmetry of χ reflecting the symmetry of P .

Apart from the coordinates of the vertices of P , the only other input to the computer program is that defining t_j and u_j . In this case, these are as shown in table 1 ($r = 5$). These indicate how the boundary nodes should ideally be distributed. There should be 1 between w_0 and w_1 , 5 between w_1 and w_2 , 5 between w_2 and w_4 , 5 between w_4 and w_5 and 1 between w_5 and w_0 . Thus

$$\sum_j t_j = 17$$

and the nearest multiple of 6 to this figure is 18, determining into how many edges each of the regular hexagon's sides should be divided (fig. 11). The 36 finite ζ_j quantities may now be evaluated and they are shown in fig. 12, together with the arcs which map under G into the edges of the mesh in Q . ($\zeta_1 = \infty$, ζ_2 and ζ_5 are off the scale.) In particular, the 17 real Ω_j values are as shown in table 2.

Minimising the sum-of-squares function yields values for p and q of

$$p = 6.93, \quad q = 0,$$

the zero value of q again reflecting P 's symmetry. These p and q values define ξ_j to be as

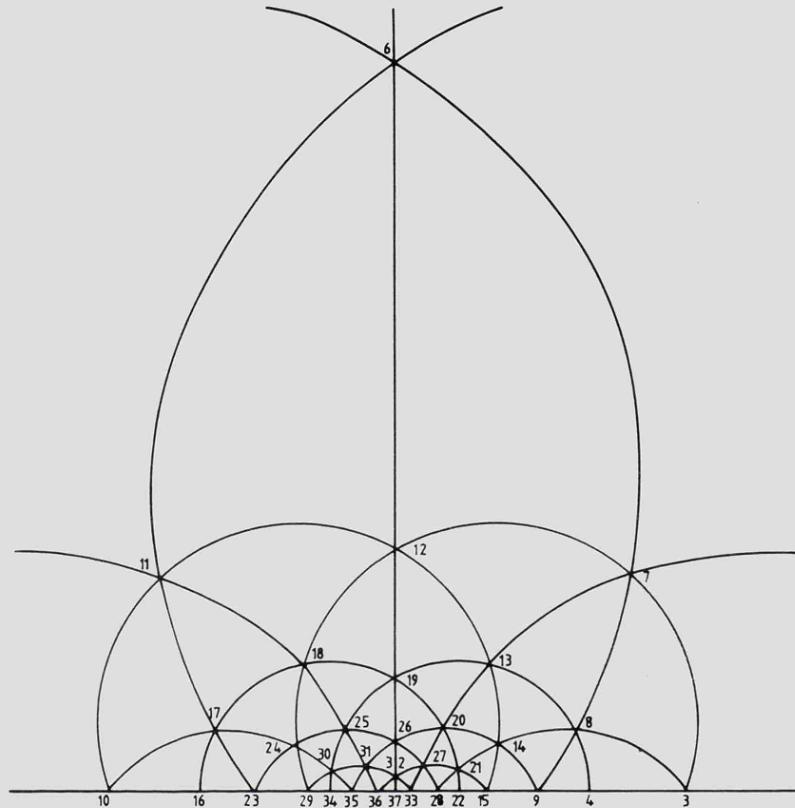
Fig. 12. Edges of the mesh in Q mapped into the ζ -plane.

Table 2
Values of Ω_j

j	Ω_j	j	Ω_j	j	Ω_j
1	-11.61	7	-0.68	13	1.38
2	-4.43	8	-0.25	14	2.17
3	-3.00	9	0.00	15	3.00
4	-2.17	10	0.25	16	4.43
5	-1.38	11	0.68	17	11.61
6	-1.00	12	1.00		

follows:

$$\xi_1 = -6.93, \quad \xi_2 = -0.86, \quad \xi_3 = 0, \quad \xi_4 = 0.86, \quad \xi_5 = 6.93,$$

so that, in this case, the desired numbers of Ω_j points between $-\infty$ and ξ_1 , ξ_1 and ξ_2 , ξ_2 and ξ_4 , ξ_4 and ξ_5 , and between ξ_5 and $+\infty$ are in fact achieved.

The images of the mesh edges in Q under the composite mapping $F \circ G^{-1}$ are shown in fig.

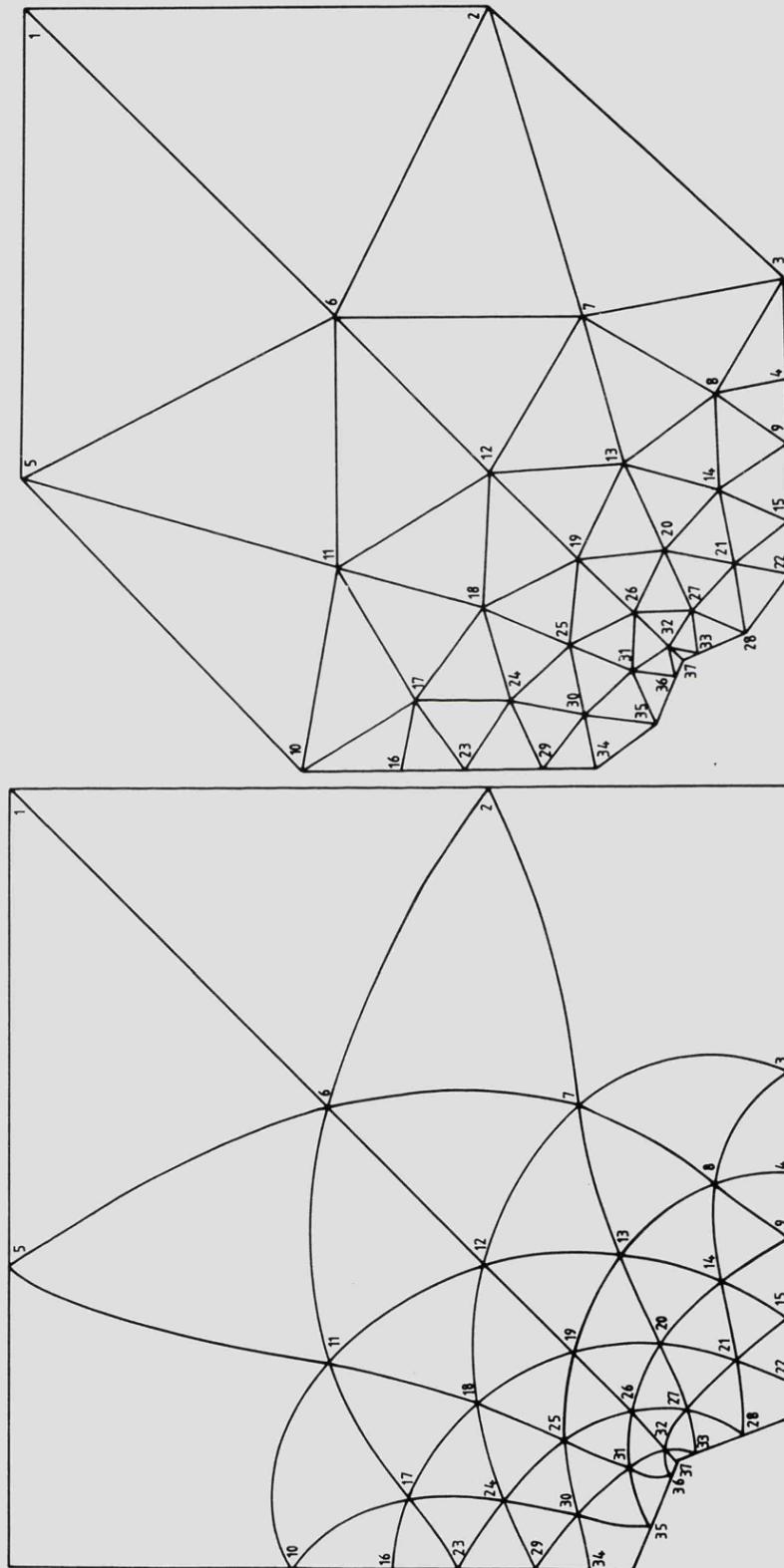
Fig. 13. Edges of the mesh in Q mapped into the w -plane.

Fig. 14. Mapped edges approximated by straight lines.

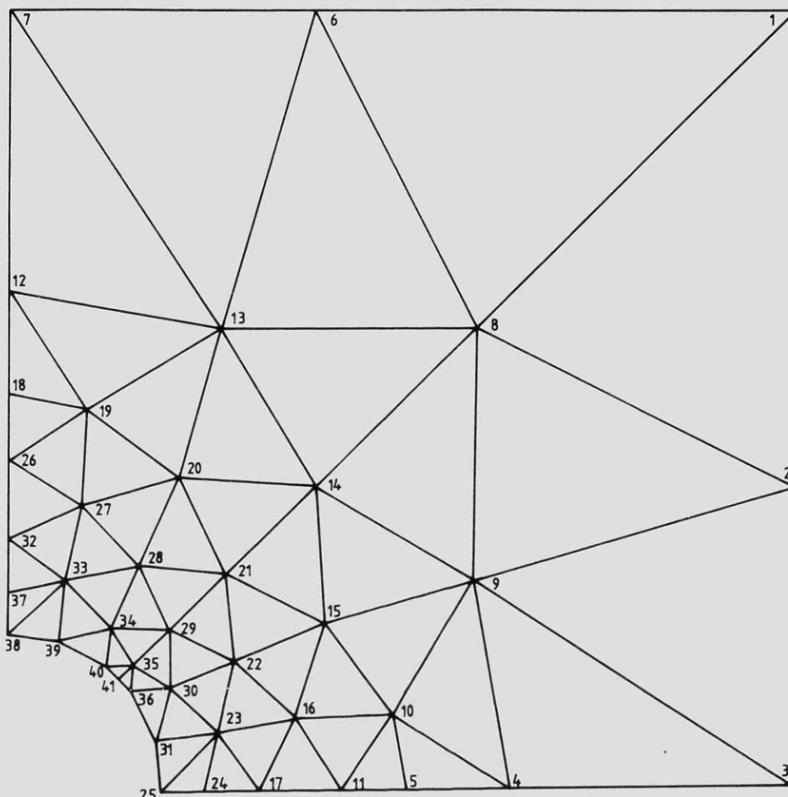


Fig. 15. Final mesh. Coordinates of some nodes altered and new nodes added.

13. These arcs, except where the mapping is not conformal, intersect at angles of 60° as did the edges in Q . The result of approximating these edges by straight lines is shown in fig. 14.

To produce the final mesh shown in fig. 15, the following actions have been taken. The coordinates of the nodes numbered 35, 36, 33 and 28 have been re-defined to lie on the circular arc centred on the origin, as opposed to the modelling straight line segments on which they previously lay. New nodes have been added at the four vertices of P , w_1 , w_2 , w_4 and w_5 appropriately numbered. The other nodes have been re-numbered accordingly.

To illustrate the power of the method which has been described, two more examples are shown in figs. 16 and 17. In the first figure, the region is a quarter of a plate containing an elliptical hole. In the second, a sharp crack, of the same length as the elliptical hole, is modelled in the quadrant of a square plate.

14. Discussions and conclusions

The method described has proved very useful in the generation of finite element meshes for many different geometries. One of its advantages is that a minimal amount of input data is

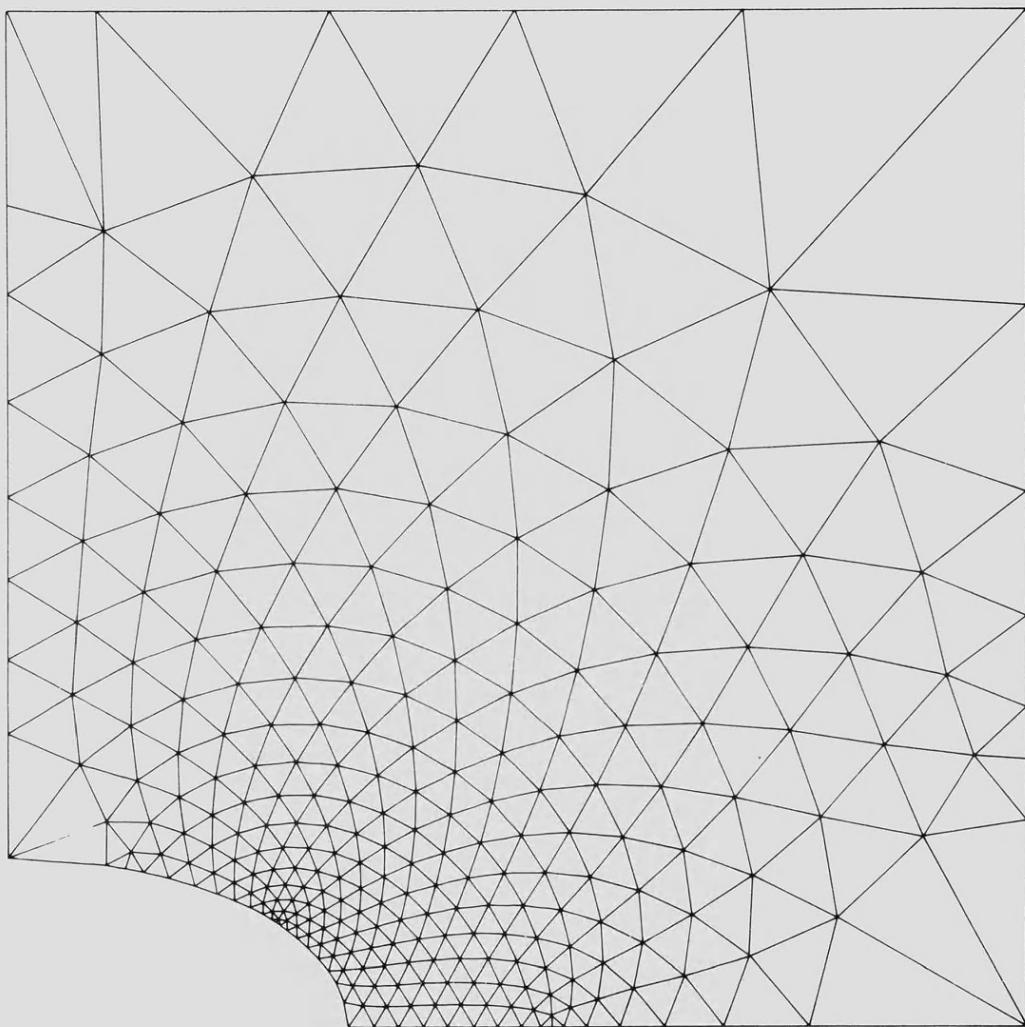


Fig. 16. Mesh drawn in the quadrant of a square plate containing an elliptical hole.

necessary – in fact to produce each of the three example meshes illustrated in figs. 15, 16 and 17, less than 40 numbers were input to the program. These consisted of the coordinates of the vertices of the regions (plus information that some of the edges were curved arcs) together with the quantities t_i and u_i determining the distribution of the boundary nodes.

It should be emphasised, however, that the method is not restricted to meshes composed of three-node triangles. All that is required is that the polygon Q should be chosen so that it can easily be filled with ideally-shaped elements and that the nodes should be numbered in a systematic manner. This numbering system will be maintained in the modelling polygon P and a good mesh with small bandwidth within P is assured.

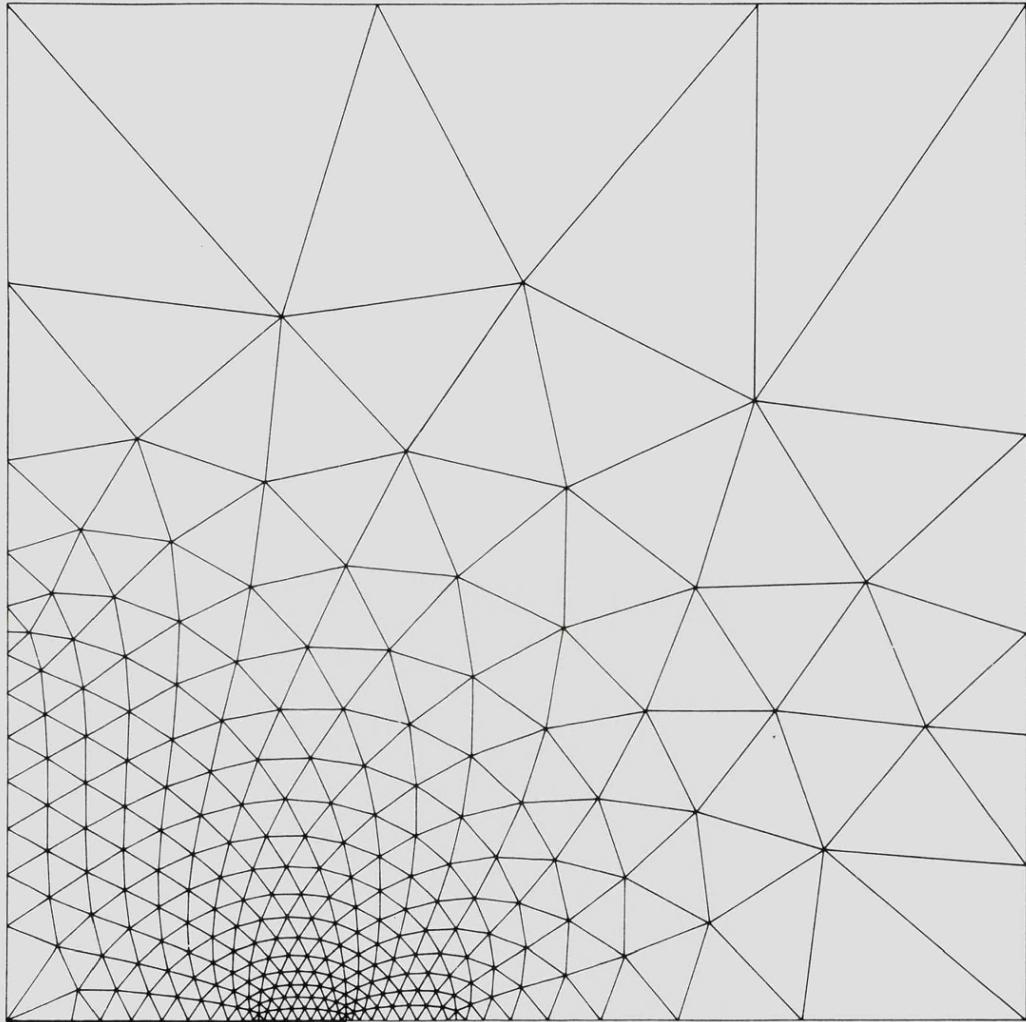


Fig. 17. Mesh drawn in the quadrant of a square plate containing a sharp crack.

Appendix. Proof that $\gamma(\xi') = \gamma(\xi)$ where $\xi'_j = p\xi_j + q$ ($p > 0$)

Consider any two points, a_1, a_2 in or on the polygon produced by the transformation

$$w = \int_{\zeta_0}^{\zeta} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \xi_k)^{1-\alpha_k/\pi}}.$$

Suppose a_1 and a_2 are the images of ζ_1 and ζ_2 . Let ζ'_1 and ζ'_2 be defined by

$$\zeta'_1 = p\zeta_1 + q, \quad \zeta'_2 = p\zeta_2 + q.$$

Then the distance between a'_1 and a'_2 , the images of ζ'_1 and ζ'_2 under the transformation

$$w = \int_{\zeta_0}^{\zeta'} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \xi'_k)^{1-\alpha_k/\pi}}$$

is

$$|a'_1 - a'_2| = \left| \int_{\zeta_1'}^{\zeta_2'} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \xi'_k)^{1-\alpha_k/\pi}} \right|.$$

But this expression is

$$\left| \int_{p\xi_1+q}^{p\xi_2+q} \frac{d\zeta}{\prod_{k=1}^n [\zeta - (p\xi_k + q)]^{1-\alpha_k/\pi}} \right|.$$

By a change of variable $\zeta = p\theta + q$, this becomes

$$\begin{aligned} \left| \int_{\zeta_1}^{\zeta_2} \frac{p d\theta}{\prod_{k=1}^n [p\theta + q - (p\xi_k + q)]^{1-\alpha_k/\pi}} \right| &= \frac{p}{p^{\sum_{k=1}^n (1-\alpha_k/\pi)}} \left| \int_{\zeta_1}^{\zeta_2} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \xi_k)^{1-\alpha_k/\pi}} \right| \\ &= p^{1-\sum_{k=1}^n (1-\alpha_k/\pi)} |a_1 - a_2|. \end{aligned}$$

Thus the distance between any two points in or on the polygon produced by the quantities $\xi'_j = p\xi_j + q$ is a constant multiple of the distance between the corresponding points in the polygon produced by the quantities ξ_j . Thus these two polygons are similar and $\gamma(\xi') = \gamma(\xi)$.

References

- [1] I.A. Nazlawy, Computer aided generation of finite element meshes, M. Phil. Thesis, Leicester University (1974).
- [2] G. Carrier, M. Krook and C. Pearson, Functions of a complex variable (McGraw-Hill, New York, 1966).
- [3] O.C. Zienkiewicz, The finite element method in engineering science, 2nd edition (McGraw-Hill, London, 1978).

APPENDIX 2

Paper "The use of the Schwarz-Christoffel transformation in mesh generation for the solution of two-dimension problems" by P.R. Brown and D.R. Hayhurst, from "Computers in Mechanical Engineering", volume 1, number 1, ASME, New York, August 1982. Originally appeared in L.E. Hulbert (ed.), "Computers in Engineering", Proceedings of International Conference, San Diego, California, 15-19 August, 1982, pp 1-8, ASME, New York (1982). Reproduced by kind permission of the publishers.

Using the Schwarz-Christoffel Transformation in Mesh Generation for the Solution of Two-Dimensional Problems

P.R. Brown and D.R. Hayhurst

Department of Engineering
University of Leicester
Leicester, England

This paper presents a computer-based finite element mesh generation scheme that aims to produce a good mesh with the minimum of input data. The scheme relies on the Schwarz-Christoffel transformation, a conformal mapping from complex variable theory.

The common use of the finite element method has meant that the stress distribution in complex structures can be analyzed theoretically by modeling the structure as a number of elements of finite size. Areas where stresses are expected to change rapidly must be modeled by many small elements but economy prohibits the use of equally small elements everywhere. Regions where stresses are expected to vary little spatially may be represented by larger elements. Even when drawing a mesh by hand, gradual change from small to large elements can be difficult to achieve and the preparation of data can still use up a high proportion of a stress analyst's time.

The computer-based mesh generation scheme presented here is designed to be noninteractive. The scheme is restricted to two-dimensional regions that are simply connected. The only input to the program consists of the coordinates defining the shape of the region in which the mesh is to be drawn and some information to define the distribution of the boundary nodes. Emphasis is placed on generating a "good" mesh, that is, a mesh composed of elements of nearly ideal shape. To do this, we make extensive use of the theory of conformal mappings, an area of complex analysis, and describe in detail a numerical method of obtaining the defining parameters of one such mapping, the Schwarz-Christoffel transformation. Two Schwarz-Christoffel transformations are used to define a composite mapping between two general polygons, and the angle-preserving nature of the mappings used will be shown to play an important part in the generation of a good mesh.

Conformal Mapping

The study of mappings forms a large part of complex variable theory in mathematics. In particular a conformal mapping has the property that angles are preserved—that is, a mapping is conformal at a point z_0 if the angle of intersection of any two smooth curves through z_0 is equal, in both magnitude and sense of rotation, to the angle of intersection of the image curves in the mapped region. The use of conformal mappings is a well-known technique in the solution of boundary value problems and in fluid dynamics.

The Schwarz-Christoffel transformation maps the upper half infinite plane onto the interior of a general polygon. The parameters that define this mapping are presented, and a practical method described which evaluates these parameters numerically.

Schwarz-Christoffel Transformation

We now consider in detail the Schwarz-Christoffel transformation, a conformal mapping from the upper half ξ -plane ($\xi = \xi + i\eta$) onto the interior of a general polygon P . The mapping is one-to-one and is also conformal along the real ξ -axis except at a finite number of points. Continuity ensures that the ξ -axis maps onto the boundary of P in w -space. The problem of finding the mapping when P is a triangle will be shown to be trivially soluble, so we shall assume that P has $(n + 1)$ sides with $n \geq 3$. Let the vertices of P be represented by the complex numbers w_0, w_1, \dots, w_n arranged in anticlockwise order (Fig.

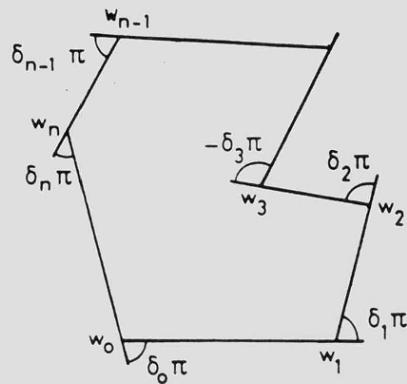


Fig. 1 General polygon P in w-space.

1) and let the exterior angles at these vertices be $\delta_0\pi, \delta_1\pi, \dots, \delta_n\pi$ ($-1 < \delta_k < 1$). It can readily be shown [1] that the Schwarz-Christoffel formula

$$w = F(\zeta) = a \int_{\zeta_0}^{\zeta} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \xi_k)^{\delta_k}} + b \quad (1)$$

is the relationship between ζ and w by which this conformal mapping is achieved. The n real numbers $\xi_1, \xi_2, \dots, \xi_n$ satisfy $\xi_1 < \xi_2 < \dots < \xi_n$ and map onto w_1, w_2, \dots, w_n under F . w_0 is the image of ∞ . Although δ_0 does not appear in (1), it is included indirectly because of the relationship

$$\sum_{k=0}^n \delta_k = 2.$$

ζ_0 is an arbitrary constant with $\text{Im } \zeta_0 \geq 0$. Once ζ_0 has been chosen, the value of b fixes the polygon in space and a determines its size by $|a|$ and orientation by $\arg a$. F is not conformal at the set of points $\{\infty, \xi_1, \xi_2, \dots, \xi_n\}$ but it is well-defined there although the integral in (1) may be singular at $\zeta = \xi_k$ if $\delta_k > 0$.

The Schwarz-Christoffel formula describes the form of the mapping we seek but, given a polygon P , the problem of finding the quantities ξ_k , a and b is in general a difficult one which can only be solved numerically. For any set of real numbers ξ_k , the nature of (1) ensures that the angles of the mapped polygon come out correctly. The difficult numerical task is finding the particular set of ξ_k which maps onto a polygon whose sides are in the same proportions as P . To do this, we define P by a different set of $2(n + 1)$ quantities from the ordered coordinates of its vertices w_k , namely

- the n exterior angles $\delta_0\pi, \delta_1\pi, \dots, \delta_{n-1}\pi$ at the vertices w_0, w_1, \dots, w_{n-1} ;
- the coordinates of just two arbitrary vertices;
- $n - 2$ quantities $\beta_1, \beta_2, \dots, \beta_{n-2}$ to describe the relative lengths of the sides of the polygon. β_j is defined by

$$\beta_j = \sum_{k=1}^j |w_k - w_{k-1}| / \sum_{k=1}^{n-1} |w_k - w_{k-1}|$$

for $j = 1, 2, \dots, n-2$.

The numbers $\delta_1, \dots, \delta_{n-2}, \beta_1, \dots, \beta_{n-2}$ describe the relative positions of the vertices w_0, w_1, \dots, w_{n-1} ; δ_0 and δ_{n-1} define the remaining two edges and so, as long as $\delta_n \neq 0$, the remaining vertex w_n at their intersections. The coordinates of any two distinct vertices fix and scale the polygon in cartesian space. We note that the restriction that δ_n be nonzero is of no consequence since if this were the case, there would be no purpose in including w_n as a vertex of P . However a degenerate vertex at w_0 with interior angle equal to π radians is allowed by this definition.

To find the values of ξ_k which map onto the vertices of a given polygon P , we proceed as follows. Suppose that for any real vector ξ whose components satisfy $\xi_k < \xi_{k+1}$, the vertices of polygon P' , v_j , are defined by

$$v_0 = \int_{\zeta_0}^{\infty} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \xi_k)^{\delta_k}}, \quad v_j = \int_{\zeta_0}^{\xi_j} \frac{d\zeta}{\prod_{k=1}^n (\zeta - \xi_k)^{\delta_k}} \quad (2)$$

and that γ_j is defined by

$$\gamma_j = \sum_{k=1}^j |v_k - v_{k-1}| / \sum_{k=1}^{n-1} |v_k - v_{k-1}|$$

for $j = 1, 2, \dots, n-2$. As the constrained vector ξ varies, the polygon with its vertices defined by (2) ranges over the set of all polygons with the given angles $\delta_k\pi$. The quantities γ_j , which may be thought of as functions of ξ , uniquely describe the relative lengths of the sides of P' . To define the Schwarz-Christoffel mapping, we must find that particular ξ for which $\gamma_j = \beta_j$. Since there are n degrees of freedom in choosing ξ_k but only $n - 2$ degrees of freedom in defining the proportions of the mapped polygon, it is evident that we cannot hope to find a unique ξ which is a solution of

$$\gamma_j(\xi) = \beta_j \quad (3)$$

In fact it can be shown [2] by a straightforward change of variable in (2) that if ξ satisfies (3) then so does ξ' where the components of ξ' and ξ are related by

$$\xi'_k = p\xi_k + q$$

for any real p, q with $p > 0$. Thus it can be seen why the $n = 2$ case is trivially soluble as noted earlier. Any ξ_1, ξ_2 with $\xi_1 < \xi_2$ will solve (3) and so suffice to define the mapping.

Numerical Solution of $\gamma_j(\xi) = \beta_j$

In order to solve (3) uniquely, the domain of the functions γ_j must be restricted so that ξ has only $n - 2$ degrees of freedom. This is done here by fixing two components of ξ at constant values: ξ_1 and ξ_n are held fixed at -1 and 1 respectively. This ensures that all the ξ_k remain bounded so that poor scaling does not become a problem. This constrained ξ is denoted by χ where the components of χ satisfy

$$-1 = \chi_1 < \chi_2 < \dots < \chi_n = 1.$$

The equations to be solved now are

$$\gamma_j(\chi) = \beta_j. \quad (4)$$

This set of $n - 2$ nonlinear equations is solved by Newton's method which, given an estimate of the solution, calculates

a set of corrections to predict a better estimate. A slight complication here, though, is that since the domain of functions γ_j is only a subset of n -dimensional space R^n , we must ensure that the addition of the Newton increment to the current estimate does not send the predicted x outside the functions' domain. The corrections must be weighted if necessary to ensure that the x_k always satisfy $x_k < x_{k+1}$. Newton's method has proved to be reliable and rapidly convergent for many geometries.

Once x , the solution of (4), has been obtained, the general solution of (3) is immediately available. It is ξ where the components satisfy $\xi_k = px_k + q$ for any real scalars p, q with p positive. These two quantities leave two degrees of freedom in the choice of mapping from the upper half plane onto the interior of P and it is clear that this is as many as may be expected. Along with where ∞ should be mapped, that is, where w_0 is situated on the boundary of P , this amounts to three degrees of freedom in choosing the conformal map. This is in agreement with the general Riemann mapping theorem of complex transformations.

Whatever the choices of p and q , the mapping is defined completely once the two complex numbers a and b have been defined, the quantities that locate and scale polygon P' , similar to P , onto P itself.

Conformal Mapping Between Two Polygons

In earlier sections, we have described a practical method for finding a transformation from the upper half ζ -plane onto the interior of a polygon P . The transformation is conformal everywhere except at the points ∞ and ξ_k , those points whose images are the vertices w_0, w_1, \dots, w_n of P . Suppose now that Q is another polygon in z -space with vertices z_k . The conformal mapping from the upper half ζ -plane onto the interior of Q can also be found numerically.

Suppose that this transformation is

$$z = G(\zeta) = c \int_{\zeta_0}^{\zeta} g(\zeta) d\zeta + d \quad (5)$$

Since the Schwarz-Christoffel transformation G is one-to-one onto, G^{-1} exists so that the composite mapping

$$w = H(z) = (F \circ G^{-1})(z) = F(G^{-1}(z))$$

can be defined. This mapping, being composed of two conformal maps is itself conformal, one-to-one onto from the interior of Q to the interior of P . It is also conformal on the boundary of Q except at the points $\{z_k\}$ and $\{G^{-1}(\xi_k)\}$, that is, at the vertices of Q and at those points that map onto the vertices of P .

The task of inverting the map G turns out to be an easy numerical problem. Since the derivative of G is readily available, namely $cg(\zeta)$, Newton's method is again used and has proved to converge very quickly. $g(\zeta)$ is defined and nonzero everywhere except at those ζ -points that map onto z_k . This proves no difficulty since the points $G^{-1}(z_k)$ are already known and do not need to be calculated.

The maps F , G , and H are represented pictorially in Fig. 2.

Generation of a Mesh in a Given Polygon

Desirable Properties of a Finite Element Mesh

Meshes that model two-dimensional regions are usually composed of triangular or quadrilateral elements. It is accepted that the best element shape that models a finite change in the dimensions of a structure is symmetrical; that

is, the element is an equilateral triangular or square. For a general region, it is impossible to define a mesh composed entirely of symmetrically shaped elements, but a mesh containing elements as near to the ideal as possible is the most desirable.

It is important that any change in element size should be gradual over the region. Since stress as a function of position is invariably continuous, so the finite elements that model a structure too should vary gradually in size.

Although not strictly a property of a finite element mesh, the size of the bandwidth of the resulting system of linear equations is closely connected to mesh generation. One tries to minimize the bandwidth by the most efficient node numbering scheme in order to make the equation solving process as economical as possible. If an optimal numbering system can be an inherent part of mesh generation, this is an obvious advantage.

Use of Mapping H to Generate a Mesh in P

We have introduced the conformal Schwarz-Christoffel transformation mapping the upper half plane onto the interior of a general polygon and have indicated that the defining parameters of this mapping may be found numerically. By combining two such mappings and using the upper half ζ -plane as a convenient intermediate region, a conformal mapping H from polygon Q to polygon P has been defined. A practical method for finding H has been described. In accordance with the Riemann mapping theorem, there are three degrees of freedom in the choice of H or, equivalently, we may choose three distinct points on the boundary of Q and specify their images under H on the boundary of P . We now show how to use the mapping H to generate a finite element mesh by making appropriate choices of P and Q .

Suppose R is a two-dimensional simply connected region in which a finite element mesh is to be drawn. It is proposed that P be so defined to model R . If the boundary of R is composed entirely of straight lines this is readily achieved. If part of the boundary of R is a curved arc, this

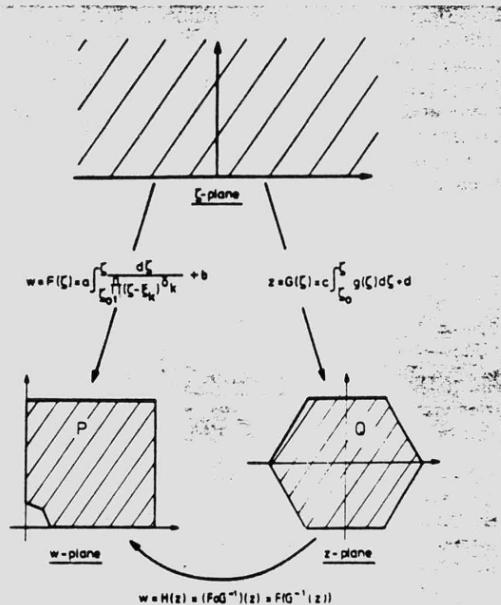


Fig. 2 Pictorial representation of the different mappings used.

may be split into a suitable number of straight line segments. Q is chosen to be a polygon in which a good finite element mesh of any density can easily be generated, that is, a mesh composed entirely of ideally shaped elements. The nodes defining the mesh in Q can easily be numbered to minimize the bandwidth of the resulting system of linear equations. This is readily achieved in a regular mesh.

We now simply apply mapping H to all the element edges in Q . Conformality of H ensures that, except at a finite number of points on the boundary, these edge images—in general they will be curves—intersect at the same angles as those edges that make up the perfect mesh in Q . The breakdown of conformality at some points of Q means that the curves into which the boundary edges of some elements map are not smooth.

To obtain a mesh in P , these curved edge images are replaced by straight lines joining the mapped nodal points. As the topology of the mesh in Q , that is, the arrangement by which the edges join the nodes, is retained in P , the bandwidths are identical. The mesh thus created is of the same type as that in Q ; moreover, in regions of P which are of greatest interest and where therefore the elements are smallest, this straight line approximation is least noticeable. Conversely, it is only in regions of least interest where element size is relatively large that the intersection angles differ significantly from those in polygon Q . Thus in those places of P where stresses are expected to be changing most rapidly and where element size is smallest, the mesh is good in the same sense as is the mesh in Q . The nature of the continuous function H ensures that the change in element size is gradual over P .

Choice of Free Parameters

There remain three degrees of freedom in the choice of mapping F and hence H , namely the positioning of vertex $w_0 = F(\infty)$ and the values of the real scalars p and q . It is these choices that will determine the distribution of the boundary nodes of the generated mesh and thus where the smallest elements should be positioned and where the largest. Since it is only at some points of the boundary that H fails to preserve angles, it is only near the boundary of P where element shape may deteriorate. We shall choose the free parameters of H by considering only the distribution of the boundary nodes—those elements away from the boundary are assured to be well-shaped.

We shall assume that Q is a constant polygon, that G has been computed previously, and that the position of vertex w_0 has been decided. Then if it is known how many nodes there should be in the generated mesh in P , it is a simple task to define the mesh in Q to have the appropriate number of nodes. Moreover, all those ζ -points that map into the nodes in Q under G can be calculated. We shall denote by λ_j those points that map onto the boundary nodes—these are all real and so lie along the ξ -axis. Any given values of p and q define ξ_k by

$$\xi_k = px_k + q$$

where χ is the solution of equations (4) and we can count how many λ_j points lie between successive ξ_k points. But the number of λ_j points lying between ξ_k and ξ_{k+1} in ζ -space is precisely the same as the number of nodes that lie between vertices w_k and w_{k+1} of P in w -space. So by appropriate choice of p and q , the distribution of the boundary nodes in the mesh generated in P may be decided.

Within the automatic mesh generation program, we choose p and q to minimize a sum of squares “penalty

function,” that is, a function which measures the difference between the situation achieved and that requested.

Alterations to the Generated Mesh

Having chosen w_0 , p and q , mapping H is completely defined. However, some slight amendments need to be made to the generated mesh in P before the modeled structure may be analyzed by a finite element program.

Any node that lies on a polygonal edge modeling a curved arc in region R must be redefined so that it lies on the nearest point on that curve to its calculated position. If too few polygonal edges were chosen to model such a curved arc, this redefinition of boundary node coordinates may upset the desired element shape in such a region.

Secondly we must ensure that there is a node at every vertex of P , except possibly at a vertex at the junction of two polygon edges each modeling the same curved arc. Although every w_k lies on the image of an element edge of the mesh in Q , since H is not conformal at w_k , this image edge is not smooth. Hence when this edge is replaced by a straight line, the effect is to cut off the corner at w_k . We consider two ways to place a node at w_k . The coordinates of the nearest node to w_k could be redefined to those of w_k . This would mean that the topology of the mesh would remain unchanged and would be an acceptable move as long as the shape of neighboring elements were not significantly altered. Alternatively a new node could be added at w_k and the surrounding elements redefined or a new element added to contain it. This new node must be carefully numbered, however, so as not to increase the bandwidth unnecessarily.

It has not proved difficult to accomplish either of these alternatives or indeed to establish a condition so that one or the other may be chosen automatically.

Examples

In Fig. 3, we illustrate a typical mesh generated by the described procedure. This mesh represents one quarter of a plate containing a circular hole which is loaded uniformly on the top edge. A stress concentration around the hole is modeled by small elements and element size increases

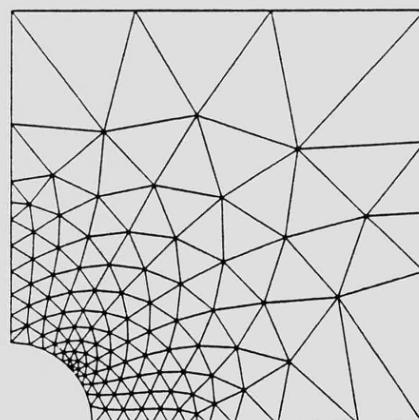


Fig. 3 Example mesh drawn in the quadrant of a square plate containing a circular hole.

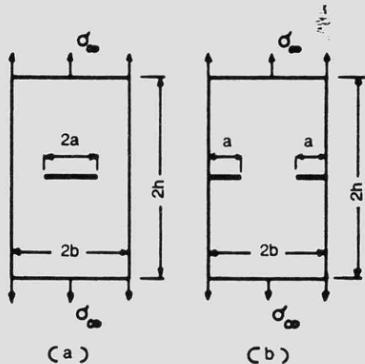


Fig. 4 Crack geometries and loading
(a) center-cracked tension specimen
(b) double-edge-cracked tension specimen.

gradually away from the hole. This mesh is composed entirely of triangular elements so a convenient choice for Q was a regular hexagon. This can be filled with ideally shaped elements, that is, equilateral triangles, to any required density. Although this choice of Q produces an acceptable mesh in this instance, it may be that a fixed Q does not produce ideal meshes for all geometries. The authors believe, however, that it is feasible to choose Q within a program that generates an optimal mesh for any region. Such a polygon Q could be any shape as long as it can be filled with ideally shaped elements, of whatever type, to any density.

A second example is illustrated, for which stress distributions have been calculated, in a later section.

Quality of Generated Mesh

One difficulty always encountered in mesh generation is that of judging its quality or effectiveness. Meshes are usually judged by comparing either point values or spatial distributions of computed field quantities within the domain of the boundary value problem, with those obtained from closed form solutions or from other finite element solutions. These techniques suffer from the deficiency that one is comparing either a point or a line property rather than a property that reflects the overall character of the mesh. For this reason the authors have selected a problem to judge the generated meshes for which a single parameter characterizes the finite element solution in a global sense. Rice's J integral [3] will be computed for center-cracked and double-edge-cracked tension specimens (Fig. 4) and compared with existing solutions. In addition to providing a means of comparison with existing solutions using a single number, the problem allows one to generate a mesh for a situation that provides an extreme test of the mesh, namely that of modeling the singular displacement, stress and strain fields at the crack tip.

Determination of Rice's J Integral

The path independent J integral is used to characterize the fracture behavior of materials and to measure the

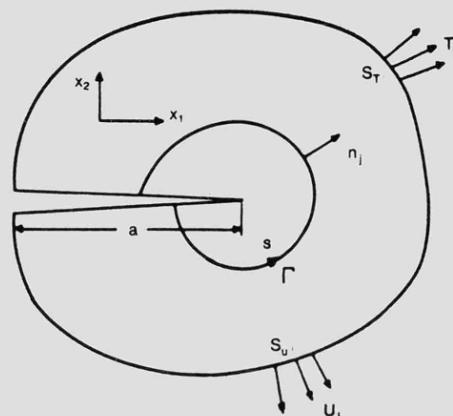


Fig. 5 Body containing a sharp crack with contour Γ indicated.

strength of singularities in the field quantities at the crack tip for nonlinear elastic materials. With reference to the cracked body shown schematically in Fig. 5, the J integral is defined by

$$J = \int_{\Gamma} \left[\int_{\Omega} \epsilon_{ij} \sigma_{ij} d\epsilon_{ij} dx_2 - \sigma_{ij} n_j \frac{\partial u_i}{\partial x_1} ds \right].$$

Γ is any contour around the crack tip and has arc length s . n_j is the unit outward normal, u_i denotes the components of displacement and σ_{ij} and ϵ_{ij} denote the stress and strain tensors. The material is assumed to be elastic, incompressible and to obey a power hardening J_2 deformation law:

$$\epsilon_{ij}/\epsilon_0 = (3/2)(\sigma_e/\sigma_0)^{n-1} s_{ij}/\sigma_0 \quad (6)$$

where ϵ_0 , σ_0 and n are material constants,

$$s_{ij} = \sigma_{ij} - \sigma_{kk} \delta_{ij}/3 \quad \text{and} \quad \sigma_e = \sqrt{(3s_{ij}s_{ij}/2)}$$

Method of Calculation

The authors have previously developed [4, 5] computer programs to obtain stationary-state creep solutions for a range of problems. The definition of a stationary state requires that the rates of change of stress are zero. When this condition is satisfied the stress distributions are identical to those for the nonlinear elastic behavior discussed above, for an n -power creep law, and the displacement and strain rates are equivalent to the nonlinear elastic displacements and strains respectively. In this paper results will be presented of J-integral values determined from creep solutions.

Finite Element Formulation for Creep Solutions

Consider a body to be idealized using triangular elements whose displacements are linear functions of the spatial coordinates. Thus for the i th element the displacements $u^i(x)$ are related to a global displacement vector U by $u^i = N^i U$, where N^i is the displacement matrix which

is dependent upon element geometry. In the same way the elastic strains are given by $\epsilon^i = B^i U$, and the isotropic stress-strain relationship can be written as

$$\sigma^i = D\epsilon^i = DB^i U, \quad (7)$$

where σ^i denotes the elastic stresses and D is a matrix of elastic constants.

The creep strain rates $\dot{\nu}^i$ are assumed to depend only on the current stress and are expressed by an equation similar to (6). The total strain rate $\dot{\epsilon}^i$ is composed of an elastic component $\dot{\epsilon}^i$ and a creep component $\dot{\nu}^i$:

$$\dot{\epsilon}^i = B^i \dot{U} = \dot{\epsilon}^i + \dot{\nu}^i.$$

Since σ^i is the stress due to the elastic strain ϵ^i , then

$$DB^i \dot{U} = \dot{\sigma}^i + D\dot{\nu}^i. \quad (8)$$

To solve the stationary-state creep problem it is convenient to introduce a residual stress field ρ^i , for element i , which varies with time t , giving the stress at t to be

$$\sigma^i = \dot{\sigma}^i + \rho^i. \quad (9)$$

Differentiation of (9) with respect to time and substitution into (8) yields

$$\dot{\rho}^i = \dot{\sigma}^i = DB^i \dot{U} - D\dot{\nu}^i. \quad (10)$$

Now consider the boundary value problem for which the nodal load vector f is given in terms of the applied load vector r by

$$f = \sum_i \int_{S_i} (N^i)^T r \, dS.$$

S_i denotes the element of surface corresponding to the i th finite element. The principle of virtual work may now be applied to the boundary value problem:

$$\begin{aligned} U^T \frac{df}{dt} &= \sum_i \int_{V_i} \epsilon^{iT} \dot{\sigma}^i \, dV, \\ &= \sum_i \int_{V_i} (B^i U)^T (DB^i \dot{U} - D\dot{\nu}^i) \, dV, \\ &= \sum_i U^T \left[\left(\int_{V_i} B^{iT} DB^i \, dV \right) \dot{U} \right. \\ &\quad \left. - \left(\int_{V_i} B^{iT} D \, dV \right) \dot{\nu}^i \right], \end{aligned} \quad (11)$$

where V_i denotes the volume of element i . For steady applied loads df/dt is zero and on substitution of $K = \sum_i B^{iT} DB^i \, dV$ for the stiffness matrix, and using $A^i = \int B^{iT} D \, dV$, (11) becomes

$$K \dot{U} = \sum_i A^i \dot{\nu}^i. \quad (12)$$

The stress rates are then given by (10).

The elastic stresses at $t = 0$ may be derived from the nonrate equivalent of (11):

$$\begin{aligned} U^T f &= \sum_i U^T \left[\left(\int_{V_i} B^{iT} DB^i \, dV \right) U \right] \\ \text{or} \quad f &= KU \end{aligned} \quad (13)$$

The stationary-state creep solution is obtained by first calculating the elastic stresses from (13) and (7), and then by time integration of (12). The integration procedure used was a fourth-order Runge-Kutta method with Merson's error estimate. Acceptance or rejection of the time increment was carried out as described in a previous publication [5]. The parameter C , defined by

$$C = \frac{\max_i \{ \|DB^i \dot{U} - D\dot{\nu}^i\|_\infty \}}{\max_i \{ \|DB^i \dot{U}\|_\infty, \|D\dot{\nu}^i\|_\infty \}},$$

where $\|a\|_\infty = \max_j \{ |a_j| \}$ denotes the max-norm of vector a , was introduced to determine when the stationary state had been reached. C varies between 0 and 2: the closer to zero it is the nearer one is to the stationary state. The stationary state was said to have been achieved when $C \leq 1.0 \times 10^{-3}$.

Computation of J

Figure 6 depicts the generated mesh used to model one quarter of the double-edge-cracked plate. The semi-contour used for the evaluation of J is shown, as is the crack. The contour is defined to be composed of linear segments joining mesh nodes running along element edges. The stress and strain rates, constant within each element, are discontinuous across element boundaries. With this formu-

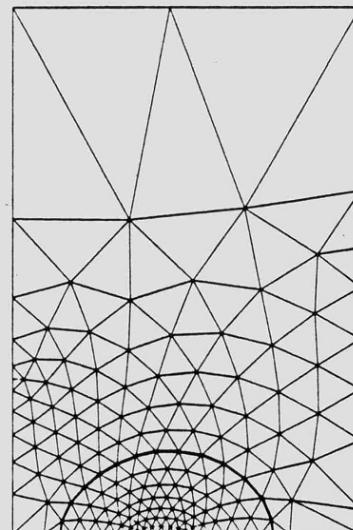


Fig. 6 Mesh used to model double-edge cracked geometry with crack and semi- Γ -contour shown.

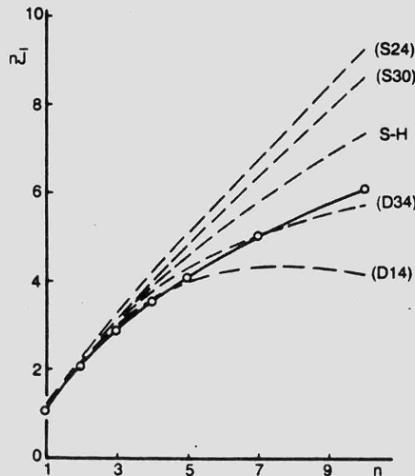


Fig. 7 Comparison of $n\bar{J}$ values for center-cracked tension specimen (○—○—○ Brown-Hayhurst).

lation, \dot{u} varies linearly with x_i within each element so that $\partial \dot{u} / \partial x_i$ is also constant within each element. Thus all the constituent parts of the integrand which defines J are discontinuous across the contour Γ . In the numerical evaluation of J such quantities are averaged between the two elements that contain a line segment of Γ . J is determined by a summation of the integrand values along all segments of Γ .

Results and Discussion

Values of J have been computed for the geometry defined by $a/b = 1/2$ and $h/b = 1.5$ with values of $n = 1, 2, 3, 4, 5, 7$, and 10 for both center-cracked and double-edge-cracked geometries. The computed values of J are presented in the normalized form $n\bar{J}$ in Figs. 7 and 8. The variation of J with n is strongly nonlinear and it is convenient to introduce the normalization

$$\bar{J} = J/(a\sigma_0 \epsilon_0 \bar{\sigma}^{(n+1)}) \quad \text{and} \quad \bar{\sigma} = b\sigma_\infty/(b-a)\sigma_0$$

For convenience the stress used to normalize the material data σ_0 will be chosen to be σ_∞ , the uniform boundary stress applied to the plate. The computed results presented in Figs. 7 and 8 are compared with results published by Ranaweera and Leckie [6, 7] (R-L) and by Shih and Hutchinson [8] (S-H) for the same problems. The results of several calculations are presented due to R-L; they are referred to by a letter followed by a number. The letter, either D or S, refers to a displacement or stress finite element formulation respectively and the number refers to the number of elements used, which were quadratic in the corresponding variable. The results of R-L and of S-H were obtained using singular crack-tip elements.

The results computed in this investigation are either in agreement with or slightly below the R-L results obtained for the displacement method. The maximum difference occurs for $n = 3 - 5$, when the values are approximately 3 percent below. The results compare surprisingly well

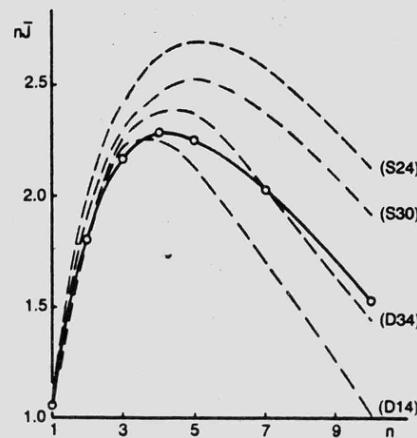


Fig. 8 Comparison of $n\bar{J}$ values for double-edge-cracked tension specimen (○—○—○ Brown-Hayhurst).

considering that the mesh used does not employ a singular crack tip element and that the simplest element type was used. The maximum number of elements used was 298, which compares with the 700 linear displacement type of elements used by S-H.

Comparison of the computed values of $n\bar{J}$ with those of R-L and S-H, while encouraging, does not recommend the use of this particular mesh generation technique to model singular displacement fields in continuum mechanics. However, the results are sufficiently good to recommend the use of the mesh generation method for problems in continuum mechanics where moderately low spatial variations in the field variables can be expected; for example, problems in stress analysis where stress-concentration factors of five or less are encountered. ■

References

- Carrier, G., Krook, M., and Pearson, C., *Functions of a Complex Variable*, New York: McGraw-Hill, 1966.
- Brown, P.R., "A Non-Interactive Method for the Automatic Generation of Finite Element Meshes using the Schwarz-Christoffel Transformation," *Computer Methods in Applied Mechanics and Engineering*, Vol. 25, 1981, pp. 101-26.
- Rice, J.R., "A Path-Independent Integral and the Approximate Analysis of Stress Concentration by Notches and Cracks," *Journal of Applied Mechanics*, Vol. 35, 1968, pp. 379-86.
- Ponter, A.R.S., and Brown, P.R., "The Finite Element Solution of Rapid Cycling Creep Problems," *International Journal for Numerical Methods in Engineering*, Vol. 12, 1978, pp. 1001-24.
- Hayhurst, D.R., and Henderson, J.T., "Creep Stress Redistribution in Notched Bars," *International Journal of Mechanical Sciences*, Vol. 19, 1977, pp. 133-46.
- Ranaweera, M.P., and Leckie, F.A., "The Use of Optimization Techniques in the Analysis of Cracked Members by the Finite Element Displacement and Stress Methods," *Computer Methods in Applied Mechanics and Engineering*, Vol. 19, 1979, pp. 367-89.
- Ranaweera, M.P., and Leckie, F.A., "J Integrals for Some Crack and Notch Geometries," *International Journal of Fracture*, Vol. 18, No. 1, Jan. 1982, pp. 3-18.
- Shih, C.F., and Hutchinson, J.W., "Fully Plastic Solutions and Large Scale Yielding Estimates for Plane Stress Crack Problems," *Journal of Engineering Materials and Technology*, Vol. 98, Oct. 1976, pp. 289-95.

APPENDIX 3

Derivation of the expressions for a_2 and b_2

In this appendix appears the derivation of the expressions for a_2 and b_2 , the scaling and positioning constants which appear in (2-6) and for which a numerical formula is given in (3-7).

The notation $\alpha, \beta, \gamma, \delta, P, Q, T, U, V, W, Z$ is used in this section only and has no connection with similar notation used elsewhere.

The quantity to be minimised is given in (3-6) and is

$$S = \sum_{k=0}^{n-1} |a_2 v_k + b_2 - w_k|^2 .$$

a_2 and b_2 are complex. Let them be denoted by

$$a_2 = \alpha + i\beta, \quad b_2 = \gamma + i\delta.$$

where α, β, γ and δ are real. Then

$$S = \sum \{ [\alpha \operatorname{Re}(v_k) - \beta \operatorname{Im}(v_k) + \gamma - \operatorname{Re}(w_k)]^2 + [\alpha \operatorname{Im}(v_k) + \beta \operatorname{Re}(v_k) + \delta - \operatorname{Im}(w_k)]^2 \} .$$

When S attains its minimum, $\partial S / \partial \alpha = \partial S / \partial \beta = \partial S / \partial \gamma = \partial S / \partial \delta = 0$ giving four equations for α, β, γ and δ :

$$P\alpha + Q\gamma + T\delta = U,$$

$$P\beta - T\gamma + Q\delta = V,$$

$$Q\alpha - T\beta + n\gamma = W,$$

$$T\alpha + Q\beta + n\delta = Z,$$

(a3-1)

where

$$\begin{aligned}
P &= \sum [\operatorname{Re}(v_k)^2 + \operatorname{Im}(v_k)^2] = \sum |v_k|^2, \\
Q &= \sum \operatorname{Re}(v_k), \\
T &= \sum \operatorname{Im}(v_k), \\
U &= \sum [\operatorname{Re}(w_k) \operatorname{Re}(v_k) + \operatorname{Im}(w_k) \operatorname{Im}(v_k)], \\
V &= - \sum [\operatorname{Re}(w_k) \operatorname{Im}(v_k) - \operatorname{Im}(w_k) \operatorname{Re}(v_k)], \\
W &= \sum \operatorname{Re}(w_k), \\
Z &= \sum \operatorname{Im}(w_k).
\end{aligned}$$

From these relationships, it can be seen that

$$\begin{aligned}
U + iV &= \sum w_k v_k^*, & Q + iT &= \sum v_k, \\
W + iZ &= \sum w_k, & Q^2 + T^2 &= |\sum v_k|^2.
\end{aligned}$$

The matrix of the linear system in (a3-1) can be inverted analytically and the solution written as

$$\begin{bmatrix} \alpha \\ \beta \\ \gamma \\ \delta \end{bmatrix} = \frac{1}{n P - Q^2 - T^2} \begin{bmatrix} n U - Q W - T Z \\ n V - Q Z + T W \\ P W - Q U + T V \\ P Z - Q V - T U \end{bmatrix}$$

so that

$$\begin{aligned}
a_2 &= \alpha + i\beta = \frac{n (U + iV) - (Q - iT)(W + iZ)}{n P - Q^2 - T^2} \\
&= \frac{n \sum w_k v_k^* - \sum v_k^* \sum w_k}{n \sum |v_k|^2 - |\sum v_k|^2},
\end{aligned}$$

and

$$b_2 = \gamma + i\delta = \frac{P (W + iZ) - (Q + iT)(U + iV)}{n P - Q^2 - T^2}$$

$$= \frac{\sum |v_k|^2 \sum w_k - \sum v_k \sum w_k v_k^*}{n \sum |v_k|^2 - |\sum v_k|^2}$$

as stated.

REFERENCES

1. I.A. Nazlawy, "Computer aided generation of finite element meshes", M. Phil thesis, Leicester University (1974).
2. O.C. Zienkiewicz and D.V. Phillips, "An automatic mesh generation scheme for plane and curved surfaces by 'isoparametric' co-ordinates", International Journal for Numerical Methods in Engineering, volume 3, pp 519-528 (1971).
3. W.J. Gordon and C.A. Hall, "Construction of curvilinear co-ordinate systems and applications to mesh generation", International Journal for Numerical Methods in Engineering, volume 7, pp 461-477 (1973).
4. R. Haber, M.S. Shephard, J.F. Abel, R.H. Gallagher and D.P. Greenberg, "A general two-dimensional, graphical finite element preprocessor utilizing discrete transfinite mappings", International Journal for Numerical Methods in Engineering, volume 17, pp 1015-1044 (1981).
5. C.O. Frederick, Y.C. Wong and F.W. Edge, "Two-dimensional automatic mesh generation for structural analysis", International Journal for Numerical Methods in Engineering, volume 2, pp 133-144 (1970).
6. J.C. Cavendish, "Automatic triangulation of arbitrary planar domains for the finite element method", International Journal for Numerical Methods in Engineering, volume 8, pp 679-696 (1974).
7. J. Fukuda and J. Suhara, "Automatic mesh generation for finite element analysis", in J.J. Oden et al. (eds.), "Advances in computational methods in structural mechanics and design", UAH Press, Huntsville, Alabama (1972).
8. A. Bykat, "Automatic generation of triangular grid: I - subdivision of a general polygon into convex subregions; II - triangulation of convex polygons", International Journal for

- Numerical Methods in Engineering, volume 10, pp 1329–1342 (1976).
- 9. E.A. Sadek, "A scheme for the automatic generation of triangular finite elements", International Journal for Numerical Methods in Engineering, volume 15, pp 1813–1822 (1980).
 - 10. R.D. Shaw and R.G. Pitchen, "Modification to the Suhara-Fukuda method of network generation", International Journal for Numerical Methods in Engineering, volume 12, pp 93–99 (1978).
 - 11. S.H. Lo, "Delaunay triangulation of non-convex planar domains", International Journal for Numerical Methods in Engineering, volume 28, pp 2695–2707 (1989).
 - 12. R. Sibson, "Locally equiangular triangulations", Computer Journal, volume 21, pp 243–245 (1978).
 - 13. L.R. Herrmann, "Laplacian-isoparametric grid generation scheme", Journal of the Engineering Mechanics Division, Proceedings of the American Society of Civil Engineers, volume 102, pp 749–756 (1976).
 - 14. M.S. Shephard and M.A. Yerry, "An approach to automatic finite element mesh generation", in L.E. Hulbert (ed.), "Computers in Engineering", pp 21–28, Proceedings of International Conference, San Diego, California, 15–19 August, 1982, pp 21–28, ASME, New York (1982).
 - 15. D.R. Hayhurst and P.R. Brown, "The teaching of the finite element method for use in component analysis and design", International Journal of Mechanical Engineering Education, volume 11, number 4, pp 263–284 (1982).
 - 16. H. Lamb, "Hydrodynamics", 6th edition, Cambridge University Press (1932).
 - 17. L.M. Milne-Thomson, "Theoretical hydrodynamics", 5th edition, Macmillan, London (1968).
 - 18. P. Henrici, "Applied and computational complex analysis", volume 1,

- Wiley-Interscience, New York (1974).
19. L.N. Trefethen, "Numerical computation of the Schwarz-Christoffel transformation", SIAM Journal on Scientific and Statistical Computing, volume 1, pp 82-102 (1980).
 20. A.R.S. Ponter and P.R. Brown, "The finite element solution of rapid cycling creep problems", International Journal for Numerical Methods in Engineering, volume 12, pp 1001-1024 (1978).
 21. O.C. Zienkiewicz, "The finite element method", 3rd edition, McGraw-Hill, London (1977).
 22. I.D. Felce, "A tri-axial tension creep specimen and test machine", M. Phil thesis, Leicester University (1983).
 23. A.R.S. Ponter and P.R. Brown, "Creep indentation solutions: variation of reference stress parameters with geometry", Leicester University Engineering Department internal report, number 85-9 (1985).
 24. O.C. Zienkiewicz, Y.C. Liu and G.C. Huang, "Error estimation and adaptivity in flow formulation for forming processes", International Journal for Numerical Methods in Engineering, volume 25, pp 23-42 (1988).
 25. M. Yokoyama, "Automated computer simulation of two-dimensional elastostatic problems by the finite element method", International Journal for Numerical Methods in Engineering, volume 21, pp 2273-2287 (1987).
 26. B.-N. Jiang and G.F. Carey, "Adaptive refinement for least-squares finite elements with element-by-element conjugate gradient solution", International Journal for Numerical Methods in Engineering, volume 24, pp 69-80 (1987).