DEPARTMENT OF DEFENCE
DEFENCE SCIENCE & TECHNOLOGY ORGANISATION
DSTO

DSTO

Advances in Structural Loadflow Visualisation and Applications to Optimal Shapes

W. Waldman, M. Heller and L.R.F. Rose

DSTO-RR-0166

Advances in Structural Loadflow Visualisation and Applications to Optimal Shapes

W. Waldman, M. Heller, R. Kaye and L.R.F. Rose

Airframes and Engines Division
Aeronautical and Maritime Research Laboratory

DSTO-RR-0166

ABSTRACT

Currently there is no generally accepted procedure for calculation of structural loadpaths, which would show how remote loads are equilibrated through a structure and could provide insight into how well a structure is performing its intended load-carrying functions. Kelly and Elsley have recently proposed a method for computing loadflow orientations and loadpaths using finite element results, which is based on iterative solutions of non-linear equations. In this paper, we have enhanced their theoretical formulation and general procedure by deriving explicit expressions for computing loadflow orientations. The new equations produce more accurate loadflow orientations compared to the prior approach and improve the fidelity of calculated loadpaths. In a series of benchmark problems, we have investigated non-optimal and optimal holes in plates using loadflow visualisation to identify their key features. We found that recirculation is apparent for non-optimal hole shapes, whereas no recirculation zone is present for optimal shapes. Although very highly refined finite element meshes were utilised, the implications are that even more refined meshes are required to fully capture the complex behaviour that exists in recirculation zones. The removal of the recirculation zone for a non-optimal shape leads to a better shape, but the improvement in peak stress is insignificant. The calculation of loadflow orientations using the new equations is simple, and could be used with any finite element analysis code, while a plotting package is required to display loadpaths. Loadflow visualisation is a powerful tool for use by structural designers to improve their understanding of structural performance, the application of which can potentially result in worthwhile improvements in structural efficiency.

RELEASE LIMITATION

Approved for public release

DEPARTMENT OF DEFENCE
DEFENCE SCIENCE & TECHNOLOGY ORGANISATION
DSTO

Published by

DSTO – Aeronautical and Maritime Research Laboratory
PO Box 4331
Melbourne Victoria 3001
Australia

Telephone: (03) 9626-7000
Fax: (03) 9626-7999

AR No. AR-011-110
October 1999

APPROVED FOR PUBLIC RELEASE

Advances in Structural Loadflow Visualisation and Applications to Optimal Shapes

Executive Summary

The use of loadflow visualisation to assist the structural analyst in establishing the performance of structures under load is an emerging new field. Recent developments in methodology and computer implementation of loadflow visualisation, through the use of finite element analysis (FEA) results, offer the potential for useful engineering insights into the way a given structure is performing its intended function. This holds promise for the use of loadflow visualisation techniques in the evaluation of airframe performance, with expected applications to life assessment and the development of life extension strategies for aircraft. Hence, loadflow visualisation may be particularly valuable for highly stressed airframes, leading to longer-term outcomes such as increased residual life and cost benefits due to longer inspection intervals.

Currently there is no generally accepted procedure for calculation of structural loadpaths, which would show how remote loads are equilibrated through a structure and could provide insight into how well a structure is performing its intended load-carrying functions. Kelly and Elsley have recently proposed a method for computing loadflow orientations and loadpaths using finite element results, which is based on iterative solutions of non-linear equations. In this paper, we have enhanced their theoretical formulation and general procedure by deriving explicit expressions for computing loadflow orientations. The new equations produce more accurate loadflow orientations compared to the prior approach and improve the fidelity of calculated loadpaths. In a series of benchmark problems, we have investigated non-optimal and optimal holes in plates using loadflow visualisation to identify their key features. We found that recirculation is apparent for non-optimal hole shapes, whereas no recirculation zone is present for optimal shapes. Although very highly refined finite element meshes were utilised, the implications are that even more refined meshes are required to fully capture the complex behaviour that exists in recirculation zones. The removal of the recirculation zone for a non-optimal shape leads to a better shape, but the improvement in peak stress is insignificant. The calculation of loadflow orientations using the new equations is simple, and could be used with any finite element analysis code, while a plotting package is required to display loadpaths. Loadflow visualisation is a powerful tool for use by structural designers to improve their understanding of structural performance, the application of which can potentially result in worthwhile improvements in structural efficiency.

The developments in loadflow visualisation reported here provide structural analysts in Airframes and Engines Division (AED) with the capability and the tools leading to a better understanding of the performance of structural components. This will enable more comprehensive analysis of critical components in RAAF aircraft, resulting in potential extensions to fatigue life and increased inspection intervals.

Authors

Witold Waldman

Airframes and Engines Division

Witold Waldman completed a BEng (with distinction) in Aeronautical Engineering from the Royal Melbourne Institute of Technology in 1981. He commenced work in Structures Division at the Aeronautical Research Laboratory in 1982. He has published a number of papers and reports, and his experience has focussed on finite element stress analysis, structural mechanics, computational unsteady aerodynamics, structural dynamics testing, digital filtering, non-linear optimisation and spectral analysis. His recent work has been in the areas of structural shape optimisation and loadflow visualisation. He has been a Senior Professional Officer (Grade C) in the Airframes and Engines Division since 1989.

Dr Manfred Heller

Airframes and Engines Division

Manfred Heller completed a BEng (Hons.) in Aeronautical Engineering at the University of New South Wales in 1981. He commenced work in Structures Division at the Aeronautical Research Laboratory in 1982. He was awarded a Department of Defence Postgraduate Cadetship in 1986, completing a PhD at Melbourne University in 1989. He has an extensive publication record focussing on the areas of stress analysis, fracture mechanics, fatigue life extension methodologies and experimental validation. Since 1992 he has led tasks that develop and evaluate techniques for extending the fatigue life of RAAF aircraft components and provide specialised structural mechanics support to the RAAF. He is currently a Senior Research Scientist in the Airframes and Engines Division.

Authors *(continued)*

Robert Kaye
Airframes and Engines Division

Having completed a BEng at the University of New South Wales in 1981, Robert Kaye joined DSTO at AMRL in 1990 as a structural engineer with a background in full-scale testing. Since that time, he has mostly been occupied with the evaluation of airframe structural details and bonded repairs using finite element methods. This has included the analysis of repairs to fuselage skin lap-joints, wing skin planks and bulkhead frames. He has also been involved in the structural and mechanical development aspects of full-scale static and fatigue test installations. He is currently a Senior Professional Officer (Grade C) in the Airframes and Engines Division.

Dr Francis Rose
Airframes and Engines Division

Francis Rose graduated with a BSc (Hons.) from Sydney University in 1971, and a PhD from Sheffield University, UK, in 1975. He was appointed as a Research Scientist at the Aeronautical Research Laboratory in 1976, and is currently the Research Leader in Fracture Mechanics in the Airframes and Engines Division. He has made important research contributions in fracture mechanics, non-destructive evaluation and applied mathematics. He is the regional editor for the International Journal of Fracture and a member of the editorial board of Mechanics of Materials. He is also a Fellow of the Institute for Applied Mathematics and its Applications, UK, and a Fellow of the Institution of Engineers, Australia.

Contents

1. INTRODUCTION 1
2. THEORY 3
   2.1 Analysis of loadflow orientations 3
   2.2 Principal stress directions 5
   2.3 FEA method for displaying loadflow orientations and loadpaths 6
3. BENCHMARK PROBLEMS FOR HOLES IN PLATES 6
   3.1 A circular hole in a uniaxial stress field 7
      3.1.1 Geometry and loading 7
      3.1.2 Principal stress directions 7
      3.1.3 Loadflow orientations 8
      3.1.4 Loadpath contours 8
      3.1.5 Calculation of K_t from loadpath compaction 10
   3.2 A circular hole in a 4:1 biaxial stress field 10
      3.2.1 Geometry and loading 11
      3.2.2 Loadflow orientations 11
      3.2.3 Loadpath contours 11
   3.3 An optimal elliptical hole in a 4:1 biaxial stress field 12
      3.3.1 Geometry and loading 12
      3.3.2 Loadflow orientations 12
      3.3.3 Loadpath contours 13
4. PREDICTION OF INITIAL REWORK SHAPES 13
   4.1 Reworked circular hole in a uniaxial stress field 13
   4.2 Reworked circular hole in a 4:1 biaxial stress field 14
5. DISCUSSION AND CONCLUSION 14
6. ACKNOWLEDGEMENT 16
7. REFERENCES 16

APPENDIX A EXTRACTING STRESSES FROM NASTRAN FEA RESULTS 43

APPENDIX B DETERMINING LOADFLOW ORIENTATIONS 49

APPENDIX C LOADPATH VISUALISATION USING TECPLOT 57

1. Introduction

The use of loadflow visualisation to assist the structural analyst in establishing the performance of structures under load is an emerging new field. Loadflow visualisation can be considered to consist of two distinct categories: i) display of loadflow orientations at discrete points in a body, and ii) display of loadpaths (trajectories) taken through a body by unit quantities of load. Recent developments in methodology and computer implementation of loadflow visualisation, through the use of finite element analysis (FEA) results, offer the potential for useful engineering insights into the way a given structure is performing its intended function. This holds promise for the use of loadflow visualisation techniques in the evaluation of airframe performance, with applications to life assessment and the development of life extension strategies for aircraft, such as shape optimisation of reworks [1]. Thus, loadflow visualisation may be particularly valuable for highly stressed airframes, leading to longer-term outcomes such as increased residual life and cost benefits due to longer inspection intervals.

There is a very limited amount of published prior work on loadflow and its visualisation. In the past, the orientations of the major principal stresses throughout a body have often been used to try to depict the flow of load through a structure. Making use of photo-elasticity in an early paper, Baud [2] noted that the mapping of principal stress direction lines in structures that have stress concentrations could be a substantial aid in explaining stress concentration. However, it has been clearly demonstrated by Kelly and Elsley [3] [4] that, when used to represent loadflow orientations, the principal stress vectors can have an angular error of up to 45° in regions of high shear stress.

In other work, Chaperon et al. [5] have used a displacement-based formulation for loadflow visualisation that is based on the orientation of energy-flow vectors to study optimal shapes, but neither loadpath lines nor compaction of loadpaths were investigated. An optimal shape is defined as one that minimises the peak stress concentration factor (SCF) in the structure for the chosen loading condition. They identified "recirculation" zones and considered that these could be a feature of a non-optimal hole shape with redundant material present that might possibly be removed with only a minimal effect on the loadflow in the structure. This concept may assist in producing an initial design for subsequent shape optimisation of the structure. However, the energy-flow vector approach has shown itself to be sensitive to the location of reference axes, and it has problems in handling symmetry conditions, which are routinely utilised to reduce the computational size of finite element (FE) models. Follow-on work by Ibrahim et al. [6] and Jones and Chen [7] on the use of energy density vectors has overcome the symmetry-related problems that complicate the use of energy-flow vectors.

Furthermore, much of the available literature is predominantly concerned with qualitative descriptions of loadflow without any method of calculation. In two engineering publications dealing extensively with stress concentration factors,

Peterson [8] and Pilkey [9] refer to a flow analogy when discussing aspects of stresses around notches. However, the pictures of "flow lines" in and around notches, shoulder fillets, T-heads and nuts are purely of a schematic nature. The concept of "force lines," where the spacing of the flow lines may be regarded as being inversely proportional to the intensity of the stress, is similarly utilised by Osgood [10] to help visualise the effects of notches and fillets on stress distribution. Broek [11] also considers loadpath (loadflow) lines around notches, holes, and fillets where these discontinuities cause a deviation in the loadflow lines, leading to stress concentration. In his work on minimising stress concentrations in fillets, Baud [12] used a hydrodynamical fluid flow analogy to show that a streamline fillet, whose contour is the same as the contour for a free jet of water issuing from a circular orifice, produces a low SCF. However, Neuber, in his treatment of the theory of notch stresses [13], has used the integration of stress components to analytically calculate the lines of stress around an internal notch for a case where an analytical solution was feasible.

Kelly and Elsley [3] [4] have recently made significant inroads into the visualisation of loadflow. They have developed a theoretical basis for calculating loadflow orientations at discrete points within general structures by iterative solution of non-linear transcendental equations. Their method relies on using the results from a FEA, computed at the nodes and/or element centroids in the FE mesh. Kelly and Elsley demonstrated that their method is accurate in regions of high shear stress, unlike loadpath visualisation based on the use of principal stress trajectories. Along the lines of Neuber [13], they implemented an automated procedure for integrating the stresses obtained from the FEA solution for determining loadpath contours, and they successfully applied this to a cracked plate. In later work, Kelly [14] demonstrated how loadflow visualisation could assist in defining an efficient FE mesh, where the orientation of the mesh is aligned with the loadpaths through the structure. Kelly and Tosh [15] have also studied the potential for using loadpath lines to help with the optimal placement of fibres in fibre-reinforced composite laminates. In recent unpublished work, Kelly and Tosh [16] have bypassed the inconvenience of the integration method of computing loadpaths by using directly the vector-like field of loadflow orientations calculated from the FEA stress values.

Initially, this paper reports on a key result extending Kelly and Elsley's [3] [4] loadflow visualisation procedure by developing explicit expressions for calculating loadflow orientation, and these are given in Section 2. The derivation presented here also disperses a degree of confusion arising from reference to the major and minor principal stress angles in the presentation of the original theory. In Section 3, the loadflow orientations and loadpaths around non-optimal and optimal rework shapes are investigated using the new simplified equations, which have obviated the need for an iterative numerical solution. This is an area that has not previously been considered in depth. By using much more refined FE meshes than those analysed by other workers, it was possible to perform loadflow visualisation at a greater level of detail than has previously been the case. In producing the loadpath lines, the new method proposed by Kelly and Tosh [16] was applied, and a detailed description of the methodology is given here. The use of the compaction of loadpath lines to compute stress concentration factors is also demonstrated. In Section 4 the potential for using recirculation zones that are present in non-optimal shapes to lead to first estimates of

optimal shapes for use in shape optimisation procedures is investigated. A discussion and some conclusions are presented in Section 5.

2. Theory

2.1 Analysis of loadflow orientations

Kelly and Elsley [3] [4] have developed a procedure that can be applied to the calculation of loadflow orientations within a structure, using the results obtained from a FEA. We will present the theory from the start and make significant improvements along the way. Although Kelly and Elsley's logic will be retained, we will make use of the notation that is commonly found in standard texts on the theory of elasticity in order to make the derivation presented here easier to understand.

Kelly and Elsley defined a loadpath to be the trajectory taken within a structure by a unit quantity of applied load, beginning at a point of application and ending at an equilibrating reaction. They introduced the concept of a hypothetical force "stream tube", which is depicted in Figure 1. For any selected resolved force direction, depicted as the x-direction in Figure 1, there is no flow of load across the boundary of the force tube, and the loadpath is therefore bounded by lines along which there is no contribution to the force in the x-direction. Because equilibrium must be satisfied over the tube, we have that

$$F_{xa} - F_{xb} = 0 \tag{1}$$

Note that in Figure 1 the applied and reaction loads F_{xa} and F_{xb} are aligned with the x-direction, but an equivalent diagram can also be produced for the forces in the y-direction (or any other chosen direction).

On a plane that is tangential to the force tube wall there are two components of stress acting, σ_n and τ_{nt}. Here σ_n is the local stress perpendicular to the plane and τ_{nt} is the local shear stress acting along the plane, where σ_n is oriented at an angle θ with respect to a reference x-axis, as depicted in Figure 1. It should be noted that the angle θ defined here is different to that used by Kelly and Elsley, and the derivation will be presented from the start for clarity.

The condition of force equilibrium is satisfied locally along any segment of loadpath wall if the force components due to the action of σ_n and τ_{nt} in the direction parallel to the applied loads cancel out. For applied forces resolved in the x-direction, the value $\theta = \theta_x$ that satisfies this condition is given by the solution to the equation

$$\sigma_n \cos\theta_x - \tau_{nt} \sin\theta_x = 0 \tag{2}$$

For applied forces resolved in the y-direction, the equivalent equation that must be satisfied to determine the value $\theta = \theta_y$ is

$$\sigma_n \sin\theta_y + \tau_{nt} \cos\theta_y = 0 \tag{3}$$

From the standard two-dimensional theory of elasticity [17], the following equations define the normal stress σ_n and the shearing stress τ_{nt} at a point on an arbitrary plane oriented at an angle θ with respect to a reference x-axis in terms of the known stresses σ_x, σ_y and $\tau_{xy} = \tau_{yx}$, as depicted in Figure 2.

$$\sigma_n = \sigma_x \cos^2\theta + \sigma_y \sin^2\theta + 2\tau_{xy} \sin\theta\cos\theta \tag{4}$$

$$\tau_{nt} = -(\sigma_x - \sigma_y)\sin\theta\cos\theta + \tau_{xy}(\cos^2\theta - \sin^2\theta) \tag{5}$$

In passing, it can be noted that, by substituting $\theta = 0°$ and $\theta = 90°$ into Equations (4) and (5), it is possible to obtain the original stresses σ_x, σ_y and τ_{xy}. These act on the original x- and y-planes, which are orthogonal, as shown in Figure 2. In Kelly and Elsley's work, Equations (2) and (3) were treated as being non-linear and were subsequently solved in an iterative manner, and the stresses σ_x, σ_y and τ_{xy} were obtained by FEA. However, it will be shown that recourse to a non-linear solution technique is unnecessary, as some additional algebraic manipulation presents us with explicit analytical expressions for the loadflow orientation angles, as computed in the two chosen orthogonal directions.

Apart from straightforward differences in nomenclature, it should be noted that in [3] and [4] the expression for τ_{nt} has what appears to be a typographical error. In Equation (3) therein, the term involving σ_x is missing a minus sign, although the loadflow orientations that were presented in those two references are correct.

Substituting the expressions for σ_n and τ_{nt} in Equations (4) and (5) into Equation (2), and after some algebraic manipulation and simplification, we finally obtain

$$\tan\theta_x = -\frac{\sigma_x}{\tau_{xy}} \tag{6}$$

In a similar manner, Equation (3) becomes

$$\tan\theta_y = -\frac{\tau_{xy}}{\sigma_y} \tag{7}$$

It can be seen that Equation (6) for computing the x-direction loadflow orientations is independent of σ_y, and Equation (7) for computing the y-direction loadflow

orientations is independent of σ_x. This characteristic was not previously apparent in prior work. At any given point along the force tube wall, the angle α of the loadflow orientation is given by the tangent to the force tube, as depicted in Figure 1. The angle α can therefore be expressed in terms of θ as

$$\alpha = \theta + 90° \tag{8}$$

Thus, at any point along the force tube wall, the local loadflow orientations α_x and α_y of the resolved x-direction and y-direction loadpath lines are simply

$$\alpha_x = \theta_x + 90° \tag{9}$$

$$\alpha_y = \theta_y + 90° \tag{10}$$

Using the above two relations, the equations for the actual loadflow orientations α_x and α_y become

$$\tan \alpha_x = \frac{\tau_{xy}}{\sigma_x} \tag{11}$$

$$\tan \alpha_y = \frac{\sigma_y}{\tau_{xy}} \tag{12}$$

Thus, the derivation presented here has led to the development of explicit equations for the x-direction and y-direction loadflow orientations, a result that has not previously been available. Equations (11) and (12) can easily be solved to produce α_x and α_y at any point in a two-dimensional stress field. This key result considerably simplifies the computation of loadflow orientations throughout a body, compared to the procedure requiring the solution of non-linear transcendental equations by an iterative numerical procedure that was used in [3] and [4]. This makes it particularly easy to compute the loadflow orientations from the stresses produced from a FEA of a structure.

2.2 Principal stress directions

For the purpose of comparing loadflow orientations with principal stress directions, it is convenient to include here the standard definition of the well-known major and minor principal stress angles, θ_{p1} and θ_{p2}, respectively, which correspond to the major and minor principal stresses, σ_{p1} and σ_{p2}. The two principal stress angles are determined using the two solutions to the equation

$$\tan 2\theta_p = \frac{2\tau_{xy}}{\sigma_x - \sigma_y} \tag{13}$$

Note that the two angles θ_{p1} and θ_{p2} are 90° apart, where one value of θ_p always lies between ±45° (inclusive), and the other one is 90° greater. The two principal stresses can be computed by substituting $\theta = \theta_{p1}$ and $\theta = \theta_{p2}$ into Equation (4), and the major principal stress σ_{p1} corresponds to the algebraically larger of the two stresses.

2.3 FEA method for displaying loadflow orientations and loadpaths

Although no theory has been given at this juncture, Kelly and Tosh [16] have put forward a method for taking a set of computed loadflow orientations and subsequently using the Tecplot [18] commercial data visualisation program to plot loadpaths. They have proposed that loadpaths can be considered to be analogous to streamtraces, where a streamtrace is the path traced by a massless particle placed at an arbitrary location in a steady-state vector field.

The method used here takes the nodal stresses computed from a standard FEA solution, together with the element topologies, and stores these data in files for subsequent use. Appendix A gives a detailed description of the method that was used when working with the MSC.NASTRAN FEA program and MSC.PATRAN pre- and post-processor, together with our custom-written Fortran utility programs used to create and store the required data. These are used to extract nodal and element information from the NASTRAN data deck, as well as in combining the nodal and element stress data extracted using PATRAN. The full-field stress results are then processed using our own in-house custom-written Fortran program, which is described and listed in Appendix B, and the loadflow orientations at each node in the original FE mesh are computed using the equations developed in Section 2.1. (The principal stress directions are also calculated for comparison purposes.) The results are then stored as a field of unit vectors in a Tecplot-compatible format and are subsequently read in by the Tecplot program. Using the concept that loadflow lines are analogous to streamtraces [16], Tecplot's built-in facility for plotting streamtraces was utilised to compute and display loadpaths for the structures of interest. The detailed step-by-step procedure is presented in Appendix C.

3. Benchmark Problems For Holes in Plates

The FE models of square plates with holes that were analysed here were created and meshed in MSC.PATRAN. The MSC.NASTRAN FEA program was then used to perform a linear elastic static analysis to compute the stresses in the structures of interest, although other general purpose FEA computer software could also have been used. The computer that was used was a dual-processor Hewlett-Packard K260 server. Four-noded quadrilateral finite elements were used for all analyses, which were conducted under two-dimensional plane stress conditions. The material properties

were those for aluminium, with Young's modulus E = 70×10³ MPa and Poisson's ratio ν = 0.30. In all the FE models analysed here, quarter symmetry was used for the displacement restraints, which consisted of $u_x = 0$ along the y-axis and $u_y = 0$ along the x-axis. Although only one quarter of the plate could have been modelled in the FEA (due to symmetry), it was decided to use complete plate models in order to retain flexibility in the post-processing and displaying of results. One of the aims was to allow a full view of the loadpaths through the structure if desired.

The meshes that were used were very refined, particularly near the holes, in order to accurately calculate the stresses in areas of high stress gradient. In contrast, the FE meshes used by Kelly and Elsley [3] [4] were at least an order of magnitude coarser. In any case, our FE models are not all that large by modern standards, and therefore their solution poses no computational difficulties. By using highly refined FE meshes it was possible to resolve the fine details in recirculation zones that would have been missed had coarser meshes been utilised.

3.1 A circular hole in a uniaxial stress field

3.1.1 Geometry and loading

The plate geometry and loading arrangement that was considered for this analysis is shown in Figure 3. Here the 200 mm square plate contains a centrally located circular hole with 20 mm diameter. A remote uniaxial 100 MPa tensile stress field aligned in the y-direction is applied to the plate along the plate edges at $y = \pm 100$ mm.

This case is considered to be a good test problem because the maximum stress concentration factor at the hole will be approximately 3.0, and therefore the hole is expected to significantly disturb the loadpath lines passing in its vicinity. The size of the plate is considered to be large in comparison to the dimensions of the hole, so the effect of the finite plate width is expected to be small. The peak stress concentration factor will occur at two points around the hole, located at $(x, y) = (\pm 10, 0)$. Another feature of this geometry is that there is a region of compressive boundary stress at the top and bottom of the hole, and the magnitude of this compressive stress is similar in value to that of the applied load. The complete mesh discretisation is shown in Figure 4a, and consists of 9200 elements. Figure 4b shows a close-up view of the mesh around the circular hole, where the highly refined nature of the mesh is evident.

3.1.2 Principal stress directions

The orientations θ_{p1} of the principal stress σ_{p1} around the circular hole are shown in Figure 5a, while Figure 5b shows the orientations θ_{p2} of the principal stress σ_{p2}. In these two figures the angles of orientation of the principal stresses are represented by the orientations of the individual short line segments. Vertical line segments signify principal stress angles of 90°, horizontal line segments represent angles of 0°, while line segments orientated at other angles represent cases lying between these two conditions.

In Figure 5a it is evident that the principal stresses σ_{p1} are largely aligned at an angle of 90°, which corresponds to the direction of the applied loading. The orientation of σ_{p1} closely follows the boundary of the hole but, as the top of the hole is approached in an anti-clockwise direction, the orientations twist around until they are largely perpendicular to the boundary of the hole. A similar effect is noted in Figure 5b, but this time the orientation of the principal stress σ_{p2} closely follows the hole boundary at the top of the hole, but it becomes aligned perpendicular to the hole boundary as we move in a clockwise sense from the top to the right-hand side of the hole. In both cases, the transition points for these changes in the orientations of the principal stresses σ_{p1} and σ_{p2} occur at the same point on the boundary of the hole.

It should be noted that, depending on the location along the free boundary of the hole, at least one of the two principal stresses, σ_{p1} and σ_{p2}, will be identically equal to zero (normal to the boundary). The non-zero principal stress will be aligned such that it is tangential to the boundary. At the four locations where the hoop stress goes from being tensile in nature to being compressive, they will both be zero. Even when a very refined mesh is used in the FE model, these conditions will usually only be approximated, due to the inherent numerical error (although small) of the FEA. Hence, in Figure 5a and Figure 5b, regions where the principal stress orientations are shown as being perpendicular to the hole boundary are in fact regions where that particular principal stress is actually zero.

3.1.3 *Loadflow orientations*

In direct comparison to the principal stress orientations, Figure 6a shows the resolved x-direction loadflow orientations, and Figure 6b shows the resolved y-direction loadflow orientations. Close-up views of the resolved x-direction and y-direction loadflow orientations around the circular hole are shown in Figure 7a and Figure 7b. In a qualitative sense, the y-direction loadflow orientations in Figure 6b are the counterpart of the σ_{p1} principal stress orientations of Figure 5a, and vice versa. Both of the loadflow orientation patterns are considerably more complex than those for the principal stress orientations. In Figure 7b it can clearly be seen that near the top of the hole there is a distinct "recirculation" region present, to use a fluid flow analogy, where the loadflow orientations quite distinctly vary through 360°. Looking along the hole boundary in Figure 7b, it is evident that the notional boundary of the recirculation zone begins at approximately the same position as the transition point described in relation to the orientations of principal stress σ_{p1} shown in Figure 5a.

3.1.4 *Loadpath contours*

The resolved x-direction and y-direction loadpaths around the circular hole are shown in Figure 8a and Figure 8b. The loadpaths were produced using the streamtrace plotting feature available in Tecplot. At this stage there is no direct theoretical basis as to why this technique could be used. However, it appears that there is some sort of

analogy between streamtraces in fluid flow and the loadpath theory that is presented here.

The resolved x-direction loadpaths in Figure 8a consist of closed curves. From this characteristic, it appears that the x-direction loadpaths are self-equilibrating, and there is no net flow of load in this resolved direction. The loadpaths were obtained by defining six streamtrace rakes radiating outwards from the origin, oriented at angles of ±90°, ±22.5°, and ±157.5° relative to the x-axis. Except for the rakes at ±90°, which were terminated at one end at $(x, y) = (0, \pm 100)$, each rake was fully contained within the plate perimeter, i.e. did not terminate anywhere on the plate edge or hole boundary. Five streamtraces were initiated from each rake. A minor degree of "spiralling" is evident in some of the loadpaths (see the darker lines in the loadpaths aligned at 157.5° and -22.5°). Loadpath spiralling occurs whenever a loadpath swirls inwards upon itself with ever decreasing radius, rather than producing a closed contour. In order to reduce the effects of spiralling for this plot, it was necessary to decrease the maximum step size used by the Tecplot streamtrace plotting integration algorithm from the default value of 0.25 of the element size to 0.125. Also, the maximum number of integration steps was reduced from the default value of 1000 to 450 in order to reduce the total pathlength of each of the loadpaths.

For the resolved y-direction loadpaths in Figure 8b, the start and end points of the loadpath lines were chosen to be uniformly distributed along the top and bottom edges of the large square plate. This results in equal quantities of load between loadpath lines. The loadpaths are largely vertically aligned over most of the plate, but they diverge around the hole, indicating that the hole acts as an impediment to the flow of load between the top and bottom of the plate. Looking at the loadpath lines that pass close to the hole, it can be seen that they first start to diverge at a distance of approximately three hole diameters from the hole centre. Looking closely at the spacing of the loadpath lines that run across the horizontal axis passing through the centre of the hole, it is evident that there is a small deflection of the loadpath lines towards the hole (this has also been depicted in [15]). This is opposite to the overall general behaviour of the loadpath lines that spread out away from the hole in order to pass around it.

Figure 9a and Figure 9b show enlarged views of the resolved x-direction and y-direction loadpaths, respectively, in the vicinity of the recirculation zone that is present near the top of the hole (see Figure 7b). In order to minimise the effects of spiralling when producing Figure 9a, the maximum integration step size was set to 0.15, and the maximum number of steps was set to 250. Also, each of the loadpaths was started just outside the visible plot area for additional control. Looking at the two loadpaths near the top of the hole in Figure 9b, spiralling is evident for the unterminated contour that runs around the closed contour. After considerable manual intervention, this was the best that could be achieved for this plot, even after adjustment of Tecplot streamtrace integration step size and total number of steps. Because the mesh is large in relative terms compared to the detail that needs to be shown, it is anticipated that additional mesh refinement in this area would assist in producing better loadpath contours in this small region that is exhibiting complex loadflow recirculation behaviour. It has also been our experience that, when originally using the non-linear solution method to

obtain the loadflow orientations, the lower accuracy adversely affected the production of the loadpath contours in some regions of the structure, sometimes causing the loadpaths to be prematurely terminated.

3.1.5 *Calculation of K_t from loadpath compaction*

When looking at the loadpath contours in Figure 8b, it is apparent that regions where the loadpaths are compacted together are indicative of areas that are experiencing higher stress levels. By comparing the relative spacing (i.e. compaction) of the loadpath contours in the vicinity of the hole to the spacing at the edges of the plate where the load is applied, it is possible to calculate the SCF at different distances from the hole. This enables an independent check of the loadpath contours to be conducted.

Figure 10a shows a circular hole of diameter $2a$ in a square plate that is loaded along the top and bottom edges by a uniform stress S. For this same loading condition, Figure 10b shows an elemental force tube whose width is Δ_b centred on a location $x = x_b$ at the top and bottom edges of the plate. As it passes around the hole, the force tube compacts and its width becomes Δ_c centred on a location $x = x_c$ along the x-axis. The SCF at the location $x = x_c$ along the x-axis is given by

$$\mathrm{SCF}(x_c) = \frac{\Delta_b}{\Delta_c} \tag{14}$$

For the uniaxially loaded square plate with a circular hole, and using Equation (14), the SCF values computed from loadflow compaction measurements at different positions along the x-axis from the edge of the hole to the edge of the plate are given in Table 1. When performing these calculations, the force tube width along the x-axis was chosen first, and then the loadpath lines were computed and their width measured at the loaded edges of the plate. This permitted better control of the loadpath lines to enable them to be calculated very close to the edge of the hole. Table 2 gives the SCF values that were computed from nodal stresses in the FE model.

Taking these two sets of results, Figure 11 compares the SCF computed from compaction of loadpath lines with that obtained from nodal stresses in the FE model. It is apparent from these results that the SCF as computed from loadpath compaction is in very good agreement with that obtained from the nodal stresses. As a result of the finite width of the plate, the value of the SCF at the edge of the hole computed from the FE results is 3.12. (It should be noted that for a circular hole in an infinite plate the expected theoretical value of SCF is 3.00.) The SCF obtained at the plate edges at $(x, y) = (\pm 100, 0)$ is 0.96, also as a result of finite-width effects.

3.2 A circular hole in a 4:1 biaxial stress field

This particular problem, consisting of a circular hole in a $y{:}x$ = 4:1 biaxial stress field, was used because the circular hole is clearly a non-optimal shape for this loading condition, and we were interested in contrasting the results with those for an optimal

hole at a later stage. It is well known that an elliptical shape in an infinite plate is optimal for this case, significantly reducing the SCF, and the elliptical hole case is subsequently studied in Section 3.3.

3.2.1 *Geometry and loading*

The plate geometry and loading arrangement that was considered for this analysis is shown in Figure 12. Here the 200 mm square plate contains a circular centrally located hole of 20 mm diameter. A remote 4:1 biaxial stress field is applied to the plate, where the applied tension stresses are 25 MPa along the edges $x = \pm 100$ mm and 100 MPa along the edges $y = \pm 100$ mm. A feature of this geometry is that there is a region of compressive boundary stress at the top and bottom of the hole. The highly refined mesh shown earlier in Figure 4 was once again utilised for the FEA. For an infinite plate the peak stress at the edge of the hole at $(x, y) = (\pm 10, 0)$ is known to be 275 MPa, indicating that there is a severe stress concentration effect caused by the presence of the hole. The results from the FEA, which include finite-width effects, gave a slightly greater peak stress of 284.7 MPa (3.5% higher). Also, the stress around the boundary of the hole is markedly non-uniform, which indicates that by choosing a more optimal shape the stress concentration effect of the hole could be reduced [1].

3.2.2 *Loadflow orientations*

Figure 13a shows the resolved x-direction loadflow orientations, where one small and distinct recirculation zone is present and is located at the top of the hole. This is in contrast to the uniaxial case, which has two large regions of recirculation in this quadrant of the plate. Figure 13b shows the resolved y-direction loadflow orientations, and a small region of recirculation can be identified. This is in contrast to the uniaxial case where a very distinct recirculation zone was present. Although the largest component of the applied load is that which is applied in the y-direction, it is noteworthy that it is the picture of the x-direction loadflow orientations that more distinctly shows the presence of a recirculation zone.

3.2.3 *Loadpath contours*

The resolved x-direction loadpaths around the circular hole are shown in Figure 14a, while the resolved y-direction loadpaths are depicted in Figure 14b. In both cases the loadpaths are uniformly distributed along opposing edges of the square plate, as appropriate for the loading orientation being considered.

In Figure 14b the y-direction loadpaths are vertically aligned throughout a large proportion of the plate in a manner that is very similar to what occurred for the uniaxially loaded plate (see Figure 8b). In comparison to this, the x-direction loadpaths in Figure 14a show considerably more deviation from the horizontal over a large part of the plate. The curvature of the x-direction loadpath lines around the hole is much more pronounced than for the y-direction loadpaths, and is quite pronounced at even at three or four hole diameters away from the centre of the hole. Looking at the x-direction loadpaths immediately above and below the hole, it is quite noticeable that they are angled in towards the horizontal centreline of the plate, starting right at the

left and right edges of the plate. Figure 15 shows a close-up view of the loadpaths in the vicinity of the recirculation region.

3.3 An optimal elliptical hole in a 4:1 biaxial stress field

This particular geometry, consisting of an elliptical hole in a y:x = 4:1 biaxial stress field, was chosen because it is well known that an elliptical hole is the optimal shape for this loading condition in an infinite plate, in the sense that it achieves the lowest value of SCF. The stress field that exists around such an elliptical hole produces an almost uniform hoop stress around the boundary, and the existence of a state of uniform stress along boundary segments is a feature of optimal shapes [1].

3.3.1 Geometry and loading

The plate geometry and loading arrangement that was considered for this analysis is shown in Figure 16. Here the square plate contains a centrally located elliptical hole with its major axis aligned with the y-direction. The major axis of the elliptical hole is 80 mm in length and its minor axis is 20 mm in width. The dimensions of the plate and the remote 4:1 biaxial stress field are the same as those used for the circular hole case in the previous section. The complete mesh discretisation for this problem is shown in Figure 17a. The mesh is highly refined around the boundary of the hole, and a close-up view is provided in Figure 17b.

For an infinite plate, the theoretical peak stress for this loading is 125 MPa, and the stress is completely uniform around the hole boundary. This peak stress is much lower than that obtained for a circular hole in the same stress field, which produces a theoretical peak stress of 275 MPa. It is interesting to note that the peak stress for the elliptical hole is considerably lower even though the area of the elliptical hole is much greater than that of the circular hole that was studied previously. The computed stresses around the boundary of the elliptical hole from the FEA, which includes finite-width effects, were 133.7 MPa ±8.2% (a mean stress of 128.8 MPa with a standard deviation of ±3.6% of the mean). The greatest variation from the theoretical solution occurs in regions near the top and bottom of the ellipse, where the hole boundary comes closest to the plate edges. In comparison, the stresses on the left and right boundaries of the ellipse at and around the region where x = ±10 mm are within ±1% of the theoretical values for the infinite-plate case. Although the results indicate that the elliptical hole in a finite-width plate that is being analysed here is sub-optimal to some degree, the difference between it and the true optimal shape for this loading and plate geometry are not considered to be very large. Hence, it is felt that the present elliptical hole in a finite-width plate can still be assumed to be optimal for our purposes.

3.3.2 Loadflow orientations

The resolved x-direction loadflow orientations are shown in Figure 18a, while the resolved y-direction loadflow orientations are shown in Figure 18b. In both of these cases the loadflow orientations follow the local hole boundary very closely. Unlike the

case for the non-optimal circular hole, there is no identifiable recirculation zone in either of the plots of loadflow orientations, which indicates the optimality of the elliptical hole.

3.3.3 *Loadpath contours*

Figure 19a and Figure 19b show the resolved x-direction and y-direction loadpaths, respectively. These were produced by uniformly distributing the loadpath starting points along the plate edges corresponding to $x = 100$ and $y = 100$. Some extra uniformly distributed x-direction loadpaths were added to the area where the loadpaths were flowing in the vicinity of the elliptical hole. In both parts of Figure 19, the loadpaths closely hug the boundary of the elliptical hole over a great portion of its length. The fact that the elliptical hole perturbs the y-direction loadpaths less than does a circular hole is another indication of the optimality of the elliptical hole. Looking at the x-direction loadpaths in Figure 19a, it is apparent that, in the vicinity of the horizontal x-axis, they diverge away from the horizontal as they approach the centre of the ellipse. In contrast to this, the loadpaths in Fig. 14(a) that pass close to the circular hole actually converge towards the horizontal centreline in the vicinity of the hole.

4. Prediction of Initial Rework Shapes

In the previous section, a number of examples have been presented that show the existence of loadflow recirculation regions for non-optimal rework shapes, while optimal rework shapes do not have a recirculation region. This leads to the hypothesis that those areas where recirculation occurs can be taken to be indicators of subregions in a structure that can be removed to achieve a more optimal shape. In view of this, it was decided to investigate the possibility of using the limits of the recirculation zone to define the boundary of an initial rework region that may provide a good starting shape for the application of a shape optimisation procedure.

4.1 Reworked circular hole in a uniaxial stress field

The loadflow orientations obtained around a circular hole in a large square plate uniaxially loaded in the y-direction have previously been shown in Figure 6 and Figure 7. The most significant recirculation zone corresponds to the resolved y-direction loadflow orientations (Figure 7b), and the rework region is selected to coincide with that subregion of the plate. By plotting loadpath lines in and around the recirculation zone, it was possible to define rework boundaries that essentially removed the recirculation zones at the top and bottom of the hole.

After the FE model of this design was created and analysed, the loadflow orientations were then computed and the results are shown in Figure 20. The amount of material removed corresponds to a maximum rework depth of approximately 2.5 mm (compare the circular hole shape in Figure 7 with the reworked hole shape in Figure 20). The removal of material has resulted in a vertical shift in the location of the recirculation zone, and the loadflow orientations are much the same as those obtained for the

original circular geometry. Also, because of the localised nature of the rework region, there is no significant change in the maximum SCF at the edge of the hole (the SCF increased by 0.4%). Hence, no benefit has been achieved at the peak stress location by the removal of the original recirculation zone. The stresses in the reworked region are altered from their initial compressive stress state to a state of tension, but the stress levels remain quite low. The loadpath lines after the removal of the recirculation zone are shown in Figure 21. Keeping in mind that an optimal rework shape has a uniform state of stress along most of its boundary [1], the fact that there is still a considerable variation in stress along the present reworked boundary is indicative of the highly non-optimal nature of this rework shape.

4.2 Reworked circular hole in a 4:1 biaxial stress field

The loadflow orientations obtained around a circular hole in a large square plate biaxially loaded by a 4:1 stress field have previously been shown in Figure 13. The most significant recirculation zone corresponds to the resolved x-direction loadflow orientations (Figure 13a) and occurs at the top and bottom of the hole. With the help of loadpath lines, the rework region is selected to coincide with that subregion of the plate. A smaller recirculation zone exists for the resolved y-direction loadflow, and this zone falls within the x-direction recirculation zone. By plotting loadpath lines in and around the recirculation zone, it was possible to define rework boundaries that essentially removed the zones of recirculation that occur at the top and bottom of the hole.

After the FE model of this design was created and analysed, the new loadflow orientations were then computed and the results are shown in Figure 22. The amount of material removed corresponds to a maximum rework depth of approximately 3.0 mm (compare the circular hole shape in Figure 13 with the reworked hole shape in Figure 22). The loadpath lines are shown in Figure 23. In contrast to the uniaxial loading case, the recirculation region has been cut out by the rework. However, because the amount of material removed was fairly minimal, as well as being restricted to the upper and lower areas of the hole, there was no significant change in the SCF at the edge of the hole (a SCF reduction of 0.2%). Although the stresses in the reworked region are altered from their initial compressive stress state to a state of tension, the stress levels remain quite low compared to the peak stress levels elsewhere in the structure.

5. Discussion and Conclusion

This report presents recent developments and advances in the analysis and visualisation of structural loadflow, with the view to establishing FEA-based numerical procedures for loadflow visualisation to gain insights into structural performance. The particular areas that have been addressed include: i) enhancement of the theoretical formulation and numerical procedure, ii) studies of the differences between non-optimal and optimal shapes, and iii) the investigation of potential applications to optimal shape reworking.

Initially, the theory of Kelly and Elsley has been revisited and, in a key development, we have derived explicit equations for the computation of loadflow orientations. These equations provide better accuracy compared to the previous approach, which was based on the iterative solution of non-linear equations. Our derivation has also dispersed an element of confusion arising from the reference to the major principal stress angle in the presentation of the original theory. Then, by applying a fluidflow analogy proposed by Kelly [16], we have used the vector-like field of loadflow orientations to determine the loadpaths in a uniaxially loaded square plate with a circular hole. Here we found that the loadflow orientations obtained by using the explicit equations improved the fidelity of the calculated loadpaths. For this case we have also studied the compaction of loadflow lines to determine the stress concentration factor, successfully verifying the use of the fluidflow analogy in the calculation of the loadpaths. The exact nature of this analogy is not understood at this point in time, and it should be studied in future work to gain a better understanding.

The above procedures were then applied to investigate the features of non-optimal and optimal holes in plates where both loadflow orientations and loadpaths were determined. Here an optimal hole shape is taken to be one that produces the minimum peak SCF around the hole boundary. Through this comparison, it was found that a distinct recirculation zone is apparent for non-optimal hole shapes, whereas no recirculation region is present for optimal shapes. In addition, the importance of using highly refined FE meshes to enable fine details in recirculation zones to be resolved must be emphasised. Although very highly refined FE meshes were utilised in our investigations, the implications are that even more refined meshes are required to fully capture the complex behaviour that exists in recirculation zones.

In the context of shape optimisation, the use of loadflow visualisation has been investigated as a potential aid in determining good initial estimates of rework shapes. Although the removal of the recirculation zone for a non-optimal hole shape leads to a better shape, the improvement in peak stress is insignificant, even when a distinct recirculation zone is present. This is largely due to the fact that the recirculation zone is quite localised in nature, whereas the achievement of an optimal shape requires that the rework region be comprised of a substantial portion of the original boundary in order to enable a significant reduction in SCF to be obtained. In situations where the original structural shape is similar to the optimal rework shape, the recirculation zone is less pronounced and it becomes more difficult to clearly identify a rework shape that may improve the design.

In general, the visualisation of loadflow orientations and loadpaths presents structural designers and analysts with a powerful tool to complement those already at their disposal. Loadflow visualisation can improve their insight into the transfer of load through a structure, thus placing them in a better position to gauge how well a structure is performing, potentially resulting in worthwhile improvements in structural efficiency. The simplicity of our improved method of calculation of loadflow orientations means that it is now just as easy, if not in fact easier, to compute loadflow orientations than it is to calculate principal stresses. Thus, there is considerable potential to have the computation of loadflow orientations implemented as a standard

feature in FEA codes, in much the same way that the calculations of principal stresses are today.

6. Acknowledgement

The authors would like to thank Professor D.W. Kelly and M.W. Tosh (University of New South Wales, Sydney, Australia) for helpful comments on the use of a fluidflow streamtrace concept, in conjunction with the commercial code Tecplot, for the visualisation of structural loadpaths.

7. References

1. **R. Kaye and M. Heller.** *Structural Shape Optimisation by Iterative Finite Element Solution.* DSTO-RR-0105, Airframes and Engines Division, Aeronautical and Maritime Research Laboratory, June 1997.

2. **R.V. Baud.** *Study of Stresses by Means of Polarized Light and Transparencies.* Proceedings of Engineers' Society of Western Penna., Vol. 44, July 1928, pp. 199–232.

3. **D.W. Kelly and M. Elsley.** *A Procedure for Determining Loadpaths in Elastic Continua.* Cooperative Research Centre for Aerospace Structures Ltd, Report CRC-AS EP 93500, August 1993.

4. **D.W. Kelly and M. Elsley.** *A Procedure for Determining Load Paths in Elastic Continua.* Engineering Computations, Vol. 12, pp. 415–424, 1995.

5. **P.R. Chaperon, M. Heller, R. Jones, S. Pitt and F. Rose.** *Load Flow Visualisation in Structural Optimisation.* Structural Optimisation, Proceedings of the Australasian Conference on Structural Optimisation, February 11–13, 1998, Sydney, Australia, pp. 517–524. Editors: G.P. Steven, O.M. Querin, H. Guan and Y.M. Xie.

6. **R. Ibrahim, P. Chaperon and R. Jones.** *Automated Determination of Three Dimensional Load Paths with Implications for Structural Shape Optimisation – Phase II.* COE-SM-97-05, June 1998, Centre of Expertise for Structural Mechanics, Department of Mechanical Engineering, Monash University.

7. **B. Chen and R. Jones.** *Automated Determination of Three Dimensional Load Paths with Implications for Structural Shape Optimisation.* Interim Report COE-SM-98-05 April 1999, Centre of Expertise for Structural Mechanics, Department of Mechanical Engineering, Monash University.

8. **R.E. Peterson.** *Stress Concentration Factors.* John Wiley & Sons, 1974, ISBN 0-471-68329-9.

9. **W.D. Pilkey**. *Peterson's Stress Concentration Factors*, Second Edition. John Wiley & Sons, 1997, ISBN-0-471-53849-3.

10. **C.C. Osgood**. *Fatigue Design*. Pergamon Press, 1982, ISBN 0-08-026166-3.

11. **D. Broek**. *The Practical Use of Fracture Mechanics*. Kluwer Academic Publishers, 1991, ISBN 90-247-3707-9.

12. **R.V. Baud**. *Fillet Profiles for Constant Stress*. Product Engineering, pp. 133–134, April 1934.

13. **H. Neuber**. *Theory of Notch Stresses: Principles for Exact Calculation of Strength with Reference to Structural Form and Material*. Translation Series, United States Atomic Energy Commission, Office of Technical Information, AEC-TR-4547, June 1961. Translated from a publication of Springer-Verlag, Berlin, Göttingen, Heidelberg, 1958.

14. **D.W. Kelly**. *Load Path Calculations and Applications Using Finite Element Analysis*. First Australasian Congress on Applied Mechanics, 21–23 February 1996, Melbourne, Australia, pp. 625–630.

15. **D.W. Kelly and M.W. Tosh**. *Load Paths and Stress Trajectories in Elasticity*. Cooperative Research Centre for Advanced Composite Structures Ltd, Report CRC-ACS EP 97022, January 1998.

16. **D.W. Kelly and M.W. Tosh**. Private communication on the use of fluidflow streamtrace concept for visualisation of loadpath lines, May 1998.

17. **S. Timoshenko and J.N. Goodier**. *Theory of Elasticity*. McGraw-Hill Book Company, Inc., 1970, 3rd Edition.

18. **Anonymous**. *Tecplot Version 7.5 User's Manual*. Amtec Engineering, Inc., Bellevue, Washington, USA, 1998.

Table 1 *Stress concentration factor* SCF *along the x-axis in the vicinity of a circular hole of radius* a *in a uniaxially loaded square plate computed from compaction of loadpath lines.*

a/x_c	x_c (mm)	Δ_c (mm)	x_b (mm)	Δ_b (mm)	SCF
0.9524	10.500	0.01	1.3993	0.0280	2.8000
0.9434	10.600	0.20	1.6794	0.5429	2.7145
0.9434	10.600	0.01	1.6819	0.0269	2.6900
0.9346	10.700	0.01	1.9535	0.0267	2.6700
0.9259	10.800	0.01	2.2191	0.0264	2.6400
0.9174	10.900	0.01	2.4779	0.0258	2.5800
0.9091	11.000	0.01	2.7327	0.0250	2.5000
0.8929	11.200	0.01	3.2218	0.0239	2.3900
0.8772	11.400	0.01	3.7065	0.0238	2.3800
0.8624	11.595	0.01	4.1633	0.0227	2.2700
0.8333	12.000	0.20	5.0592	0.4241	2.1205
0.8000	12.500	0.20	6.0912	0.3998	1.9990
0.7692	13.000	0.20	7.0628	0.3796	1.8980
0.7407	13.500	0.20	7.9673	0.3485	1.7425
0.7143	14.000	0.20	8.8200	0.3283	1.6415
0.6667	15.000	0.20	10.4339	0.3146	1.5730
0.6250	16.000	0.20	11.9538	0.2906	1.4530
0.5882	17.000	0.20	13.3776	0.2838	1.4190
0.5556	18.000	0.20	14.7319	0.2633	1.3165
0.5263	19.000	0.20	16.0304	0.2496	1.2480
0.5000	20.000	0.20	17.2718	0.2496	1.2480
0.4545	22.000	0.20	19.7167	0.2394	1.1970
0.4167	24.000	0.20	22.0530	0.2326	1.1630
0.3846	26.000	0.20	24.3433	0.2257	1.1285
0.3571	28.000	0.20	26.5897	0.2232	1.1160
0.3125	32.000	0.20	30.9716	0.2183	1.0915
0.2857	35.000	0.20	34.1956	0.2155	1.0775
0.2778	36.000	0.20	35.2537	0.2104	1.0520
0.2500	40.000	0.20	39.4768	0.2123	1.0615
0.2222	45.000	0.20	44.7117	0.2102	1.0510
0.1961	51.000	0.20	50.9355	0.2062	1.0310
0.1667	60.000	0.20	60.1853	0.2046	1.0230
0.1429	70.000	0.20	70.3166	0.2002	1.0010
0.1250	80.000	0.20	80.3687	0.2006	1.0030
0.1111	90.000	0.20	90.2631	0.1976	0.9880
0.1053	95.000	0.20	95.1443	0.1976	0.9880
0.1005	99.500	0.20	99.6036	0.1995	0.9975

Table 2 *Stress concentration factor* SCF *in the vicinity of a circular hole of radius* a *in a uniaxially loaded large square plate as computed from finite element analysis nodal stresses taken along the x-axis.*

a/x	x (mm)	SCF
0.1000	100.0000	0.9643
0.1053	94.9301	0.9754
0.1110	90.1187	0.9835
0.1169	85.5527	0.9901
0.1231	81.2194	0.9957
0.1297	77.1071	1.0006
0.1366	73.2045	1.0051
0.1439	69.5008	1.0092
0.1515	65.9860	1.0131
0.1596	62.6503	1.0168
0.1681	59.4848	1.0205
0.1771	56.4806	1.0242
0.1865	53.6296	1.0279
0.1964	50.9240	1.0317
0.2068	48.3563	1.0357
0.2178	45.9196	1.0399
0.2293	43.6070	1.0444
0.2415	41.4124	1.0493
0.2543	39.3297	1.0546
0.2677	37.3532	1.0605
0.2819	35.4774	1.0670
0.2968	33.6973	1.0743
0.3124	32.0079	1.0824
0.3289	30.4047	1.0917
0.3462	28.8832	1.1022
0.3644	27.4393	1.1143
0.3836	26.0690	1.1281
0.4037	24.7685	1.1441
0.4249	23.5344	1.1626
0.4472	22.3632	1.1840
0.4706	21.2517	1.2091
0.4951	20.1969	1.2384
0.5209	19.1959	1.2727
0.5481	18.2459	1.3131
0.5766	17.3443	1.3607
0.6065	16.4887	1.4170
0.6379	15.6768	1.4835
0.6709	14.9062	1.5625
0.7055	14.1749	1.6563
0.7418	13.4809	1.7679
0.7799	12.8223	1.9009
0.8199	12.1972	2.0597
0.8618	11.6041	2.2494
0.9057	11.0411	2.4762
0.9518	10.5069	2.7478
1.0000	10.0000	3.1165

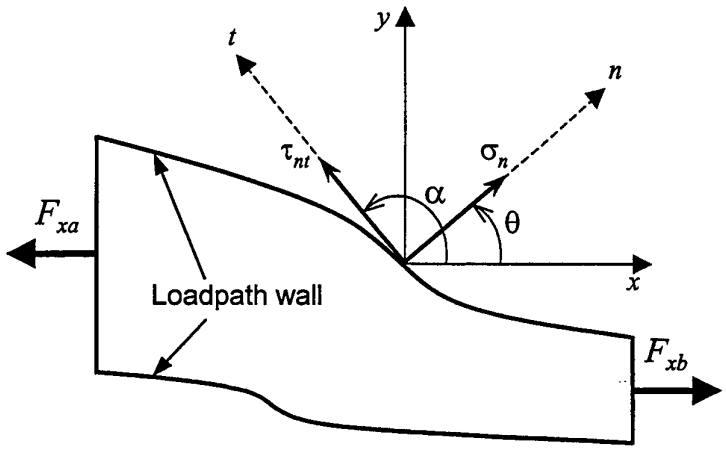


Figure 1 *Schematic of a force "stream tube" comprised of loadpath walls proposed by Kelly and Elsley [3] [4], shown for the resolved x-direction case.*

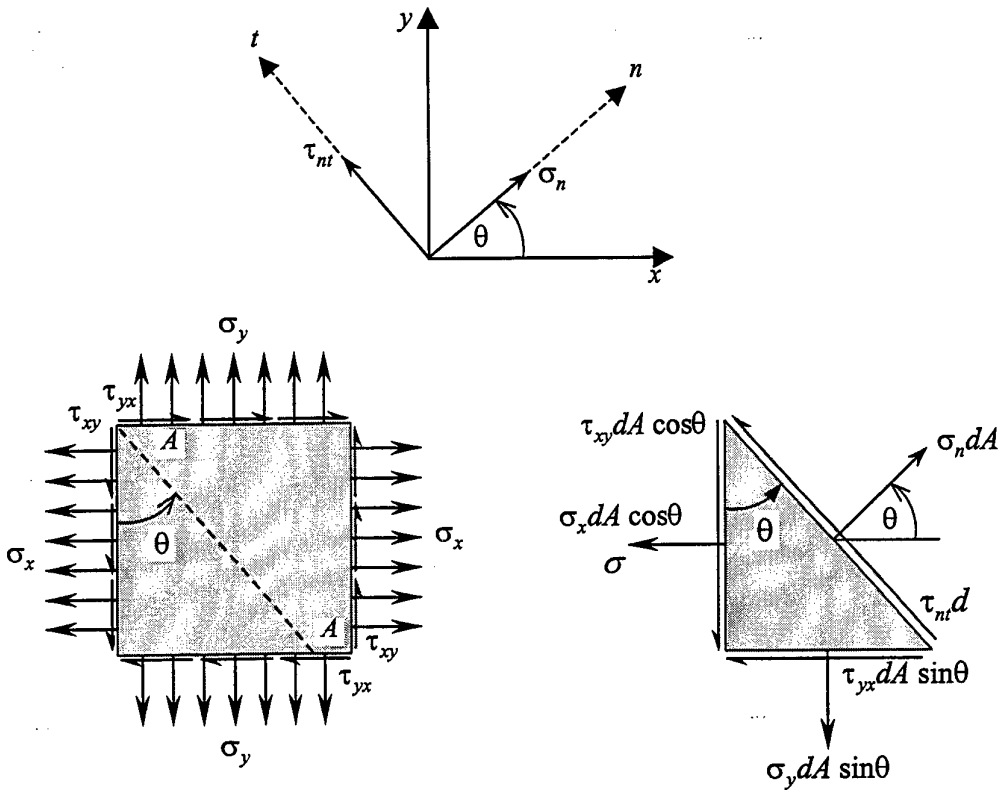


Figure 2 *Geometry and notation for stress components at a point in a body acting on a triangular prism.*

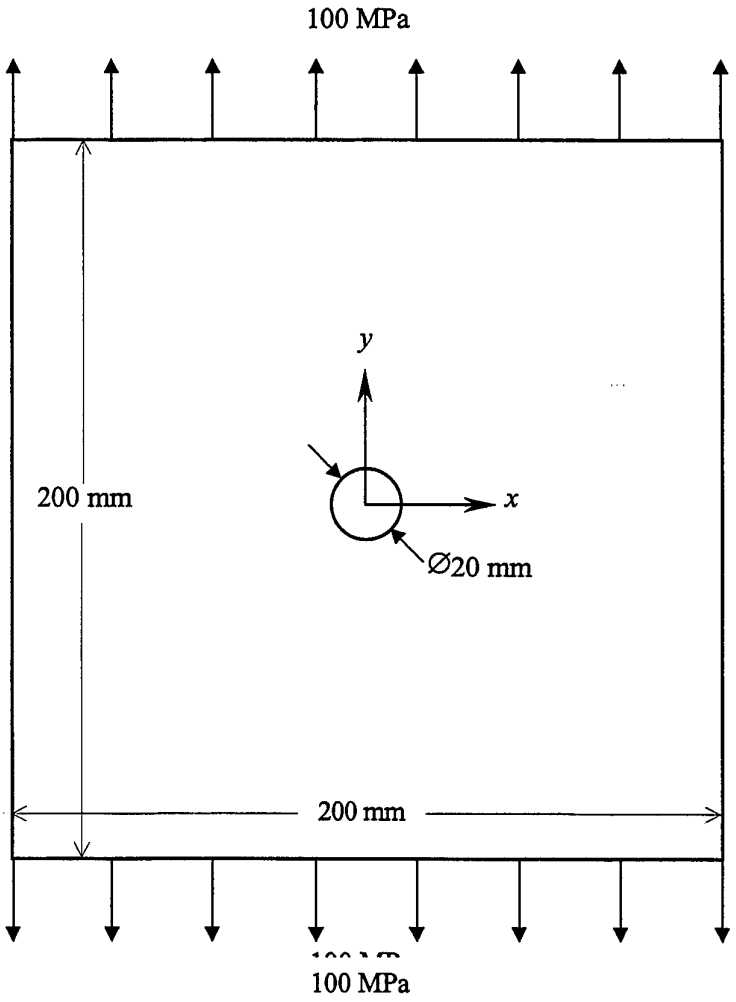
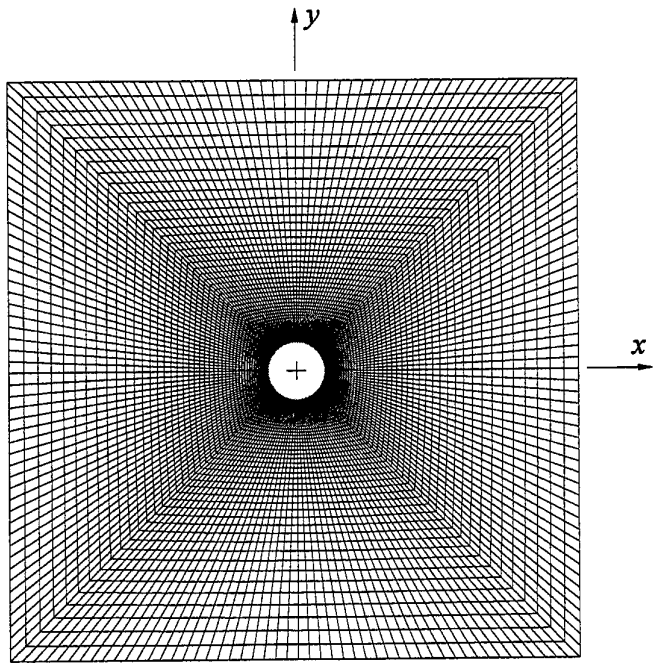
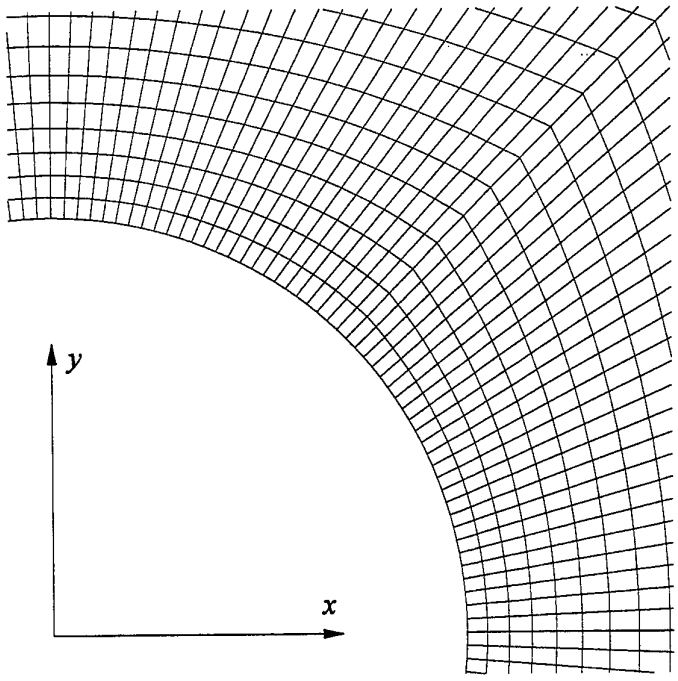


Figure 3 *Geometry and loading arrangement for a large square plate containing a circular hole with a remote uniaxial stress field.*

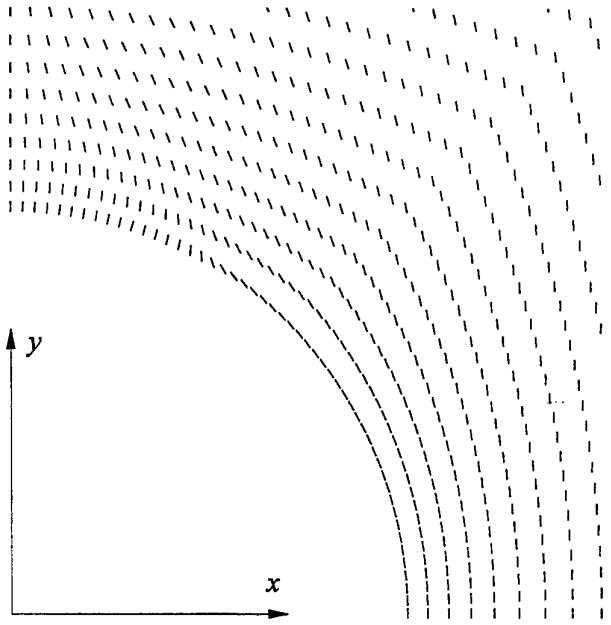


(a) Complete finite element mesh

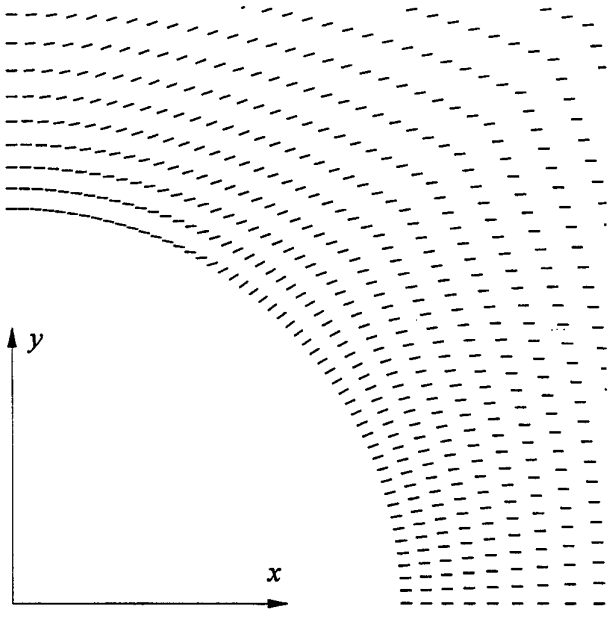


(b) Detail of finite element mesh around circular hole

Figure 4 *Finite element mesh for modelling a large square plate containing a circular hole.*

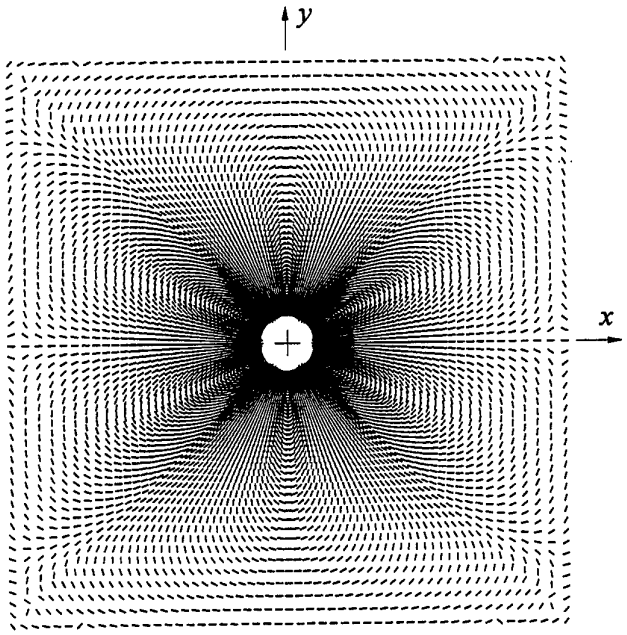


(a) Orientations of principal stress σ_{p1}

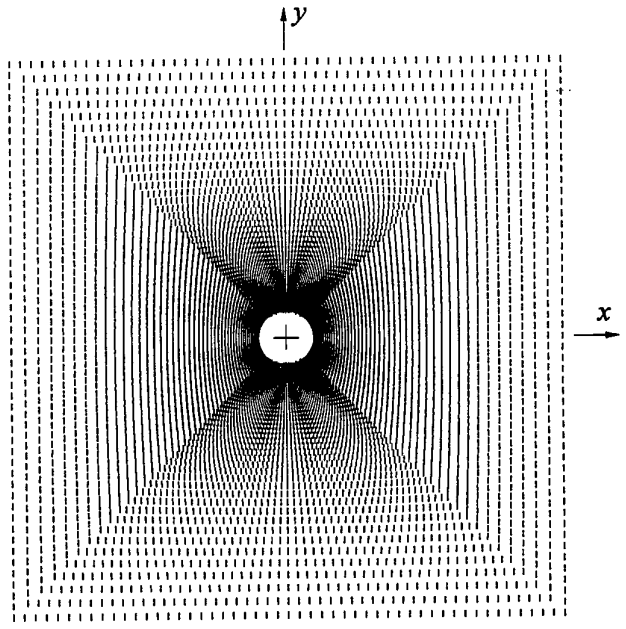


(b) Orientations of principal stress σ_{p2}

Figure 5 *Orientations of principal stresses σ_{p1} and σ_{p2} around a circular hole in a large square plate uniaxially loaded in the y-direction.*

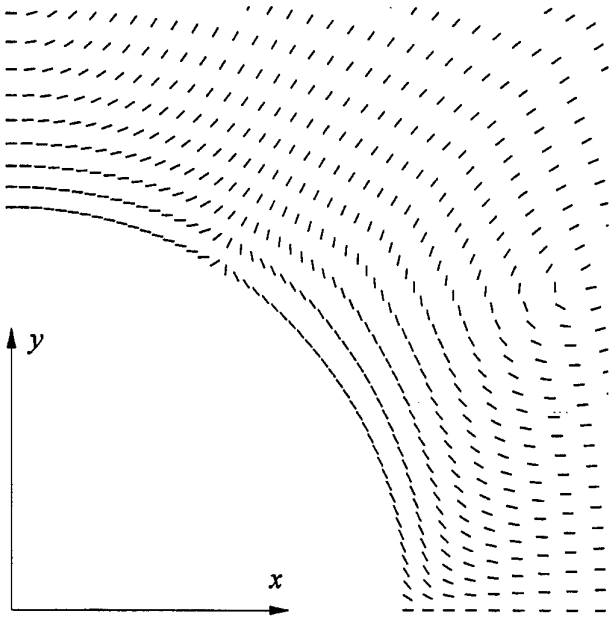


(a) Resolved x-direction loadflow orientations

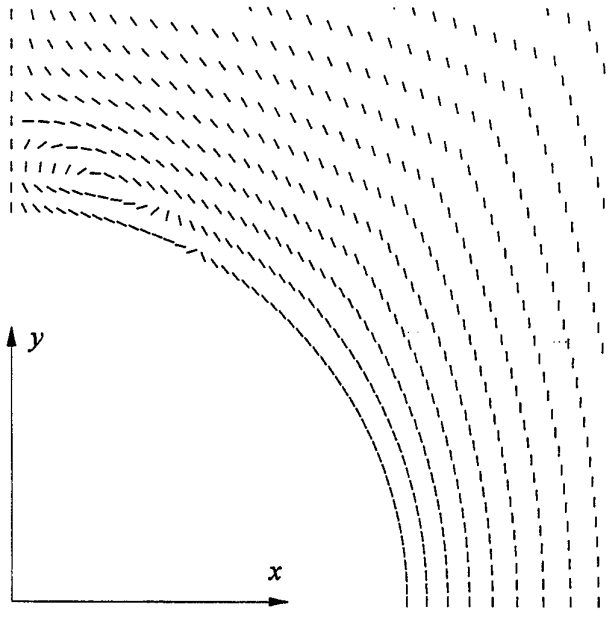


(b) Resolved y-direction loadflow orientations

Figure 6 *Resolved x-direction and y-direction loadflow orientations for a circular hole in a large square plate uniaxially loaded in the y-direction.*

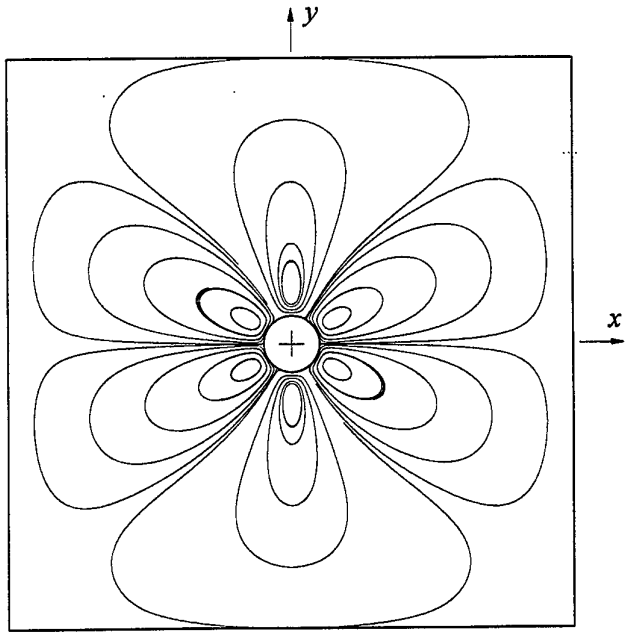


(a) Resolved x-direction loadflow orientations

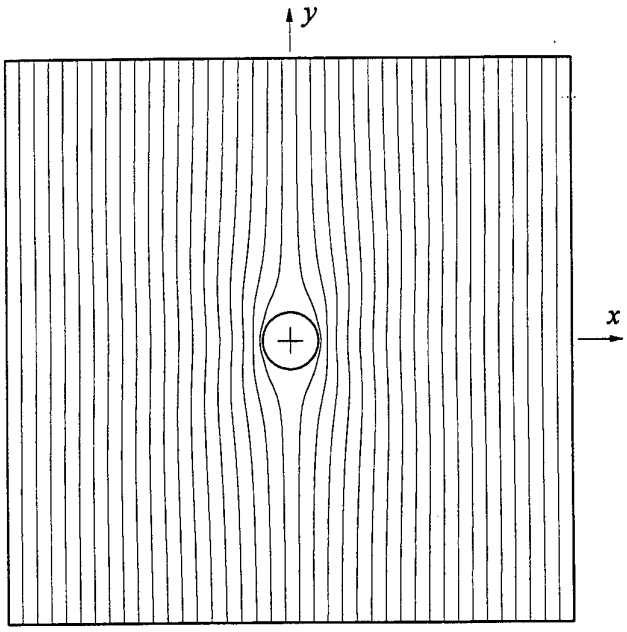


(b) Resolved y-direction loadflow orientations

Figure 7 *Resolved x-direction and y-direction loadflow orientations obtained around a circular hole in a large square plate uniaxially loaded in the y-direction.*

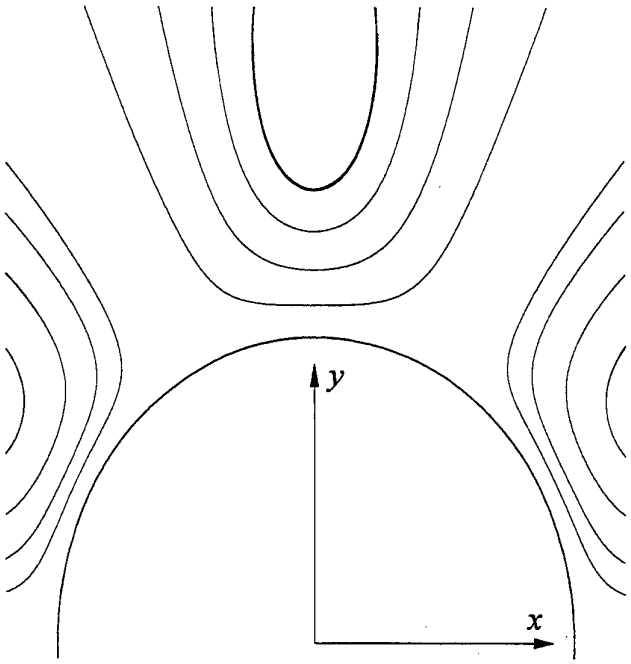


(a) Resolved x-direction loadpaths

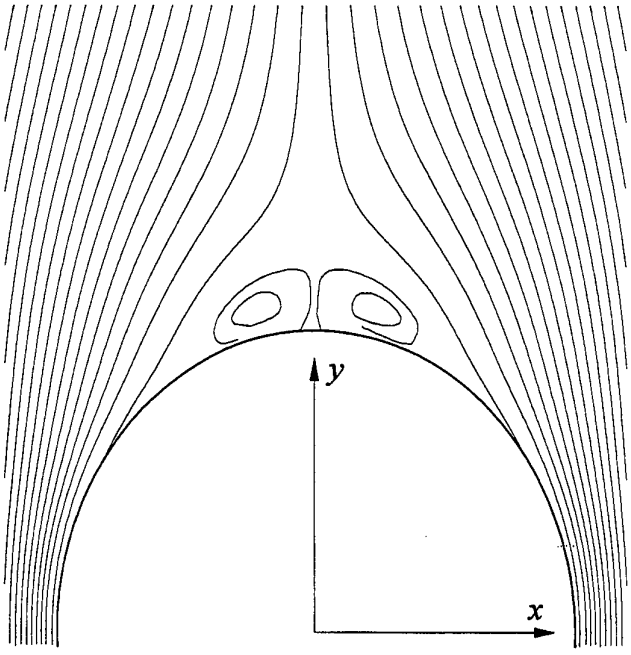


(b) Resolved y-direction loadpaths

Figure 8 *Loadpaths around a circular hole in a large square plate uniaxially loaded in the y-direction.*

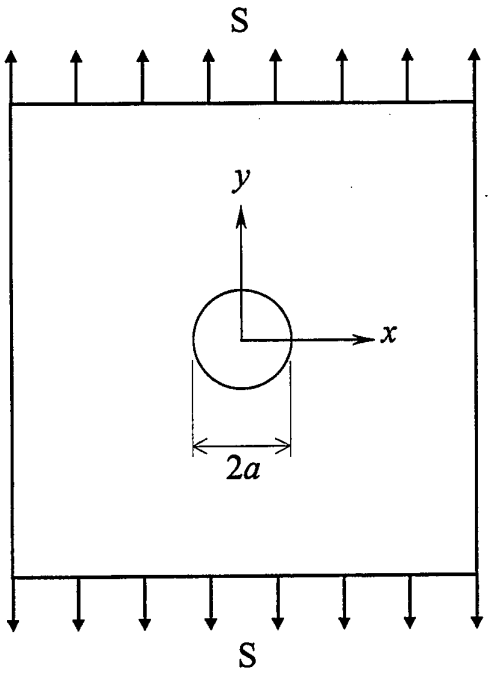


(a) Resolved x-direction loadpaths

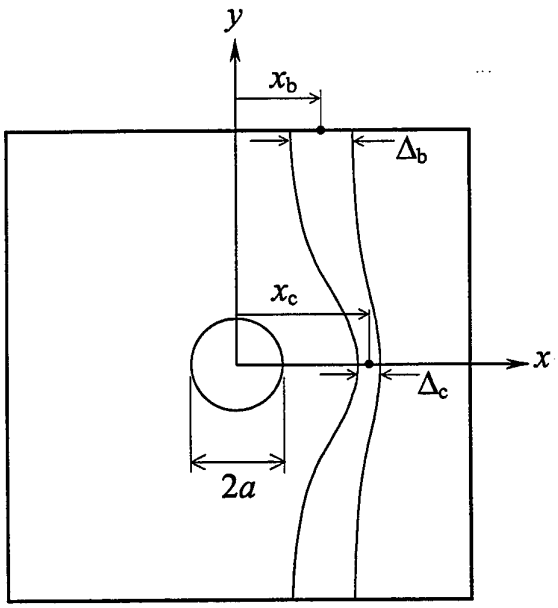


(b) Resolved y-direction loadpaths

Figure 9 *Loadpaths in vicinity of recirculation region for a circular hole in a large square plate uniaxially loaded in the y-direction.*



(a)



(b)

Figure 10 *Circular hole in a large square plate under uniform uniaxial loading: (a) general geometry and applied stress, and (b) definition of parameters used to calculate SCF from compaction of loadpath contours.*

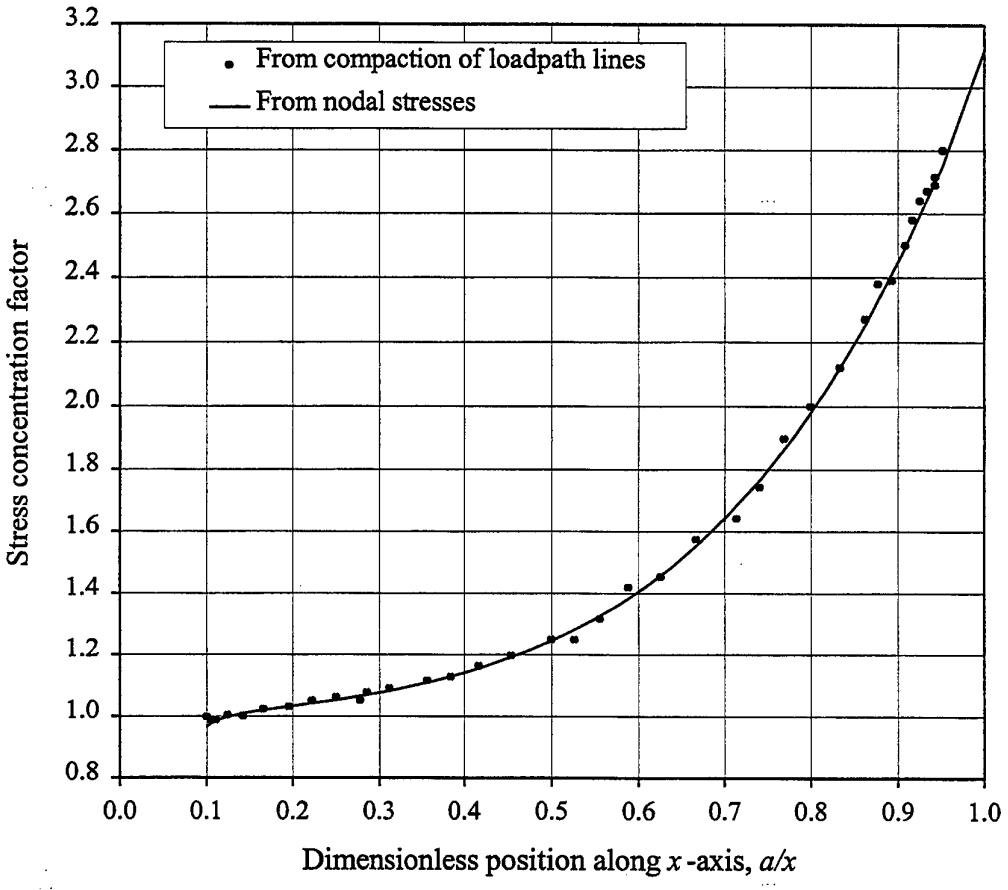


Figure 11 *Comparison of stress concentration factors computed from compaction of loadpath lines and finite element model nodal stresses for a uniaxially loaded large square plate with a circular hole.*

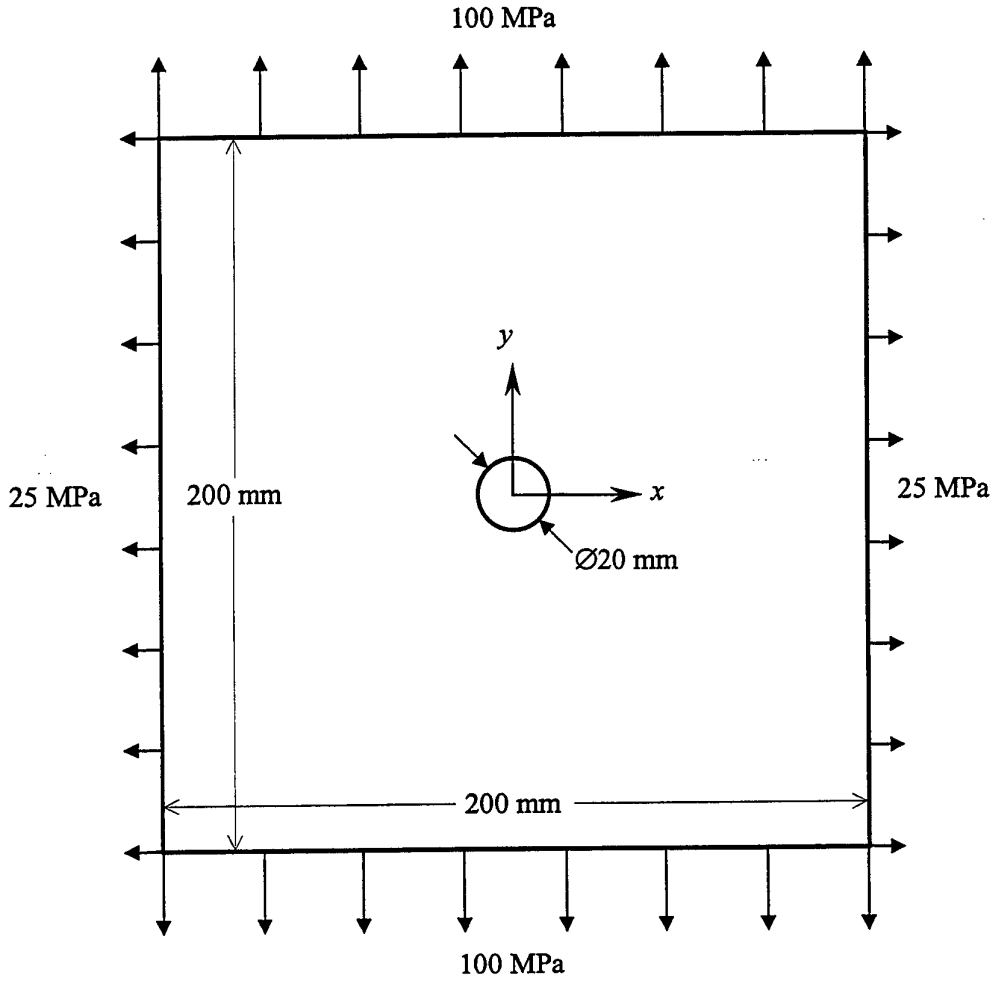
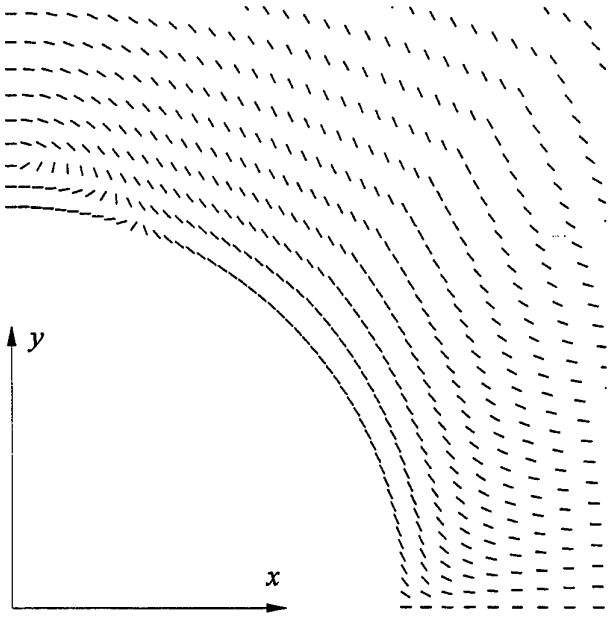
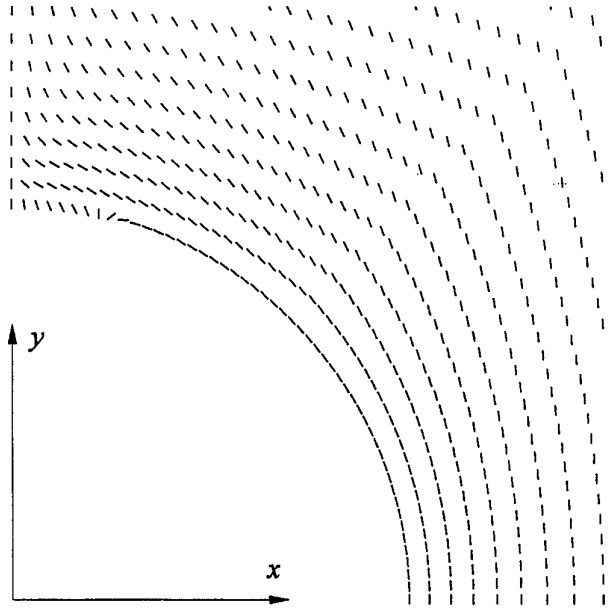


Figure 12 Geometry and loading arrangement for a large square plate containing a circular hole with a remote 4:1 biaxial stress field.

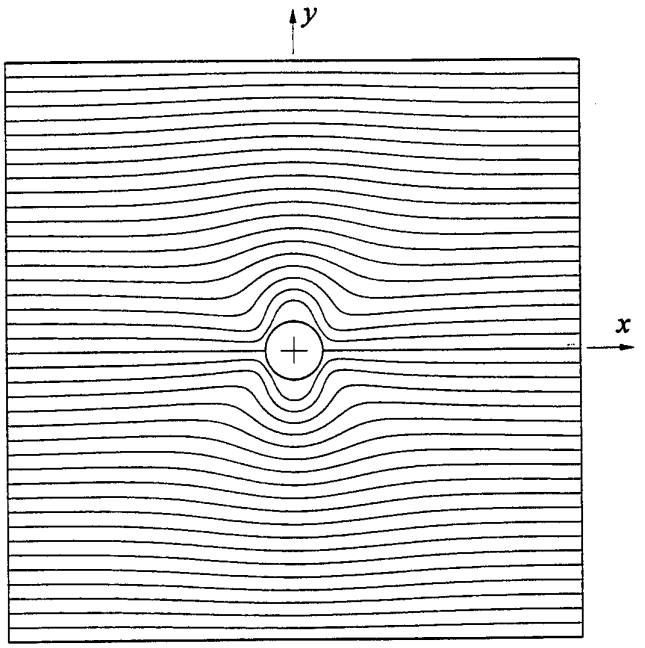


(a) Resolved x-direction loadflow orientations

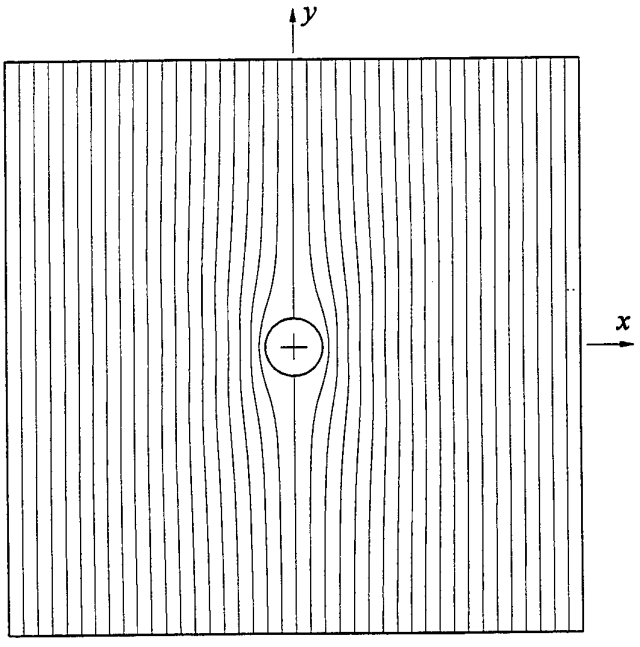


(b) Resolved y-direction loadflow orientations

Figure 13 *Resolved x-direction and y-direction loadflow orientations obtained around a circular hole in a large square plate under y:x = 4:1 biaxial loading.*

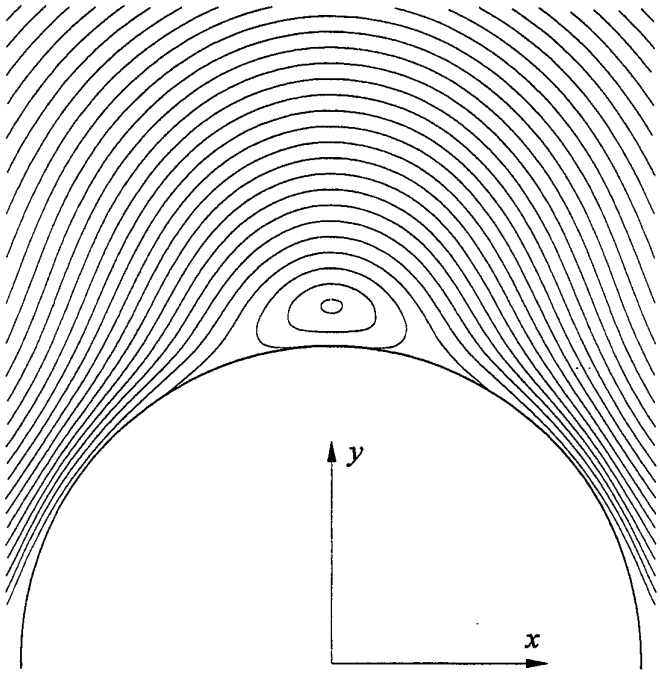


(a) Resolved x-direction loadpaths

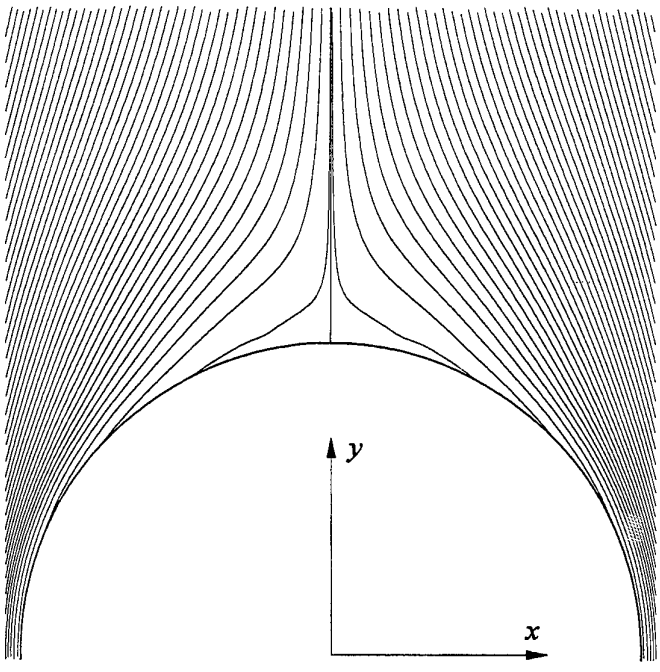


(b) Resolved y-direction loadpaths

Figure 14 *Resolved x-direction and y-direction loadpaths obtained around a circular hole in a large square plate under y:x = 4:1 biaxial loading.*



(a) Resolved x-direction loadpaths



(b) Resolved y-direction loadpaths

Figure 15 *Loadpaths in vicinity of recirculation region for a circular hole in a large square plate under y:x = 4:1 biaxial loading.*

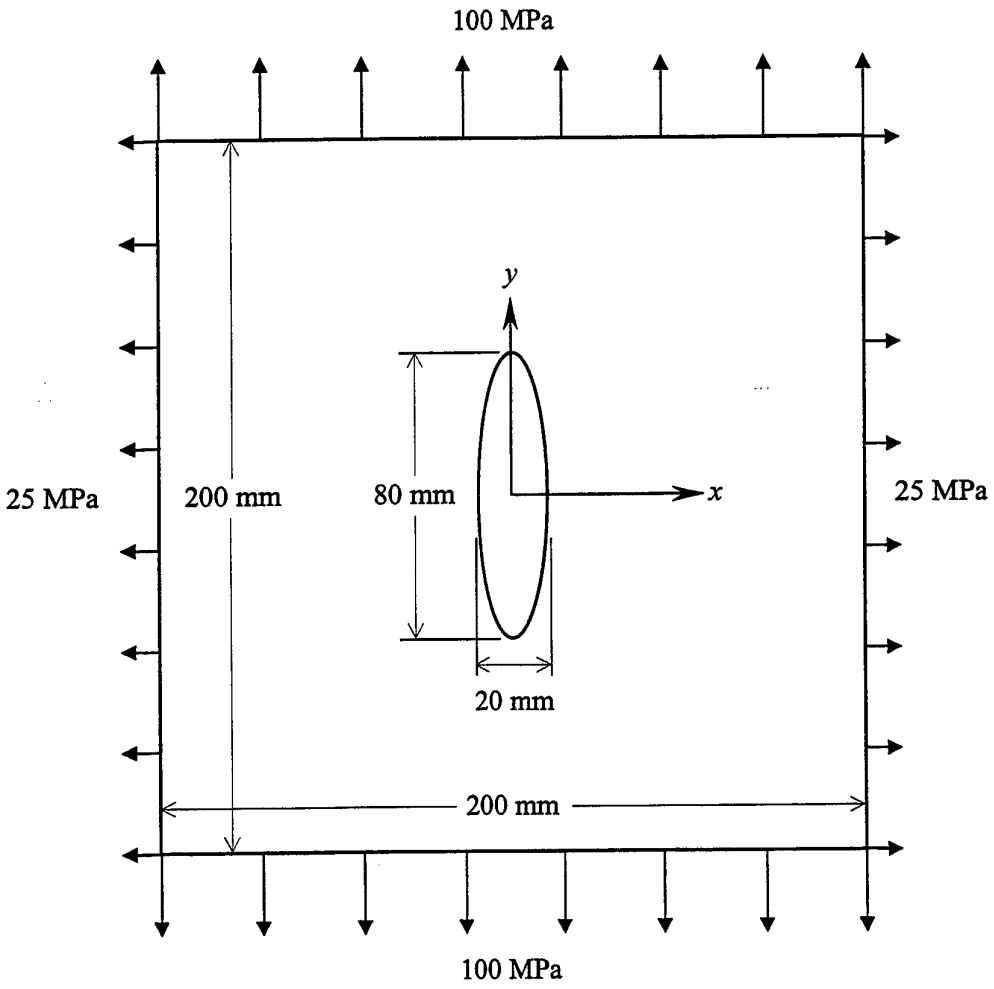
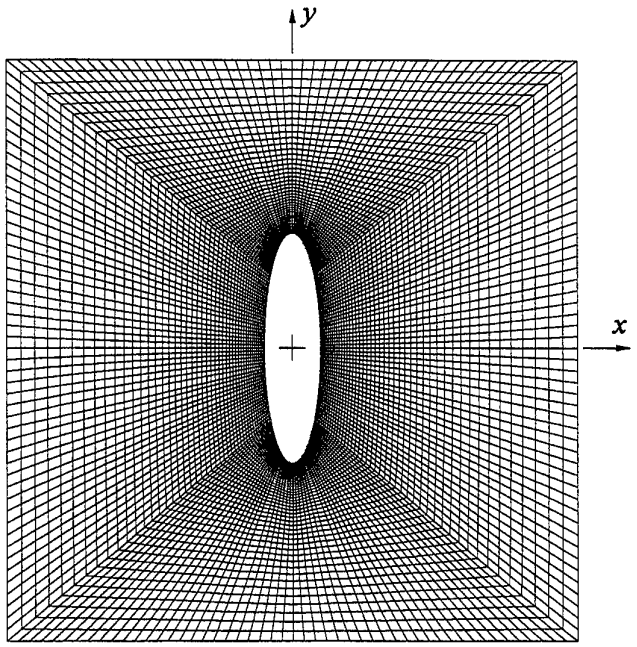
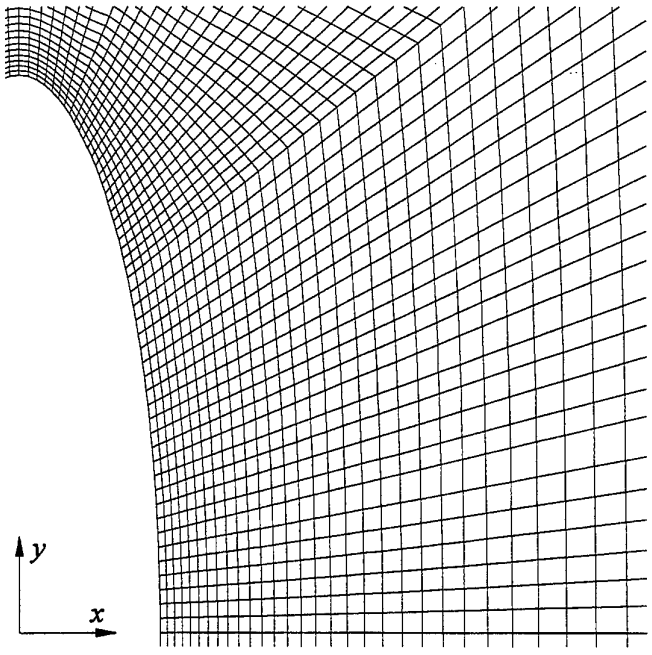


Figure 16 *Geometry and loading arrangement for a large square plate containing an optimal 4:1 elliptical hole with a remote y:x = 4:1 biaxial stress field.*

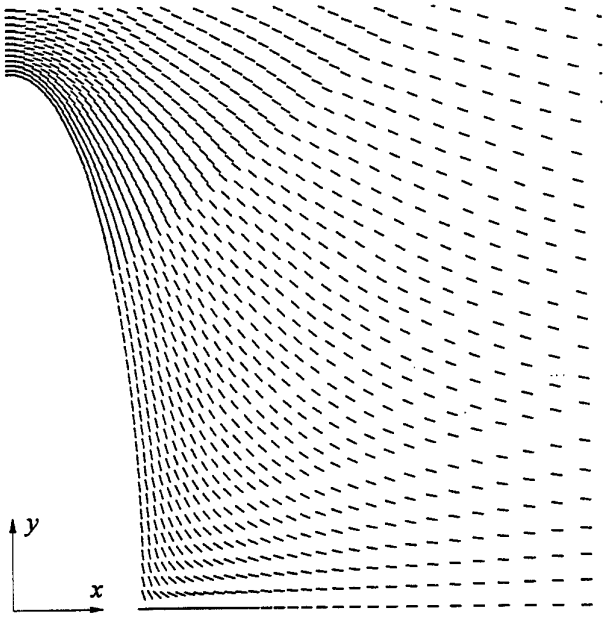


(a) Complete finite element mesh

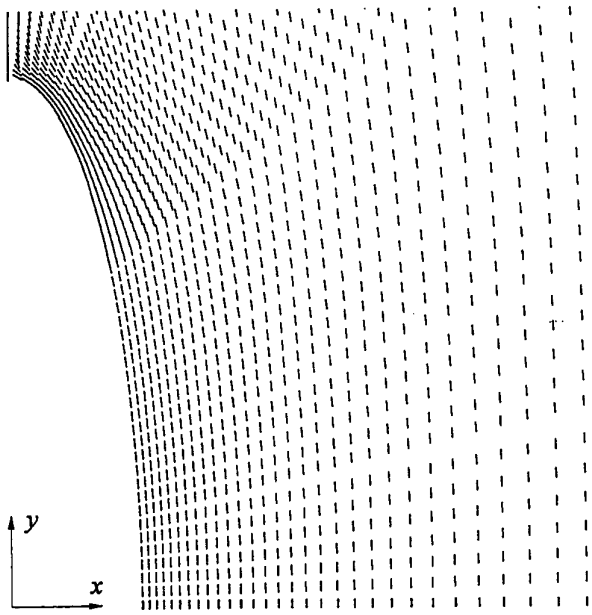


(b) Detail of finite element mesh around elliptical hole

Figure 17 *Finite element mesh for modelling a large square plate containing a y:x = 4:1 elliptical hole.*

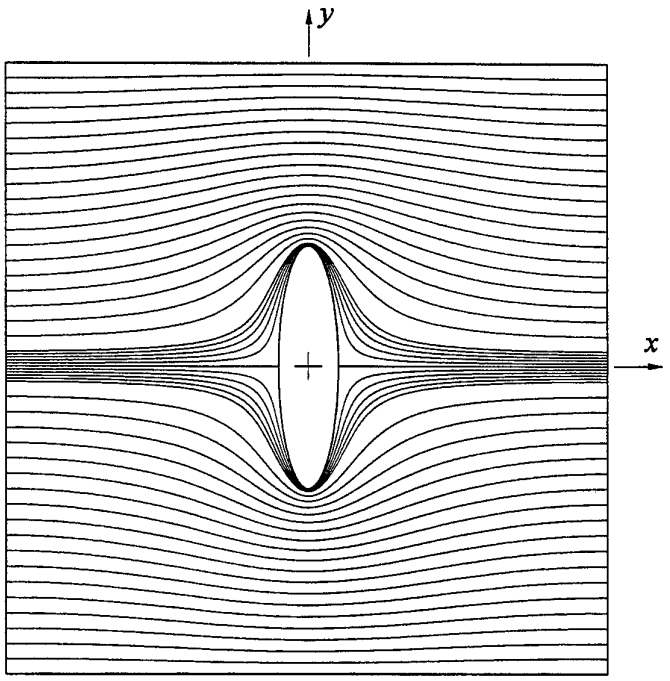


(a) Resolved x-direction loadflow orientations

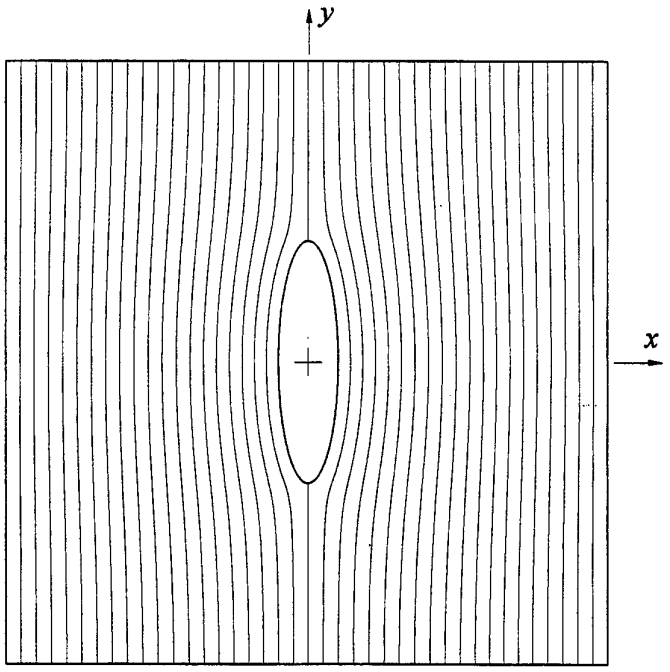


(b) Resolved y-direction loadflow orientations

Figure 18 *Resolved x-direction and y-direction loadflow orientations obtained around an optimal 4:1 elliptical hole in a large square plate under y:x = 4:1 biaxial loading.*

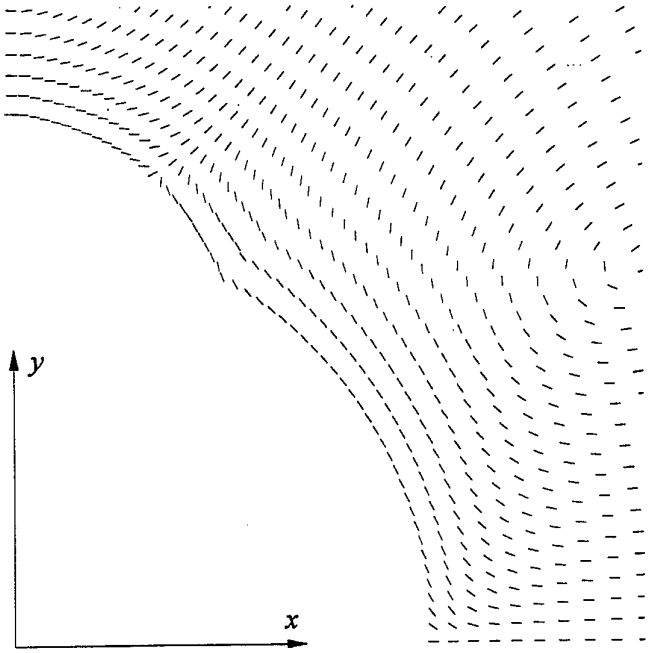


(a) Resolved x-direction loadpaths

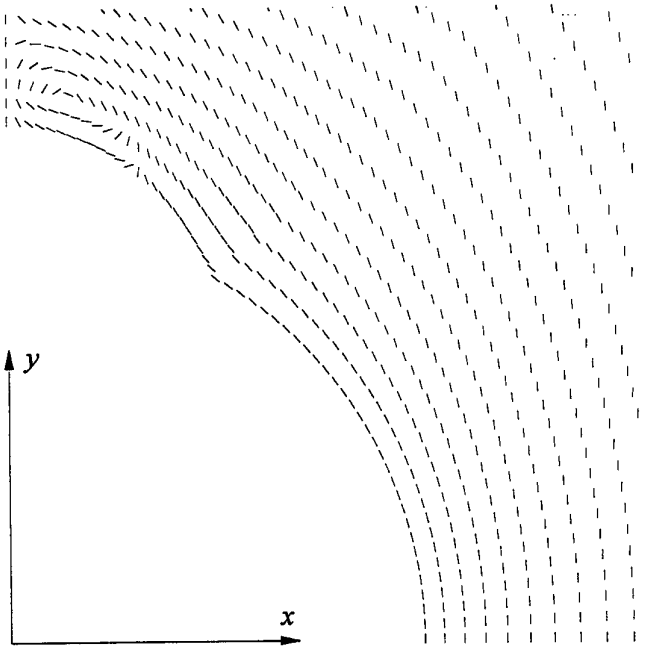


(b) Resolved y-direction loadpaths

Figure 19 *Resolved x-direction and y-direction loadpaths obtained around an optimal 4:1 elliptical hole in a large square plate under y:x = 4:1 biaxial loading.*

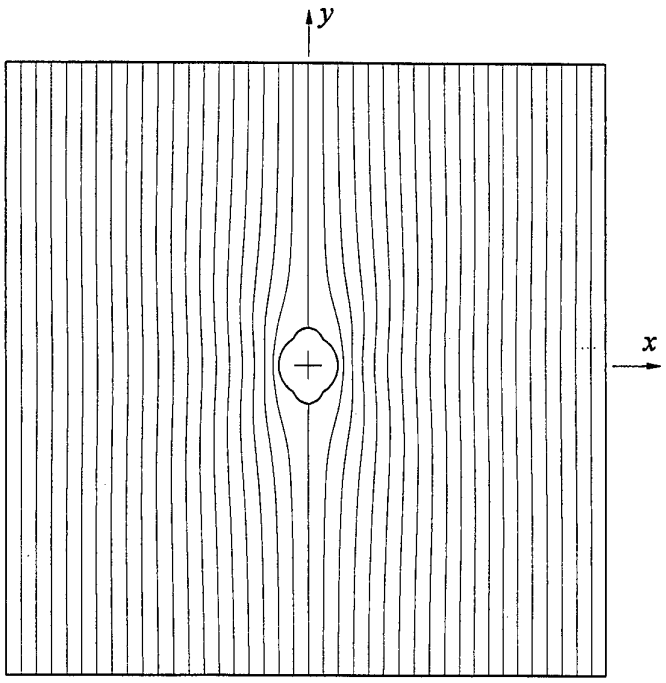


(a) Resolved x-direction loadflow orientations

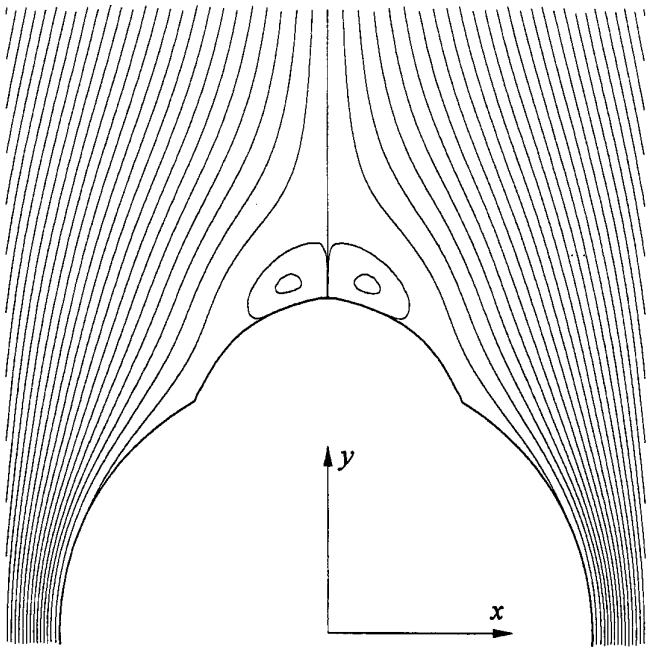


(b) Resolved y-direction loadflow orientations

Figure 20 *Resolved x-direction and y-direction loadflow orientations obtained around a reworked circular hole in a large square plate uniaxially loaded in the y-direction.*

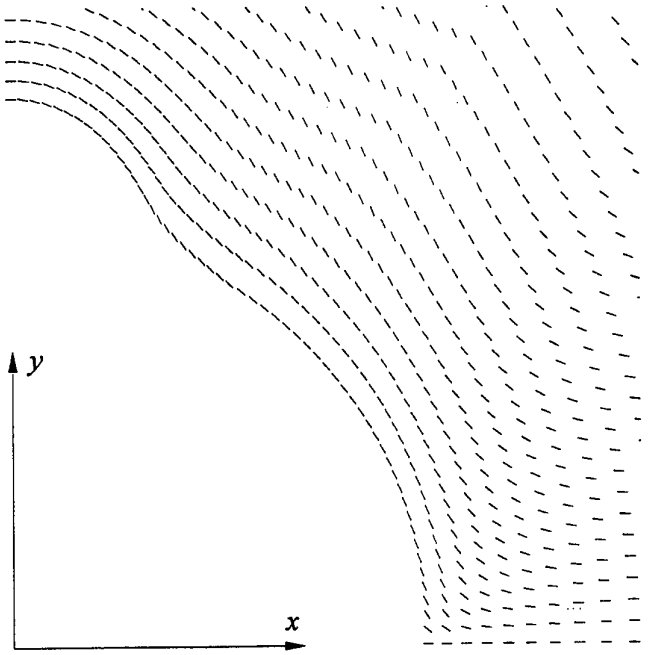


(a) Full-field loadpaths

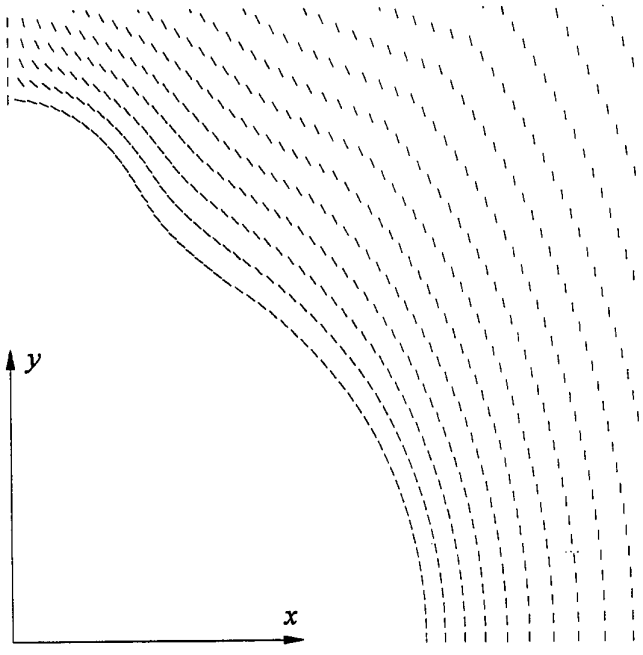


(b) Loadpaths near hole boundary

Figure 21 Resolved y-direction loadpaths around a reworked circular hole (after one manual iteration) in a large square plate uniaxially loaded in the y-direction.

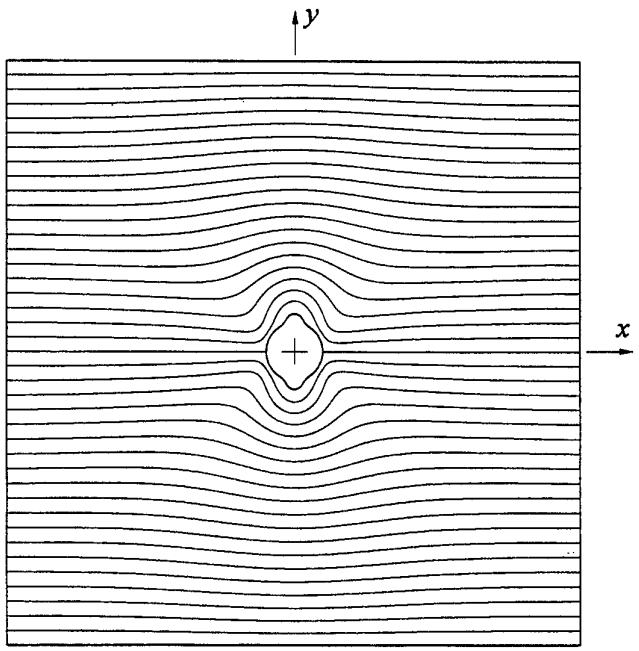


(a) Resolved x-direction loadflow orientations

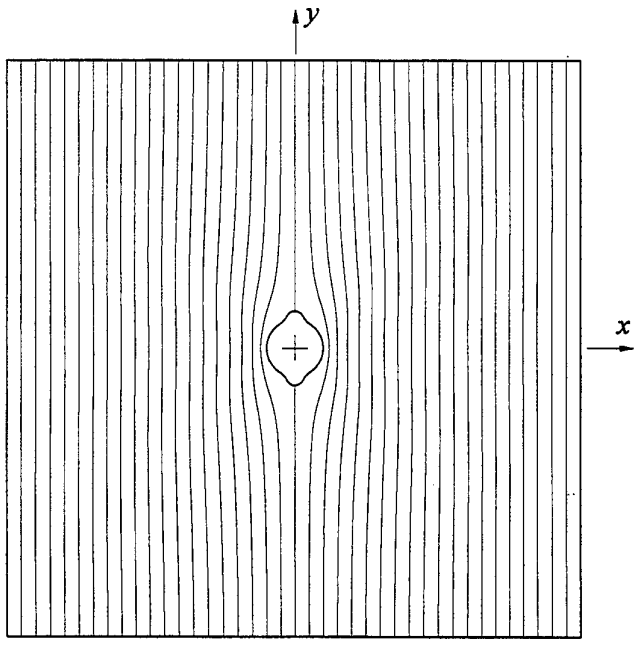


(b) Resolved y-direction loadflow orientations

Figure 22 *Resolved x-direction and y-direction loadflow orientations obtained around a reworked circular hole in a large square plate under y:x = 4:1 biaxial loading.*



(a) Resolved x-direction loadpaths



(b) Resolved y-direction loadpaths

Figure 23 *Resolved x-direction and y-direction loadpaths obtained around a reworked circular hole in a large square plate under y:x = 4:1 biaxial loading.*

Appendix A

Extracting Stresses From NASTRAN FEA Results

A.1 General comments

In order to be able to perform loadflow calculations for a structure modelled in a FEA package, it is necessary to be able to extract out the stresses that occur at nodes and elements in the FE model. These results are saved in separate files and used as input files by the program that computes loadflow orientations. When using the MSC.NASTRAN FEA package and MSC.PATRAN pre- and post-processor, the procedure that was used to create the nodal stress file is:

1. Use the Fortran program pnode.f to pull node data lines from the NASTRAN data deck and store the results in output file nds.dat.

2. Use PATRAN utilities available under the menu selection **Utilities/Results/Results Utilities** to create individual files containing nodal global stress outputs, one component per file (sxx.dat, syy.dat, and sxy.dat).

3. Use the Fortran program snode.f to combine the above files and calculate the values of the principal stresses, and store the results in output file lfsn.dat.

A similar process is used to create the element stress file containing the stresses at the element centroids. The element stresses are sometimes used to produce a less dense display of loadflow orientations only.

1. Use the Fortran program pelem.f to pull element lines from the NASTRAN data deck and store the results in output file elms.dat.

2. Use PATRAN utilities to create individual files containing the element centroid global stress outputs, one component per file (exx.dat, eyy.dat, and exy.dat).

3. Use the Fortran program selem.f to combine the above files and calculate the values of the principal stresses, and store the results in output file to lfse.dat.

The resulting files lfsn.dat and lfse.dat are the input files to the program for computing loadflow orientations, which is described in Appendix B. The different programs referred to above are listed in the following four sections.

A.2 Fortran program pnode.f for extracting nodal data lines

```
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc

      program pnode

C This program writes all grid points (nodes) from the NASTRAN data
C deck file whose name is given in the variable nameif.
```

```
C
C The results are written into the file whose name is given in the
C variable nameof.

      character aline*80,bline*80,tokens(5)*30
      character nameif*80,nameof*80
      integer   index

      nameif = 'lf0clrwk.dat'
      nameof = 'lf0clrwknode.dat'

      write(*,*) 'Name of input  file = ',nameif(1:index(nameif,' '))
      write(*,*) 'Name of output file = ',nameof(1:index(nameof,' '))

      open(unit=10,file=nameif)
      open(unit=20,file=nameof)

      n = 0

      do while (.true.)
        read(10,'(a)',end=2000) aline
        if (aline(1:4).eq.'GRID') then
          n = n + 1
          if (aline(5:5).eq.'*') then
            read(10,'(a)') bline
            aline=aline(1:71)//bline(9:)
            call gettokens(aline,' ',ntokens,tokens)
            call chkfloat(tokens(3))
            call chkfloat(tokens(4))
            call chkfloat(tokens(5))
            aline = tokens(2)(1:index(tokens(2),' '))//
     &              tokens(3)(1:index(tokens(3),' '))//
     &              tokens(4)(1:index(tokens(4),' '))//
     &              tokens(5)(1:index(tokens(5),' '))
            read(aline,*) nid,x,y,z
          else
            read(aline,'(8x,i8,8x,3f8.4)') nid,x,y,z
          endif
          write(20,'(a,i8,a,3(a,f12.4))')
     &          'GRID,',nid,',',',',x,',',y,',',z
        end if
      end do
 2000 continue

      write(*,*) 'Number of GRID points processed = ',n

      close(10)
      close(20)

      stop
      end

cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc

      subroutine chkfloat(s)

      character s*(*)

      integer   index

      n = index(s,' ')-1
      i = n-2
      if (i.gt.1.and.(s(i:i).eq.'-' .or. s(i:i).eq.'+')) then
        s = s(1:i-1)//'E'//s(i:)
      endif

      return
      end

cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc

      subroutine gettokens(string,delimiters,ntokens,tokens)

      implicit none

      integer   ntokens
      character string*(*),delimiters*(*),tokens(*)*(*)

      character token*80
```

```
      character strtok*80,char*1

      ntokens   = 0
      tokens(1) = strtok(string, delimiters)

      do while (token .ne. char(0))
        ntokens = ntokens + 1
        token   = strtok(char(0), delimiters)
        if (token.ne.char(0)) then
          tokens(ntokens+1) = token
        endif
      enddo

      return
      end

cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc

      character*80 function strtok(srce, delim)

C Tokenize a string in a similar manner to C. The usage is
C very similar to the C function.
C
C Input:  srce   = source string to tokenize (see usage note).
C         delim  = delimiter string.  used to determine the
C                  beginning/end of each token in a string.
C
C Output: strtok =   string token.
C
C Usage:  First call strtok with the string to tokenize
C         as srce, and the delimiter string used to tokenize.
C         Then call strtok with srce set to char(0).
C         If strtok returns char(0), no more tokens are
C         available.

      implicit none

      character srce*(*), delim*(*)

      character savestr*1024
      integer   start, ln, begpos, endpos

      save      start, savestr

      integer   index
      character char*1

      if (srce(1:1) .ne. char(0)) then
        start = 1
        if (len(srce).gt.len(savestr)) then
          write(*,*) '*** warning: source string too long for buffer.'
        endif
        savestr = srce
      endif

      begpos = start
      ln = len(savestr)

5     continue
      if ((begpos .le. ln) .and.
     &     (index(delim,savestr(begpos:begpos)) .ne. 0) ) then
        begpos = begpos + 1
        goto 5
      endif

      if (begpos .gt. ln) then
        strtok = char(0)
        return
      endif
      endpos = begpos

10    continue
      if ((endpos .le. ln) .and.
     &     (index(delim,savestr(endpos:endpos)) .eq. 0) ) then
        endpos = endpos + 1
        goto 10
      endif

      strtok = savestr(begpos:endpos-1)
      start  = endpos
```

```
      return
      end
```

A.3 Fortran program pelem.f for extracting element data lines

```
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc

      program pelem

C This program writes all elements from the NASTRAN data deck file
C whose name is given in the variable nameif.
C
C The results are written into the file whose name is given in the
C variable nameof.

      character aline*80,nameif*80,nameof*80
      integer   index

      nameif = 'lf0clrwk.dat'
      nameof = 'lf0clrwkelem.dat'

      write(*,*) 'Name of input  file = ',nameif(1:index(nameif,' '))
      write(*,*) 'Name of output file = ',nameof(1:index(nameof,' '))

      open(unit=10,file=nameif)
      open(unit=20,file=nameof)

      n = 0

      do while (.true.)
        read(10,'(a)',end=2000) aline
        if (aline(1:6).eq.'CQUAD4') then
          n = n + 1
          i = len(aline)
          do while (aline(i:i).eq.' ')
            i = i - 1
          end do
          write(20,'(a)') aline(1:i)
        end if
      end do
 2000 continue

      write(*,*) 'Number of elements processed = ',n

      close(10)
      close(20)

      stop
      end
```

A.4 Fortran program snode.f for combining nodal stresses

```
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc

      program snode

      character aline*80

      open(unit=10,file='lf0clrwknode.dat')
      open(unit=20,file='lf0clrwknxx.dat')
      open(unit=30,file='lf0clrwknyy.dat')
      open(unit=40,file='lf0clrwknxy.dat')
      open(unit=50,file='lf0clrwksn.dat')

      do i=1,18
        read(20,*)
        read(30,*)
        read(40,*)
      end do

      write(50,'(a8,2x,5(a10,2x))')
     &      'id','x','y','sxx','syy','sxy'
```

```
      n = 0

      do while (.true.)

        read(10,'(a)',end=10) aline
        aline = aline(6:13)//' '//aline(16:)
        read(aline,*) id,x,y,z
        read(20,*) idummy,jdummy,sxx
        read(30,*) idummy,jdummy,syy
        read(40,*) idummy,jdummy,sxy

        n = n+1

        write(50,'(i8,5(2x,f10.4))') id,x,y,sxx,syy,sxy

      end do

   10 write(*,*) 'Number of nodes processed = ',n

      close(10)
      close(20)
      close(30)
      close(40)
      close(50)

      stop
      end
```

A.5 Fortran program selem.f for combining element centroid stresses

```
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc

      program selem

      character slab*8

      open(unit=10,file='lf0clrwkelem.dat')
      open(unit=20,file='lf0clrwkexx.dat')
      open(unit=30,file='lf0clrwkeyy.dat')
      open(unit=40,file='lf0clrwkexy.dat')
      open(unit=50,file='lf0clrwkse.dat')

      do i=1,18
        read(20,*)
        read(30,*)
        read(40,*)
      end do

      write(50,'(5a8,3(2x,a10))') 'id','n1','n2','n3','n4',
     &                            'sxx','syy','sxy'

      n = 0

      do while (.true.)

        read(10,*,end=10) slab,id,njunk,n1,n2,n3,n4
        read(20,*) idummy,jdummy,sxx
        read(30,*) idummy,jdummy,syy
        read(40,*) idummy,jdummy,sxy

        n = n+1

        write(50,'(5i8,3(2x,f10.4))') id,n1,n2,n3,n4,sxx,syy,sxy

      end do

   10 write(*,*) 'Number of elements processed = ',n

      close(10)
      close(20)
      close(30)
      close(40)
      close(50)

      stop
      end
```

Appendix B

Determining Loadflow Orientations

B.1 General comments

A Fortran program lfvector.f was developed for determining the loadflow orientations from a set of datafiles containing node and element stresses. In our case, these intermediate datafiles are obtained by using the MSC.PATRAN FEA post-processing software to extract the desired stresses from the FE analysis performed by MSC.NASTRAN. The results produced by lfvector.f are written to an output file in a Tecplot-compatible format that enables Tecplot to be used for visualisation of the loadflow orientations for the structure that has been analysed. Both the x-direction and y-direction loadflow orientations are computed, and the results are stored in two formats: as a field of unit vectors with x- and y-components set to produce the required angular orientation (for use within Tecplot), and as angles in degrees. The program also computes the principal stress directions and stores them in a similar manner, enabling comparisons with loadflow orientations to be easily obtained. A listing of the lfvector.f program is provided in the next section.

B.2 Fortran program lfvector.f for calculating loadflow orientations

```
ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc

      program lfvector

C 10-06-1998 Witold Waldman
C
C This program computes the orientations of loadpath vectors from a
C set of global stress data (sxx, syy, and sxy) at a series of points.
C
C It then outputs the results of the calculations in Tecplot format.
C
C 17-08-1998 Witold Waldman
C
C Modified code to accept file names from command line. This is
C accomplished by making use of HP-UX operating system intrinsic
C functions and subroutines that are available in the Fortran language.
C To link with the required support libraries, it is necessary to
C supply the +U77 command line switch to the Hewlett-Packard f77
C compiler.
C
C 21-05-1999 Witold Waldman
C
C Modified code by updating the equations used to compute loadflow
C orientations to the latest theoretical formulation.

      implicit none

      integer   lui,luo,maxnode,maxelem,maxnptr

      parameter (lui=1)
      parameter (luo=2)
      parameter (maxnode=100000)
      parameter (maxelem=100000)
      parameter (maxnptr=100000)

      character nodefile*80
      character elemfile*80
```

```
character vectorfile*80
integer   i,j
integer   ios
integer   nelem
integer   nnode
integer   nlist
integer   nclarg
integer   lenn
integer   lene
integer   lenv
integer   nodeptr(maxnptr)
integer   nodenum(maxnode)
integer   nodelist(maxnode)
real*8    nodex(maxnode)
real*8    nodey(maxnode)
real*8    nodesxx(maxnode)
real*8    nodesyy(maxnode)
real*8    nodesxy(maxnode)
real*8    nodes11(maxnode)
real*8    nodes22(maxnode)
real*8    nodet11(maxnode)
real*8    nodet22(maxnode)
real*8    nodexsn(maxnode)
real*8    nodextau(maxnode)
real*8    nodexalpha(maxnode)
real*8    nodeysn(maxnode)
real*8    nodeytau(maxnode)
real*8    nodeyalpha(maxnode)
real*8    lists11(maxnode)
real*8    lists22(maxnode)
real*8    listt11(maxnode)
real*8    listt22(maxnode)
real*8    listxsn(maxnode)
real*8    listxtau(maxnode)
real*8    listxalpha(maxnode)
real*8    listysn(maxnode)
real*8    listytau(maxnode)
real*8    listyalpha(maxnode)
integer   elemnum(maxelem)
integer   elemn1(maxelem)
integer   elemn2(maxelem)
integer   elemn3(maxelem)
integer   elemn4(maxelem)
real*8    elemxc(maxelem)
real*8    elemyc(maxelem)
real*8    elemsxx(maxelem)
real*8    elemsyy(maxelem)
real*8    elemsxy(maxelem)
real*8    elems11(maxelem)
real*8    elems22(maxelem)
real*8    elemt11(maxelem)
real*8    elemt22(maxelem)
real*8    elemxsn(maxelem)
real*8    elemxtau(maxelem)
real*8    elemxalpha(maxelem)
real*8    elemysn(maxelem)
real*8    elemytau(maxelem)
real*8    elemyalpha(maxelem)
real*8    x1,x2,x3,x4
real*8    y1,y2,y3,y4
logical   doselectednodes
logical   doelements
logical   donodes

real*8    cosd
real*8    sind
integer   jmod
integer   iargc

! Specify the names of the data files that contain stress results
! for individual nodes and elements, as well as the name of the
! output file.

nclarg=iargc()

if (nclarg.eq.3) then
  call getarg(1,nodefile)
  call getarg(2,elemfile)
  call getarg(3,vectorfile)
else
```

```
   write(*,'(/,1x,a,/)')
&    'Usage: lfvector nodalstressfile elementstressfile outputfile'
   stop
 endif

 call lenstr(nodefile,lenn)
 call lenstr(elemfile,lene)
 call lenstr(vectorfile,lenv)

 write(*,*)
 write(*,*) 'File of node stresses (input) : ',nodefile(1:lenn)
 write(*,*) 'File of elem stresses (input) : ',elemfile(1:lene)
 write(*,*) 'File of load vectors  (output): ',vectorfile(1:lenv)
 write(*,*)

 doselectednodes=.false.
 doelements=.true.
 donodes=.true.

 ! Read in the stress results for the centre of each element.

 open(unit=lui,file=elemfile,status='old')
 i=0
 ios=0
 read(lui,*)
 do while (ios.eq.0)
   i=i+1
   read(lui,*,iostat=ios) elemnum(i),
&                         elemn1(i),elemn2(i),elemn3(i),elemn4(i),
&                         elemsxx(i),elemsyy(i),elemsxy(i)
   if (ios.eq.0) then
     call s1s2t1t2(elemsxx(i),elemsyy(i),elemsxy(i),
&                  elems11(i),elems22(i),elemt11(i),elemt22(i))
   endif
 enddo
 close(lui)
 nelem=i-1
 write(*,*) 'Number of elements = ',nelem

 ! Read in the stress results at the nodes.

 open(unit=lui,file=nodefile,status='old')
 i=0
 ios=0
 read(lui,*)
 do while (ios.eq.0)
   i=i+1
   read(lui,*,iostat=ios) nodenum(i),nodex(i),nodey(i),
&                         nodesxx(i),nodesyy(i),nodesxy(i)
   if (ios.eq.0) then
     nodeptr(nodenum(i))=i
     call s1s2t1t2(nodesxx(i),nodesyy(i),nodesxy(i),
&                  nodes11(i),nodes22(i),nodet11(i),nodet22(i))
   endif
 enddo
 close(lui)
 nnode=i-1
 write(*,*) 'Number of nodes    = ',nnode

 ! Compute and store the coordinates of the centre of each element.

 do i=1,nelem
   x1=nodex(nodeptr(elemn1(i)))
   x2=nodex(nodeptr(elemn2(i)))
   x3=nodex(nodeptr(elemn3(i)))
   x4=nodex(nodeptr(elemn4(i)))
   y1=nodey(nodeptr(elemn1(i)))
   y2=nodey(nodeptr(elemn2(i)))
   y3=nodey(nodeptr(elemn3(i)))
   y4=nodey(nodeptr(elemn4(i)))
   elemxc(i)=(x1+x2+x3+x4)/4.0d0
   elemyc(i)=(y1+y2+y3+y4)/4.0d0
 enddo

 ! Compute angle of the loadpath vector for selected nodes.
 ! This feature may be of assistance when debugging.

 if (doselectednodes) then
   nlist = 2
   nodelist(1) = 7
```

```
      nodelist(2) = 1
      call wrtaxismsg(0,' for selected nodes')
      do i=1,nlist
        j=nodeptr(nodelist(i))
        write(*,*) 'Processing node',nodenum(j),'.'
        call s1s2t1t2(nodesxx(j),nodesyy(j),nodesxy(j),
&                     lists11(i),lists22(i),listt11(i),listt22(i))
        call sntau(1,nodesxx(j),nodesyy(j),nodesxy(j),
&                  listxsn(i),listxtau(i),listxalpha(i))
        call sntau(2,nodesxx(j),nodesyy(j),nodesxy(j),
&                  listysn(i),listytau(i),listyalpha(i))
      enddo
    endif

    ! Compute angle of the loadpath vector at each of the nodes.

    if (donodes) then
      call wrtaxismsg(0,' for all nodes')
      do i=1,nnode
        call sntau(1,nodesxx(i),nodesyy(i),nodesxy(i),
&                  nodexsn(i),nodextau(i),nodexalpha(i))
        call sntau(2,nodesxx(i),nodesyy(i),nodesxy(i),
&                  nodeysn(i),nodeytau(i),nodeyalpha(i))
        if (jmod(i,10000).eq.0) write(*,*) i,' nodes processed.'
      enddo
      if (jmod(nnode,10000).ne.0) write(*,*) nnode,' nodes processed.'
    endif

    ! Compute angle of the loadpath vector at the centre of
    ! each of the elements.

    if (doelements) then
      call wrtaxismsg(0,' for all elements')
      do i=1,nelem
        call sntau(1,elemsxx(i),elemsyy(i),elemsxy(i),
&                  elemxsn(i),elemxtau(i),elemxalpha(i))
        call sntau(2,elemsxx(i),elemsyy(i),elemsxy(i),
&                  elemysn(i),elemytau(i),elemyalpha(i))
        if (jmod(i,10000).eq.0) write(*,*) i,' elements processed.'
      enddo
      if (jmod(nelem,10000).ne.0) write(*,*) nelem,
&                                            ' elements processed.'
     endif

    ! Write the results of the stress calculations to an output
    ! file in Tecplot format.

    if (donodes.or.doelements.or.doselectednodes) then
      write(*,*)
&       'Writing loadflow orientations to Tecplot output file.'
      open(unit=luo,file=vectorfile)
      write(luo,'(a)')
&       'title="Loadflow orientations and stress results"'
      write(luo,'(a)') 'variables='
      write(luo,'(a6,19a13)')
&       '"id"','"x"','"y"','"sxx"','"syy"','"sxy"',
&       '"s11"','"theta11"','"s11x"','"s11y"',
&       '"s22"','"theta22"','"s22x"','"s22y"',
&       '"xlfth"','"xlfthx"','"xlfthy"',
&       '"ylfth"','"ylfthx"','"ylfthy"'
      if (doselectednodes) then
        write(*,*) 'Zone: selected nodes.'
        write(luo,'(a)') 'zone t="Selected nodes"'
        do i=1,nlist
          j=nodeptr(nodelist(i))
          write(luo,'(i6,19f13.4)')
&           nodenum(j),nodex(j),nodey(j),
&           nodesxx(j),nodesyy(j),nodesxy(j),
&           lists11(i),listt11(i),cosd(listt11(i)),sind(listt11(i)),
&           lists22(i),listt22(i),cosd(listt22(i)),sind(listt22(i)),
&           listxalpha(i),cosd(listxalpha(i)),sind(listxalpha(i)),
&           listyalpha(i),cosd(listyalpha(i)),sind(listyalpha(i))
        enddo
      endif
      if (donodes) then
        write(*,*) 'Zone: all nodes.'
        write(luo,'(a,i6,a,i6,a)')
&           'zone t="Nodes" N=',nnode,' E=',nelem,
&           ' F=FEPOINT ET=QUADRILATERAL'
        do i=1,nnode
```

```
          write(luo,'(i6,19f13.4)')
     &      nodenum(i),nodex(i),nodey(i),
     &      nodesxx(i),nodesyy(i),nodesxy(i),
     &      nodes11(i),nodet11(i),cosd(nodet11(i)),sind(nodet11(i)),
     &      nodes22(i),nodet22(i),cosd(nodet22(i)),sind(nodet22(i)),
     &      nodexalpha(i),cosd(nodexalpha(i)),sind(nodexalpha(i)),
     &      nodeyalpha(i),cosd(nodeyalpha(i)),sind(nodeyalpha(i))
        enddo
        do i=1,nelem
          write(luo,'(4i8)') nodeptr(elemn1(i)),nodeptr(elemn2(i)),
     &                       nodeptr(elemn3(i)),nodeptr(elemn4(i))
        enddo
      endif
      if (doelements) then
        write(*,*) 'Zone: all elements.'
        write(luo,'(a)') 'zone t="Elements"'
        do i=1,nelem
          write(luo,'(i6,19f13.4)')
     &      i,elemxc(i),elemyc(i),
     &      elemsxx(i),elemsyy(i),elemsxy(i),
     &      elems11(i),elemt11(i),cosd(elemt11(i)),sind(elemt11(i)),
     &      elems22(i),elemt22(i),cosd(elemt22(i)),sind(elemt22(i)),
     &      elemxalpha(i),cosd(elemxalpha(i)),sind(elemxalpha(i)),
     &      elemyalpha(i),cosd(elemyalpha(i)),sind(elemyalpha(i))
        enddo
      endif
      close(luo)
    endif

    stop
    end

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

    subroutine wrtaxismsg(iaxis,msg)

    implicit none

    integer   iaxis
    character msg*(*)

    if (iaxis.eq.1) then
      write(*,*) 'Computing x-direction loadpath orientation'//
     &           msg//'.'
    else if (iaxis.eq.2) then
      write(*,*) 'Computing y-direction loadpath orientation'//
     &           msg//'.'
    else
      write(*,*) 'Computing x- & y-direction loadpath '//
     &           'orientation'//msg//'.'
    endif

    return
    end

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

    subroutine s1s2t1t2(sxx,syy,sxy,s1,s2,theta1,theta2)

C This subprogram computes the principal stresses s1 and s2 oriented
C at angles theta1 and theta2 using the stresses sxx, syy, and sxy.
C
C s1 is the numerically largest principal stress and theta1 is its
C associated angle.
C
C s2 is the numerically smallest principal stress and theta2 is its
C associated angle.
C
C Note that, by the definition of the principal angles obtained from
C "Mechanics of Materials" by Higdon et al., one of the principal
C stress angles lies between +/-45 degrees (inclusive), and the other
C is 90 degrees greater.

    implicit none

    real*8 sxx,syy,sxy,s1,s2,theta1,theta2

    real*8 ps1,ps2,pth1,pth2

    real*8 datand,dcosd,dsind
```

```
      if (sxx-syy.ne.0.0d0) then
        pth1 = datand(2.0d0*sxy/(sxx-syy))/2.0d0
      else
        if (sxy.gt.0.0d0) then
          pth1 = +45.0d0
        else if (sxy.lt.0.0d0) then
          pth1 = -45.0d0
        else
          pth1 = 0.0d0
        endif
      endif

      pth2 = pth1 + 90.0d0

      if (pth2.gt.90.0d0) pth2 = pth2 - 180.0d0

      ps1 = (sxx+syy)/2.0d0 + (sxx-syy)/2.0d0*dcosd(2.0d0*pth1) +
     &      sxy*dsind(2.0d0*pth1)

      ps2 = (sxx+syy)/2.0d0 + (sxx-syy)/2.0d0*dcosd(2.0d0*pth2) +
     &      sxy*dsind(2.0d0*pth2)

      if (ps1.ge.ps2) then
        s1 = ps1
        s2 = ps2
        theta1 = pth1
        theta2 = pth2
      else
        s1 = ps2
        s2 = ps1
        theta1 = pth2
        theta2 = pth1
      endif

      return
      end

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      subroutine snnsnt(sxx,syy,sxy,theta,snn,snt)

C This subprogram computes the normal and shearing stresses snn and snt
C on an arbitrary plane through a point oriented at an angle theta
C with respect to a reference x axis and the known stresses sxx, syy
C and sxy.

      implicit none

      real*8  sxx,syy,sxy,theta,snn,snt

      real*8  sintheta,costheta,sin2theta,cos2theta

      real*8  dsind,dcosd

      sintheta  = dsind(theta)
      costheta  = dcosd(theta)
      sin2theta = sintheta**2
      cos2theta = costheta**2

      snn = sxx*cos2theta + syy*sin2theta + 2.0d0*sxy*sintheta*costheta

      snt = -(sxx-syy)*sintheta*costheta + sxy*(cos2theta-sin2theta)

      return
      end

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      subroutine sntau(iaxis,sxx,syy,sxy,sn,tau,alpha)

C This subprogram determines the angle alpha which causes the x- or
C y-components of the loadflow to be zero. The associated values of
C normal stress sn and shear stress tau are also computed for the
C estimated value of alpha.

      implicit none

      integer iaxis
      real*8  sxx,syy,sxy,sn,tau,alpha
```

```fortran
      real*8  theta

      real*8  datan2d

      if (iaxis.ne.1 .and. iaxis.ne.2) then
        write(*,*) '*** Invalid value for iaxis was supplied.'
        stop
      endif

      if (iaxis.eq.1) then
        ! Consider force equilibrium in x-direction.
        alpha = datan2d(sxy,sxx)
      else if (iaxis.eq.2) then
        ! Consider force equilibrium in y-direction.
        alpha = datan2d(syy,sxy)
      endif

      theta = alpha - 90.0d0

      call snnsnt(sxx,syy,sxy,theta,sn,tau)

      return
      end

cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc

      subroutine lenstr(s,ls)

C Determine the length ls of a string s.

      implicit none

      character s*(*)  ! String.
      integer   ls     ! Number of characters in string.

      integer   i

      ! Determine the location of the rightmost non-blank character
      ! and use this to define the length of the string.

      ls = len(s)
   10 if (s(ls:ls).eq.' ') then
        ls = ls - 1
        if (ls.gt.0) goto 10
      endif

      ! Check to see if a null character is present, and adjust the
      ! length of the string to exclude this.

      do i=1,ls
        if (ichar(s(i:i)).eq.0) then
          ls = i - 1
          goto 20
        endif
      enddo
   20 continue

      return
      end
```

Appendix C

Loadpath Visualisation Using Tecplot

The following procedure has been successfully used for loadpath visualisation using the Tecplot data visualisation program [18]:

1. Conduct a FEA of the structure of interest and compute the stresses σ_x, σ_y and τ_{xy} at each of the nodes in the FE model. Store the nodal coordinates and corresponding nodal stresses in a file for subsequent post-processing.

2. Use the previously computed stresses to resolve the loadflow orientations in the x- and y-directions by using Equation (11) and Equation (12). The loadflow orientation can be represented by unit vectors with appropriate x- and y-components (direction cosines).

3. Produce a data file in Tecplot FEPOINT format (see [18], Chapter 16, Working with Finite-Element Data) where the (x, y) coordinates of the nodal points are stored, together with the direction cosines associated with the computed loadflow orientation at each of the points. This defines a vector-like field of unit vectors with their associated orientations.

4. Open Tecplot and read in direction cosines of the loadflow orientation, and then activate the Vector layer in the Tecplot sidebar.

5. Set the x- and y-components of the vector field. This is accomplished by choosing **Vector Variables** from the **Field** menu.

6. Once the Vector checkbox has been selected and the vector components have been chosen, you can redraw the Tecplot screen and see the vector plot.

7. If desired, it is possible to alter vector plot attributes using the **Vector Attributes** dialog. Tecplot allows four different types of vectors: tail at point, anchored at midpoint, head at point, and head only. Anchored at midpoint is a good choice. Now, because loadflow orientations are not a true vector quantity, it is useful to remove the arrowhead from the vector display. This can be accomplished by setting the size of the vector arrowheads to zero by choosing **Vector Arrowheads** from the **Field** menu.

8. In Tecplot, it is also possible to modify the characteristics of many other aspects of vector plots (e.g. vector length). For further details see [18], Chapter 11, Creating Vector Plots.

9. Having defined the vector components to be associated with the field of loadflow orientations, you can place streamtraces one at a time or in groups called rakes (see [18], Chapter 12, Streamtraces). A streamtrace rake is a set of streamtraces with starting points along a given line. You can place streamtraces or streamtrace rakes by choosing **Streamtrace Placement** from the **Field** menu. This brings up the Streamtrace Placement dialog. To place a rake of streamlines. Select the **Enter Rake Positions** checkbox and

enter the end positions for the rake as (x, y) coordinates. After defining where streamtraces are to start, press the **Place Streamtrace(s)** button to have them drawn on the plot. If desired, you toggle the display of the loadflow (vector) orientations by clicking the **Vector** button in the Tecplot sidebar.

10. In some cases, streamtraces will spiral rather than producing the desired closed contour. It is possible to use a streamtrace termination line that terminates any streamtraces that cross it. To create a streamtrace termination line, from the **Term Line** page of the **Streamtrace Details** dialog, click the **Draw Stream Term Line** button. You can then use the mouse to click the desired starting point for the termination line. Then click at addition points to define the desired polyline. To end the termination line, press the Esc key. Note that only one termination line can exist at any one time in a given frame. You can control the streamtrace termination line from the **Term Line** page of the **Streamtrace Details** dialog (e.g. activating and deactivating).

Tecplot uses a predictor-corrector integration algorithm to calculate streamtraces [18]. The basic idea is to create the streamtrace by moving in a series of small steps from the starting point in the direction of, or in opposition to, the local vector field. Each step is only a fraction of a cell or element. Tecplot automatically adjusts the step size based on local cell shape and vector field variation. The streamtrace integration procedure can be controlled by modifying parameters in the Integration page of the **Streamtrace Details** dialog.

DISTRIBUTION LIST

Advances in Structural Loadflow Visualisation and Applications to Optimal Shapes

W. Waldman, M. Heller, R. Kaye and L.R.F. Rose

AUSTRALIA

DEFENCE ORGANISATION

Task sponsor
RAAF DTA-LC

S&T Program
Chief Defence Scientist
FAS Science Policy
AS Science Corporate Management
} shared copy
Director General Science Policy Development
Counsellor Defence Science, London (Doc Data Sheet only)
Counsellor Defence Science, Washington (Doc Data Sheet only)
Scientific Adviser to MRDC Thailand (Doc Data Sheet only)
Scientific Adviser Policy and Command
Navy Scientific Adviser (Doc Data Sheet and distribution list only)
Scientific Adviser - Army (Doc Data Sheet and distribution list only)
Air Force Scientific Adviser
Director Trials

Aeronautical and Maritime Research Laboratory
Director

Chief of Airframes and Engines Division
Chief of Maritime Platforms Division
Research Leader Aerospace Composite Structures
Research Leader Propulsion
Research Leader Structural Dynamics
Research Leader Structural Integrity
Authors: W. Waldman (5 copies)
M. Heller (10 copies)
R. Kaye
L.R.F. Rose
R. Callinan
P. Chalkley
R. Chester
D. Graham
J. Hou
D. Lombardo
L. Molent
S. Pitt
S. Sanderson
K. Watters
B. Wicks
A. Wong

DSTO Library and Archives
Library Fishermans Bend
Library Maribyrnong
Library Salisbury (2 copies)
Australian Archives
Library, MOD, Pyrmont (Doc Data Sheet only)
Library, MOD, HMAS Stirling (Doc Data Sheet only)
US Defense Technical Information Center (2 copies)
UK Defence Research Information Centre (2 copies)
Canada Defence Scientific Information Service (1 copy)
NZ Defence Information Centre (1 copy)
National Library of Australia (1 copy)

Capability Development Division
Director General Maritime Development (Doc Data Sheet only)
Director General Land Development (Doc Data Sheet only)
Director General C3I Development (Doc Data Sheet only)
Director General Aerospace Development (Air)

Air Force
ASI-LC
TLFM SQN Williamtown (Chief Engineer)
Aircraft Research and Development Unit
Tech Reports, CO Engineering Squadron, ARDU

Army
ABCA Standardisation Officer, Puckapunyal (4 copies)
SO (Science), DJFHQ(L), MILPO Enoggera, Queensland 4051 (Doc Data Sheet only)
NAPOC QWG Engineer NBCD c/- DENGRS-A, HQ Engineer Centre Liverpool Military Area, NSW 2174 (Doc Data Sheet only)

Navy
SO (Science), Director of Naval Warfare, Maritime Headquarters Annex, Garden Island, NSW 2000 (Doc Data Sheet only)

Intelligence Program
DGSTA Defence Intelligence Organisation
Defence Intelligence Organisation-Information Centre

Corporate Support Program
OIC TRS, Defence Regional Library, Canberra

UNIVERSITIES AND COLLEGES
Australian Defence Force Academy
Library
Head of Aerospace and Mechanical Engineering
Deakin University, Serials Section (M list), Deakin University Library, Geelong, 3217
Senior Librarian, Hargrave Library, Monash University
Librarian, Flinders University

OTHER ORGANISATIONS

NASA (Canberra)
AGPS

OUTSIDE AUSTRALIA

ABSTRACTING AND INFORMATION ORGANISATIONS

Library, Chemical Abstracts Reference Service
Engineering Societies Library, USA
Materials Information, Cambridge Scientific Abstracts, USA
Documents Librarian, The Center for Research Libraries, USA

INFORMATION EXCHANGE AGREEMENT PARTNERS

Acquisitions Unit, Science Reference and Information Service, UK
Library - Exchange Desk, National Institute of Standards and Technology, USA
National Aerospace Laboratory, Japan
National Aerospace Laboratory, Netherlands

CANADA

Bombardier Inc, Canadair, Military Aircraft Division
N.N. Trong
CDSD Engineering Manager - J. Roussel
Canadian Forces National Defence Headquarters
DAS ENG 6-3 (2 copies)
National Research Council, Institute for Aerospace Research, Ottawa
Library
B.M.K. Lee
D. Simpson

SPARES: 5 copies
TOTAL: 91 copies

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION
DOCUMENT CONTROL DATA

1. PRIVACY MARKING/CAVEAT (OF DOCUMENT)

2. TITLE

Advances in Structural Loadflow Visualisation and Applications to Optimal Shapes

3. SECURITY CLASSIFICATION

Document	(U)
Title	(U)
Abstract	(U)

4. AUTHORS

W. Waldman, M. Heller, R. Kaye and L.R.F. Rose

5. CORPORATE AUTHOR

Aeronautical and Maritime Research Laboratory
PO Box 4331
Melbourne VIC 3001
Australia

6a. DSTO NUMBER	6b. AR NUMBER	6c. TYPE OF REPORT	7. DOCUMENT DATE
DSTO-RR-0166	AR-011-110	Research Report	October 1999

8. FILE NUMBER	9. TASK NUMBER	10. TASK SPONSOR	11. NO. OF PAGES	12. NO. OF REFERENCES
M1/9/608	AIR 98/220	RAAF DTA-LC	58	18

13. DOWNGRADING/DELIMITING INSTRUCTIONS

None

14. RELEASE AUTHORITY

Chief, Airframes and Engines Division

15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT

Approved for public release

OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE, PO BOX 1500, SALISBURY, SA 5108

16. DELIBERATE ANNOUNCEMENT — No limitations

17. CASUAL ANNOUNCEMENT — Yes

18. DEFTEST DESCRIPTORS

shape, structural optimization, holes (openings), finite element analysis, stress concentration

19. ABSTRACT

Currently there is no generally accepted procedure for calculation of structural loadpaths, which would show how remote loads are equilibrated through a structure and could provide insight into how well a structure is performing its intended load-carrying functions. Kelly and Elsley have recently proposed a method for computing loadflow orientations and loadpaths using finite element results, which is based on iterative solutions of non-linear equations. In this paper, we have enhanced their theoretical formulation and general procedure by deriving explicit expressions for computing loadflow orientations. The new equations produce more accurate loadflow orientations compared to the prior approach and improve the fidelity of calculated loadpaths. In a series of benchmark problems, we have investigated non-optimal and optimal holes in plates using loadflow visualisation to identify their key features. We found that recirculation is apparent for non-optimal hole shapes, whereas no recirculation zone is present for optimal shapes. Although very highly refined finite element meshes were utilised, the implications are that even more refined meshes are required to fully capture the complex behaviour that exists in recirculation zones. The removal of the recirculation zone for a non-optimal shape leads to a better shape, but the improvement in peak stress is insignificant. The calculation of loadflow orientations using the new equations is simple, and could be used with any finite element analysis code, while a plotting package is required to display loadpaths. Loadflow visualisation is a powerful tool for use by structural designers to improve their understanding of structural performance, the application of which can potentially result in worthwhile improvements in structural efficiency.