

Titre: Numerical solutions of compressible inviscid flows in complex
Title: internal ducts

Auteur: Yao Wang
Author:

Date: 1989

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Wang, Y. (1989). Numerical solutions of compressible inviscid flows in complex
Citation: internal ducts [Mémoire de maîtrise, École Polytechnique de Montréal].
PolyPublie. <https://publications.polymtl.ca/56719/>

 **Document en libre accès dans PolyPublie**
Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/56719/>
PolyPublie URL:

**Directeurs de
recherche:** Ricardo Camarero, & Marcelo Reggio
Advisors:

Programme: Mathématiques appliquées
Program:

CA2P

UP 8

1989

W246

UNIVERSITÉ DE MONTRÉAL

Numerical Solutions of Compressible Inviscid Flows
in Complex Internal Ducts

par

Yao WANG

DÉPARTEMENT DE MATHÉMATIQUES APPLIQUÉES
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU GRADE DE MAÎTRE ÈS SCIENCES APPLIQUÉES (M.Sc.A)
(MATHÉMATIQUES APPLIQUÉES)

Août 1989

©Yao Wang 1989



National Library
of Canada

Bibliothèque nationale
du Canada

Canadian Theses Service Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-58203-0

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE

Ce mémoire est intitulé:

NUMERICAL SOLUTIONS OF COMPRESSIBLE INVISCID FLOWS IN
COMPLEX INTERNAL DUCTS

présenté par: M. Yao Wang

en vue de l'obtention du grade de: MAÎTRE ÈS SCIENCES APPLIQUÉES

(M.Sc.A) a été dûment accepté par le jury d'examen constitué de:

M. René Kahawita Ph.D., Président

M. Ricardo Camarero Ph.D.

M. Marcelo Reggio Ph.D.

M. Robert Jeanjean Ph.D.

Sommaire

Le disjoncteur est un appareil utilisé sur les réseaux de transport d'énergie à haute tension. Il a deux tâches distinctes à accomplir: celle d'interrupteur et celle de fusible. Dans la première, l'appareil est employé pour la mise en service et hors de service de lignes de transport en situation normale. Dans la seconde, le disjoncteur joue le rôle de protecteur de ces mêmes lignes lors de la production d'une défaillance, par exemple, un court-circuit. Pour parvenir à ses deux objectifs, le disjoncteur emploie un mécanisme dans lequel des électrodes initialement en contact sont séparées lors de l'ordre d'ouverture. Un arc électrique s'allume alors entre les électrodes, arc par lequel le courant de la ligne continue de circuler. Simultanément à la séparation des électrodes, un gaz est comprimé dans un cylindre et un jet axisymétrique de gaz est dirigé sur l'arc. Ce jet procure le confinement et le refroidissement de l'arc, jusqu'à son extinction. Actuellement, le développement d'un nouveau modèle de disjoncteur nécessite une longue phase d'essais en laboratoire sur des prototypes, et très peu d'outils d'analyse sont disponibles pour aider le concepteur dans cette tâche de développement.

Dans ce contexte, la disponibilité d'un outil informatique pour l'obtention de la physique de l'écoulement pourrait permettre, non seulement de réduire le nombre d'essais en laboratoire, mais également une meilleure connaissance des paramètres

de design.

Dans le cadre de ce travail, une simulation numérique des écoulements de gaz compressibles nonvisqueux dans les conduites est effectuée dans le but de prédire les performances d'un disjoncteur industriel lors de son ouverture. Le problème est caractérisé par deux difficultés: une géométrie complexe et la présence d'ondes de choc résultant de l'écoulement. Les équations d'Euler pour des écoulements 2-D, axi-symétriques sont proposées comme modèle de base pour la simulation. La résolution des équations de l'écoulement compressible a été effectuée à l'aide d'un schéma de volumes finis. Deux discrétisations spatiales avec deux types d'intégrations dans le temps sont évaluées pour résoudre numériquement les équations d'Euler. Une nouvelle technique de maillage triangulaire nonstructuré et adaptatif est utilisée. Le schéma de calcul utilise une intégration de Runge-Kutta sur des volumes de contrôles où toutes les variables sont stockées aux noeuds. La procédure numérique est programmée et expertisée pour deux cas simples. Les résultats du traitement sur ordinateur sont comparés soit avec la solution analytique, soit avec les données expérimentales. Ensuite, l'écoulement dans une géométrie réelle est calculé et les effets du maillage et de la viscosité artificielle sont évalués.

L'objectif de cette étude est l'élaboration d'un cadre informatique pour la simulation numérique. Dans le chapitre 1 on décrit brièvement les équations d'Euler — un cas particulier des équations de conservation généralisées et la transformation de coordonnées. Le chapitre 2 présente les techniques de discrétisations spatiales et les itérations dans le temps. Enfin le chapitre 3 présente les calculs numériques exécutés avec les techniques décrites au chapitre 3.

Le schéma numérique proposé nécessite une certaine viscosité artificielle pour

lisser les instabilités qui apparaissent en présence d'un choc. L'exécution et la validation ont montré que la dissipation numérique joue un rôle décisif dans la qualité des résultats. Un amortissement insuffisant ou trop lourd diminue la précision de la résolution numérique. Les valeurs optimales des paramètres ont été obtenues par essai et erreur et sont valables pour des configurations données. Une démarche automatique pour le choix de ces valeurs serait préférable à cette approche heuristique et pourrait constituer un prochain sujet de recherche.

Abstract

A numerical simulation of compressible inviscid gas flows in complicated ducts is undertaken subject to the practical requirement of predicting the performance of an industrial gas circuit breaker during the discharge phase. The problem is characterized by two difficulties: the complex geometries and the presence of shocks in the resulting flow pattern. The Euler equations for 2-D and axisymmetric compressible flows are proposed as the elementary mathematical model for such a simulation. Two spatial discretisations and two time-integration processes are surveyed as the related numerical solvers to the Euler equations. A novel unstructured adaptive triangular mesh is utilized in conjunction with the cell vertex scheme and the Runge-Kutta process for this problem. The numerical procedure is programmed and tested under two simple cases, the computed results are compared with either the analytic solution or the experimental data. Finally, the flow in a real geometry is calculated and the effects of grid and artificial viscosity are evaluated.

Acknowledgements

I owe a great debt of gratitude to my academic directors, Professors Ricardo Camarero and Marcelo Reggio, for their supervising this work. I wish to thank Dr. Jean-Yves Trépanier, Mr. Yves Lauzé and Dr. He-rong Yang for their help in the computational aspects. I am also indebted to Mrs. Anne-Marie Goulet, who typed the manuscript.

The financial support from the NSERC of Canada and Cegelec Industries is acknowledged.

Contents

| | |
|---|-----------|
| Sommaire | iv |
| Abstract | vii |
| Acknowledgements | viii |
| List of Figures | xi |
| List of Tables | xiv |
| Introduction | 1 |
| 1 Mathematical Model | 3 |
| 1.1 Euler equations | 3 |
| 1.2 Transformation of the governing equations | 6 |
| 2 Numerical Methods | 10 |
| 2.1 Spatial discretisations | 11 |
| 2.1.1 Cell centre scheme | 12 |
| 2.1.2 Cell vertex scheme | 14 |
| 2.1.3 Some properties of the two schemes | 19 |

| | |
|---|-----------|
| | x |
| 2.2 Time stepping iterations | 21 |
| 2.2.1 Lax-Wendroff algorithms | 22 |
| 2.2.2 Runge-Kutta algorithms | 29 |
| 2.3 Other computational points | 32 |
| 2.3.1 The boundary conditions | 32 |
| 2.3.2 The form of dissipation | 33 |
| 2.3.3 The criterion of mesh refinement | 34 |
| 3 Computational Results | 35 |
| 3.1 Straight channel flow | 35 |
| 3.2 Nozzle flow | 36 |
| 3.3 Complex duct flow | 39 |
| 3.3.1 The effects of grid and iteration | 40 |
| 3.3.2 The effects of the additional dissipation | 42 |
| 3.3.3 Summary | 43 |
| Conclusions | 95 |
| References | 97 |

List of Figures

| | | |
|-----|---|----|
| 2.1 | Geometry for flow variables at cell centres: (a) general quadrilateral mesh; (b) nonuniform rectangular mesh; (c) triangular mesh. | 17 |
| 2.2 | Geometry for flow variables at cell vertices: (a) the vertices around cell C ; (b) the cells used to update vertex 1; (c) triangular mesh. . . | 18 |
| 3.1 | (a) computational domain and mesh with 152 triangles; (b) comparison of numerical solution with analytic one in terms of pressure and Mach number vs. position along the channel. | 44 |
| 3.2 | (a) domain and mesh with 550 triangles; (b) comparison of numerical result with experimental data in terms of Mach number vs. wall position. | 46 |
| 3.3 | (a) domain and mesh with 1032 triangles; (b) comparison of numerical result with experimental data in terms of Mach number vs. wall position. | 48 |
| 3.4 | (a) domain and mesh with 1433 triangles; (b) comparison of numerical result with experimental data in terms of Mach number vs. wall position. | 50 |

| | | |
|------|--|----|
| 3.5 | (a) domain and mesh with 2795 triangles; (b) comparison of numerical result with experimental data in terms of Mach number vs. wall position. | 52 |
| 3.6 | (a) domain and mesh with 3297 triangles; (b) comparison of numerical result with experimental data in terms of Mach number vs. wall position. | 54 |
| 3.7 | (a) domain and mesh with 1425 triangles; (b) comparison of numerical result with experimental data in terms of Mach number vs. wall position. | 56 |
| 3.8 | (a) domain and mesh with 1920 triangles; (b) solution on mesh <i>a</i> ; (c) solution with more iterations on <i>b</i> | 58 |
| 3.9 | (a) configuration of the interior of a gas circuit breaker; (b) computational domain; (c) zoning pattern. | 61 |
| 3.10 | (a) mesh with 672 triangles; (b) lines of constant Mach number; (c) positions of line segments; (d) variation of Mach number along the axis; (e) variation of Mach number along \overline{AB} | 64 |
| 3.11 | (a) area to be refined; (b) refined mesh with 1594 triangles; (c) lines of constant Mach number; (d) variation of Mach number along the axis; (e) variation of Mach number along \overline{AB} | 69 |
| 3.12 | (a) area to be refined; (b) refined mesh with 2747 triangles; (c) lines of constant Mach number; (d) variation of Mach number along the axis; (e) variation of Mach number along \overline{AB} | 74 |
| 3.13 | (a) area to be refined; (b) refined mesh with 4482 triangles; (c) lines of constant Mach number; (d) variation of Mach number along the axis; (e) variation of Mach number along \overline{AB} | 79 |

| | | |
|------|---|----|
| 3.14 | Result obtained on mesh in Figure 3.13 <i>b</i> , (<i>a</i>) lines of constant Mach number; (<i>b</i>) variation of Mach number along the axis; (<i>c</i>) variation of Mach number along \overline{AB} | 84 |
| 3.15 | Result obtained on mesh in Figure 3.12 <i>b</i> , (<i>a</i>) lines of constant Mach number; (<i>b</i>) variation of Mach number along the axis; (<i>c</i>) variation of Mach number along \overline{AB} | 87 |
| 3.16 | Result obtained on mesh in Figure 3.12 <i>b</i> , (<i>a</i>) lines of constant Mach number; (<i>b</i>) variation of Mach number along the axis; (<i>c</i>) variation of Mach number along \overline{AB} | 90 |
| 3.17 | Different levels of refinement on mesh in Figure 3.13 <i>b</i> | 93 |

List of Tables

| | |
|---------------------------------------|----|
| 3.1 Computational parameters. | 94 |
|---------------------------------------|----|

Introduction

High voltage transmission networks should be provided with safe and reliable systems in order to turn off the power rapidly in case of disorder. This switching function is usually performed by gas-blast circuit breakers or gas circuit breakers for short in this presentation. When a fault takes place they come into operation. As the normally contacted terminals of the circuit breaker separate, the current continues to flow in the form of an arc induced between the terminals until the current zero is reached. At this point the arc is much smaller than it was in the high-current phase and the current is momentarily interrupted, but then a recovery voltage pulses across the terminals of the circuit breaker. If the breaker has been designed properly, it will be able to resist this recovery voltage and power interruption will be successful. However, the recovery voltage may be so high that the arc can be reignited and this will result in the breaker failure to achieve interruption.

A protection against this breakage is taken by blowing a stream of gas from a high-pressure gas reservoir through the breaker nozzles toward the terminals. Removing energy from the arc, this process sets up suitable dielectric strength and withstands arc reignition.

The extinction and reignition characteristics of gas circuit breaker are strongly

affected by the gas flow which depends heavily on the geometry of the interrupter. Therefore a thorough understanding of the relationship among the arc, flow and geometry, as well as a deep insight of the associated phenomena will benefit the design of effective gas circuit breakers and improvement of their performances.

In the present study and as an initial stage leading to the more sophisticated investigation of the problem and a representation of a real step toward the computer-aided design of gas circuit breakers, we have simplified the physical reality under some hypotheses. We ignore chemical reactions, and do not take into account electric mechanisms which are governed by Maxwell's equations. We assume that the physical domain is two dimensional or axisymmetrical, the boundaries of computation are static instead of moving, the flow is neither viscous nor conductive and replace the working gas SF_6 with an ideal gas. Following these, the mathematical model dealing with such a fluid flow is based on the Euler equations.

The purpose of this undertaking is to establish a computational framework for the simulation. Chapter 1 describes briefly the Euler equations — a specific case of the general conservation law form and the coordinate transformation. Chapter 2 discusses the widely applied spatial discretizations and time iterations. Finally, Chapter 3 illustrates and examines the computational results implemented by using the numerical approaches in Chapter 2.

Chapter 1

Mathematical Model

The mathematical model used to describe the motion of real fluids is the Navier-Stokes equations. So far the enormous mathematical difficulties encountered when solving them have precluded any analytic form in which the convective terms interact in a general way with the friction terms. However, in this presentation we consider such flows where the effects of viscosity are negligible, namely, the motion of the flows is governed by the Euler equations, a simplified form of the Navier-Stokes equations.

1.1 Euler equations

Let us start with a system of the universal conservation law:

$$\vec{q}_t + \sum_{i=1}^d \vec{f}^i(\vec{q})_{x_i} = \vec{0}, \quad (1.1)$$

here $\vec{q}(\vec{x}, t)$ is an n column vector of unknowns, $\vec{f}^i(\vec{q})$ is a vector valued function of n components, $\vec{x} = (x_1, \dots, x_d)$ and $f = (f^1, \dots, f^d)$. Note that for $d = 3$, $\vec{x} = (x_1, x_2, x_3) = (x, y, z)$.

Rewrite (1.1) in the form:

$$\vec{q}_t + \sum_{i=1}^d A^i(\vec{q}) \vec{q}_{x_i} = \vec{0},$$

where $A^i(\vec{q}) = \vec{f}_q^i$ is the Jacobian matrix. The system (1.1) is called hyperbolic if the matrix

$$\sum_{i=1}^d \omega_i A^i(\vec{q})$$

has real eigenvalues and a complete set of eigenvectors for all real ω_i [1].

Denote the variables:

| | |
|----------------------|--|
| ρ | the density, |
| u, v and w | the velocity components in the x, y and z directions respectively, |
| m_u, m_v and m_w | the momentum components in the x, y and z directions respectively, |
| T | the temperature, |
| P | the pressure, |
| S | the entropy, |
| E | the energy, |
| R | the universal gas constant, |
| γ | the ratio of specific heats, |
| c | the sound speed — $(\gamma RT)^{1/2}$, |
| H | the total enthalpy. |

Here T, P, S and H follow the thermodynamic relations:

$$T = \frac{\gamma - 1}{R} \left(\frac{E}{\rho} - \frac{m_u^2 + m_v^2 + m_w^2}{2\rho^2} \right),$$

$$P = (\gamma - 1) \left(E - \frac{m_u^2 + m_v^2 + m_w^2}{2\rho} \right),$$

$$S = \ln \left(E - \frac{m_u^2 + m_v^2 + m_w^2}{2\rho} \right) - \gamma \ln \rho,$$

$$H = \frac{E + P}{\rho} = \frac{\gamma}{\gamma - 1} \frac{P}{\rho} + \frac{1}{2\rho^2} (m_u^2 + m_v^2 + m_w^2).$$

\vec{q} will always indicate the vector:

$$\vec{q} = \begin{pmatrix} \rho \\ m_u \\ m_v \\ m_w \\ E \end{pmatrix} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{pmatrix}.$$

Now the Euler equations are of the form:

$$\vec{q}_t + \vec{f}_x^1 + \vec{f}_y^2 + \vec{f}_z^3 = \vec{0}, \quad (1.2)$$

where

$$\vec{f}^1 = \begin{pmatrix} \rho u \\ \rho u^2 + P \\ \rho uv \\ \rho uw \\ (E + P)u \end{pmatrix}, \quad \vec{f}^2 = \begin{pmatrix} \rho v \\ \rho v^2 + P \\ \rho vw \\ (E + P)v \end{pmatrix}, \quad \vec{f}^3 = \begin{pmatrix} \rho w \\ \rho w^2 + P \\ (E + P)w \end{pmatrix}.$$

The system is hyperbolic. In the following chapter for convenience and without confusion, we may write $\vec{u}, \vec{f}, \vec{g}$ for $\vec{q}, \vec{f}^1, \vec{f}^2$ accordingly.

1.2 Transformation of the governing equations

For many applications one can express the Euler equations in different coordinates. To this end, a general transformation of the Euler equations from a Cartesian coordinate system (x, y, z) to an arbitrary coordinate system (ξ, η, ζ) can be cast through first consideration of the mapping:

$$\begin{aligned}\xi &= \xi(x, y, z), \\ \eta &= \eta(x, y, z), \\ \zeta &= \zeta(x, y, z).\end{aligned}$$

Using the chain rule leads to

$$\begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{pmatrix} = \begin{pmatrix} \xi_x & \eta_x & \zeta_x \\ \xi_y & \eta_y & \zeta_y \\ \xi_z & \eta_z & \zeta_z \end{pmatrix} \begin{pmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \\ \frac{\partial}{\partial \zeta} \end{pmatrix} = A \begin{pmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \\ \frac{\partial}{\partial \zeta} \end{pmatrix}. \quad (1.3)$$

On the other hand, x, y, z may also be represented as functions of ξ, η, ζ , i.e.,

$$\begin{aligned}x &= x(\xi, \eta, \zeta), \\ y &= y(\xi, \eta, \zeta), \\ z &= z(\xi, \eta, \zeta).\end{aligned}$$

Consequently,

$$\begin{pmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \\ \frac{\partial}{\partial \zeta} \end{pmatrix} = \begin{pmatrix} x_\xi & y_\xi & z_\xi \\ x_\eta & y_\eta & z_\eta \\ x_\zeta & y_\zeta & z_\zeta \end{pmatrix} \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{pmatrix} = B \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{pmatrix}.$$

If $\det A \neq 0$, or equivalently, $\det B \neq 0$, it follows

$$\begin{aligned} A &= B^{-1} \\ A^T &= (B^{-1})^T = (B^T)^{-1} \end{aligned}$$

Thus the relationships between the covariant and contravariant metric coefficients are

$$\begin{aligned} \xi_x &= J(y_\eta z_\zeta - y_\zeta z_\eta), \\ \xi_y &= -J(x_\eta z_\zeta - x_\zeta z_\eta), \\ \xi_z &= J(x_\eta z_\zeta - x_\zeta z_\eta), \\ \eta_x &= -J(y_\xi z_\zeta - y_\zeta z_\xi), \\ \eta_y &= J(x_\xi z_\zeta - x_\zeta z_\xi), \\ \eta_z &= -J(x_\xi y_\zeta - x_\zeta y_\xi), \\ \zeta_x &= J(y_\xi z_\eta - y_\eta z_\xi), \\ \zeta_y &= -J(x_\xi z_\eta - x_\eta z_\xi), \\ \zeta_z &= J(x_\xi y_\eta - x_\eta y_\xi), \end{aligned} \tag{1.4}$$

where

$$\begin{aligned} J &= 1/\det B \\ &= 1/[x_\xi(y_\eta z_\zeta - y_\zeta z_\eta) - x_\eta(y_\xi z_\zeta - y_\zeta z_\xi) + x_\zeta(y_\xi z_\eta - y_\eta z_\xi)]. \end{aligned}$$

Applying the generalized transformation (1.3) to the Euler equations written in the vector form (1.2) one gets

$$\begin{aligned} \vec{q}_t &+ \xi_x \bar{f}_\xi^1 + \eta_x \bar{f}_\eta^1 + \zeta_x \bar{f}_\zeta^1 \\ &+ \xi_y \bar{f}_\xi^2 + \eta_y \bar{f}_\eta^2 + \zeta_y \bar{f}_\zeta^2 \\ &+ \xi_z \bar{f}_\xi^3 + \eta_z \bar{f}_\eta^3 + \zeta_z \bar{f}_\zeta^3 = \vec{0}. \end{aligned}$$

It can be rearranged into, after dividing by J then adding and subtracting like terms, the following vector equation [2]:

$$\begin{aligned} \left(\frac{\vec{q}}{J}\right)_t &+ \left(\frac{\bar{f}^1 \xi_x + \bar{f}^2 \xi_y + \bar{f}^3 \xi_z}{J}\right)_\xi + \left(\frac{\bar{f}^1 \eta_x + \bar{f}^2 \eta_y + \bar{f}^3 \eta_z}{J}\right)_\eta + \left(\frac{\bar{f}^1 \zeta_x + \bar{f}^2 \zeta_y + \bar{f}^3 \zeta_z}{J}\right)_\zeta \\ &- \bar{f}^1 \left[\left(\frac{\xi_x}{J}\right)_\xi + \left(\frac{\eta_x}{J}\right)_\eta + \left(\frac{\zeta_x}{J}\right)_\zeta \right] \\ &- \bar{f}^2 \left[\left(\frac{\xi_y}{J}\right)_\xi + \left(\frac{\eta_y}{J}\right)_\eta + \left(\frac{\zeta_y}{J}\right)_\zeta \right] \\ &- \bar{f}^3 \left[\left(\frac{\xi_z}{J}\right)_\xi + \left(\frac{\eta_z}{J}\right)_\eta + \left(\frac{\zeta_z}{J}\right)_\zeta \right] = \vec{0}. \end{aligned}$$

The last three terms in square brackets can be dropped by using (1.4). Since the present study is aimed at plane flows, the final equation is in two dimensional strong conservation law form:

$$\frac{\partial \vec{q}_1}{\partial t} + \frac{\partial \bar{f}_1^1}{\partial \xi} + \frac{\partial \bar{f}_1^2}{\partial \eta} = \vec{0},$$

where

$$\vec{q}_1 = \vec{q}/J,$$

$$\bar{f}_1^1 = (\bar{f}^1 \xi_x + \bar{f}^2 \xi_y) / J,$$

$$\bar{f}_1^2 = (\bar{f}^1 \eta_x + \bar{f}^2 \eta_y) / J.$$

Chapter 2

Numerical Methods

The basic idea of numerical solution to the unsteady Euler equations is first to transform them into a set of coupled semi-discrete ordinary differential equations and then to integrate these ordinary differential equations numerically in time. The spatial discretization approach invoked in this study is the finite volume method based upon an integral form of the equations to be solved. The computational region is divided into elementary volumes over which the integration is carried out. Such a procedure allows one to deal with complicated geometries without resort to the equations written in curvilinear coordinates and also preserves the property of conservation [3].

Consider the two dimensional Euler equations:

$$\frac{\partial \vec{u}}{\partial t} + \frac{\partial \vec{f}}{\partial x} + \frac{\partial \vec{g}}{\partial y} = \vec{0}.$$

In the finite volume formulation, the steady state is described as the net flux into a finite volume cell being equal to zero. In smooth flow regions this is equivalent to the integration of the differential equations in conservation law form over an arbitrary cell Ω with boundary $\partial\Omega$ to give

$$\iint_{\Omega} [\vec{f}(\vec{u})_x + \vec{g}(\vec{u})_y] dx dy = \vec{0},$$

here \vec{f} and \vec{g} are the flux vectors in the x and y directions, respectively. Applying the divergence theorem we obtain the boundary integral

$$\oint_{\partial\Omega} [\vec{f}(\vec{u}) dy - \vec{g}(\vec{u}) dx] = \vec{0}.$$

For the discrete version, where \vec{U} , the approximation to \vec{u} , is adopted, the above boundary integral is replaced by, without loss of generality, the sum over the n cell sides of a polygon:

$$\sum_{i=1}^n [\vec{f}(\vec{U})|_i \Delta y_i - \vec{g}(\vec{U})|_i \Delta x_i] = \vec{0}, \quad (2.1)$$

where $(\Delta x_i, \Delta y_i)$ defines the orientation of side i , and $\vec{f}(\vec{U})|_i, \vec{g}(\vec{U})|_i$ are flux functions considered as averages along it. Thus $\vec{f}(\vec{U})|_i \Delta y_i - \vec{g}(\vec{U})|_i \Delta x_i$ is the normal flux traversing that side. Obviously, if one further divides Ω into a number of subcells, and performs integration over each of them and then sums up the corresponding boundary integrals with respect to Ω , one obtains the same result.

2.1 Spatial discretisations

The key to discretisation is how these averages are evaluated in terms of \vec{U} . The two most evident choices are to keep \vec{U} at the cell centres, and average across the cell sides [4]; or to keep \vec{U} at the cell vertices, and average along the cell sides [5]. These two concepts were first applied to the structured quadrilateral grid because of its high efficiency to the computation. However, unstructured

grids 1) provide a great deal more flexibility for fitting complicated configurations and furnish a natural setting for employing adaptive meshing; 2) facilitate the implementation of vectorization, and they can most easily be achieved by using triangular elements. Hence for simulation of flow in the complex interior of a gas circuit breaker, the unstructured triangular mesh is desirable. Recently, Morton and Paisley [6] studied these two different approaches on structured quadrilateral meshes, throwing some new light on the intrinsic attributes of each. Their paper is of great value to us in our work and a number of arguments in it are cited in this chapter. For completeness and varieties of application, we survey the above two discretisations respectively on both structured quadrilateral and unstructured triangular meshes below.

2.1.1 Cell centre scheme

First we construct the discrete steady spatial residual (2.1) for an arbitrary quadrilateral cell with reference to Figure 2.1a by averaging \vec{U} across cell boundaries. Dividing by the cell area V_C , denoting the residual by $\vec{R}_C(\vec{U})$, and writing \vec{f}_C for $\vec{f}(\vec{U})_C$, etc., we have

$$\begin{aligned}
\vec{R}_C(\vec{U}) &= \frac{1}{V_C} \left[\frac{1}{2}(\vec{f}_C + \vec{f}_E)(y_2 - y_1) - \frac{1}{2}(\vec{g}_C + \vec{g}_E)(x_2 - x_1) \right. \\
&\quad + \frac{1}{2}(\vec{f}_C + \vec{f}_N)(y_3 - y_2) - \frac{1}{2}(\vec{g}_C + \vec{g}_N)(x_3 - x_2) \\
&\quad + \frac{1}{2}(\vec{f}_C + \vec{f}_W)(y_4 - y_3) - \frac{1}{2}(\vec{g}_C + \vec{g}_W)(x_4 - x_3) \\
&\quad \left. + \frac{1}{2}(\vec{f}_C + \vec{f}_S)(y_1 - y_4) - \frac{1}{2}(\vec{g}_C + \vec{g}_S)(x_1 - x_4) \right] \tag{2.2} \\
&= \frac{1}{2V_C} [\vec{f}_E(y_2 - y_1) - \vec{f}_W(y_3 - y_4) - \vec{f}_N(y_2 - y_3) + \vec{f}_S(y_1 - y_4) \\
&\quad - \vec{g}_E(x_2 - x_1) + \vec{g}_W(x_3 - x_4) + \vec{g}_N(x_2 - x_3) - \vec{g}_S(x_1 - x_4)].
\end{aligned}$$

For a uniform rectangular mesh of cell dimensions Δx and Δy , it reduces to

$$R_{j,k}(\vec{U}) = \frac{\vec{f}_E - \vec{f}_W}{2\Delta x} + \frac{\vec{g}_N - \vec{g}_S}{2\Delta y},$$

the familiar central difference in both x and y directions, resulting in a second order accuracy.

Similarly, for an arbitrary triangular cell in Figure 2.1c, we get

$$\begin{aligned}
\vec{R}_C(\vec{U}) &= \frac{1}{V_C} \left[\frac{1}{2}(\vec{f}_C + \vec{f}_{C_1})(y_2 - y_1) - \frac{1}{2}(\vec{g}_C + \vec{g}_{C_1})(x_2 - x_1) \right. \\
&\quad \left. + \frac{1}{2}(\vec{f}_C + \vec{f}_{C_2})(y_3 - y_2) - \frac{1}{2}(\vec{g}_C + \vec{g}_{C_2})(x_3 - x_2) \right]
\end{aligned}$$

$$\begin{aligned}
& + \frac{1}{2}(\vec{f}_C + \vec{f}_{C_3})(y_1 - y_3) - \frac{1}{2}(\vec{g}_C + \vec{g}_{C_3})(x_1 - x_3) \Big] \\
= & \frac{1}{2V_C} [\vec{f}_{C_1}(y_2 - y_1) - \vec{g}_{C_1}(x_2 - x_1) \\
& + \vec{f}_{C_2}(y_3 - y_2) - \vec{g}_{C_2}(x_3 - x_2) \\
& + \vec{f}_{C_3}(y_1 - y_3) - \vec{g}_{C_3}(x_1 - x_3)].
\end{aligned}$$

2.1.2 Cell vertex scheme

Now keeping \vec{U} at the cell vertices, constructing the residual (2.1) and denoting it by \vec{R}_C , then averaging \vec{U} along the cell boundaries, we obtain, for an arbitrary quadrilateral cell in Figure 2.2a,

$$\begin{aligned}
\vec{R}_C(\vec{U}) &= \frac{1}{V_C} \left[\frac{1}{2}(\vec{f}_1 + \vec{f}_2)(y_2 - y_1) - \frac{1}{2}(\vec{g}_1 + \vec{g}_2)(x_2 - x_1) \right. \\
&\quad + \frac{1}{2}(\vec{f}_2 + \vec{f}_3)(y_3 - y_2) - \frac{1}{2}(\vec{g}_2 + \vec{g}_3)(x_3 - x_2) \\
&\quad + \frac{1}{2}(\vec{f}_3 + \vec{f}_4)(y_4 - y_3) - \frac{1}{2}(\vec{g}_3 + \vec{g}_4)(x_4 - x_3) \\
&\quad \left. + \frac{1}{2}(\vec{f}_4 + \vec{f}_1)(y_1 - y_4) - \frac{1}{2}(\vec{g}_4 + \vec{g}_1)(x_1 - x_4) \right] \\
&= \frac{1}{2V_C} [(\vec{f}_1 - \vec{f}_3)(y_2 - y_4) + (\vec{f}_2 - \vec{f}_4)(y_3 - y_1) \\
&\quad - (\vec{g}_1 - \vec{g}_3)(x_2 - x_4) - (\vec{g}_2 - \vec{g}_4)(x_3 - x_1)].
\end{aligned} \tag{2.3}$$

This corresponds to using the trapezoidal rule to approximate the integrals along each of the cell sides. On a uniform rectangular mesh it can be written as the box scheme average of differences

$$\vec{R}_{j+\frac{1}{2},k+\frac{1}{2}}(\vec{U}) = \frac{1}{2} \left[\frac{(\vec{f}_2 - \vec{f}_3) + (\vec{f}_1 - \vec{f}_4)}{\Delta x} + \frac{(\vec{g}_2 - \vec{g}_1) + (\vec{g}_3 - \vec{g}_4)}{\Delta y} \right],$$

possessing second order approximation also.

A triangular form can similarly be constructed as

$$\begin{aligned}
\vec{R}_C(\vec{U}) &= \frac{1}{V_C} \left[\frac{1}{2}(\vec{f}_1 + \vec{f}_2)(y_2 - y_1) - \frac{1}{2}(\vec{g}_1 + \vec{g}_2)(x_2 - x_1) \right. \\
&\quad + \frac{1}{2}(\vec{f}_2 + \vec{f}_3)(y_3 - y_2) - \frac{1}{2}(\vec{g}_2 + \vec{g}_3)(x_3 - x_2) \\
&\quad \left. + \frac{1}{2}(\vec{f}_3 + \vec{f}_1)(y_1 - y_3) - \frac{1}{2}(\vec{g}_3 + \vec{g}_1)(x_1 - x_3) \right]
\end{aligned}$$

$$\begin{aligned}
&= \frac{1}{2V_C} [\vec{f}_1(y_2 - y_3) - \vec{g}_1(x_2 - x_3) \\
&\quad + \vec{f}_2(y_3 - y_1) - \vec{g}_2(x_3 - x_1) \\
&\quad + \vec{f}_3(y_1 - y_2) - \vec{g}_3(x_1 - x_2)].
\end{aligned}$$

However in practice, the discretisation procedure is carried out node by node [7]. The flow variables are stored at the vertices of the triangles and the control volume for a particular node i is taken to be the union of all triangles surrounding this node, as shown in Figure 2.2c, thus

$$\begin{aligned}
\vec{R}_C(\vec{U}) &= \frac{1}{2V_C} [\vec{f}_1(y_2 - y_n) - \vec{g}_1(x_2 - x_n) \\
&\quad + \vec{f}_2(y_3 - y_1) - \vec{g}_2(x_3 - x_1) \\
&\quad + \dots \\
&\quad + \vec{f}_{n-1}(y_n - y_{n-2}) - \vec{g}_{n-1}(x_n - x_{n-2}) \\
&\quad + \vec{f}_n(y_1 - y_{n-1}) - \vec{g}_n(x_1 - x_{n-1})].
\end{aligned}$$

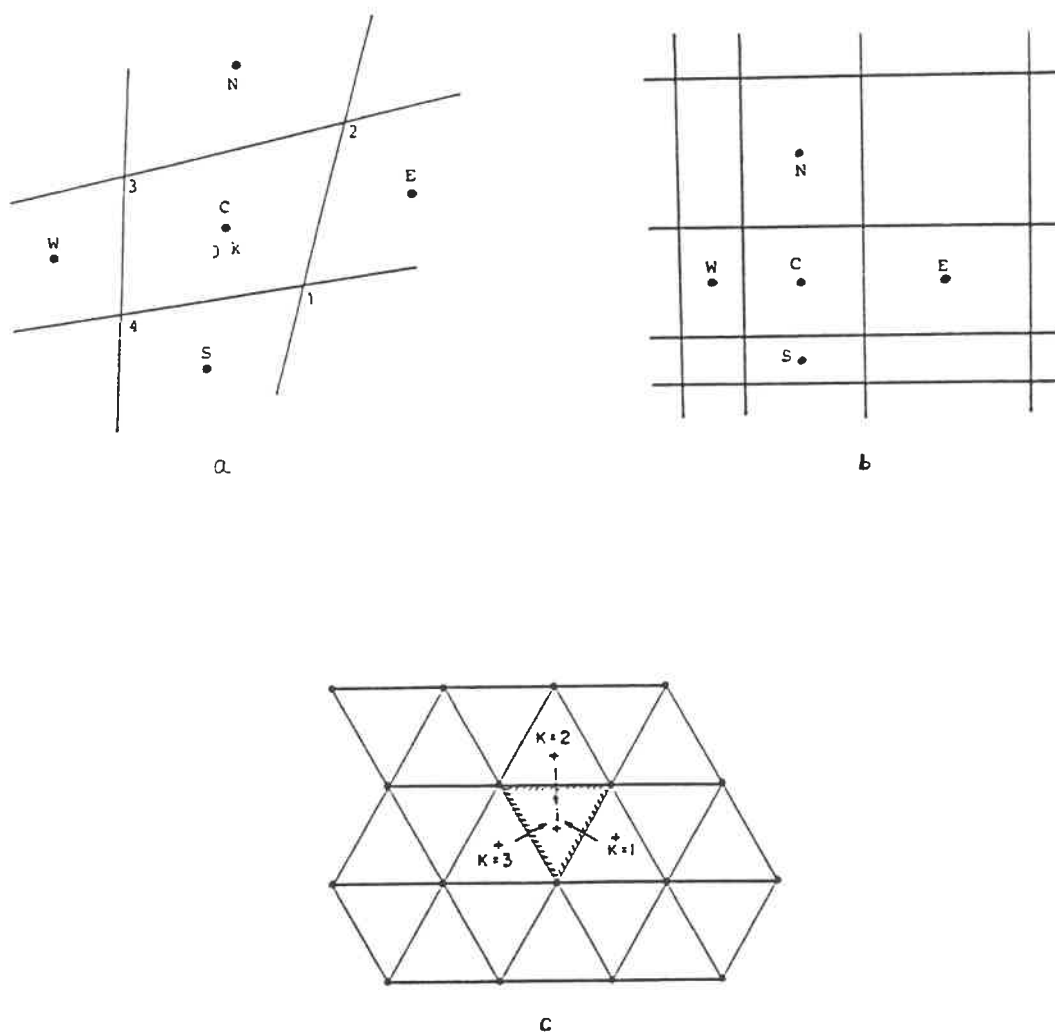


Figure 2.1: Geometry for flow variables at cell centres: (a) general quadrilateral mesh; (b) nonuniform rectangular mesh; (c) triangular mesh.

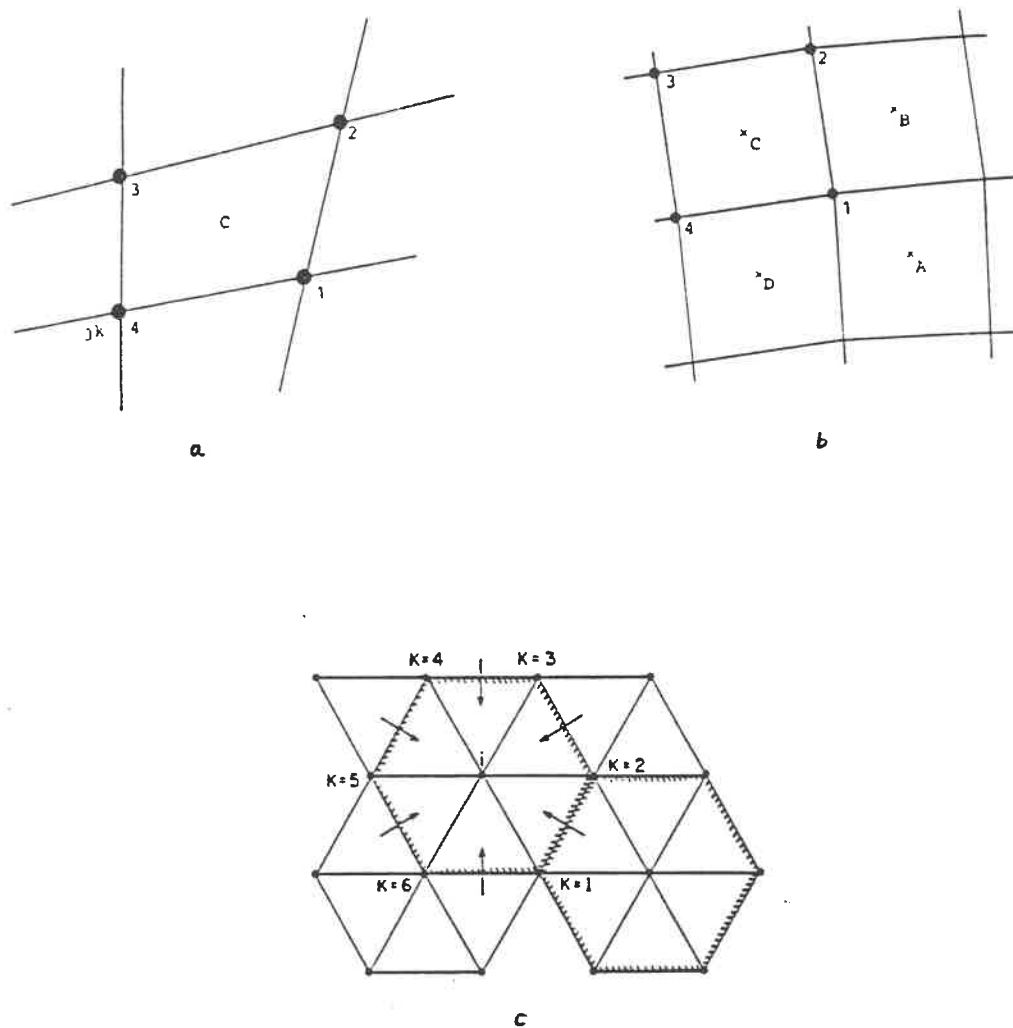


Figure 2.2: Geometry for flow variables at cell vertices: (a) the vertices around cell C ; (b) the cells used to update vertex 1; (c) triangular mesh.

This equation indicates that the cell vertex scheme is independent of the cell interior division. Therefore it relies on meshing less than the cell center scheme. By letting $n = 4$, we get the above quadrilateral formula (2.3).

2.1.3 Some properties of the two schemes

The objective, in calculation of an approximate steady solution by means of some iterative process, is to make the spatial residuals given in 2.1.1 or 2.1.2 close to zero. A universal analysis on the principal differences in the character of the approximation obtained by setting the residuals in 2.1.1 and 2.1.2 to zero — in particular, the accuracy on nonuniform meshes and the tendency to exhibit spurious solution modes for a general grid is not liable to conduct. However, for a special case of the structured quadrilateral mesh, Morton and Paisley [6] compared these two schemes in connexion with the above properties. They suggested that the cell vertex scheme possesses a greater resilience to distortions in the mesh than does the cell center scheme. They showed that the basic flaw with the cell center scheme as in (2.2) on an uneven mesh is the evaluation of the flux on a cell boundary as a simple average of the values in neighboring cells. On a nonuniform mesh this average is not centered at that boundary and errors are immediately introduced. Hence it is clear that the cell center approach as it stands must be adapted if it is expected to match the accuracy of the cell vertex method on a given nonuniform mesh. They also argued that another distinct advantage of the cell vertex scheme over the cell center scheme is its smaller number of spurious solution modes.

An improvement of anti-distortion performance of the cell center scheme can be conducted from the identity

$$D_2 = \begin{cases} \frac{2(n+1)}{2n+1}D_1 - \frac{1}{2n+1}, & n = 0, 1; \\ \frac{4}{n+2}D_1 + \frac{n-2}{n+2}, & n \geq 2, \end{cases}$$

where $D_1 = (a + 2b + c)/b$, $D_2 = (a + nb + c)/(n + 2)b$, and $b \neq 0$. One can regard D_1 as a general form of a central difference with three points, and D_2 with arbitrary point(s). Suppose $D_1 = 1 + O(h^k)$, $k \geq 0$; fix the other parameters and let n get sufficiently large such that quantity $4/(n + 2)$ is equal to or less than h , we may perceive that $|D_2 - 1|$ has a higher order with respect to h than $|D_1 - 1|$. This implies that for a center difference represented by the leading terms in the truncation error such as

$$\vec{R}_C(\vec{U}) = \frac{\Delta x_E + n\Delta x_C + \Delta x_W}{(n+2)\Delta x_C} \vec{f}_x|_{C^*} + \frac{\Delta y_E + n\Delta y_C + \Delta y_W}{(n+2)\Delta y_C} \vec{g}_y|_{C^*} + \dots,$$

where, as shown in Figure 2.1b, C^* denotes the center of cell C , $\Delta x_C, \Delta y_C$ are the dimensions of cell C , $\Delta x_E, \Delta y_E$ are the dimensions of the cell to the right, et cetera. On condition that $\vec{f}_x + \vec{g}_y = \vec{0}$, its accuracy of approximation will enhance as n , the weight of the central term, increases. On the other hand, when $n > 2$ the difference cannot be expressed without using extra points. Hence the calculation is complicated, and the additional spurious modes may also be brought because generally speaking, the more the points a difference scheme uses, the more the spurious modes it exhibits.

Nevertheless, the following example shows an abnormal feature about points and spurious modes. Consider a uniform scalar difference equation with five points in x and y directions respectively,

$$\begin{aligned}
& \frac{U_{j+2,k} - U_{j-2,k}}{2\Delta x} + \underbrace{\frac{U_{j+1,k} - U_{j-1,k}}{2\Delta x}}_{2nd} \\
& + \frac{U_{j,k+2} - U_{j,k-2}}{2\Delta y} + \underbrace{\frac{U_{j,k+1} - U_{j,k-1}}{2\Delta y}}_{4th} = 0.
\end{aligned}$$

Looking for modes of the form $U_{j,k} = \hat{U}\mu^j\nu^k$, we have

$$\begin{aligned}
& \frac{1}{2\Delta x}[(\mu^4 - 1)\nu^2 + (\mu^3 - \mu)\nu^2] \\
& + \frac{1}{2\Delta y}[(\nu^4 - 1)\mu^2 + (\nu^3 - \nu)\mu^2]\hat{U}\mu^{j-2}\nu^{k-2} \\
& = \left[\frac{1}{2\Delta x}(\mu^2 - 1)(\mu^2 + \mu + 1)\nu^2 \right. \\
& \left. + \frac{1}{2\Delta y}(\nu^2 - 1)(\nu^2 + \nu + 1)\mu^2 \right] \hat{U}\mu^{j-2}\nu^{k-2} = 0.
\end{aligned}$$

Its nonzero real solution set is $\{(\mu, \nu) \mid \mu, \nu = \pm 1\}$, the same as that to a three-point difference consisting of the second and fourth terms in this equation.

2.2 Time stepping iterations

By far the most widely used form of iteration or smoother, is that based on a time stepping approximation to the unsteady Euler equations. It is here that the cell center scheme has its main attraction, since the residuals in 2.1.1 is correctly centered for updating the value \vec{U}_C . However, once it has been decided how contiguous residuals can be combined to update the values of unknowns at

a vertex [5], the cell vertex scheme again possesses definite advantages. It seems, habitually, that the cell center scheme adopts a Runge-Kutta procedure [8] while the cell vertex scheme a Lax-Wendroff process [5].

2.2.1 Lax-Wendroff algorithms

For convenience we discuss this iteration on quadrilateral meshes. The parallel conclusions for triangular meshes can be obtained by similar reasoning. The Lax-Wendroff method is based on a Taylor series expansion in time

$$\delta \bar{u}^{n+1} = \bar{u}^{n+1} - \bar{u}^n = \Delta t \bar{u}_t^n + \frac{1}{2!} \Delta t^2 \bar{u}_{tt}^n + O(\bar{1} \Delta t^3), \quad (2.4)$$

in which the time derivatives are replaced by spatial derivatives via the differential equation. This can be taken as a two-step method, though it is advantageous for steady problems to consider it as one step. Thus for the unsteady system of differential equations

$$\bar{u}_t + \bar{f}(\bar{u})_x + \bar{g}(\bar{u})_y = \bar{0},$$

the second time derivative in (2.4) is replaced by spatial derivatives as

$$\begin{aligned} \bar{u}_{tt} &= \frac{\partial}{\partial t}(\bar{u}_t) = -\frac{\partial}{\partial t}[\bar{f}(\bar{u})_x + \bar{g}(\bar{u})_y] \\ &= -\frac{\partial}{\partial x}(A\bar{u}_t) - \frac{\partial}{\partial y}(B\bar{u}_t) \\ &= \frac{\partial}{\partial x}\{A[\bar{f}(\bar{u})_x + \bar{g}(\bar{u})_y]\} + \frac{\partial}{\partial y}\{B[\bar{f}(\bar{u})_x + \bar{g}(\bar{u})_y]\}, \end{aligned}$$

where A, B are the Jacobian matrices $\partial \vec{f} / \partial \vec{u}, \partial \vec{g} / \partial \vec{u}$ respectively. Substituting this expression into (2.4) we get the time increment in the solution:

$$\delta \vec{U}^{n+1} = -\Delta t (\vec{f}_x + \vec{g}_y)^n + \frac{1}{2} \Delta t^2 \left\{ \frac{\partial}{\partial x} [A(\vec{f}_x + \vec{g}_y)] + \frac{\partial}{\partial y} [B(\vec{f}_x + \vec{g}_y)] \right\}^n. \quad (2.5)$$

The spatial discretisation of the first term in this equation, say, with reference to Figure 2.2b, at point 1, is given by

$$-\Delta t \frac{V_A \vec{R}_A + V_B \vec{R}_B + V_C \vec{R}_C + V_D \vec{R}_D}{V_A + V_B + V_C + V_D},$$

an area weighted average of residuals (2.3) in the adjoining cells.

On a general mesh this average corresponds to the boundary integral around the perimeter of the group of four cells, ensuring conservation is maintained during the transient phase and values at the central vertex are eliminated. The effect is brought to light even in one dimension [6].

The second term of (2.5) is in the form of a divergence and again can be cast as a boundary integral, this time around the inner quadrilateral $ABCD$. The flux functions at the cell centers are now given by the product of the respective Jacobian matrix (where all the values of the conserved variables needed for the matrix entries are evaluated at the cell centers, as averages of the vertex values) with the residual vector for that cell. This gives

$$\begin{aligned} \frac{\Delta t^2}{4V_1} \{ & [(A\vec{R})_A - (A\vec{R})_C](y_B - y_D) + [(A\vec{R})_B - (A\vec{R})_D](y_C - y_A) \\ & - [(B\vec{R})_A - (B\vec{R})_C](x_B - x_D) - [(B\vec{R})_B - (B\vec{R})_D](x_C - x_A) \}, \end{aligned}$$

where V_1 is the area of the quadrilateral formed by the centers of the four cells — A, B, C and D . If the four edges of the group of the four cells are especially all straight line segments, we have

$$V_1 = \frac{1}{4}(V_A + V_B + V_C + V_D).$$

Thus the total change at a point is represented by a weighted average of contiguous residuals (the weights depend on the geometry and the Jacobians). Namely,

$$\delta \vec{U}^{n+1} = -\frac{1}{4} \Delta t (D_A \vec{R}_A + D_B \vec{R}_B + D_C \vec{R}_C + D_D \vec{R}_D), \quad (2.6)$$

where D_Ω are the “distribution” matrices [6]:

$$D_A = \frac{4V_A}{\sum V_\Omega} I - \frac{\Delta t}{V_1} (y_B - y_D) A_A + \frac{\Delta t}{V_1} (x_B - x_D) B_A,$$

$$D_B = \frac{4V_B}{\sum V_\Omega} I - \frac{\Delta t}{V_1} (y_C - y_A) A_B + \frac{\Delta t}{V_1} (x_C - x_A) B_B,$$

$$D_C = \frac{4V_C}{\sum V_\Omega} I - \frac{\Delta t}{V_1} (y_B - y_D) A_C + \frac{\Delta t}{V_1} (x_B - x_D) B_C,$$

$$D_D = \frac{4V_D}{\sum V_\Omega} I - \frac{\Delta t}{V_1} (y_C - y_A) A_D + \frac{\Delta t}{V_1} (x_C - x_A) B_D.$$

An important feature of this one-step formulation is that at convergence, the weighted average of cell residuals in (2.6) tends to zero as $\delta \vec{U}^{n+1} \rightarrow 0$. This is also crucial to individual residual vanishing under appropriate boundary conditions.

The evaluation of the Jacobians necessary for the one-step form can be circumvented by casting the iteration in a two-step form as is often done for unsteady problems, e.g., to predict values at cell centers (denoted here by $*$) and then obtain corrected values at the vertices [6]:

$$\vec{U}_C^* = \frac{1}{4}(\vec{U}_1^n + \vec{U}_2^n + \vec{U}_3^n + \vec{U}_4^n) - \frac{1}{2}\Delta t \vec{R}_C,$$

$$\vec{U}_1^{n+1} = \vec{U}_1^n - \Delta t \vec{R}_1^*$$

where \vec{R}_1^* is evaluated from $\vec{U}_A^*, \vec{U}_B^*, \vec{U}_C^*$ and \vec{U}_D^* . For convergence we must have $\vec{R}_j^* \rightarrow \vec{0} \quad \forall j$ and this does not necessarily imply that $\vec{R}_\Omega^n \rightarrow \vec{0} \quad \forall \Omega$. Thus if the two-step form is used the residual constructed from average values in neighbouring cells is set to zero, rather than what is actually required — the residual for each individual cell.

For the limits on Δt to ensure convergence, and the optimum local choice, consider the two dimensional scalar wave equation

$$u_t + a u_x + b u_y = 0,$$

with a and b constant. On a uniform rectangular mesh, we find [6]

$$\begin{aligned} \delta U^{n+1} = & -\frac{1}{4}\Delta t[(1 - \nu_x + \nu_y)R_A + (1 - \nu_x - \nu_y)R_B \\ & + (1 + \nu_x - \nu_y)R_C + (1 + \nu_x + \nu_y)R_D] \end{aligned} \quad (2.7)$$

with

$$R_C = \frac{a}{2} \left(\frac{U_2 - U_3}{\Delta x} + \frac{U_1 - U_4}{\Delta x} \right) + \frac{b}{2} \left(\frac{U_2 - U_1}{\Delta y} + \frac{U_3 - U_4}{\Delta y} \right), \text{ etc.}$$

and $\nu_x = a(\Delta t/\Delta x)$, $\nu_y = b(\Delta t/\Delta y)$. Performing the usual Fourier analysis and assuming the solution being of the form $U_{j,k}^n = \lambda^n e^{i(j\xi + k\eta)}$, where $\xi = k_x \Delta x$, $\eta = k_y \Delta y$ and k_x, k_y are the wave numbers in the x and y directions respectively, we obtain the amplification factor:

$$\begin{aligned} \lambda &= 1 - i \left(\nu_x \sin \xi \cos^2 \frac{\eta}{2} + \nu_y \cos^2 \frac{\xi}{2} \sin \eta \right) \\ &\quad - 2\nu_x^2 \sin^2 \frac{\xi}{2} \cos^2 \frac{\eta}{2} - \nu_x \nu_y \sin \xi \sin \eta \\ &\quad - 2\nu_y^2 \cos^2 \frac{\xi}{2} \sin^2 \frac{\eta}{2}. \end{aligned}$$

Rearranging, using the half-angle formulae and putting $m = \nu_x \sin \frac{\xi}{2} \cos \frac{\eta}{2} + \nu_y \cos \frac{\xi}{2} \sin \frac{\eta}{2}$ leads to

$$\lambda = 1 - 2im \cos \frac{\xi}{2} \cos \frac{\eta}{2} - 2m^2,$$

with

$$|\lambda|^2 = 1 - 4m^2 \left(1 - m^2 - \cos^2 \frac{\xi}{2} \cos^2 \frac{\eta}{2} \right).$$

For convergence we need $|\lambda|^2 \leq 1$ for all ξ, η in the range $[0, 2\pi]$. The necessary and sufficient condition for is

$$0 \leq m^2 \leq 1 - c_x^2 c_y^2,$$

here we write c_x for $\cos \frac{\xi}{2}$, c_y for $\cos \frac{\eta}{2}$ and later s_x for $\sin \frac{\xi}{2}$, s_y for $\sin \frac{\eta}{2}$.

It can be deduced [6] that for the upper bound on m^2 , which corresponds to the stability bound for the unsteady case, it is necessary and sufficient that

$$\nu_x^2 + \nu_y^2 \leq 1. \quad (2.8)$$

Sufficiency was well proved by using the Cauchy-Schwarz inequality (see [6]), while necessity can be verified briefly by *reductio ad absurdum* underneath.

Suppose that

$$\nu_x^2 + \nu_y^2 > 1,$$

this inequality can be further written as

$$\nu_x^2 + \nu_y^2 = 1 + d = 1 + \frac{d}{2} + \frac{d}{2}, \quad d > 0,$$

where we may specify that

$$\nu_y^2 = \frac{d}{2}, \quad \nu_x^2 = 1 + \frac{d}{2} > 1.$$

Now let $\xi = \pi$, $\eta = 0$, then it follows

$$s_x = c_y = 1, \quad c_x = s_y = 0.$$

Consequently,

$$m^2 = (\nu_x s_x c_y + \nu_y c_x s_y)^2 = \nu_x^2 > 1.$$

On the other hand, according to the hypothesis of necessity, the inequality

$$m^2 \leq 1 - c_x^2 c_y^2 = 1$$

holds. The contradiction in the expressions of m^2 implies the impropriety of the preliminary assumption: $\nu_x^2 + \nu_y^2 > 1$. Thus the validity of the above proposition for necessity is testified.

The stability or convergence limit (2.8) gives the largest Δt that can be used; for the differential equation, which has no dissipation so that the steady state is reached only by driving initial perturbations out of the domain, taking the largest time step possible is the optimal strategy [6]. Since the Lax-Wendroff algorithm (2.7) contains some dissipation which depends on Δt , a smaller value may be preferable. For example, in one dimension, $|\lambda| = 1 - 4\nu^2(1 - \nu^2)s^4$, where $\nu = a\Delta t/\Delta x$, $s = \sin \frac{\xi}{2}$, so that one obtains maximum damping for all error modes by taking $\nu^2 = \frac{1}{2}$. In practice the time step can be differed locally to take account of local mesh lengths and the maximum permissible value is unclear when mesh is irregular. Therefore the fastest convergence rate is obtained by employing the local time step [6]. An example of this is given by a geometry independent formula using two one-dimension limits [5]

$$\Delta t_C < \min \left\{ \frac{V_C}{|u\Delta y^l - v\Delta x^l| + c\Delta l}, \frac{V_C}{|u\Delta y^m - v\Delta x^m| + c\Delta m} \right\},$$

where $\Delta x^l, \Delta y^l, \Delta l, \Delta x^m, \Delta y^m, \Delta m$ depend on the cell geometry:

$$\Delta x^l = \frac{1}{2}(x_2 + x_3 - x_1 - x_4); \quad \Delta x^m = \frac{1}{2}(x_1 + x_2 - x_3 - x_4);$$

$$\Delta y^l = \frac{1}{2}(y_2 + y_3 - y_1 - y_4); \quad \Delta y^m = \frac{1}{2}(y_1 + y_2 - y_3 - y_4);$$

$$\Delta l = [(\Delta x^l)^2 + (\Delta y^l)^2]^{1/2}; \quad \Delta m = [(\Delta x^m)^2 + (\Delta y^m)^2]^{1/2},$$

and $c = \sqrt{\gamma P/\rho}$ is the speed of sound. This condition also guarantees the distribution matrices in (2.6) to be nonsingular [6].

2.2.2 Runge-Kutta algorithms

The Runge-Kutta methods or the multistage methods were initially aimed at solving the ordinary differential equations. However, once the spatial derivatives in the Euler equations are appropriately treated according to some discretisation approach, say, one of the two schemes in 2.1, the methods can also be applied to calculate the partial differential equations such as the unsteady Euler equations. Thus we integrate the unsteady Euler equations over a quadrilateral or triangular cell i and construct the residuals using either the cell vertex scheme or the cell center scheme to obtain

$$\frac{d\vec{U}_i}{dt} + \vec{R}_i = \vec{0}, \quad i = 1, 2, \dots, n. \quad (2.9)$$

The scheme as it stands is not resistant to high frequency oscillations between the odd and even mesh points. Dissipative terms must be put in to suppress spurious oscillations of this type, and to prevent the appearance of unsightly wiggles in the neighborhood of shock waves. With the addition of dissipative terms \vec{D}_i , the system of semidiscrete equations (2.9) takes the form [4]

$$\frac{d\vec{U}_i}{dt} + \vec{R}_i - \frac{\vec{D}_i}{V_i} = \vec{0}, \quad (2.10)$$

here V_i stands for the volume of cell i , and \vec{D} consists of a blend of the second and fourth differences. The second differences are introduced primarily to smooth out the oscillations in regions of severe pressure gradients associated with the shock capturing. While the fourth differences are used for background dissipation to hinder the odd and even mesh point decoupling. For further details see [4,7,8,9].

Equation (2.10) can be rewritten as [4]

$$\frac{d\vec{U}_i}{dt} + \frac{1}{V_i}(\vec{Q}_i - \vec{D}_i) = \vec{0},$$

where $\frac{\vec{Q}_i}{V_i} = \vec{R}_i$, $\vec{Q}(\vec{U})$ and $\vec{D}(\vec{U})$ can be regarded as the convective and the dissipative operators respectively. Dropping the subscripts i , the standard five-stage hybrid scheme, to advance a time step Δt , can be given by [7]

$$\begin{aligned} \vec{U}^{(0)} &= \vec{U}^n, \\ \vec{U}^{(1)} &= \vec{U}^{(0)} - \frac{\alpha_1 \Delta t}{V} [\vec{Q}(\vec{U}^{(0)}) - \vec{D}(\vec{U}^{(0)})], \\ \vec{U}^{(2)} &= \vec{U}^{(0)} - \frac{\alpha_2 \Delta t}{V} [\vec{Q}(\vec{U}^{(1)}) - \vec{D}(\vec{U}^{(1)})], \\ &\vdots \\ \vec{U}^{(5)} &= \vec{U}^{(0)} - \frac{\alpha_5 \Delta t}{V} [\vec{Q}(\vec{U}^{(4)}) - \vec{D}(\vec{U}^{(1)})], \\ \vec{U}^{n+1} &= \vec{U}^{(5)}, \end{aligned} \tag{2.11}$$

where \vec{U}^n and \vec{U}^{n+1} are the values at the beginning and the end of the n th time step, α_i are the coefficients endowed with

$$\alpha_1 = 1/4, \quad \alpha_2 = 1/6, \quad \alpha_3 = 3/8, \quad \alpha_4 = 1/2, \quad \alpha_5 = 1.$$

The convective operator $\vec{Q}(\vec{U})$ is evaluated at each stage in one time step, while the dissipative operator $\vec{D}(\vec{U})$ is calculated only twice, each in the first two stages. This scheme is specifically designed to damp high frequency error modes and is adaptable to multigrid methods.

The admissible time step of this explicit hyperbolic equation solver is confined by the Courant-Friedrichs-Lewy condition. A method of extending the maximum time step limit can be derived from strengthening the support of the scheme by averaging the residuals and their neighbors, e.g., in [7] the speed of iteration is traded off with the complexity of computation as follows

$$\vec{R}_i = \vec{R}_i + \epsilon \nabla^2 [(1 - \theta) \vec{R}_i + \theta \vec{R}_i],$$

where $0 \leq \theta \leq 1$, ϵ is a small weighting parameter and ∇^2 is the undivided Laplacian operator. For the unstructured triangular mesh, the Laplacian is evaluated by

$$\nabla^2 (*)_i = \sum_{k=1}^n (*)_k - n(*)_i,$$

and n is either taken 3 as for the cell center scheme or as the total number of edges ringing node i for the cell vertex scheme.

When $\theta = 0$, the average is performed explicitly, and the support is only increased slightly. Nevertheless, when $\theta = 1$, the implicitly calculated average makes the support infinite and the CFL condition no longer imposes any restriction on the time step, provided the value of ϵ is taken adequately [7].

For $\theta \neq 0$, the implicit system of equations needs to be solved iteratively. Because ϵ is usually small (say, 0.5), the coefficient matrix related to this procedure becomes strongly diagonally dominant. According to [7], in practice two simple Jacobi iterations are normally sufficient to produce an approximate solution to the smoothed residuals such that the convergence of the time stepping can be significantly accelerated. Moreover, using the maximum permissible time step at each point can also expedite the convergence to the steady state.

2.3 Other computational points

This section is devoted to present briefly the boundary conditions, the form of dissipation and the criterion of mesh refinement, all of these are involved in the computations of next chapter.

2.3.1 The boundary conditions

The physical boundary conditions for computations later are imposed, according to [10], such as:

Inlet plane (subsonic)

- | | |
|------------------------------------|------------------------|
| 1) total pressure $P_0 (= P_{in},$ | $= 1$ (uniform value), |
| the pressure at entrance) | |
| 2) total temperature T_0 | $= 1$ (uniform value), |
| 3) flow angle at entrance | |

Outlet plane (subsonic)

- | |
|--------------------------------------|
| 1) static pressure $P_s (= P_{out},$ |
| the pressure at exit) |

Outlet plane (supersonic)

none

2.3.2 The form of dissipation

Additional dissipative terms are indispensable to deter the appearance of oscillations near shock waves and to damp the high frequency uncoupled modes that may be triggered on both structured quadrilateral mesh and unstructured triangular mesh. For the latter, which, together with the Runge-Kutta time stepping process, was adopted in this undertaking, dissipation can be of the form [7]

$$\vec{D}(\vec{U}_i) = \sum_{k=1}^n d_{ik},$$

where

$$d_{ik} = \epsilon_1(\vec{U}_i - \vec{U}_k) - \epsilon_2 \nabla^2(\vec{U}_i - \vec{U}_k),$$

and

$$\nabla^2 \vec{U}_i = \sum_{k=1}^n \vec{U}_k - n\vec{U}_i.$$

Here ϵ_2 is a small coefficient determined empirically, ϵ_1 an adaptive coefficient which is made proportional to an undivided Laplacian with respect to the pressure, and n has been defined in 2.2.2.

Thus with the dissipation added and referring to the Runge-Kutta iteration (2.11), equation (2.9) can formally be written as

$$\frac{d\vec{U}_i}{dt} + \vec{R}_i^+ = \vec{0}, \quad i = 1, 2, \dots, n,$$

where

$$\vec{R}_i^+ = \vec{R}_i - \frac{\vec{D}_i}{V_i}.$$

2.3.3 The criterion of mesh refinement

The mesh refinement of the code utilized in this investigation is based on the following algorithm:

Giving a positive real number C_r as the bound of refinement, Supposing that n_1, n_2, \dots, n_m are the neighbors to node n (here m is a function of node number, which takes positive integer values), and evaluating

$$D_l = \max\{d_{n_1}, d_{n_2}, \dots, d_{n_m}\},$$

then the mesh is refined on condition that D_l is greater than or equal to C_r , or not. Here

$$d_j \propto \sum_{i=1}^m \frac{|P_j - P_{j_i}|}{|P_j + P_{j_i}|}, \quad j = n_1, n_2, \dots, n_m,$$

and P_k is the pressure at node k . Since in real problems it is necessary to have $D_l \geq 0$, therefore $C_r = 0$ implies that refinement is applied everywhere while $C_r = \infty$ (or a positive number being large enough) means that no refinement occurs.

Chapter 3

Computational Results

In this study, a code based on the cell vertex scheme on unstructured triangular grid using the Runge-Kutta multistage time stepping procedure for the Euler equations has been developed. Since the equations vary with different types of flow regimes, the simulations of three distinct inviscid compressible duct flows were carried out to test and validate the code. In order to present the results step by step, it is helpful to start with the easiest one.

3.1 Straight channel flow

In this example an isentropic parallel flow was examined. The flow was subsonic and uniform everywhere along the channel. For simulation of such a flow, the boundary condition — a ratio of the outlet pressure to the inlet pressure was accordingly imposed as $P_{out}/P_{in} = 0.8$. In this chapter the inlet pressure is identified with the total pressure. The flow angle at the entrance was set to zero and it will not be changed at the exit because of the flow pattern. Beginning the marching procedure with an initial Mach number of zero, the solution converged after

10,000 iterations. Since pressure and Mach number can be linked by the formula:

$$P_{out} = P_{in} \left(1 + \frac{\gamma - 1}{2} M^2 \right)^{\gamma/(\gamma-1)} \quad (3.1)$$

for this problem, more reasonable initial values can be placed. For example, a choice of Mach number of 0.45, reduces iterations by about two thirds.

Substituting $P_{out}/P_{in} = 0.8$ into (3.1) one obtains the theoretical Mach number as

$$M = 0.5737.$$

The computed maximum and minimal Mach numbers are

$$\begin{aligned} (M_{\Omega})_{max} &= 0.5737; \\ (M_{\Omega})_{min} &= 0.5736, \end{aligned}$$

thus the error bound is given by

$$E = \max_{\Omega} |M - M_{\Omega}| \leq M - (M_{\Omega})_{min} = 10^{-4},$$

where Ω is the computational domain. The mesh used for the calculation is shown in Figure 3.1a. The computed data of pressure and Mach number are plotted in Figure 3.1b where the two straight line segments are the corresponding analytic solutions.

3.2 Nozzle flow

Here a converging-diverging nozzle with small throat radius of curvature was utilized for testing two dimensional duct flow. The ratio of the outlet pressure to

the inlet pressure was set to $P_{out}/P_{in} = 0.2$. The flow angle at the entrance was assigned equal to zero and the angle at the exit was left to be updated from the computation. There is no exact solution for this case, however experimental data are available [11].

Figure 3.2 illustrates the preliminary mesh and the numerical solution on it. Figures 3.3–3.7 show the results in the same sequence as Figure 3.2: first the domain and mesh are illustrated and then the comparison of numerical result with experimental data is revealed by Mach number vs. wall position of the nozzle. In Figures 3.8*b* and *c* two such comparisons are displayed.

Each of the solutions in Figures 3.3*b*–3.6*b* was obtained through refinement and continuing iteration based on its predecessor. It is seen that on coarse meshes numerical solutions are unable to converge to the relevant experimental results in the throat part. However, the differences lessen with local mesh refinement of this area. The reason is that the coarse grid cannot capture the sharp changes of flow variables caused by the geometry there. The mesh in Figure 3.2*a* was obtained through refining the initial mesh with C_r — the bound of refinement mentioned in 2.3.3, of 3; the meshes in Figures 3.3*a*–3.6*a* were obtained through refining the meshes in Figures 3.2*a*–3.5*a* respectively, with C_r of 6, 6, 4, 4 for each.

The solutions shown in Figures 3.7 and 3.4 have near the same amount of grid points but their distributions differ. The grid for the first solution was more refined in the throat area and thus the solution has a better approximation there while the second solution used an improved grid in the downstream region and consequently possesses a better configuration here.

Figure 3.8*a* indicates an improved mesh regularity, *b* and *c* display solutions obtained on this mesh. Figure 3.8*c* is a follow-up result of *b*. The converged

solution of Figure 3.8c required fewer refinements and iterations than the solution of Figure 3.5b, because the mesh in Figure 3.8a has a better resemblance to the nozzle wall as well as a more regular form of the internal cells. According to the discussion in 2.1.3, it is well appreciated that the regularity of cell shape reduces the discretisation errors and the fitness of boundary curve results in the boundary conditions being imposed properly. Therefore the solution in Figure 3.8c converged swiftly.

Although Mach number of zero is a universally practicable value for starting iteration, during the calculation of this problem it is found that since the practical flow at the outlet is supersonic, taking an appropriate positive value instead of zero as the initial Mach number, say, 0.5, can effectively speed up the convergence of iteration and cut down the time of computation.

Damping is necessary even though there was no shock presented in this test. Calculations without damping or with insufficient damping induced oscillations or even diverged within a number of time marching steps, after initially appearing to converge. Since the two added damping factors *sig2* and *sig4*, which will be explained in 3.3.2, vary asymptotically from large to small, in a sense of absolute value and the same as follows, during the iteration, we only list the values of *sig2* and *sig4* for all the solutions in Figures 3.2b–3.6b at the final converged stages such as: $sig2 = 0.001$ and $sig4 = 0.07$. At the beginning of calculation, the two factors, particularly *sig2*, should be endowed with larger values to suppress the oscillations inherently embedded in the numerical process. The influence of damping will be further illustrated in the following section.

3.3 Complex duct flow

Now we go back to the problem proposed at the beginning — simulation of the flow phenomena in the interior of a gas circuit breaker. To date, there is neither theoretical calculation nor experimental implementation available due to intricate flow behavior and prohibitive experimental costs. In the previous two simpler cases, satisfactory results (compared with either analytic solution or experimental data) were obtained by using the cell vertex code on unstructured triangular meshes. However, to use this code with adaptive meshing to calculate duct flows in complex geometries like the interior of a gas circuit breaker is a novel application of such a computational procedure.

In this simulation, the original static flow was first accelerated to supersonic condition and then shocked down by the electrode. For the test circumstances the ratio of pressure was imposed such that $P_{out}/P_{in} = 0.2$. The flow angle at the entrance was taken as 180° and that at the exit was left to be determined computationally. The initial Mach number was 0. The configuration of the circuit breaker interior and the relevant computational domain and zoning pattern are illustrated in Figure 3.9*a, b* and *c*, respectively.

Figure 3.10*a* shows the introductory grid. Figure 3.10*c* shows two line segments along which the Mach number changes will be displayed. The one that can be clearly viewed is denoted by \overline{AB} . It is parallel to the axis of symmetry — the bottom contour line of the computational domain. The other is the projection of \overline{AB} on the axis of symmetry.

3.3.1 The effects of grid and iteration

The solutions shown from Figures 3.11 to 3.13 were obtained using finer meshes. Figure 3.10*b* displays a preliminary approximation on the coarse mesh, it seems that the shock is not captured and the lines of iso-Mach are prone to vary with iterations in regions around the throat and the flange of the electrode. Figure 3.10*d* and *e* illustrate the corresponding variations of Mach number along the two line segments — \overline{AB} and the axis of symmetry, with oscillations embedded. In the succeeding discussion, figures describing Mach number vs. \overline{AB} and the axis of symmetry will always assume such that alphabetical order corresponds to lower-upper position.

In order to get better results, grid refining must be invoked. The predetermined values of refinement were settled such as 0.08, 0.5, 0.7 for obtaining meshes in Figures 3.11*b*–3.13*b* respectively. The refinement was advanced level by level, e.g., the mesh in Figure 3.11*b* was obtained through refining the mesh in Figure 3.10*a* using the value of 0.08, and so forth. According to the criterion of refinement in 2.3.3, the areas where the values of D_i are greater than or equal to these predestined values (denoted by C_r) will be refined and those where the values of D_i are less than the predestined refinement values will not. Figure 3.11*a* shows the area to be refined, which is surrounded by four iso-lines in terms of the values derived from the operation of the second difference of the pressure. It is seen that in this area the flow variables have notable changes. Figure 3.11*b* depicts the refined grid. The computed result in Figure 3.11*c* denotes significant improvements compared with Figure 3.10*b*, the regions where flow variables alter sharply in Figure 3.10*b* shrink to narrower ones, and two shocks are detected: one is in the vicinity of the electrode flange, the other is under the electrode flange. The curves in Figure

3.11*d* and *e* look obviously more reasonable than their counterparts in Figures 3.10*d* and *e*, despite the behavior of oscillation on the axis.

Figure 3.12*b* is obtained via a small regional refinement of the mesh in Figure 3.11*b*. The areas for refinement are depicted in Figure 3.12*a*. At this time an interesting phenomenon appears: the bigger shock on the right hand side of the electrode together with the small one under the electrode forms a distinct λ -shock as shown in Figure 3.12*c*. It is also perceivable that the lines of constant Mach number tend to be steady except those in and near the shock regions in comparison with Figure 3.11*c*. Figure 3.12*e* gives a satisfactory result though *d* reveals an appreciable crest situated upstream the bigger shock. This explains that to capture shocks precisely one needs a grid with further refinement in the relevant regions.

The places of the last refinement are shown in Figure 3.13*a*. Figure 3.13*b* is the refined mesh, *c* is the solution conducted on *b*, where two stagnation points exist. One is on the axis facing the middle of the first fork from the entrance, the other is on the front of the electrode flange. There are no substantial differences between the results in Figures 3.13*c,e* and their corresponding predecessors in Figures 3.12*c,e*. At present the crest in Figure 3.12*d* vanishes in the updated solution as illustrated by Figure 3.13*d*. Bearing in mind the fine grids and on which the converged solutions were obtained, the concordance between these two solutions is easily understood. The solution in Figure 3.14 was obtained on the same grid as that in Figure 3.13*b*, nevertheless with 2,000 iterations fewer than the solution in Figure 3.13. These two results as shown by Figures 3.13*c,d,e* and Figures 3.14*a,b,c* accordingly, agree so well that one can nearly be regarded as a copy of another. This, together with the above comparison of solutions on the

meshes of different refinements (i.e., Figures 3.12 and 3.13), persuades that the most updated solution depicted in Figure 3.13 is convergent and robust. Therefore it was considered as a proper solution. Figure 3.17 illustrates an enlarged regional view of mesh with different refinement levels in Figure 3.13b.

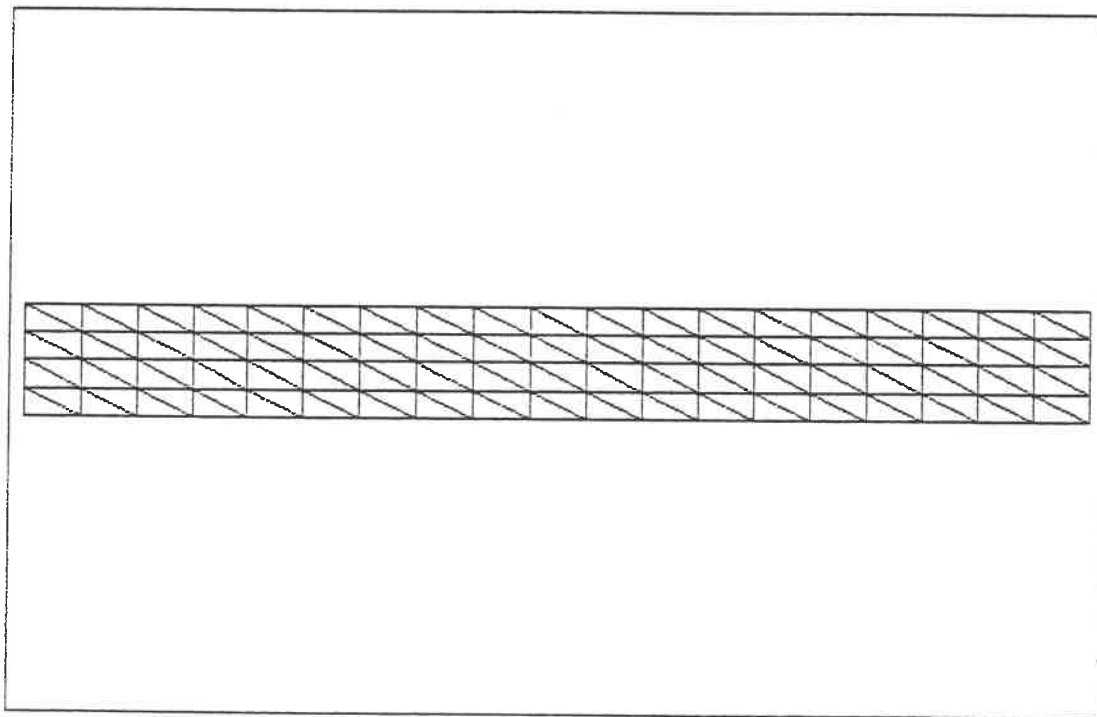
3.3.2 The effects of the additional dissipation

The computational parameters are summarised in Table 3.1, where the preliminary mesh corresponds to level 1, $sig2$ and $sig4$ are the second and fourth differences respectively, whose functions were described in the previous chapter. The table shows that the fourth difference term behaves as a constant whereas the second difference exhibits unmonotonic variations on diverse meshes. In other words, the second difference plays a decisive or dominant role in the calculation. The values of dissipation were chosen such that shocks can be captured and oscillations can be damped as well.

Figure 3.15 illustrates a solution as a result of insufficient damping ($sig2 = sig4 = 0.05$). Oscillations can be seen through small circles around the shocks in Figure 3.15a and crests located upstream the bigger shock in both b and c . By contrast, Figure 3.16 expresses one with excessive dissipation ($sig2 = 3, sig4 = 0.07$). Shocks were suppressed, and smooth but “unnatural” curves are subject to see in Figures 3.16a–c. Since Figures 3.12 and 3.15, 3.16 share the same mesh, these two instances suggest that dissipation is important to shock capturing (or in other words, shock positions are critically and also sensitively dependent on the dissipation added), even with considerably fine grid it still needs to be treated carefully.

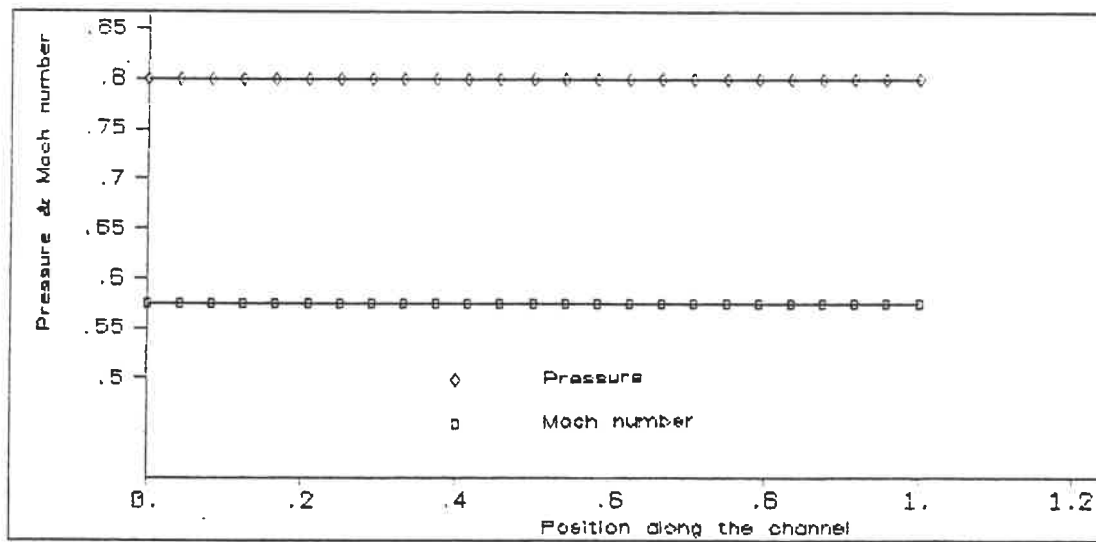
3.3.3 Summary

A numerical simulation of the intricate flow in a circuit breaker has been attempted. The results were obtained in such a way that the calculations were enforced to be convergent on meshes of different refinement level. Namely, at the beginning a convergent numerical solution was found on an elementary mesh which may be referred to as level n . Then the mesh was refined to level $n + 1$ and a new convergent solution was obtained on it. Thus level by level, a series of convergent results were achieved on meshes of different refinement level. This procedure was terminated when the difference between the two neighboring solutions was within an error limit. In other words, the solutions no longer varied significantly with mesh refinement and time iteration. Therefore the results in the sequence of calculation gradually converged to the final stage which reflects best the real flow behavior. Such a procedure has been repeated for several cases with various flow conditions and the results were found to be quite robust and mesh-independent.



a

Figure 3.1: (*a*) computational domain and mesh with 152 triangles; (*b*) comparison of numerical solution with analytic one in terms of pressure and Mach number vs. position along the channel.



b

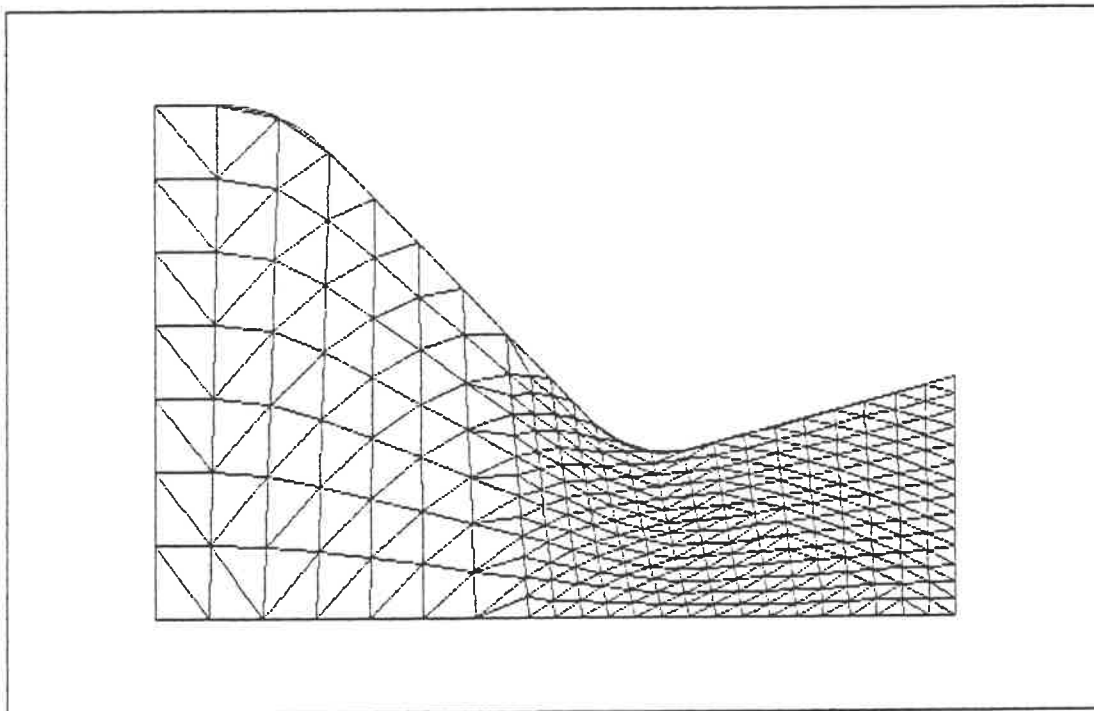
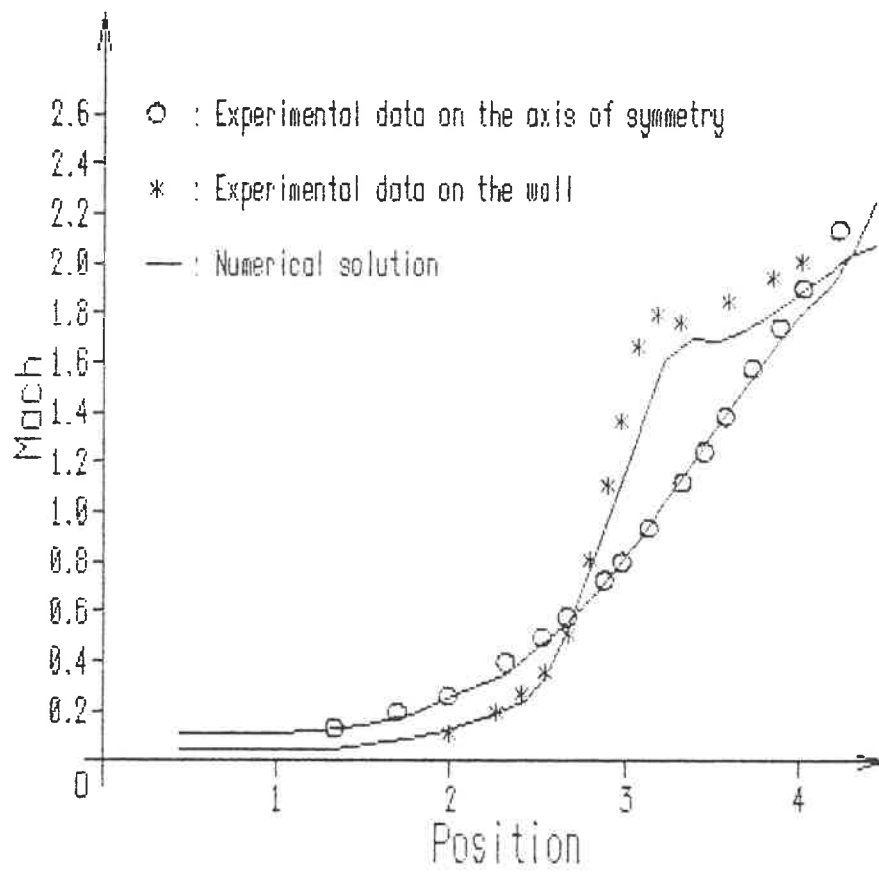
*a*

Figure 3.2: (a) domain and mesh with 550 triangles; (b) comparison of numerical result with experimental data in terms of Mach number vs. wall position.



b

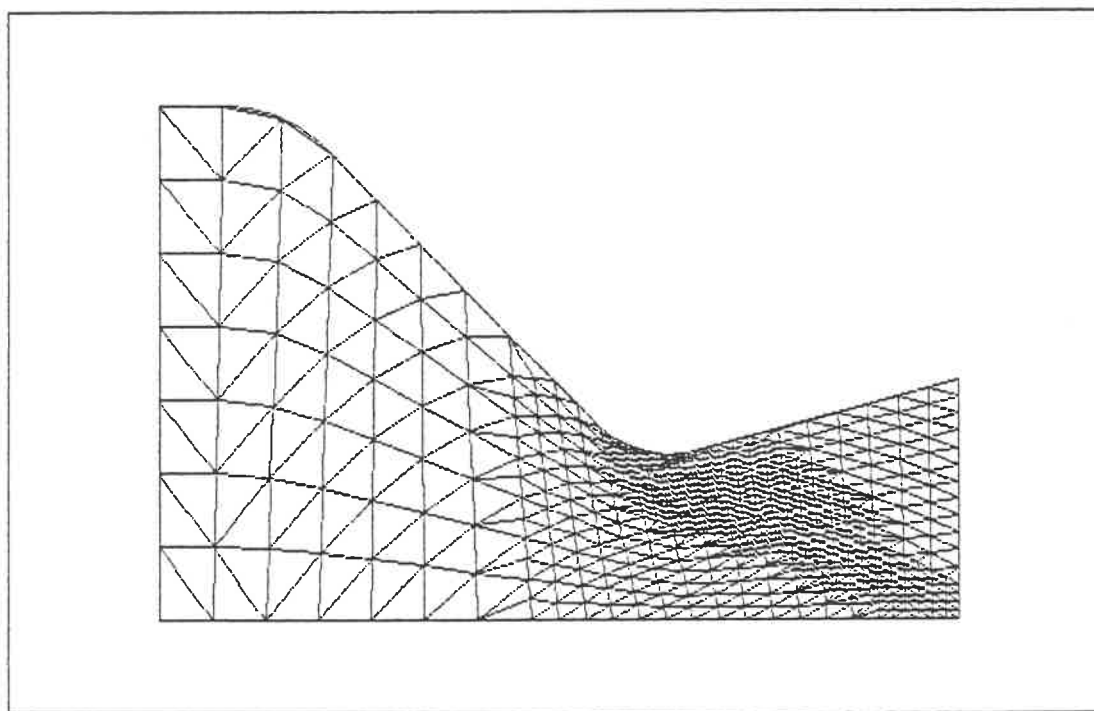
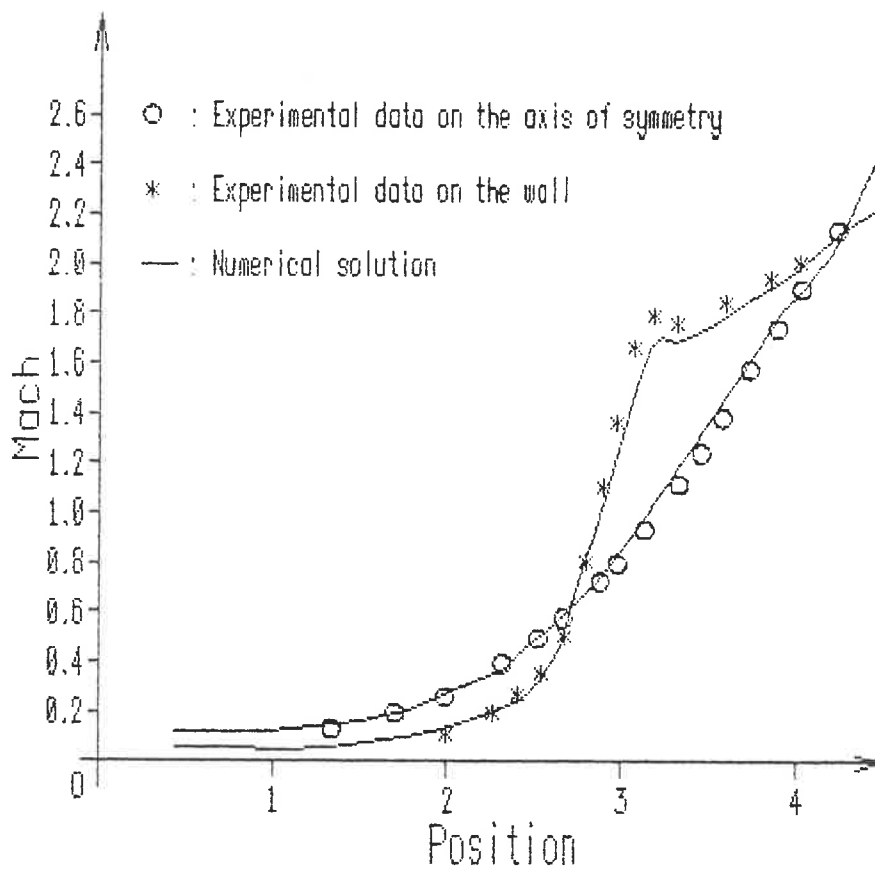
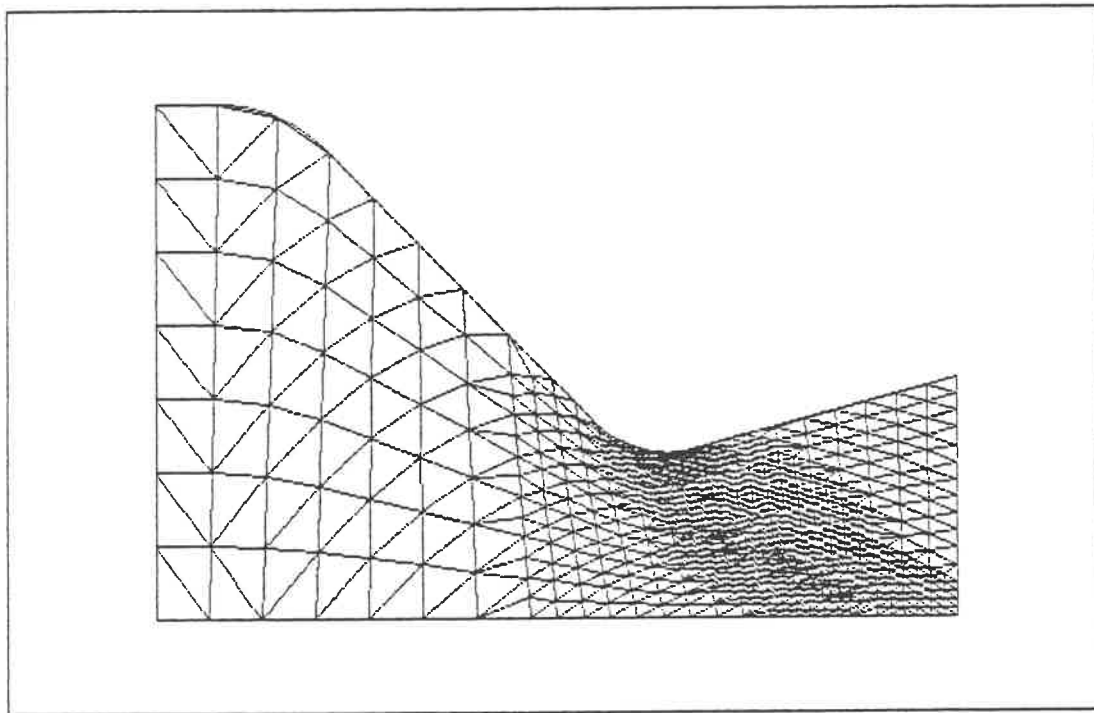
*a*

Figure 3.3: (a) domain and mesh with 1032 triangles; (b) comparison of numerical result with experimental data in terms of Mach number vs. wall position.

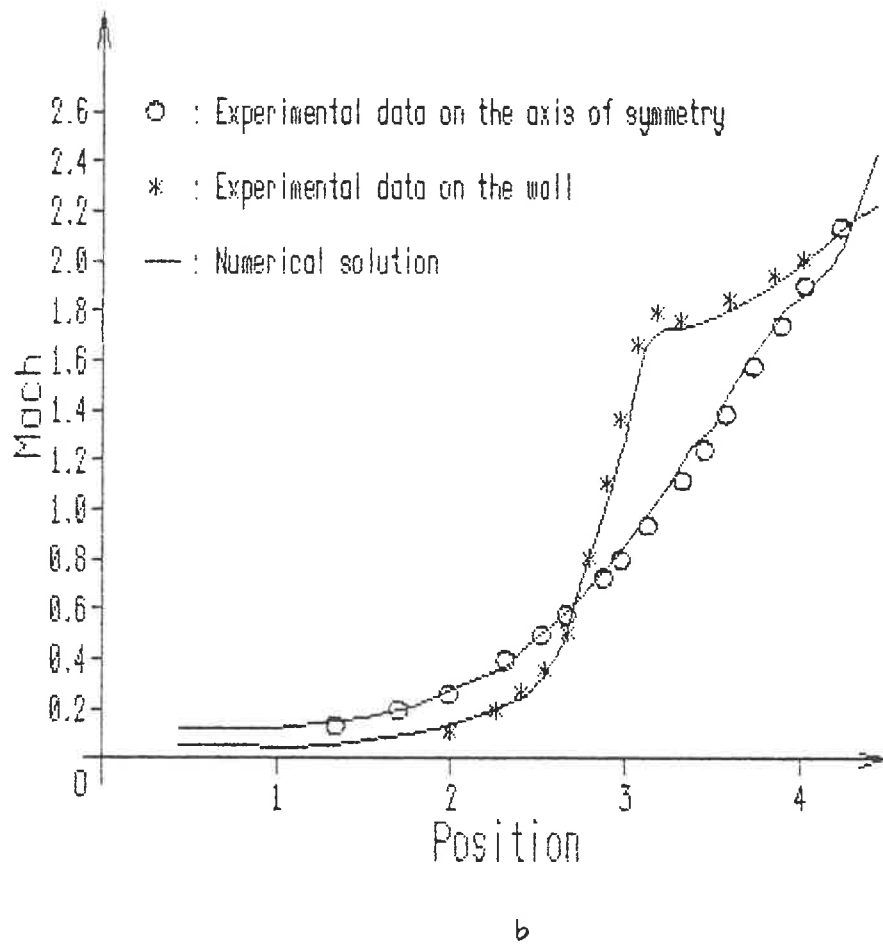


b



a

Figure 3.4: (a) domain and mesh with 1433 triangles; (b) comparison of numerical result with experimental data in terms of Mach number vs. wall position.



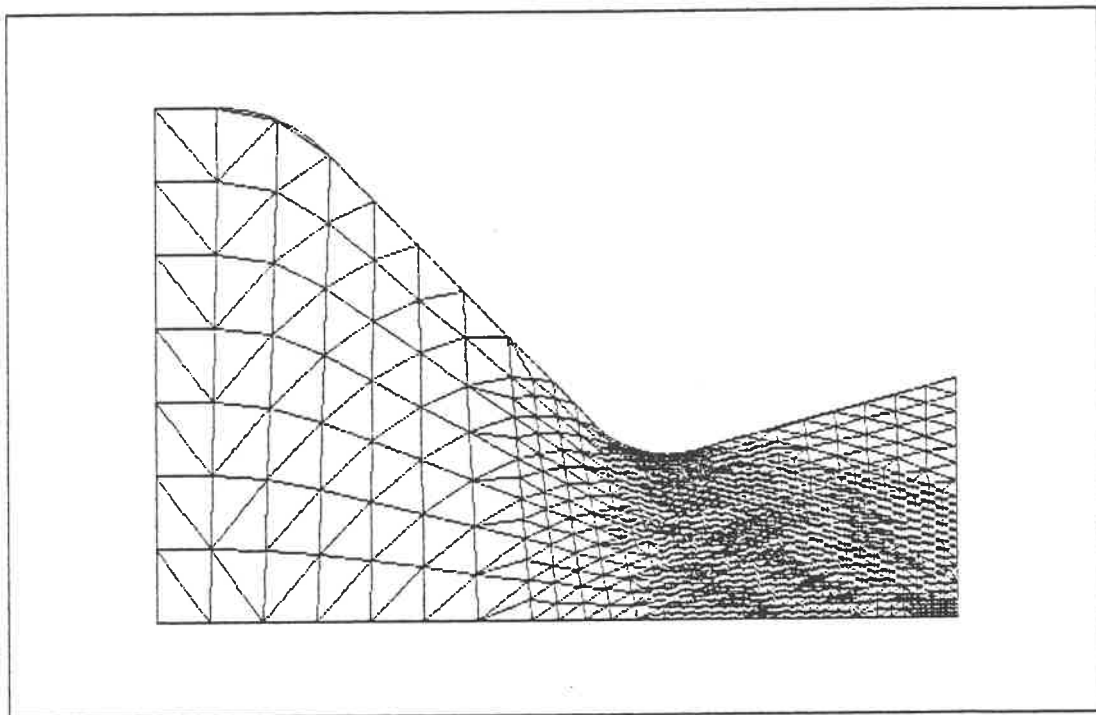
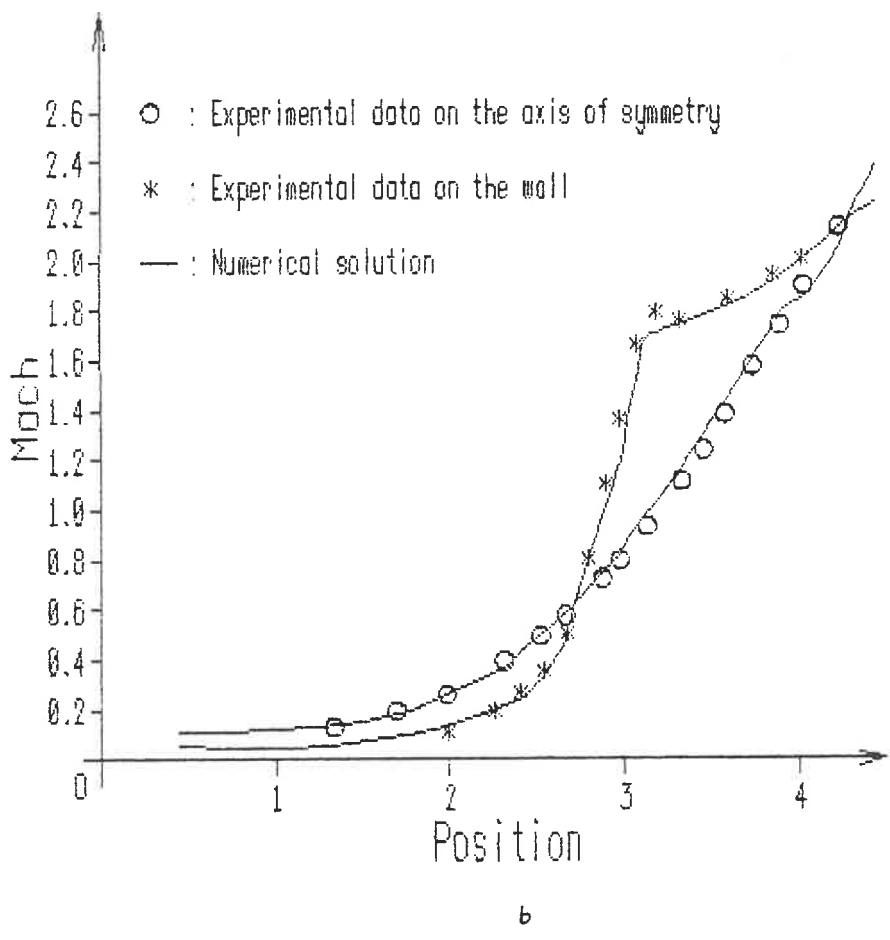
*a*

Figure 3.5: (a) domain and mesh with 2795 triangles; (b) comparison of numerical result with experimental data in terms of Mach number vs. wall position.



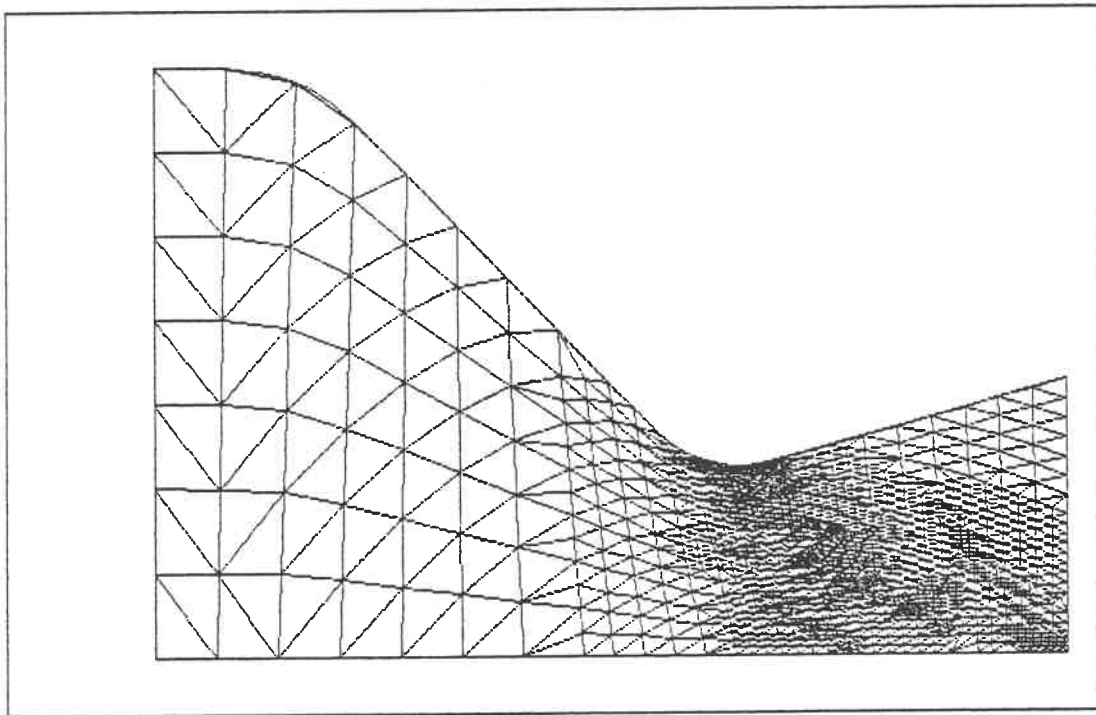
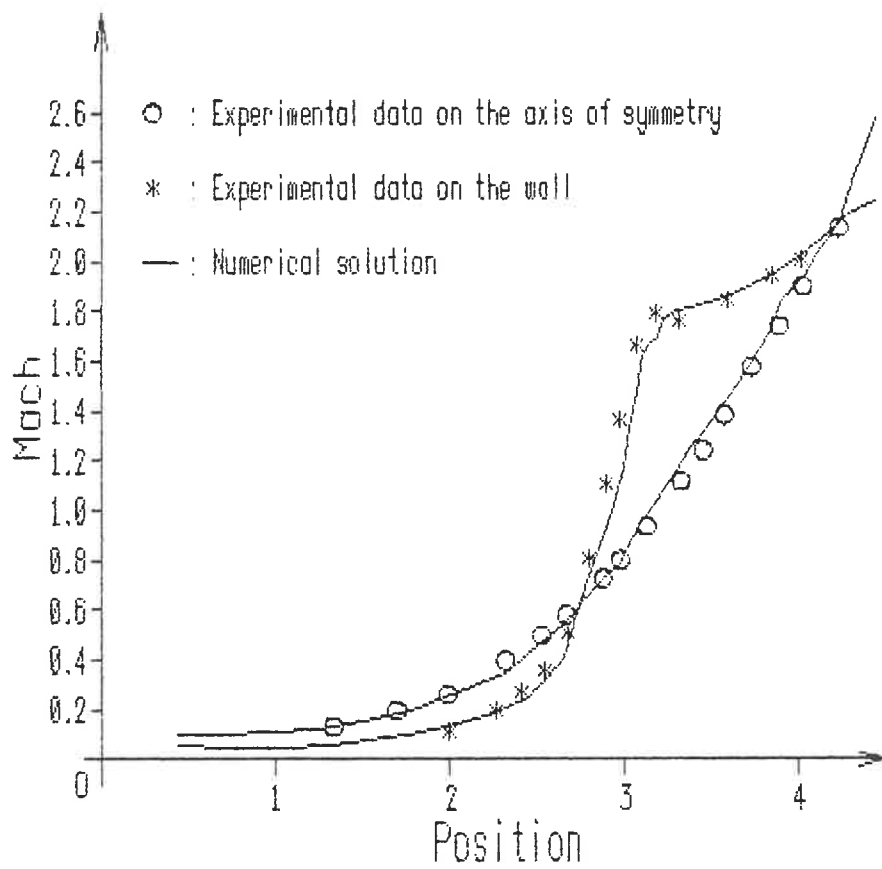
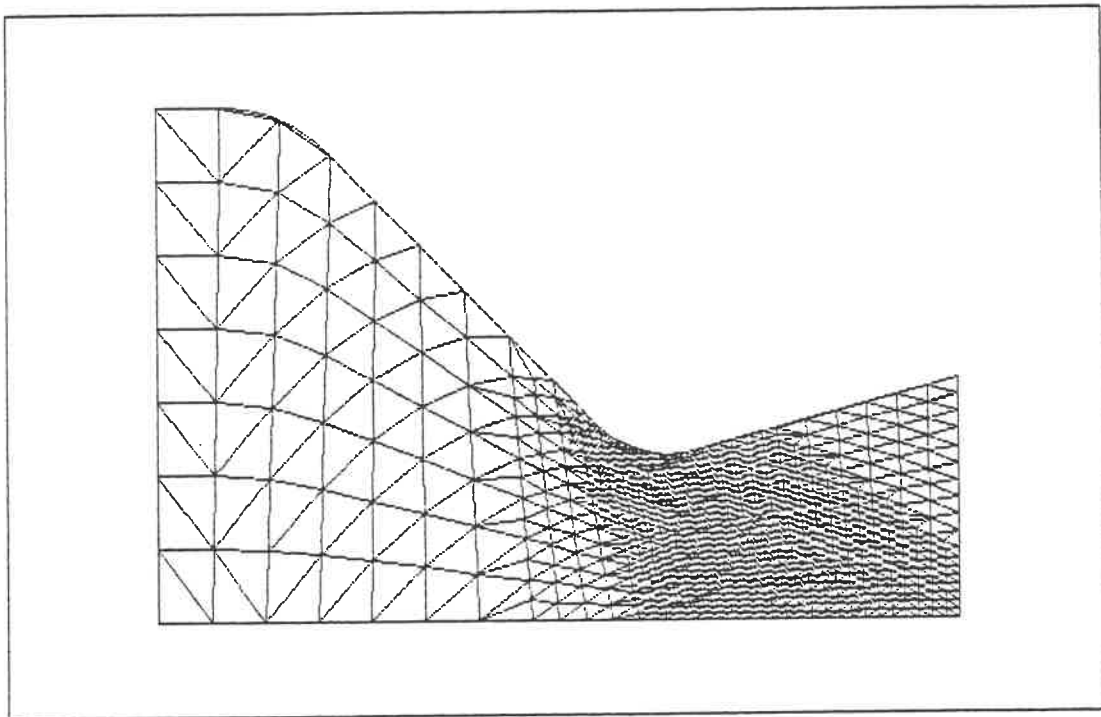
*a*

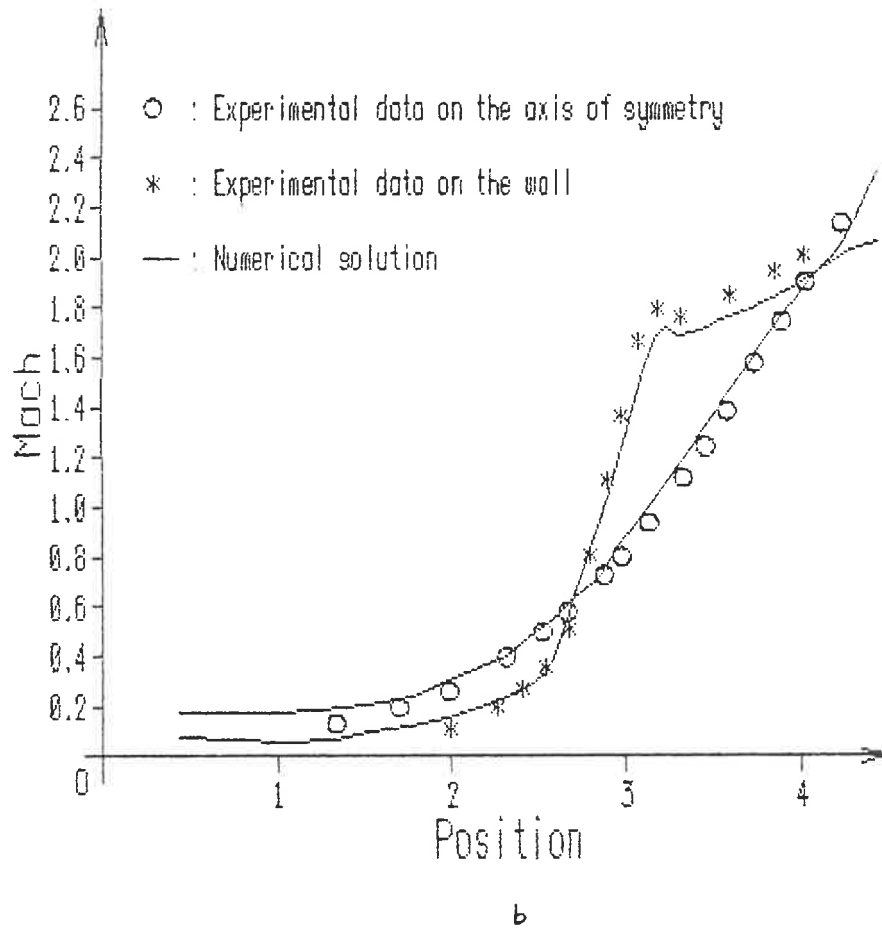
Figure 3.6: (a) domain and mesh with 3297 triangles; (b) comparison of numerical result with experimental data in terms of Mach number vs. wall position.

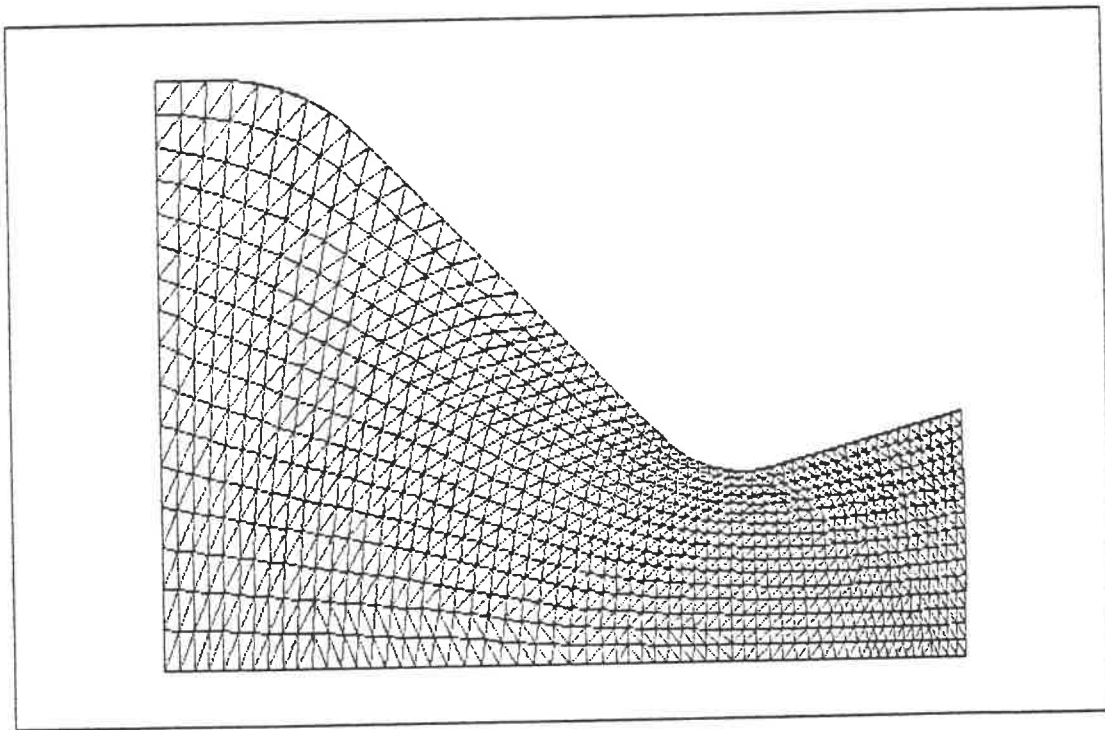




a

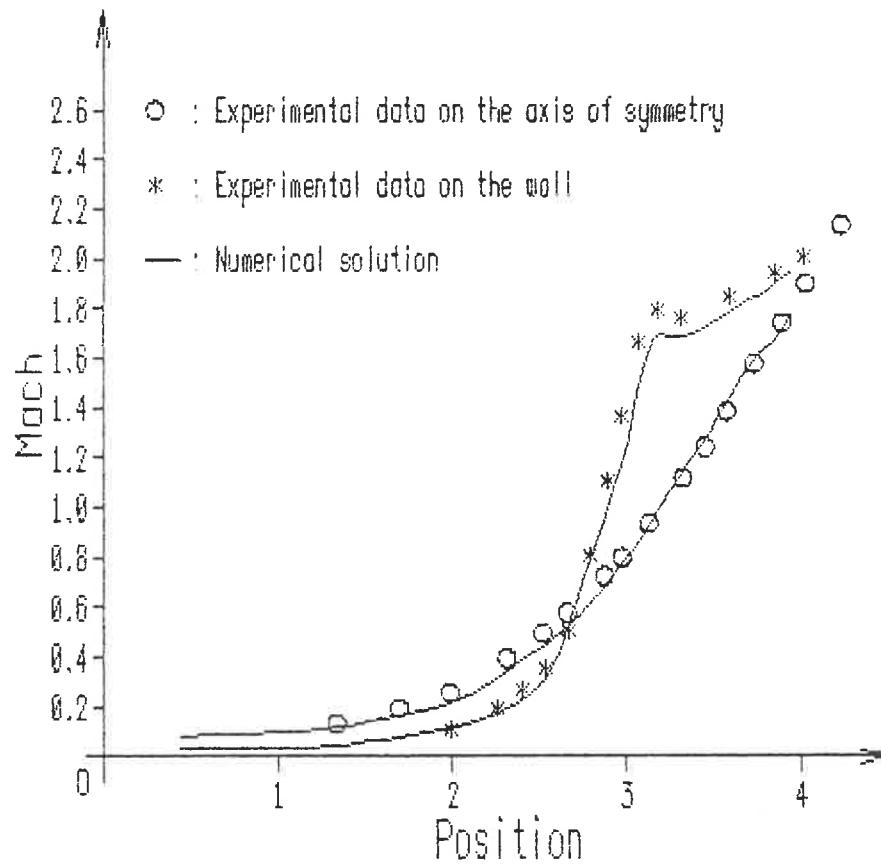
Figure 3.7: (a) domain and mesh with 1425 triangles; (b) comparison of numerical result with experimental data in terms of Mach number vs. wall position.

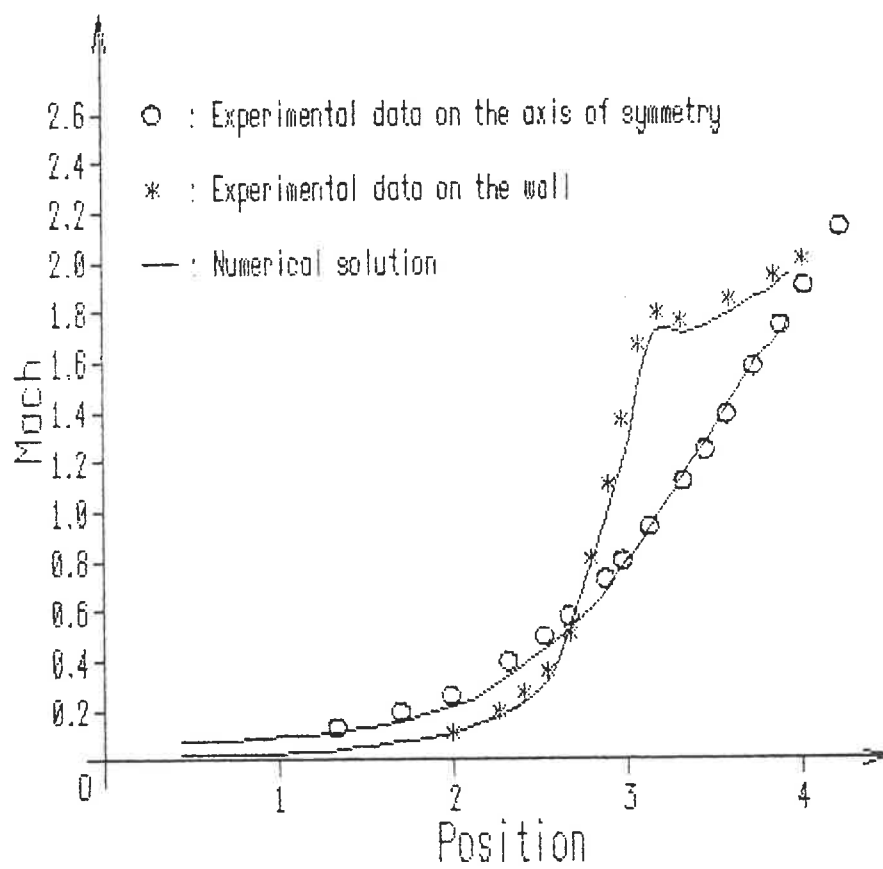


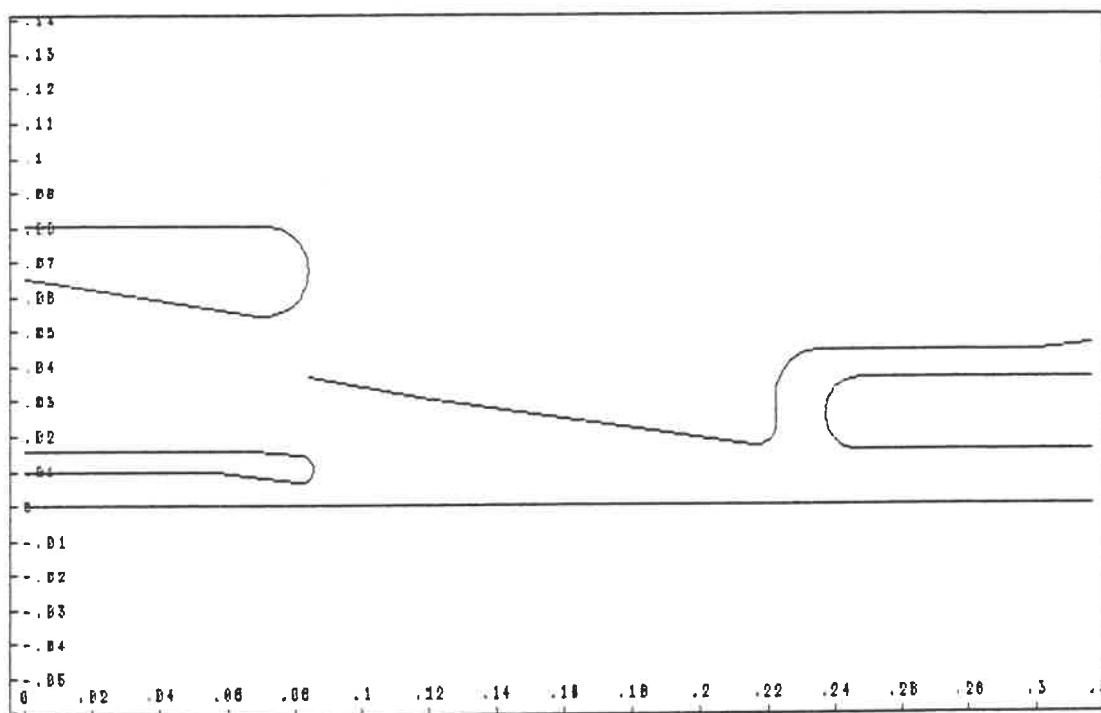


a

Figure 3.8: (a) domain and mesh with 1920 triangles; (b) solution on mesh *a*; (c) solution with more iterations on *b*.

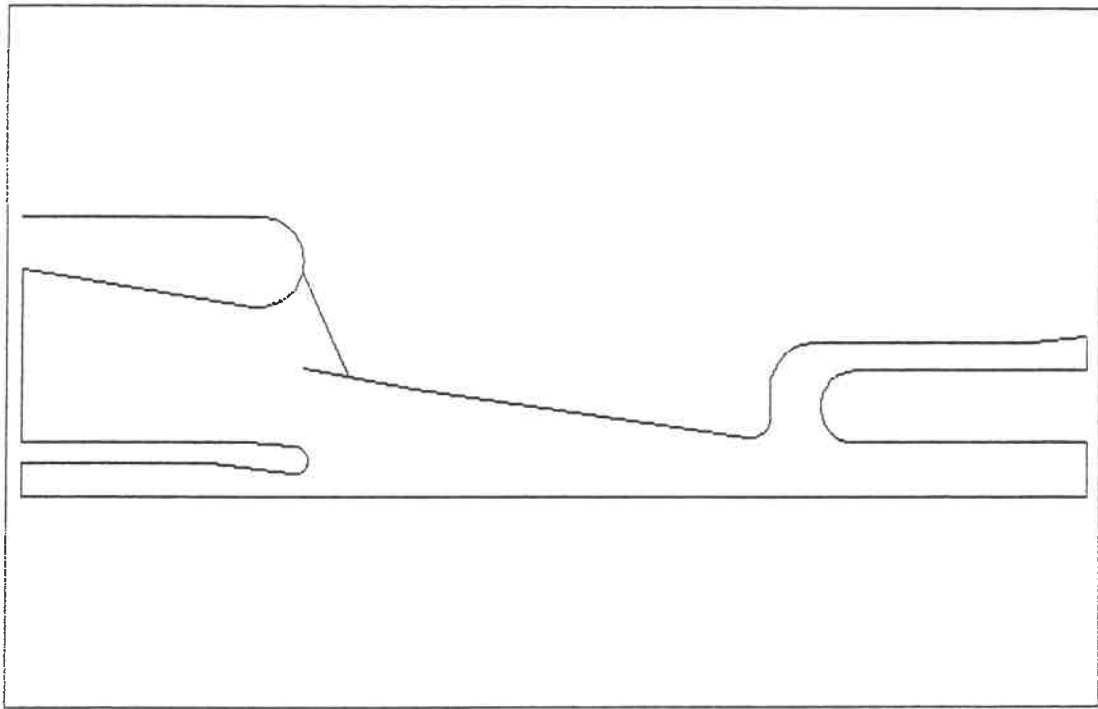




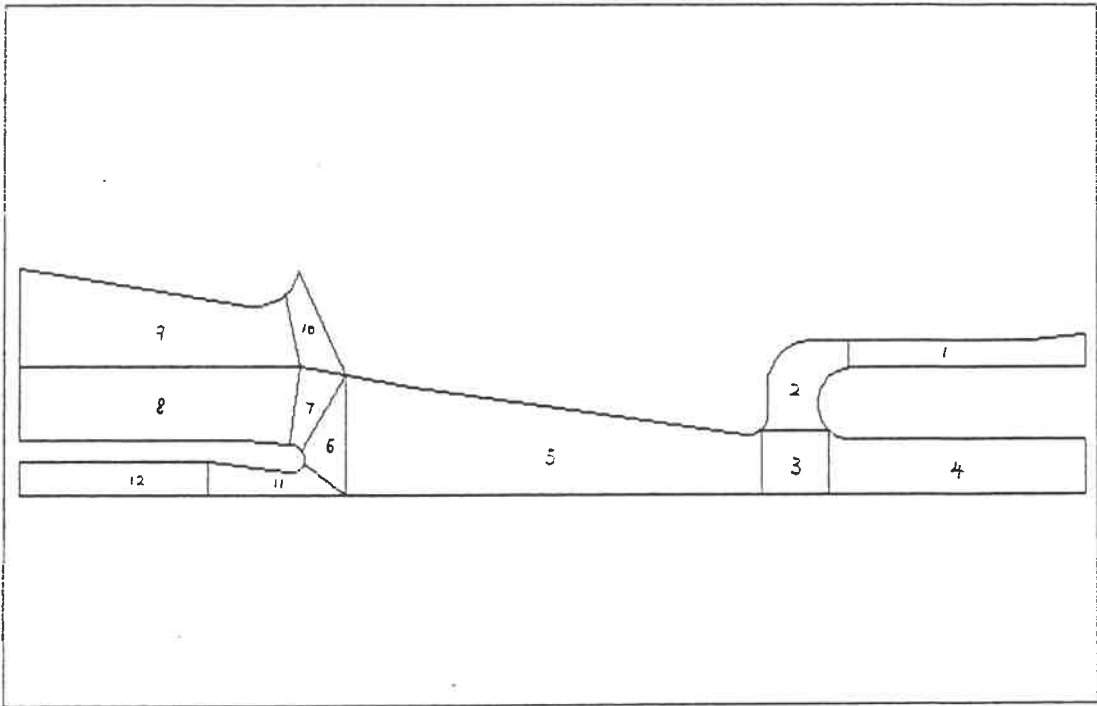


a

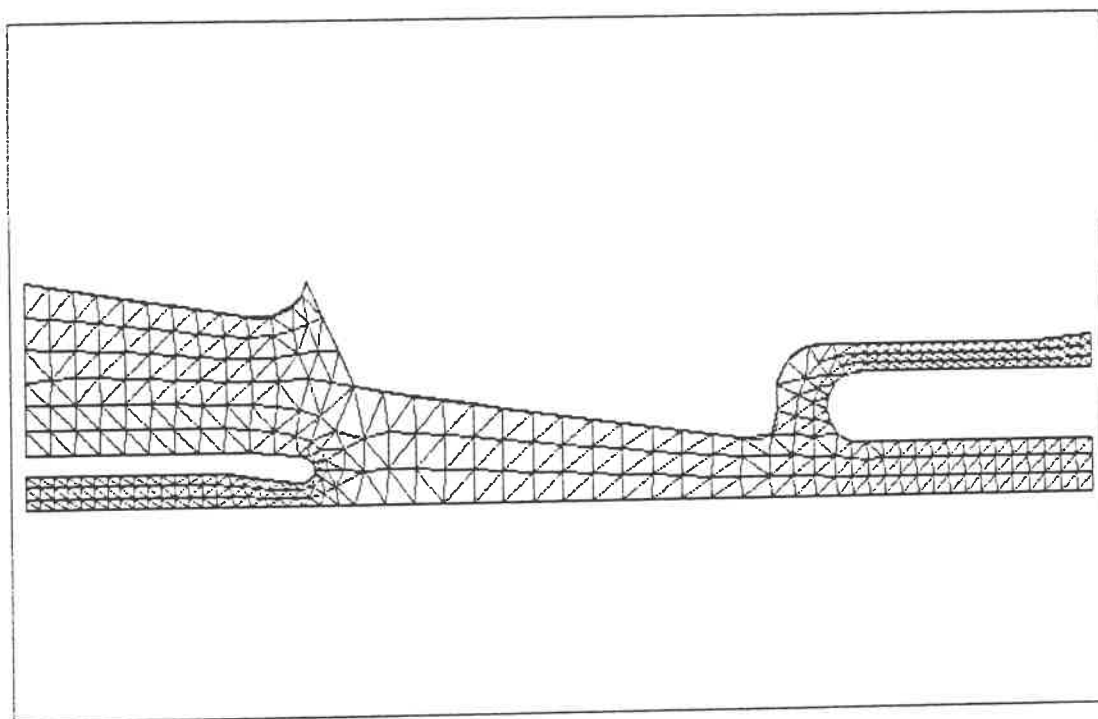
Figure 3.9: (a) configuration of the interior of a gas circuit breaker; (b) computational domain; (c) zoning pattern.



b

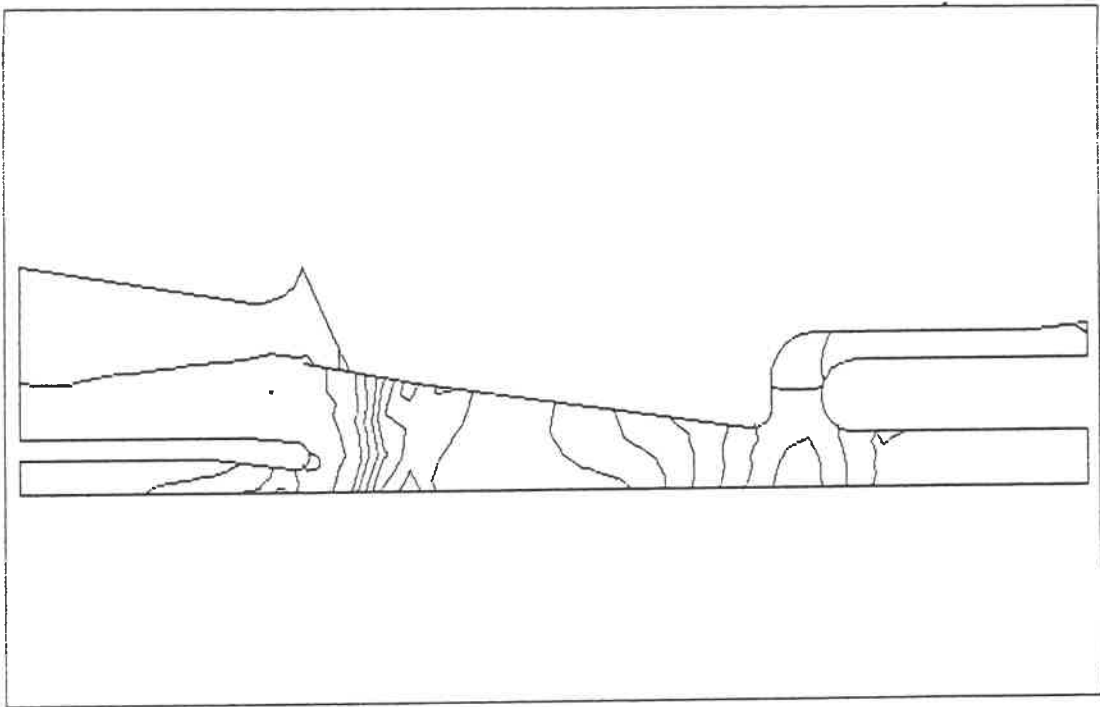


C

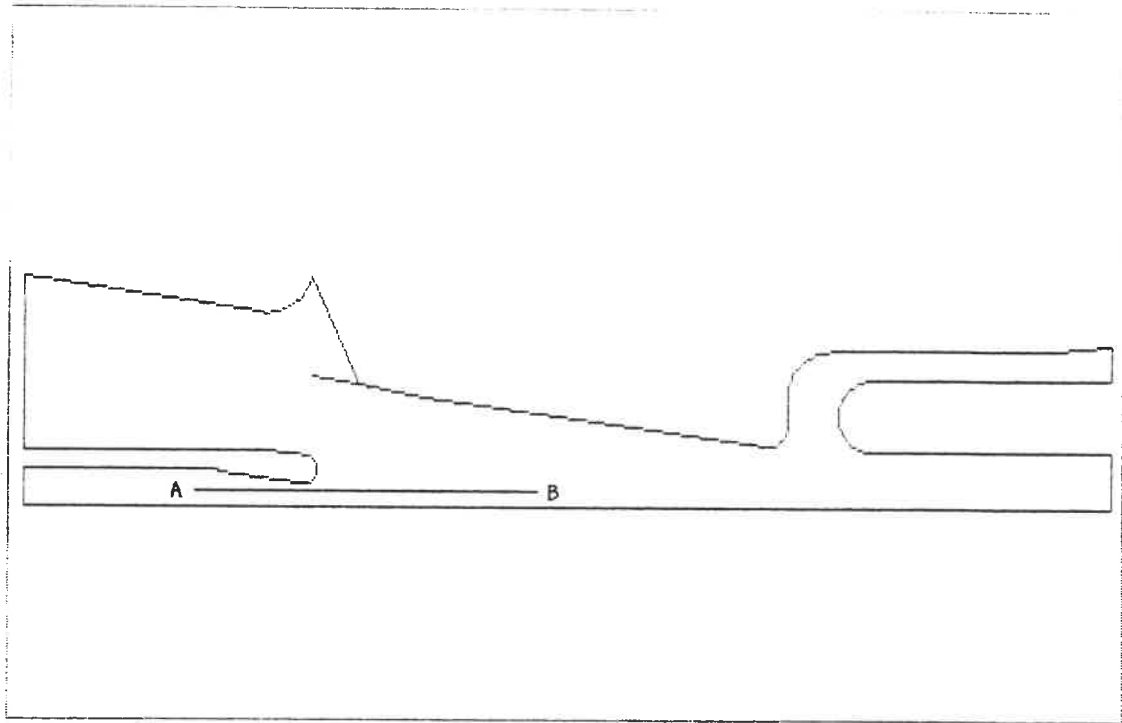


a

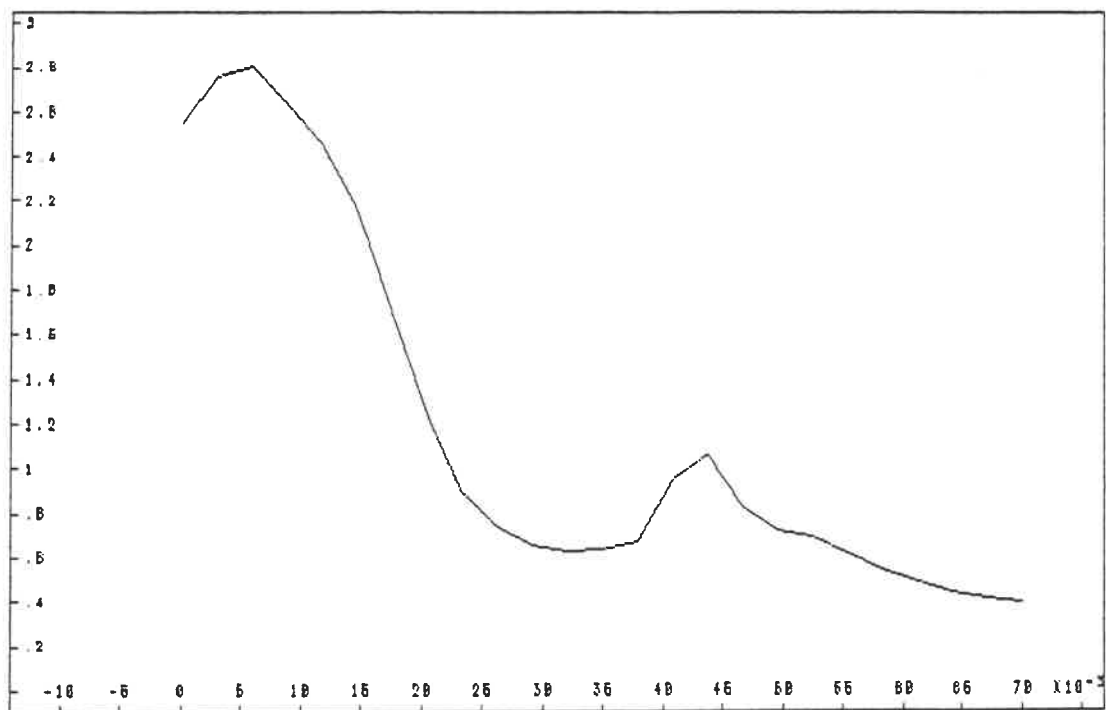
Figure 3.10: (a) mesh with 672 triangles; (b) lines of constant Mach number; (c) positions of line segments; (d) variation of Mach number along the axis; (e) variation of Mach number along \overline{AB} .

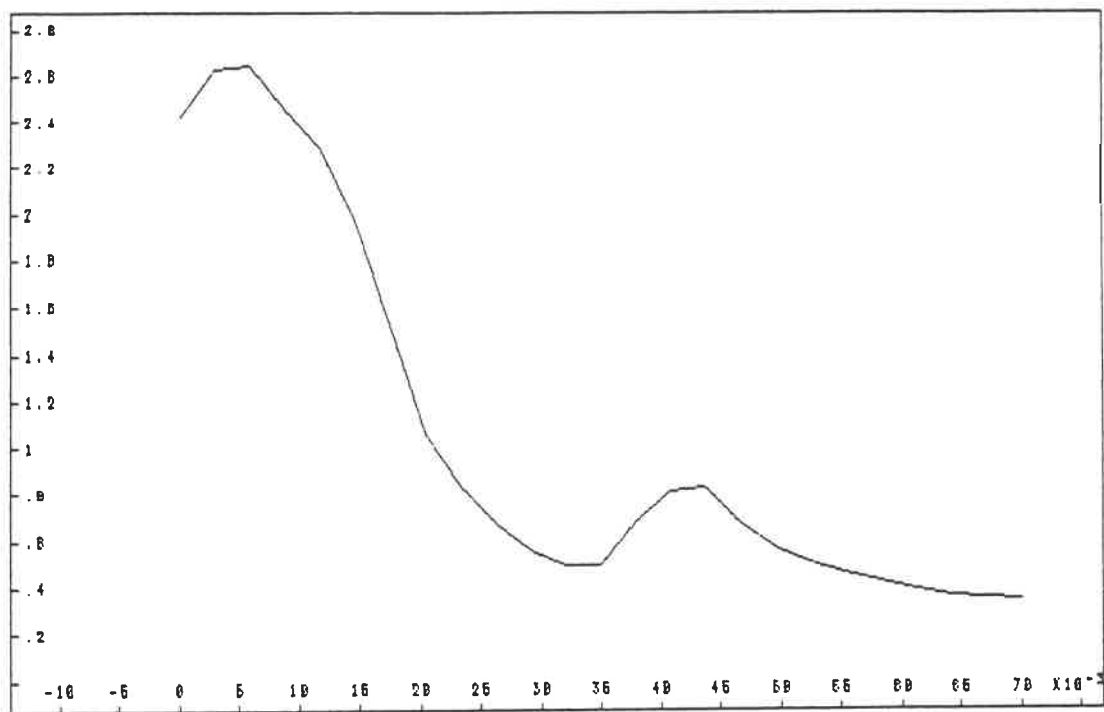


b

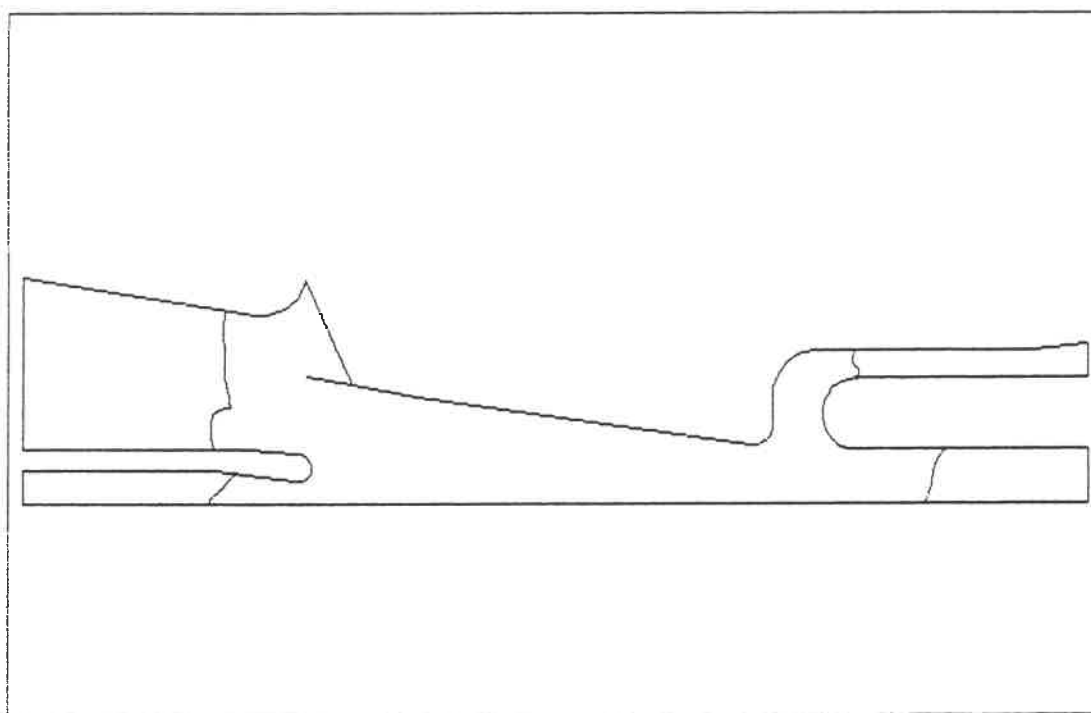


C

 d

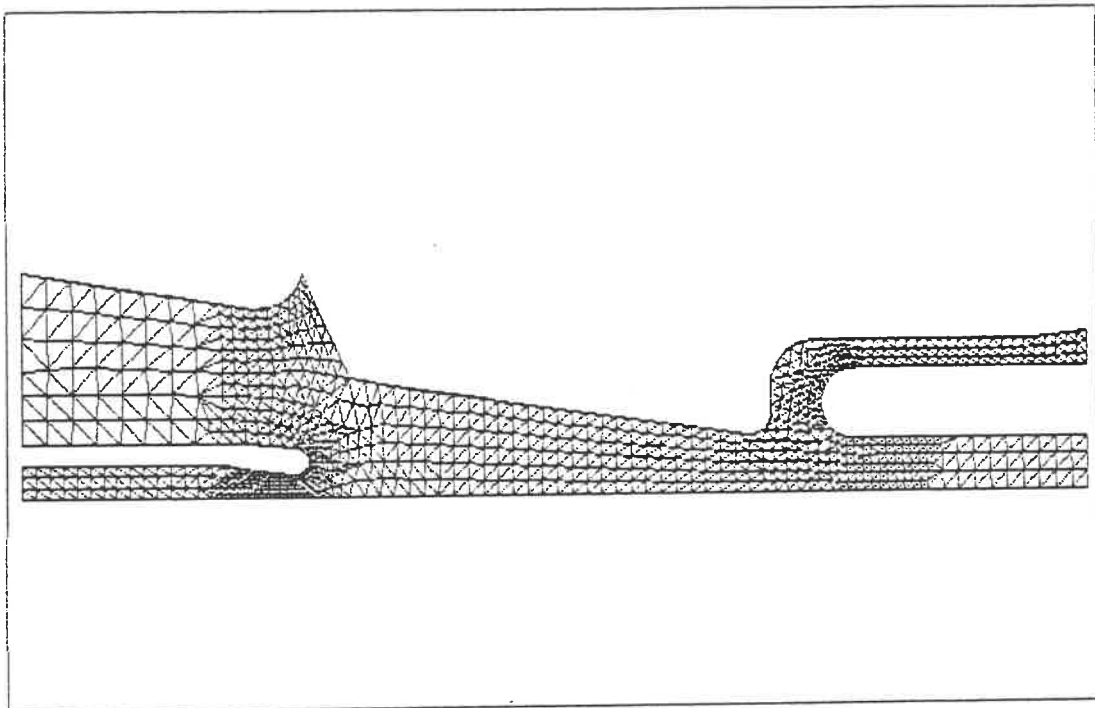


e

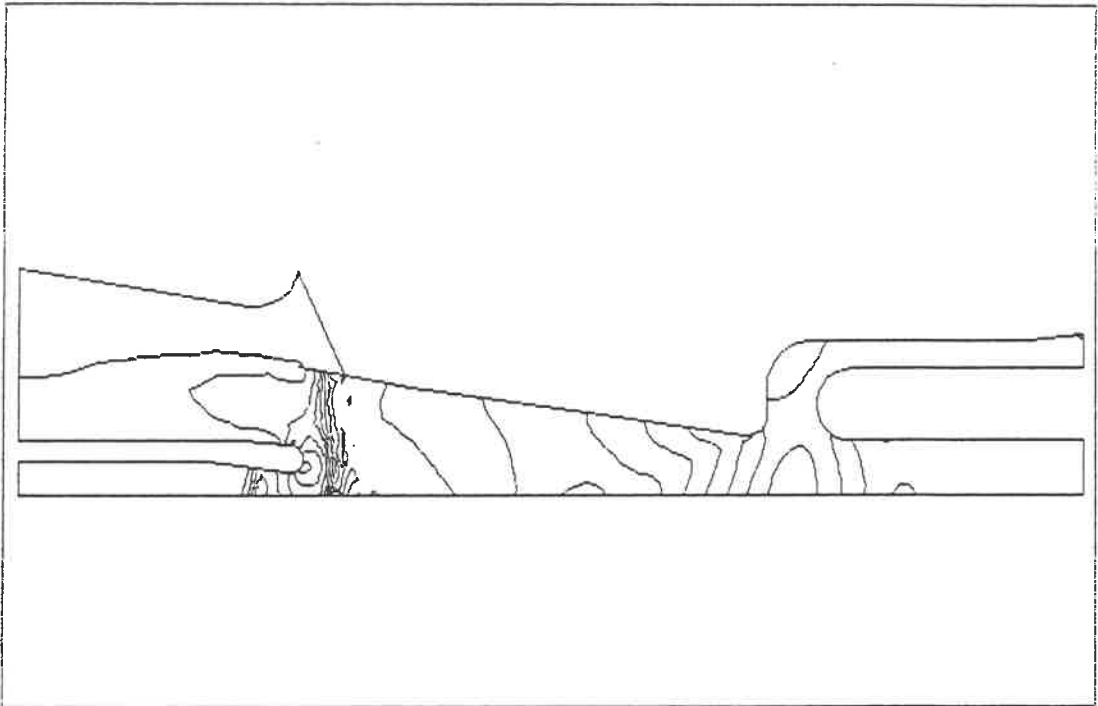


a

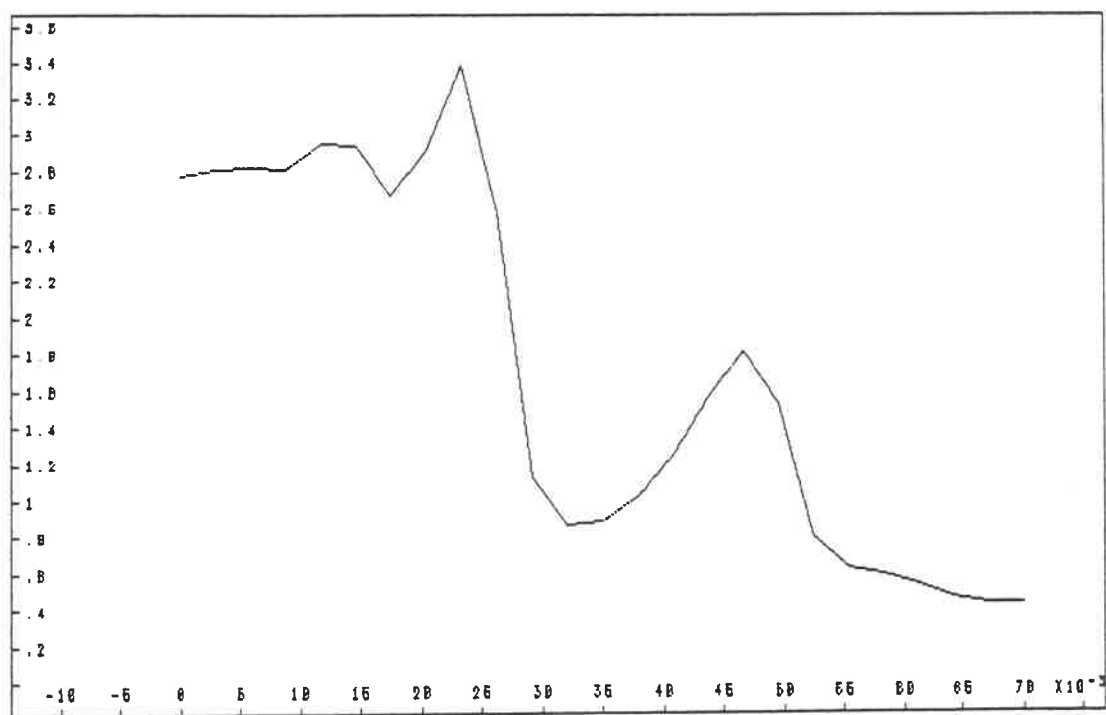
Figure 3.11: (*a*) area to be refined; (*b*) refined mesh with 1594 triangles; (*c*) lines of constant Mach number; (*d*) variation of Mach number along the axis; (*e*) variation of Mach number along \overline{AB} .

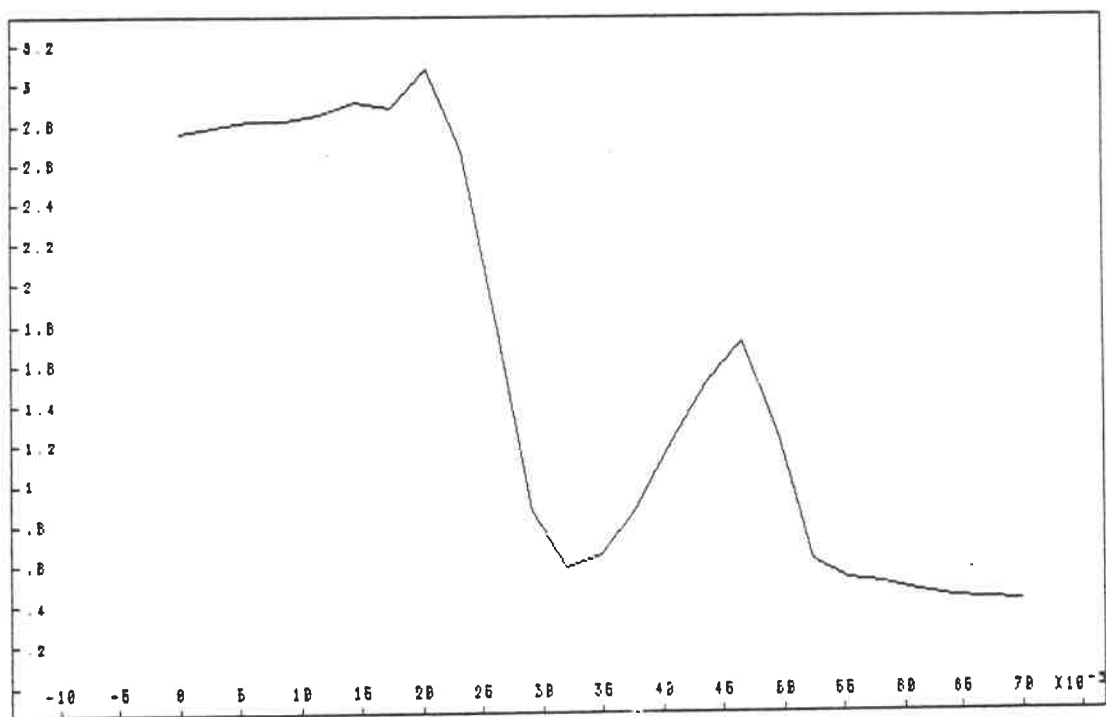


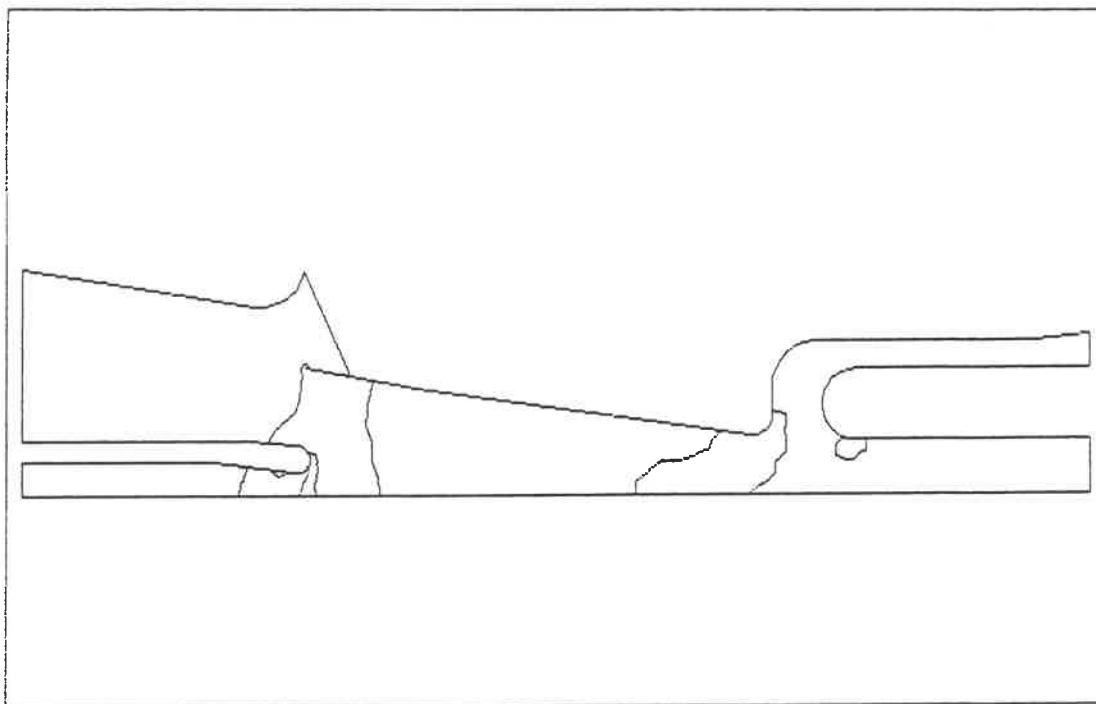
b



c

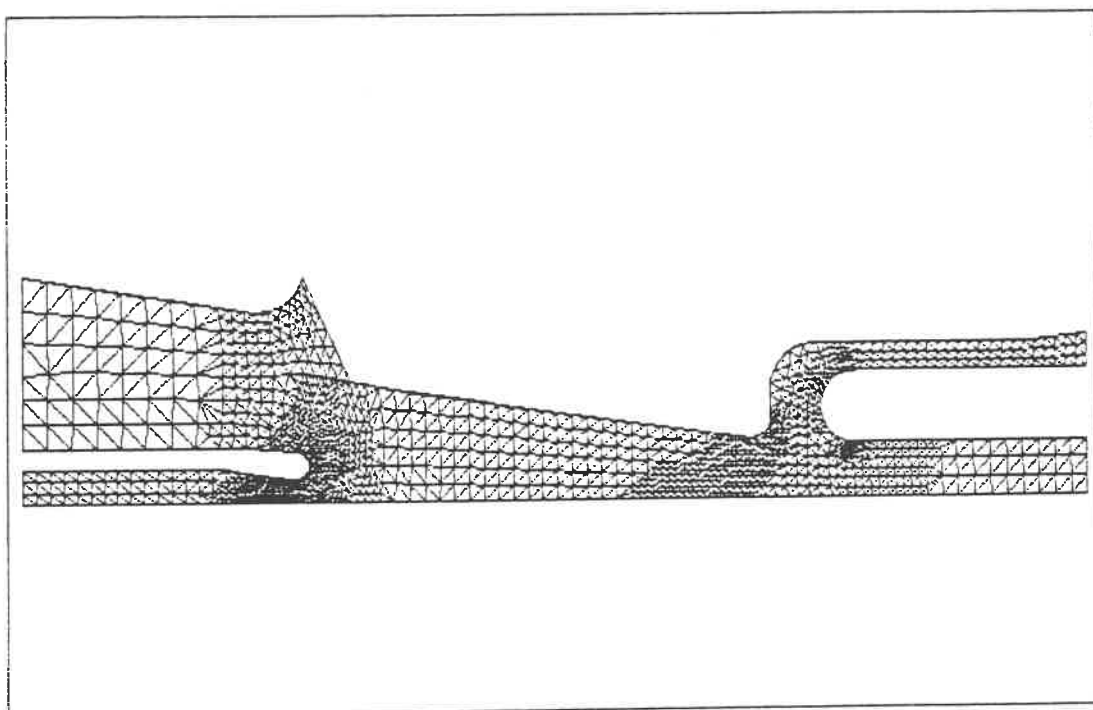
 d

*e*

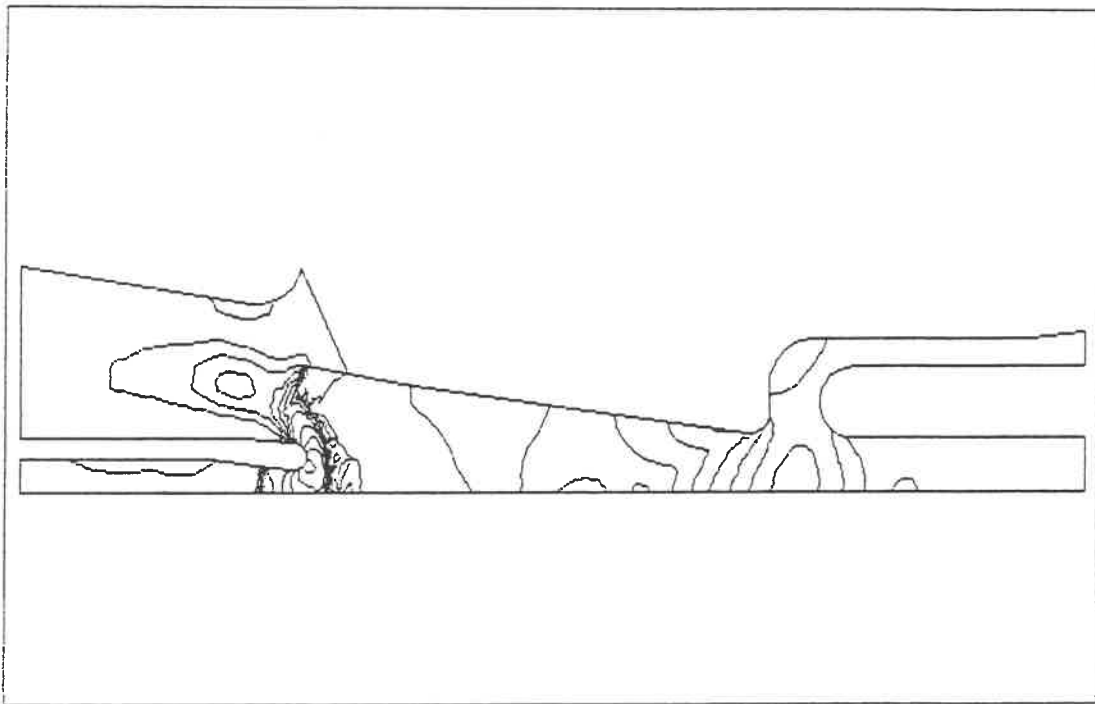


a

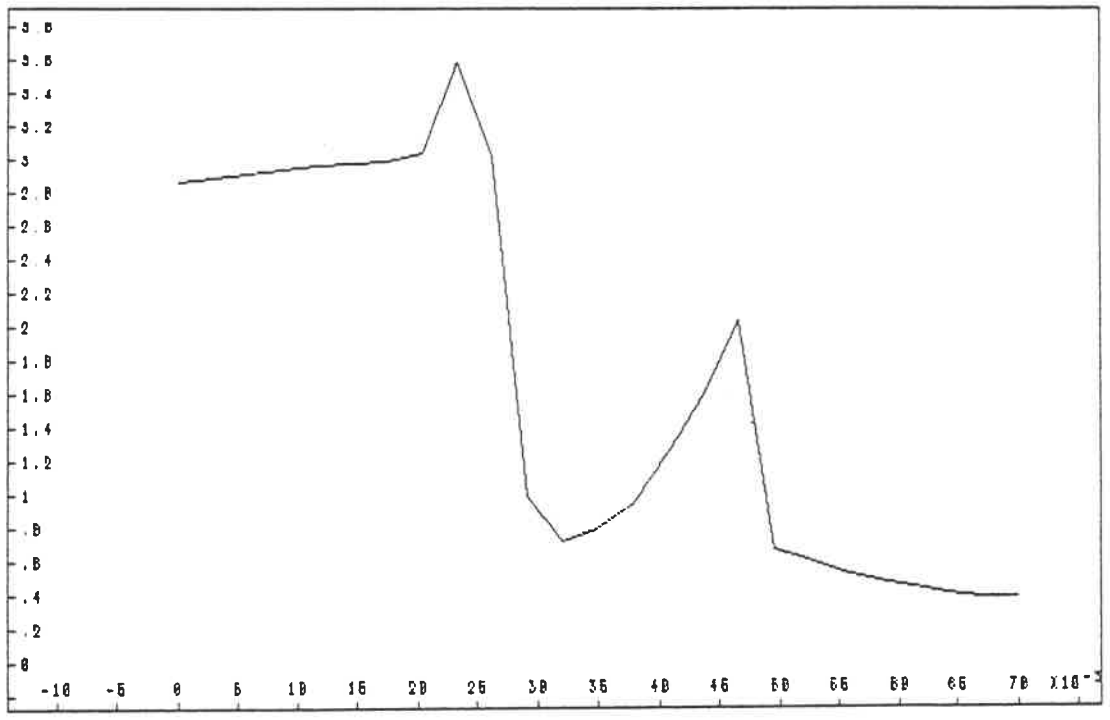
Figure 3.12: (a) area to be refined; (b) refined mesh with 2747 triangles; (c) lines of constant Mach number; (d) variation of Mach number along the axis; (e) variation of Mach number along \overline{AB} .



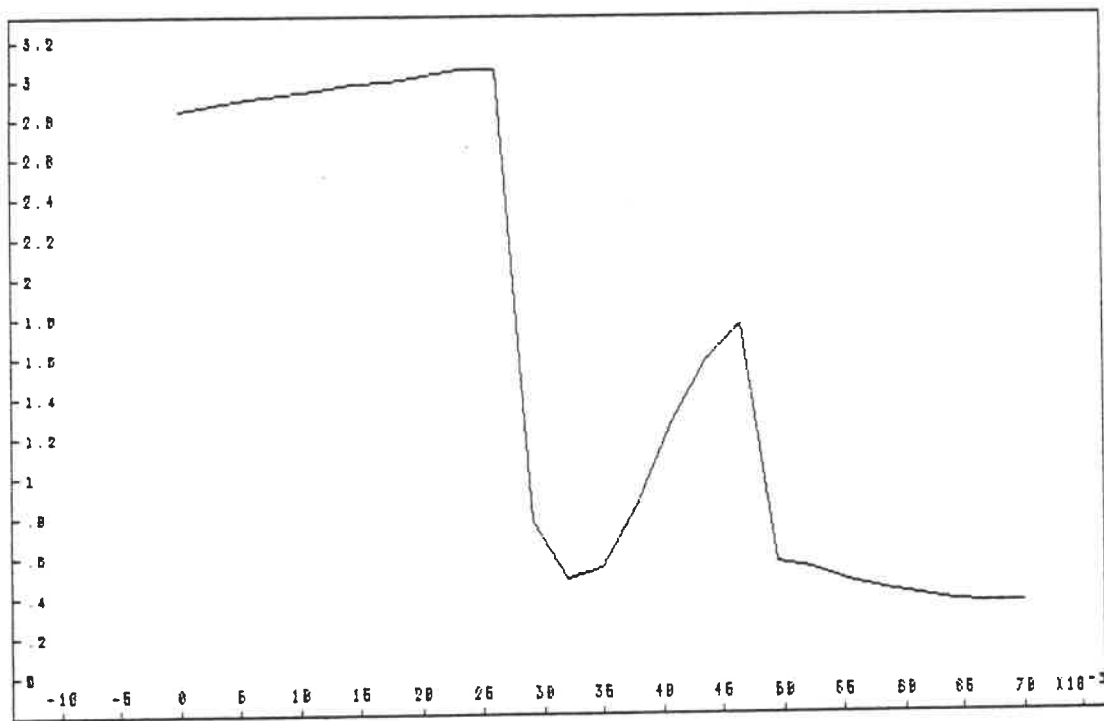
b



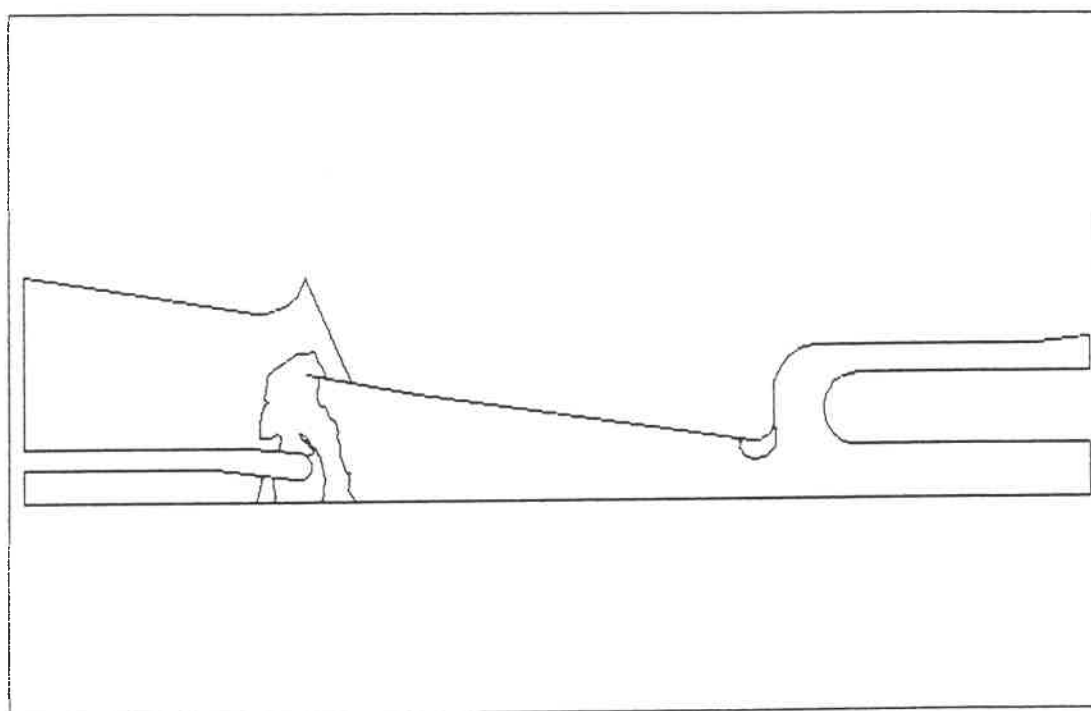
c



d

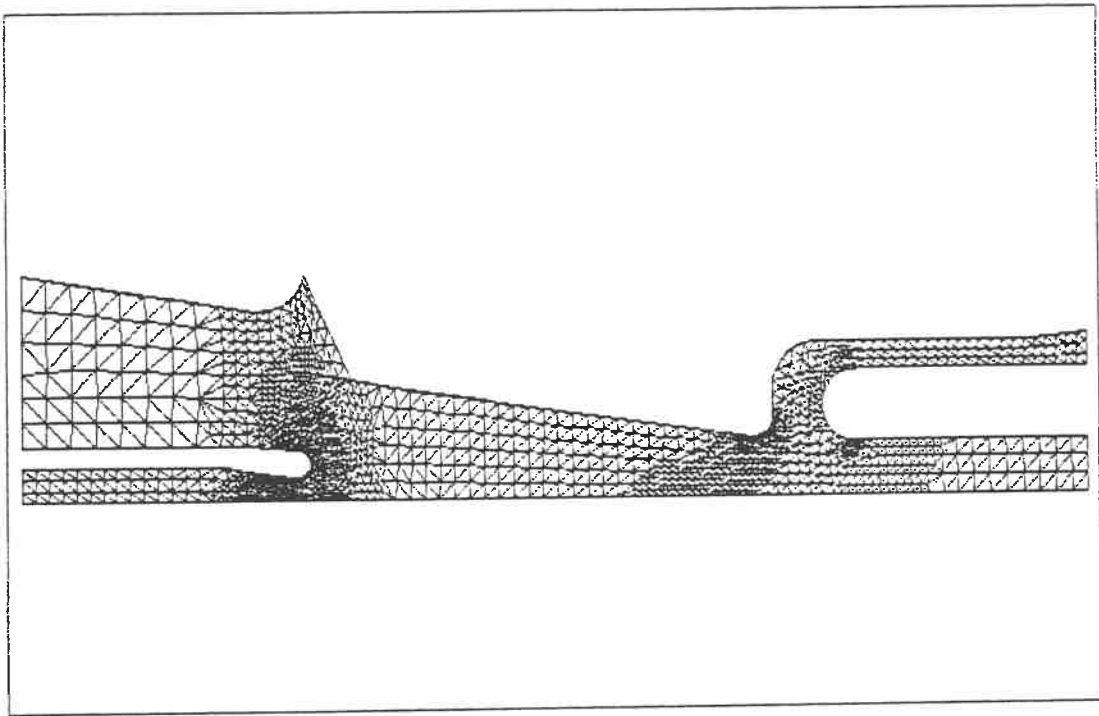


e

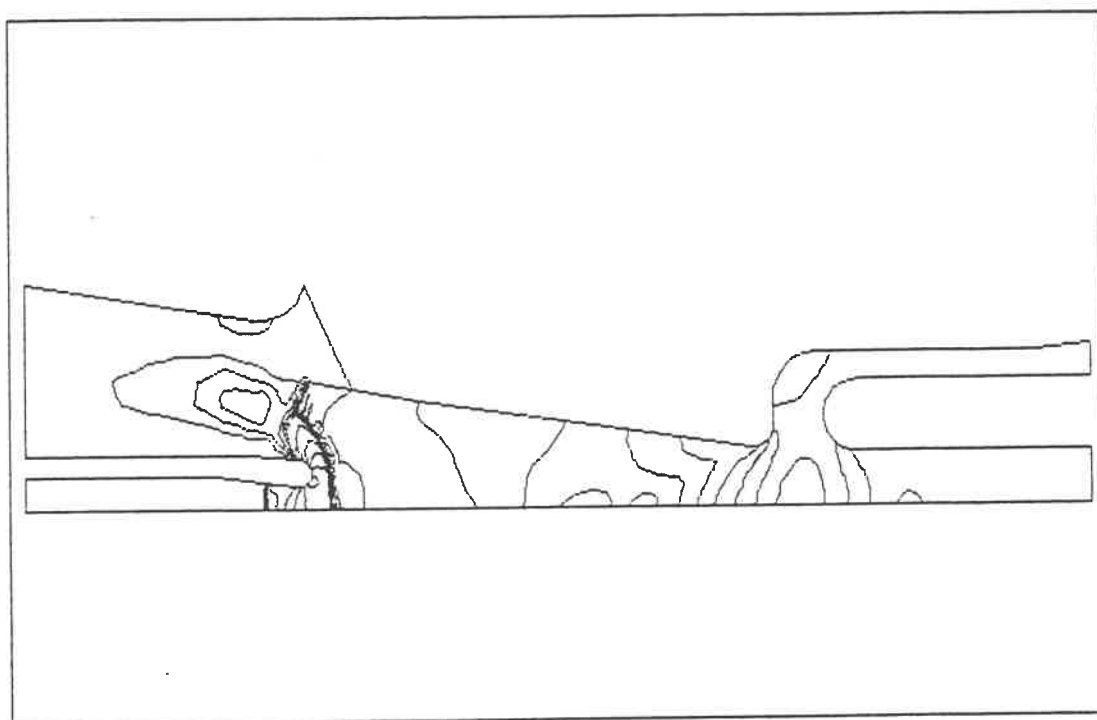


a

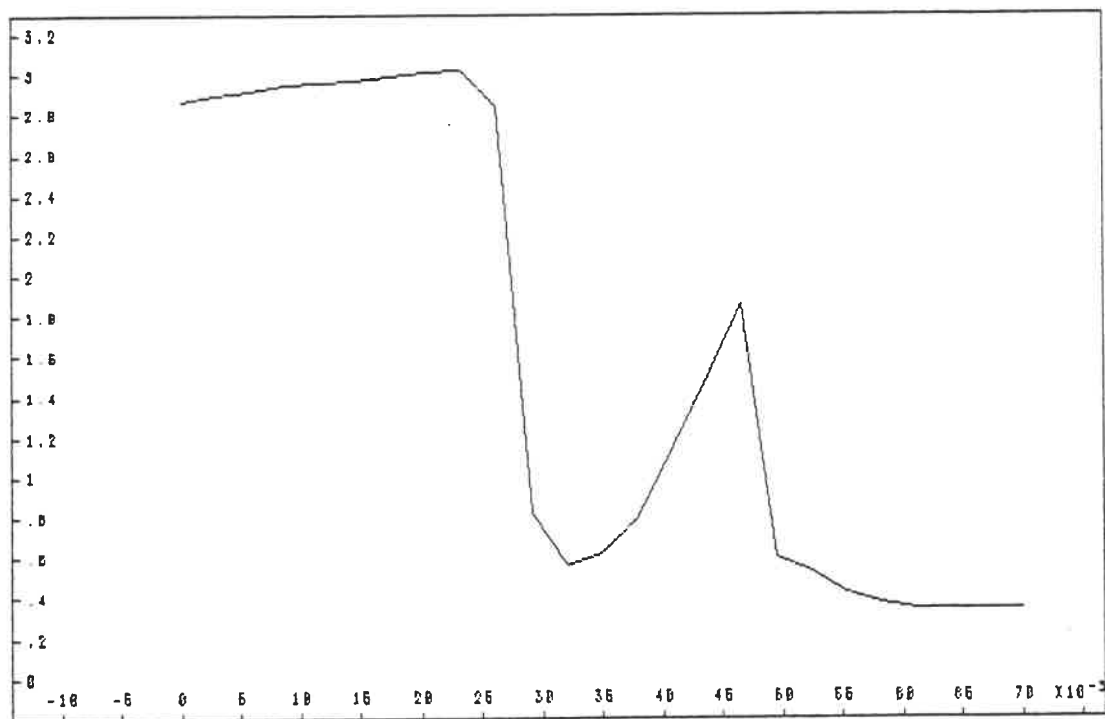
Figure 3.13: (a) area to be refined; (b) refined mesh with 4482 triangles; (c) lines of constant Mach number; (d) variation of Mach number along the axis; (e) variation of Mach number along \overline{AB} .

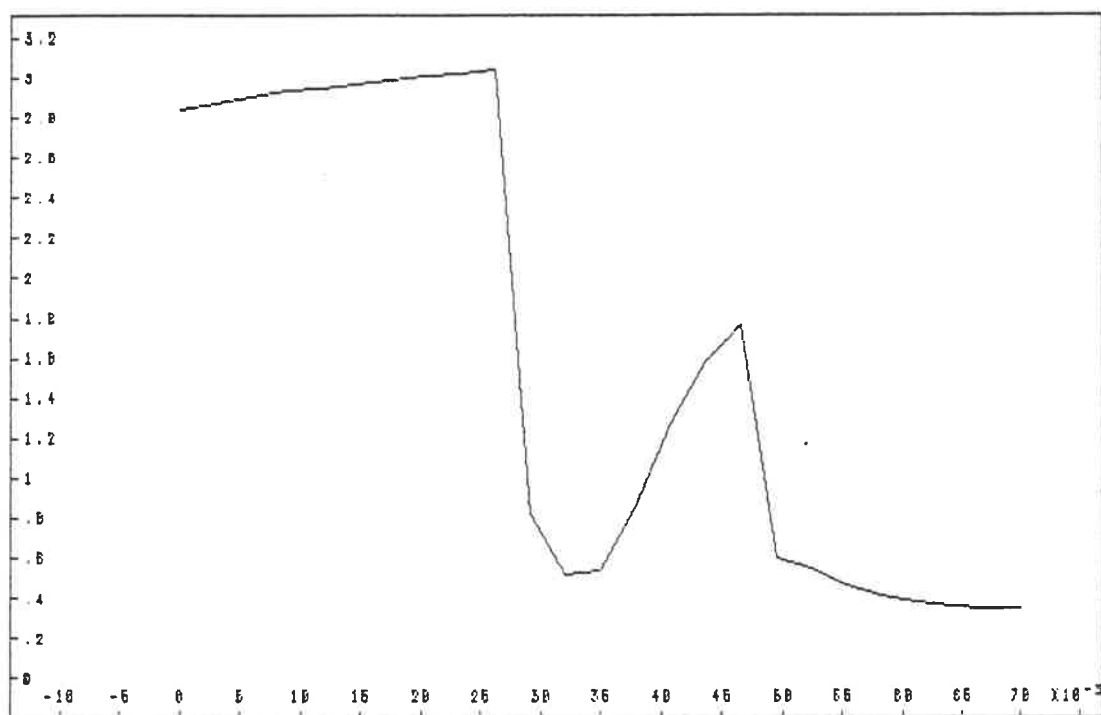


b

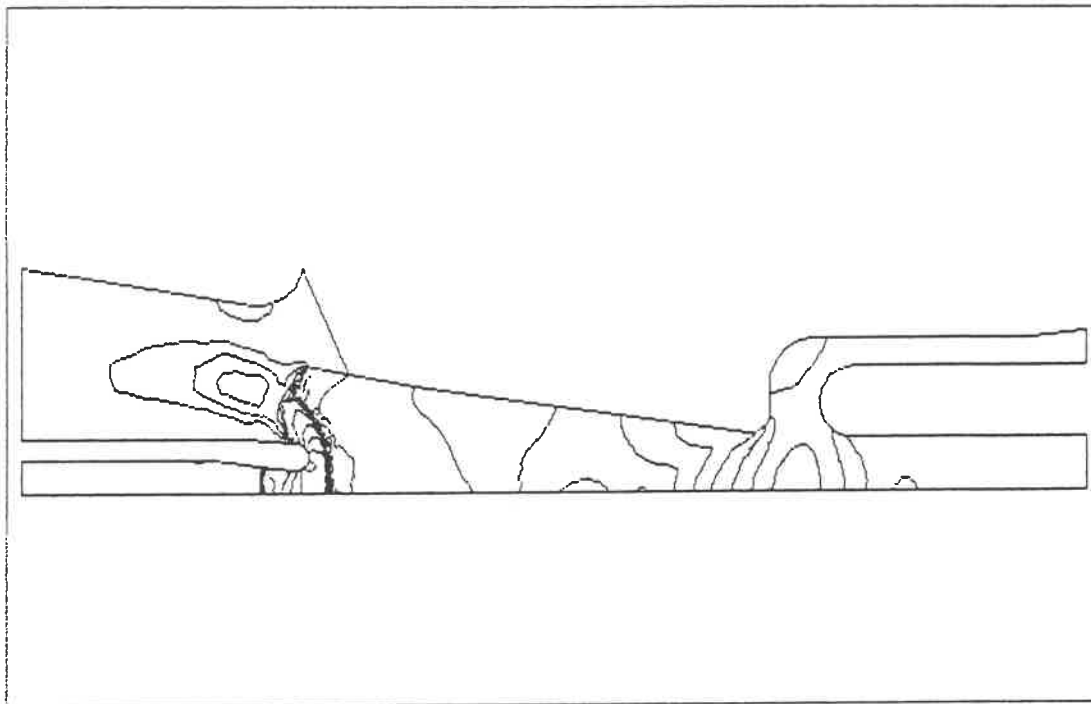


c

*d*

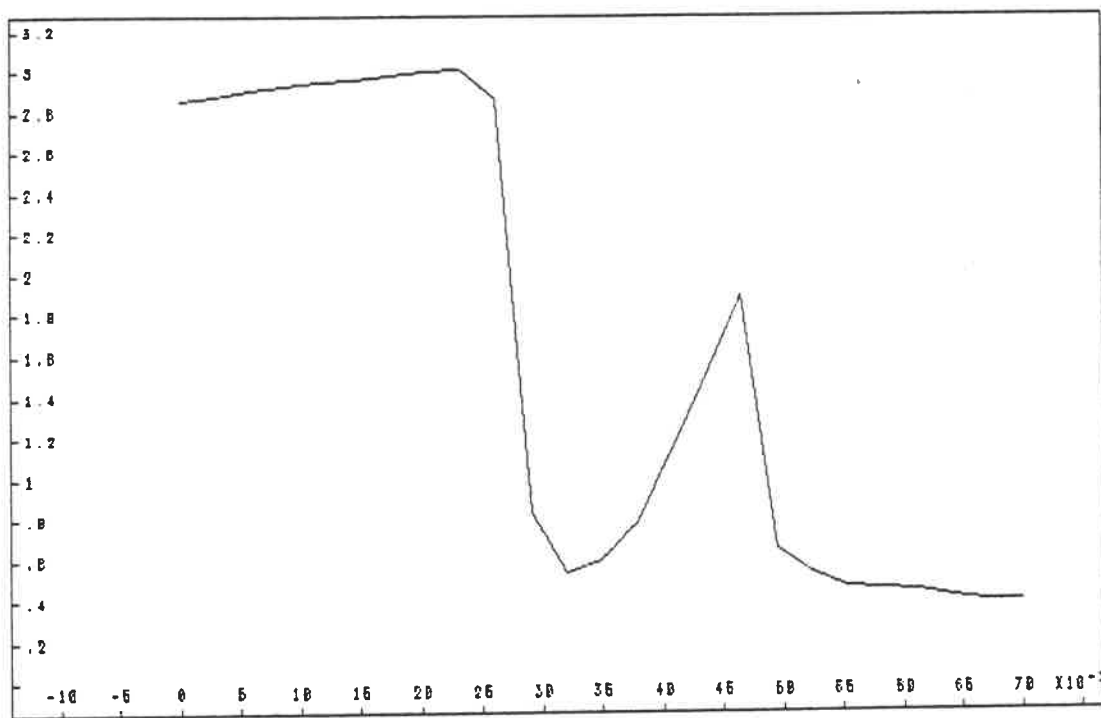


e

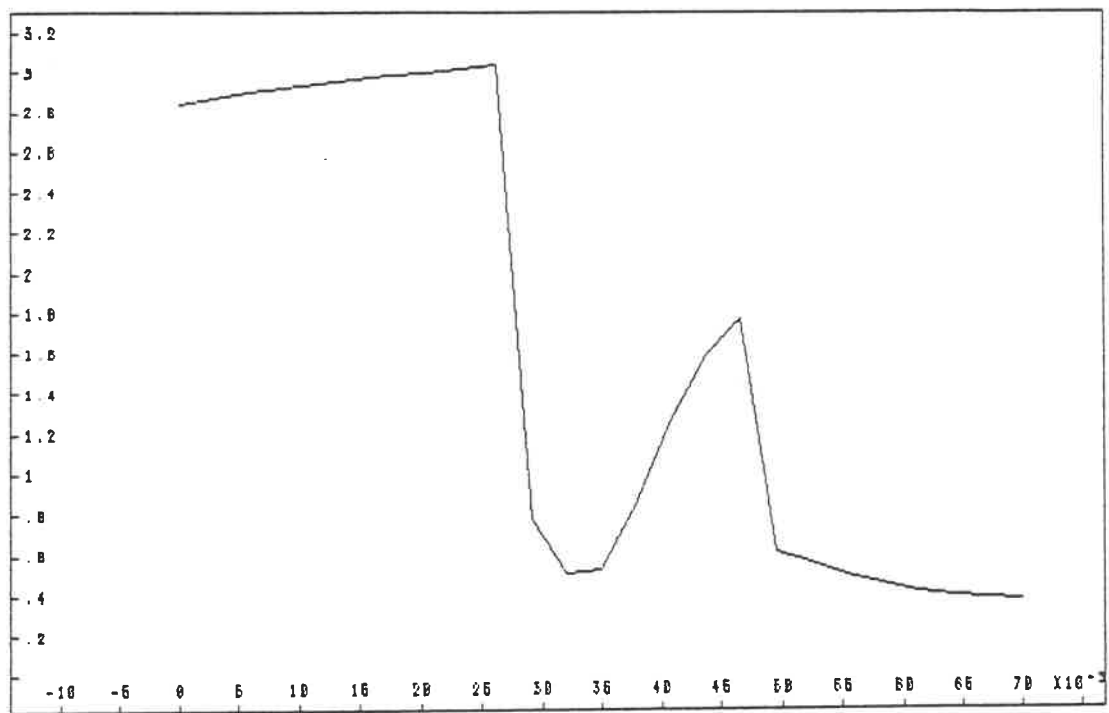


a

Figure 3.14: Result obtained on mesh in Figure 3.13*b*, (*a*) lines of constant Mach number; (*b*) variation of Mach number along the axis; (*c*) variation of Mach number along \overline{AB} .



b



c

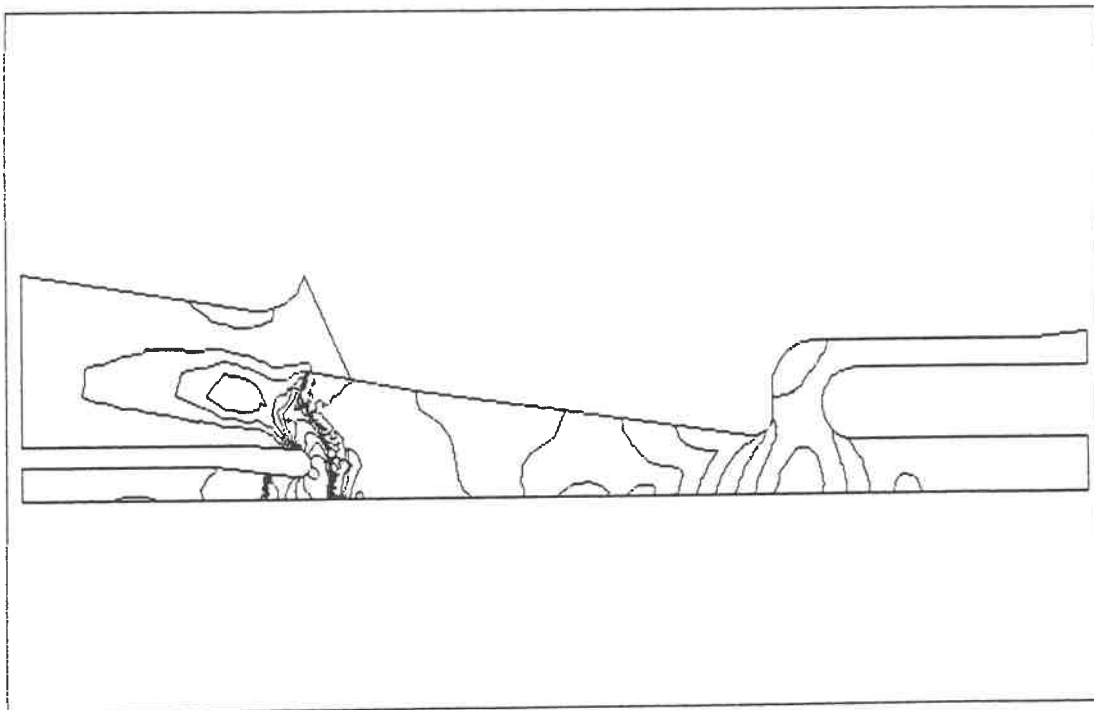
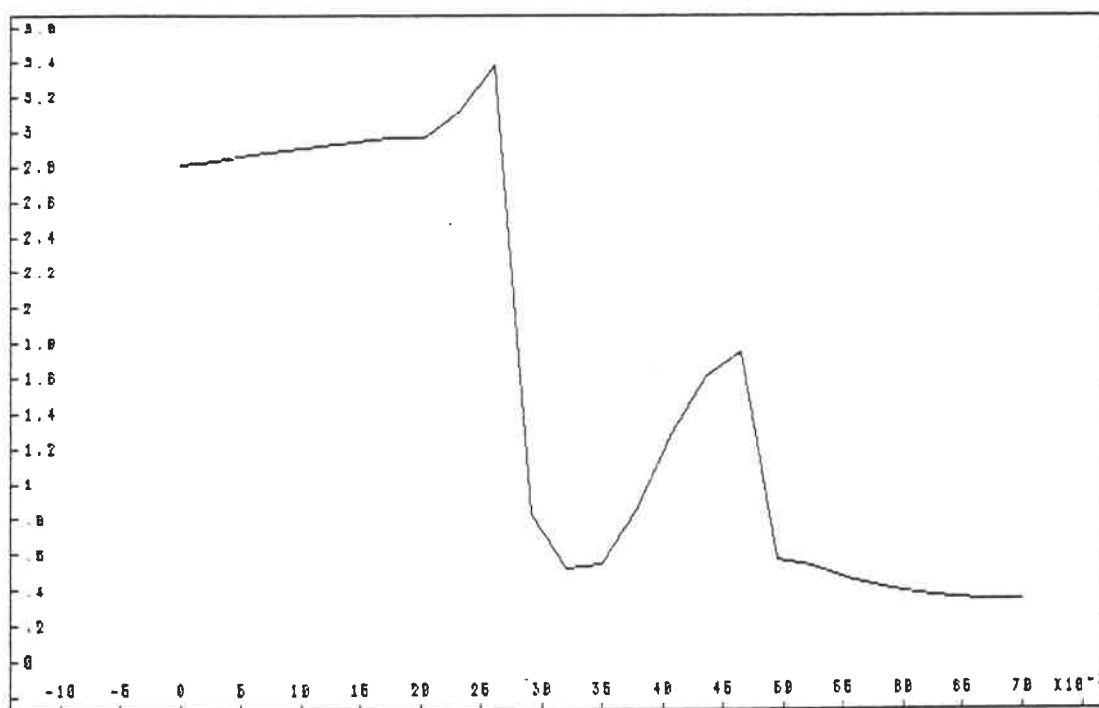
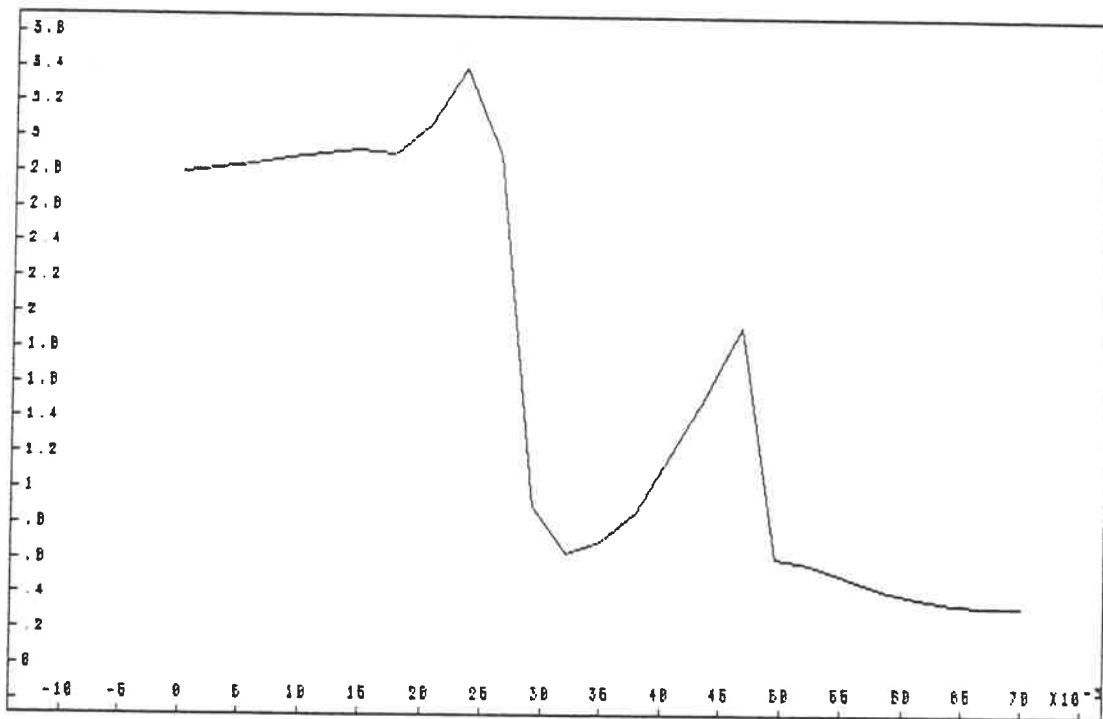
*a*

Figure 3.15: Result obtained on mesh in Figure 3.12*b*, (*a*) lines of constant Mach number; (*b*) variation of Mach number along the axis; (*c*) variation of Mach number along \overline{AB} .



b



C

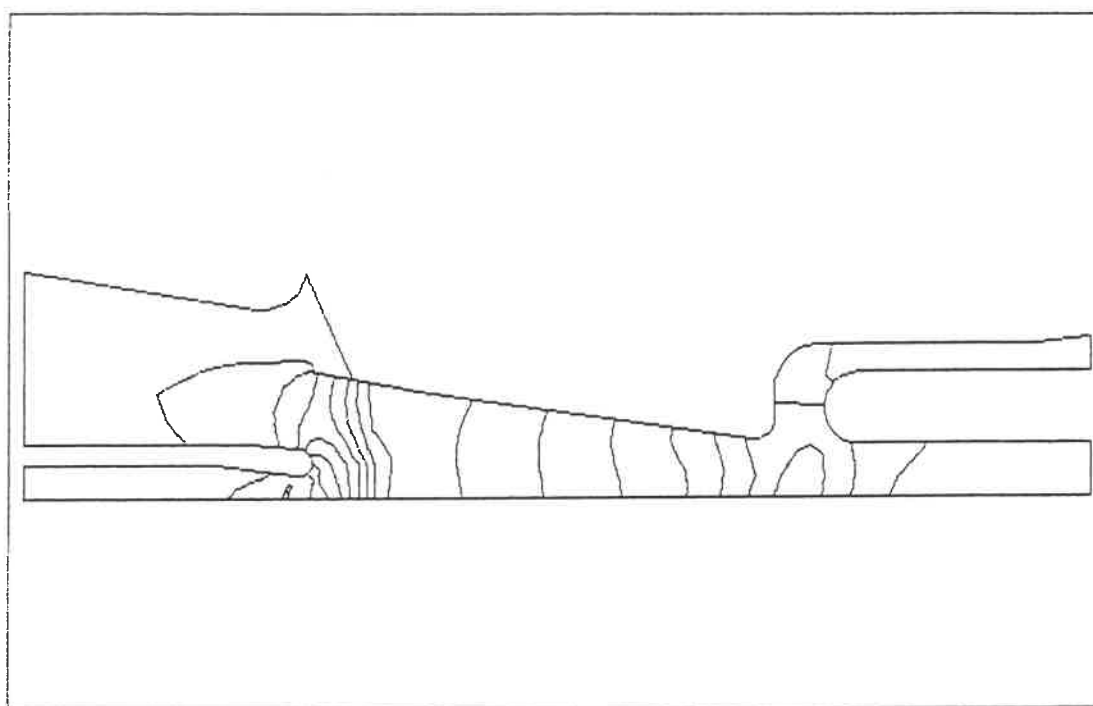
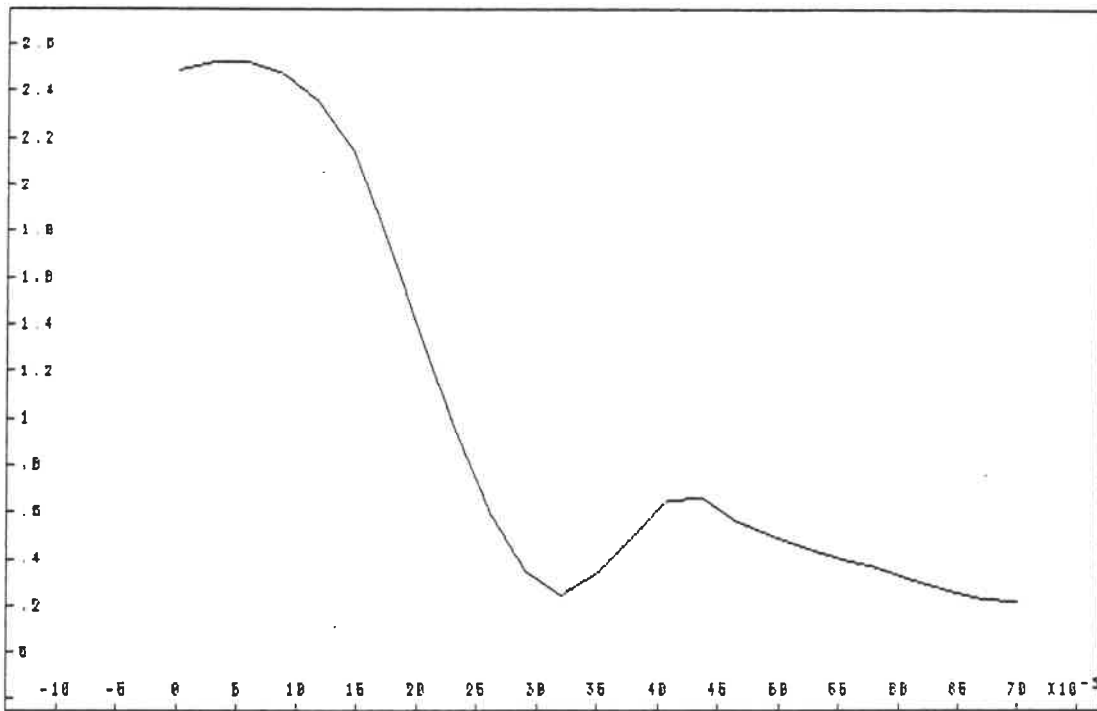
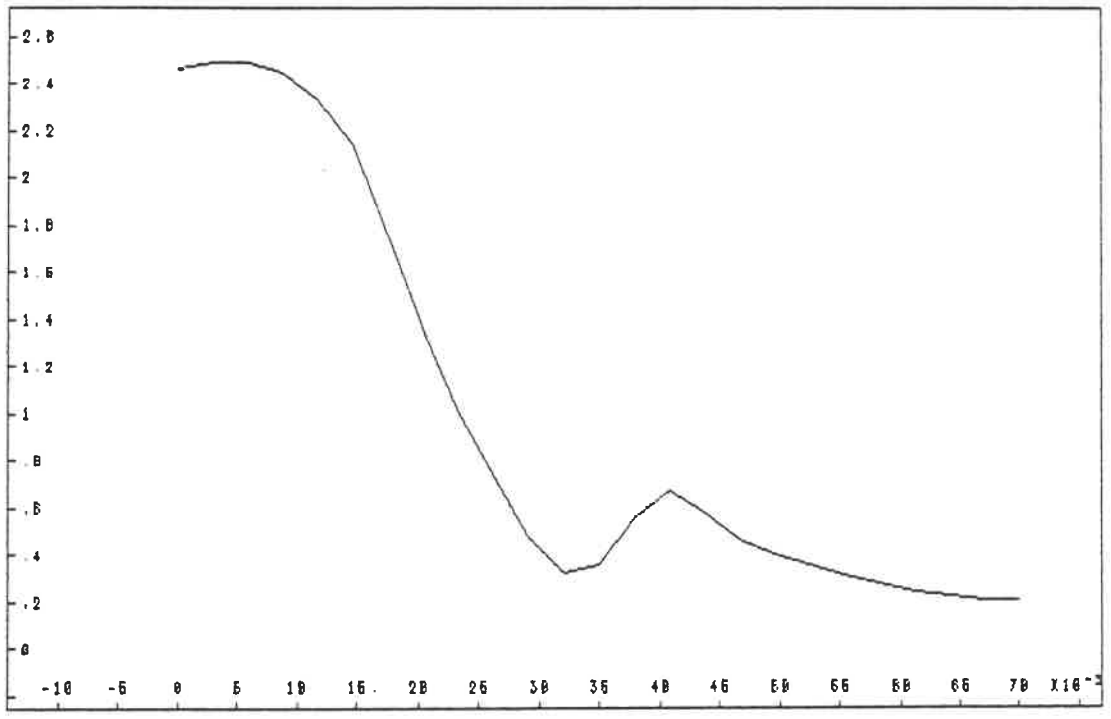
*a*

Figure 3.16: Result obtained on mesh in Figure 3.12*b*, (*a*) lines of constant Mach number; (*b*) variation of Mach number along the axis; (*c*) variation of Mach number along \overline{AB} .



b



c

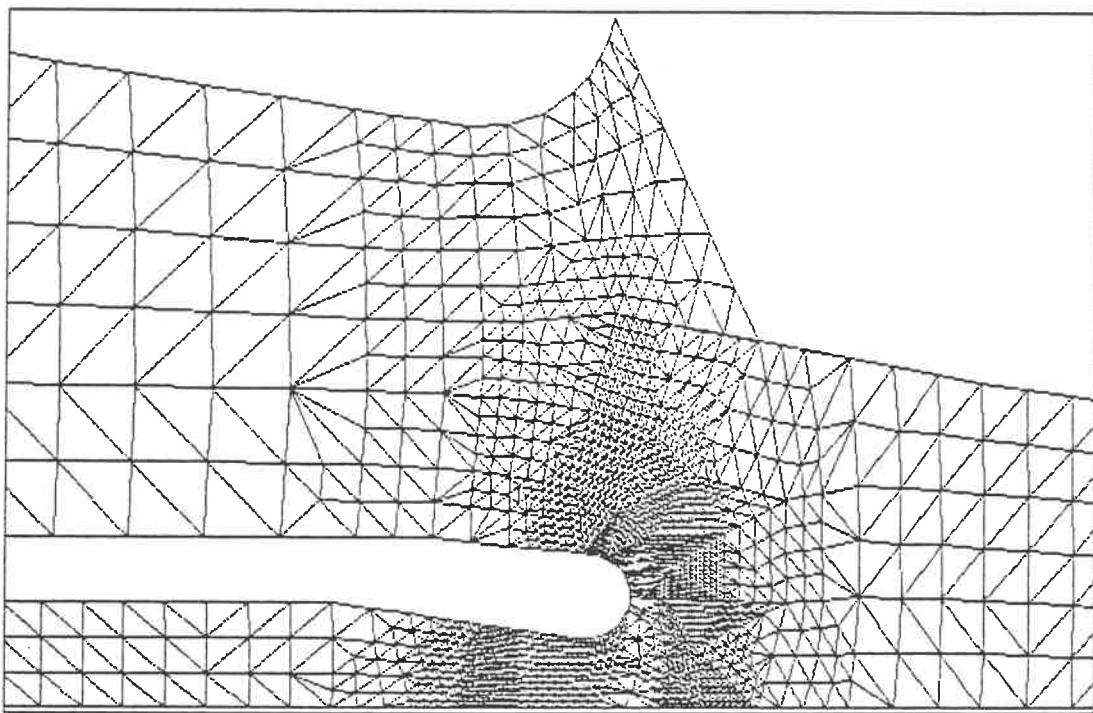


Figure 3.17: Different levels of refinement on mesh in Figure 3.13b.

| Figure No. | Level | Ratio and bound (C_r) of refinement | Iterations | Coefficients of dissipation ($sig2, sig4$) |
|------------|-------|--|------------|---|
| 3.10 | 1 | nil | 27,000 | 0.15, 0.07 |
| 3.11 | 2 | 302/672, 0.08 | 5,000 | 0.01, 0.07 |
| 3.12 | 3 | 357/1594, 0.05 | 15,000 | 0.07, 0.07 |
| 3.13 | 4 | 528/2747, 0.7 | 5,000 | 0.05, 0.07 |

Ratio of refinement = the number of refined triangles[†] / the total of triangles[†]

[†] Refer to the mesh read at the beginning of calculation, i.e., the mesh of the preceding level.

Table 3.1: Computational parameters.

Conclusions

In this work, the 2-D and axisymmetric compressible Euler equations were proposed as a primary mathematical model dealing with the flow behavior in an industrial gas circuit breaker. A numerical approach for solving such equations was programmed, tested and evaluated. It consists of node-based finite-volume spatial discretisation and Runge-Kutta time integration methods and was performed on unstructured triangular grids. The refinement of the discretisation was dynamically carried out through an adaptive process driven by the flow field. The validity of the Euler-equation solver was justified in comparison with both the analytic solution in a simple case and the experimental data for the flow passing through a converging-diverging nozzle from which the accuracy and the effectiveness of the adaptive process were also examined.

The numerical simulation of the intricate flow in a circuit breaker was then attempted. The results were obtained in a way that guaranteed the convergence of the calculations on meshes at different levels of refinement. This was achieved by first obtaining a convergent numerical solution on an elementary mesh referred to as level n . Then the mesh was refined to level $n + 1$ and a new convergent solution obtained. Thus level by level, a series of convergent results were achieved on meshes at different refinement levels. This procedure was terminated when the

difference between two neighboring solutions was within an error limit determined by the user. In other words, the solutions no longer varied significantly with mesh refinement (solutions on meshes at different refinement levels) and time iteration (solutions on a fixed mesh but with differing iteration levels). This procedure was repeated for several cases with various flow conditions and the results were found to be quite robust and mesh-independent.

The current numerical scheme needs a controlled amount of artificial viscosity in order to suppress the instability which appears near a shock and/or in regions of high Mach numbers. In the implementation and validation phase of this work, it was noticed that numerical dissipation played a decisive role in governing the quality of results. Insufficient or excessive damping reduced the accuracy of numerical resolution. This has been comprehensively documented in [12]. The working values of dissipation were determined by trial and error, and they were observed to be effective for a given discretisation approach. A code with an automatic setting of the optimal dissipation values would be preferable to this heuristic investigation and is thus deserving of further study.

References

1. P. Dutt, Stable Boundary Conditions and Difference Schemes for Navier-Stokes Equations, *SIAM J. Numer. Anal.*, Vol. 25, No. 2, 1988, pp. 245–267.
2. D. A. Anderson, J. C. Tannehill and R. H. Pletcher, *Computational Fluid Mechanics and Heat Transfer*, Hemisphere, 1984.
3. R. Peyret and T. D. Taylor, *Computational Methods for Fluid Flow*, Springer-Verlag, 1983.
4. A. Jameson, A Nonoscillatory Shock Capturing Scheme Using Flux Limited Dissipation, *Large-Scale Computations in Fluid Mechanics, Lectures in Applied Mathematics*, Vol. 22, Part 1, Edited by B. E. Engquist, S. Osher and R. C. J. Somerville, American Mathematical Society, 1985, pp. 345–370.
5. R.-H. Ni, A Multiple-Grid Scheme for Solving the Euler Equations, *AIAA J.*, Vol. 20, No. 11, 1982, pp. 1565–1571.
6. K. W. Morton and M. F. Paisley, A Finite Volume Scheme with Shock Fitting for the Steady Euler Equations, *J. of Computational Physics*, Vol. 80, 1989, pp. 168–203.

7. D. Mavriplis and A. Jameson, Multigrid Solution of the Two-Dimensional Euler Equations on Unstructured Triangular Meshes, AIAA Paper, 87-0353, 1987.
8. A. Jameson, A. Rizzi, W. Schmidt and E. Turkel, Numerical Solutions of the Euler Equations by Finite Volume Methods Using the Runge-Kutta Time Stepping Schemes, AIAA Paper, 81-1259, 1981.
9. M. J. Siclari, P. Delguidice and F. Koch, Euler Flows about Arbitrary Cross Sections Using a Node Centered Finite Volume Scheme, *Comput. Fluids*, Vol. 17, No. 1, 1989, pp. 263-287.
10. A. Arts, Cascade Flow Calculations Using a Finite Volume Method, *Numerical Methods for Flow in Turbomachinery Bladings*, VKI Lectures Series 5, 1982, pp. 31-35.
11. R. F. Cuffel, L. H. Back and P. F. Massier, Transonic Flowfield in a Supersonic Nozzle with Small Throat Radius of Curvature, *AIAA J.*, Vol. 7, No. 7, 1969, pp. 1364-1366.
12. D. R. Lindquist and M. B. Giles, A Comparison of Numerical Schemes on Triangular and Quadrilateral Meshes, 11th International Conference on Numerical Methods in Fluid Dynamics, *Lecture Notes in Physics 323*, D. L. Dwoyer, M. Y. Hussaini and R. G. Voigt (Eds.), Springer-Verlag, 1989, pp. 369-373.

ÉCOLE POLYTECHNIQUE DE MONTRÉAL



3 9334 00290875 2