

1513461

A THESIS

entitled

Control Volume Unstructured Mesh Procedure for Convection-Diffusion Solidification Processes

by

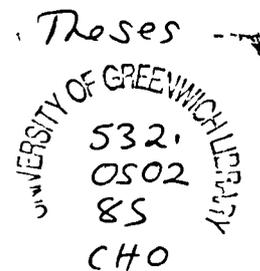
Peter M-Y Chow.

Thesis submitted in partial fulfilment of the requirements for the award of the

DEGREE OF DOCTOR OF PHILOSOPHY

of the

UNIVERSITY OF GREENWICH



Centre for Numerical Modelling and Process Analysis
School of Mathematics, Statistics and Computing
the University of Greenwich
London

January 1993

09282678

Abstract

The research work presented herein addresses the unstructured mesh problem in finite volume (FV) or control volume (CV) method used in numerical simulations. The modelling work conducted is in context of solidification for casting processes.

The control volume-unstructured mesh (CV-UM) method can be categorised into two approaches, a vertex-centred and a cell-centred approach. The classification of the approach is based on the relationship between the control volume and the unstructured mesh. The vertex-centred is naturally unstructured and has been used successfully in fluid flow and heat transfer calculations. The cell-centred on the other hand has always been associated with structured (quadrilateral) meshes, this has been extended to handle unstructured mesh in the current work and is called the irregular control volume (ICV) method. Both approaches have been studied for solidification by conduction only, using several standard phase change test cases and one with experimental data from the casting industry. The result of this work is reported and their suitability for solidification addressed.

For the ICV method, the extension to solve the full convective-diffusive solidification was undertaken, these are primarily the fluid flow and energy equations solved using the well known SIMPLE algorithm. One spin-off from the ICV is the appearance of "high-order cell" control volumes, control volumes with more than the standard four cell faces in two-dimensions. The high-order cell technique is exhibiting the same characteristics as high-order schemes used in standard CV method, when applied to standard CFD test cases. The one current drawback for the technique is the generation of these high-order cells, currently no fully- or semi-automatic mesh generation is available. This prevented further study of the technique and used in the solidification test cases, where in one, experimental data is available for the phase change fronts. This was carried out using quadrilateral meshes, but solved using the unstructured approach of the ICV. The predicted solution is in qualitative agreement with experiment.

The second convective-diffusive solidification problem is the first to demonstrate the CV-UM integrated framework by solving two major casting components simultaneously, the solidification (the work undertaken in this research) and the residual stress for deformation. This is still an on going research work, where refinement and validation are required and further integration of casting processes, such as mould filling, are necessary to complete the various stages of the shape casting process.

This kind of integrated simulation requires huge amount of computations, it will take days for traditional scalar computers to do one prediction. Vector and parallel machines offer ways in which to bring down the computing times to a level that is in hours instead of days. To utilise machines with vector and parallel capability efficiently, the algorithm of the model process need to be mapped onto such architectures for it to take full advantage of the computing powers. The solidification algorithm in three-dimensions has been vectorised and a speed-up of five is possible. This was part of a collective study into mapping algorithms onto vector and parallel computers, where it emerged that the ideal computing architecture is a network of processors each with its own vector capabilities.

Acknowledgements

The author would like to express his sincere gratitude to his supervisors Professor Mark Cross and Dr. Koulis. Pericleous for their invaluable advice, encouragement and guidance received during the course of this research.

He would also like to express his thanks to Dr. Mayur Patel for the invaluable discussions and assistance at various stages of the research.

Thanks also go to the staff at the school of Mathematics, Statistics and Computing, and to the postgraduates at the Centre for Numerical Modelling and Process Analysis of the University of Greenwich, for they contributed many stimulating discussions and ideas, and providing a good working environment.

To my family, to whom this thesis is dedicated, without their love, support and encouragement this research would not be possible.

Finally, the financial support provided by the Science and Engineering Research Council is gratefully acknowledged.

Table of Contents

Abstract	i
Acknowledgements	ii
Table of Contents	iii
Chapter 1	1
1.0 Introduction	2
1.1 Objectives	2
1.2 Practical relevance	2
1.3 Background information	5
1.4 Present contribution	8
1.5 Outline of present work	10
Chapter 2	13
2.0 Mapping a Solidification Algorithm onto Vector Architectures	14
2.1 Introduction	14
2.2 Enthalpy algorithm for solidification	14
2.3 Scalar algorithm	16
2.4 Solution procedure for the scalar algorithm	18
2.5 The model problem	19
2.6 Overview of the Masscomp 5400 system	20
2.7 The vectorised algorithm	22
2.8 Results and discussion	24
2.9 Closure	27
Figures	29

Chapter 3	32
3.0 Control Volume Based Unstructured Mesh Methods	33
3.1 Introduction	33
3.2 Vertex-centred Approach	34
3.2.1 Domain discretisation	34
3.2.2 Local-global co-ordinate transformation	35
3.2.5 Boundary conditions	42
3.2.6 Multi-material	43
3.2.6.1 Weighted average	45
3.2.6.2 Thin elements	46
3.2.6.3 Coincident nodes	47
3.3 Cell-centred Approach	50
3.3.1. Domain discretisation	50
3.3.2 Discretisation of solidification equations	51
3.3.3 Boundary conditions	52
3.4 Closure	53
Figures	54
 Chapter 4	 61
4.0 Solidification by conduction - Validations and results	62
4.1 Introduction	62
4.2 Solution procedure	63
4.3 Test cases	63
4.3.1 Test case I - Carslaw and Jaeger	63
4.3.2 Test case II - Rathjen and Jiji	69
4.3.3 Test case III - Tao	73
4.4 A casting problem	75
4.5 Assessment/Conclusion	82

4.6 Closure	84
Figures	85
Chapter 5	108
5.0 The full convective-diffusive solidification model	110
5.1 Introduction	110
5.2 Governing equations for fluid flow and heat transfer	111
5.2.1 Momentum equations	112
5.2.2 Continuity equation	113
5.2.3 Energy equation	113
5.2.4 The general conservation equation	113
5.3 Domain discretisation	114
5.4 Discretisation of the governing equations with solidification ..	115
5.4.1 General equation	115
5.4.2 Momentum equations	118
5.4.3 Energy equation	121
5.4.4 Continuity equation	122
5.4.5 Pressure-correction equation	123
5.5 Boundary conditions	125
5.5.1 Estimation of the boundary pressure.	126
5.5.2 Pressure correction gradient.	128
5.6 Solution procedure	129
5.7 Non-staggered grid	130
5.7.1 The Rhie and Chow Interpolation method.	131
5.8 Closure	134
Figures	135

Chapter 6	138
6.0 Convective-diffusive solidification - validation and results	140
6.1 Introduction	140
6.2 The solution algorithm	141
6.3 Fluid flow and heat transfer bench mark cases	143
6.3.1 Case I - Moving lid cavity	144
6.3.1.1 The problem	144
6.3.1.2 Results with quadrilateral cells	144
6.3.1.3 Results with non-quadrilateral cells	148
6.3.2 Case II - Natural convection driven cavity	152
6.3.2.1 The problem	152
6.3.2.2 Results with quadrilateral cells	154
6.3.2.3 Results with non-quadrilateral cells	158
6.4 Solidification with both convection and diffusion	161
6.4.1 Case I - Melting of pure gallium in rectangular cavity	161
6.4.1.1 The problem	161
6.4.1.2 Results	163
6.4.2 Case II - A demonstration of an integrated approach	165
6.4.2.1 The problem	165
6.4.2.2 Results	167
6.5 Closure	169
Figures	171
 Chapter 7	 234
7.0 Conclusions and Suggestions for further work	235
7.1 Conclusions	235
7.2 Suggestions for further work	238
7.2.1 Generating orthogonal high-order cells	238

7.2.2	Partial staggering to remove pressure oscillation	239
7.2.3	Cross-diffusion term for non-orthogonal meshes	241
7.2.4	Equation solvers	241
7.2.5	Extension to three-dimensions	242
	Figures	243
	References	245

Chapter 1

1.0 Introduction	2
1.1 Objectives	2
1.2 Practical relevance	2
1.3 Background information	5
1.4 Present contribution	8
1.5 Outline of present work	10

1.0 Introduction

1.1 Objectives

In this study, the principle aim is to examine and to develop a suitable control volume-unstructured mesh (CV-UM) method in context of numerical simulation of solidification for the casting processes. This has wider applications, since the study hopes to find or to develop a method that can handle all the major components of the casting process; the mould filling, heat transfer, solidification, residual stress development and deformation of cast. All these components have previously been solved independently and by a variety of methods. It is hoped that a method based on the control volume approach can deal with all the processes mentioned simultaneously and with the same qualities that have been shown by the individual methods.

The underlying goal of this research is to extend the scope of the control volume method to permit handling of finite element (ie. unstructured) meshes, necessary in dealing with complex geometries, prevalent in industrial castings.

1.2 Practical relevance

Over the last twenty five years, with the emergence and rapid development of ever more powerful digital computers, the field of Computational Fluid Dynamics (CFD) has advanced to a stage that, in some engineering applications, it has become a standard and

an invaluable research and design tool for engineers. Engineering problems such as those encountered in the aerospace, automobile, fire research, nuclear power, metal foundry, is just a few aspects where industry is using CFD to improve component design that is safety-critical, by making it more reliable and cost effective to manufacture. The advantages of numerically simulated experiments over full-scale experimental investigations are :-

i) The human and financial resources required to carry out a full-scale experiment can be extremely expensive. Take the case of fire research, it ranges from around £10,000 for one simple full scale compartment to around £100,000 for a series of full scale compartment fire tests. In safety-critical castings for the aerospace industry, the investment put in to the design and the casting of the first component ranges from around £5,000 for one that is small and simple in shape to around £500,000 for one that is large and complicated. Unfortunately, even then, there is no guarantee that the cast produced will meet the required standard as specified.

ii) In some engineering applications, experimentation is generally not feasible. This is due to one or a combination of the following factors, geometric complexity, scale, interaction of different reactive processes and most importantly the safety aspect of the experiment. One such an example is in the prevention of nuclear accidents. The various scenarios for an accident in and around a nuclear power station need to be looked at and removed for safety reasons. With experimentation out of the question, CFD simulation becomes the only choice.

CFD deals with the solution of discretised forms of the partial differential equation's

representing transport of mass, momentum and energy. This discretisation takes place in a solution grid or mesh which divides the solution domain into small control volumes. Various methods of solution of the discretised equation and different grid philosophies have been developed over the years; the two that have emerged pre-eminent are the Finite Volume (FV) or Control Volume (CV) and Finite Element (FE) methods. Both have their advantages and disadvantages. When it comes to topological consideration finite elements are seen to have the edge while for highly non-linear flows involving turbulence and/or combustion, control volume methods are preferable. The computational cost between the two methods is generally in the ratio of three to one in favour of the control volume.

Recently, there has been clear evidence that both the methods are moving toward a common ground. Algorithms and techniques developed previously solely for use by one method is now being adapted and employed in a similar procedure by the other. Therefore, in the not-too-distant future it will be difficult to distinguish the two with the cross-fertilization currently taking place. This has already led to hybrid methods and the development of generic algorithms that can be implemented in either of the methods as they stand.

A good example of this cross-fertilization is in the casting processes. The integration of the mould filling, heat transfer, solidification and residual stress development processes, previously solved independently and by a variety of methods is fast becoming a reality, all coming under the umbrella of a hybrid method. In finite elements, the

advancement has been from the solution of stress and working towards the solution of flow. Commercial stress code such as ANSYS by SAS [ANSYS] can do fluid flow calculations seperated from stress, while FE codes such as FIDAP by FDI [FIDAP] and FEAT by NE [FEAT] has been developed for fluid flow analysis. In control volumes, the advancement has been in the opposite direction, ie. flow to stress. Commercial codes such as PHOENICS by CHAM [PHOENICS] and HARWELL-FLOW3D by AEA [FLOW3D] are very efficient doing CFD calculations. And there are modules being developed to bolt into these codes for doing stress as part of the combined CFD calculations. For stress only calculations, there is currently no FV or CV formulated code out in the market. This may change in the future with the advancement currently taking place for the FV or CV formulation for stresses, especially in non-linear problems. Along the way towards the ultimate goal, each method has taken ideas from the other in moving towards a solution for a process that is naturally coupled.

1.3 Background information

In the last fifteen years or so, we have seen a focus upon Finite Volume (FV) or Control Volume (CV) approach to the solution of fluid flow and heat transfer problems or Computational Fluid Dynamics (CFD) as it is more generally known. This success is mostly due to the conservation principles as in the balance of flows, whether diffusive, convective or other means that the terms in the resulting algebraic equations have a specific physical interpretation. The straight forward formulation and low computational

cost compared with other methods have made the CV the preferred choice for most CFD practitioners. This coupled with the robustness of the SIMPLE (semi-implicit pressure-linked equation) solution algorithm of Patankar and Spalding (1972) have catapulted the method to the forefront of CFD study.

The major weakness with the method is in mapping complex geometries. This is due to the control volume being rectangular in two-dimensions and cuboid in three-dimensions. The original method can only represent curved boundaries in a very crude "step-like" fashion. In majority of cases, mesh refinement and partial blockages would be sufficient in reduce the inaccuracy to an acceptable level. This would inevitably increase the computational overheads but compared with other numerical methods the increase in computer time remains small.

In recent years, several control volume based-unstructured mesh (CV-UM) approaches have in some ways overcome the structured nature of the control volume method. The simplest, is the block-structured approach. By building the irregular shape in blocks, the blockages previously used have been removed and complicated shapes can be represented. For curved boundaries the problem remains. The works by Gordon and Hall (1973) and Thompson et al (1974, 1982) overcame the problem by mapping the physical domain to a solution domain that is regular using the curvilinear co-ordinate transformation. This is generally referred as Curvilinear Grids (CG) or Body Fitted Coordinates (BFC). Combining this with the block-structure offers a solution to the problem of mapping irregular shapes. This joint approach has proved successful in

solving a variety of engineering problems and is available in most of today's commercial CFD codes such as PHOENICS by CHAM [PHOENICS] and HARWELL-FLOW3D by AEA [FLOW3D] for dealing with irregular geometries. With the method retaining the structured approach of the original CV method, local mesh refinement is generally not possible. The method can be regarded as a compromise or a half way house towards a truly unstructured mesh approach.

Another approach towards an unstructured mesh has been to combine some aspects of the finite element method. By combining the best of both worlds, finite element and control volume, a hybrid method generally known as the Control Volume based-Finite Element mesh (CV-FE) method comes to existence. The basic idea behind the method is to discretise the calculation domain into elements, triangles and/or quadrilaterals, just like in finite element calculations. A control volume is then formulated around the vertices of the element and the conservation principles applied to it. This approach was first used for electromagnetic calculations in two-dimensions by Winslow (1966), and later for fluid flow and heat transfer by workers such as Baliga and Patankar (1980, 1983); Schneider and Raw (1986) and Lonsdale and Webster (1989) are just a few of the CFD practitioners actively involved with the advancement of the method. Commercial CFD codes such as ASTEC from AEA [ASTEC] use this technique for their numerical computations.

The CV-FE technique provides a solution to the unstructured mesh problem associated with the control volume method. In the combining process with finite elements, some

qualities of the original CV method have been lost. For example, the formulations have become more complicated. The elements-control volume relationship increases the complexity in the convection term of the transport equations. Diagonal dominance of the discretised equations is not that readily apparent and the computational costs had more than doubled. In spite of these drawbacks, this is currently the only method that offers a truly unstructured mesh approach to control volume based algorithms.

1.4 Present contribution

In this study, the development of a new CV formulation that can handle irregular (unstructured) meshes for convective-diffusive solidification in shape casting has been derived. The Irregular Control Volume method (ICV) as it will be called, uses the SIMPLE solution procedure of Patankar and Spalding (1972) for the simulation of fluid flow and heat transfer is adopted. To suppress pressure oscillations, a non-staggered grid arrangement is adopted and the Rhie and Chow (1982) flux interpolation is used.

The new method is ideally suited to casting simulations for all major components involved in shape casting; mould filling, heat transfer, solidification, residual stress development and deformation of cast. Therefore, making the integration of all the processes a reality, as demonstrated under an on going ACME project [ACME]. A detailed description of the component integration procedure can be found in the paper by Bailey, Cross, Chow and Pericleous (1992).

One added bonus of the ICV is the appearance of "high-order cell" control volumes, where the control volumes can have more than the standard four cell faces in two-dimensions and more than six in three. The results obtained with the technique so far have showed the same characteristics as one would expect in high-order schemes employed in the normal CV method, where high-order scheme requires discretisation of the governing equations.

In the present study, both cell-centred and the vertex-centred approaches have been examined and compared for their suitability in the simulation of solidification by conduction only in the casting environment, using two-dimensional finite element meshes that consist of triangles and quadrilateral elements. In the vertex-centred approach, the material interface needs to be handled correctly; otherwise unrealistic solutions can result. The possible techniques to overcome the problem are given in *Chapter 5*. For the cell-centred case, the method is more sensitive to the orientation or quality of the mesh. With regards to mesh generation, currently, all automatic mesh generators are based around finite elements made up of triangles and/or quadrilaterals. Automatic generation of orthogonal high-order cells is not yet available. The "circle-cell" idea has been put forward in this study as a possible solution to the generation of orthogonal high-order cells. Work needs to be done to confirm this idea.

In fluid flow calculations, the Rhie and Chow (1982) velocity interpolation technique is effective in suppressing pressure oscillations. However, where a boundary pressure needs to be prescribed in the calculations, care needs to be taken. It was found that

approximating the boundary pressure can sometimes have a profound influence on the solution if forward/backward differencing is used, the Bernoulli condition may not necessarily be satisfied. This also applies in approximating the boundary pressure-correction value. A better and more consistent method in evaluating the pressure-correction value, based on the Bernoulli approximation of the boundary pressure has been derived, and has shown a significant improvement over the forward/backward differencing. Still the problem with the correct boundary pressure persists; the solution may lie with the partially staggered approach, since a fully staggered arrangement is not a viable option in irregular meshes.

The new ICV formulation has been derived from the need to simulate complex geometries associated with shape casting, namely convective-diffusive solidification in this study. It is hoped the new method will provide the unstructured mesh option, previously unavailable to the control volume CFD practitioner, in his or her work to tackle the complex geometric requirements in today's CFD simulations.

1.5 Outline of present work

The solidification algorithm used in this study originates from the Voller and Cross (1985) enthalpy based method. The fundamental description of the algorithm in relation to the CV method is presented in *Chapter 2*, where the study of vectorisation of the solidification algorithm is reported. The amount of computational time that can be

gained by vectorisation and/or parallelisation in solidification calculations can be significant, as much as five times faster than the serial, by just vectorisation alone. This work was conducted in a joint study on parallelisation and vectorisation of solidification algorithms [Cross, Johnson and Chow (1989)].

In *Chapter 3*, the control volume unstructured mesh method is described and classification of the two different approaches, the vertex-centred and cell-centred are illustrated. The vertex-centred approach has been successfully employed in fluid flow and heat transfer modelling. For solidification, the current study is the first to use this unstructured approach in control volume formulations. The cell-centred approach has previously been associated only with structured meshes. A method based on the cell-centred for unstructured meshes was developed during the this study. Irregular control volume (ICV) as it is referred to in this thesis, can be viewed as an extension to the classical CV method.

Several test cases were used to validate the two approaches for solidification before a real casting problem was used with physical properties that are dependent on temperatures. *Chapter 4* shows the results obtained using both methods. Their suitability for solidification and advantages and disadvantages is fully reported in the chapter.

Chapter 3 only described the ICV method for heat conduction. The full convection-

diffusion coupling is explained in *Chapter 5* in two-dimensions, using the SIMPLE solution algorithm of Patankar and Spalding (1972). The discretisation of all the governing equations such as, the momentum, pressure-correction and the energy are derived. Also highlighted in the chapter are the components needed for convective-diffusive solidification.

Two test cases were used to validate the ICV method for fluid flow and heat transfer. The results of these using both structured and unstructured meshes are in first half of *Chapter 6*. In this, the result of the high-order cells used shows the same characteristics as the high-order schemes used for structured mesh result as reported in the literature. The high-order schemes require rediscretisation of the equations for each of the scheme. No such rediscretisation is necessary for the various high-order cells.

In the second half of *Chapter 6*, the results of two solidification problems are reported. The first concerns the experiment performed by Gau and Viskanta (1984) for the melting of pure gallium for which experimental data are available. The second problem is a demonstration of the coupling currently being developed under the ACME project for the convective-diffusive solidification, residual stress and deformation all under a single code.

Finally, conclusions and suggestions for future work are presented in *Chapter 7*, with references and other auxiliary material completing this thesis.

Chapter 2

2.0 Mapping a Solidification Algorithm onto Vector Architectures	14
2.1 Introduction	14
2.2 Enthalpy algorithm for solidification	14
2.3 Scalar algorithm	16
2.4 Solution procedure for the scalar algorithm	18
2.5 The model problem	19
2.6 Overview of the Masscomp 5400 system	20
2.7 The vectorised algorithm	22
2.8 Results and discussion	24
2.9 Closure	27
Figures	29

2.0 Mapping a Solidification Algorithm onto Vector Architectures

2.1 Introduction

In this chapter, the mapping of a popular control volume enthalpy-based algorithm by conduction only in three-dimensions onto a vector computer [Masscomp (1984)] will be reported. The mapping of numerical algorithms was part of a collective investigation into the feasibility and the associated benefits of putting it onto vector and parallel machines. The parallel part of the mapping will not be reported here, but is in the joint publication by Cross, Johnson and Chow (1989).

A model problem with data relevant to solidification in casting was designed for use as a vehicle for exploring the problem associated with the mapping of numerical algorithms onto non-scalar machines.

Described within are the solidification algorithm to be mapped, the discretisation of the governing equations and the scalar and vectorised solution procedures. Also referenced is the worthiness of the vectorisation option compared with parallelisation in such application as solidification.

2.2 Enthalpy algorithm for solidification

The solidification algorithm taken for the mapping onto vector computers is the one

based on the popular enthalpy-based method, those of Voller and Cross (1985, 1987).

This algorithm is mathematically detailed below:

The solidification equation by conduction only is given as

$$\frac{\partial(\rho h)}{\partial t} = \nabla \cdot (k \nabla(h/c)) - L \frac{\partial(\rho f)}{\partial t} \quad 2.1$$

where the total enthalpy, H , is

$$H = h + \Delta H \quad 2.2$$

where h is the sensible enthalpy and is given as

$$h = \int_{T_{\infty}}^T c \, dT \quad 2.3$$

and the latent heat of solidification is expressed as

$$\Delta H = Lf \quad 2.4$$

L is the latent heat of the material and f is the liquid fraction. The liquid fraction, f at a temperature T may, for example be expressed by the relationship

$$f = \begin{cases} 1 & T_l < T \\ (T - T_s) / (T_l - T_s) & T_s < T < T_l \\ 0 & T < T_s \end{cases} \quad 2.5$$

where T_s and T_l are the solidus and liquidus temperatures of the material. Other possible liquid fractions representation can be found in the publication by Voller and Swaminathan (1991).

Equations 2.1 to 2.5 together with suitable initial and boundary conditions defines the process of solidification by conduction only.

2.3 Scalar algorithm

Using the control volume (CV) method as described by Patankar (1980). The solidification equation 2.1 is integrated over an arbitrary cubical control volume as illustrated in *Figure 2.1* and *Figure 2.2* which yields

$$\begin{aligned} \iiint_V \frac{\partial}{\partial t}(\rho h) dV &= \iint_A k \frac{\partial}{\partial x}(h/c) dA + \iint_A k \frac{\partial}{\partial y}(h/c) dA \\ &+ \iint_A k \frac{\partial}{\partial z}(h/c) dA - \iiint_V L \frac{\partial}{\partial t}(\rho f) dV \end{aligned} \quad 2.6$$

Therefore, the discretised form of equation 2.6 at a control volume P is

$$\begin{aligned} \frac{(\rho h)_P - (\rho h)_P^o}{\Delta t} \Delta x \Delta y \Delta z &= \left[k_e \frac{\left(\frac{h}{c}\right)_E - \left(\frac{h}{c}\right)_P}{\delta x_e} - k_w \frac{\left(\frac{h}{c}\right)_P - \left(\frac{h}{c}\right)_w}{\delta x_w} \right] \Delta y \Delta z + \\ &\left[k_n \frac{\left(\frac{h}{c}\right)_N - \left(\frac{h}{c}\right)_P}{\delta y_n} - k_s \frac{\left(\frac{h}{c}\right)_P - \left(\frac{h}{c}\right)_s}{\delta y_s} \right] \Delta x \Delta z + \\ &\left[k_t \frac{\left(\frac{h}{c}\right)_T - \left(\frac{h}{c}\right)_P}{\delta z_t} - k_b \frac{\left(\frac{h}{c}\right)_P - \left(\frac{h}{c}\right)_B}{\delta z_b} \right] \Delta x \Delta y - L \frac{(\rho f)_P - (\rho f)_P^o}{\Delta t} \Delta x \Delta y \Delta z \end{aligned} \quad 2.7$$

and equation 2.7 can be rewritten more compactly as

$$a_p h_p = \sum_{A=N,E,S,W,T,B} a_A h_A + b \quad 2.8$$

where

$$\begin{aligned} a_E &= k_e \frac{\Delta y \Delta z}{\delta x_e} & a_W &= k_w \frac{\Delta y \Delta z}{\delta x_w} \\ a_N &= k_n \frac{\Delta x \Delta z}{\delta y_n} & a_S &= k_s \frac{\Delta x \Delta z}{\delta y_s} \\ a_T &= k_t \frac{\Delta x \Delta y}{\delta z_t} & a_B &= k_b \frac{\Delta x \Delta y}{\delta z_b} \end{aligned}$$

$$a_p = \sum_{A=N,E,S,W,T,B} a_A + \frac{\rho_p}{\Delta t} \Delta x \Delta y \Delta z$$

$$b = \frac{(\rho h)_p^o}{\Delta t} \Delta x \Delta y \Delta z - L \frac{(\rho f)_p - (\rho f)_p^o}{\Delta t} \Delta x \Delta y \Delta z$$

To complete the formulation, boundary conditions can be defined as follows;

i) Given boundary temperature (Dirichlet)

$$k \frac{\partial T}{\partial n} = k \frac{T_p - T_{Bdy}}{\delta n}$$

ii) Given boundary heat flux (Neumann)

$$k \frac{\partial T}{\partial n} = q_{Bdy}$$

For a convective boundary condition, $q_{Bdy} = H(T_p - T_\infty)$, where H is the heat transfer coefficient and T_∞ is the temperature of the surrounding fluid.

2.4 Solution procedure for the scalar algorithm

The main computational steps for the solidification algorithm are given below:

1. Set h_p^o , f_p^o and ρ_p^o to initial or previous time values.
2. Evaluate coefficients of matrix $[A]$ and vector \mathbf{b} of equation 2.7.
3. Solve for enthalpy h_p .
4. Calculate temperature T_p and liquid fraction f_p from h_p .
5. If T_p and f_p have converged, then advance a time step; otherwise, go to step 2.

The set of equations in the form of 2.7 can be assembled and represented in the more general matrix form, $[A]\mathbf{h} = \mathbf{b}$, which can then be solved using any suitable solver. The Successive Over-Relaxation (SOR) technique for solving sets of equations was selected for use in the scalar algorithm. Once the new enthalpy values have been obtained, the nodal temperature value is then evaluated from equation 2.3 using the nodal enthalpy and specific heat values. If the specific heat, c , is itself a function of temperature, an iterative procedure is required in the evaluation. The liquid fraction at a node P , f_p , involves a correction update using equation 2.5. For materials with a small mushy region, or for more sophisticated approach, Voller and Swaminathan (1991) offers a few alternatives.

The criteria used for convergence are that both the changes in temperature and liquid fraction fall below a set tolerance level between two successive iterations.

Temperature criterion

$$\frac{\|T_p^i - T_p^{i-1}\|}{\text{MAX} [\|T_p^i\|, \epsilon]} \leq \zeta \quad 2.9$$

Liquid fraction criterion

$$\|f_p^i - f_p^{i-1}\| \leq \zeta \quad 2.10$$

where ζ is the tolerance level required and ϵ is a very small value for preventing division by zero.

Because of the high non-linearity of the equations, some under-relaxation is necessary to avoid divergence in the iterative process. The relaxation employed here is a linear under-relaxation of the form

$$\phi^{new} = \phi^{old} + \alpha_\phi (\phi^{new} - \phi^{old}) \quad 2.11$$

where α_ϕ is the relaxation value, for under-relaxation this value is less than one. For values greater than one over-relaxation occurs and with the value of one no relaxation is exercised.

2.5 The model problem

The model problem [Walthier (1984)] is a simple 40cm cube of a metal that is initially held at 1632°C. The metal simulated is AISI 1086 steel and its essential properties are

given below:

Specific heat ,	$c_p(T) = 0.105 + 1.08667E-04 T$	$J/kg^{\circ}C$
Conductivity,	$k(T) = 0.01203 - 6.9647E-06 T$	$W/m^2^{\circ}C$
Density,	$\rho(T) = 7853.08 - 0.3229 T$	kg/m^3

The latent heat of steel is taken as $66 J/kg$ and the solidus and liquidus temperatures are $1508^{\circ}C$ and $1602^{\circ}C$ respectively. The boundary conditions are simple radiative conditions in which the ambient temperature is assumed to be $25^{\circ}C$ and the heat transfer coefficient is fixed at $4.7766E-03 W/m^2s^{\circ}C$.

2.6 Overview of the Masscomp 5400 system

A simple illustration of the Masscomp architecture [Masscomp (1984)] is shown in *Figure 2.3*. The host processor is a 68020 processor with two megabytes of RAM and an eighty megabytes hard disk with a UNIX operating system. It possesses standard FORTRAN-77 compiler with a library of function calls that enable the user to control the operation of the Vector Processor (VP). The processor's speed for both the host and VP are 0.06 and 0.3 megaflop respectively. The VP itself has a simple architecture with an on-board memory unit, a pipeline arithmetic unit, and a memory management unit. The on-board VP memory is crude; it has 32,000 32-bit locations, and the user is directly responsible for the management of data through a set of subroutine calls. To

use the VP, the user has to

1. Grab the vector processor.
2. Load the vector(s) or scalar into the vector memory and, in so doing, prescribe the location.
3. Perform some mathematical operation through the arithmetic unit and put the result back to the vector memory.
4. Write the result back to the host processor.

If the total problem size is small enough, the entire problem with any working spaces can all be fitted into the VP memory. Therefore, it is worthwhile in using the low level functions provided by the manufacturer [Masscomp (1984)]. The reason being, a load or a store operation between the host and VP memory takes 8 to 10 times as long as a mathematical operation in the arithmetic unit.

For large problems, the VP memory can often be fully utilised in performing just one vector operation. To make the vector processor more accessible for the large problems in which we are interested, a set of high-level routines has been developed and was made available to this study by Ierotheou and Cross (1987) that will perform the same operations as the low level routines. With the task of managing the data transfer and management operations optimised, the whole process is totally transparent to the user. The high-level routines were used in this investigation.

$$h^{r+1} = b - \sum_{A=T,N,E,W,S,B} (a_A h^r)_A \quad 2.14$$

where

$$a_A(i) = a_A(i) / a_p(i)$$

$$a_A = (a_{A,1}, a_{A,2}, \dots, a_{A,n})$$

$$a_A h^r = (a_{A,1} h^r_1, a_{A,2} h^r_2, \dots, a_{A,n} h^r_n)$$

The $()_A$ is the shift index value for the vector summation, for example,

$$t + s_E = (t_2 + s_1, t_3 + s_2, \dots, t_n + s_{n-1})$$

$$t + s_W = (t_1 + s_2, t_2 + s_3, \dots, t_{n-1} + s_n)$$

where $s_A = (a_A h^r)_A$ and t is the temporary vector for the summation.

In the solution procedure, the process that is most computationally demanding is the solution solver. By efficiently vectorising the solver, the overall time for the computation is significantly reduced. Other vectorisable processes are; the evaluation of conductivity, k , specific heat, c , and temperature, T , and the processes involved in initial set-up of the problem were all vectorised. In the case of calculating liquid fractions, the procedure involved cannot be fully vectorised. It requires conditional vector operations that are not available on the Masscomp but are available in other vector processors.

The full vectorisation in computing the coefficients of a_A and b is not a profitable venture, owing to the overwhelming data transfer overheads on short vectors. If full vectorisation was to proceed, not only would it prove non-profitable, it would result in a computationally loss-making venture (an increase in computing time). For the Masscomp, the "break-even" point for vector operations varies with the mathematical operations. The average break-even length is eighty elements. Below are just some vector operation break-even lengths.

Vector Operation	Vector break-even length
Reciprocal	Greater than 70 elements
Multiply	Greater than 70 elements
Square Root	Greater than 70 elements
Copy	Greater than 100 elements

A complete break down of all the vector operations and its break-even length is given in the report by Ierotheou and Cross (1987).

2.8 Results and discussion

The model problem described above actually takes about 320 minutes in real time to achieve full solidification. The time-step used for all the simulations was 60 seconds. All the simulations were performed on the Masscomp system and the comparison for

speed-up between scalar and vector are based on the optimal method in each. The biggest scalar simulation achieved was 15,625 nodes (25^3) and for vector it was 17,576 nodes (26^3). The size of problem achieved may seem small compared with the parallel work done on the same problem by Steve Johnson [Cross, Johnson and Chow (1989)]. This is entirely due to the Masscomp computer having a maximum of two megabytes of on-board memory and is the only limiting factor on the size of the problem possible.

Nodes	Serial JOR	Serial SOR	Vector JOR
5^3	0.094	0.080	0.088
10^3	0.975	0.595	0.252
15^3	4.455	2.252	0.824
20^3	11.771	6.043	2.053
25^3	28.271	14.060	4.687

Table 2.1 CPU times of scalar JOR, scalar SOR and vector JOR.

Figure 2.4 shows the comparison between the scalar SOR, scalar Jacobi and vectorised Jacobi. Their CPU times are in *Table 2.1*. As expected, the SOR solver is optimal in scalar and is a factor of two faster than scalar Jacobi. But in vector, the vectorised Jacobi is a factor of three faster than the scalar SOR. Note, the break-even point for the problem lies somewhere just above 125 nodes (5^3). As the computational time indicated at 125 nodes, see *Table 2.1*, the scalar SOR is slightly faster than the vectorised Jacobi.

A speed-up of three is encouraging, and will be higher for larger problems as the graph in *Figure 2.4* shows no sign of levelling off. A look at the communication overhead revealed, some 50% of the total execution time is involved in exchanging data between the host and the vector memories. If this could be minimized and assuming some 20% of the calculation being scalar [Ierotheou, Richards and Cross (1989)], the vector to scalar speed-up could be of order just under five. The speed-up may be further enhanced if the vector to scalar performance is increased from 5:1. However, the inherent limitation in speed-up probably has more to do with a combination of the following factors;

- i) The fact that solution of the linear equation for h , the enthalpy vector, converges very quickly at each stage in the iterative process.

- ii) The essentially scalar nature of the calculation procedure for f and the coefficients of a_A and b .

than with the vector to scalar performance ratio. The limitation in the speed-up to less than five is confirmed when the breakdown of the computational effort is plotted as show in *Figure 2.5*. The evaluations of a_A , b and f are essentially scalar. This means that almost 20% of the calculation procedure is scalar, and so the speed-up factor cannot be greater than five, a finding that is consistent with those of Ierotheou, Richards and Cross (1989).

As part of a collective investigation into mapping enthalpy-based solidification algorithms onto vector and parallel machines [Cross, Johnson and Chow (1989)], the part conducted by Steve Johnson in parallelising the solidification algorithm proved the solution time increases in a roughly linear manner. This result is more encouraging because it confirms that the limitation in computational performance has more to do with the number of processors available than with any significant constraints with essentially scalar operations. From this, it seemed that vectorisation is not a worthwhile option compared with parallelisation in such an application as solidification. However, it is equally clear that scalar architectures with "bolt-on" vector accelerators can be used to speed up conventional solidification algorithms by a factor of up to five. It is emerging that the ideal architecture for many large scale numerical procedures is a network of processors, each of which has vector capabilities.

2.9 Closure

Computational speed-up is possible with vectorisation on numerical algorithms, as demonstrated in the mapping of the enthalpy-based solidification algorithm onto a vector machine (Masscomp). Where vast amount of cells (or nodes) are needed, (common in the process of solidification analysis) they can be stored in long vectors and operated on with vector operations.

In many numerical algorithms, there will always be some inherently scalar and/or non-

profitable elements for vectorising in the algorithm. This could be coefficient set-up as in the solidification case. These sort of essentially scalar operations limit the speed-ups of the vectorised algorithm to an increase of about five.

Vectorisation by itself may not be a worthwhile option compared with parallelisation. However, it is equally clear that a further increase in performance, a speed-up of five, can be had by vectorisation when included with the parallelisation, where each processor is attached to a vector processor of its own.

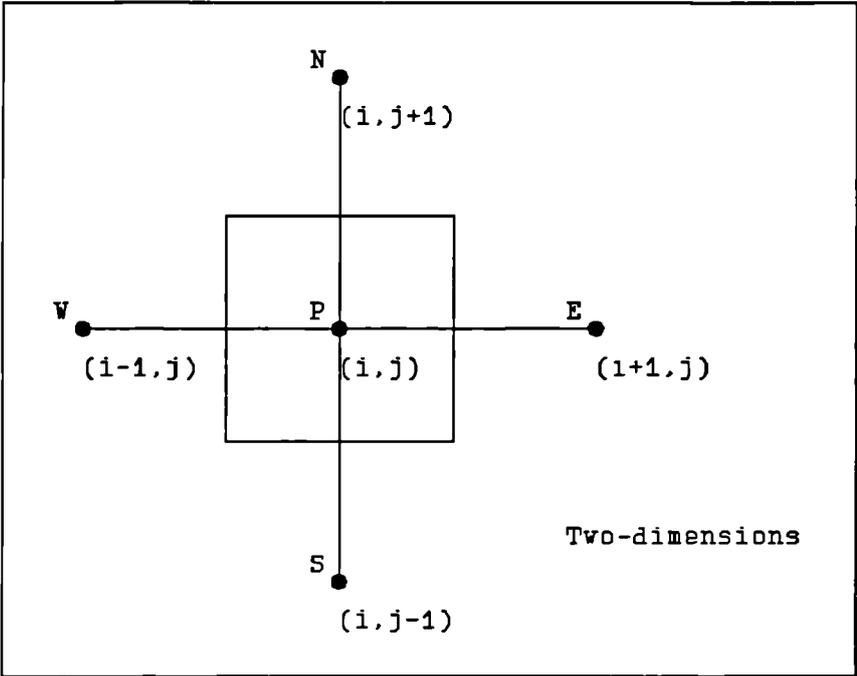


Figure 2.1 Two-dimensional topology.

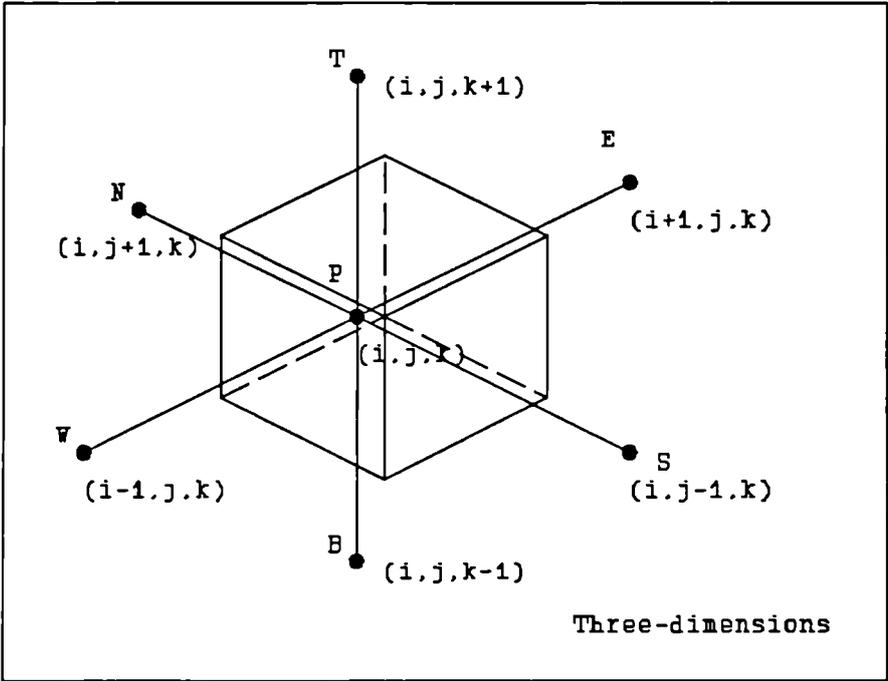


Figure 2.2 Three-dimensional topology.

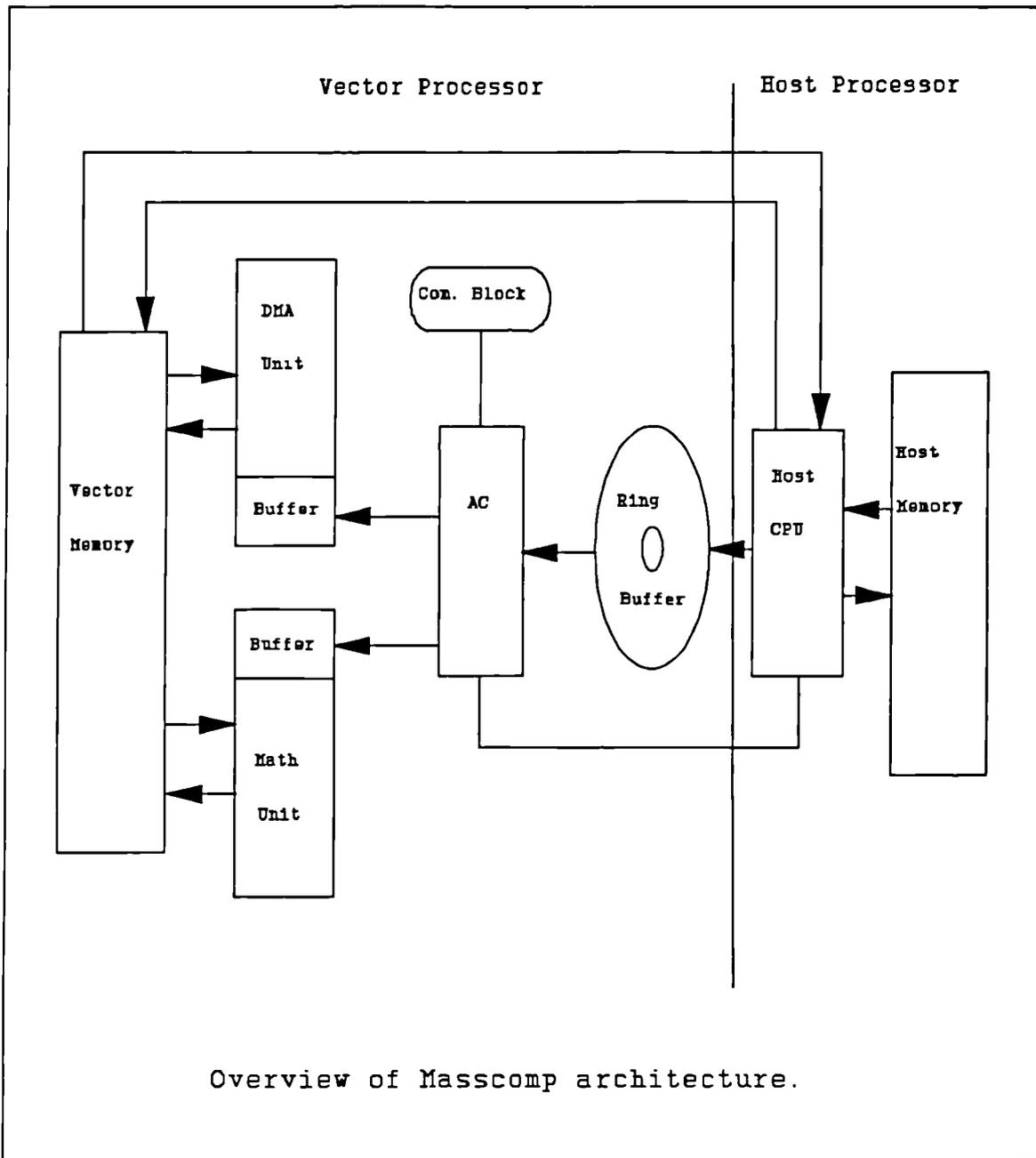


Figure 2.3 Masscomp architecture.

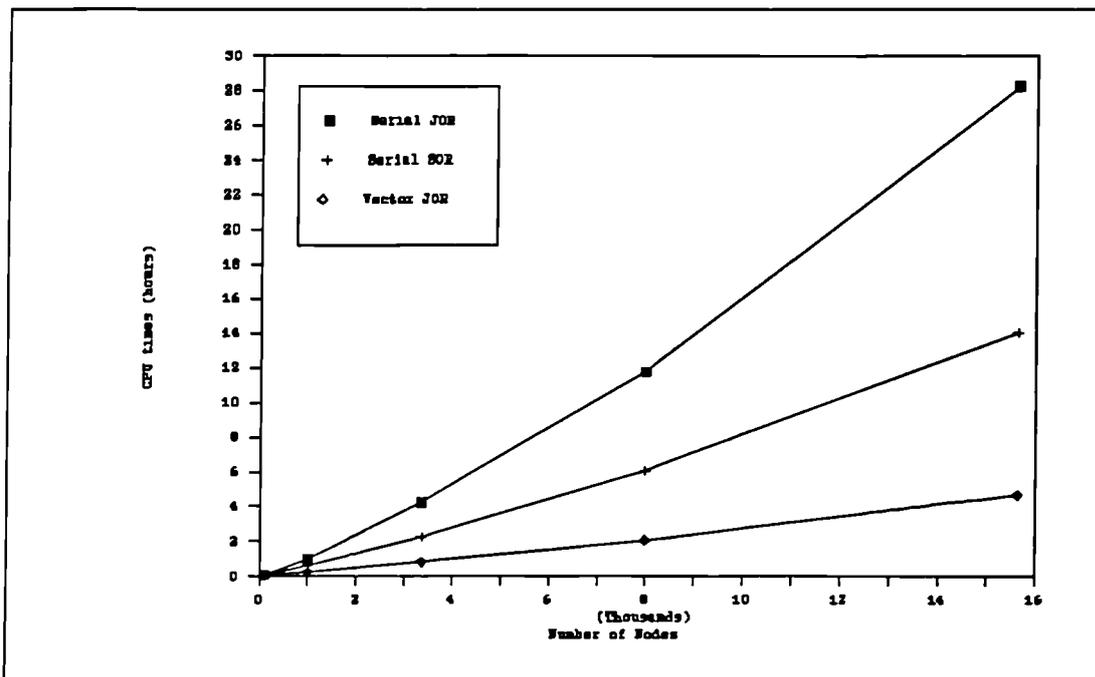


Figure 2.4 CPU comparison between vector and scalar.

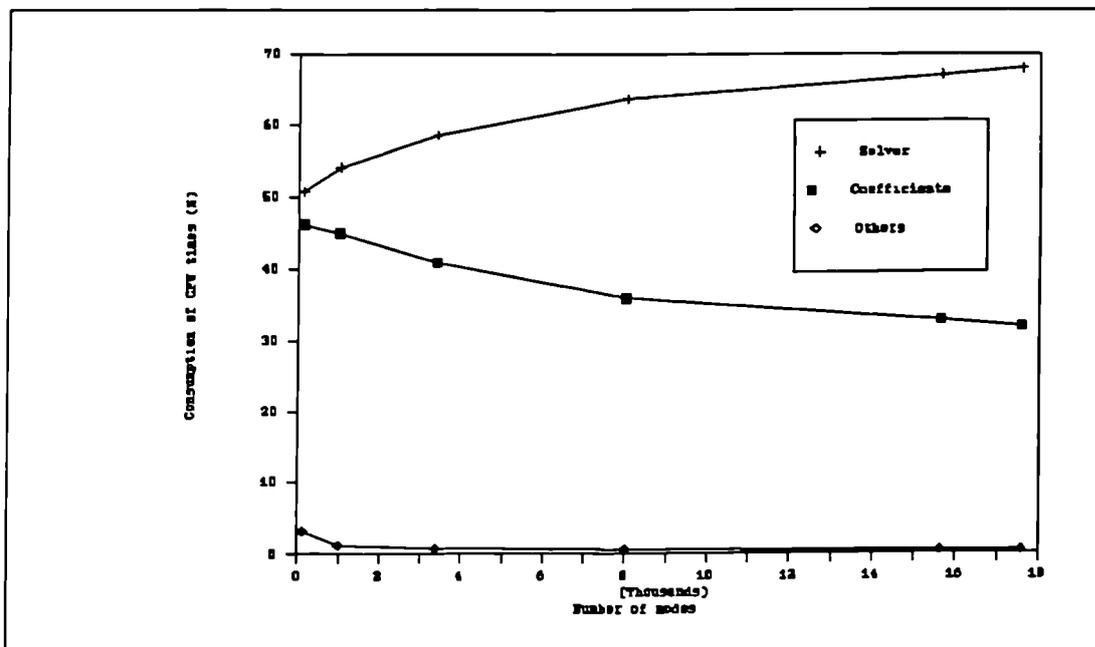


Figure 2.5 Breakdown of vector computational times.

Chapter 3

3.0 Control Volume Based Unstructured Mesh Methods	33
3.1 Introduction	33
3.2 Vertex-centred Approach	34
3.2.1 Domain discretisation	34
3.2.2 Local-global co-ordinate transformation	35
3.2.5 Boundary conditions	42
3.2.6 Multi-material	43
3.2.6.1 Weighted average	45
3.2.6.2 Thin elements	46
3.2.6.3 Coincident nodes	47
3.3 Cell-centred Approach	50
3.3.1. Domain discretisation	50
3.3.2 Discretisation of solidification equations	51
3.3.3 Boundary conditions	52
3.4 Closure	53
Figures	54

3.0 Control Volume Based Unstructured Mesh Methods

3.1 Introduction

In this chapter, the Control Volume based Unstructured Mesh (CV-UM) methods are to be described in context of solidification by conduction only. The solidification algorithm itself will not be addressed here since it has already been described in *Chapter 2* under *sections 2.2* and *2.4* of this thesis. In general, the CV-UM methods can be categorised into two approaches, a vertex-centred or a cell-centred approach. The classification of the approach is based on the relationship between the control volume and the finite element like unstructured mesh.

The first approach to be described is the vertex-centred or more generally known as the Control Volume Based-Finite Element Mesh (CV-FE) method by works such as Baliga and Patankar (1988), Schneider (1988), Lonsdale and Webster (1989), etc, who have been involved in the development of the method for fluid flow and heat transfer.

The second approach to be described is the cell-centred or as it is named, Irregular Control Volume method (ICV). It came to existence during this program of research. The method itself can be viewed as an extension of the control volume method [Patankar (1980)], since with a structured (rectangular) mesh it is the direct analogue of the standard control volume method.

Also described in this chapter will be the obstacles that each method need to address

when applied to solidification algorithms.

3.2 Vertex-centred Approach

In this sub-section, the vertex-centred or the control volume based-finite element mesh (CV-FE) method [Baliga and Patankar (1988), Schneider (1988), Lonsdale and Webster (1989)] is to be described for the solidification by conduction only in two-dimensions. The solidification algorithm and solution procedure are the ones described in *Chapter 2* under *sections 2.2* and *2.4* respectively. The focus here will be on the discretisation of the domain into an assemblage of elements, determining the necessary geometric transformation relations to relate element geometry to domain geometry, and defining the control volumes over which the conservation principle will be applied. Also attention is drawn to some problems that the vertex-centred method has to address when applied to solidification algorithms.

3.2.1 Domain discretisation

In a discrete solution procedure, the solution domain is subdivided into smaller regions and nodes are distributed throughout the domain, the connections between the nodes and the subregions is known as a mesh. In a finite element mesh, the subregions are called elements, with the vertices of the elements being the nodal locations.

There is a variety of element types possible for an FE mesh. For the vertex-centred approach only the basic elements were considered, four-noded quadrilaterals and three-noded triangles, either employing the basic element by itself or combination of the two. In this way, the full geometric flexibility associated with the FE method is retained.

Following the standard finite element methods, a typical FE mesh is shown in *Figure 3.1*, which represents the subdivision of the solution domain into much smaller domains, called finite elements. Nodes are located at every element corner and all of the problem unknowns are located at each node. For solidification, this includes enthalpy, temperature, liquid fraction, etc.; only the material properties of the element are held within it, for the simple reason that this makes it straightforward to ensure that internal boundaries are coincident with the vertices (characteristic of FE mesh).

3.2.2 Local-global co-ordinate transformation

In finite elements it is convenient to work in local co-ordinates so that each element may be treated identically (for each individual class of elements) irrespective of how distorted any element may actually be in terms of global co-ordinates. The local co-ordinate systems employed for triangle and quadrilateral elements are shown in *Figure 3.2*. Since the conservation laws are applied in terms of global co-ordinates, it is necessary to relate the local and global co-ordinates. This is done by using "shape

functions". The co-ordinate variation within an element is expressed in terms of local co-ordinates, so that if, for example, s is changed while t is held constant, then, in general, both x and y will change with a linear dependence on s for the case of the "bilinear" element considered here.

$$\begin{aligned} N_1(s,t) &= \frac{(1+s)(1+t)}{4} & N_2(s,t) &= \frac{(1-s)(1+t)}{4} \\ N_3(s,t) &= \frac{(1-s)(1-t)}{4} & N_4(s,t) &= \frac{(1+s)(1-t)}{4} \end{aligned} \quad 3.1$$

$$\begin{aligned} N_1(s,t) &= \frac{(1+2s)}{3} \\ N_2(s,t) &= \frac{(1-s+\sqrt{3}t)}{3} \\ N_3(s,t) &= \frac{(1-s-\sqrt{3}t)}{3} \end{aligned} \quad 3.2$$

Above are the relevant bilinear shape functions used, equation 3.1 is for quadrilaterals and equation 3.2 for triangles. So given x_i and y_i to be the global co-ordinates at local node i , the co-ordinate variation is conveniently described by

$$\begin{aligned} x(s,t) &= \sum_{i=1}^{N_p} N_i(s,t) x_i \\ y(s,t) &= \sum_{i=1}^{N_p} N_i(s,t) y_i \end{aligned} \quad 3.3$$

where N_p denote number of nodes for the element under consideration. Similarly with any variable ϕ , its variation within the element can be described by the same shape functions employed for the geometric description.

$$\phi(s,t) = \sum_{i=1}^{N_r} N_i(s,t) \phi_i \quad 3.4$$

Since the shape functions are continuous, the x and y derivatives of the variable ϕ can be differentiated to yield

$$\begin{aligned} \left. \frac{\partial \phi}{\partial x} \right|_{s,t} &= \sum_{i=1}^{N_r} \left. \frac{\partial N_i}{\partial x} \right|_{s,t} \phi_i \\ \left. \frac{\partial \phi}{\partial y} \right|_{s,t} &= \sum_{i=1}^{N_r} \left. \frac{\partial N_i}{\partial y} \right|_{s,t} \phi_i \end{aligned} \quad 3.5$$

Using the chain rule for partial derivatives, the x and y derivatives of the shape functions can be determined. In matrix form

$$\begin{bmatrix} \frac{\partial N_i}{\partial s} \\ \frac{\partial N_i}{\partial t} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} \\ \frac{\partial x}{\partial t} & \frac{\partial y}{\partial t} \end{bmatrix} \begin{bmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \end{bmatrix} \quad 3.6$$

where

$$\begin{aligned} \frac{\partial x}{\partial s} &= \sum_{i=1}^{N_r} \frac{\partial N_i}{\partial s} x_i & \frac{\partial x}{\partial t} &= \sum_{i=1}^{N_r} \frac{\partial N_i}{\partial t} x_i \\ \frac{\partial y}{\partial s} &= \sum_{i=1}^{N_r} \frac{\partial N_i}{\partial s} y_i & \frac{\partial y}{\partial t} &= \sum_{i=1}^{N_r} \frac{\partial N_i}{\partial t} y_i \end{aligned} \quad 3.7$$

The local derivatives of the shape functions are determined by differentiating equations 3.1 for quadrilateral and 3.2 for triangle elements. By solving the system in 3.6 the x

and y derivatives of the N_i can be determined.

$$\begin{bmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \end{bmatrix} = \frac{1}{J} \begin{bmatrix} \frac{\partial y}{\partial t} & -\frac{\partial y}{\partial s} \\ -\frac{\partial x}{\partial t} & \frac{\partial x}{\partial s} \end{bmatrix} \begin{bmatrix} \frac{\partial N_i}{\partial s} \\ \frac{\partial N_i}{\partial t} \end{bmatrix} \quad 3.8$$

where

$$J = \frac{\partial x}{\partial s} \frac{\partial y}{\partial t} - \frac{\partial y}{\partial s} \frac{\partial x}{\partial t} \quad 3.9$$

Equations 3.5 to 3.9 define all the steps necessary to calculate global x and y derivatives of ϕ in terms of local s, t co-ordinates and the nodal values.

It is also necessary to establish the transformation for a surface normal vector, as this will be needed to determine derivatives and integrals as they apply to a control volume. By considering the general line segment $(x,y)_a$ to $(x,y)_b$ shown in *Figure 3.3*, if this line segment represents part of an anticlockwise traversal of a control volume, then the vector Δs represents an outward normal for that segment and control volume.

$$\Delta s = \Delta y \hat{i} - \Delta x \hat{j} \quad 3.10$$

where

$$\begin{aligned} \Delta x &= x_b - x_a \\ \Delta y &= y_b - y_a \end{aligned} \quad 3.11$$

By using the chain rule, we have

$$\begin{aligned} dx &= \frac{\partial x}{\partial s} ds + \frac{\partial x}{\partial t} dt \\ dy &= \frac{\partial y}{\partial s} ds + \frac{\partial y}{\partial t} dt \end{aligned} \quad 3.12$$

therefore

$$\begin{aligned} \Delta x &= \int_a^b \frac{\partial x}{\partial s} ds + \int_a^b \frac{\partial x}{\partial t} dt \\ \Delta y &= \int_a^b \frac{\partial y}{\partial s} ds + \int_a^b \frac{\partial y}{\partial t} dt \end{aligned} \quad 3.13$$

If the variation of x and y is linear with respect to s and t , we have

$$\begin{aligned} \Delta x &= \left. \frac{\partial x}{\partial s} \right|_m \Delta s + \left. \frac{\partial x}{\partial t} \right|_m \Delta t \\ \Delta y &= \left. \frac{\partial y}{\partial s} \right|_m \Delta s + \left. \frac{\partial y}{\partial t} \right|_m \Delta t \end{aligned} \quad 3.14$$

where the subscript m denotes evaluation at the midpoint of the line segment, and where

$$\begin{aligned} \Delta s &= s_b - s_a \\ \Delta t &= t_b - t_a \end{aligned} \quad 3.15$$

3.2.3 Control-volume definition

In the solution domain, each node is associated with one control volume. Each surface of the control volume is defined from the centroid of the element to the midpoint of one

of its sides as shown in *Figure 3.4*. So each of the elements is divided into several quadrants (three for a triangle and four for a quadrilateral) by these control surfaces (CS). These quadrants are called sub-control volumes (SCV) and are illustrated in *Figure 3.5*. Therefore, a control volume is made up from the sum of sub-control volumes and is polygonal in shape, which can be assembled at the element level. Note that since the elements may have differing material characteristics a nodal control volume may consist of more than one material; we will return to this point later in a section that specifically deals with this problem. For now a nodal control volume can only be of one material.

For each control surface, the flows across it will need to be determined by an integral. These integrals will be approximated by the midpoint approximation for each control surface. To effect this midpoint approximation, the argument of the integrals is required at the midpoint of the control surface. These points will be referred as integration points (IP) and are illustrated in *Figure 3.6*. It is for these surfaces that the outward normal vector, described in previous section, will be required.

3.2.4 Discretisation of solidification equations

Integrating the solidification equation 2.1 in *Section 2.2* over an arbitrary control volume yields:

$$\iint_V \frac{\partial}{\partial t} (\rho h) dV = \int_S k \frac{\partial}{\partial x_i} (h/c) dn_i - \iint_V L \frac{\partial}{\partial t} (\rho f) dV \quad 3.16$$

where dn_i represents the components of the outward normal surface vector and an anticlockwise traversal control volume integration is assumed. Where $dn_1 = dy$ and $dn_2 = -dx$, so that the outward surface vector is $dn = dy \hat{i} - dx \hat{j}$. Therefore, the transient term in equation 3.16 is represented by

$$\iint_V \frac{\partial}{\partial t} (\rho h) dV = \frac{(\rho h)_P - (\rho h)_P^o}{\Delta t} V_P \quad 3.17$$

and the latent heat source term is represented as

$$\iint_V L \frac{\partial}{\partial t} (\rho f) dV = L_P \frac{(\rho f)_P - (\rho f)_P^o}{\Delta t} V_P \quad 3.18$$

where V is the area of the control volume, and is evaluated for the vertex case as

$$V_P = \sum_{i=1}^{Np_{scv}} V_{scv_i} \quad 3.19$$

where Np_{scv} is the total number of SCV's that make up the control volume associated with the node P . For the diffusion term, the variables within an element are represented by the use of local co-ordinate shape functions, see *section 3.2.3*. Therefore, since each element makes two surface contributions to the nodal control volume boundary, then

$$\begin{aligned}
\int_S k \frac{\partial}{\partial x_i} (h/c) dn_i &= \sum_{j=1}^{N_{pscv}} \left(\int_{SS_{,1} \rightarrow SS_{,2}} k \frac{\partial}{\partial x_i} (h/c) dn_i \right) \\
&= \sum_{j=1}^{N_{pscv}} \left(\sum_{i=1}^2 k_i^j \left(\sum_{m=1}^{N_{pt}^j} \left(\frac{\partial N_m^{lj}}{\partial x} \Delta x_i^j - \frac{\partial N_m^{lj}}{\partial y} \Delta y_i^j \right) (h/c)_m^j \right) \right)
\end{aligned} \tag{3.20}$$

where N_{pt}^j is the number of nodes associated with an element j .

Clearly, the algebraic expression in equation 3.17, 3.18 and 3.20 can be substituted into equation 3.16 to yield a set of equations in the form of

$$a_p h_p = \sum_{A=1}^{N_{adj}} a_A h_A + b_p \tag{3.21}$$

where N_{adj} is number of adjacent nodes to the node P and with the set of equations it can be solved with suitable initial and boundary conditions by the algorithm described in Section 2.4.

3.2.5 Boundary conditions

After assembly of the nodal control volume equations, complete conservation equations will exist for all interior control volumes. However, at solution boundaries, the corresponding control volume will have one or more control surfaces for which boundary conditions must be applied to complete the equations for conservation.

Figure 3.7 illustrates sub-control volumes for a quadrilateral element where either one or two of its sides form part of the solution domain boundary. In evaluating the boundary conditions along these sides, boundary integration points (BIP) and the corresponding boundary control surfaces (BCS) are needed as illustrated in *Figure 3.8*. The usual boundary conditions can now be applied.

- *Fixed value* : $\phi_P = \phi_{bdy}$ at domain boundary
- *Specified flux* : $\int_{BCS} \frac{\partial \phi}{\partial x_i} dn_i = q_{bdy}$

for convective boundary condition, $q_{bdy} = H(\phi_P - \phi_{ext})$, where H is the heat transfer coefficient and ϕ_{ext} is the external value.

3.2.6 Multi-material

As indicated in *section 3.2.4*, the material properties are stored within the elements so that internal boundaries are always associated with the element vertices. This means that to evaluate the liquid fraction, f , at a node P whose CV contains more than one material, then the liquid fraction of each SCV has to be stored and evaluated from

$$f_P = \frac{1}{V_P} \sum_{i=1}^{N_{scv}} (fV)_i \quad 3.22$$

where all the SCV liquid fractions are dependent on the nodal enthalpy value. Other variables that accounts for the multiple materials at a nodal control volume includes temperature, density, and latent heat. Their values at a node P are as follows:

Temperature

$$T_P = \frac{h_P}{V_P} \sum_{i=1}^{N_{scv}} \frac{V_i}{c_i} \quad 3.23$$

This equation is solved iteratively because the specific heat, c , is usually a function of temperature.

Latent heat source in equation 3.18

$$(L\rho)_P = \frac{1}{V_P} \sum_{i=1}^{N_{scv}} (L\rho V)_i \quad 3.24$$

Density

$$\rho_P = \frac{1}{V_P} \sum_{i=1}^{N_{scv}} (\rho V)_i \quad 3.25$$

It should be noted, apart from the extra calculations and storage required in evaluating the variables that are affected by the multiple materials, for heat conduction problems, the multi-material control volume encounter no problems.

This is not true when solidification is involved in the analysis, as in the case of casting. Where liquid metal and the sand mould can be co-inhabiting the same control volume, with a single liquid fraction, f , as representative mean of the liquid state inside the

control volume. It is this liquid fraction that gives rise to the physically unrealistic result. For example, an f value of 0.5, would indicate the control volume is part liquid and part solid. In the multi-material control volumes of liquid metal and sand mould, the representative f is indicating both the metal and sand is in a liquid or semi liquid state, for which the sand mould can never be in.

The full numerical effects of solving a real casting problem are reported in the next chapter. Here, we should be looking at ways of handling the inconsistency and their physical implications.

One way around the problem is to store all the SCV liquid fractions and evaluate a nodal liquid fraction as given in equation 3.22. Note, this nodal liquid fraction is not an accurate representation of the control volume liquid state. The alternative to this approach is single material control volumes.

3.2.6.1 Weighted average

The weighted average method, which assigns the material that is dominant in the control volume, is the simplest and the quickest to implement. Also it affixes no constraint on the FE mesh. A disadvantage of the technique is that there will always be a finite error incurred in the method even on very fine meshes.

The effects of this fault on the solution is either an under- or over-prediction depending on the assigned material to the control volume. This is understandable, since by taking the dominant material it has shifted the material boundary from its defined positions *Figure 3.9*. Thereby, the result obtained is also shifted by a significant amount, hence, the under- or over-predictions of the result. Apart from one-dimensional cases, the amount shifted is generally not possible to calculate in advance, especially when the material properties vary with temperature, or some other dependent variables.

3.2.6.2 Thin elements

The thin element idea is the same as the weighted average method described previously, but with an added constraint placed on the FE mesh. This constraint minimises the shifting of the material boundary from the defined location, by the use of a thin element at the material boundary as illustrated in *Figure 3.10*. This thin element is usually a quadrilateral in form; a triangle element is generally not suitable since a thin triangle possesses some undesired characteristics, well publicised in finite element analysis.

The positioning and thickness of the thin element can have a great effect on the overall solutions. This can be easily explained, for positioning, as applied to metal casting, the thin element should be placed on the side of the sand mould. The reasoning is simple; if placed on the side of the metal, the sand mould can be in a liquid state which has been emphasized above as impossible. And as for the thickness, it is self explanatory,

the thinner the element the smaller the inaccuracy.

The controlling of the position and the thickness of the thin element, should ideally be done at the mesh generation stage. The mesh generation software, if any good, should handle the extra requirements without any difficulty.

3.2.6.3 Coincident nodes

For both the weighted average and thin element methods, there will always be a finite inaccuracy in representing the material boundary. As long as the thickness of the element is greater than zero.

For the coincident node technique [Sammonds et al (1985)], the representation of the material boundary has no such inaccuracy since there is no shifting of the material boundary. The technique itself is simple, consider two adjacent elements as in *Figure 3.11*. The element topology for these elements can be listed as

Element No.	Nodes			
1	1	2	3	4
2	2	5	6	3

If however, a second spatially coincident node is placed at each material boundary node, with one node belonging exclusively to one element as illustrated in *Figure 3.12*, then

the element topology will be listed as

Element No.	Nodes			
1	1	2	3	4
2	5	6	7	8

With the FE mesh segregated into material regions by the coincident nodes, making each fragment of the mesh totally independent. The heat flux that normally flows across the material boundary has been disconnected, letting the boundary assume an adiabatic boundary condition. A link has to be re-established in order for each fragmented mesh to take into account the influence of its neighbours.

The Link-up is made via the relationships between the two coincident nodes, with the material boundary assuming a convective type heat transfer boundary condition [Lewis and Roberts (1987)] that is expressed as

$$q_{bdy} = h_{eff} (T_{metal} - T_{mould}) \quad 3.26$$

where the effective heat transfer coefficient, h_{eff} , can assume experimental values or can be calculated from the expression [Michalek, Kelly and Dantzig (1986)]

$$h_{eff} = \frac{k_{gap}}{\Delta_{gap}} \quad 3.27$$

where k_{gap} is the thermal conductivity of the gap medium (usually air) and Δ_{gap} is the gap distance. This method of modelling the material boundary has been successfully employed by both Lewis and Roberts (1987) in *Finite Element* and Bailey et al (1992)

in *Control Volume*, both in casting. In Bailey et al, they were predicting the deformation of castings in moulds, in which separation between the cast and the mould is a real physical phenomenon while the cast is undergoing cooling.

The coincident nodes technique can represent and accommodate more of the physical phenomena that occur in casting than any of the previous two methods. The price for the added versatility and physics are the extra computations and procedures involved for generating the coincident nodes, re-numbering the topology of the affected elements and relinkage of the fragmented mesh.

3.3 Cell-centred Approach

In this sub-section, the cell-centred approach to unstructured mesh or the more appropriate name of Irregular Control Volume (ICV) is to be described for the solidification by conduction only in two dimensions. The solidification algorithm and solution procedure are the ones described in *Chapter 2* under *sections 2.2* and *2.4* respectively. Given a conventional structured mesh, the cell-centred approach is the direct analogue to the standard control volume (CV) method. The control volume method is well documented and the full fluid flow and heat transfer using ICV is to be described in a later chapter. The emphasis here will be focus on the discretisation of the solidification equations as applied to an unstructured mesh, on which the conservation principle will be applied to the irregular control volumes.

3.3.1. Domain discretisation

A full description of the domain discretisation has been given in section 3.2.2 which also relates itself to what is explained here. For clarity, two of the terms that have been associated with a finite element mesh will now be redefined here as applied to the cell-centred method. The element will now be referred as the cell or the control volume, with the vertices of the elements/cells referred as grid points, see *Figure 3.13*.

Nodes are defined at the centroid of the cell, where all the problem unknowns and

information about the cell are stored. For solidification, this includes enthalpy, temperature, liquid fraction, material properties, etc. The cell itself is polygonal in shape, which makes it possible to accept any meshes that can be produced by finite element mesh generators.

3.3.2 Discretisation of solidification equations

Integrating the solidification equation 2.1 in *Section 2.2* over an arbitrary control volume has already been done for the vertex case in *Section 3.2.4*. For cell-centred, only the diffusion term in equation 3.16 needs to be formulated differently and this is done below.

$$\int_S k \frac{\partial}{\partial x_i} (h/c) \, dn_i = \sum_{i=1}^{N_s} k_i \left(\left(\frac{h}{c} \right)_i - \left(\frac{h}{c} \right)_p \right) \frac{\Delta s_i}{\Delta n_i} \hat{n}_i \quad 3.28$$

where N_s is the number of cell faces, and by looking at the i face, Δs_i is the surface area, Δn_i is the distance between the two nodes on either side of the face, and \hat{n}_i is the unit normal to the face. Note, the cross-diffusion term [Demirdzic and Peric (1990)] in equation 3.28 has been omitted. For orthogonal meshes the term disappears, and is small compared to the other part if the mesh non-orthogonality is not severe. Consequently the method in its current form is opened to influences by meshes that are severely non-orthogonal. But for meshes that are produced by a fully- or semi-

automatic mesh generators they are generally near orthogonal, for a fixed mesh the cross-diffusion term is insignificant and can be omitted from the calculations.

By substituting equation 3.17, 3.18 and 3.28 into equation 3.16 to yield a set of equations in the form of equation 3.21, and with suitable initial and boundary conditions, the solidification process by heat conduction only has been discretised using the cell-centred method and can be solved by the algorithm described in *Section 2.4*.

3.3.3 Boundary conditions

Before the discretised equations can be solved, boundary conditions are needed to complete the equations for conservation at the solution domain boundaries.

The usual boundary conditions, see section 3.2.6, can be applied in the same way as standard control volumes. With a polygonal control volume, the flux across the domain boundary needs to be evaluated perpendicular to the cell face.

For internal boundaries such as material interface, normal control volume practices can be successfully applied without resorting to thin elements or coincident nodes as in the vertex-centred method.

3.4 Closure

The Control Volume based Unstructured Mesh (CV-UM) methods, both vertex- and cell-centred approaches have been fully described for the solidification algorithm by conduction only in two-dimensions. With references to problems that need to be addressed by each method as it applies to solidification algorithms.

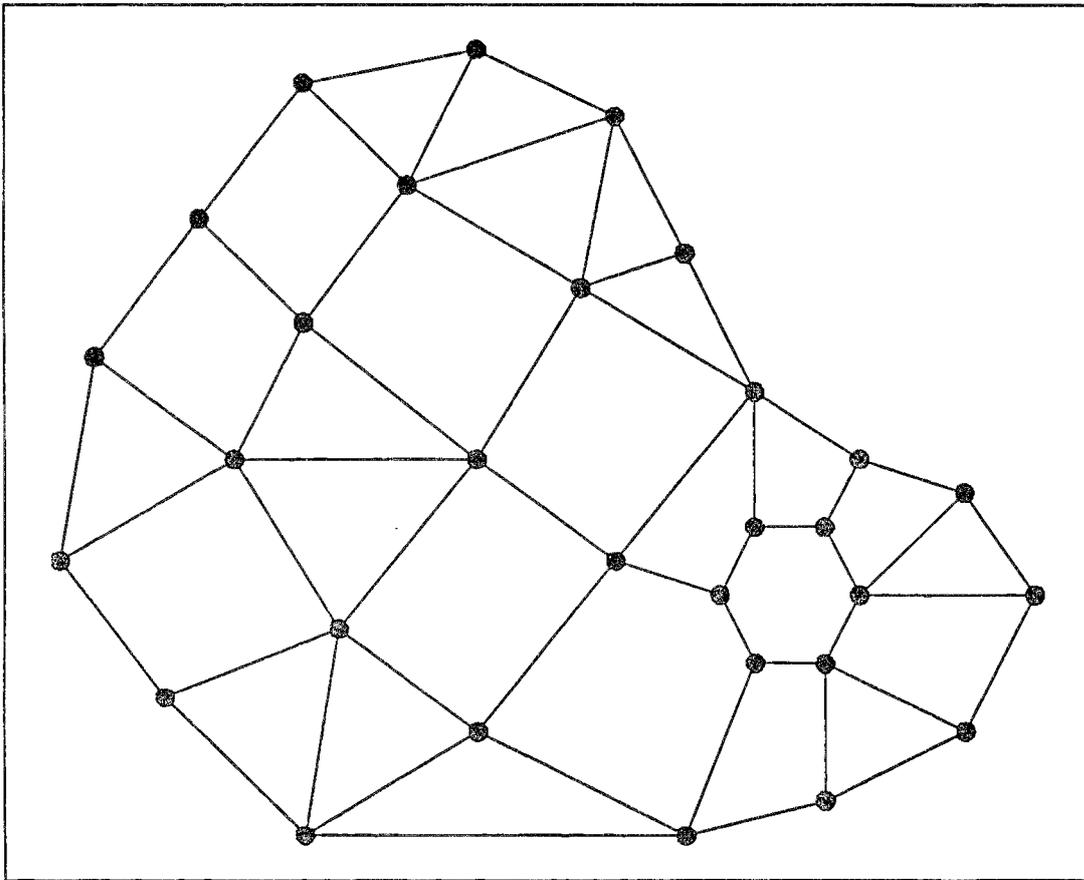


Figure 3.1 Finite element mesh.

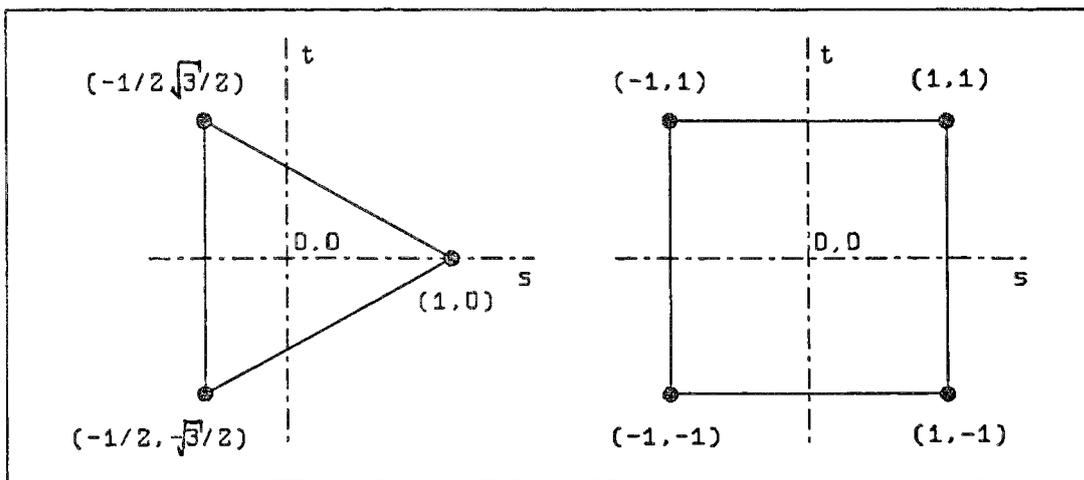


Figure 3.2 Local co-ordinate systems.

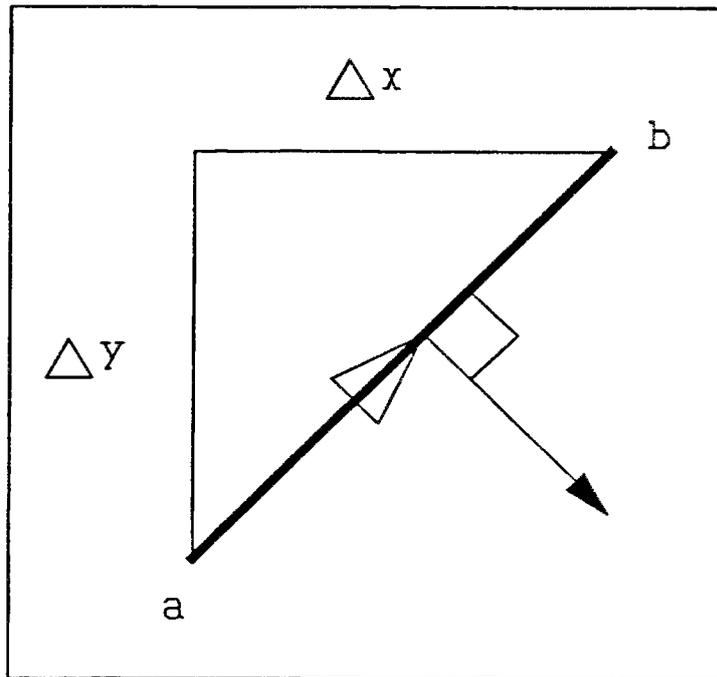


Figure 3.3 Line segment of surface normal vector.

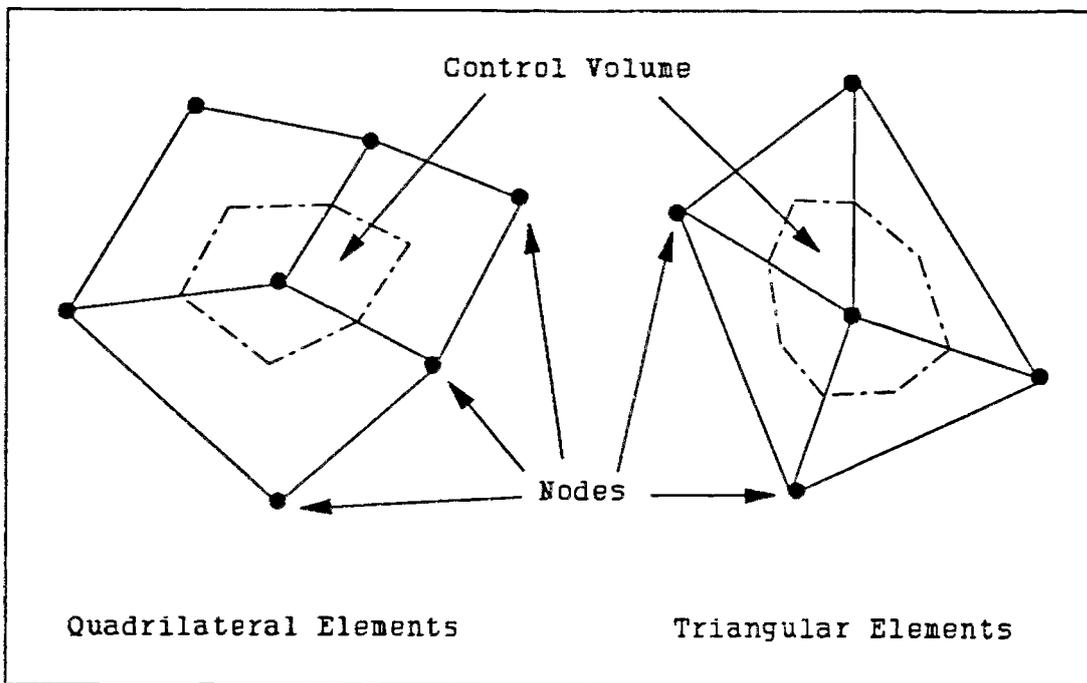


Figure 3.4 Vertex-centred mesh-control volume relationship.

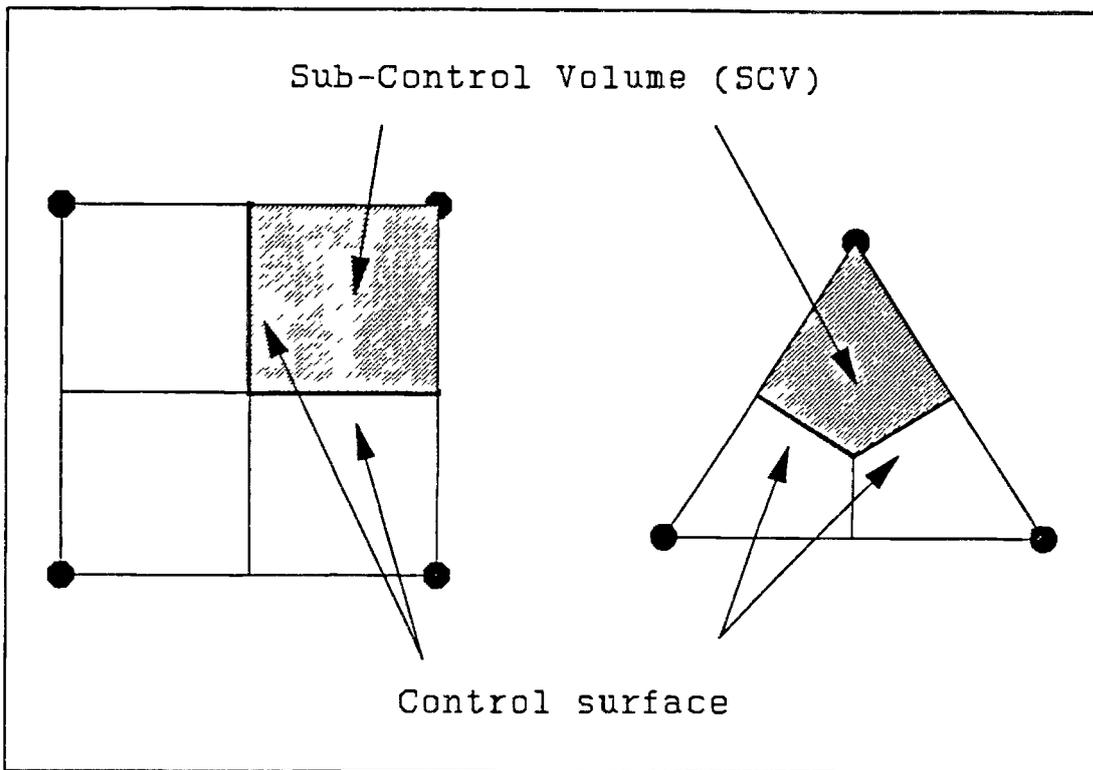


Figure 3.5 Definition of sub-control volumes.

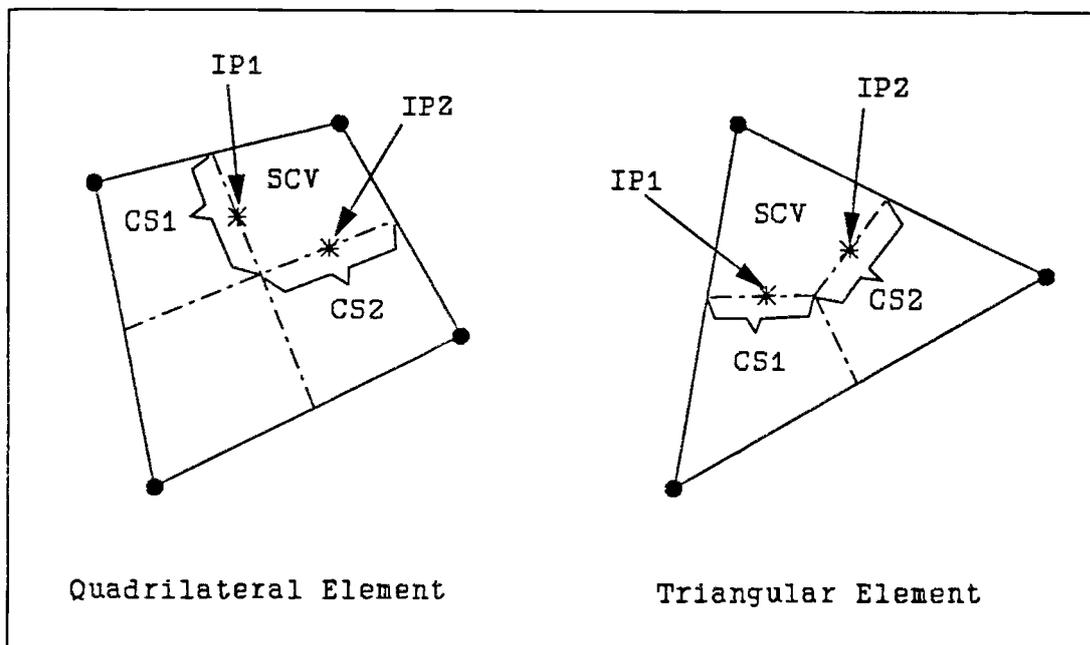


Figure 3.6 Definition of integration points.

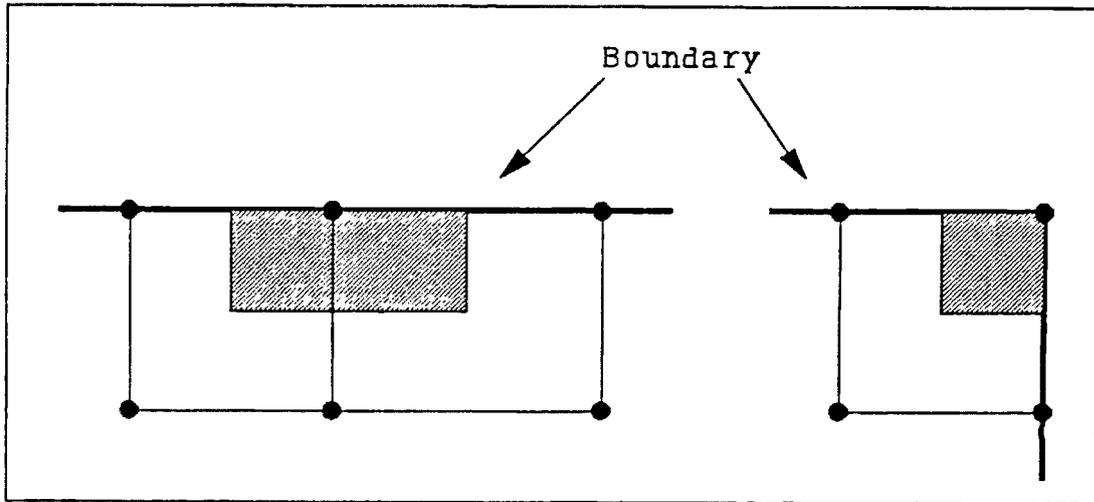


Figure 3.7 Element boundary.

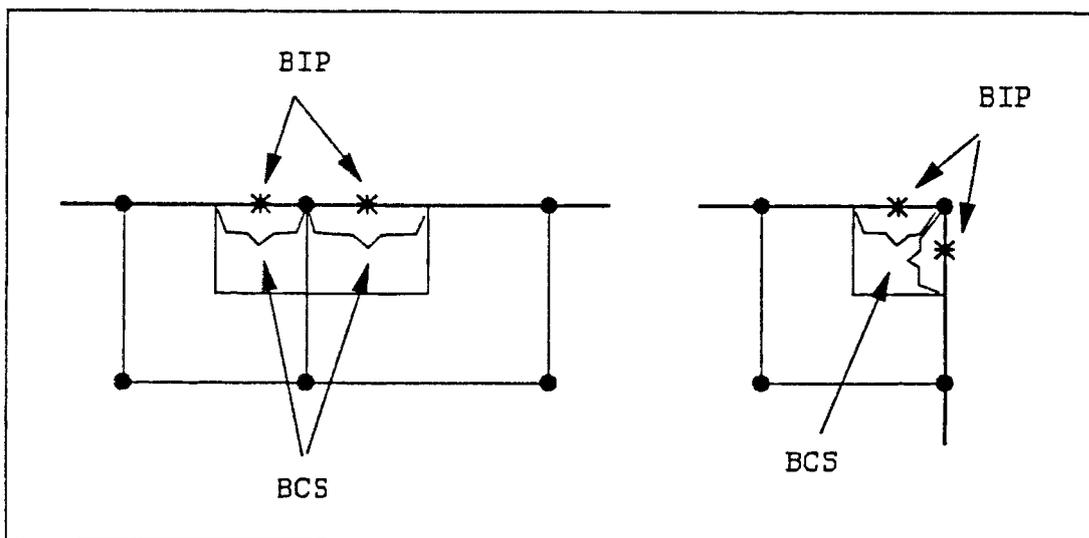


Figure 3.8 Boundary integration points and control surfaces.

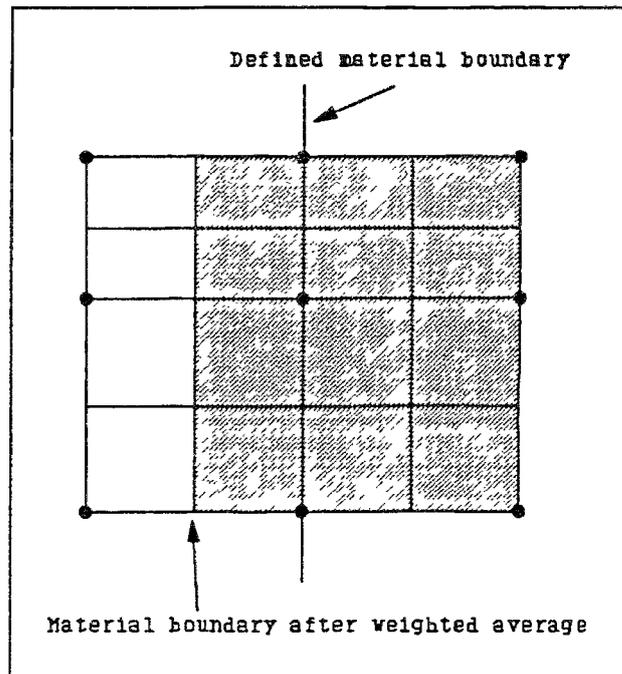


Figure 3.9 Weighted average method.

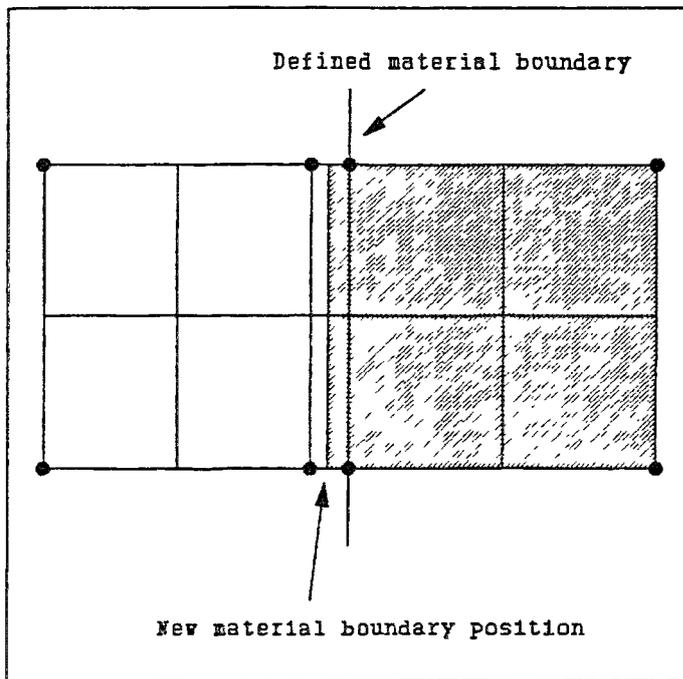


Figure 3.10 Thin element approach.

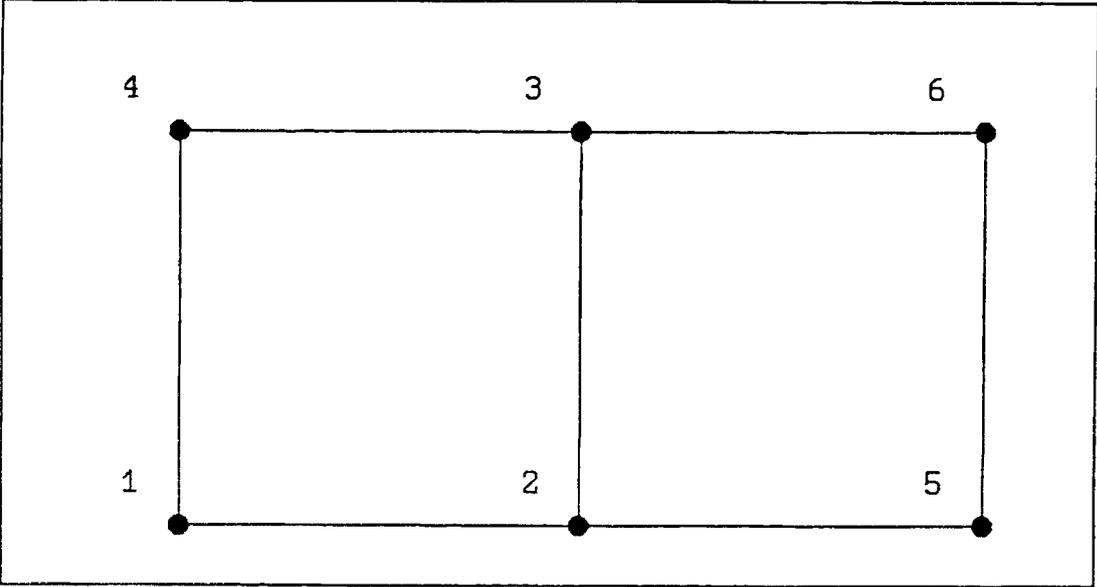


Figure 3.11 Standard element numbering.

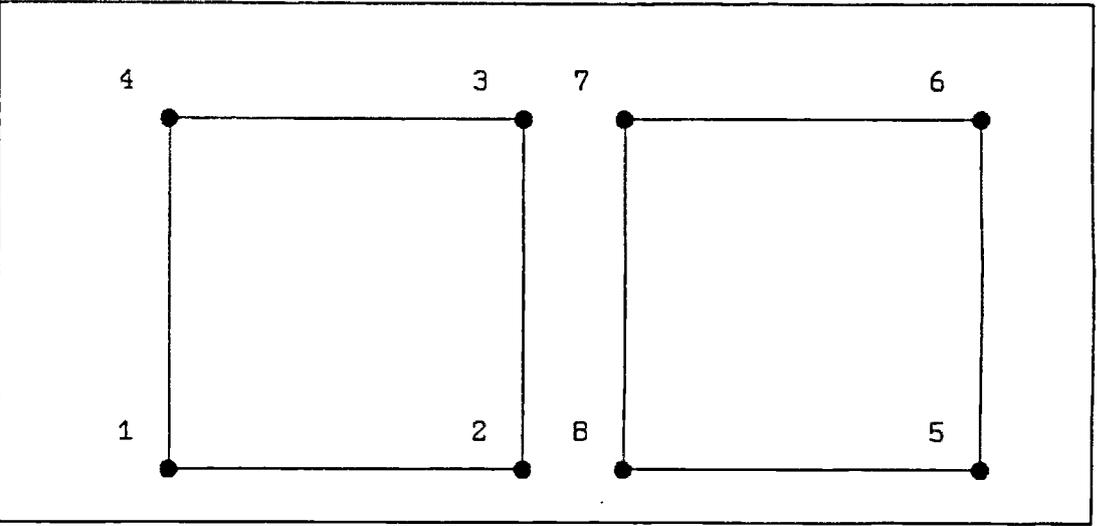


Figure 3.12 coincident nodes numbering.

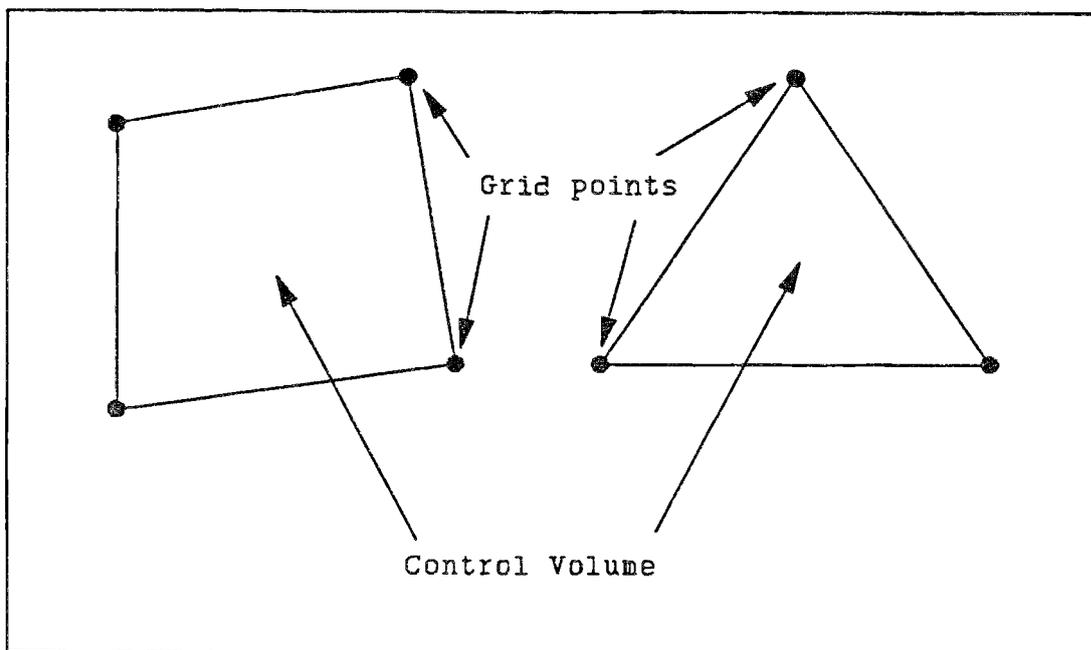


Figure 3.13 Cell-centred mesh-control volume relationship.

Chapter 4

4.0 Solidification by conduction - Validations and results	62
4.1 Introduction	62
4.2 Solution procedure	63
4.3 Test cases	63
4.3.1 Test case I - Carslaw and Jaeger	63
4.3.2 Test case II - Rathjen and Jiji	69
4.3.3 Test case III - Tao	73
4.4 A casting problem	75
4.5 Assessment/Conclusion	82
4.6 Closure	84
Figures	85

4.0 Solidification by conduction - Validations and results

4.1 Introduction

In the previous chapter, the control volume-unstructured mesh (CV-UM) approaches have been detailed in context of solidification by conduction only. To assess how these unstructured methods perform on the solidification algorithm based on Voller and Cross (1985), given in *Chapter 2* of this thesis, will be presented in this chapter. The unstructured methods under evaluation are; the vertex-centred, control volume-finite element (CV-FE) and the cell-centred, irregular control volume (ICV) methods.

The test cases used for the assessment consist of three standard phase change problems and a realistic problem with experimental data relevant to casting. The three standard cases are; the semi-infinite one-dimensional slab as presented by Carslaw and Jaeger (1959), the isothermal freezing at a corner in two-dimensions by Rathjen and Jiji (1971) and an axi-symmetric problem as presented by Tao (1967). For the casting problem, the one presented by Zeng and Pehlke (1985) was chosen for their careful laboratory experiments carried out to verify a finite difference algorithm of their own.

In each test problem, the same quadrilateral and triangle finite element meshes were used for both the unstructured methods. The finite element meshes have no special pre-treatment given to them, such as, mesh adaption or refinement to suit the problem. The purpose of this is to assess the sensitivity of the numerical solution to the orientation or quality of the mesh, for each unstructured methods. Using the same meshes in both the

methods, the computational cost of each method can be compared and in the quadrilateral mesh case, the overhead cost associated with the unstructured mesh approach over the structured one can be measured also.

4.2 Solution procedure

The solution procedure for phase change problems and the criteria used for convergence have already been given in *Section 2.4*. In the solution of system of equations for enthalpy, the Gauss-Seidal with successive over-relaxation (SOR) is used. The default tolerance value taken for all the problems considered in this chapter is set at 1.0E-03 for both temperature and liquid fraction.

All the results and timings reported in this chapter are conducted on a single T800/20 transputer hosted on a PC-AT machine.

4.3 Test cases

4.3.1 Test case I - Carslaw and Jaeger

This is the classical semi-infinite one-dimensional slab as presented by Carslaw and Jaeger (1959), where the movement of the phase change front can be determined analytically. The geometrical specification of the problem is illustrated in *Figure 4.1*,

and the physical properties and dimensions are given in *Table 4.1*. For the problem to be solved numerically, a finite domain needs to be specified. So a reasonably large value taken of the length dimension was used to minimise the semi-infinite characteristic of the problem. See *Table 4.1* for the dimensions taken.

Physical properties		
Conductivity	0.001	W/mC
Specific heat	1.0	MJ/kgC
Density	1.0	kg/m ³
Latent heat	5.0	MJ/kg
Solidus temperature	0.0	C
Liquidus temperature	0.0	C
Dimensions		
Length	1.0	m
Width	1.0	m

Table 4.1 Physical properties and dimensions of Carslaw and Jaeger test case.

The problem was solved using triangular and quadrilateral meshes with various mesh densities. The triangular meshes are formed by dividing the quadrilateral element into two, by joining the opposite corners, a mesh for the NE/SW and NW/SE directions and one in alternating fashion. For a full illustration of all the meshes used, see *Figure 4.2*. In the quadrilateral case, the same mesh was used in the conventional control volume approach. This provided a comparison of solution accuracy and computing performance between the structured and the two unstructured methods using the same mesh for the solution of the same problem.

The time-step used for all the simulations is 10 seconds, for a total of 50 time-steps.

Relaxation was found necessary in updating the liquid fraction for all the simulations to prevent divergence and to aid fast convergence. Optimum values were found and they all range between 0.2 to 0.6 inclusive for all the methods with various mesh densities.

A contour plot of the temperature for each of the meshes are show in *Figures 4.3* and *4.4* for both the CV-FE and ICV methods. In *Figure 4.4*, the NE-SW and NW-SE temperature contour is non-vertical. This is due to the severe non-orthogonality of the meshes for the present ICV method, as explained in *Section 3.3.2*. The numerical calculation of the phase change front movement was compared with the analytical results. *Figures 4.5* and *4.6* shows two of the results obtained using CV-FE with mesh sizes, 4x21 and 4x41 respectively. The ICV results using the same two meshes are displayed respectively in *Figures 4.7* and *4.8*. A table showing all the root mean square (RMS) values for the position of the phase change, for all the mesh orientations and densities are in *Table 4.2*. For a break down of the computing times for each method and mesh densities, see *Table 4.3*. In *Table 4.3*, computational cost per node is also given, this is a more meaningful cost than the overall cost, since the CV-FE and ICV may be using the same mesh, but because of the element-control volume relationship, it does not necessarily mean the same number of nodes used in the calculations.

The comparison between the analytic and the numerical evaluation of the phase change front for both the unstructured methods are reasonable. All the methods obtained similar

accuracy in all these simulations except ICV with meshes that are severely non-orthogonal. The main difference is in the computing time as given in *Table 4.3*, where the overhead involved in using unstructured mesh is about a factor of between 2 to 3 for the CV-FE per node, and for the ICV there is no difference between the structured and the rectangle mesh, while the triangle work out to be less per node than structured.

	4x11	4x21	4x31	4x41
CV	0.013994	0.005322	0.004837	0.004322
No. Nodes	30	60	90	120
CV-FE				
NE-SW (/)	0.017700	0.005920	0.004427	0.005068
NW-SE (\)	0.017700	0.005920	0.004427	0.005068
Alter. (X)	0.024670	0.009416	0.005643	0.004768
Rectangle	0.017782	0.004956	0.005011	0.003085
No. Nodes	44	84	124	164
ICV				
NE-SW (/)	0.012467	0.013006	0.014341	0.015006
NW-SE (\)	0.012467	0.012982	0.014335	0.015039
Alter. (X)	0.007460	0.004527	0.004056	0.004593
No. Nodes	60	120	180	240
Rectangle	0.013994	0.005322	0.004837	0.004322
No. Nodes	30	60	90	120

Table 4.2 The RMS for the methods in various mesh densities.

	4x11	4x21	4x31	4x41
CV	64 (2.07)	128 (2.13)	192 (2.13)	256 (2.13)
CV-FE				
NE-SW (/)	92 (2.09)	254 (3.02)	531 (4.28)	804 (4.90)
NW-SE (\)	92 (2.09)	254 (3.02)	530 (4.27)	803 (4.90)
Alter. (X)	103 (2.34)	283 (3.37)	621 (5.01)	1023 (6.24)
Rectangle	82 (1.86)	200 (2.38)	435 (3.51)	722 (4.40)
ICV				
NE-SW (/)	128 (2.13)	192 (1.60)	320 (1.78)	384 (1.60)
NW-SE (\)	128 (2.13)	192 (1.60)	320 (1.78)	384 (1.60)
Alter. (X)	64 (1.07)	192 (1.60)	256 (1.42)	320 (1.33)
Rectangle	64 (2.07)	128 (2.13)	192 (2.13)	256 (2.13)

Number in bracket () is computing time per node.

All times are in seconds.

Table 4.3 The computing times for the methods in various mesh densities.

4.3.2 Test case II - Rathjen and Jiji

The second test case is taken from Rathjen and Jiji (1971), the problem geometrical specification is illustrated in *Figure 4.9* and the physical properties and dimensions taken are given in *Table 4.4*. The problem is concerned with isothermal phase change in two-dimensions at a corner, it has the benefit of an analytical solution for temperature and the movement of the phase change front, with which the computed solution can compare against.

Physical properties		
Conductivity	30	W/mC
Specific heat	750	J/kgC
Density	7200	kg/m ³
Latent heat	262500	J/kg
Solidus temperature	1773	C
Liquidus temperature	1773	C
Dimensions		
Length	0.381	m
Width	0.381	m

Table 4.4 Physical properties and dimensions of Rathjen and Jiji test case.

Using similar mesh pattern as in test case I. A time-step of 60 seconds is used in all the simulations, for a total of 100 time-steps. Again, relaxation was found necessary for liquid fraction updates within the range 0.1 to 0.9 inclusive for all the methods with various mesh densities.

A temperature contour plot of the simulation after 100 minutes of cooling is shown in

Figures 4.10 and 4.11 for each of the meshes using the two methods, in the ICV, the two non-orthogonal mesh results have been swayed by the orientation of the mesh. The numerical calculated movement of the phase change front can be compared with the analytic solution. Taking the mesh sizes of 21x21 and 41x41 for illustration, the CV-FE solutions are in *Figures 4.12 and 4.13*, and *Figures 4.14 and 4.15* are the ICV results. *Figure 4.16* show the temperature residual mean square (RMS) error values for each method with respect to the analytical solution for an increasingly refined mesh; the RMS values are given in *Table 4.5*. *Table 4.6* show the overall computing times for each method with the various meshes used for the computations, and the relevant computational cost per node for each method. A graphical representation of these computing times, both overall and per node, for each method in an increasingly refined mesh are shown in *Figures 4.17 and 4.18* respectively.

The accuracy of the unstructured meshes is very similar and only very marginally worse than that of the comparable structured mesh solution. The same general computational performances are observed as in test case I, however, the overhead for the CV-FE has now got up to 12-15 and for the ICV 1.2-1.9 for larger meshes.

Mesh size	11x11	21x21	31x31	41x41
CV	5.174	2.656	2.411	2.295
No. Nodes	100	400	900	1600
CV-FE				
NE-SW (/)	4.598	2.658	2.501	2.594
NW-SE (\)	4.630	2.643	2.506	2.596
Alter. (X)	4.280	2.695	2.571	2.625
Rectangle	4.412	2.639	2.482	2.407
No. Nodes	121	441	961	1681
ICV				
NE-SW (/)	13.446	13.756	13.942	14.029
NW-SE (\)	17.714	16.822	17.074	17.240
Alter. (X)	3.435	2.257	2.289	2.264
No. Nodes	200	800	1800	3200
Rectangle	5.174	2.656	2.411	2.295
No. Nodes	100	400	900	1600

Table 4.5 The RMS error for the Rathjen and Jiji test case.

Mesh size	11x11	21x21	31x31	41x41
CV	192 (1.92)	704 (1.76)	1792 (1.99)	3904 (2.44)
CV-FE				
NE-SW (/)	449 (3.71)	3665 (8.31)	18345 (19.09)	62412 (37.13)
NW-SE (\)	449 (3.71)	3664 (8.31)	18345 (19.09)	62410 (37.13)
Alter. (X)	454 (3.75)	3750 (8.50)	18309 (19.05)	64262 (38.23)
Rectangle	385 (3.18)	2890 (6.55)	15144 (15.76)	52659 (31.33)
ICV				
NE-SW (/)	384 (1.92)	2176 (2.72)	6272 (3.48)	14528 (4.54)
NW-SE (\)	384 (1.92)	2112 (2.64)	5888 (3.27)	133376 (4.18)
Alter. (X)	384 (1.92)	1920 (2.40)	5568 (3.09)	12544 (3.92)
Rectangle	192 (1.92)	768 (1.92)	2048 (2.28)	4608 (2.88)

Number in bracket () is the computing time per node.

All times are in seconds.

Table 4.6 Computing times for Rathjen and Jiji test case.

4.3.3 Test case III - Tao

In the previous two test cases, both problems have had rectangular geometries. In this test case, the domain under consideration is one quarter of a circle. The geometrical specification of the domain is illustrated in *Figure 4.19*, and the physical properties and dimensions are in *Table 4.7*. This problem is derived from Tao (1967), and in reality, is an axi-symmetric problem that can be solved in one-dimension. However, as a test problem with a curved boundary in cartesian co-ordinates, the two-dimensional irregular domain in *Figure 4.19* can be solved and the movement of the phase change front can be compared with Tao's one-dimensional results.

Physical properties	
Conductivity	1
Specific heat	1
Density	1
Latent heat	1
Solidus temperature	1
Liquidus temperature	1
Dimensions	
Radius	1

Table 4.7 Physical properties and dimensions of Tao test case.

This problem was also solved amongst others by Voller and Cross (1983) using conventional control volume in cylindrical co-ordinates. Their result provided the structured approach solution for comparison with the unstructured solving in cartesian co-ordinate system.

Following from the previous test cases, quadrilateral and triangle finite element meshes were used for the calculations for both the unstructured methods. Using similar mesh size interval and time-step of those published by Voller and Cross (1983), so comparable solutions may be obtained and compared. The mesh size interval and time-step used for the simulation are 0.025 and 0.00025 respectively, and the simulation terminates when it is fully solidified. *Figures 4.20* and *4.21* illustrate the quadrilateral and triangle meshes used in obtaining the CV-FE and ICV results. *Figures 4.22* and *4.23* show the computed results of the solidification front against those of Tao and Voller and Cross respectively using the CV-FE method. For the ICV results and comparisons, see *Figures 4.24* and *4.25* respectively.

The computing times for the two unstructured methods and meshes are in *Table 4.8*. The same general computational performance between the two unstructured methods are observed. Both set of results are reasonable when compared against the one-dimensional result of Tao (1967) and of structured mesh approach of Voller and Cross (1983).

	Triangle	Quadrilateral
CV-FE (Nodes)	952	869
CPU times (secs) (per node)	119565 (125.59)	91973 (105.84)
ICV (Nodes)	1764	816
CPU times (secs) (per node)	37440 (21.22)	14336 (17.57)

Table 4.8 Computing cost for the Tao test case.

4.4 A casting problem

So far, the cases used to test the CV-UM algorithm may seem artificial in relevance to casting, since they all consist only of the solidifying material and without the casting mould, as it would not be the case in real casting. So the next natural step for the unstructured method testing is to solve a realistic problem, consisting of the metal cast and mould for the cast to solidify in, as it changes phase from liquid to solid.

Zeng and Pehlke (1985) provided such a problem, they carried out some very careful laboratory experiments to verify a finite difference algorithm of their own. A schematic showing the experimental set-up and the thermocouple locations in the casting and sand mould are illustrated in *Figure 4.26*. Following from the Zeng and Pehlke's published results, only the cylindrical cast and the sand mould is to be simulated. By using a cylindrical co-ordinate system, the problem can be solved in two-dimensions with a symmetry on one of its sides, as shown in *Figure 4.26*.

The metal to be solidified is metal-gray cast iron (3.4%C and 2.5%Si) with an initial temperature of 1433K and solidification range of 1416K to 1430K. The sand mould is essentially 80-mesh Wedron silica, and the initial temperature of the sand is assumed to be at room temperature (298K). The experimental and finite difference results of Zeng and Pehlke's are shown in *Figures 4.27* and *4.28* respectively. The physical properties for both materials are described in the publication by Pehlke, Jeyarajan and Wada (1982), with the conductivity and specific heat dependent on temperatures. A

summary of these physical properties is given below;

Gray iron

Thermal conductivity (W/mK)

$108.15596 - 0.1040462 T$	$T \leq 600.9278$	K
$61.560339 - 0.0264788 T$	$600.9278 < T \leq 989.8167$	K
$106.29194 - 0.0716486 T$	$989.8167 < T \leq 1174.8167$	K
$-61.324073 + 0.0710255 T$	$1174.8167 < T \leq 1416.4833$	K
$1160.0747 - 0.7912498 T$	$1416.4833 < T \leq 1430.3722$	K
$-118.68767 + 0.1028001 T$	$1430.3722 < T$	K

Specific heat capacity (J/kgK)

$508.89157 + 0.1333835 T$	$T \leq 600.9278$	K
$-157.90458 + 1.4822909 T$	$934.8167 < T \leq 1076.4833$	K
$2533.1851 - 0.6440844 T$	$1076.4833 < T \leq 1473.15$	K
$1584.3645 - 0.0400226 T$	$1473.1500 < T$	K

Density	7141	kg/m^3
Latent heat	312600	J/kg
Liquidus temperature	1430.3722	K
Solidus temperature	1416.4833	K
Initial temperature	1433.15	K

Sand mould

Thermal conductivity (W/mK)

$0.776832 + 9.17896E-4T - 2.83169E-6T^2 + 2.18350E-9T^3$	$T \leq 477.5944$	K
$1.264380 - 9.30598E-4T - 6.02893E-7T^2 + 1.15883E-9T^3$	$477.5944 < T \leq 644.2611$	K
$1.808460 - 2.52916E-3T + 5.33552E-7T^2 + 1.50771E-9T^3$	$644.2611 < T \leq 810.9278$	K
$1.920900 - 2.83781E-3T + 8.67141E-7T^2 + 1.26103E-9T^3$	$810.9278 < T \leq 1033.15$	K
$2.040530 - 3.15623E-3T + 1.18729E-6T^2 + 1.13614E-9T^3$	$1033.15 < T$	K

Specific heat capacity (J/kgK)

$395.43163 + 1.2057272 T$	$T \leq 477.5944$	K
$647.36540 + 0.6782215 T$	$477.5944 < T \leq 644.2611$	K
$939.66658 + 0.2260738 T$	$644.2611 < T \leq 810.9278$	K
$999.77644 + 0.1507159 T$	$810.9278 < T \leq 1033.15$	K
$1062.0613 + 0.0904295 T$	$1033.15 < T$	K

Density	1682	kg/m^3
Initial temperature	298.15	K

The above casting problem was solved using both the CV-UM methods, using both triangular and rectangular meshes. In the rectangular mesh, nodes were arranged to coincide with the thermocouple locations. This resulted in a non-uniform rectangular mesh, which was also the basis for the triangulated mesh. By dividing each rectangle into two triangles, we have the triangular mesh, only the alternating triangular mesh was used for the analysis. This is because of the undesired properties of thin triangular elements, explained in *Section 3.2.6.2*, and the severe non-orthogonality of the non-alternating triangle meshes can have on the solutions as in ICV (without the cross-diffusion term). The meshes used are illustrated in *Figures 4.29* and *4.30* for the respective quadrilateral and triangle meshes.

A major numerical difficulty arose from the CV-FE method that is associated with nodes located along the metal-sand interface, where the control volumes constructed around these nodes consist of two different materials that behave differently during solidification. The full physical meaning has already been explained in the previous chapter under *section 3.2.6*; here we are concerned on its implication to the numerical solution. *Figure 4.31* shows the effects of temperature cooling curves at the thermocouple locations when nothing is done about these metal-sand control volumes. Notice the oscillatory nature of the cooling curves compared with the experimental result of Zeng and Pehlke (1985) in *Figure 4.27*, and the resulting problem with convergence. It has a deleterious effect upon the overall computation time of the numerical results that is illustrated in *Table 4.9*. The oscillatory nature of the cooling curves can be cured by

assigning a single material at these metal-sand control volumes, how this is done is explained in *Section 3.2.6*. This simple modification reduced computational time by a factor of 10 and gave a smooth cooling curves shown in *Figures 4.32* and *4.33*.

Depending on the material assigned, the solution will be shifted away by a finite amount from the correct solution. This is illustrated in the temperature cooling curves at the thermocouple locations, *Figures 4.32* and *4.33* shows the results of assigning a single material, sand in the first instant, iron in the latter, for those metal-sand control volumes. By assigning sand, it has shifted the solution to the left and by assigning iron, it has shifted the solution to the right. From these results, one can see the actual solution lies somewhere between the two. By using finer meshes or thin (quadrilateral) elements at the material boundary this finite shift in solution can be significantly reduced. *Table 4.10* shows the final thermocouple temperature values in the metal at the end of the simulation period, and suggests an average difference from the experimental values of 33-40K or a relative error of 2-3%. While the accuracy of the algorithm is not unsatisfactory, it is rather unsatisfactory to have to approximate the material properties of a control volume to be able to generate such solutions.

The ICV has no such numerical difficulty, by taking the element as the control volume, the metal-sand boundary lies exactly on the face of the control volume making it straight forward to deal with. It was for this numerical complication that the ICV method was initiated and subsequently developed.

Using the same meshes used in the CV-FE method, the ICV solutions are shown in *Figures 4.34* and *4.35* for their respective rectangle and triangle meshes. The comparison with the experimental data is good everywhere and at the end of the experiment, see *Table 4.10*, the agreement is better than 0.4% average relative error for rectangular mesh and 1.0% for the triangular mesh. Note that no special consideration has been made concerning the metal-mould interface heat transfer except that there is perfect contact. The comparative computing times and equivalent performance characteristics for both the methods are shown in *Table 4.9*. The performance of the ICV method is consistent with its performance on the earlier test cases.

It turned out that the thermocouple measurements made in the sand were not regarded as reliable [Zeng and Pehlke (1985)] and so no detail discussion or comparison with these experimental values are made.

APPROACH	STRUCTURED MESH				UNSTRUCTURED MESH					
	No of cells		time/cell		Quadrilateral			Triangular		
	No of cells	cpu time	time/cell		No of cells	cpu time	time/cell	No of cells	cpu time	time/cell
Vertex centred CV					1034	41370	40.01	1034	70529	68.21
Vertex centred CV - modified					1034	6628	6.41	1034	12522	12.11
					1034	5274	5.10	1034	8841	8.55
Cell centred CV	966	800	0.83		966	1211	1.25	1932	2918	1.51

Table 4.9 Computational times for the Zeng & Pehlke problem.

APPROACH	1	2	3	4	7	8	Av rel error%
Experiment	1218.71	1266.76	1294.26	1301.48	1213.71	1294.82	-
Structured CV (966 nodes)	1219.13	1263.68	1296.84	1305.77	1198.70	1290.94	0.38
Vertex centred quadrilateral CV (1034 nodes)	1206.54	1252.33	1283.79	1293.28	1188.35	1280.68	1.1
Vertex centred triangular CV (1034 nodes)	1204.26	1250.88	1283.08	1293.62	1185.93	1280.79	1.2
Vertex centred quadrilateral CV - modified (1034 nodes)	Iron	1254.63	1298.49	1331.93	1341.05	1237.74	2.6
	Sand	1183.03	1227.71	1258.33	1267.05	1159.45	3.2
Vertex centred triangular CV - modified (1034 nodes)	Iron	1253.76	1297.34	1330.83	1339.32	1235.12	2.4
	Sand	1182.29	1226.93	1257.59	1265.83	1160.99	3.4
Cell centred quadrilateral CV (966 nodes)	1219.13	1263.68	1296.84	1305.77	1198.90	1290.94	0.38
Cell centred triangular CV (1932 nodes)	1228.18	1274.07	1308.82	1318.13	1203.88	1302.57	1.05

Table 4.10 Temperature at thermocouple locations.

4.5 Assessment/Conclusion

It is important, in the simulation of casting of components, to be able to use a technique that can match the geometry of the external domain and that of metal-mould configuration closely. The control volume enthalpy based algorithms in the structured mesh approach have proved to be both accurate and efficient in computer time. In this chapter the unstructured control volume enthalpy based algorithm formulations have been assessed via three standard phase change test cases and a realistic casting problem with carefully measured laboratory experimental data.

The difference between the two unstructured approaches, CV-FE and ICV, lies in where the node is located in relation to the element/cell, at cell vertices or at the cell centre.

The relative merits of those two approaches may be summarised as follows:-

Cell-vertex approach

- a) Its numerical accuracy is relatively insensitive to the orientation or quality of the mesh.
 - b) The overhead associated with the formulation of the method, when evaluating faces per cell, is naturally 2-3 times more than the CV. This is why it is more expensive in computing time, which can be 12-15 times the cost over the structured.
 - c) Special consideration has to be given to the interface between two materials (eg. metal-mould) otherwise the algorithm is difficult to
-

converge and can give inaccurate results.

Cell-centred approach

- a) The simple node, element and control volume relationship reduces the complexity of the formulation and reduces the computational overhead so that quadrilateral mesh is about 18% and the triangular mesh is about 86% more than the structured approach.
- b) The accuracy of the solutions without the cross-diffusion term can be affected by the orientation of the mesh, unless it is reasonably orthogonal. Meshes generated by a fully- or semi-automatic mesh generator is naturally near orthogonal, therefore, the omission is possible.
- c) Internal boundaries are easy to represent accurately in a natural way and so require no special attention.

It has been demonstrated that both methods can be used to model the solidification process in complex geometry with a high degree of accuracy. It is difficult and unfair to state which method is best suited in numerical modelling; each has its advantages and disadvantages. It may be necessary to employ the method that most suited for the task, as in the recent works by Cross, Bailey, Chow and Pericleous (1992), where both approaches have been employed for coupling of the convective-diffusive solidification with that of the residual stress calculations. The convective-diffusive solidification is done in cell-centred and residual stress in vertex-centred. The two approaches have been found to co-exist in perfect harmony and without any special treatment needed for

coupling the two methods.

4.6 Closure

The results obtained show that both the unstructured methods CV-FE and ICV can be used to solve solidification by conduction problems with finite element meshes. The ICV method is generally more sensitive to mesh orientation requiring the cells/elements to be reasonably orthogonal in shape. For the CV-FE, it is relatively insensitive to the mesh quality due to the formation of the control volume relationship with the mesh, ie. control volume constructed by sub-control volumes. This assembly of sub-control volumes puts a heavy penalty on the computing time. The ICV has no such extra computation to carry due to its straight forward mesh-control volume relationship.

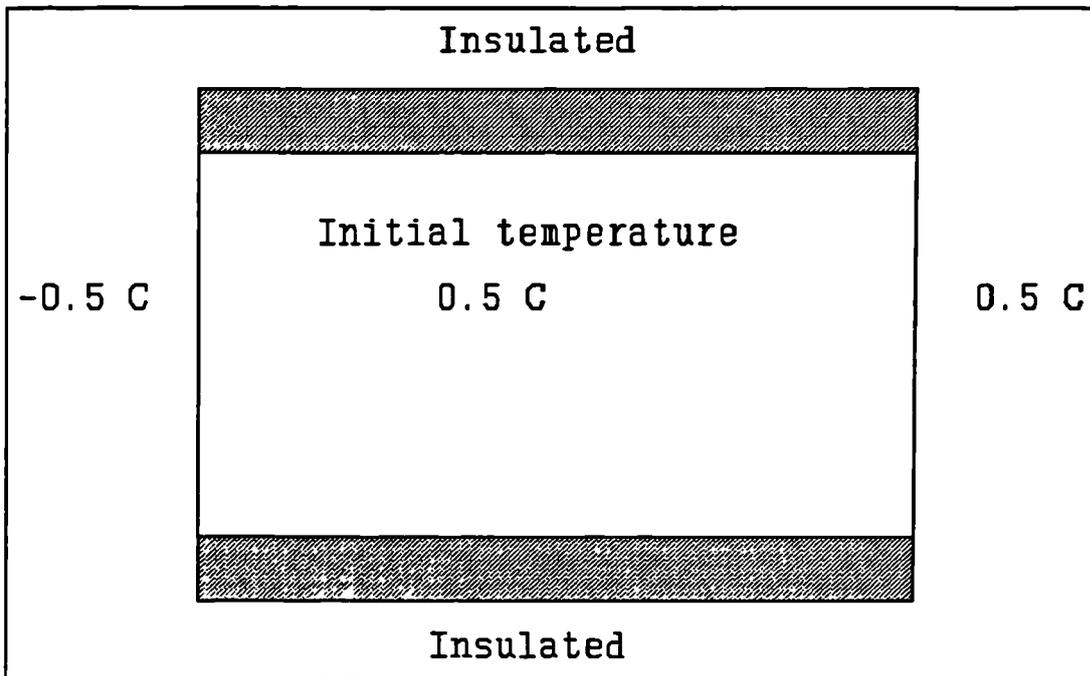


Figure 4.1 Geometrical specification of Carslaw and Jaeger test case.

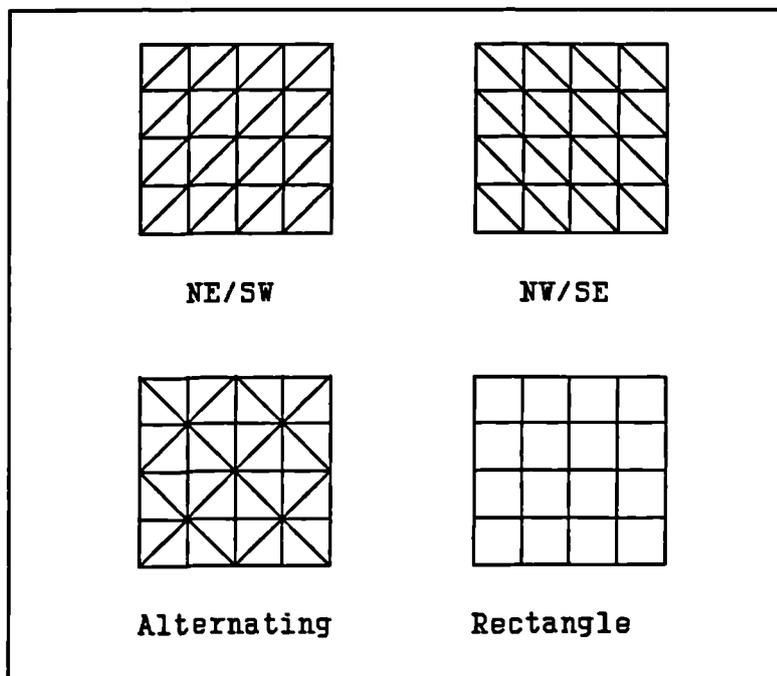


Figure 4.2 The four mesh orientation.

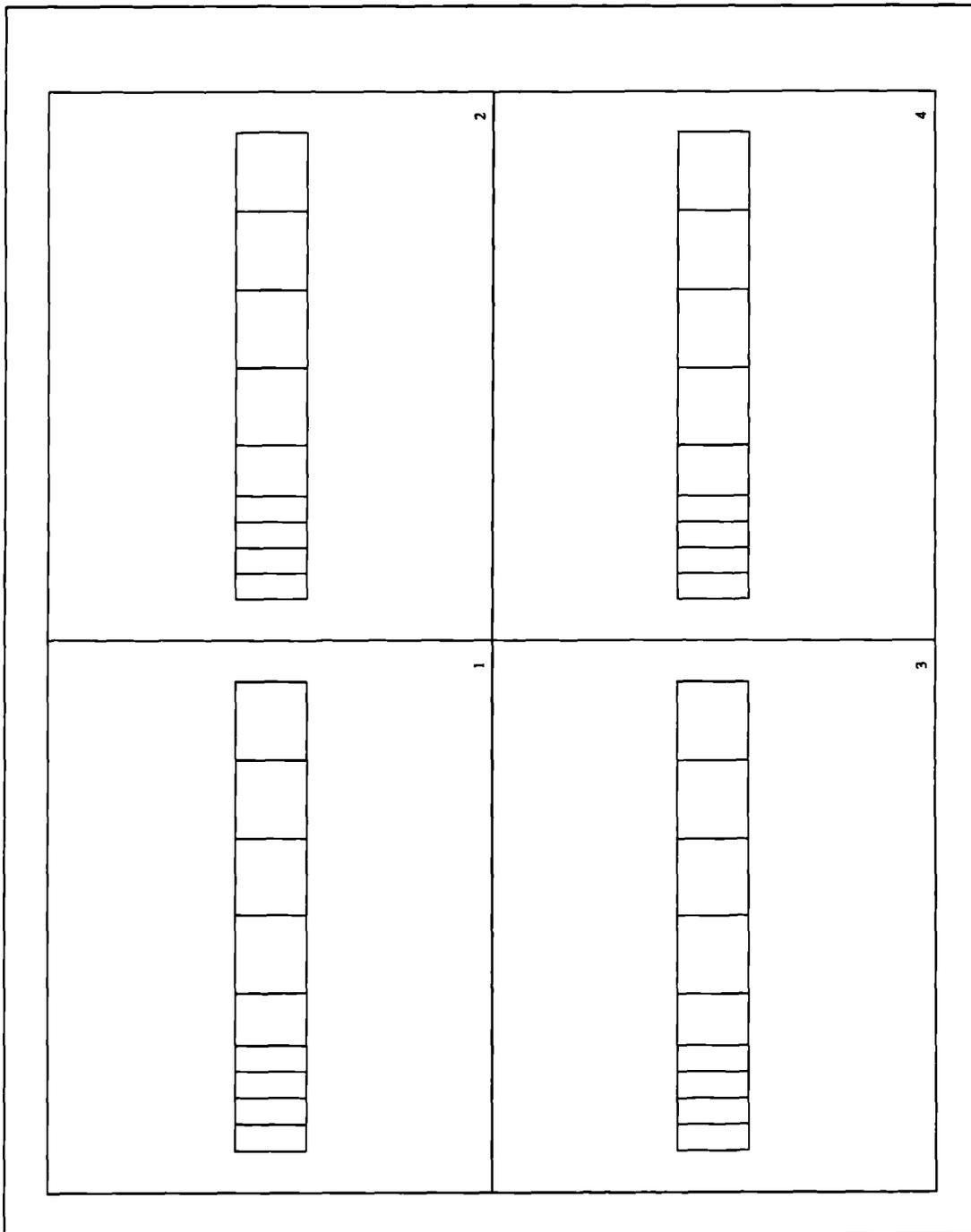


Figure 4.3 CV-FE temperature plot of Carslaw and Jaeger.

- 1) Rectangle result.
- 2) Alternating triangle result.
- 3) NE-SW triangle result.
- 4) NW-SE triangle result.

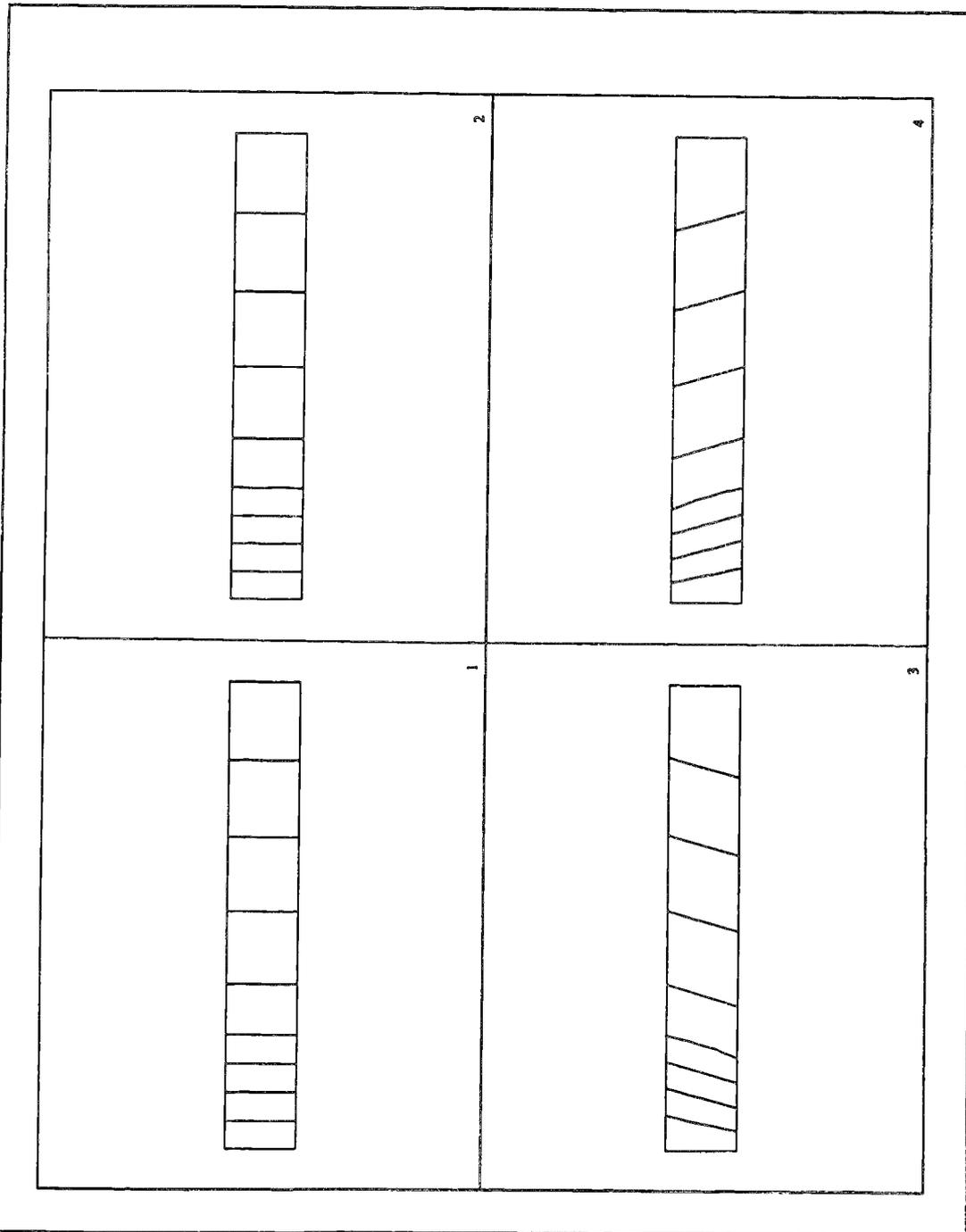


Figure 4.4 ICV temperature plot of Carslaw and Jaeger.

- 1) Rectangle result.
- 2) Alternating triangle result.
- 3) NE-SW triangle result.
- 4) NW-SE triangle result.

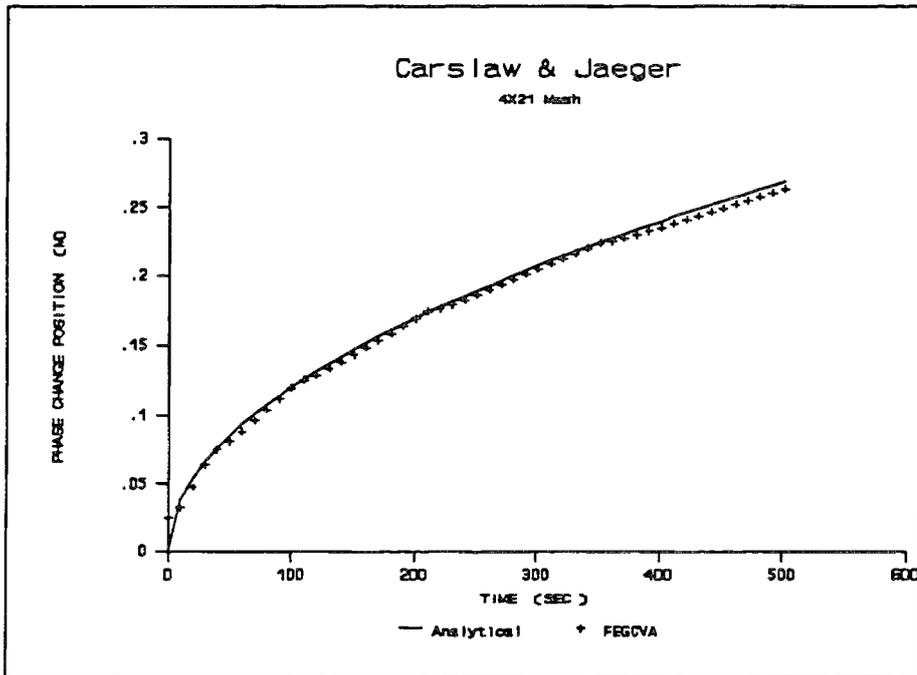


Figure 4.5 Result of 4x21 mesh using CV-FE.

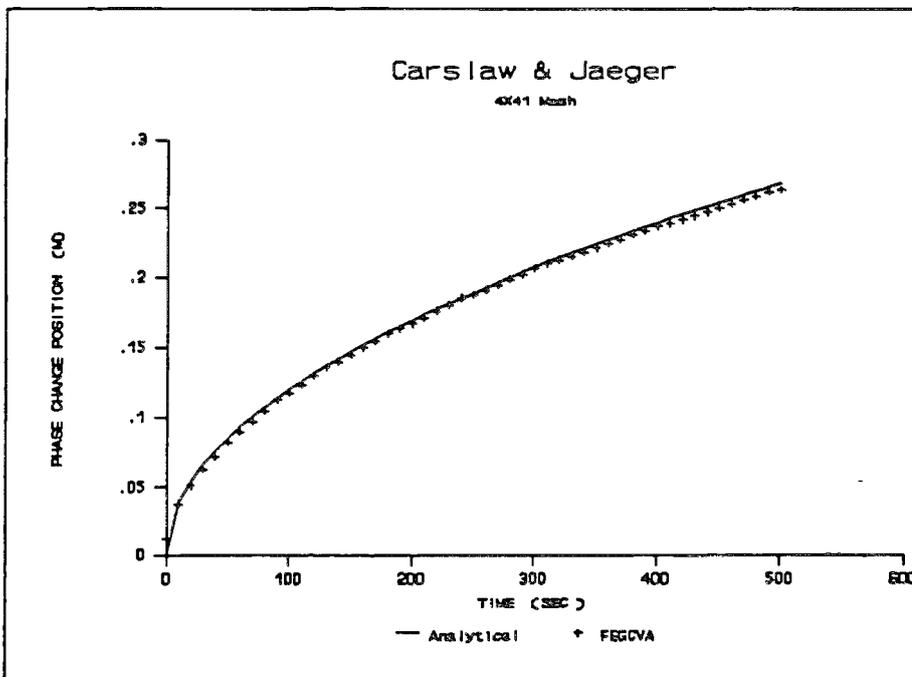


Figure 4.6 Result of 4x41 mesh using CV-FE.

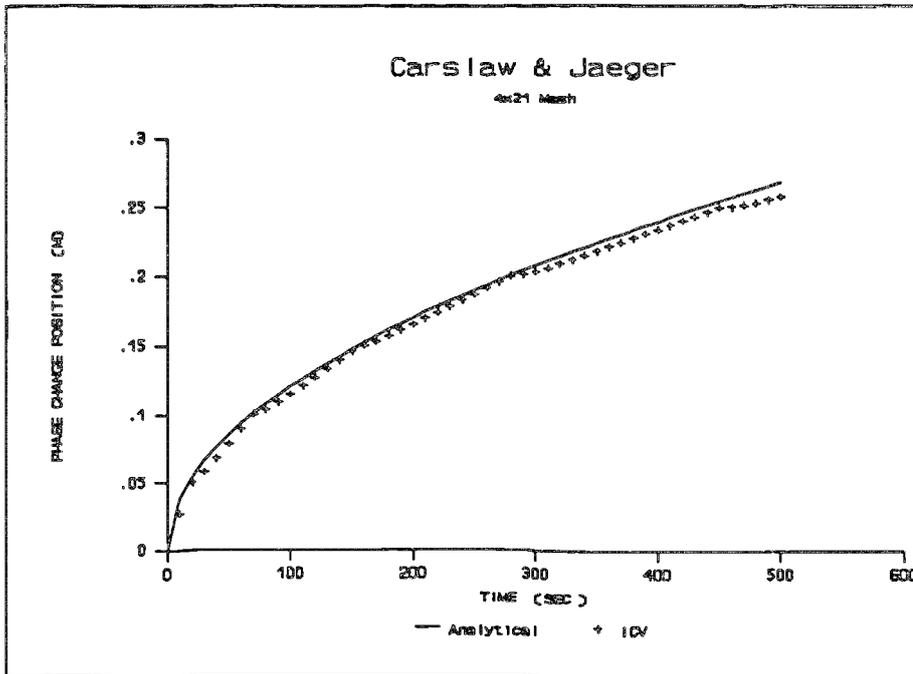


Figure 4.7 Result of 4x21 mesh using ICV.

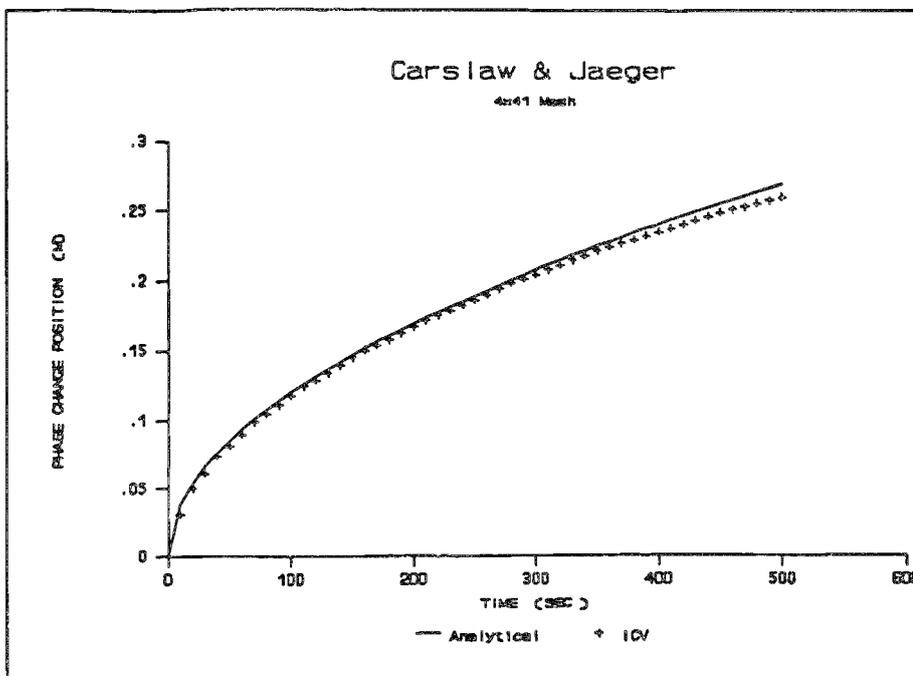


Figure 4.8 Result of 4x41 mesh using ICV.

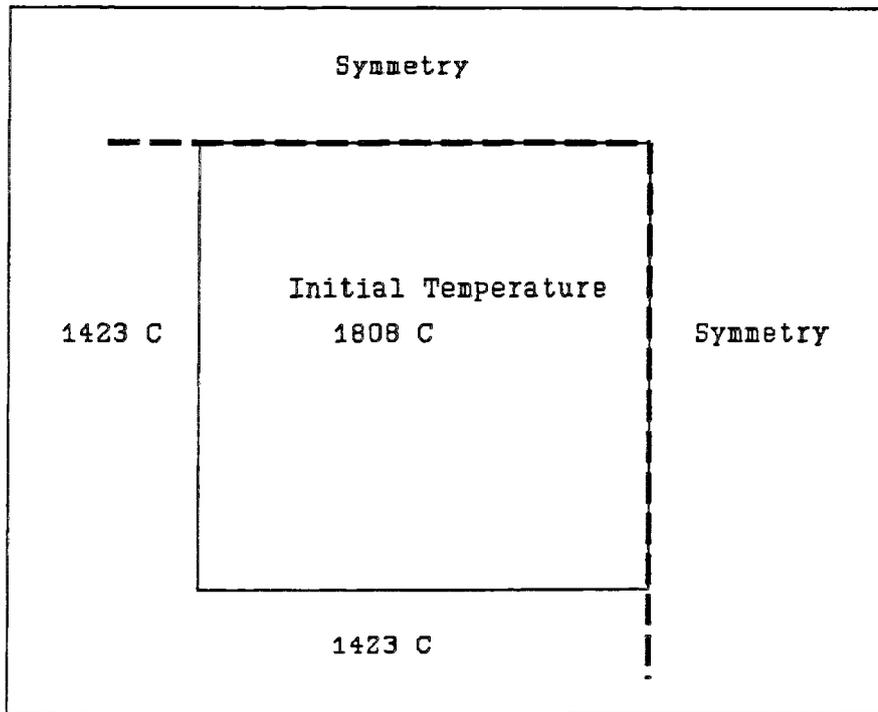


Figure 4.9 Geometrical specification of Rathjen and Jiji test case.

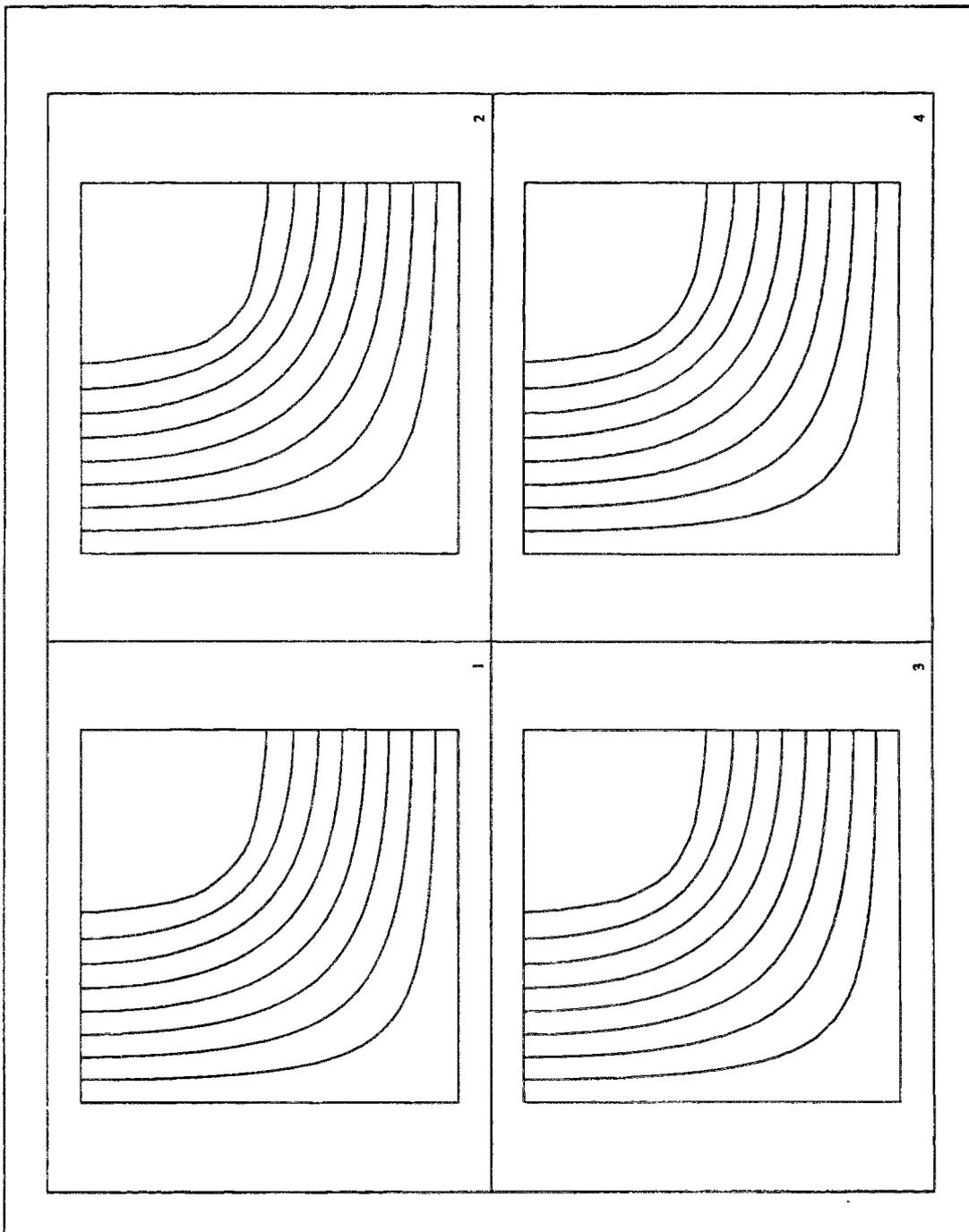


Figure 4.10 CV-FE temperature plot after 100 minutes of cooling.

- 1) Rectangle result.
- 2) Alternating triangle result.
- 3) NE-SW triangle result.
- 4) NW-SE triangle result.

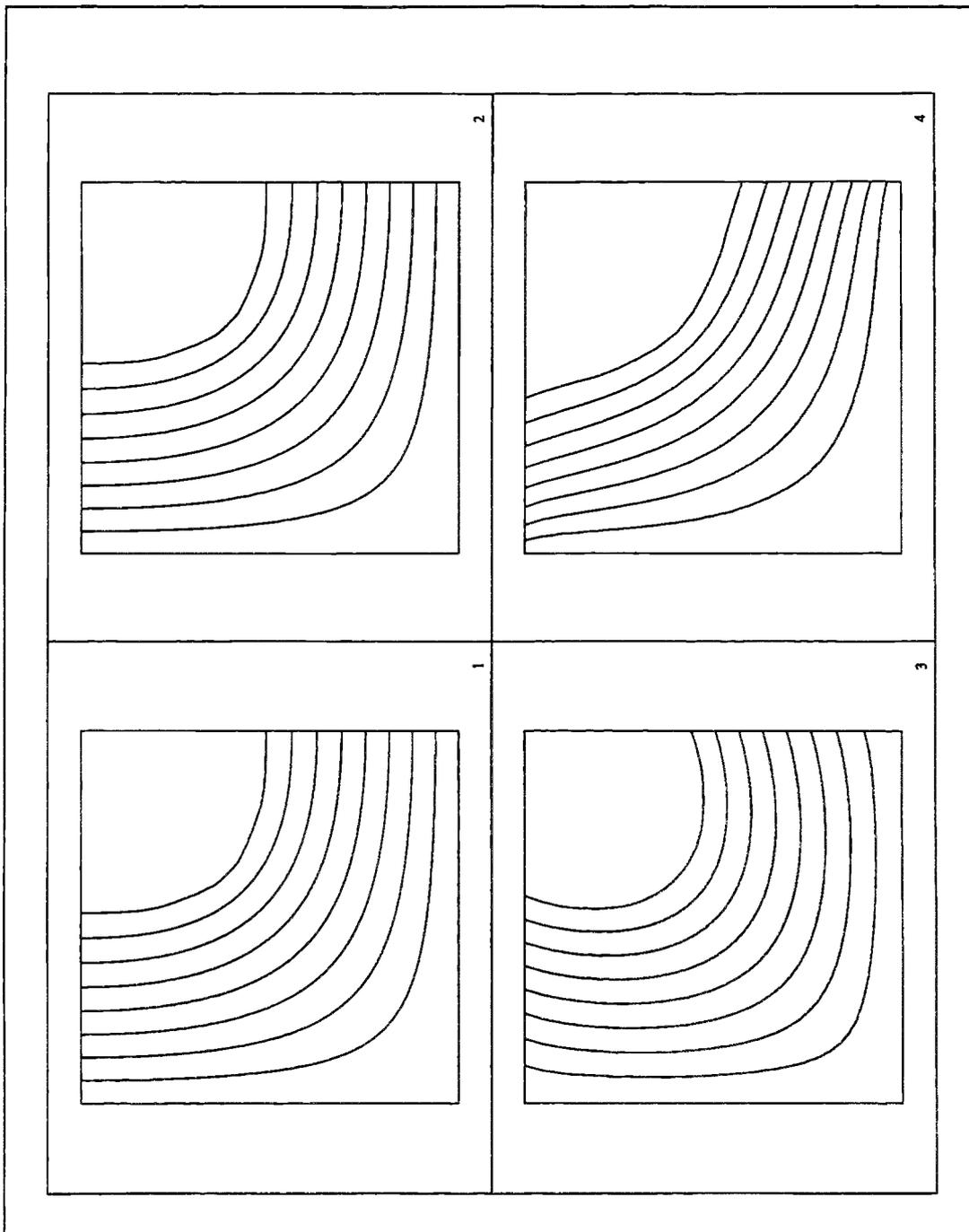


Figure 4.11 ICV temperature plot after 100 minutes of cooling.

- 1) Rectangle result.
- 2) Alternating triangle result.
- 3) NE-SW triangle result.
- 4) NW-SE triangle result.

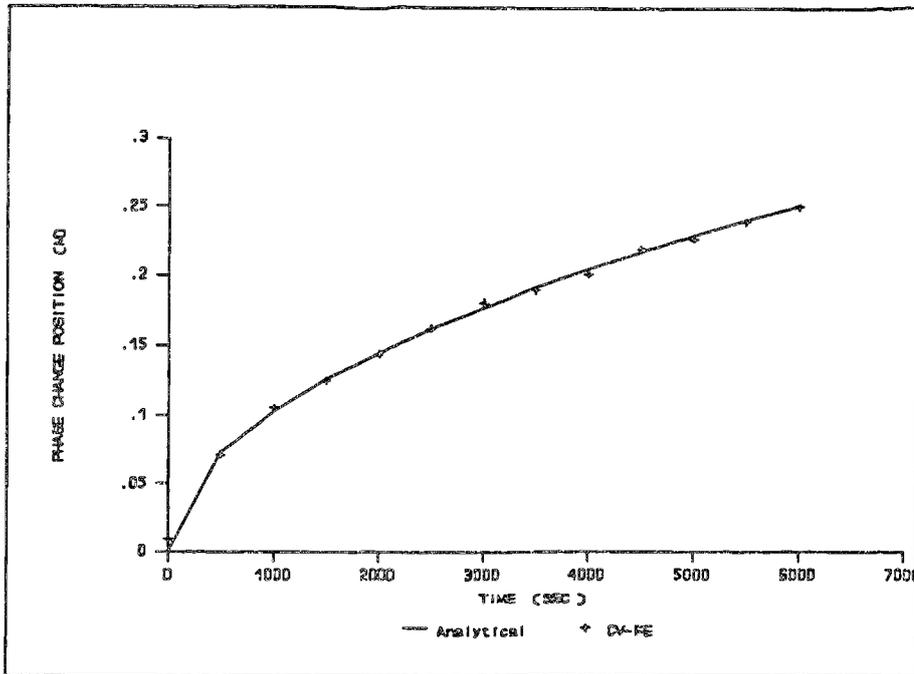


Figure 4.12 CV-FE 21x21 mesh against the analytic solution.

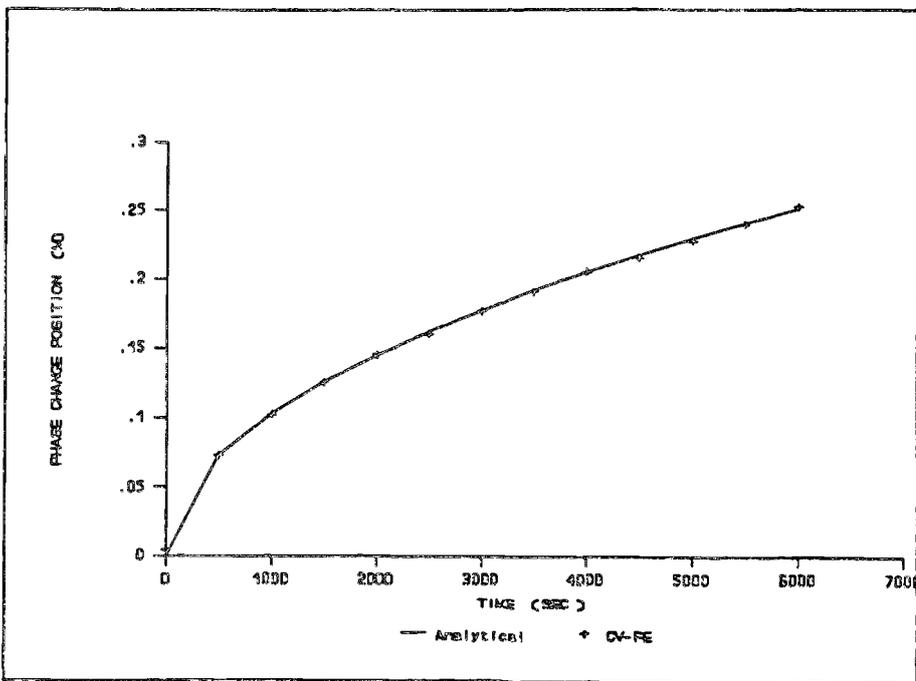


Figure 4.13 CV-FE 41x41 mesh against the analytic solution.

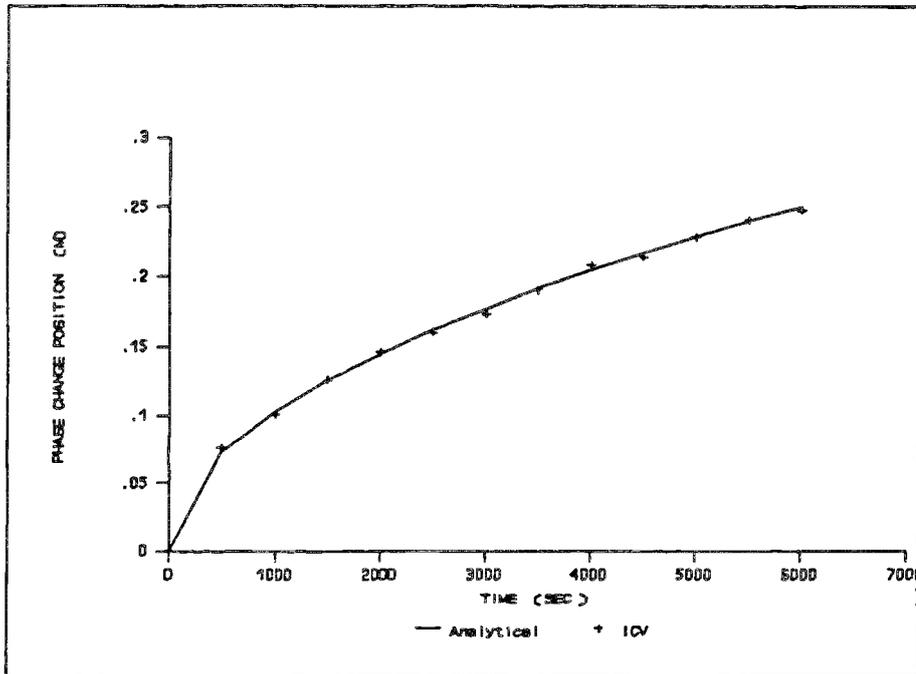


Figure 4.14 ICV 21x21 mesh against the analytic solution.

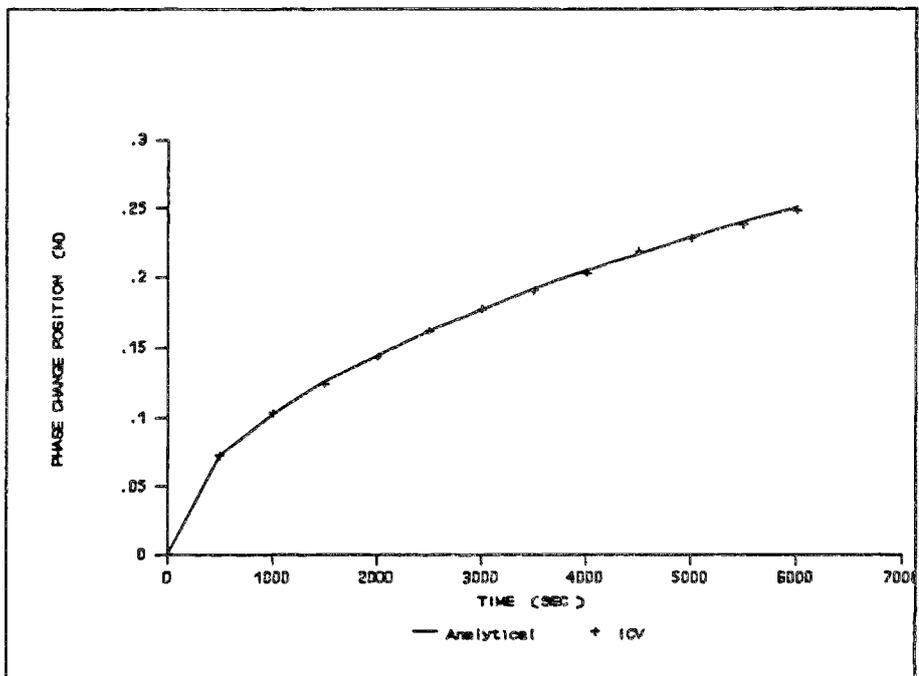


Figure 4.15 ICV 41x41 mesh against the analytic solution.

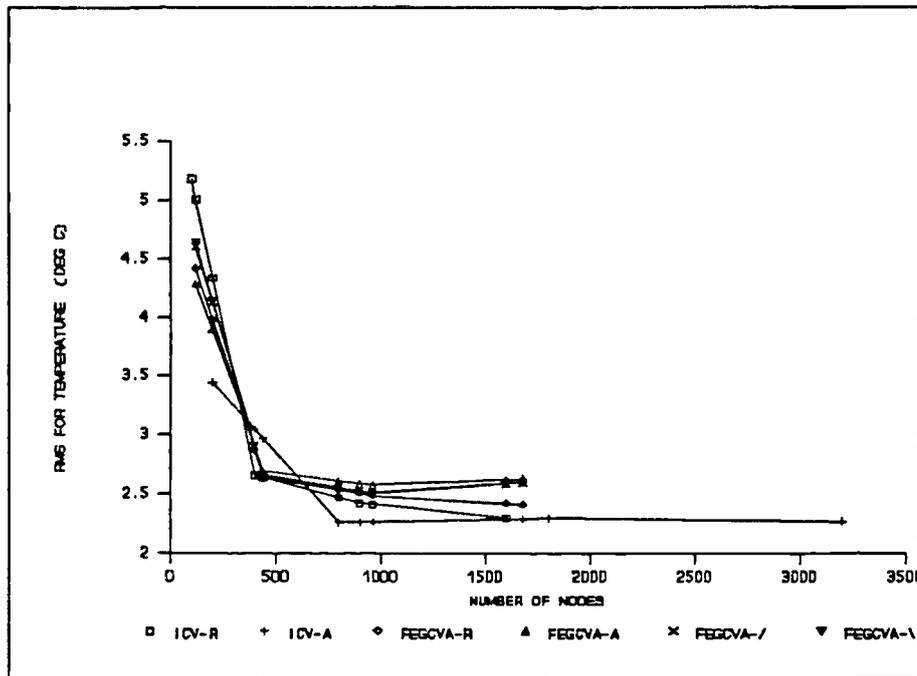


Figure 4.16 RMS for the various mesh densities.

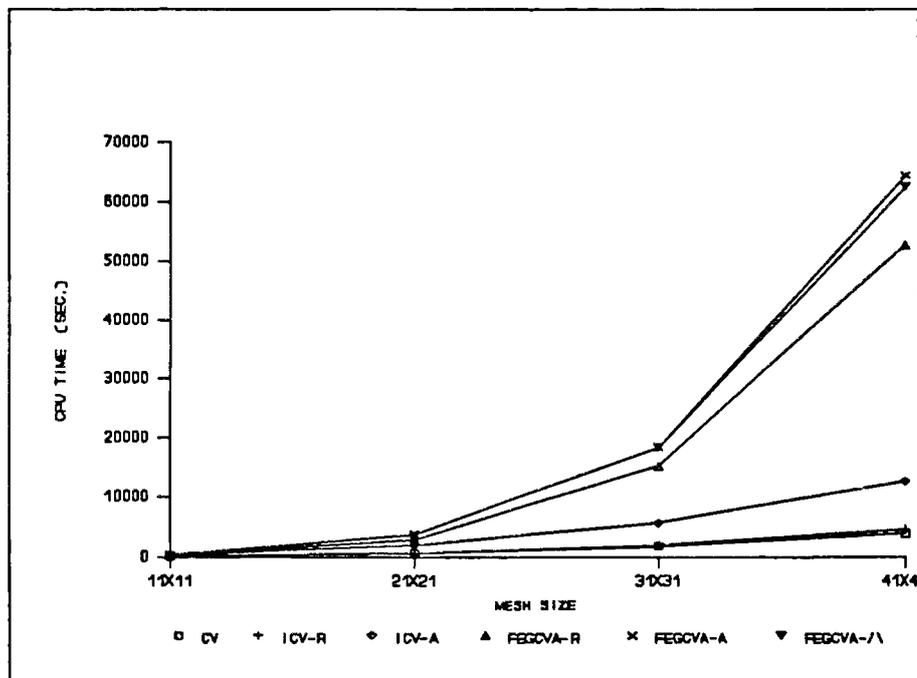


Figure 4.17 Total CPU time for the various meshes.

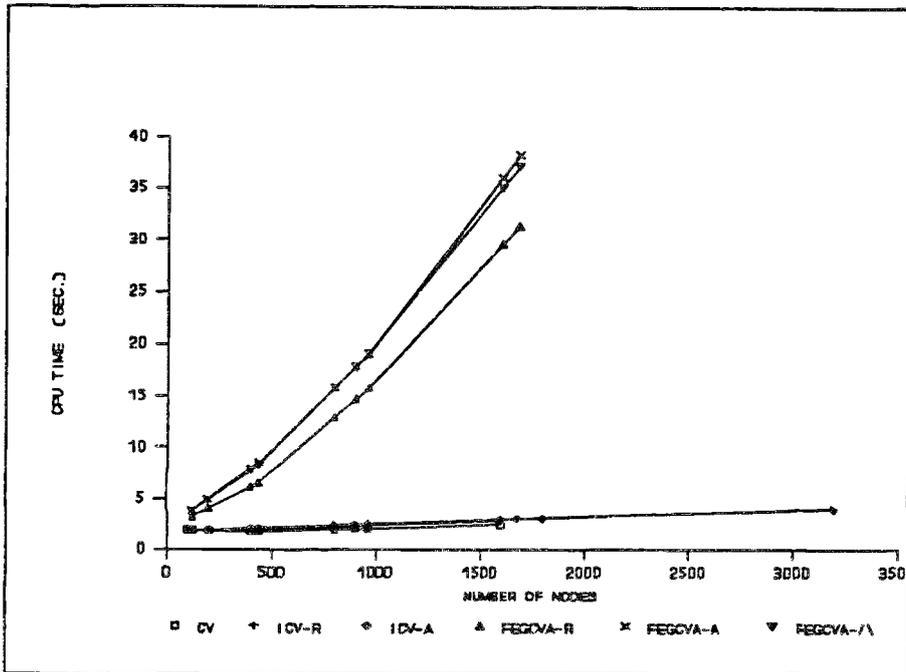


Figure 4.18 CPU cost per node with refined meshes.

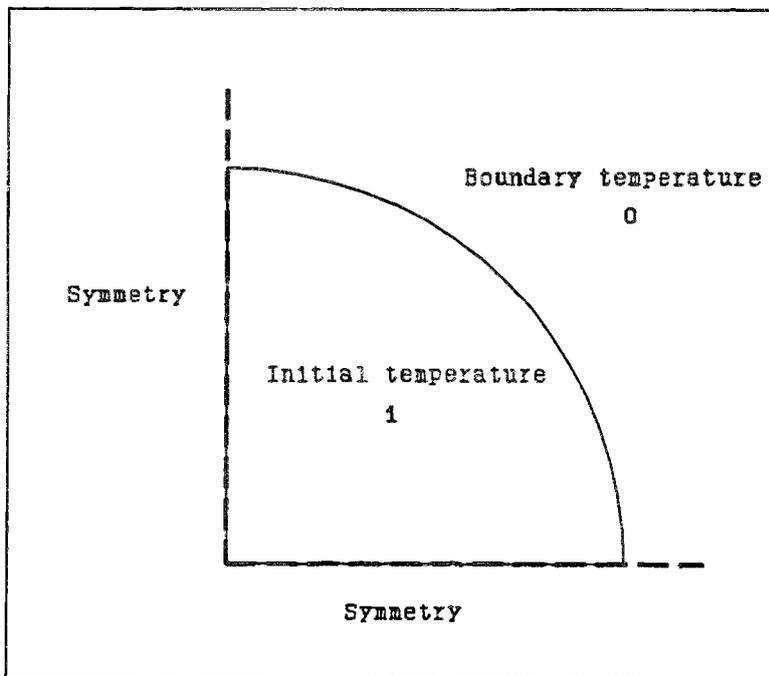


Figure 4.19 Geometrical specification of Tao test case.

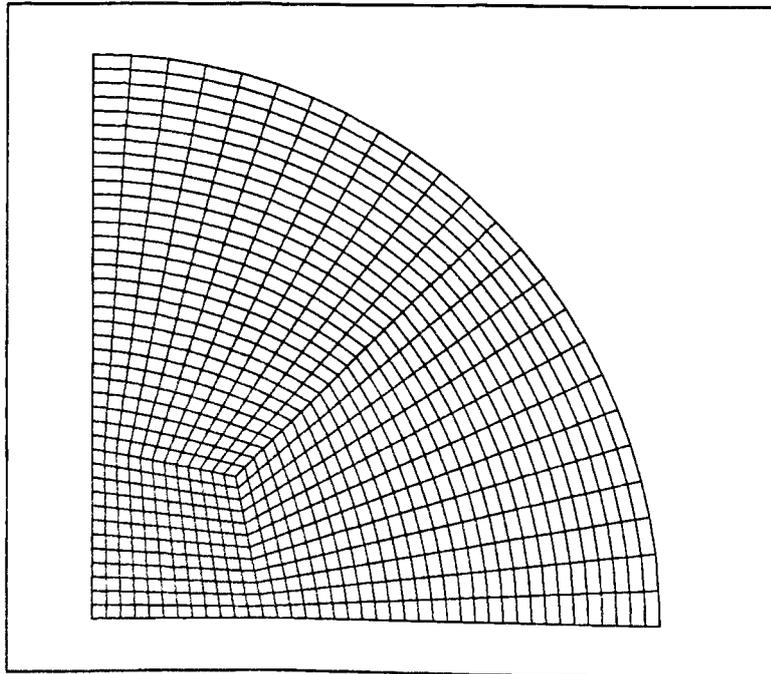


Figure 4.20 Quadrilateral mesh for Tao test case.

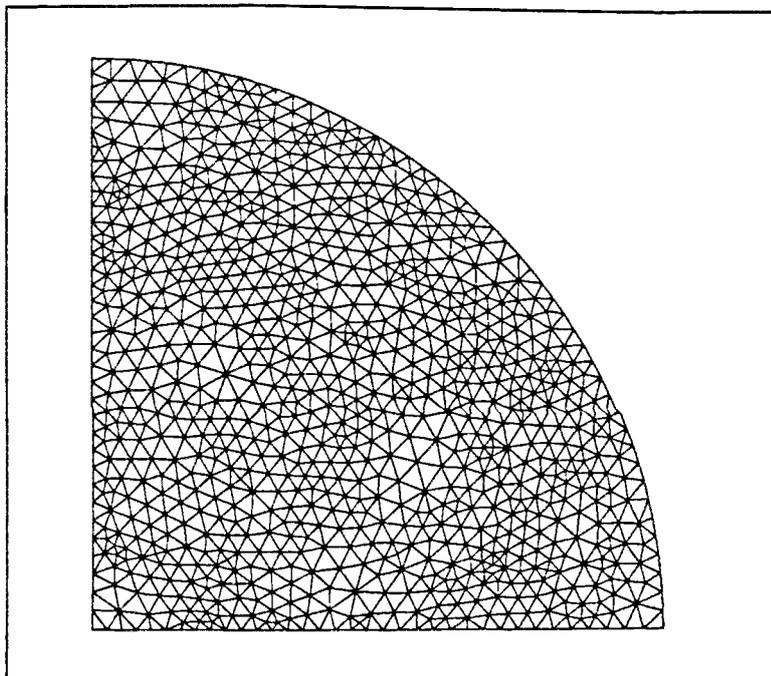


Figure 4.21 Triangle mesh for Tao test case.

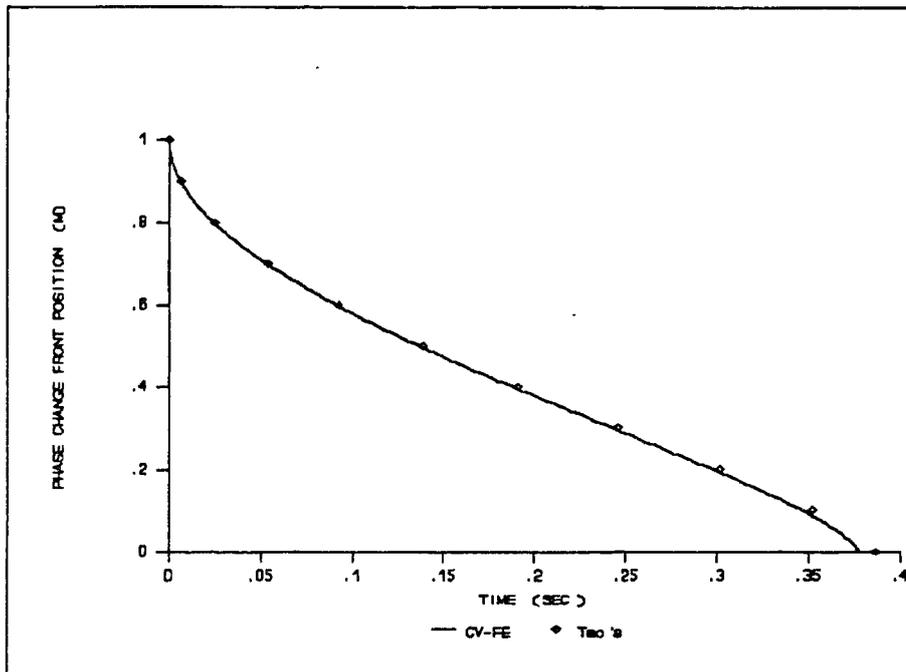


Figure 4.22a CV-FE (quadrilateral) result with those of Tao's.

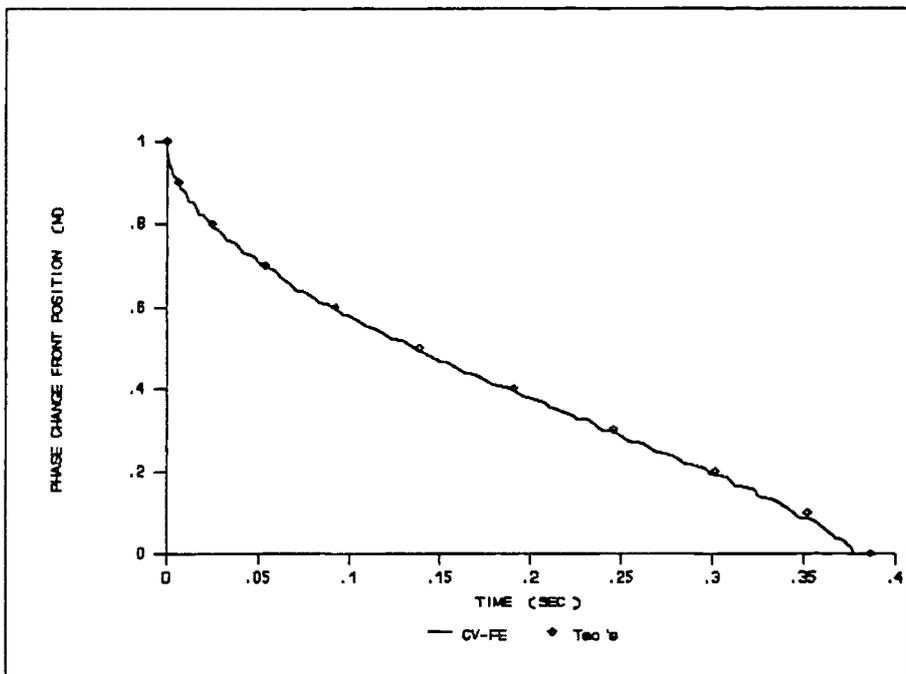


Figure 4.22b CV-FE (Triangle) result with those of Tao's.

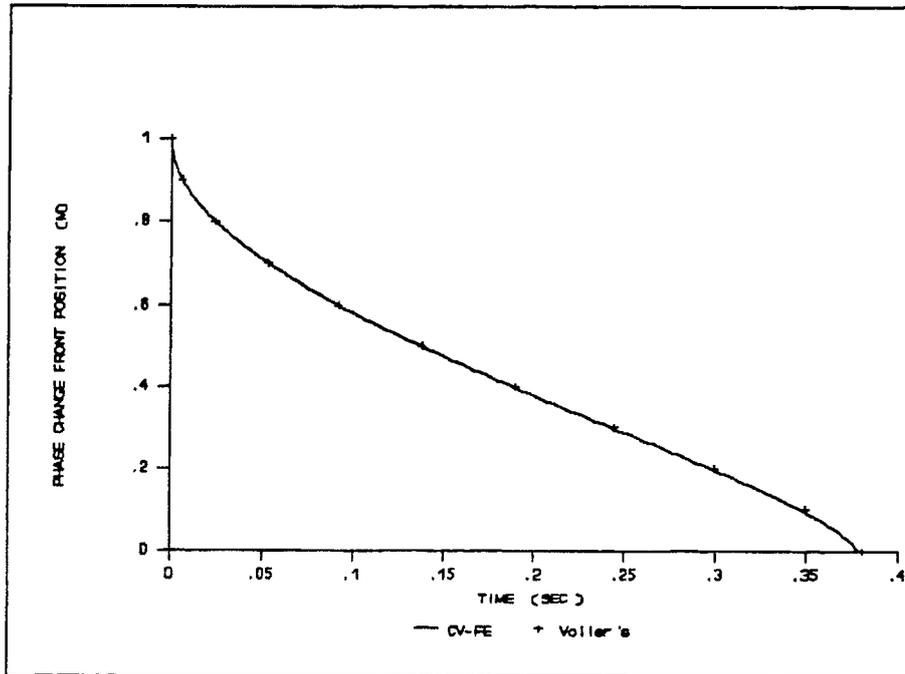


Figure 4.23a CV-FE (quadrilateral) result with those of Voller's.

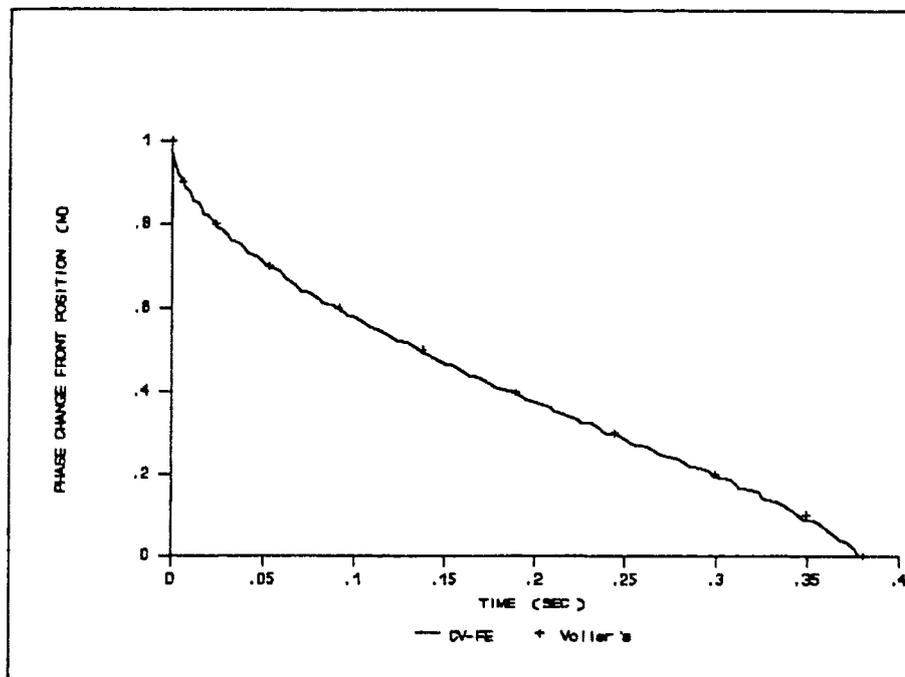


Figure 4.23b CV-FE (Triangle) result with those of Voller's.

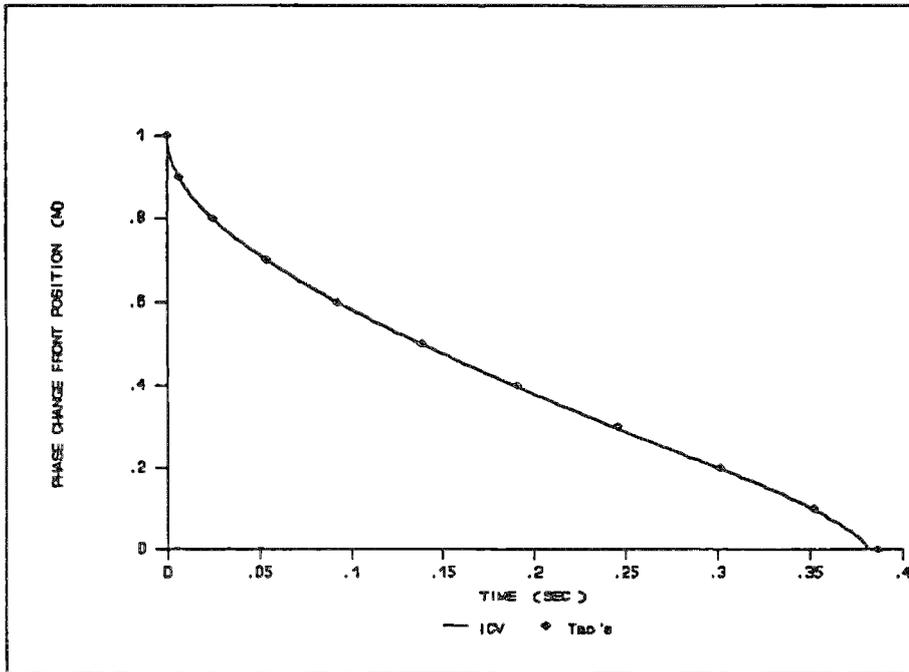


Figure 4.24a ICV (quadrilateral) result with those of Tao's.

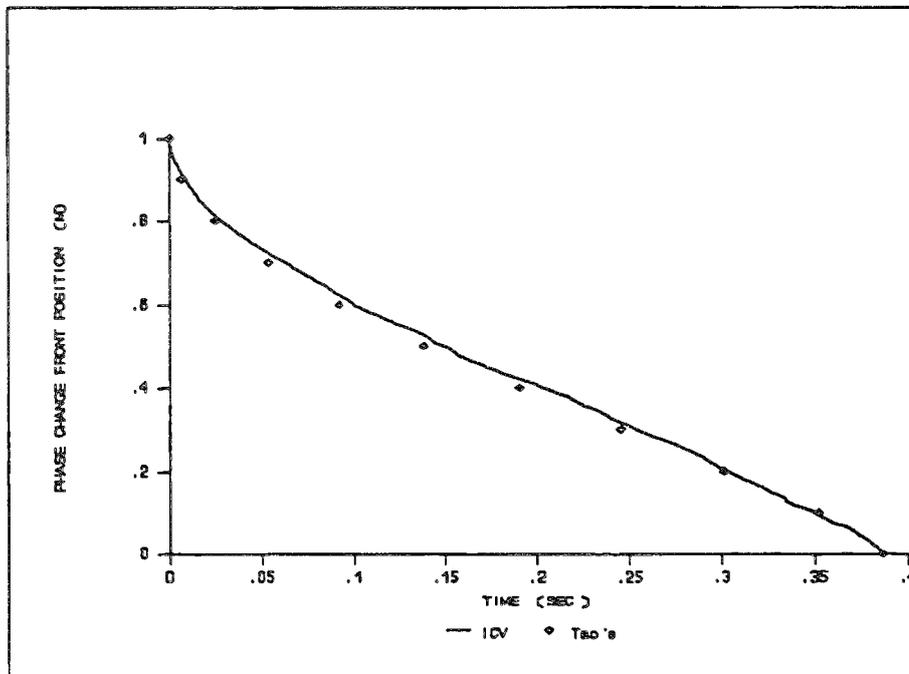


Figure 4.24b ICV (Triangle) result with those of Tao's.

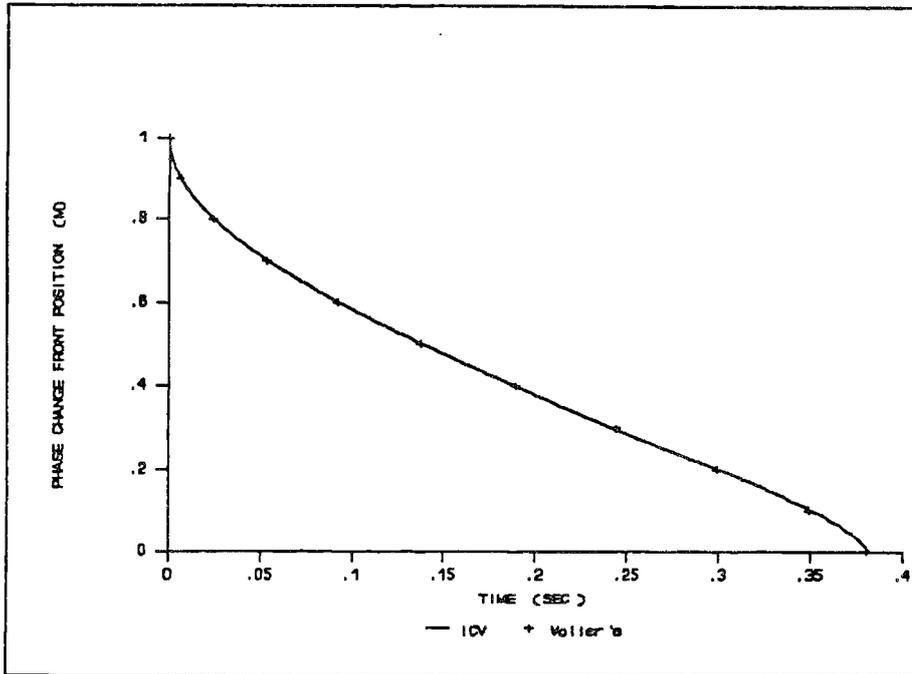


Figure 4.25a ICV (quadrilateral) result with those of Voller's.

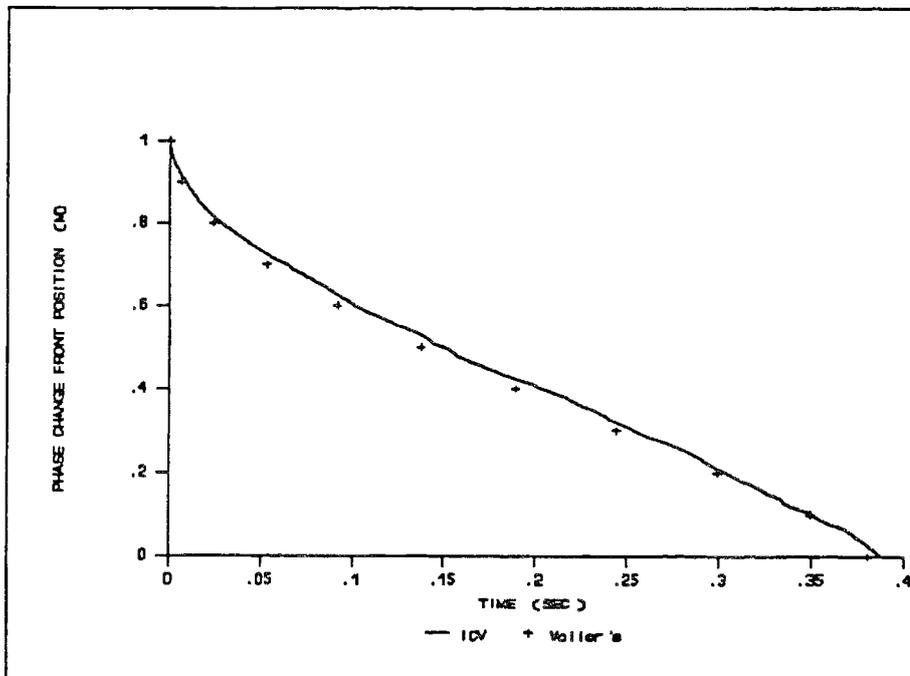


Figure 4.25b ICV (Triangle) result with those of Voller's.

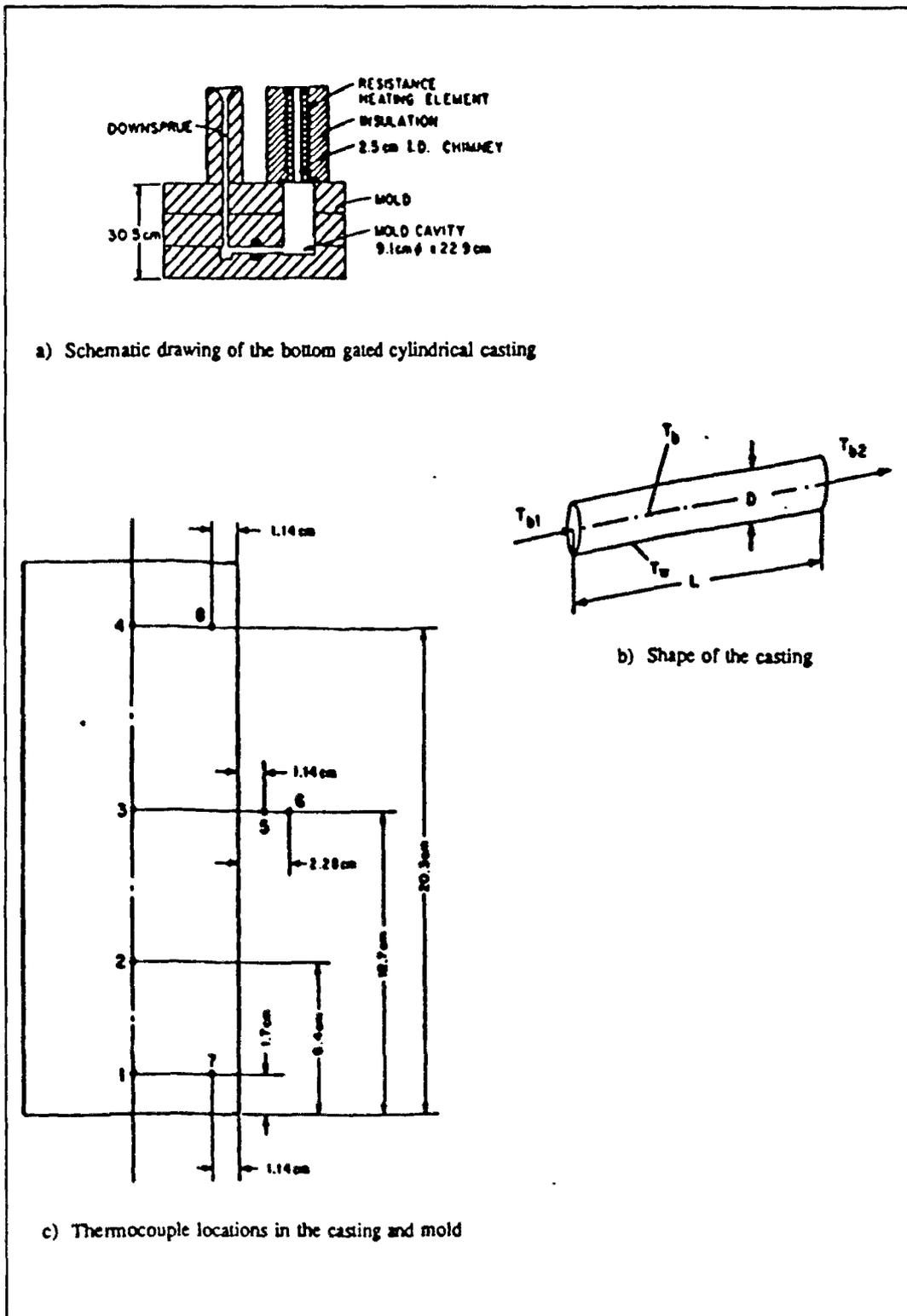


Figure 4.26 Key feature of Zeng and Pehlke casting experimental setup.

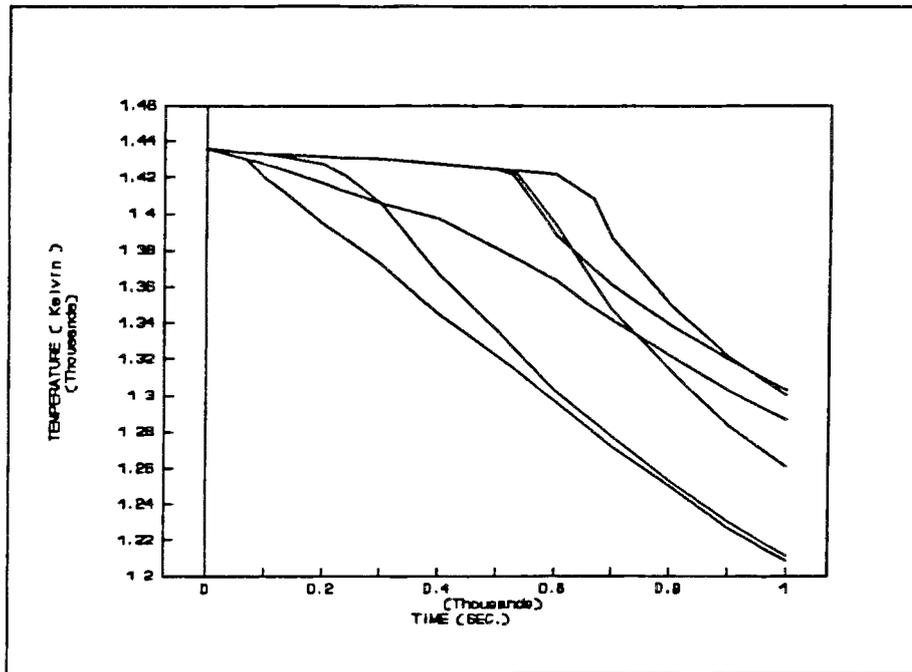


Figure 4.27 Experimental results of Zeng and Pehlke.

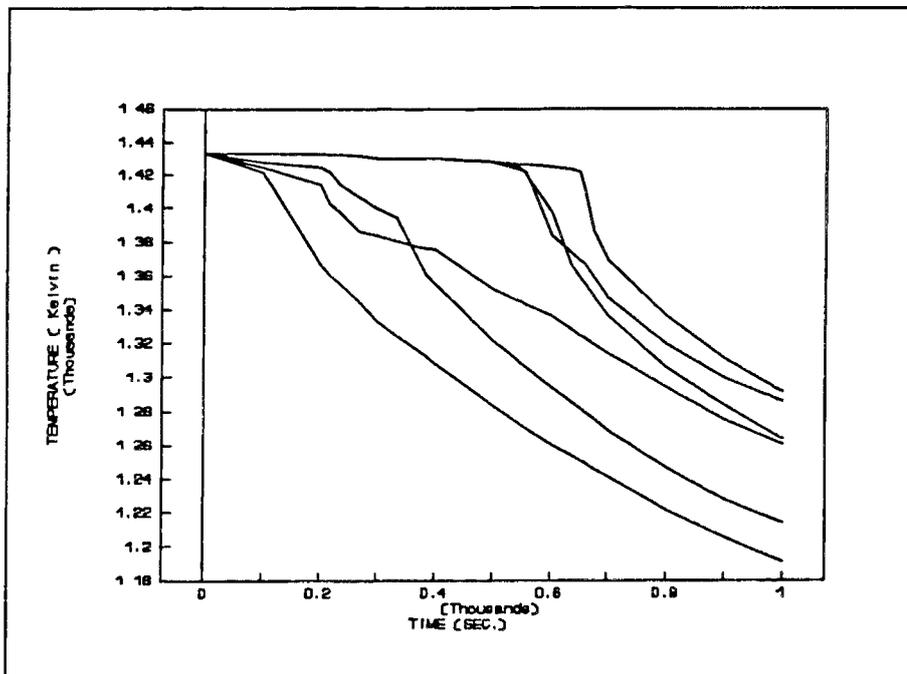


Figure 4.28 Numerical results of Zeng and Pehlke.

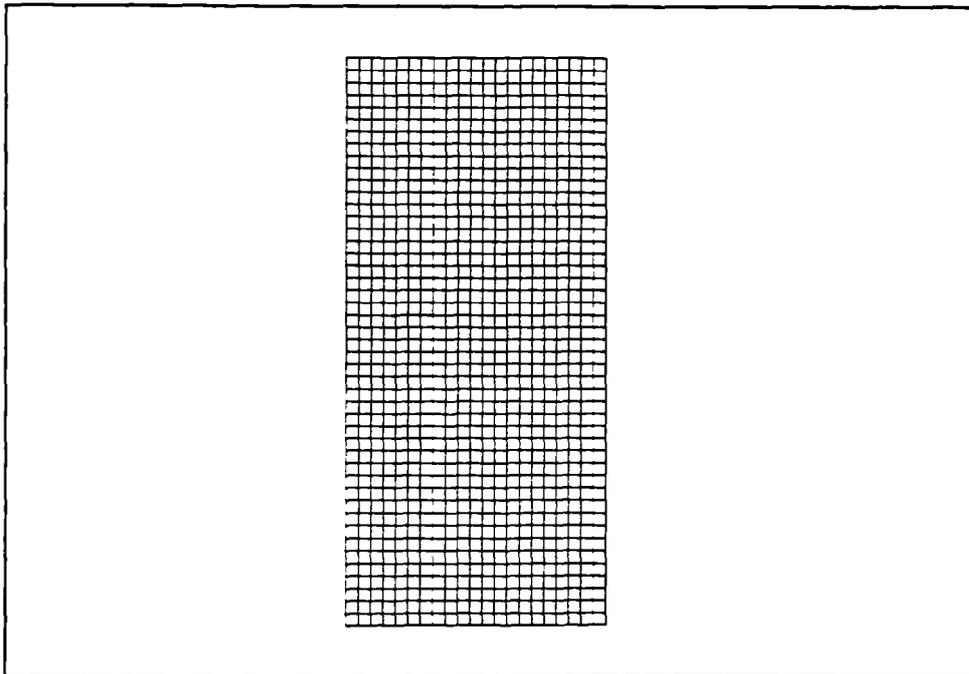


Figure 4.29 Quadrilateral mesh for the Zeng and Pehlke problem.

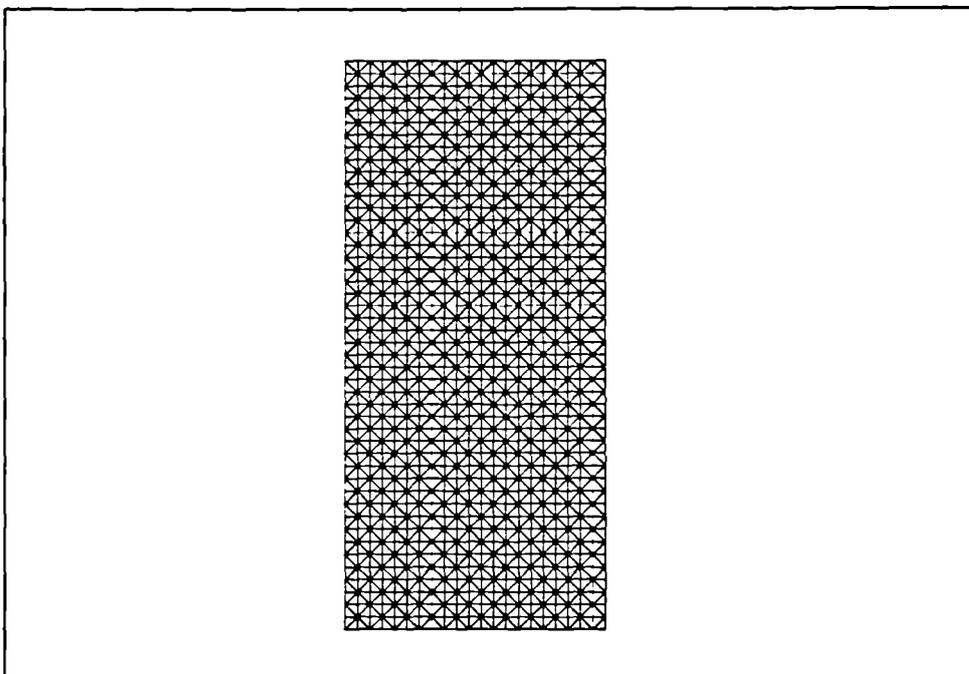
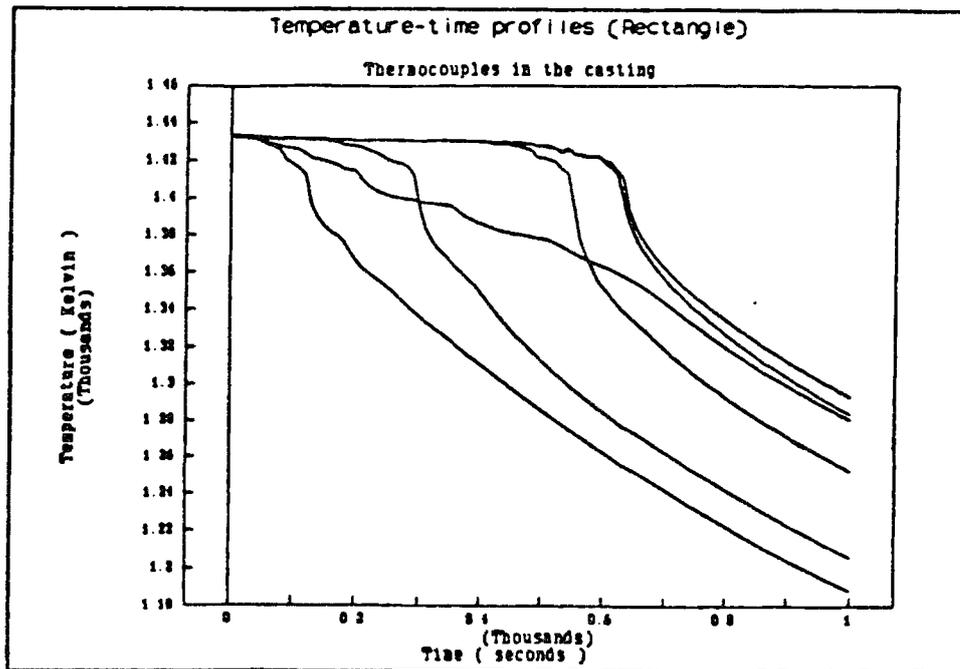
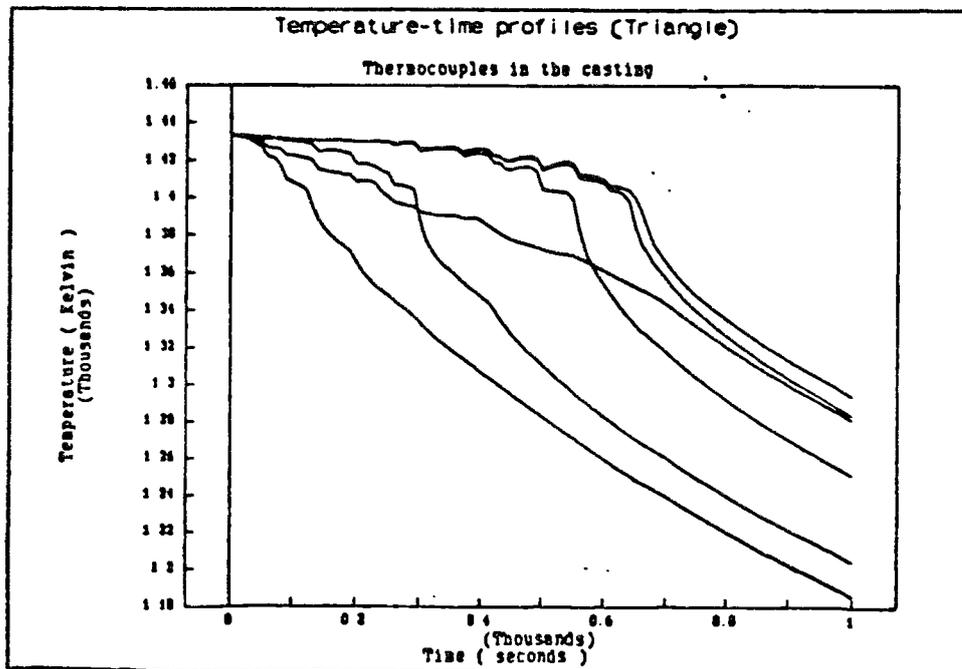


Figure 4.30 Triangle mesh for the Zeng and Pehlke problem.



a) Temperature profiles for quadrilateral mesh.



b) Temperature profiles for triangle mesh.

Figure 4.31 CV-FE initial implementation on the Zeng and Pehlke problem.

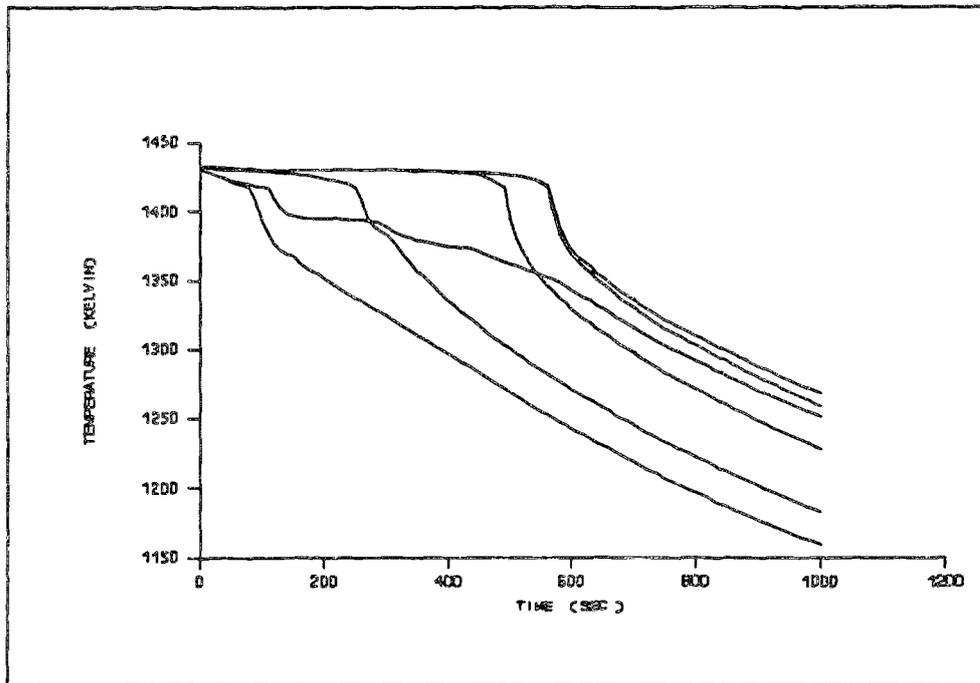


Figure 4.32 CV-FE temperature profiles for quadrilateral mesh (Sand).

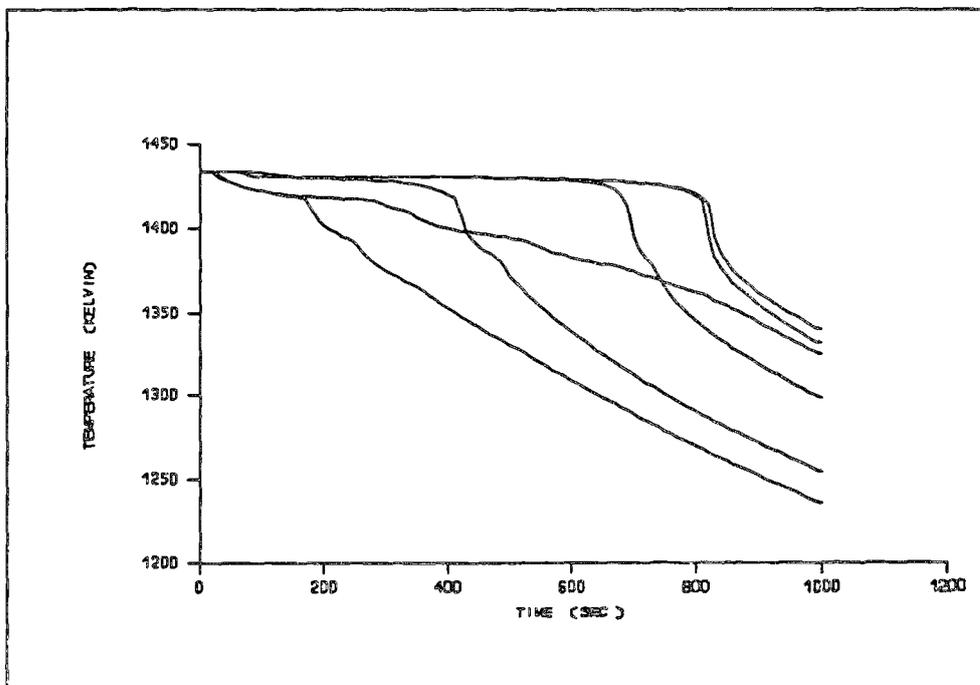


Figure 4.33 CV-FE temperature profiles for triangle mesh (Iron).

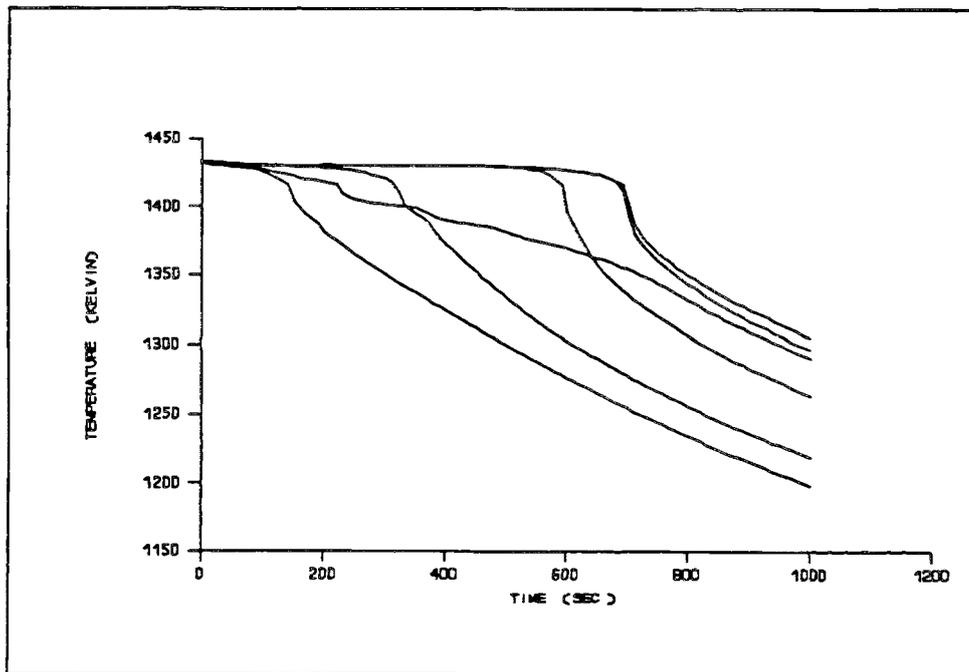


Figure 4.34 ICV temperature profiles for quadrilateral mesh.

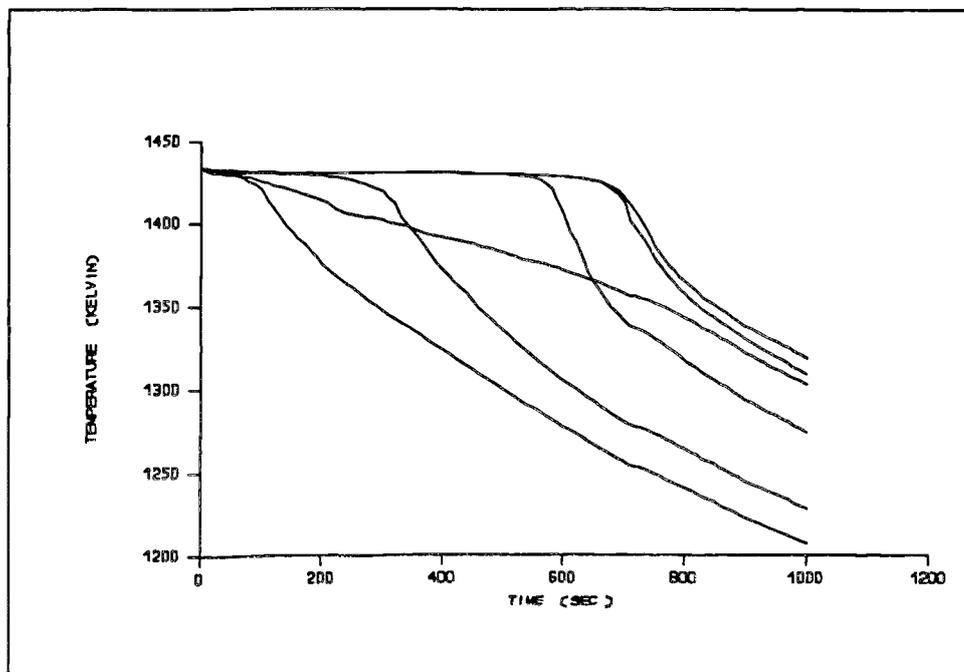


Figure 4.35 ICV temperature profiles for triangle mesh.

Chapter 5

5.0 The full convective-diffusive solidification model	110
5.1 Introduction	110
5.2 Governing equations for fluid flow and heat transfer	111
5.2.1 Momentum equations	112
5.2.2 Continuity equation	113
5.2.3 Energy equation	113
5.2.4 The general conservation equation	113
5.3 Domain discretisation	114
5.4 Discretisation of the governing equations with solidification	115
5.4.1 General equation	115
5.4.2 Momentum equations	118
5.4.3 Energy equation	121
5.4.4 Continuity equation	122
5.4.5 Pressure-correction equation	123
5.5 Boundary conditions	125
5.5.1 Estimation of the boundary pressure.	126
5.5.2 Pressure correction gradient.	128
5.6 Solution procedure	129
5.7 Non-staggered grid	130
5.7.1 The Rhie and Chow Interpolation method.	131

5.8 Closure 134

Figures 135

5.0 The full convective-diffusive solidification model

5.1 Introduction

In this chapter, the Irregular Control Volume (ICV) method described in *Section 3.3* for heat conduction only, is now extended to solve the full convective-diffusive solidification equations. These are primarily the fluid flow and energy equations solved using the SIMPLE algorithm of Patankar and Spalding (1972).

The new ICV method is an extension of the successful Control Volume (CV) method as given by Patankar (1980). The extension is made in the discretisation of a Finite Element (FE) mesh, which will be referred as unstructured, in a way that is typically control volume. This is why, with a rectangular (structured) mesh, the ICV is no different from the CV method. With the new method, computations can be performed on complex geometries without the need for structured or block meshes, as in the Body Fitted Co-ordinates (BFC) method, where curvilinear co-ordinate transformation is used to map between the physical and the structured solution domain. Local mesh refinement of the type previously confined solely to finite elements is now possible with the new method.

One added bonus with the ICV is the creation of "high-order cell" control volumes, ie. control volumes having more than the standard four cell faces in two-dimensions and six in three-dimensions. The results obtained using these high-order cells are showing the same characteristics as the high-order flow directed schemes used to reduce

numerical smearing in structured meshes.

In any successful fluid flow calculation, the pressure oscillation caused by the close coupling between the velocity and pressure needs to be suppressed. The traditional CV technique uses the staggered grid approach (velocity components and pressure stored at different locations) to remove the pressure oscillation. This means a total of three different meshes in two-dimensions and four in three. This staggered arrangement is not so easily adaptable to unstructured meshes. Instead a non-staggered grid approach (all variables stored at the control volume centre), resulting in a single mesh is more appropriate under the circumstances. To remove the oscillation some special technique in form of Rhie and Chow (1982) velocity interpolation is required.

The full description of the ICV discretisation is given for the fluid flow and heat transfer equations, as applied to solidification. Other essential components required for a successful calculation are also detailed; these are, the derivation of the pressure-correction equation for a non-staggered grid, the Rhie and Chow interpolation for a cell face velocity and estimating the boundary pressure. All these are described within this chapter.

5.2 Governing equations for fluid flow and heat transfer

The full convective-diffusive solidification model is primarily concerned with solving

the equations governing fluid flow and heat transfer, with added terms introduced to simulate the solidification processes.

When the laws governing the physical phenomena of fluid flow and heat transfer have been expressed in mathematical form, they are usually in terms of partial differential equations. It will be shown that all these equations can be generalised into a single general differential equation, which expresses the transport of any conserved variable in a flow field.

5.2.1 Momentum equations

The differential equation governing the conservation of momentum in a two-dimensional cartesian co-ordinate system for an incompressible fluid can be written as

$$\frac{\partial(\rho u)}{\partial t} + \nabla \cdot (\rho \underline{V} u) = \nabla \cdot (\mu \nabla u) - \frac{\partial p}{\partial x} + S_u \quad 5.1$$

$$\frac{\partial(\rho v)}{\partial t} + \nabla \cdot (\rho \underline{V} v) = \nabla \cdot (\mu \nabla v) - \frac{\partial p}{\partial y} + S_v \quad 5.2$$

where μ is the viscosity, ρ is the density, p is the pressure, \underline{V} is the resultant velocity, S_u and S_v are the sources for the x and y direction respectively. The u and v are the cartesian velocity components in the respective direction.

5.2.2 Continuity equation

For mass conservation in the flow field an additional equation, namely, the continuity equation should be satisfied, the equation is

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \underline{V}) = 0 \quad 5.3$$

5.2.3 Energy equation

The general conservation of energy equation can be expressed as

$$\frac{\partial(\rho h)}{\partial t} + \nabla \cdot (\rho \underline{V} h) = \nabla \cdot (k \nabla T) + S_h \quad 5.4$$

where h is the specific enthalpy, k is the thermal conductivity, T is the temperature, and S_h is the source term for the volumetric rate of heat generation. For the equation to be solely in terms of enthalpy, $h = cT$, ie. the product of specific heat, c , and temperature [Patankar (1980)], it can be rewritten as

$$\frac{\partial(\rho h)}{\partial t} + \nabla \cdot (\rho \underline{V} h) = \nabla \cdot (k \nabla (h/c)) + S_h \quad 5.5$$

5.2.4 The general conservation equation

From the differential equations seen so far, all the dependent variables of interest appear

to obey a generalised conservation form. If we denote the conserved dependent variable to be ϕ as described fully by Patankar (1980), the general differential equation can be expressed as

$$\frac{\partial(\rho\phi)}{\partial t} + \nabla \cdot (\rho \underline{V}\phi) = \nabla \cdot (\Gamma_{\phi} \nabla \phi) + S_{\phi} \quad 5.6$$

where Γ_{ϕ} and S_{ϕ} are the diffusion coefficient and the source term respectively, and are specific to a particular meaning of ϕ .

The four terms in the general differential equation 5.6 are, from left to right, the transient term, the convection term, the diffusion term, and the source term. The dependent variables ϕ in this instant are the two velocity components and enthalpy, but other dependent variables, such as turbulence-kinetic-energy can be represented by ϕ .

5.3 Domain discretisation

The full description of the domain discretisation has already been given in *Section 3.3.2* and will not be repeated here. The essence being that the nodes are defined at the centres of the control volumes (element in a FE mesh), where also all the unknowns, the material properties and other information are stored.

Holding the pressure and the velocity components at the same location cause the well

known numerical phenomenon of pressure oscillation. A special formulation is needed to prevent this from occurring as will be addressed in a section 5.7. For now, such method reduces the number of meshes required to one as opposed to three in two-dimensions (four in three-dimensions) necessary for the staggered approach to preventing the oscillation.

5.4 Discretisation of the governing equations with solidification

5.4.1 General equation

Integrating the general differential equation 5.6 over an arbitrary irregular control volume gives

$$\iiint_V \frac{\partial(\rho\phi)}{\partial t} dV + \int_S \rho \underline{V} \phi ds_i = \int_S \Gamma_\phi \frac{\partial\phi}{\partial n_i} ds_i + \iiint_V S_\phi dV \quad 5.7$$

where s_i represent the components of the outward normal area vector, with $ds_1 = dy$ and $ds_2 = -dx$ in counterclockwise traversal of the control volume boundary, and n_i is the co-ordinate direction, in which $n_1 = x$ and $n_2 = y$.

The terms in equation 5.7 are evaluated as follows:

The transient term

$$\iiint_V \frac{\partial(\rho\phi)}{\partial t} dV = \frac{(\rho\phi)_P - (\rho\phi)_P^o}{\Delta t} V_P \quad 5.8$$

where P denotes the arbitrary irregular control volume under consideration, the

superscript o denotes the old time step value, V_p is the volume of the irregular control volume, and Δt is the time step.

The source term

The source S_ϕ can be expressed in more general form [Patankar (1980)]:

$$S_\phi = S_C + S_P \phi \quad 5.10$$

If a source is non-linear in ϕ , it can be appropriately linearised [Patankar (1980)] and cast in form of equation 5.10, where the values of S_C and S_P are to prevail over the irregular control volume.

The diffusion term

$$\int_S \Gamma_\phi \frac{\partial \phi}{\partial n_i} ds_i = \sum_{A=1}^{N_s} (\phi_A - \phi_P) \left(\Gamma_\phi \hat{n} \left(\frac{\Delta y}{\delta x} - \frac{\Delta x}{\delta y} \right) \right)_A \quad 5.11$$

where N_s is the total number of control faces, A is the adjacent control volume that shares a common face with the control volume, P . The symbol \hat{n} is the cross flux unit normal to the cell face, Δx and Δy are the face surface areas and δx and δy are the distances between the node A and P in each component. The $()_A$ implies the variables inside the bracket are to be evaluated with each adjacent control volume. Note, as already explained in Section 3.3.2 the cross-diffusion term has been left out.

The convection term

$$\int_S \rho \underline{V} \phi \, dn_i = \sum_{A=1}^{N_s} (\rho \underline{V} \phi (\Delta y - \Delta x) \hat{n})_A \quad 5.12$$

where \underline{V} is the velocity vector in form of, $\underline{V} = u \hat{i} + v \hat{j}$, and the value of ϕ at the face is calculated by the upwind-differencing scheme.

To express the total convective-diffusive cross flux across a face in the same format as the standard CV method, we now have

$$\begin{aligned} a_A &= D_A + \max[0, -C_A] \\ C_A &= \rho \underline{V} \cdot \underline{S} \quad D_A = \Gamma_\phi \frac{\underline{S} \cdot \underline{N}}{|\underline{N}|^2} \end{aligned} \quad 5.13$$

where D_A and C_A are the diffusive and convective parts respectively, \underline{S} is the outward normal surface vector, with $\underline{S} = \Delta y \hat{i} - \Delta x \hat{j}$, and \underline{N} is the nodal distance vector, with $\underline{N} = \delta x \hat{i} + \delta y \hat{j}$. The generalised convection-diffusion formulation employing some basic schemes given by Patankar (1980) can now be added to equation 5.13.

$$a_A = D_A F(|P_A|) + \max[0, -C_A] \quad 5.14$$

where P_A is the Peclet number, given by C_A / D_A and $F(|P_A|)$ is the generic function for the various differencing schemes that can be employed.

By summing all the adjacent contributions for an irregular control volume P , in form of equation 5.14, and substituting it with 5.7 and 5.8 into equation 5.6 yields a set of

equations in form of

$$a_P \phi_P = \sum_{A=1}^{N_s} a_A \phi_A + b_P \quad 5.15$$

where

$$a_P = \sum_{A=1}^{N_s} a_A + \frac{\rho_P}{\Delta t} V_P$$

$$b_P = \frac{(\rho \phi)_P^o}{\Delta t} V_P + S_\phi V_P$$

With equation 5.15, the dependent variable ϕ can be solved for with any suitable solver.

Note, for a structured mesh, the system of equations, $[A]\underline{\phi} = \underline{b}$, is identical with the one produced by standard CV formulations.

For the dependent variables considered here, u , v , and h , the Γ_ϕ and S_ϕ values will be described in turn as applies to each variable, in association with the solidification model.

5.4.2 Momentum equations

Integrating the momentum equations 5.1 and 5.2 over an arbitrary irregular control volume results in the following equations.

$$\iint_V \frac{\partial(\rho u)}{\partial t} dV + \int_S \rho \underline{V} u ds_i = \int_S \frac{\partial u}{\partial n_i} ds_i - \int_S p ds_1 + \iint_V S_u dV \quad 5.16$$

$$\iint_V \frac{\partial(\rho v)}{\partial t} dV + \int_S \rho \underline{V} v ds_i = \int_S \frac{\partial v}{\partial n_i} ds_i - \int_S p ds_2 + \iint_V S_v dV \quad 5.17$$

The evaluation of the transient, the convection, and the diffusion terms have already been described for the general equation. The Γ value in the diffusion term for the momentum equations is the viscosity value. This leaves the pressure gradient and any other terms such as those specific to the modelling of solidification to be evaluated.

The pressure gradient term in each momentum equation is resolved perpendicular to the velocity component, and can be treated as follows:

$$\int_S p ds_1 = \sum_{A=1}^{N_s} (p \Delta y)_A \quad 5.18$$

$$\int_S p ds_2 = - \sum_{A=1}^{N_s} (p \Delta x)_A$$

In the modelling of the solidification processes, where both the convection and diffusion effects are considered, the momentum equations acquire two extra terms, which have the desired effects for the modelling of the solidification process. They are the buoyancy force and the Darcy resistance terms, which can be treated as source terms in the two momentum equations.

The buoyancy term is to account for the natural convection effects as the material undergoes freezing or melting. The buoyancy source term can be defined in one of two

ways, using a constant or variable density. For constant density, the Boussinesq approximation used is expressed as

$$S_y = -\rho_{ref}\beta g(T - T_{ref}) \quad 5.19$$

where ρ_{ref} is the reference density and is set to the constant density ρ , β is the liquid volumetric thermal expansivity coefficient, g is the gravity in the vertical direction and T_{ref} is the reference temperature. For variable density that is dependent on temperature, the buoyancy source is

$$S_y = \rho(T) g \quad 5.20$$

The gravity force applied via the buoyancy source is in the vertical direction only, therefore, there is no buoyancy source in the u-momentum equation. Alternatively, the gravity can be expressed in the vector format, then the above buoyancy source will apply to the other momentum equations, each with its part of the gravity component.

The Darcy source term is used to suppress or initiate the velocity components as the material undergoes a phase change, from a liquid state to a solid state or vice-versa. The advantages and disadvantages for using the Darcy source technique as opposed to other methods such as the variable viscosity can be found in the paper by Voller, Cross and Markatos (1987). The Darcy source is generally given as

$$S_u = -\frac{\mu}{K}u \quad S_v = -\frac{\mu}{K}v \quad 5.21$$

where K is the permeability, and is calculated using the Carman-Kozeny equation

[Carman (1937)].

$$K = \frac{f_l^3}{C(1 - f_l)^2} \quad 5.22$$

where f_l is the liquid fraction. The value of C will depend on the morphology of the porous medium [Voller and Prakash (1987)]. In the case of Samonds and Waite (1991) it was taken as $C = 5M_s^2$, where M_s is the volumetric specific area of a typical dendrite arm, the approximation of M_s is done assuming the dendrite is conical in shape.

5.4.3 Energy equation

Integrating the energy equation 5.5 over an arbitrary irregular control volume results in

$$\iiint_V \frac{\partial \rho h}{\partial t} dV + \int_S \rho \underline{V} h ds_i = \int_S k \frac{\partial (h/c_p)}{\partial n_i} ds_i + \iiint_V S_h dV \quad 5.23$$

As is the case with the momentum equations, the term that needs to be evaluated is the source term in the equation for the modelling of solidification. The rest of the terms have already been detailed in the general equation 5.7. The solidification algorithm used is based on the Voller and Cross (1985) enthalpy-based approach with the convective-diffusive model from Voller, Cross and Markatos (1987). The solidification source in their case is

$$S_h = -\frac{\partial (L\rho f_l)}{\partial t} - \nabla \cdot (L\rho \underline{V} f_l) \quad 5.24$$

The first term in the source is the energy source/sink term. In melting, a sink, the energy is supplied to change the state of a material from a solid to a liquid state. For freezing, a source, the reverse process, the energy is released from the liquid medium as it turns to solid.

The second term in the source account for the convective effects arising due to the presence of a mushy region (a mixture of solid and liquid in the area between the solidus and liquidus states) during the phase change. In an isothermal phase change (no mushy region) this term disappears. Integrating the solidification source term over an arbitrary irregular control volume gives

$$\iiint_V S_A dV = -L_P \frac{(\rho f_l)_P - (\rho f_l)_P^o}{\Delta t} V_P - \sum_{A=1}^{N_s} (\rho \underline{V} L f_l (\Delta y - \Delta x) \hat{n})_A \quad 5.25$$

For the convective term, up-winding is applied. Thereby for a given face of the irregular control volume, it can be evaluated as

$$(\rho \underline{V} L f_l (\Delta y - \Delta x) \hat{n})_A = \text{MAX}[0, -C_A](L f_l)_A + \text{MAX}[0, C_A](L f_l)_P \quad 5.26$$

where L is the latent heat coefficient of fusion, and f_l is the liquid fraction indicating the state of the medium inside the control volume.

5.4.4 Continuity equation

Integrating the continuity equation 5.3 over an arbitrary irregular control volume gives

$$\iint_V \frac{\partial \rho}{\partial t} dV + \int_S \rho \underline{V} ds_i = 0 \quad 5.27$$

The terms here have already been evaluated in a generic form, which can be found in *section 5.4.1* under the heading of discretisation of the general equation. In the general equation there exists a dependent variable ϕ ; for continuity, this variable has the value of one.

5.4.5 Pressure-correction equation

The pressure-correction equation is derived from the continuity equation by substituting all the velocity components with the velocity correction formulas. These correction formulas are derived from the following equations

$$p = p^* + p' \quad 5.28$$

$$\begin{aligned} u &= u^* + u' \\ v &= v^* + v' \end{aligned} \quad 5.29$$

For a full detailed explanation of the velocity corrections, see Patankar (1980).

The velocity correction formulas are expressed below for a face i .

$$\begin{aligned} u_i &= u_i^* + \left(\frac{\Delta y}{a^u} \right)_i (p'_P - p'_A) \\ v_i &= v_i^* - \left(\frac{\Delta x}{a^v} \right)_i (p'_P - p'_A) \end{aligned} \quad 5.30$$

Where p' is known as the pressure-correction variable, the dependent variable for the pressure-correction equation. The u_i^* and v_i^* are the "starred" velocities, ie. the guessed velocities at the end of the previous iteration, and a_i^u and a_i^v are the respective u and v coefficients. How those starred velocities and the u and v coefficients are calculated are described fully in *section 5.7*. Here the focus is on obtaining a discretised pressure-correction equation.

From equation 5.13 the convective mass flux for a given face i is

$$\begin{aligned} \rho \underline{V} \cdot \underline{S} &= ((\rho u)_i \hat{i} + (\rho v)_i \hat{j}) \cdot (\Delta y_i \hat{i} - \Delta x_i \hat{j}) \\ &= (\rho u \Delta y)_i - (\rho v \Delta x)_i \end{aligned} \quad 5.31$$

by substituting the expression given in 5.30 and rearranging in terms of p' gives

$$(\rho u^* \Delta x)_i - (\rho v^* \Delta x)_i + \left(\frac{\rho \Delta y^2}{a^u} + \frac{\rho \Delta x^2}{a^v} \right)_i (p'_P - p'_A) \quad 5.32$$

Therefore, for an irregular control volume P , it can be written as

$$a_P p'_P = \sum_{A=1}^{N_s} a_A p'_A + b_P \quad 5.33$$

Where

$$a_A = \left(\left(\frac{\rho \Delta y^2}{a^u} \right)_i + \left(\frac{\rho \Delta x^2}{a^v} \right)_i \right)_A$$

$$a_p = \sum_{A=1}^{N_s} a_A + \frac{\rho_p V_p}{\Delta t}$$

$$b_p = \frac{\rho_p^o V_p}{\Delta t} + \sum_{A=1}^{N_s} \left((\rho u \cdot \Delta y)_i - (\rho v \cdot \Delta x)_i \right)_A$$

With the set of equations in form of 5.33, the pressure-correction can be solved for with any suitable solver, and then used to update the variables in equations 5.28 and 5.29.

5.5 Boundary conditions

For control volumes that have a face coincident with the domain boundary, see *Figure 5.1*, no information is available at that face of the control volume with the numerical solution procedure taken, which makes the equations incomplete and cannot be solved. This is where boundary information is introduced into the equations, to complete the formulation and for it to be solved.

The boundary conditions needed for the convective-diffusive solidification model are those appropriate for an inlet, outlet, wall and symmetry boundary. These are usually specified in terms of external velocity components, u and v , temperature, T , and pressure p . Listed below is the involvement of each variable for each of the boundary types.

<i>Inlet:</i>	Velocity:	Specified velocity components or specified mass flux.
	Temperature:	Specified temperature or specified heat flux.
	Pressure:	Not specified.
<i>Outlet:</i>	Velocity:	Mass flux evaluated using velocity components at cell centre.
	Temperature:	Heat flux evaluated using temperature at cell centre.
	Pressure:	Specified pressure.
<i>Wall:</i>	Velocity:	Zero mass flux, parallel velocity to the face can be specified, zero slip condition.
	Temperature:	Specified temperature or specified heat flux.
	Pressure:	Not specified.
<i>Symmetry:</i>	Velocity:	Zero mass flux, perpendicular component is zero, parallel component has the cell centre value.
	Temperature:	Zero heat flux and face has the temperature at cell centre.
	Pressure:	Face has the pressure at cell centre.

5.5.1 Estimation of the boundary pressure.

In the momentum equations 5.16 and 5.17, the pressure gradient term in each equation requires a face pressure for the inlet and wall boundary conditions, since in both, pressure is not specified. How this face pressure is estimated can have a profound

influence on the overall behaviour of the solution.

The straightforward forward/backward differencing can result in a large error in the pressure gradient term, when buoyancy plays a major role in the calculations. This can be highlighted with a simple cavity problem that is buoyancy driven, with the wall hot on the top, cold at the bottom, and symmetry in both sides, see *Figure 5.2*. The answer to the problem should be zero velocities everywhere, and the pressure gradient in the vertical direction equal the buoyancy value in all the control volumes.

Figure 5.3 shows the plot of buoyancy with pressure gradient in the vertical direction using forward/backward differencing method. The graph shows under/over prediction in the pressure gradient at both the boundaries. *Figure 5.4* shows the same variables being plotted, but with the boundary pressure estimated using the Bernoulli equation. A under/over prediction at both the boundaries remained, but the error is significantly reduced. Both the methods of estimating the pressure value at the boundary will improve with grid refinements.

Note, there will always be a finite error in the pressure gradient for control volumes that coincide with the boundary, due to approximate estimation of the boundary pressure. The staggered grid arrangement has no such problem; no estimation of the boundary pressure is required.

5.5.2 Pressure correction gradient.

With the estimation of the boundary pressure in the pressure gradient term, the need now is to estimate a pressure correction value at the boundary, which is to be used in the pressure correction gradient to update the velocity components. The forward/backward differencing can be used to estimate the pressure correction value when the Bernoulli is used to estimate the pressure, if one wishes to do so.

A better and more consistent way of evaluating the pressure correction is to use the same basic principle that derived the pressure correction equation 5.33. The Bernoulli equation in estimating the boundary pressure is

$$p_B = p_P + \frac{\rho}{2}(V_P^2 - V_B^2) + \rho g \Delta h \quad 5.34$$

with guessed pressure field, p^* and starred velocity V^* the guessed boundary value is

$$p_B^* = p_P^* + \frac{\rho}{2}(V_P^{*2} - V_B^2) + \rho g \Delta h \quad 5.35$$

with the boundary velocity known, subtracting equation 5.35 from 5.34, we have

$$p_B' = p_P' + \frac{\rho}{2}(V_P^2 - V_P^{*2}) \quad 5.36$$

where, $V = V^* + V'$, and $V' = d \nabla p'$. $\nabla p'$ is the pressure correction gradient and is treated like the pressure gradient in equation 5.18, and $d = 1 / a_p$.

By considering the u velocity component case, where the boundary is on the west face,

Figure 5.5.

$$\nabla p' = \sum_{A=1}^{N_s} (p' \Delta y)_A = (p' \Delta y)_e + (p' \Delta y)_B \quad 5.37$$

by substituting the boundary pressure correction of 5.36 into 5.37, we have

$$\nabla p' = (p' \Delta y)_e + (p'_p + \frac{\rho}{2} (2V^* d \nabla p' + (d \nabla p')^2)) \Delta y_B \quad 5.38$$

and by grouping terms, we have

$$-\frac{\rho}{2} d^2 \Delta y_B (\nabla p')^2 + (1 - \rho V^* \Delta y_B d) \nabla p' = (p' \Delta y)_e + p'_p \Delta y_B \quad 5.39$$

which can be solved for directly or iteratively, for use in the velocity corrections.

5.6 Solution procedure

The algorithm used for solving the discretised equations of fluid flow and heat transfer is the Semi-Implicit Method for Pressure-Linked Equations (SIMPLE) by Patankar and Spalding (1972). The sequence of operations for the SIMPLE algorithm with solidification are as follows:

1. Set initial/current values to old time values and guess the pressure field p^* .
2. Solve the discretised momentum equations of 5.16 and 5.17 to obtain u^* and v^* .
3. Solve the discretised pressure-correction equation of 5.33.

4. Calculate p from $p = p^* + p'$.
5. Calculate u and v from their starred values using the formulas in 5.30.
6. Solve the discretised energy equation 5.23.
7. Calculate temperature dependent material properties and liquid fraction.
8. Treat the corrected pressure p as new guessed pressure p^* , return to step 2, and repeat the whole procedure until a converged solution is obtained before advancing a time step.

5.7 Non-staggered grid

The advantages and disadvantages of using a non-staggered grid arrangement (velocity components are calculated and stored at the control volume centres) as opposed to a staggered grid (velocity components are calculated and stored at the control volume faces) can be found in reference [Melaen (1992)]. *Figure 5.6* shows both the non-staggered and staggered grid arrangements for the velocity components in relationship to a pressure control volume.

The deployment of a staggered grid is to remove the pressure oscillation, or what Patankar called the checker-boarding effect [Patankar (1980)], due to the coupling of the pressure and velocity equations. This staggering is to allow the velocity components to lie on the faces of the control volumes where pressure and non-velocity variables are

stored. This results in a grid for all non-velocity variables such as pressure, and a grid for each velocity component. The grids for velocity and non-velocity variables will lead to overlapping control volumes. *Figure 5.6* shows the overlapping that occurs between the velocity and the non-velocity control volumes. With an unstructured grid, the staggered grid arrangement in removing the pressure oscillation is not a viable option. It may be so for a partially staggered one [Chen (1991)], this arrangement will be addressed in *Section 7.2.2*. Nevertheless, the partially staggered grid still requires more than one grid system. For the current work with unstructured meshes, we will be adopting a single grid for clarity and simplicity without the added complications of multiple unstructured meshes.

To remove the pressure oscillation in non-staggered grid system, due to the coupling of the pressure and velocity equations, Rhie and Chow (1982) have derived a method that suppresses the oscillation by interpolating the velocity components on the control volume faces by the velocity components on the centres of the two control volumes that share a common face. The Rhie and Chow interpolation is fully explained in the next section. For consistency, all face velocity components will be estimated in the same way for any calculations involving the convective element.

5.7.1 The Rhie and Chow Interpolation method.

The Rhie and Chow (1982) interpolation method addresses the close coupling of the

pressure and the convective velocity required for a successful non-staggered grid calculation, by interpolation of the velocity components at the control volume faces from those on the control volume centres. The straightforward weighted linear interpolation between the two centre velocity components leads to the well known numerical phenomenon of pressure oscillation. The reason is due to the $2\delta x$ differences for the computation of the pressure gradients at the control volume centres, which effectively decouple the even and odd grid points. Therefore, the momentum equations discretised at the control volume centres are insensitive to $1\delta x$ variations in pressure.

By considering the momentum equation 5.16 in its discretised form and given in a shorthand representation for a control volume P

$$u_P + (d\nabla p)_P = (au)_P + S_P \quad 5.40$$

Consider now the interpolation for the east face of the control volume P , where the east control volume discretised momentum is

$$u_E + (d\nabla p)_E = (au)_E + S_E \quad 5.41$$

Following the conservation principle of the control volume formulation, the east face velocity, u_e , also must have a discretised momentum equation of the form

$$u_e + (d\nabla p)_e = (au)_e + S_e \quad 5.42$$

This is true in the staggered grid case.

The Rhie and Chow interpolation is simply a method of approximating a solution of

5.42 from the discretised equations of 5.40 and 5.41. By assuming the terms on the right hand side of 5.42 may be approximated using weighted linear interpolations by the corresponding terms in equations 5.40 and 5.41, then we have

$$u_e + (d\nabla p)_e = \overline{(au)}_e + \mathcal{S}_e = \bar{u}_e + \overline{(d\nabla p)}_e \quad 5.43$$

The overbar indicate the weighted linear interpolation used for the variable. Therefore, the east face velocity is interpolated to be

$$u_e = \bar{u}_e + \overline{(d\nabla p)}_e - (d\nabla p)_e \quad 5.44$$

Assuming also that $d_e \approx \bar{d}_e$, and $\overline{(d\nabla p)}_e \approx \bar{d}_e \nabla \bar{p}_e$

we arrive at the Rhie and Chow interpolation formula

$$u_e = \bar{u}_e + \bar{d}_e (\nabla \bar{p}_e - \nabla p_e)$$

For completeness, the right hand side of equation 5.45 is evaluated as follows, where α is the weighted factor in the weighted linear interpolation.

$$\bar{u}_e = \alpha u_p + (1 - \alpha) u_E$$

$$\nabla \bar{p}_e = \alpha \nabla p_p + (1 - \alpha) \nabla p_E$$

$$\nabla p_e = \Delta y_e (p_p - p_E)$$

$$a_e = \alpha a_p + (1 - \alpha) a_E$$

$$\bar{d}_e = a_e^{-1}$$

5.8 Closure

The ICV method has been fully described in this chapter for solving the fluid flow and heat transfer equations based on the SIMPLE algorithm of Patankar and Spalding (1972), as it applied to the full convective-diffusive solidification algorithm of Voller, Cross and Markatos (1987).

The associated numerical tools that are essential for a successful calculation when the non-staggered grid is employed, tools such as, the Rhie and Chow interpolation, deriving the pressure-correction equation, pressure boundary approximation plus many other small but essential details are also fully documented in the chapter.

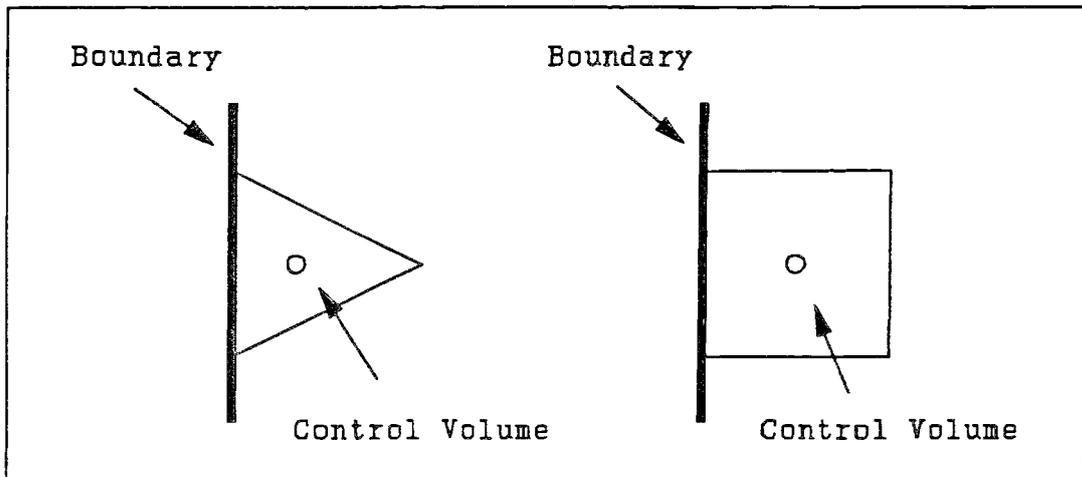


Figure 5.1 Boundary cells.

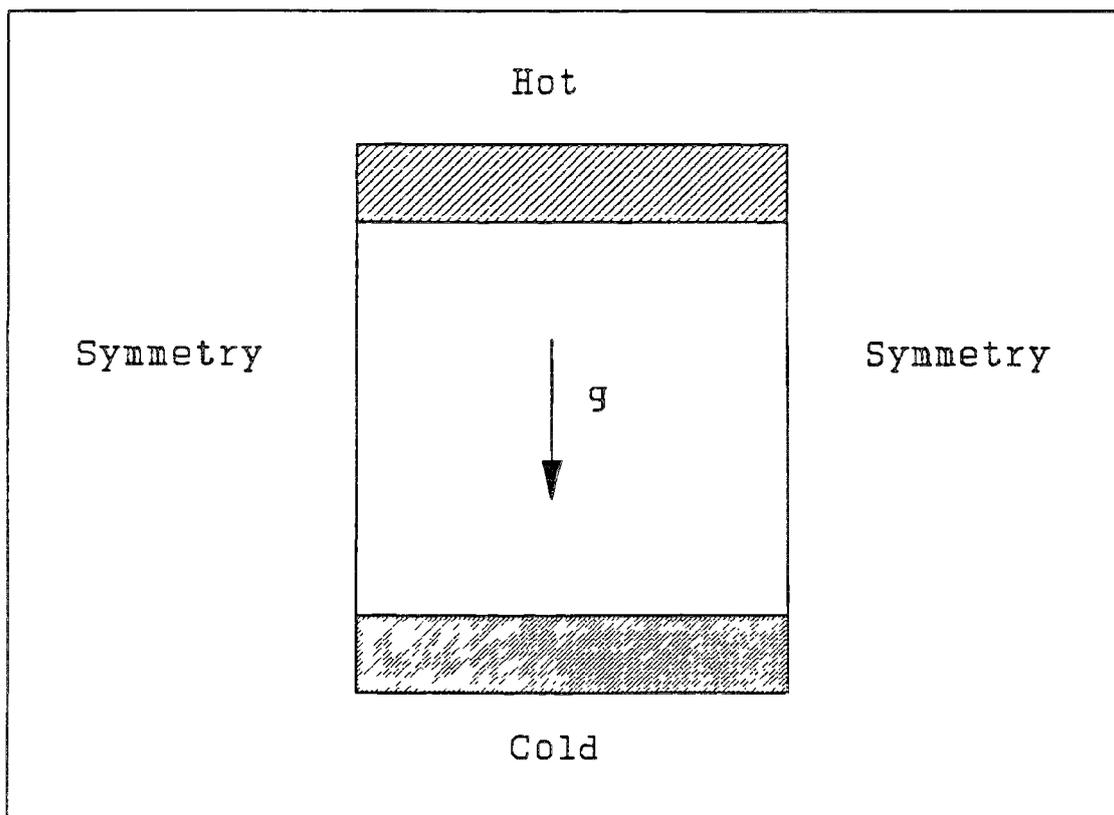


Figure 5.2 Problem specification.

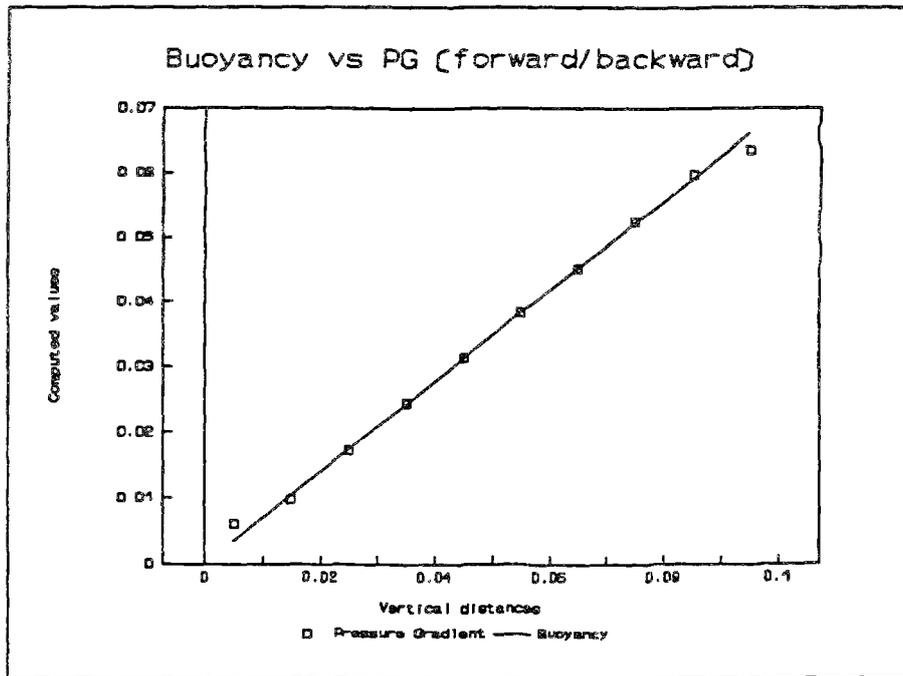


Figure 5.3 Results using forward/backward differencing.

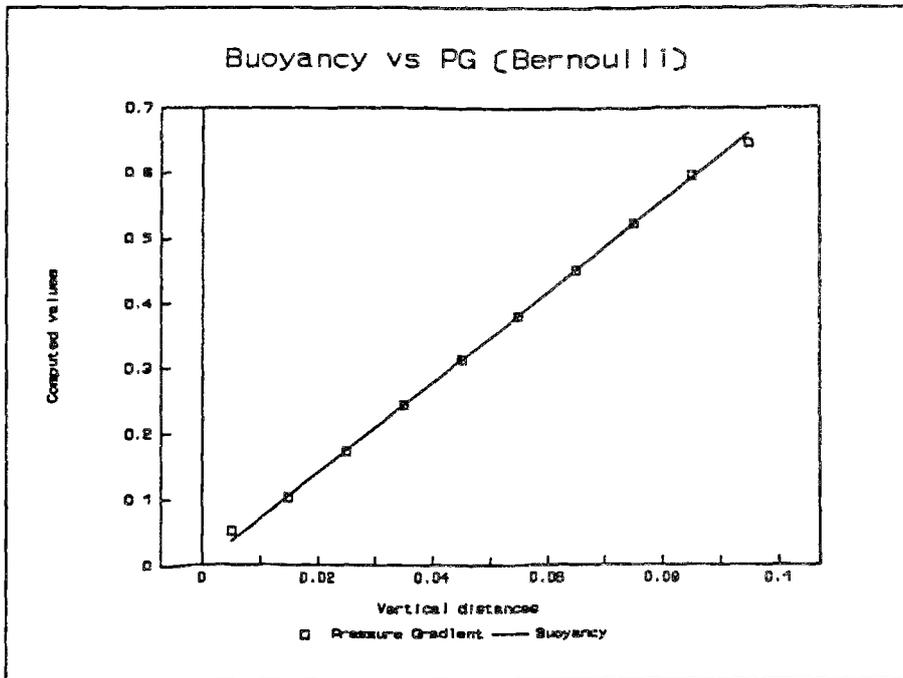


Figure 5.4 Results using Bernoulli.

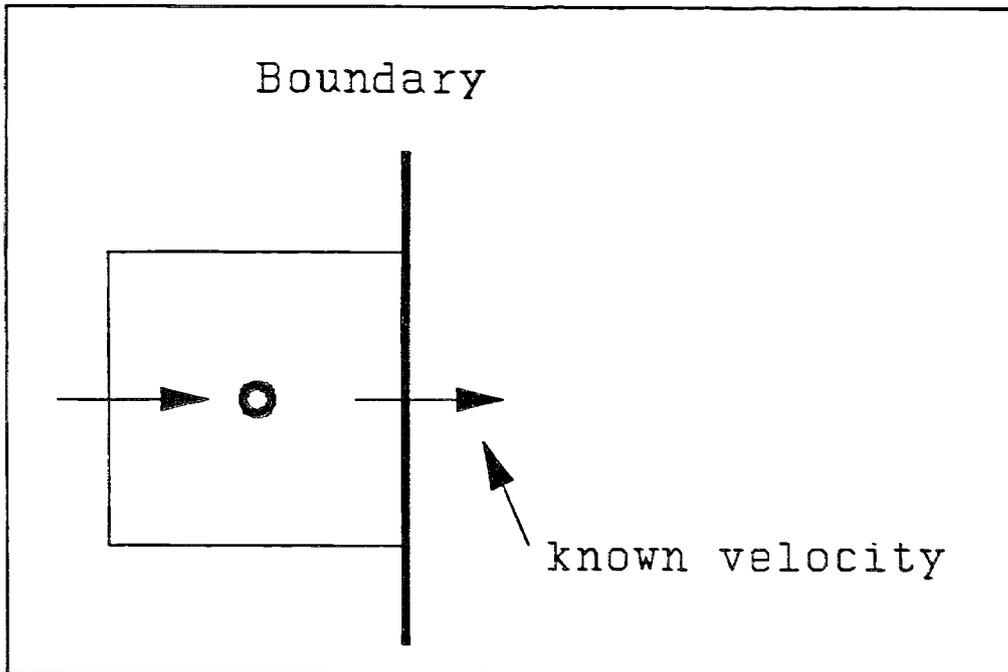


Figure 5.5 Boundary velocity.

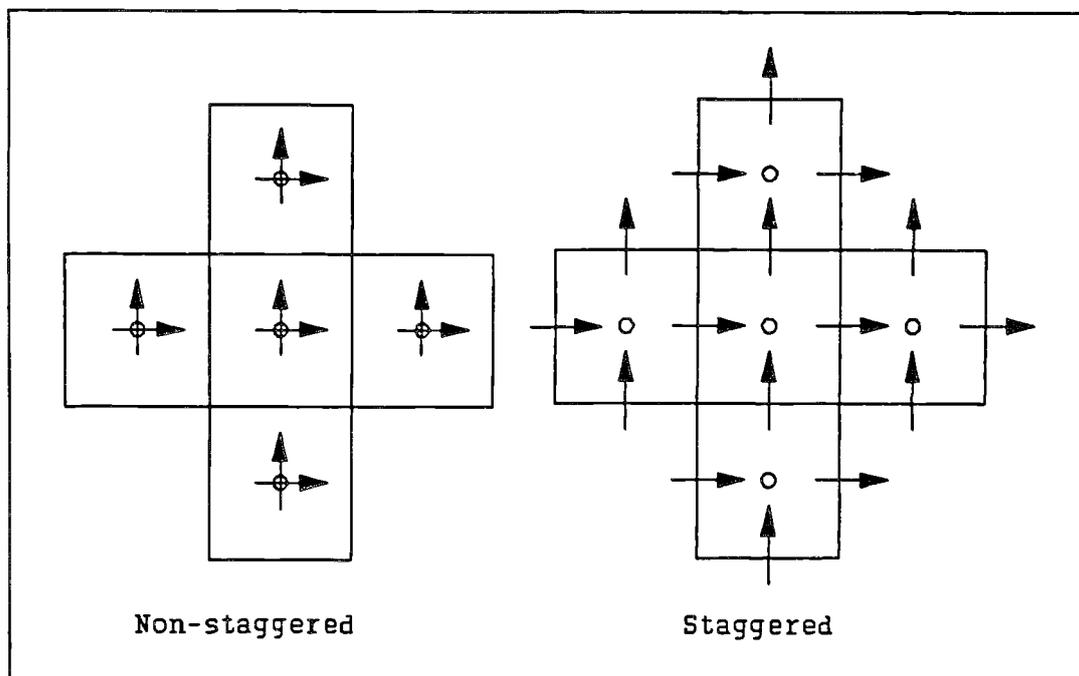


Figure 5.6 Staggered and non-staggered arrangement.

Chapter 6

6.0 Convective-diffusive solidification - validation and results	140
6.1 Introduction	140
6.2 The solution algorithm	141
6.3 Fluid flow and heat transfer benchmark cases	143
6.3.1 Case I - Moving lid cavity	144
6.3.1.1 The problem	144
6.3.1.2 Results with quadrilateral cells	144
6.3.1.3 Results with non-quadrilateral cells	148
6.3.2 Case II - Natural convection driven cavity	152
6.3.2.1 The problem	152
6.3.2.2 Results with quadrilateral cells	154
6.3.2.3 Results with non-quadrilateral cells	158
6.4 Solidification with both convection and diffusion	161
6.4.1 Case I - Melting of pure gallium in rectangular cavity	161
6.4.1.1 The problem	161
6.4.1.2 Results	163

6.4.2 Case II - A demonstration of an integrated approach	165
6.4.2.1 The problem	165
6.4.2.2 Results	167
6.5 Closure	169
Figures	171

6.0 Convective-diffusive solidification - validation and results

6.1 Introduction

In this chapter, the ICV method described in the previous chapter is used to solve two standard benchmark Computational Fluid Dynamics (CFD) cases for heat and mass transfer, followed by two casting simulation cases. The first deals solely with convection-diffusion phase change and the second is an integrated, fluid flow, heat transfer, solidification, residual stress and cast deformation prediction, all coupled and solved together in a single solution procedure.

The two benchmark CFD cases are, the moving lid cavity and the natural convection driven cavity. In both cases, a structured mesh was first used to obtain a numerical solution for comparison with the benchmarks. This validate the ICV method using structured mesh, and secondly, the numerical solution can be used as reference for comparing with unstructured mesh results. Therefore, any differences can be emphasised by not just with the benchmark solutions but against a numerical solution obtained with a structured (quadrilateral) mesh.

For the two casting cases, the first is to validate the convective-diffusive model implemented for freezing and melting processes. The case chosen is the melting of pure gallium in a rectangular cavity, where both experimental and numerical data are available in publications, an extremely rare occurrence in casting, especially of experimental data that deal with both convection and diffusion in solidification. In the

second case, an integration of fluid flow, heat transfer, solidification and residual stress for deformation have all been coupled into a single numerical code called UIFS (a two-dimensional unstructured CFD code that was initiated and subsequently developed during this study). The residual stress formulation is based on the vertex-centred approach, as described in *Chapter 3* of this thesis. The model developed by Fryer, Bailey, Cross and Lai (1991) has been integrated with the fluid flow, heat transfer and solidification modules that are cell-centred based. This integration demonstrated not just that the two unstructured approaches can be used together but also that two major components of the casting process, convective-diffusive solidification and residual stress for deformation, can now be coupled and solved simultaneously. This work was initiated under the SERC/ACME casting initiative: "Advanced Numerical Modelling Programme" and it is still continuing, so the case will be a demonstration rather than a validation exercise. Still the case highlights the need for an unstructured mesh approach in numerical simulations of the casting process.

6.2 The solution algorithm

The SIMPLE solution procedure of Patankar and Spalding (1972) is a predictor-corrector method, where the momentum equations (predictor) are not solved to any high degree of convergence, while the pressure-correction (corrector) is solved to a high degree of convergence and then used to correct both the pressure and velocity components. This predicting-correcting process is repeated until all the variables are said to have

converged. Thereby iterative solvers are appropriate in solving the equations within the SIMPLE solution procedure.

Two such iterative solvers were selected for use. They are, the Jacobi with over/under-relaxation (JOR) and Gauss-Seidel with successive over-relaxation (SOR) methods. For solving a set of linear equations the SOR has a much faster rate of convergence than the JOR [Varga (1962)]. It is this slower rate of convergence that the JOR is employed for solving the momentum equations. The reason for this is twofold: First, the changes made to the velocity fields by the JOR are small and this means relaxation is minimised. Second, the JOR has more stable influence within the SIMPLE solution procedure [Ierotheou (1990)] and requiring less relaxation for the velocity components. For the solution of the pressure-correction, where a high degree of convergence is required, the much faster convergence rate of the SOR solver is used. For other variables such as enthalpy, the default solver is SOR, but the user has the option of changing to other available solvers. Here, the default number of iterations (sweeps) for each variable are, two iterations given to each momentum equation, twenty for the pressure-correction and twenty for all other variables such as enthalpy.

In this implementation of the SIMPLE solution procedure, a converged solution is said to have been obtained when the velocity components and mass residuals plus any other dependent variables such as enthalpy all satisfy the following criterion.

$$\| \phi_{err} \|_{norm} \leq \xi \quad 6.1$$

The ϕ_{err} is the change between two successive iterations of a variable. The change can be taken as the absolute or the relative difference depending on the variable under consideration. The *norm* used in this implementation are, l_1 , l_2 and l_∞ varieties. By default, l_∞ is used for all variables, but the user can change this option if a tighter constraint is necessary. And finally, ξ is the tolerance set for the convergence required, the default is set at 1.0E-03. For the results presented in this chapter, these default settings are used; otherwise, it will be stated with the problem concerned.

6.3 Fluid flow and heat transfer benchmark cases

In this section, two standard benchmark cases for Computational Fluid Dynamics (CFD) will be examined using the ICV method. The first is the moving lid cavity problem for fluid flows. The second is the natural convection driven cavity problem, where both heat and mass transfer are involved.

6.3.1 Case I - Moving lid cavity

6.3.1.1 The problem

This laminar incompressible flow in a square cavity whose top wall moves with a uniform velocity in its own plane, see *Figure 6.1*, is commonly used as a model problem for testing and evaluating numerical techniques. This steady state problem was first examined by Burggraf (1966) and has since attracted many others into solving and announcing their numerical solution to the problem. Some notable authors are those of Bozman and Dalton (1973), de Vahl Davis and Malinson (1976) and Ghia et al (1982). Recently, Ghia and Ghia (1988) published a comprehensive set of results for Reynolds numbers from 100 right up to 10,000.

The boundary conditions for the cavity are shown in *Figure 6.1*. The moving wall has velocity components $u=1\text{ms}^{-1}$ and $v=0\text{ms}^{-1}$ and all the other walls assumed a non-slip boundary conditions with velocity components $u=0\text{ms}^{-1}$ and $v=0\text{ms}^{-1}$.

6.3.1.2 Results with quadrilateral cells

A structured uniform mesh of 33x33 cells (*Figure 6.2*) was used to examine the ICV technique using the Rhie and Chow (1982) approximation for velocity at the cell face, due to the employment of a non-staggered grid configuration. A set of results obtained using the structured uniform mesh showing the u and v velocity components, pressure

and a velocity vector plot for Reynolds numbers of 100, 400 and 1000 are shown in *Figures 6.3 to 6.5* respectively. A smooth pressure field is obtained for all three Reynolds numbers with the Rhie and Chow approximation.

A plot of the u and v velocity components in the vertical and horizontal directions respectively in the middle of the cavity are shown against the published results of Ghia and Ghia (1988) using the standard first-order schemes. The Up-wind, Hybrid, Power law and Exponential scheme results are shown in *Figures 6.6 to 6.8* with the default convergence setting stated in the previous section. Only the velocity components are relaxed to a value within or on the 1.0 - 0.75 range, the actual relaxation value used for each scheme are given in *Tables 6.1 to 6.4*.

Re	100	400	1000
Relaxation	1.0	1.0	1.0
200 iterations			
CPU (sec.)	360	360	360
Default convergent criteria			
Iterations	76	88	80
CPU (sec.)	138	158	143

Table 6.1 Up-wind scheme statistical information

Re	100	400	1000
Relaxation	1.0	1.0	0.8
200 iterations			
CPU (sec.)	362	362	362
Default convergent criteria			
Iterations	81	88	85
CPU (sec.)	147	160	154

Table 6.2 Hybrid scheme statistical information

Re	100	400	1000
Relaxation	1.0	1.0	0.75
200 iterations			
CPU (sec.)	368	368	368
Default convergent criteria			
Iterations	80	89	91
CPU (sec.)	147	164	168

Table 6.3 Power-law scheme statistical information

Re	100	400	1000
Relaxation	1.0	1.0	0.75
200 iterations			
CPU (sec.)	368	368	368
Default convergent criteria			
Iterations	80	89	90
CPU (sec.)	147	164	166

Table 6.4 Exponential scheme statistical information

To improve accuracy, each of the schemes were given 200 iterations each and the results of these are shown in *Figures 6.9 to 6.11*. A good agreement is obtained for all with up-winding scheme slightly worsen at Reynolds number 100. For high Reynolds numbers (Re-400 and Re-1000), there is a slight improvement but differences still exist. This difference is due to the well known undesirable numerical diffusion in low order schemes, with hybrid performing slightly better than the rest. All these findings are consistent with Chen (1991) who solved the problem using control volume techniques with low order schemes in both staggered and non-staggered grid arrangements.

A table of computational time, relaxation values and number of iterations for each of the four schemes with the set tolerance and with the 200 iterations is presented in *Tables 6.1 to 6.4*.

To highlight the moving lid problem that it is solved in an unstructured fashion, the cavity is rotated with the structured uniform mesh to an angle between 0-90 degrees for all three Reynolds numbers with the Hybrid scheme. The purpose of this exercise is to mis-align the velocity components, so that the resultant velocity and any inaccuracy or errors in the implementation can be identified when compared with results obtained in the normal fashion, by rotating the solution back to the original angle. Where the two solutions should be identical. *Figure 6.12* shows a typical set of results with the cavity at an angle of 60 degrees.

6.3.1.3 Results with non-quadrilateral cells

In the previous section, the moving lid cavity was solved with a rectangular mesh using an unstructured approach in formulation. The major advantage of the unstructured ICV method is its ability to handle non-quadrilateral control volumes. Here, the non-quadrilateral control volumes (cells) used to solve the same moving lid problem include triangles, which can be easily generated using Finite Element mesh generation packages. And for high-order cells, polygons of N sides with N greater than four, there is currently no ready-made mesh generation program available in the market that generate these type of cells. To demonstrate the use of high-order cells, several very specific mesh generation programs were developed and the meshes generated are shown in *Figure 6.13* for an hexagonal based mesh and *Figure 6.14* for an octagonal based mesh.

Taking the structured uniform mesh consisting of 1089 cells (33×33) with the hybrid scheme solution as a benchmark, the non-quadrilateral cells are used to solve the moving lid problem. For an unbiased comparison, similar number of cells is used in all simulations. For the triangular mesh, see *Figure 6.15*, consisting a total of 1152 triangles, these triangles were generated using a structured uniform mesh framework for ease of generation and consistency between coarse and fine meshes. In the hexagonal mesh, see *Figure 6.13*, containing a total of 1042 cells, it consisted of 323 hexagons, 36 quadrilaterals and 684 triangles. And for the octagonal mesh, see *Figure 6.14*, containing a total of 1097 cells, the mesh is made up of 481 octagons, 544

quadrilaterals, 60 hexagons and 12 triangles. In the latter two meshes, mixing of several different kinds of cells is possible and it incurs no extra step in the formulations.

Runs were made for Reynolds number of 100, 400 and 1000 using the three non-quadrilateral meshes. All the numerical solutions were obtained using the Hybrid differencing scheme with 200 iterations given for the SIMPLE solution procedure.

The results for the three meshes will be presented and assessed by the order of their Reynolds numbers (100, 400 and 1000). Plots of the velocity vectors and contours of pressure, u and v velocity components are presented for each mesh in turn, triangle, hexagon and octagon. As with mesh generation of polygon (high-order) cells, no ready made software on the market that can do the job, this is also true for contouring these polygonal meshes. Therefore, a polygon contouring program was developed to plot the results. Simple linear interpolation is used for the contouring and is the reason why some plot contained jagged lines. In assessing the results, plots of the u and v velocity components along the vertical and horizontal middle of the cavity are plotted against the results of Ghia and Ghia (1988).

For Reynolds number 100, *Figures 6.16 to 6.18* show the triangle, hexagon and octagon results respectively. For the u and v plots against Ghia and Ghia, see *Figures 6.19 to 6.21*. From the plots, all three numerical solutions have a good agreement with Ghia and Ghia. The triangle is slightly under predicting the maximum and minimum peaks in the v -velocity component, the octagon on the other hand is slightly over while the

hexagon is between the two.

For Reynolds number 400, *Figures 6.22 to 6.24* show the triangle, hexagon and octagon results respectively. For the *u* and *v* plots against Ghia and Ghia, see *Figures 6.25 to 6.27*. In the plots both the triangle and hexagon are under predicting the maximum and minimum peaks in both the *u* and *v* velocity components. With the hexagon slightly better than the triangle. The octagon on the other hand captured the peaks but it is slightly over predicting both the velocity components. Still the octagon gives by far the best result obtained using either quadrilateral or non-quadrilateral meshes.

For Reynolds number 1000, *Figures 6.28 to 6.30* show the triangle, hexagon and octagon results respectively. For the *u* and *v* plots against Ghia and Ghia, see *Figures 6.31 to 6.33*. In the plots both the triangle and hexagon are under predicting the maximum and minimum peaks in both the *u* and *v* velocity components. With the hexagon doing slightly better than the triangle. The octagon mesh on the other hand is slightly under predicting the peaks, but it is by far the best solution out of the set. Note, Ghia and Ghia used a very fine uniform mesh, 23,104 cells (152x152), in obtaining their solution. Here, only 1,097 cells were used.

With these encouraging results from the octagon mesh, other first order schemes, namely the Up-wind and the Power law were used to see their influence on the numerical solution of the octagon mesh. From the rectangular uniform mesh results, here and

elsewhere, the Hybrid scheme is generally performing slightly better than all the other first order schemes.

Figures 6.34 to 6.36 are the plots of the u and v velocity components against Ghia and Ghia for the Up-wind scheme and similarly *Figures 6.37 to 6.39* are the Power law results against those of Ghia and Ghia. The results obtained from the two first order schemes show a significant improvement over their results obtained using the rectangular mesh. In all the cases, it has a better match with the Ghia and Ghia than the rectangular results. It also can be said that all the results obtained with the various schemes on the octagon mesh perform no poorer than the best case obtained for the rectangular mesh, and each scheme displayed the same characteristics as in the rectangular result. This leads one to suggest the ICV is a super-set of the CV method.

The numerical results obtained for both quadrilateral and non-quadrilateral meshes can be used to explain why the meshes have such an influence on the solution. The answer lies in the average number of connections (with neighbouring nodes) in a control volume. As it is with high-order schemes used in quadrilateral meshes, as more nodes are accounted for in the surrounding region of a control volume, the numerical diffusion for flows with changing direction being the main course of the inaccuracy is reduced.

In the triangle and quadrilateral meshes, the average number of connections per control volume is 3 and 4 respectively. For hexagon and octagon meshes this is not so. In the hexagon mesh, the average connection per control volume is 3.97. This is mainly due

to the large number of triangles in the mesh, and hence the low connection value. This can in some way explained why the results are much more similar to the quadrilateral than with the others. For octagon mesh, the average connection per control volume is 5.85. It is almost two connections per control volume more than that of the quadrilateral and hexagon, and hence explaining why the superior results.

An under-relaxation value of 0.8 was applied to the momentum equations for the octagon mesh to avoid divergence in all three Reynolds numbers. No relaxation was necessary in the other two meshes.

6.3.2 Case II - Natural convection driven cavity

6.3.2.1 The problem

In this problem, the natural convection in a cavity closure where the flow is driven by buoyancy forces. The buoyancy in this instant is induced by a temperature difference between the two vertical walls. Where one wall is hot and the other cold, and perfectly insulated at the top and bottom surfaces. This problem was first proposed by Jones (1970) as a suitable test case for validating CFD codes, also being of practical interest for mathematical modellers and engineers. In the solidification processes, the convection in the molten liquid is mostly driven by buoyancy where it is induced by a temperature difference.

In 1983, de Vahl Davis and Jones published what is arguably the most extensive comparison and discussion of the single problem. Thirty seven numerical contributions from thirty groups were considered, against the de Vahl Davis and Jones (1983) benchmark solutions. The problem was solved as an incompressible fluid, this assumes that density is constant except in the buoyancy term in the momentum equations, where the Boussinesq approximation is used (*Section 5.4.2* for the approximation).

The co-ordinates, x , y and the u and v velocity components are non-dimensionalised using the length of the cavity L and the thermal diffusivity α , thereby

$$\begin{aligned}x &= \frac{x}{L} & u &= \frac{uL}{\alpha} \\y &= \frac{y}{L} & v &= \frac{vL}{\alpha}\end{aligned}\tag{6.2}$$

The dimension L of the cavity is used to modify the Rayleigh number (Ra), this is defined by

$$Ra = \frac{\beta g \Delta T L^3}{\alpha \nu}\tag{6.3}$$

where β is the thermal expansion coefficient, g is gravity and ν is the kinematic viscosity.

The temperature difference between the two vertical walls, ΔT is one. Therefore, the prescribed temperature boundary conditions are, for $x=0$, $T=1K$ and $x=1$, $T=0K$, with all the walls assumed a no-slip boundary conditions with velocity components $u=0ms^{-1}$

and $v=0\text{ms}^{-1}$, see *Figure 6.40*. Numerical solutions were obtained for Rayleigh numbers of 10^3 , 10^4 , 10^5 and 10^6 . With the cavity medium taken to be air, which has a Prandtl number of 0.71.

6.3.2.2 Results with quadrilateral cells

Following the practices of other authors such as those of Markatos and Pericleous (1984), a non-uniform structured mesh of 33×33 (1089) cells was used, see *Figure 6.41*, with 200 iterations given for the SIMPLE solution procedure. The initial velocity components, pressure and temperature are $u=0\text{ms}^{-1}$, $v=0\text{ms}^{-1}$, $p=0\text{Nm}^{-2}$ and $T=0\text{K}$, for Rayleigh number of 10^3 . For the other three Rayleigh numbers, the initial values are taken from the numerical solution of the previous Rayleigh number, ie. the solution of $\text{Ra}-10^3$ is the initial value for $\text{Ra}-10^4$ and $\text{Ra}-10^4$ solution is the initial value for $\text{Ra}-10^5$ and so on. A uniform structured mesh of the same size was also used. The purpose for this is twofold. First, and it is the major reason, is for comparison with non-quadrilateral high-order cells. Which will be explained in the following section, why an adaptive mesh is currently impossible to generate but necessary in obtaining a good agreement with the benchmark solution without resorting to fine meshes. And by comparing with a uniform mesh, it gives us some idea how the high-order cells will perform in a similar non-adaptive mode. Second, it reconfirmed the need for non-uniform mesh in this problem for high Rayleigh numbers.

Figures 6.42 to 6.45 show the non-uniform and *figures 6.46 to 6.49* the uniform numerical solution obtained for Rayleigh number of 10^3 to 10^6 respectively. Within each figure, a plot of the velocity vectors and contour plots of the temperature and the u and v velocity components is depicted. *Tables 6.5 and 6.6* show the solutions obtained in this study for the non-uniform and uniform mesh respectively against the benchmark solution of de Vahl Davis (1983). The table showing the maximum u-velocity and its location on the vertical mid-plane, the minimum and its location on the horizontal mid-plane, and on the vertical boundary of the cavity at $x=0$, the maximum local Nusselt number and its location, the minimum local Nusselt number and its location and the average Nusselt number on the vertical boundary.

For the non-uniform mesh, there is a good agreement with the benchmark solution for all the Rayleigh numbers. The difference is slightly high at $Ra=10^6$ than with the rest. This is because of the thin boundary layers developed at high Rayleigh number in the two conducting walls. In the benchmark solution a finer mesh (81x81) was used that was accurate in depicting the thin boundary layers. For the uniform mesh, the results conform with the expectation, good at low Rayleigh numbers but differences appearing in the high numbers.

	10^3		10^4		10^5		10^6	
u_{max}	3.638	(3.649)	16.151	(16.178)	34.861	(34.73)	65.173	(64.63)
y	0.825	(0.813)	0.825	(0.823)	0.855	(0.855)	0.855	(0.85)
v_{max}	3.696	(3.697)	19.7	(19.617)	68.416	(68.59)	220.82	(219.36)
x	0.175	(0.178)	0.115	(0.119)	0.07	(0.066)	0.03	(0.0379)
Nu_0	1.114	(1.117)	2.213	(2.238)	4.54	(4.509)	9.345	(8.817)
Nu_{max}	1.507	(1.505)	3.558	(3.528)	7.998	(7.717)	19.778	(17.925)
x	0.09	(0.092)	0.145	(0.143)	0.07	(0.081)	0.03	(0.0378)
Nu_{min}	0.692	(0.692)	0.586	(0.586)	0.726	(0.729)	1.028	(0.989)
y	0.99	(1)	0.99	(1)	0.99	(1)	0.99	(1)

Values in brackets () are from de Vahl Davis

Table 6.5 A comparison between the non-uniform mesh and the de Vahl Davis solution

	10^3		10^4		10^5		10^6	
u_{max}	3.63	(3.649)	16.127	(16.178)	34.822	(34.73)	65.4	(64.63)
y	0.803	(0.813)	0.833	(0.823)	0.864	(0.855)	0.864	(0.85)
v_{max}	3.677	(3.697)	19.533	(19.617)	67.295	(68.59)	208.8	(219.36)
x	0.167	(0.178)	0.106	(0.119)	0.076	(0.066)	0.045	(0.0379)
Nu_0	1.118	(1.117)	2.213	(2.238)	4.662	(4.509)	9.629	(8.817)
Nu_{max}	1.51	(1.505)	3.6	(3.528)	8.34	(7.717)	21.133	(17.925)
x	0.076	(0.092)	0.136	(0.143)	0.076	(0.081)	0.152	(0.0378)
Nu_{min}	0.69	(0.692)	0.582	(0.586)	0.715	(0.729)	0.979	(0.989)
y	0.985	(1)	0.985	(1)	0.985	(1)	0.985	(1)

Values in brackets () are from de Vahl Davis.

Table 6.6 A comparison between the uniform mesh and the de Vahl Davis solution

For a speedy convergence for each Rayleigh number, some under-relaxation was applied to the velocity components. Relaxation values plus the CPU times and iterations taken for the default convergence criteria are in Tables 6.7 and 6.8 for the non-uniform and uniform mesh respectively for each Rayleigh number.

	10^3	10^4	10^5	10^6
Relax.	1.0	1.0	1.0	0.3
200 iterations				
CPU (sec.)	540	540	540	540
Default convergent criteria				
Iterations	86	53	32	55
CPU (sec.)	234	144	87	150

Table 6.7 Non-uniform mesh statistical information.

	10^3	10^4	10^5	10^6
Relax.	1.0	1.0	1.0	0.3
200 iterations				
CPU (sec.)	540	540	540	540
Default convergent criteria				
Iterations	93	58	41	58
CPU (sec.)	251	157	111	157

Table 6.8 Uniform mesh statistical information.

6.3.2.3 Results with non-quadrilateral cells

As the Rayleigh number gets higher, a thin boundary layers developed in the two conducting walls. Unlike the non-uniform structured mesh, *Figure 6.41*, where the mesh has been adapted to pick-up the thin boundary layers near the surface of the walls with the mesh remaining orthogonal. This mesh adaption cannot be easily applied to non-quadrilateral meshes, with triangle being the exception. Such mesh adaption is not a straight forward process in a square cavity and with triangles having one degree of freedom (connection) less than quadrilaterals. The solution would not be any better than the quadrilateral results obtained using the non-uniform mesh. The interest here is to investigate what benefits, if any, are to be had with the high-order cells. Even without any adaptation of the mesh, the results obtained would still be of benefit for a better insight of the high order cell technique. From the earlier work done with the moving lid cavity, the octagon mesh proved to have the best results and consequently the same mesh is used for this investigation.

Figures 6.50 to 6.53 show the numerical solution obtained for the four Rayleigh numbers. Within each the figures hold a plot of the velocity vectors and contour plots of the temperature and the u and v velocity components. *Table 6.9* show the quantities for making comparison between the octagon results and the benchmark solution.

	10^3		10^4		10^5		10^6	
u_{\max}	3.697	(3.649)	16.394	(16.178)	35.023	(34.73)	67.011	(64.63)
y	0.813	(0.813)	0.813	(0.823)	0.844	(0.855)	0.844	(0.85)
v_{\max}	3.705	(3.697)	19.839	(19.617)	69.923	(68.59)	216.54	(219.36)
x	0.186	(0.178)	0.125	(0.119)	0.0625	(0.066)	0.0313	(0.0379)
Nu_0	1.144	(1.117)	2.289	(2.238)	4.431	(4.509)	6.789	(8.817)
Nu_{\max}	1.548	(1.505)	3.613	(3.528)	7.363	(7.717)	11.919	(17.925)
x	0.094	(0.092)	0.156	(0.143)	0.0938	(0.081)	0.0625	(0.0378)
Nu_{\min}	0.703	(0.692)	0.589	(0.586)	0.717	(0.729)	0.958	(0.989)
y	0.99	(1)	0.99	(1)	0.99	(1)	0.99	(1)

Values in brackets () are from de Vahl Davis.

Table 6.9 A comparison between the octagon and the de Vahl Davis solution

The octagon result is not as good as the structured non-uniform mesh, but it is similar to those of the uniform one, fine at low Rayleigh numbers but differing at high values. The differences are highlighted extremely well by the u and v velocity contour plots at $Ra \cdot 10^6$. For the uniform mesh, the usual features for a contour plot of the u and v velocity are all present. In the octagon mesh, the u-velocity contour plot is picking up the two circles, which are in the benchmark u-velocity plot. The same plot on the non-uniform mesh is only just visible and it is not as well defined as the octagon. The same cannot be said for the v-velocity contour plot. There is a significant difference between the benchmark contour and that of the octagon. The reason for this is; the thin boundary layers at the two conducting walls, where both the heat and mass transfer is convection dominated. In this instant the rectangle cell is better suited than the high-order ones, since the flow is aligned with the cell, which makes the neighbouring

diagonal contribution insignificant. This has also been found by authors using high-order schemes [Patel (1987)], where convection dominated flow with high-order scheme result back to a first-order one. For recirculating heat and mass transfer, the high-order cells are able to pick up the information where quadrilateral cells cannot, as shown in the u-velocity contour plot. Thereby, similar characteristics exist between the high-order cells and high-order schemes.

Table 6.10 shows the relaxations used on the velocity components, the CPU usage and number of iterations taken for convergence with the default convergent criteria for the octagon mesh.

	10^3	10^4	10^5	10^6
Relax.	0.8	0.8	0.7	0.2
200 iterations				
CPU (sec.)	814	814	814	814
Default convergent criteria				
Iterations	90	58	37	52
CPU (sec.)	367	239	152	214

Table 6.10 Octagon mesh execution statistical information.

6.4 Solidification with both convection and diffusion

In this section, two relevant casting cases are to be examined. The first is the melting of pure gallium in a rectangular cavity, where convection-diffusion phase change occurs and experimental data are available. The second is an integrated simulation of a sand cast, where the fluid flow, heat transfer and solidification are solved simultaneously with the residual stress calculations for the cast deformation predictions.

In this examination, no high-order cells were used. This is due the difficulty in generating high-order cells that is or near orthogonal. The programs that generated the meshes for the previous CFD cases is based on a square geometry, and with the cases considered here not geometrically square, this make the mesh incompatible. With triangular meshes obtaining no better results than the quadrilateral ones, as shown in the previous fluid flows cases, quadrilateral meshes are used for cases considered here.

6.4.1 Case I - Melting of pure gallium in rectangular cavity

6.4.1.1 The problem

This concerns laminar natural-convection fluid flow and heat transfer for the melting of pure gallium in a rectangular cavity, where the top and bottom surfaces are perfectly insulated and the two vertical walls are, one hot, and one cold. The hot wall is above the melting temperature of gallium and the cold wall is below the melting temperature.

A comprehensive experimental study on the melting of gallium in a rectangular cavity had been conducted by Gau and Viskanta (1988). They presented detail traces of the morphology of the melt front at various times, which makes comparison with numerical solution possible. This transient, highly non-linear and coupled problem has been solved by several authors, such as Brent, Voller and Reid (1988), where they have used the same basic enthalpy model as described in the previous chapter. This allows validation of the convection-diffusion solidification model both numerically and experimentally. Also, all the information regarding material properties, mesh size and the necessary data for computation are in the Brent et al publication.

The boundary conditions for the rectangular cavity are shown in *Figure 6.54*. The hot wall temperature being 38°C, the initial and the cold wall is at 28.3°C. All walls assumed a non-slip boundary conditions with velocity components $u=v=0\text{ms}^{-1}$. The x and y dimensions for the rectangular domain 8.89cm and 6.35cm respectively. The physical properties for pure gallium are well documented [Cubberley (1979), Duggin (1969)], and the values used in this simulation can be found in *Table 6.11*.

Density (liquid), ρ	6093.0 kg m ⁻³
Reference density, ρ_{ref}	6095.0 kg m ⁻³
Reference temperature, T_{ref}	29.78 °C
Thermal expansion coefficient of liquid, β	1.2 x 10 ⁻⁴
Thermal conductivity, k	32.0 W m ⁻¹ K ⁻¹
Specific heat capacity, c_p	381.5 J kg ⁻¹
Dynamic viscosity,	1.81 x 10 ⁻³ kg m ⁻¹ s ⁻¹
Melting point, T_m	29.78 °C
Latent heat of fusion, L	80160.0 J kg ⁻¹

Table 6.11 Physical properties of pure gallium

6.4.1.2 Results

Using the same mesh size employed by Brent, Voller and Reid (1988), a mesh size of 42x32 (1344) cells was used for this study. A constant time step of 5 seconds was found to be more stable and trouble free over the whole simulation than by increasing the time step in stages. For switching on/off the velocity components in cells undergoing a phase change, a value of 1.5E+05 is used for the morphology constant C .

In this highly non-linear coupled problem, under-relaxation was found to be necessary to avoid divergence in the current work, this was also reported by Brent et al. The under-relaxation values are; a false time step of 0.1 for the two momentum equations, a value of 0.8 for the pressure correction, a linear under-relaxation of 0.8 for enthalpy and a value of 0.1 for the liquid fraction update. The position of the phase front was determined by plotting the liquid fraction, f , at its 0.5 value for an instance in time.

This provided a smoother and more precise estimate of the phase front position than plot of the cells undergoing the phase change. The convergence criteria, apart from the standard mass residual and all the solving variables, included the temperature and the liquid fraction in equation 6.1 before a converged solution is said to have been obtained for a time step. The l_2 norm was found to be more appropriate in this instance than the default norm of l_∞ . It gives a residual that is more representative of the errors than by the maximum value given by l_∞ .

Figures 6.55 to 6.58 show in each, the velocity vectors, and contour of the pressure, temperature, and liquid fraction at 3, 6, 10 and 19 minutes. At 3 minutes, the natural convection field has just begun to develop with the melt front still virtually planar. The heat transfer up to this stage is primarily accounted for by conduction with very little influence by convection. As time progresses this is going to change from a diffusion driven to convection driven problem. At 6 minutes, the natural convection has developed and it is beginning to have a significant influence on the shape of the melt front. As the melted liquid is heated by the hot wall and cooled by the cold solid front. The hot convective current is driven to the top of the rectangular cavity, where it impinges on the upper section of the solid front, thereby pushing the front further into the solid than at the base of the cavity as it loses the heat when the current moves from top to bottom near the solid boundary.

The convective current intensifies with time and the formation of the melt front becomes

more acute as the melting progresses. This is shown in both *Figures 6.57* and *6.58* after 10 and 19 minutes respectively. At these times, the melt front and the heat transfer is governed primarily by the convection with conduction exerting very little influence. The melt fronts calculated in the present study and those of experimentally determined by Gau and Viskanta (1986) at various times are plotted in *Figure 6.59*. The plot shows the predicted solutions are in qualitative agreement with experiment.

6.4.2 Case II - A demonstration of an integrated approach

6.4.2.1 The problem

The casting problem undertaken is a cold sand mould filled with a hot molten material that is allowed to cool down and solidify. Natural convection results as the material undergoes cooling and thereby fluid flows within the liquid regions of the cast. Once the material becomes solidified it will deform and may move away from the mould. The formation of a gap between the cast/mould interface will result if deformation occurs, where it will effect the flow of heat from the cast to the mould, thereby inhibiting the cooling and solidification rates.

Modelling this highly complex process involves solving the partial differential equations of momentum, enthalpy and stress. These equations are highly coupled as temperature is dependent on the change in heat loss at the cast/mould interface and both the momentum and stress equations are coupled with the temperatures. In the past

numerical procedures developed for casting simulation have decoupled the thermal convection and residual stresses. Recently deformation and thermal calculations have been coupled using finite element or control volume formulation. In finite elements, Alexandre et al (1990) and Joshua and Prince (1990) using a coupled thermomechanical formulation available in the commercial FE code, ABAQUS [Hibbitt, Karlson and Sorenson (1989)] have undertaken this task. While in control volume, Bailey, Fryer, Cross and Chow (1992) have undertaken the coupling using an unstructured control volume approach on a code developed during this study for solidification using the vertex-centred approach described in *Chapter 3*.

The fully coupled fluid flow, solidification and stress solution procedure and the stress formulation were not part of the aims of the study and with the full details requiring a full chapter of its own, no details of the solution procedure or stress formulation is given here. For a full details description see the publications by Cross, Bailey, Chow and Pericleous (1992).

Density, ρ	1682.0 kg m ⁻³
Thermal conductivity, k	0.8558 W m ⁻¹ K ⁻¹
Specific heat capacity, c_p	761.0 J kg ⁻¹

Table 6.12 Physical properties of sand

The physical properties of the molten material are taken to be the same as for gallium, see *Table 6.11*, and for the sand the physical properties used in this study are given in *Table 6.12* these values have been extracted from the Zeng and Pehlke (1987) problem

done in *Chapter 4*. The dimension of the cast and mould is 0.1m^2 with a symmetry plane on the left hand side of the mould, see *Figure 6.60*. A dirichlet boundary condition is assumed at the other three external walls with a temperature of 298.15K (room temperature). This is also the initial temperature for the sand mould which is assumed to be rigid. The initial temperature for gallium is 323.15K and its melting/freezing temperature at 318.15K . At the gallium/mould interface a heat transfer coefficient, h_{eff} , of $3000\text{ W/m}^2\text{K}$ is assumed when full contact is present. The coefficient of thermal expansion in the solid has been multiplied by 1000 so as the deformation of the material can be observed. The other material properties required for deformation are; youngs modulus of $21 \times 10^{10}\text{Pa}$ and poissons ratio of 0.29. All the cast walls assume a non-slip boundary conditions with $u=0\text{ms}^{-1}$ and $v=0\text{ms}^{-1}$. The case demonstrated here may be considered by some as an academic one, but it has all the ingredients and physical properties of a real casting process.

6.4.2.2 Results

Figures 6.61 to 6.66 show the solution obtained for the integrated model just over one hour period for the cast to solidify, the results are at 0.5, 1, 10, 30, 50 and 65 minutes respectively. In each figure, a vector plot of the velocity and contour plots of the temperature, liquid fraction and residual stresses are shown. The results were obtained using a mesh of 20×20 (400) cells with a constant time step of 5 seconds.

From the plots of the temperature, most of the heat in the cast is lost via the top open air boundary where the dirichlet boundary condition exists. Heat loss is not as great through the cast/mould interface. This is because the sand has a lower conductivity value than gallium and the gap developed between the cast and mould also slows down the heat transfer between the two.

The velocity vector plots indicate that convection plays a significant part at the early periods of heat transfer, bringing the hot liquid gallium at the base of the cast column to the top where the major heat transfer takes place. Once a significant amount of heat has been lost solidification occurs, this in turn terminates the convection and heat is then lost via conduction. This occurs after about 1 minute of cooling, at which time there is only a small recirculation at the base of the cast column. The major fluid flow movement is now at the centre of the cast, ie. near the axis of symmetry, where a large concentration of heat remains.

As the liquid gallium solidifies residual stress develops and deformation occurs. In the early periods most the stress develops along the shell. As the solidification front moves toward the bottom of the column and into the main cast, the major stress area is at the interior of the right angle of the L-shape cast. This is due to the rigid mould reacting with the shrinkage of the cast thereby developing a high stress region, as shown in *Figures 6.65 and 6.66*.

The solidification front moves from the top of the column into main body of the cast

and it is then joined by the front created by the cast/mould interface, together they converge on the symmetry plane and then full solidification occurs.

6.5 Closure

In this chapter, four CFD and solidification cases were solved using the ICV method as described in *Chapter 5*. Two of these cases are specifically for benchmarking CFD codes and numerical techniques in heat and mass transfer. Both quadrilateral (structured) and non-quadrilateral (unstructured) cells, including high-ordered ones, were used in the calculations. The solution obtained for the quadrilaterals agree with other published results using first-order schemes. Using high-order cells, higher resolution can be obtained using first-order schemes, previously only possible in high-order schemes using quadrilateral meshes. For each high-order scheme, rediscratisation is required, not necessary for the high-order cells.

In the two casting cases described, the first is a full convection-diffusion melting of pure metal. Using the enthalpy based convective-diffusive solidification model, the results obtained are in qualitative agreement with experiment and other numerical solutions, in predicting the solid/liquid interface. For the second casting case, a demonstration of an integrated casting simulation is presented. Two major components of the casting process, solidification and shape deformation have been coupled and solved in the same solution procedure all within the same CFD code. Convective-diffusive solidification

using the ICV (cell-centred) and deformation using CV-FE (vertex-centred) method. This not just demonstrated the two casting processes can be coupled, but also the two unstructured control volume methods can co-exist in a single CFD solidification code.

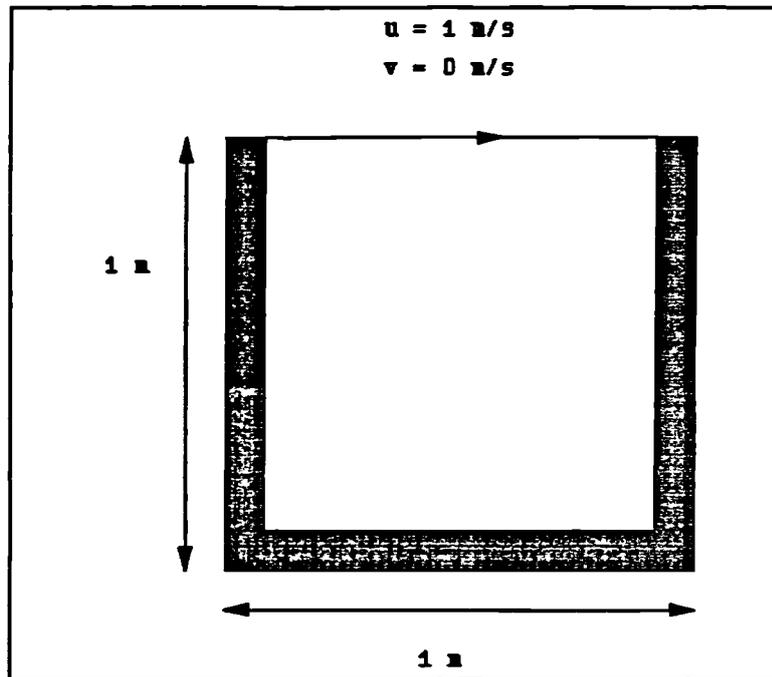


Figure 6.1 Moving lid cavity geometrical specification.

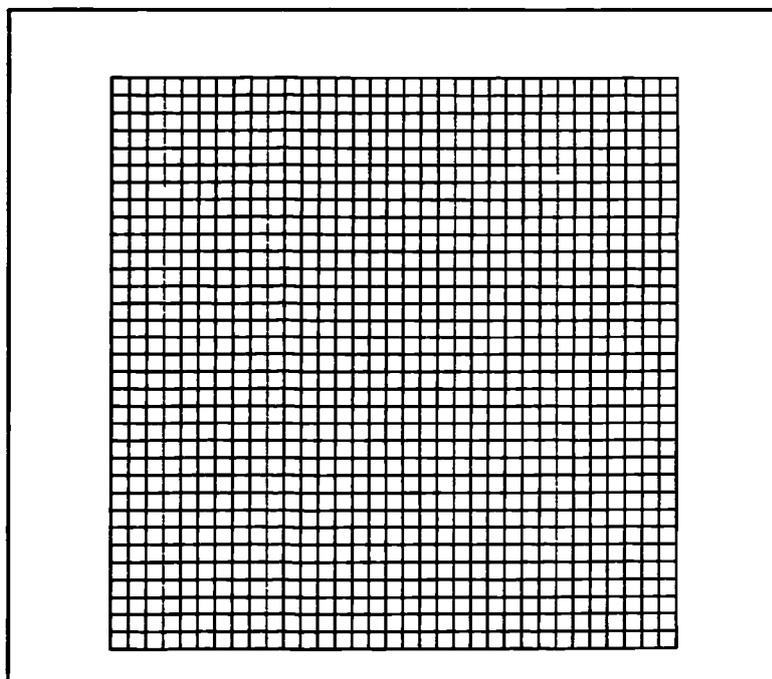


Figure 6.2 A uniform 33x33 mesh.

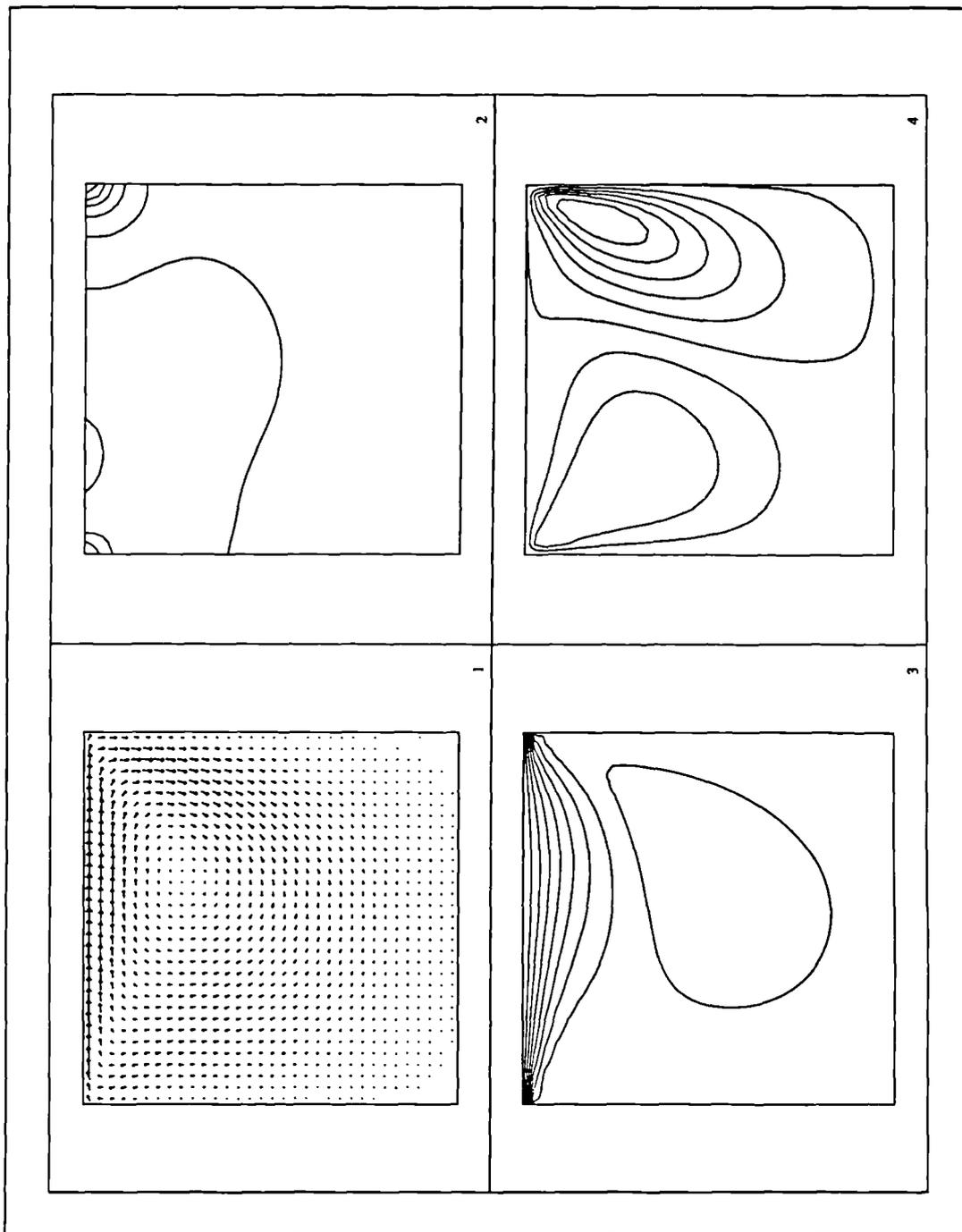


Figure 6.3 Uniform mesh results for $Re=100$.

- 1) Velocity vector.
- 2) Pressure.
- 3) U-velocity.
- 4) V-velocity.

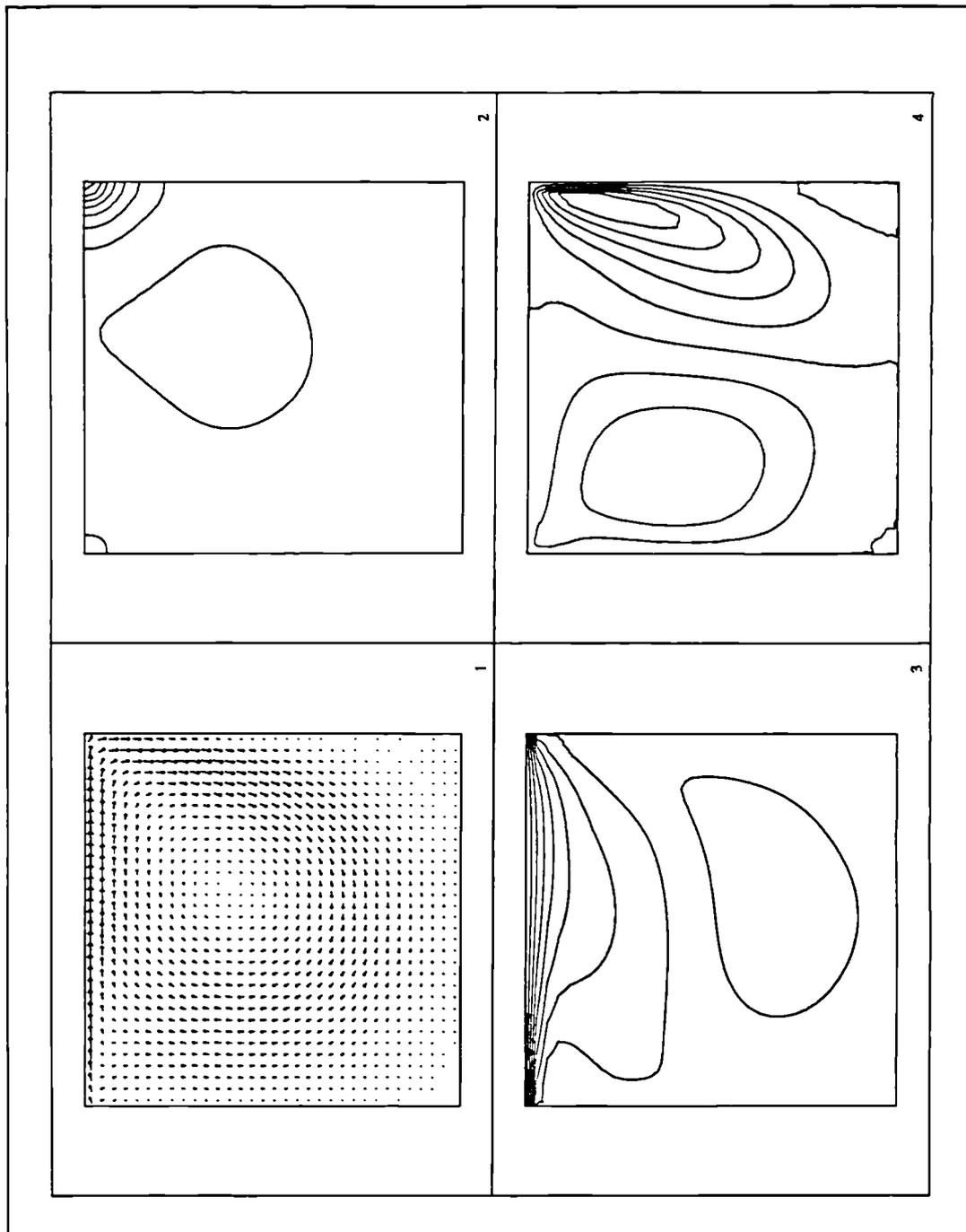


Figure 6.4 Uniform mesh results for $Re=400$.

- 1) Velocity vector.
- 2) Pressure.
- 3) U-velocity.
- 4) V-velocity.

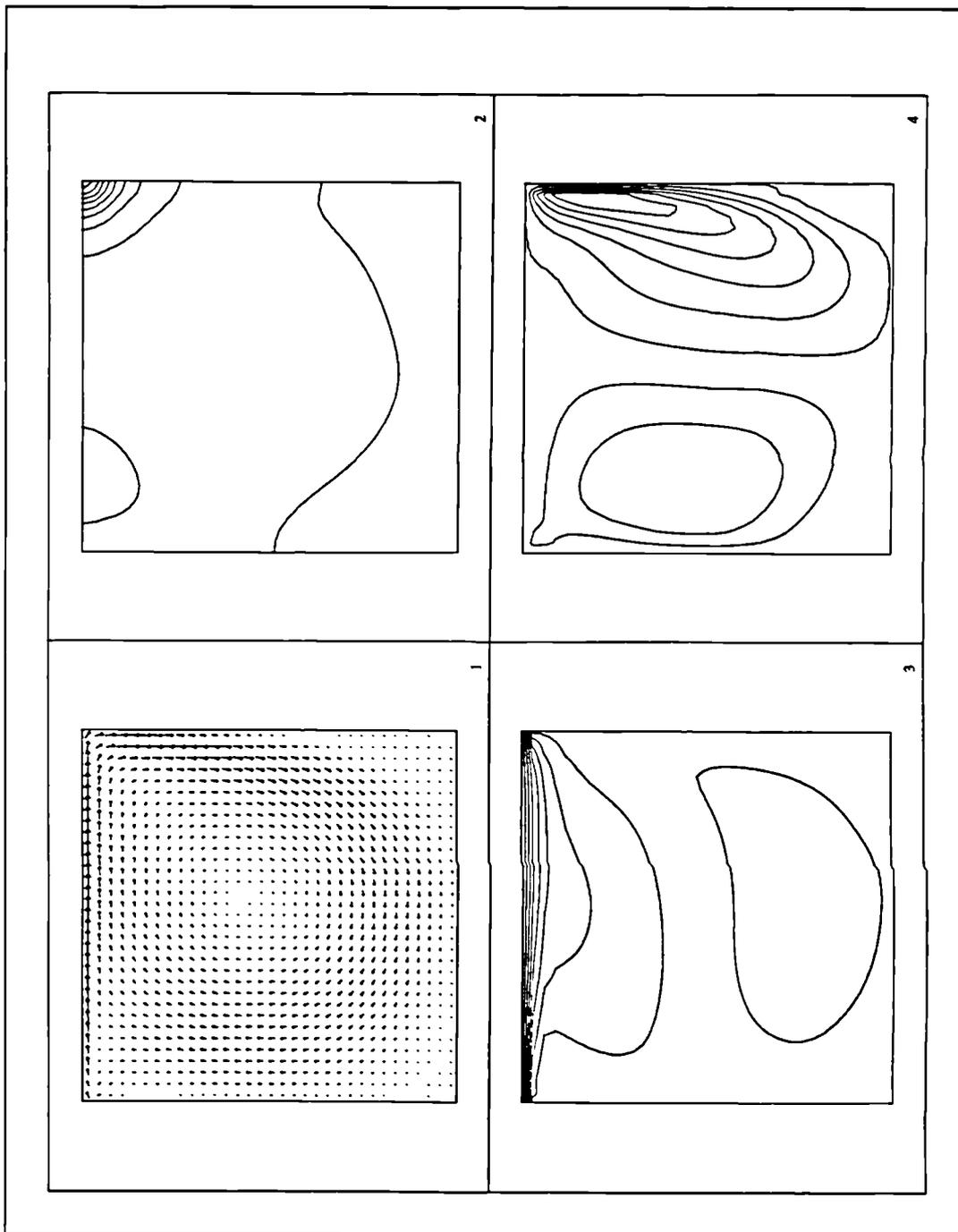


Figure 6.5 Uniform mesh results for $Re=1000$.

- 1) Velocity vector.
- 2) Pressure.
- 3) U-velocity.
- 4) V-velocity.

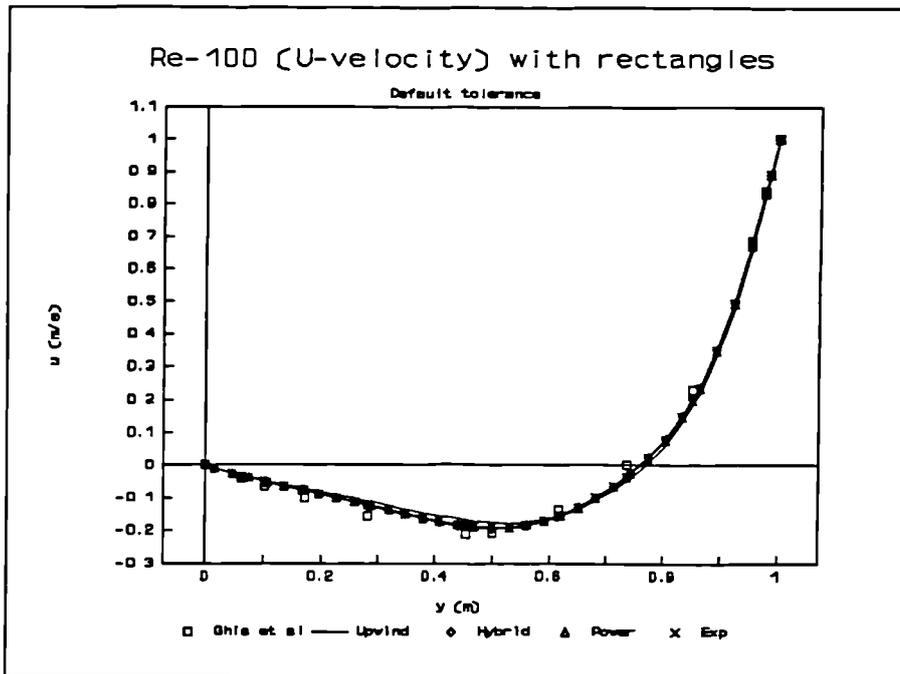


Figure 6.6a U-velocity at vertical middle of cavity for Re-100.

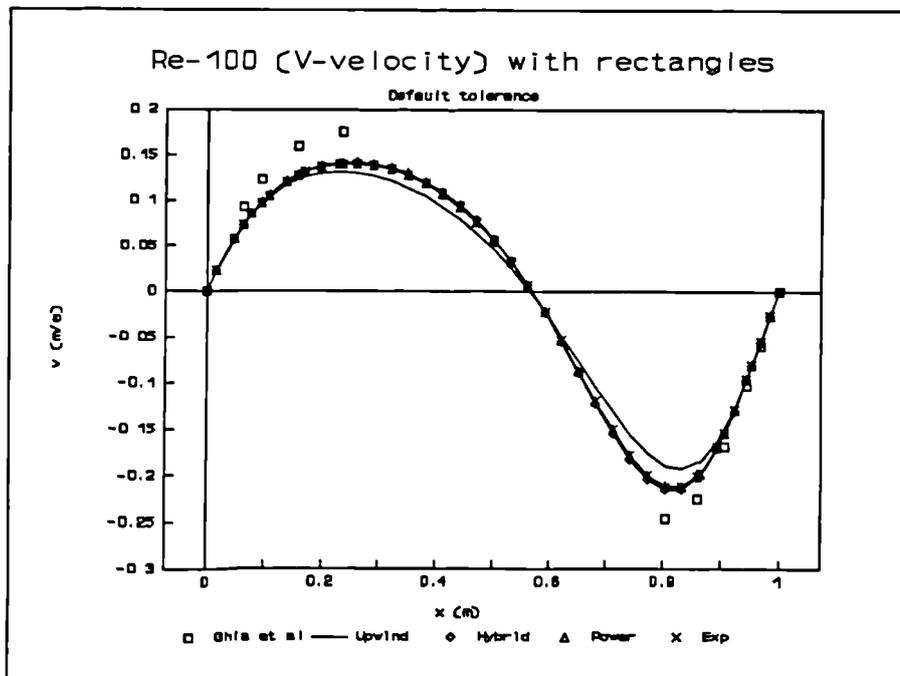


Figure 6.6b V-velocity in horizontal middle cavity for Re-100.

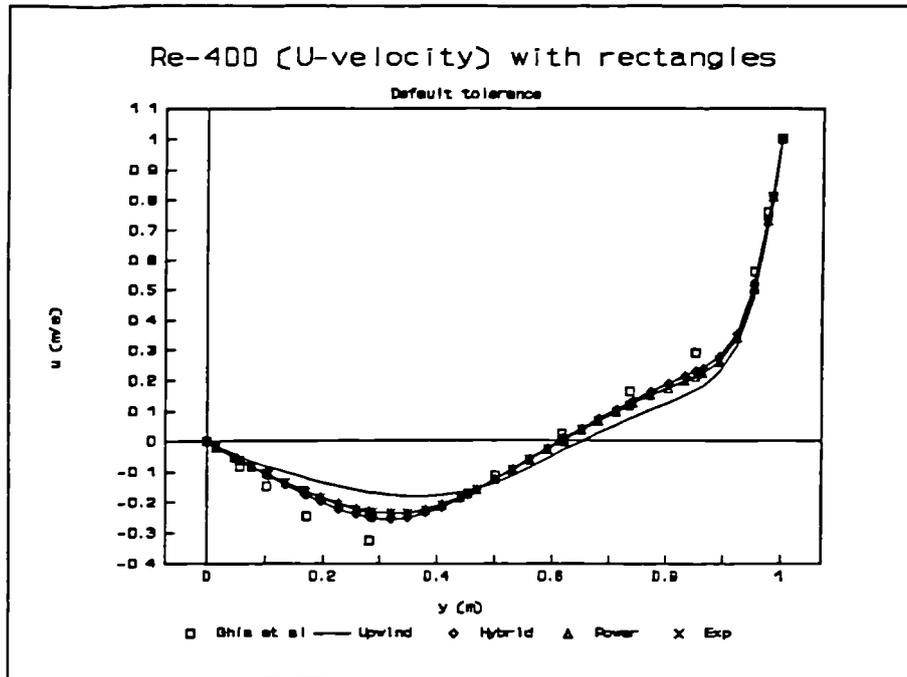


Figure 6.7a U-velocity at vertical middle of cavity for Re-400.

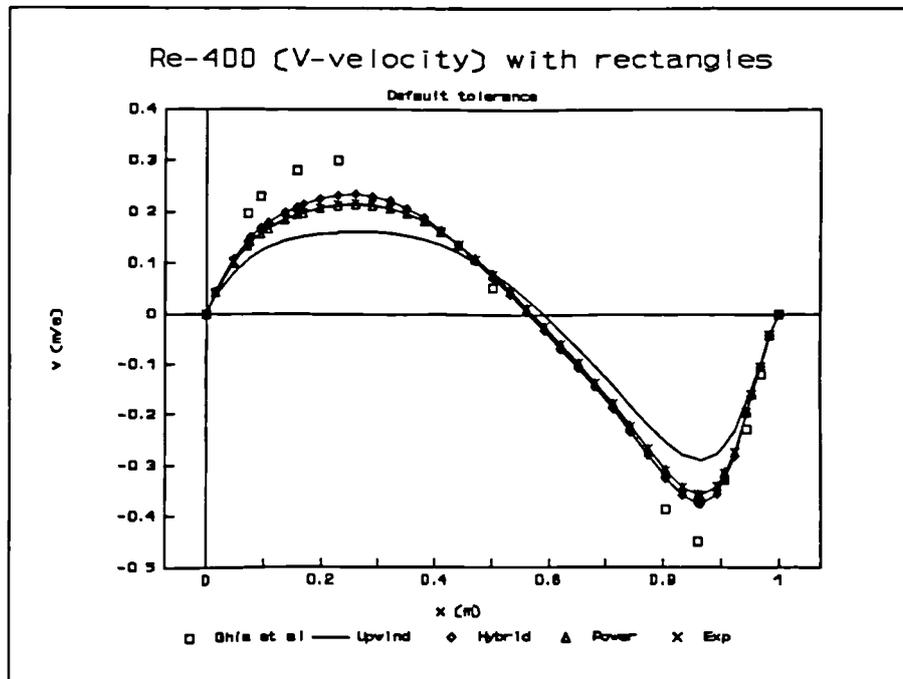


Figure 6.7b V-velocity in horizontal middle cavity for Re-400.

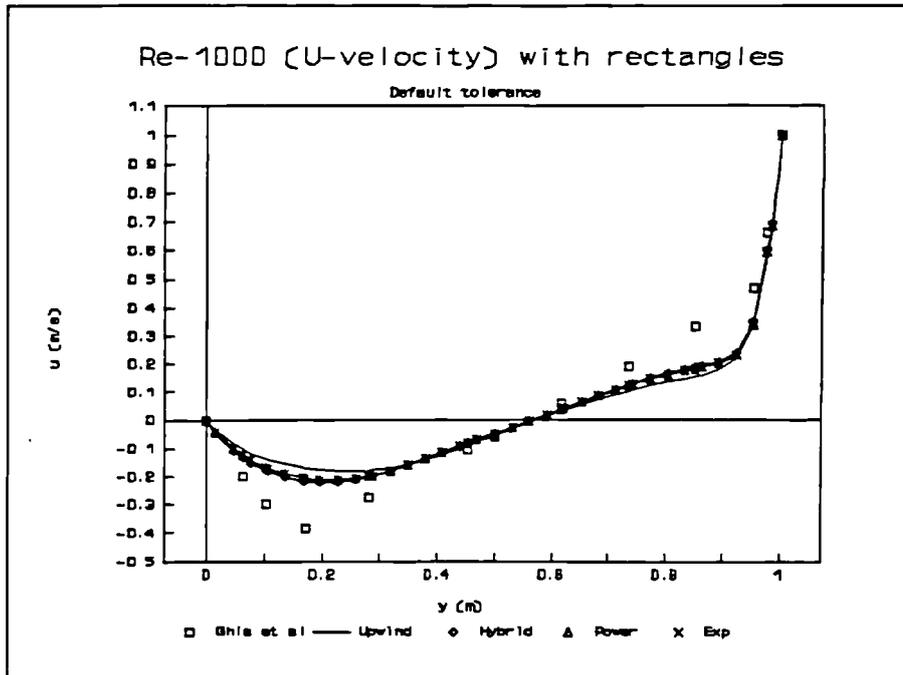


Figure 6.8a U-velocity at vertical middle of cavity for Re-1000.

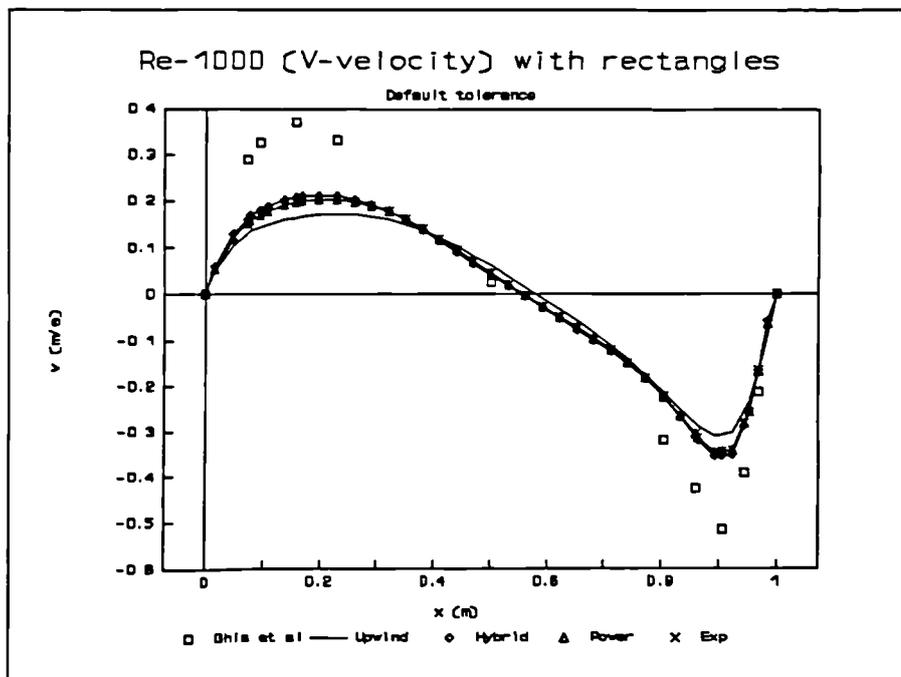


Figure 6.8b V-velocity in horizontal middle cavity for Re-1000.

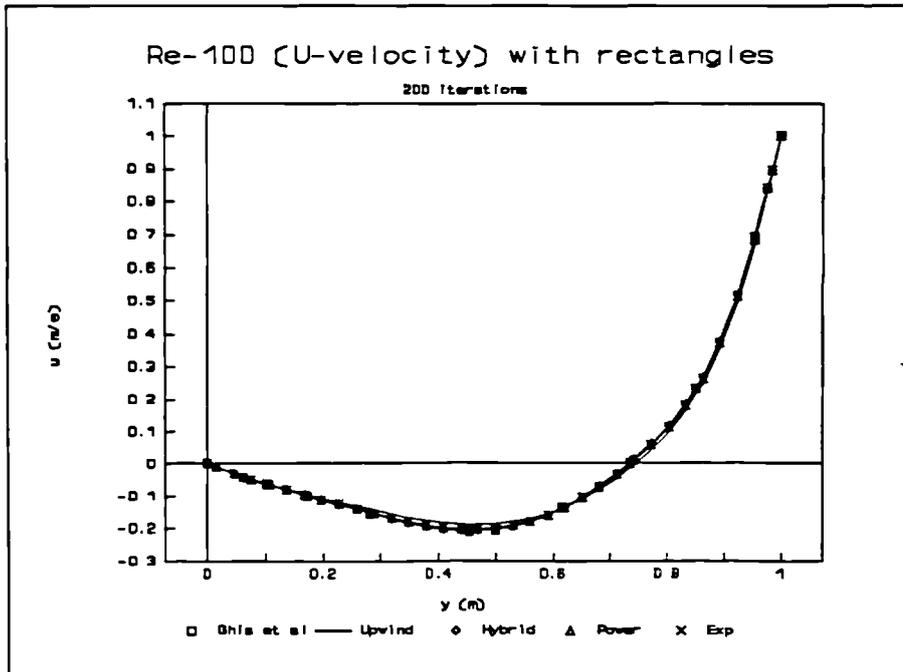


Figure 6.9a U-velocity at vertical middle of cavity for Re-100.

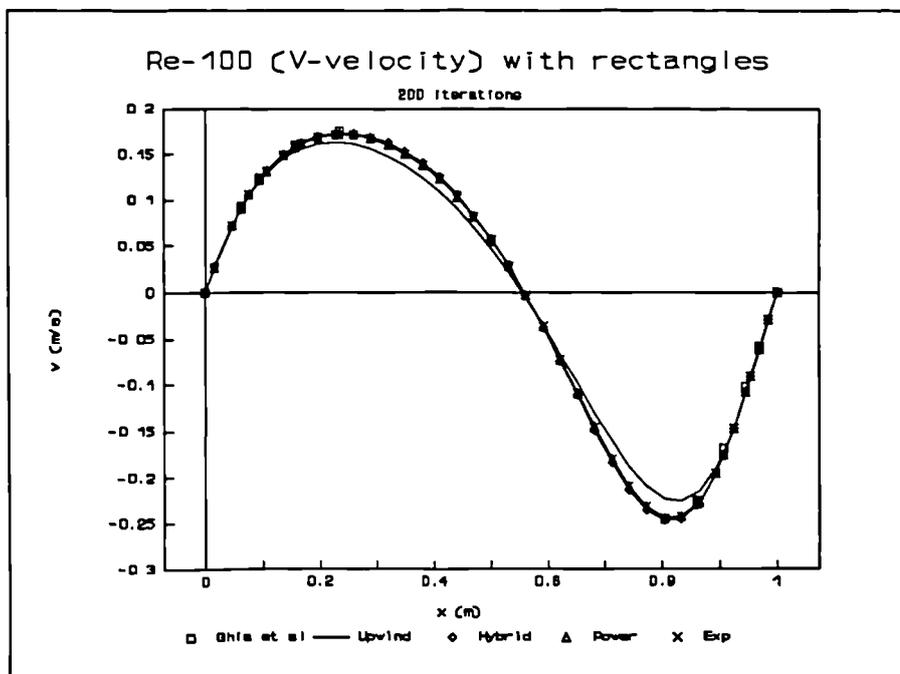


Figure 6.9b V-velocity in horizontal middle cavity for Re-100.

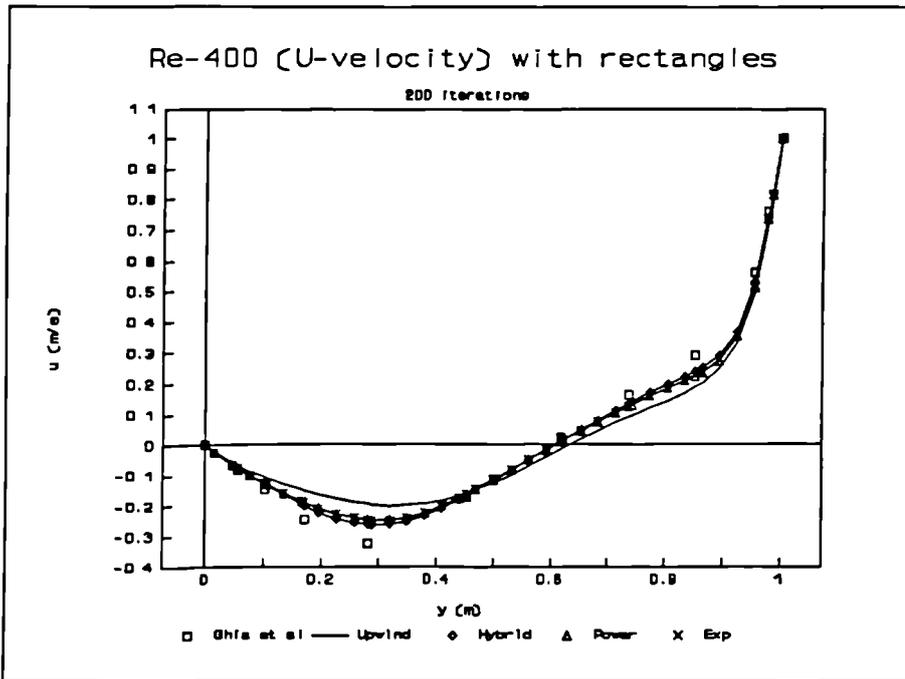


Figure 6.10a U-velocity at vertical middle of cavity for Re-400.

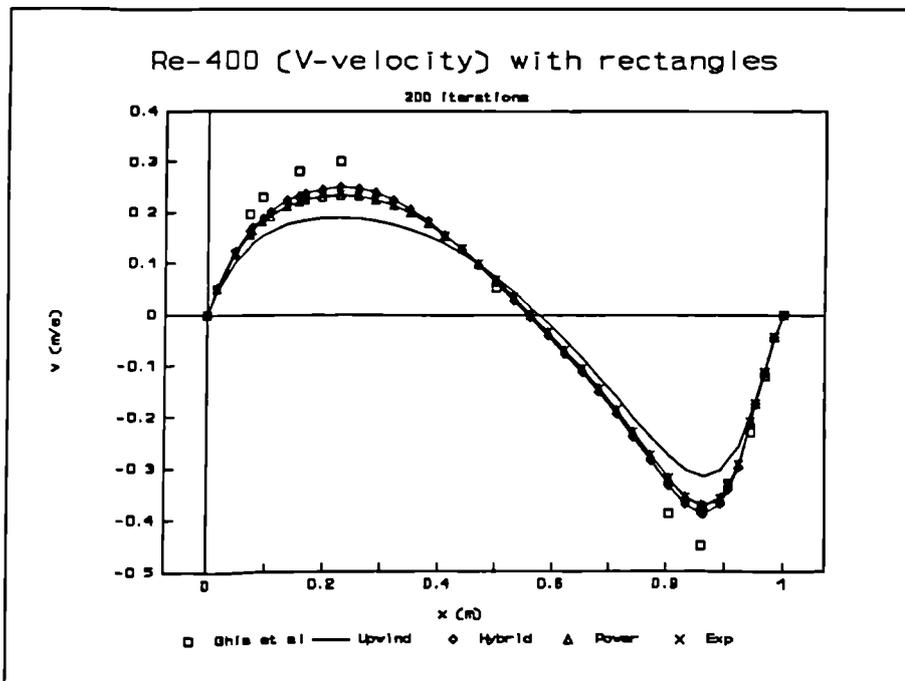


Figure 6.10b V-velocity in horizontal middle cavity for Re-400.

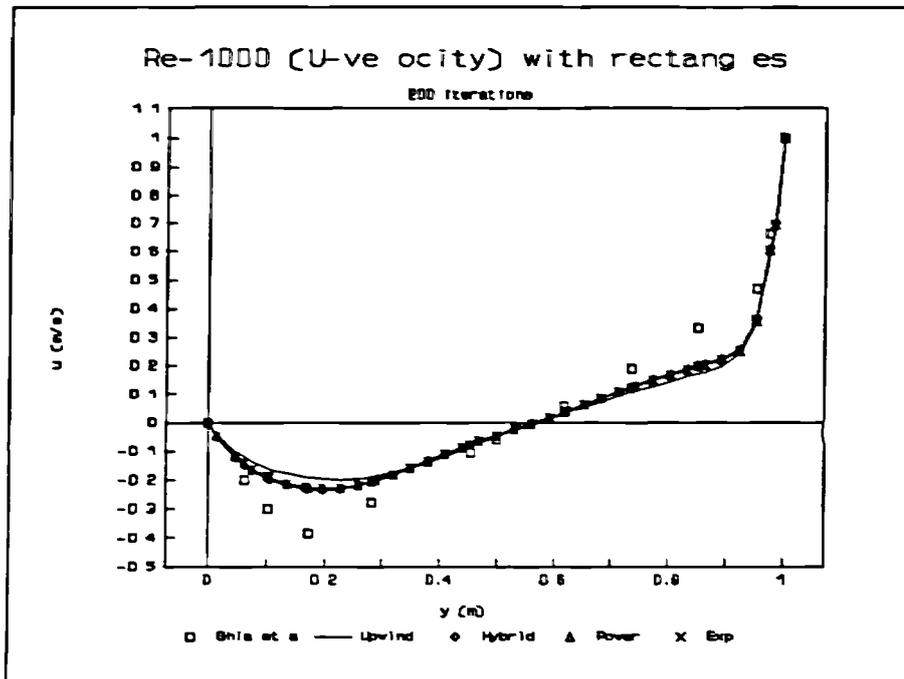


Figure 6.11a U-velocity at vertical middle of cavity for Re-1000.

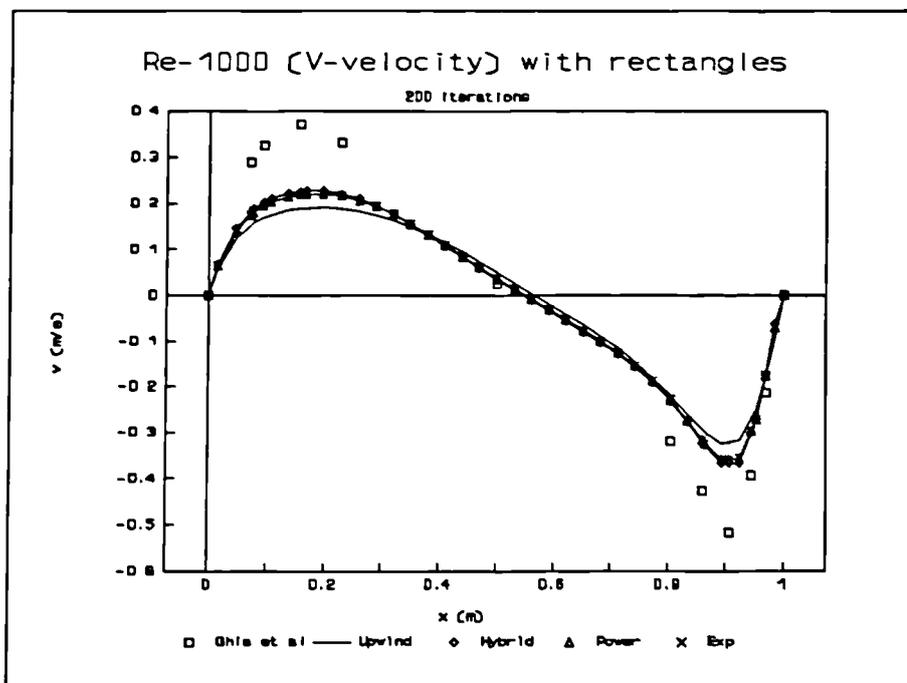


Figure 6.11b V-velocity in horizontal middle cavity for Re-1000.

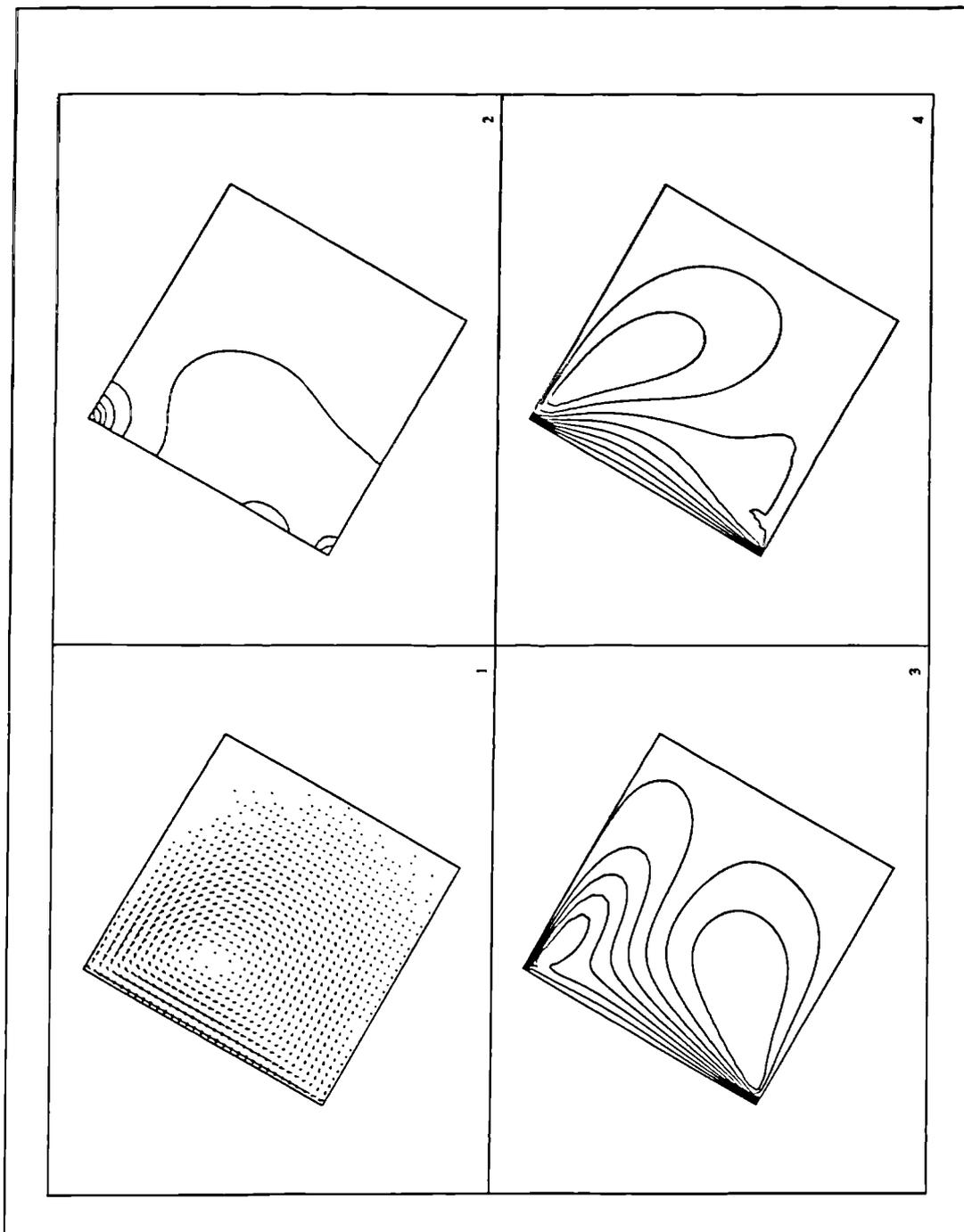


Figure 6.12 Moving lid cavity results at 60 degree for Re-100.

- 1) Velocity vector.
- 2) Pressure.
- 3) U-velocity.
- 4) V-velocity.

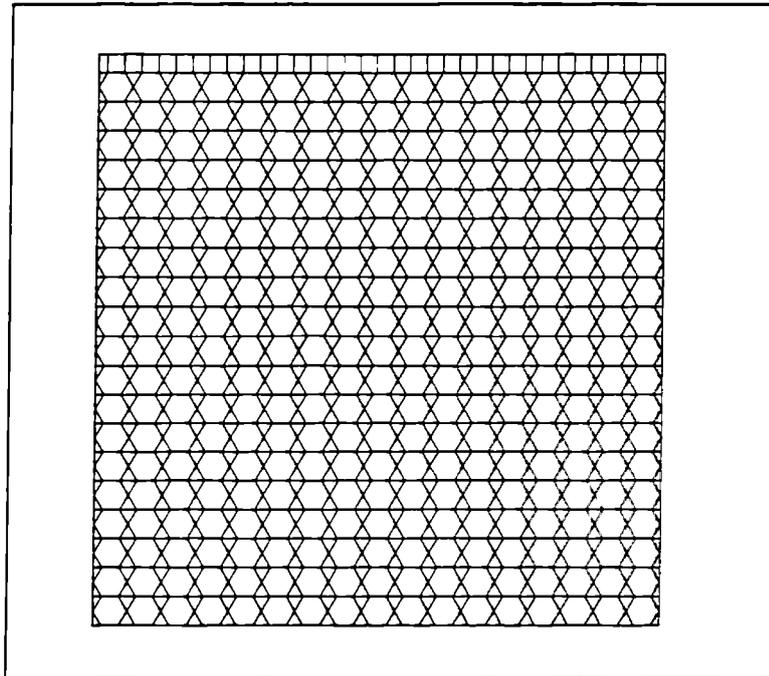


Figure 6.13 Hexagon based mesh.

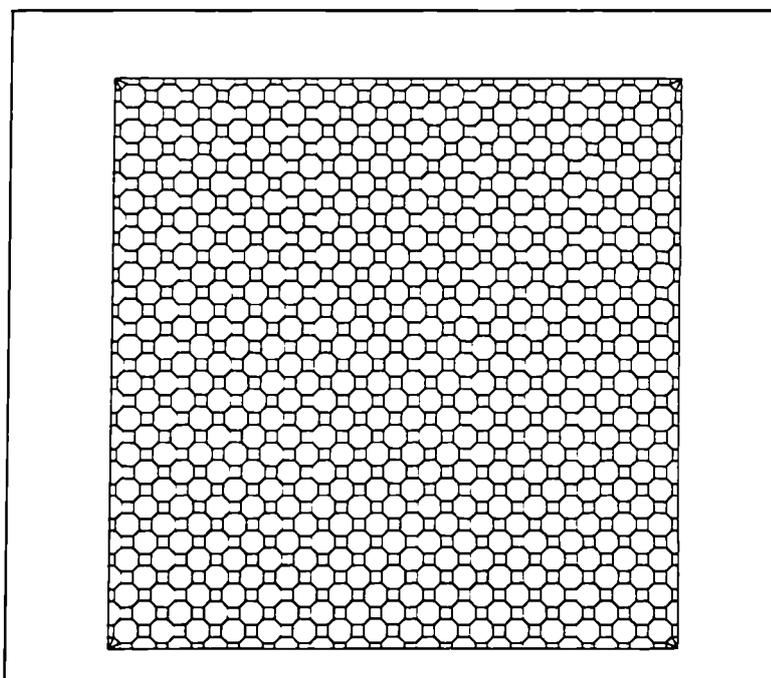


Figure 6.14 Octagon based mesh.

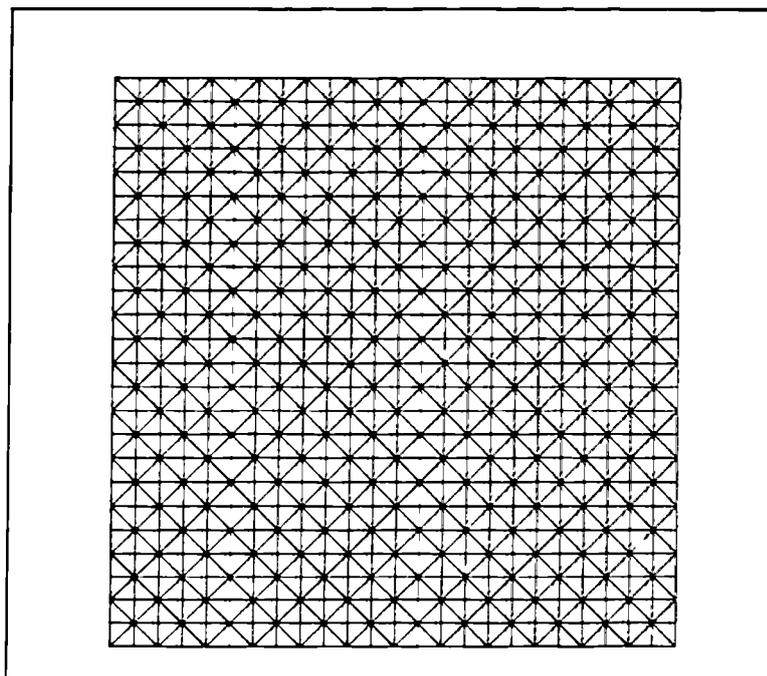


Figure 6.15 Triangle mesh.

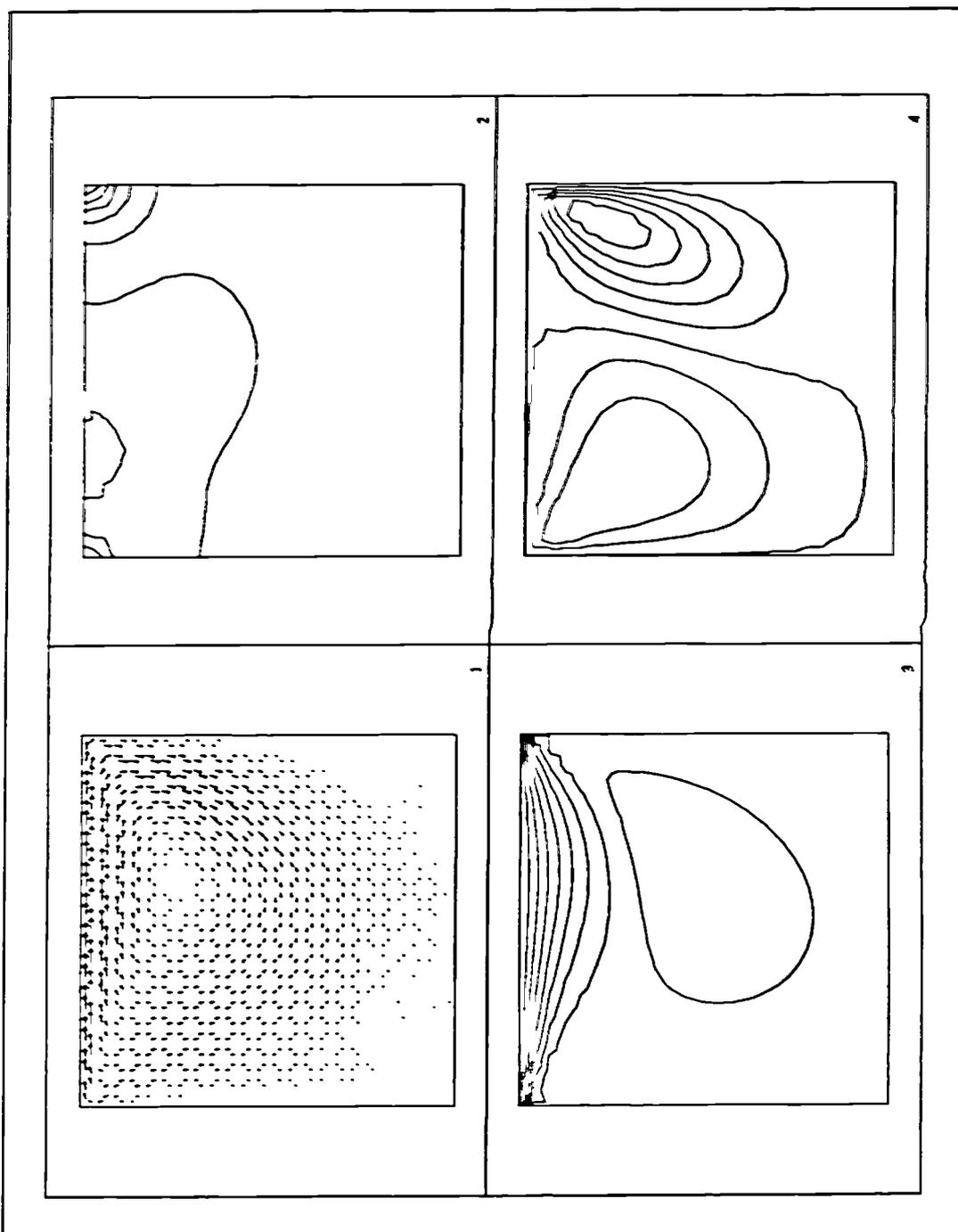


Figure 6.16 Triangle mesh results for Re-100.

- 1) Velocity vector.
- 2) Pressure.
- 3) U-velocity.
- 4) V-velocity.

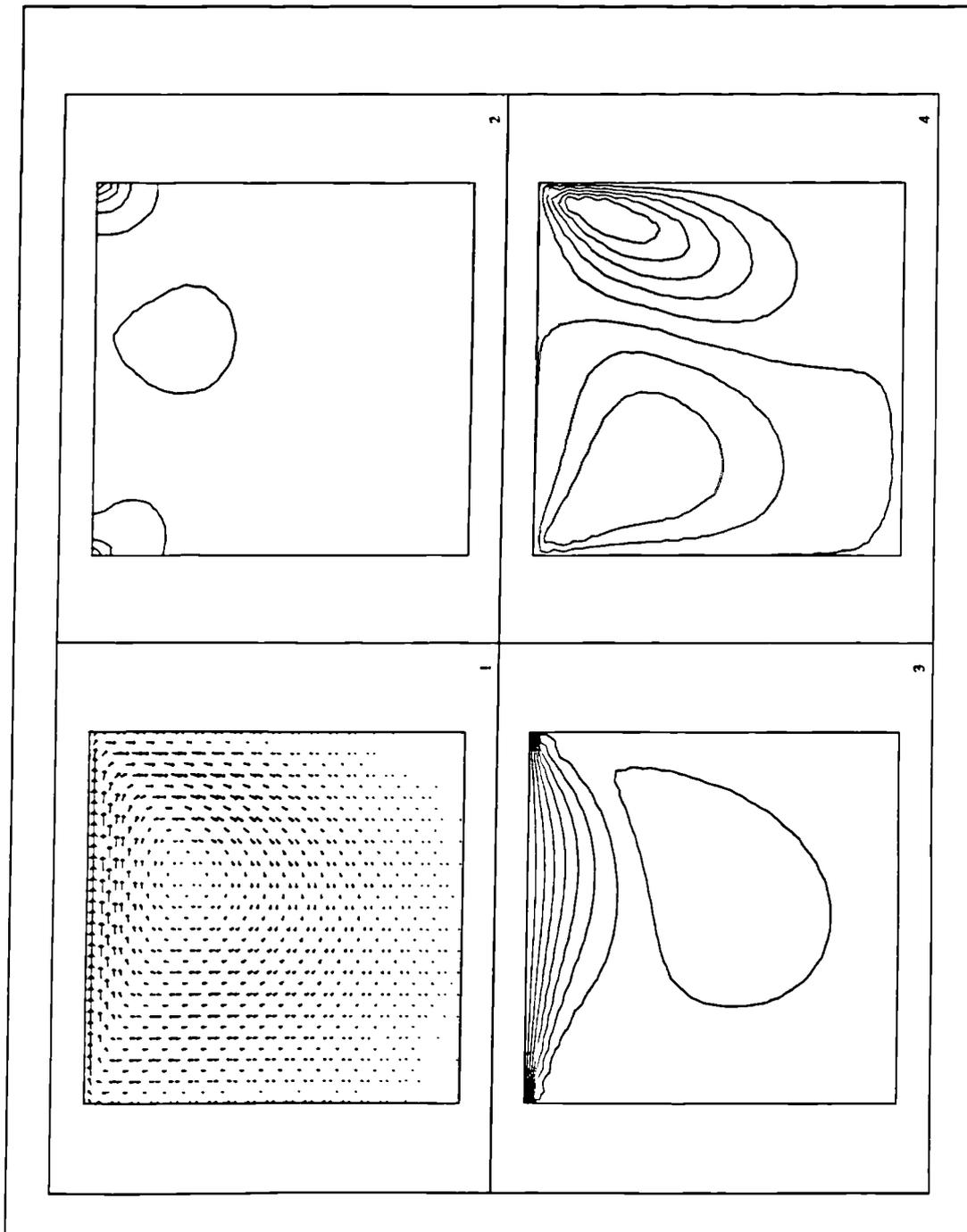


Figure 6.17 Hexagon based mesh results for $Re=100$.

- 1) Velocity vector.
- 2) Pressure.
- 3) U-velocity.
- 4) V-velocity.

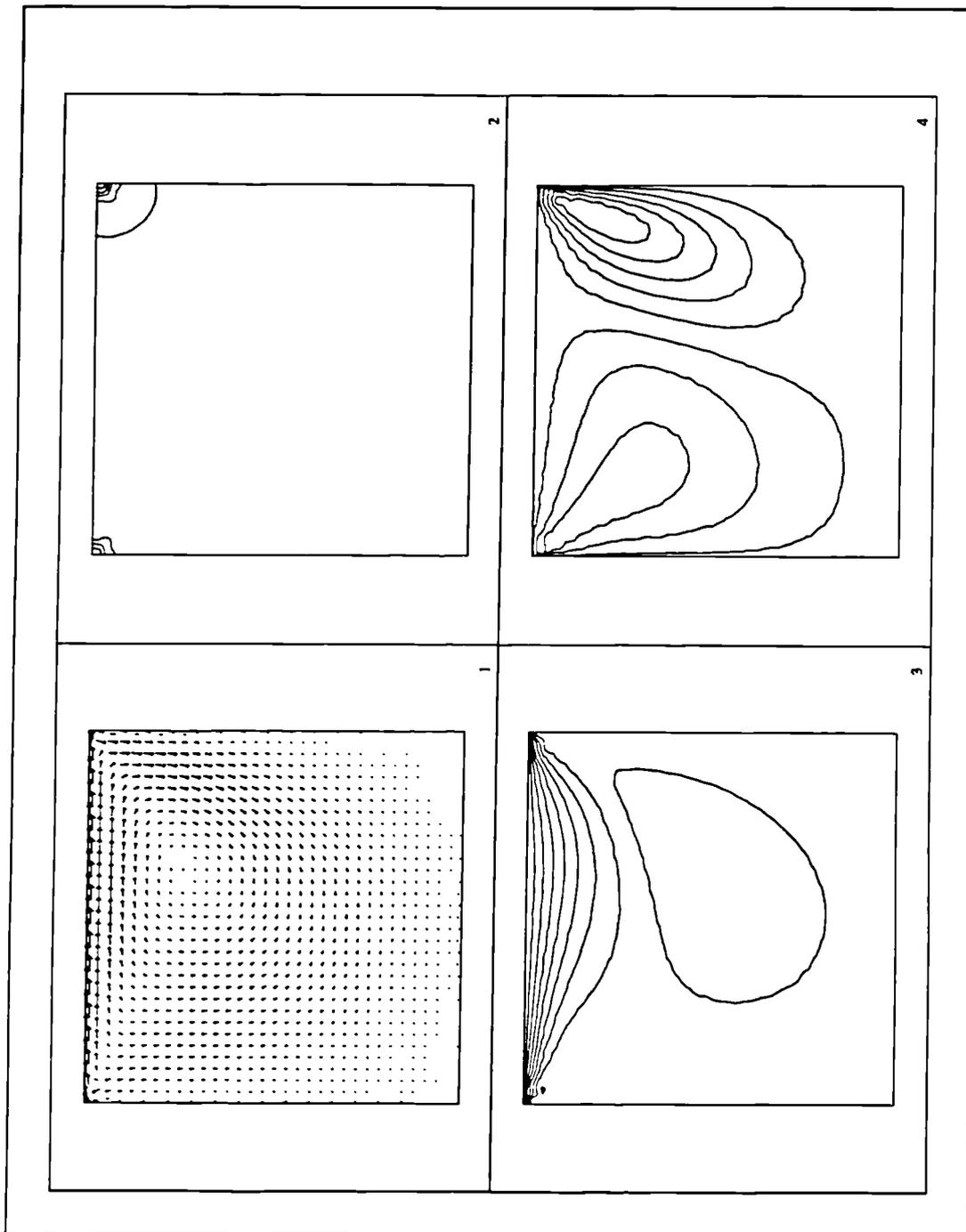


Figure 6.18 Octagon based mesh results for Re-100.

- 1) Velocity vector.
- 2) Pressure.
- 3) U-velocity.
- 4) V-velocity.

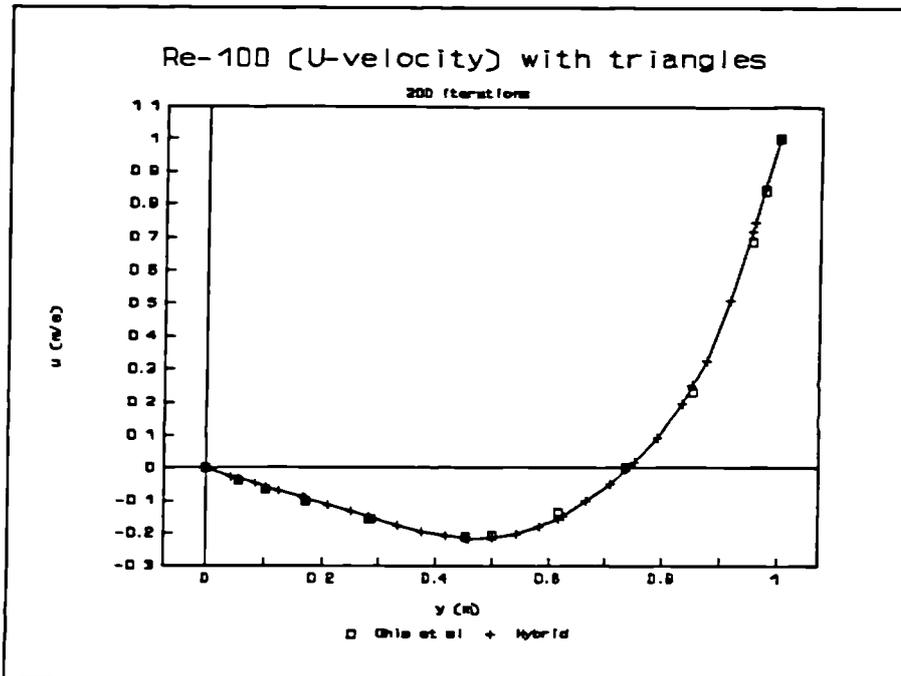


Figure 6.19a Triangle mesh results for Re-100.

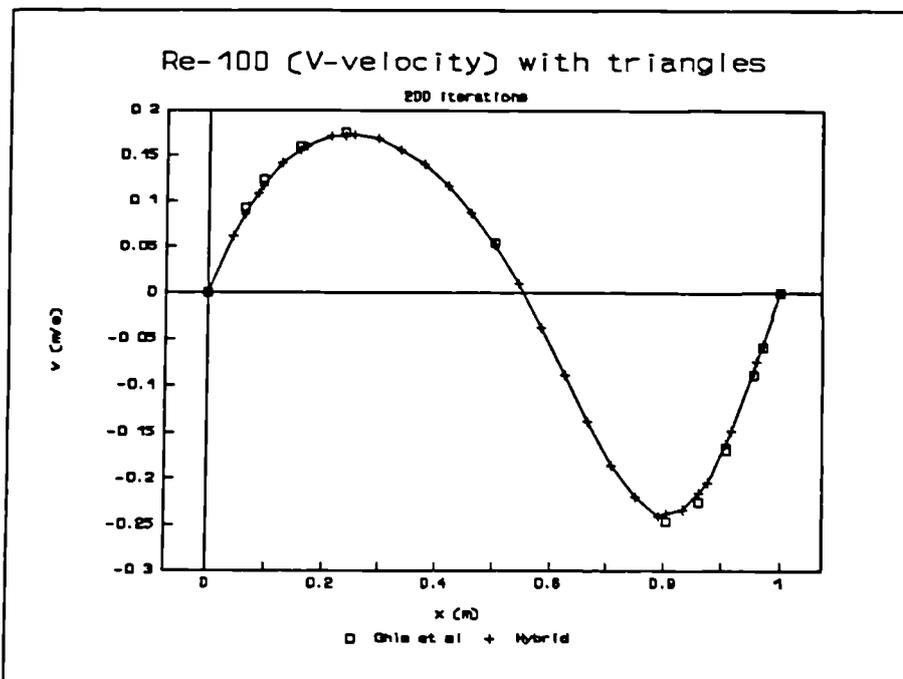


Figure 6.19b Triangle mesh results for Re-100.

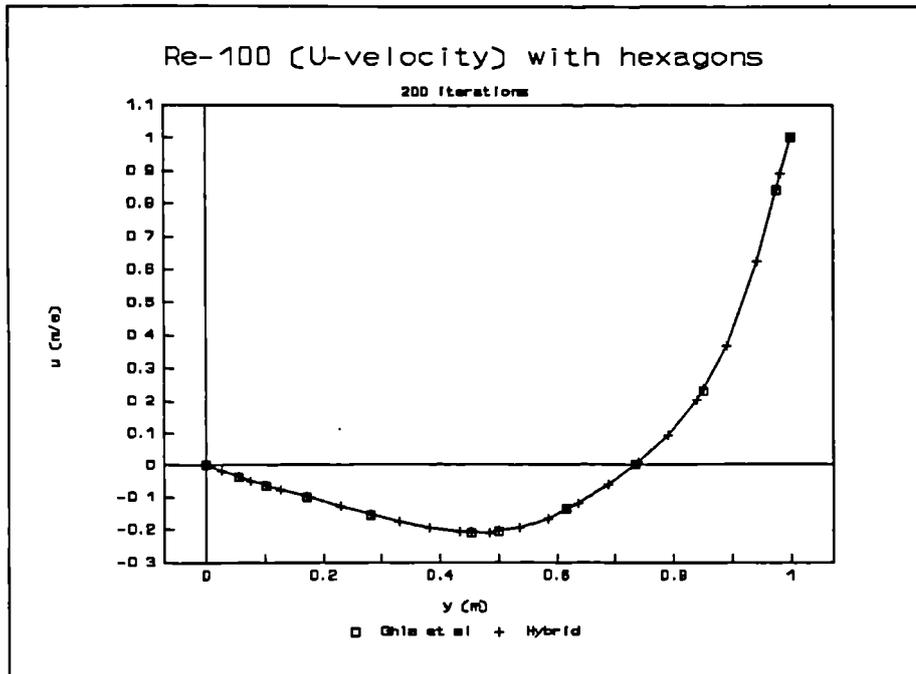


Figure 6.20a Hexagon based mesh results for Re-100.

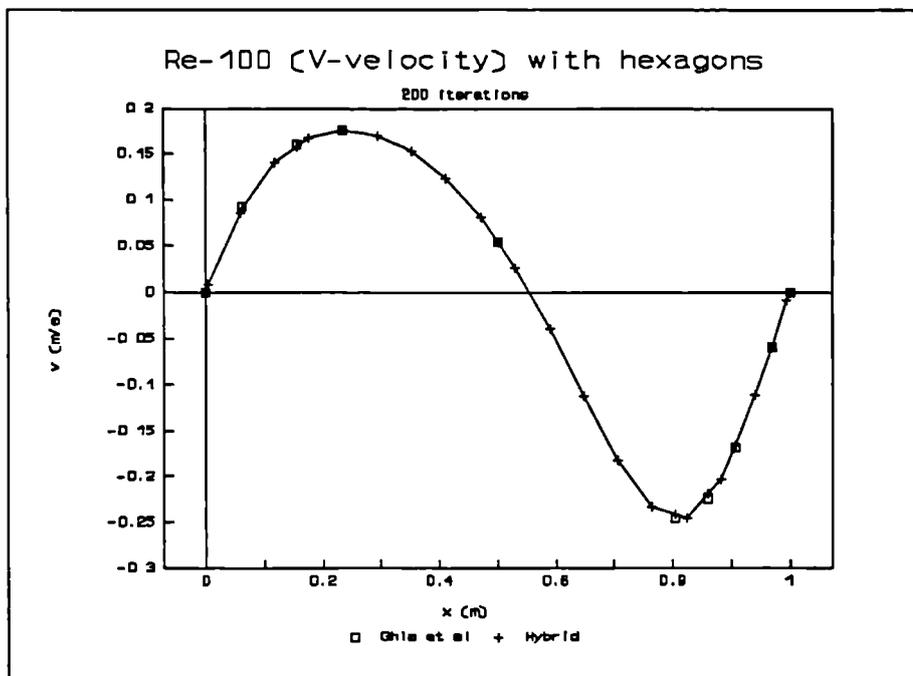


Figure 6.20b Hexagon based mesh results for Re-100.

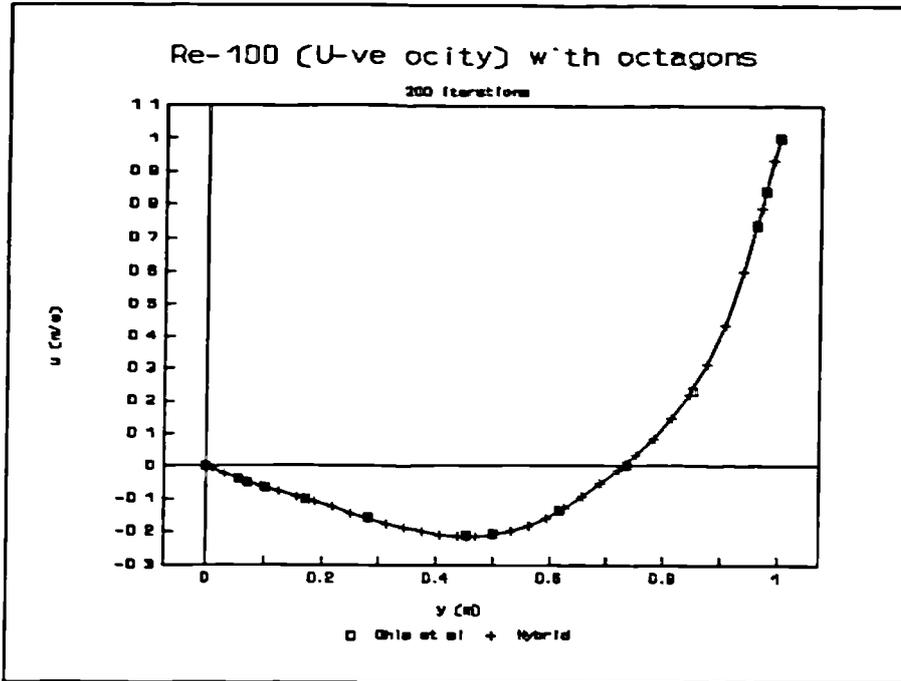


Figure 6.21a Octagon based mesh results for Re-100.

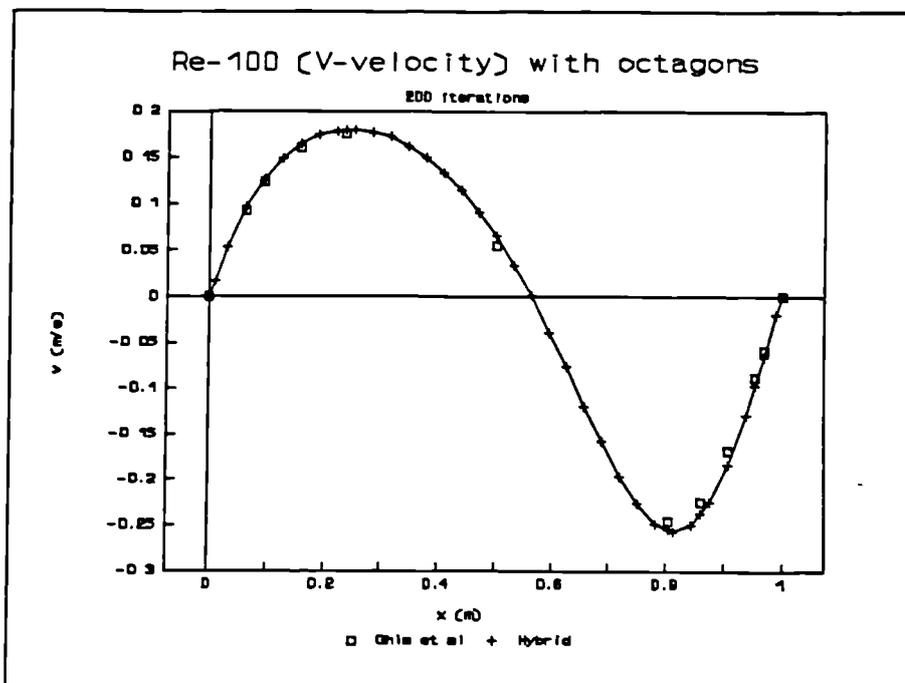


Figure 6.21b Octagon based mesh results for Re-100.

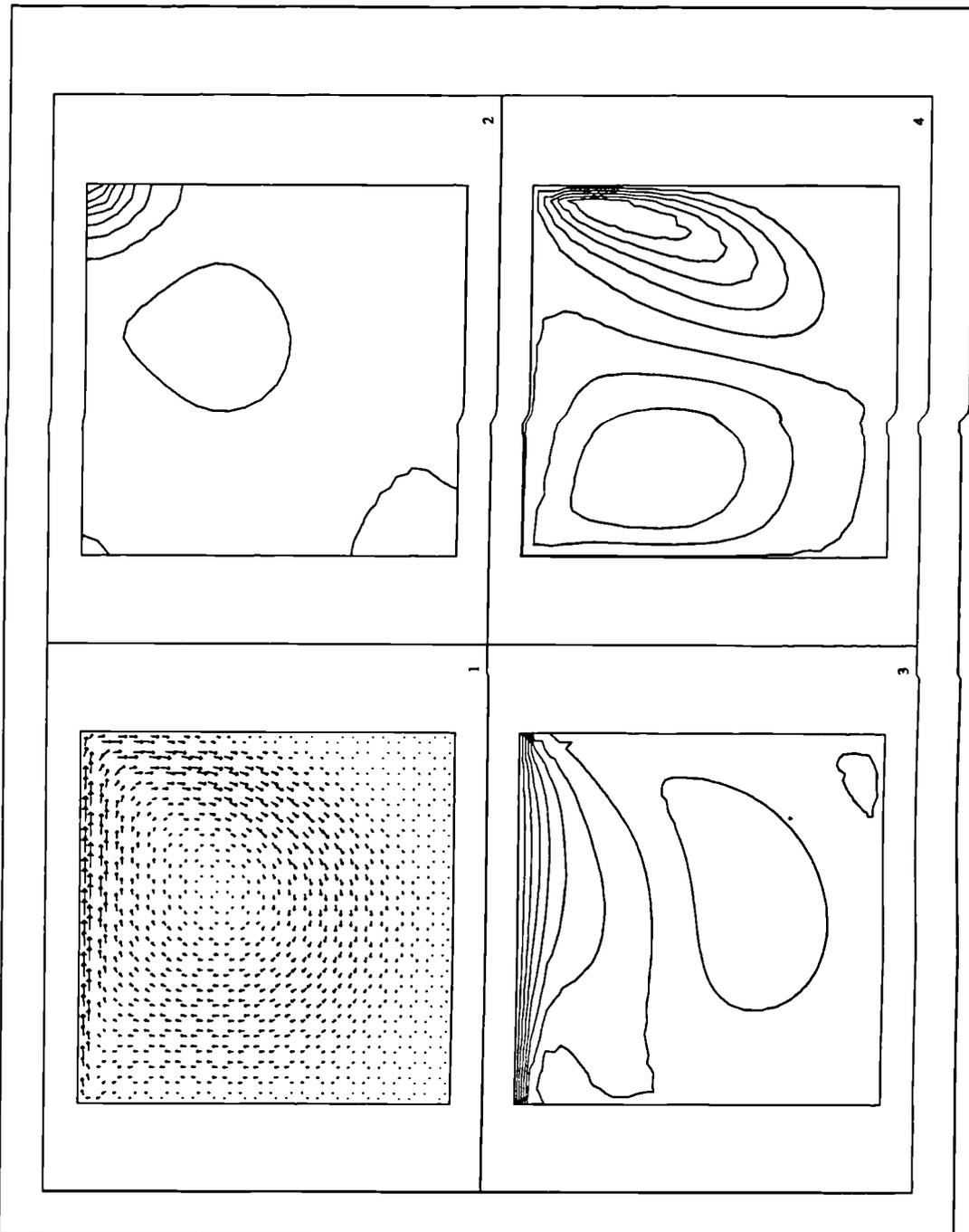


Figure 6.22 Triangle mesh results for $Re=400$.

- 1) Velocity vector.
- 2) Pressure.
- 3) U-velocity.
- 4) V-velocity.

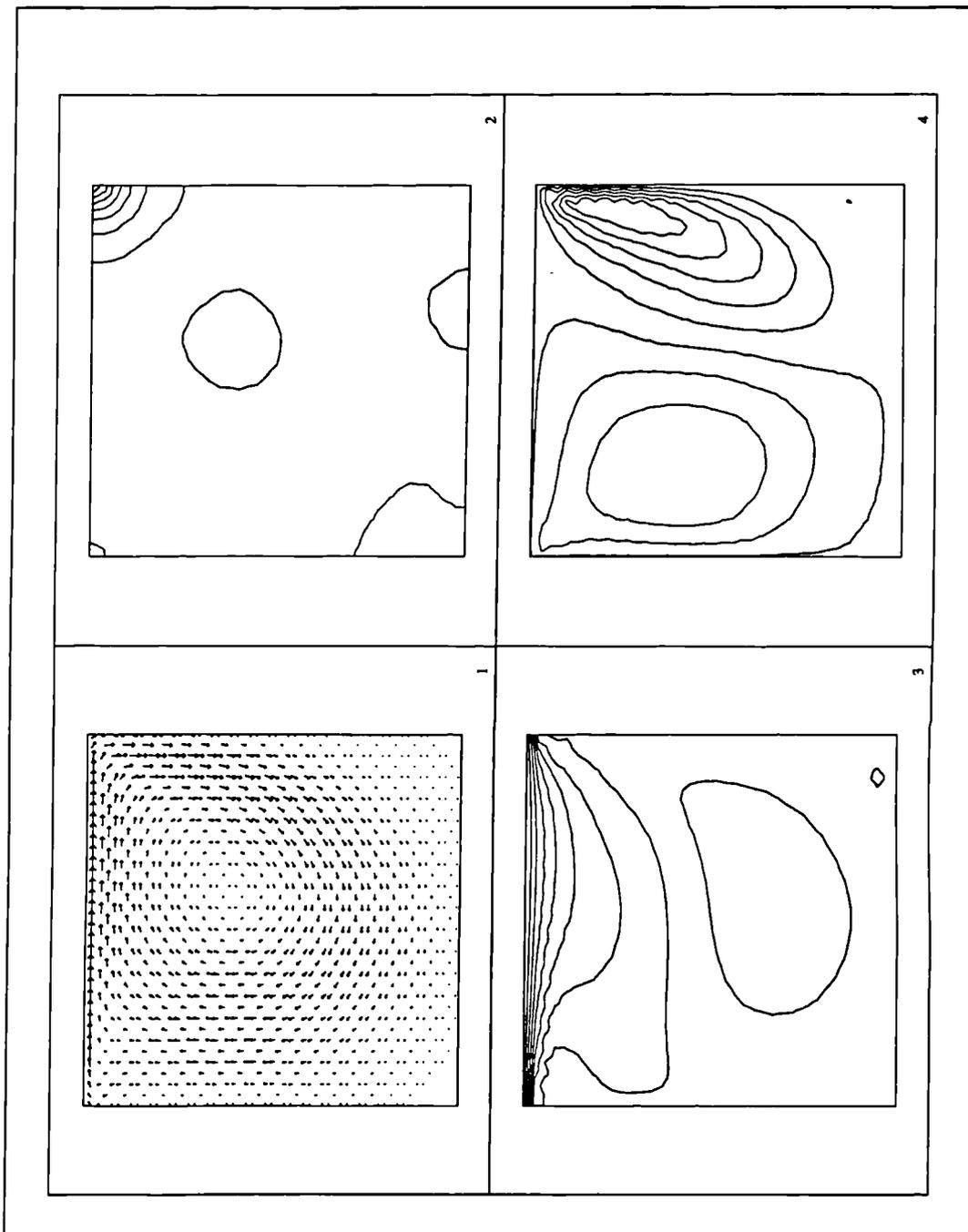


Figure 6.23 Hexagon based mesh results for $Re=400$.

- 1) Velocity vector.
- 2) Pressure.
- 3) U-velocity.
- 4) V-velocity.

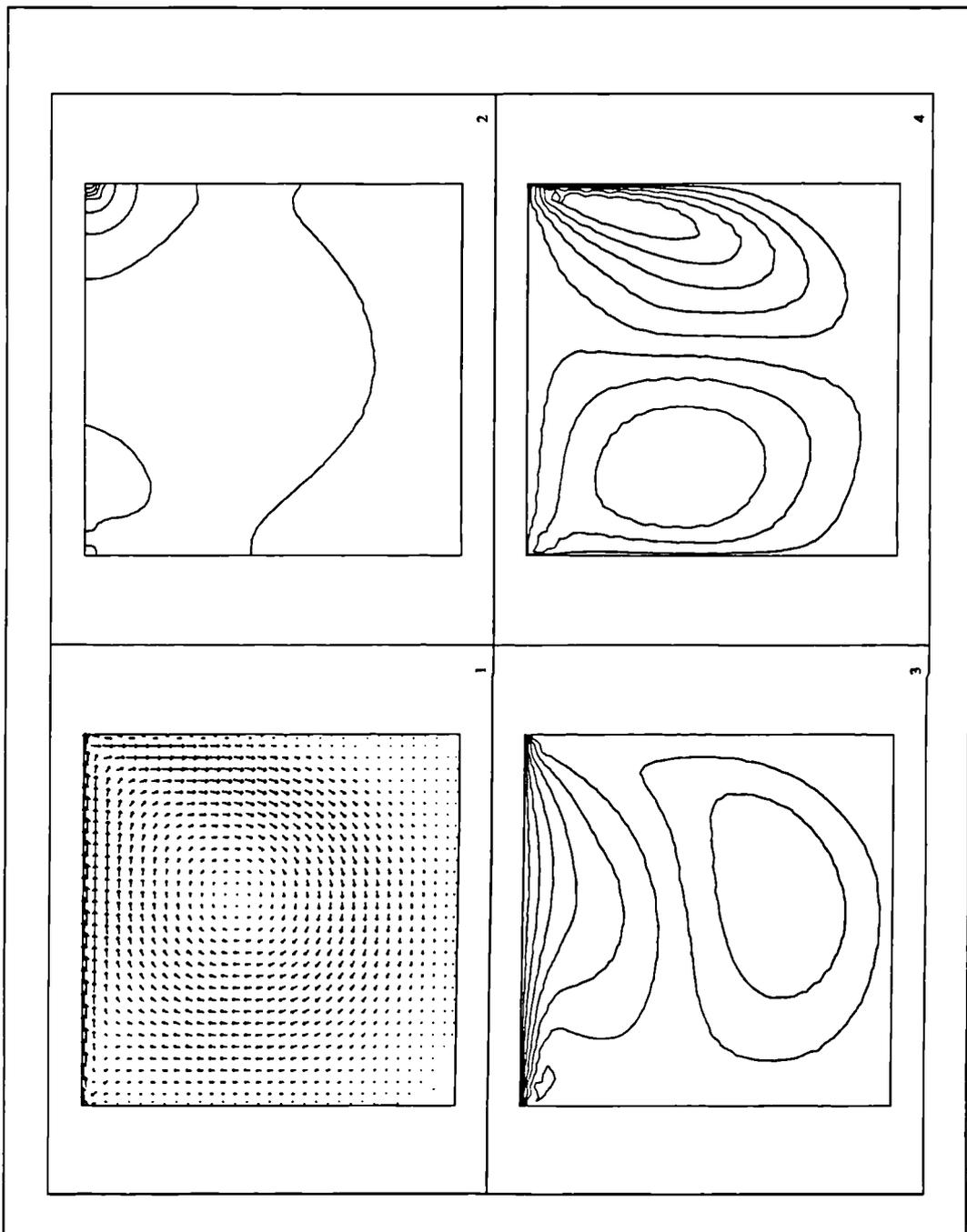


Figure 6.24 Octagon based mesh results for $Re=400$.

- 1) Velocity vector.
- 2) Pressure.
- 3) U-velocity.
- 4) V-velocity.

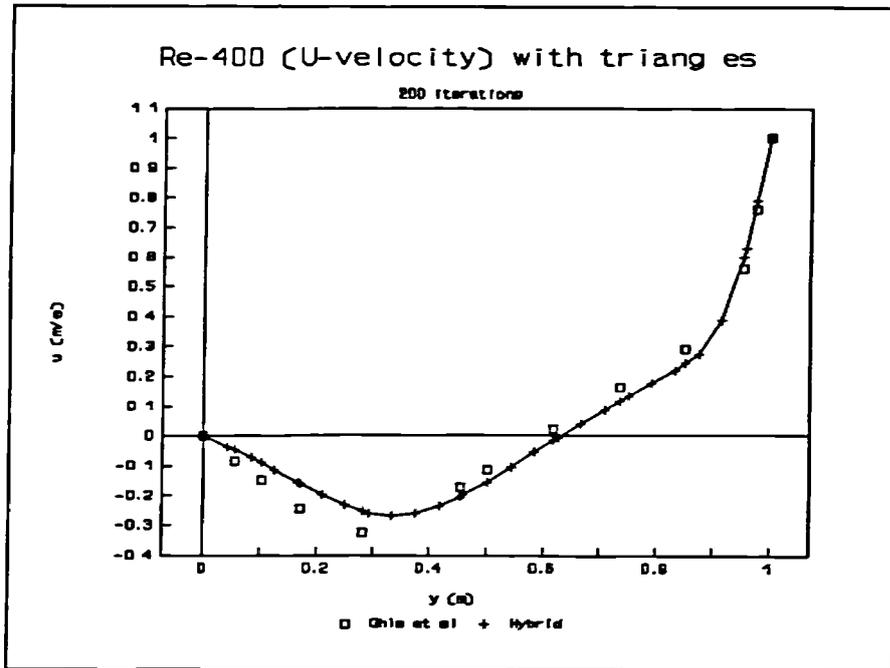


Figure 6.25a Triangle mesh results for Re-400.

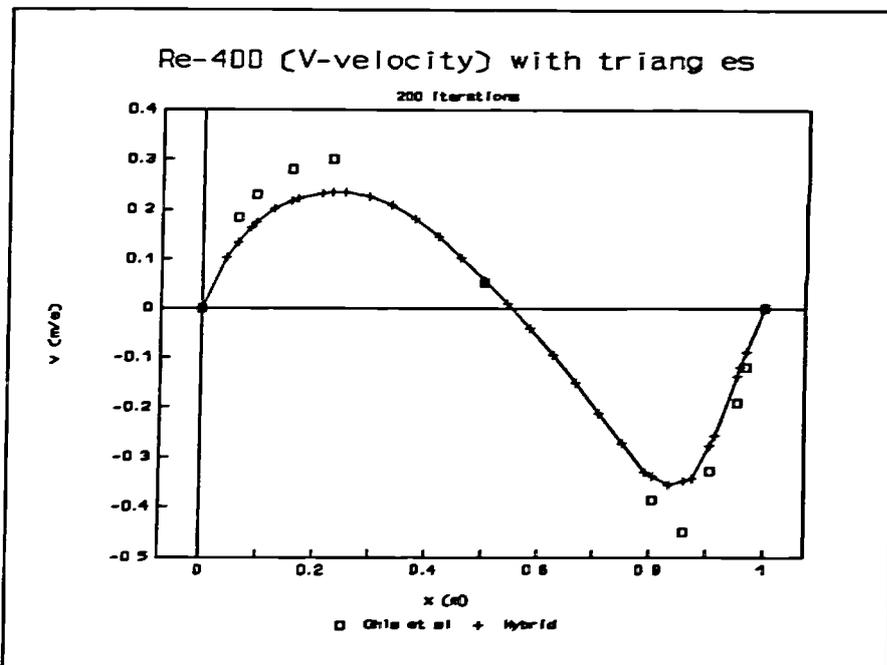


Figure 6.25b Triangle mesh results for Re-400.

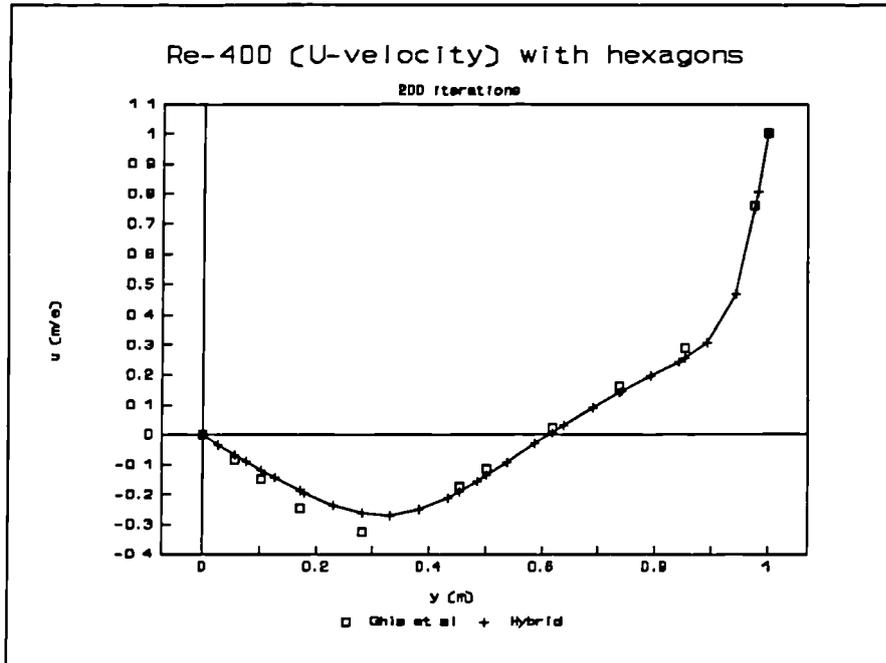


Figure 6.26a Hexagon based mesh results for Re-400.

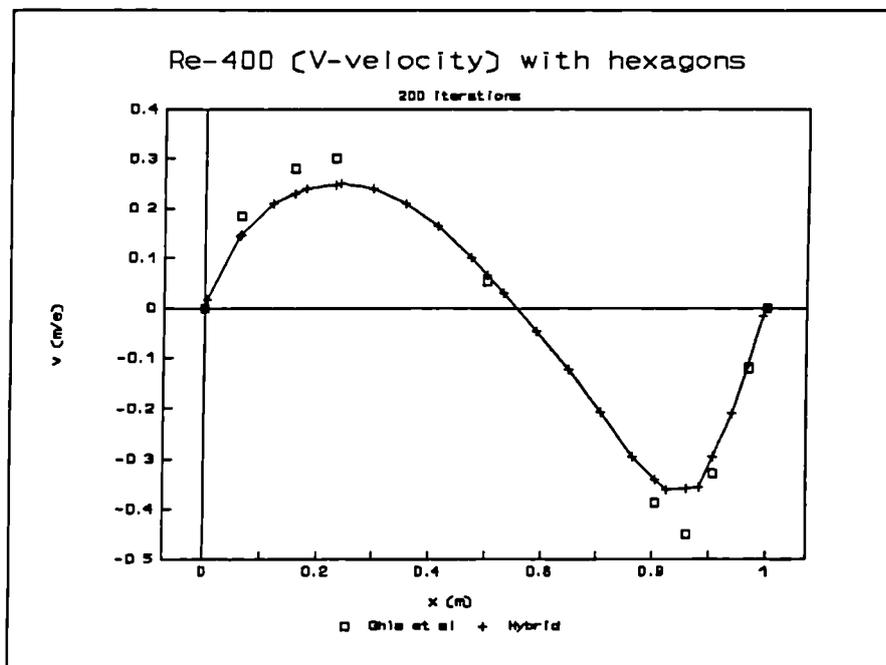


Figure 6.26b Hexagon based mesh results for Re-400.

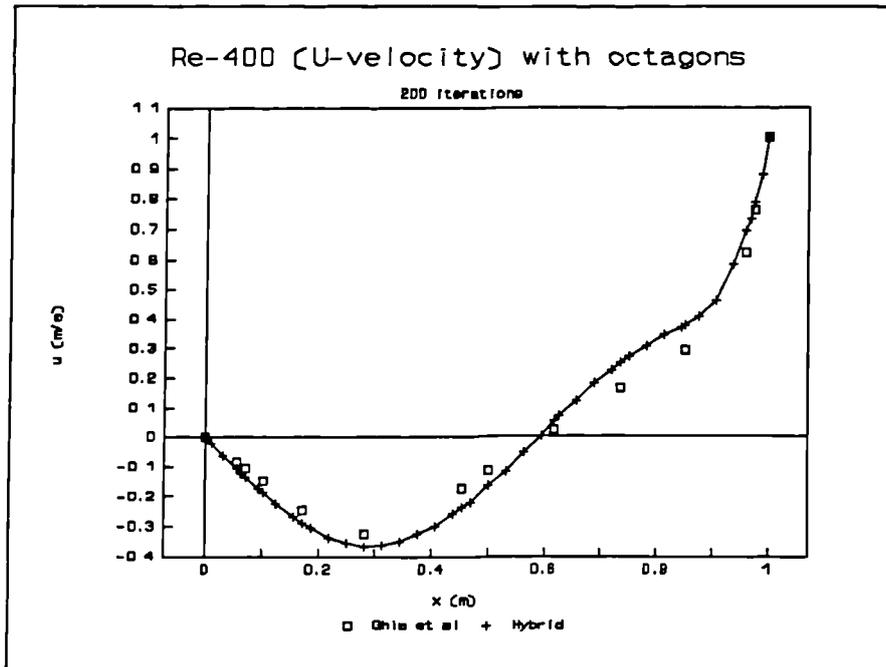


Figure 6.27a Octagon based mesh results for Re-400.

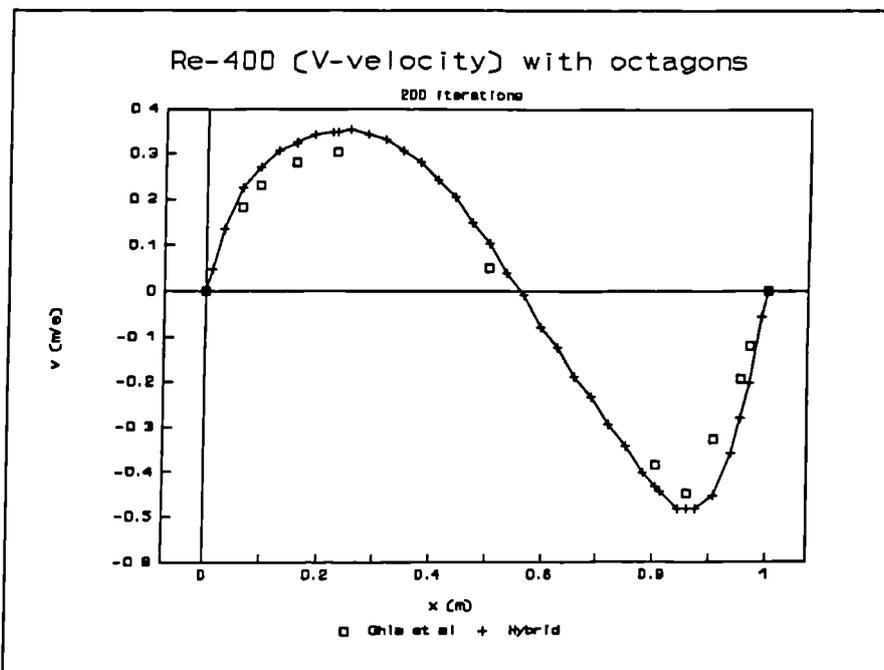


Figure 6.27b Octagon based mesh results for Re-400.

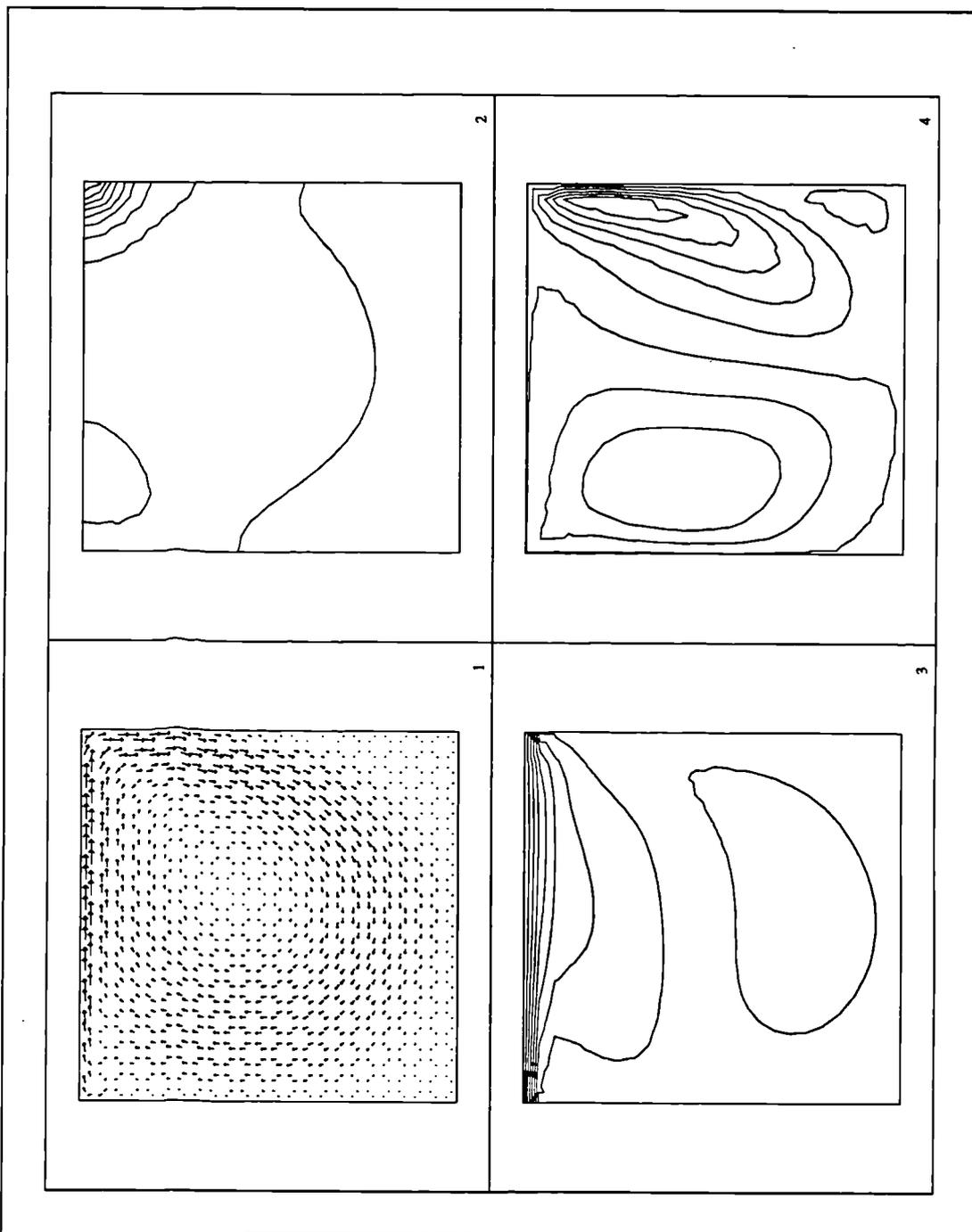


Figure 6.28 Triangle mesh results for $Re=1000$

- 1) Velocity vector.
- 2) Pressure.
- 3) U-velocity.
- 4) V-velocity.

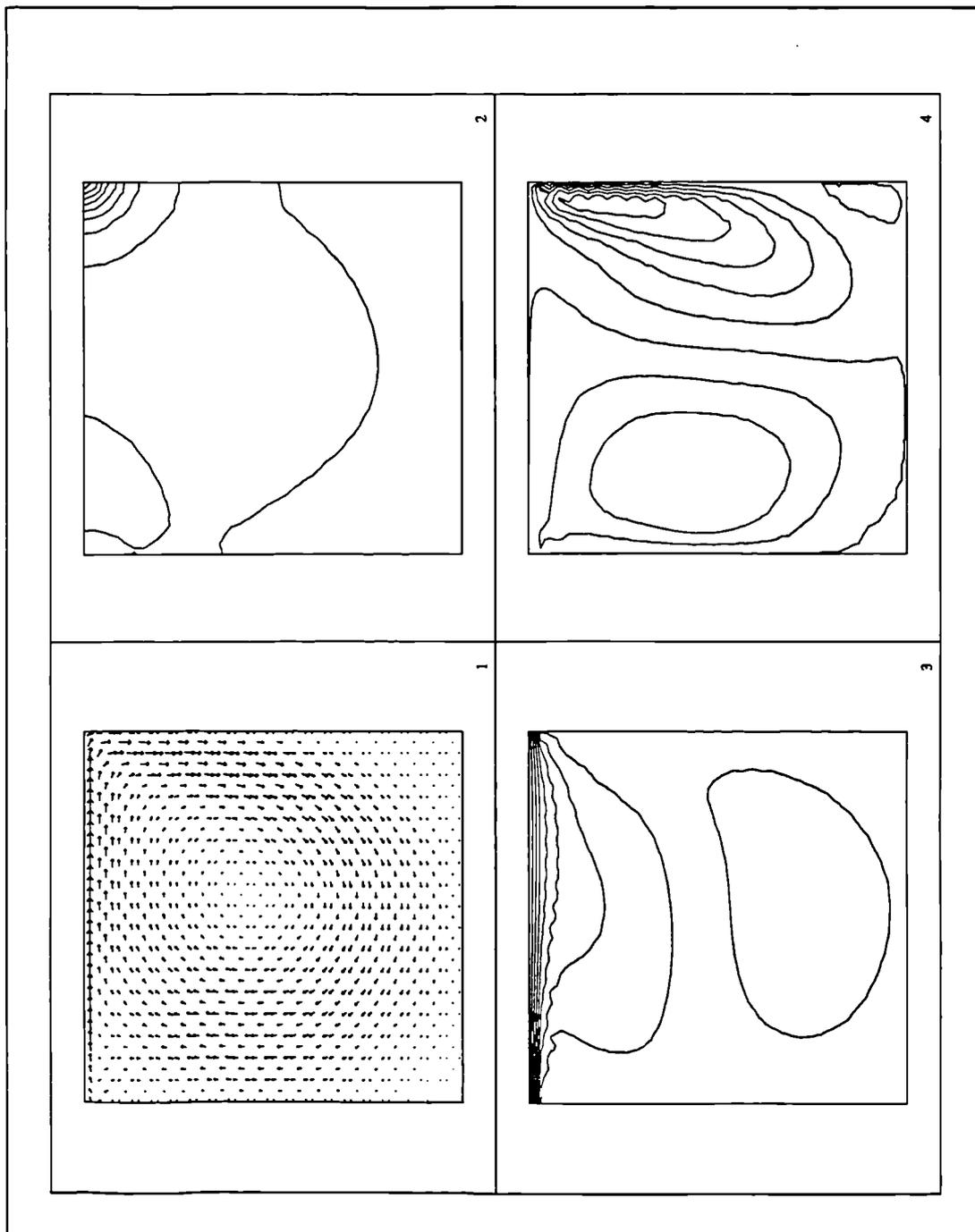


Figure 6.29 Hexagon based mesh for $Re=1000$.

- 1) Velocity vector.
- 2) Pressure.
- 3) U-velocity.
- 4) V-velocity.

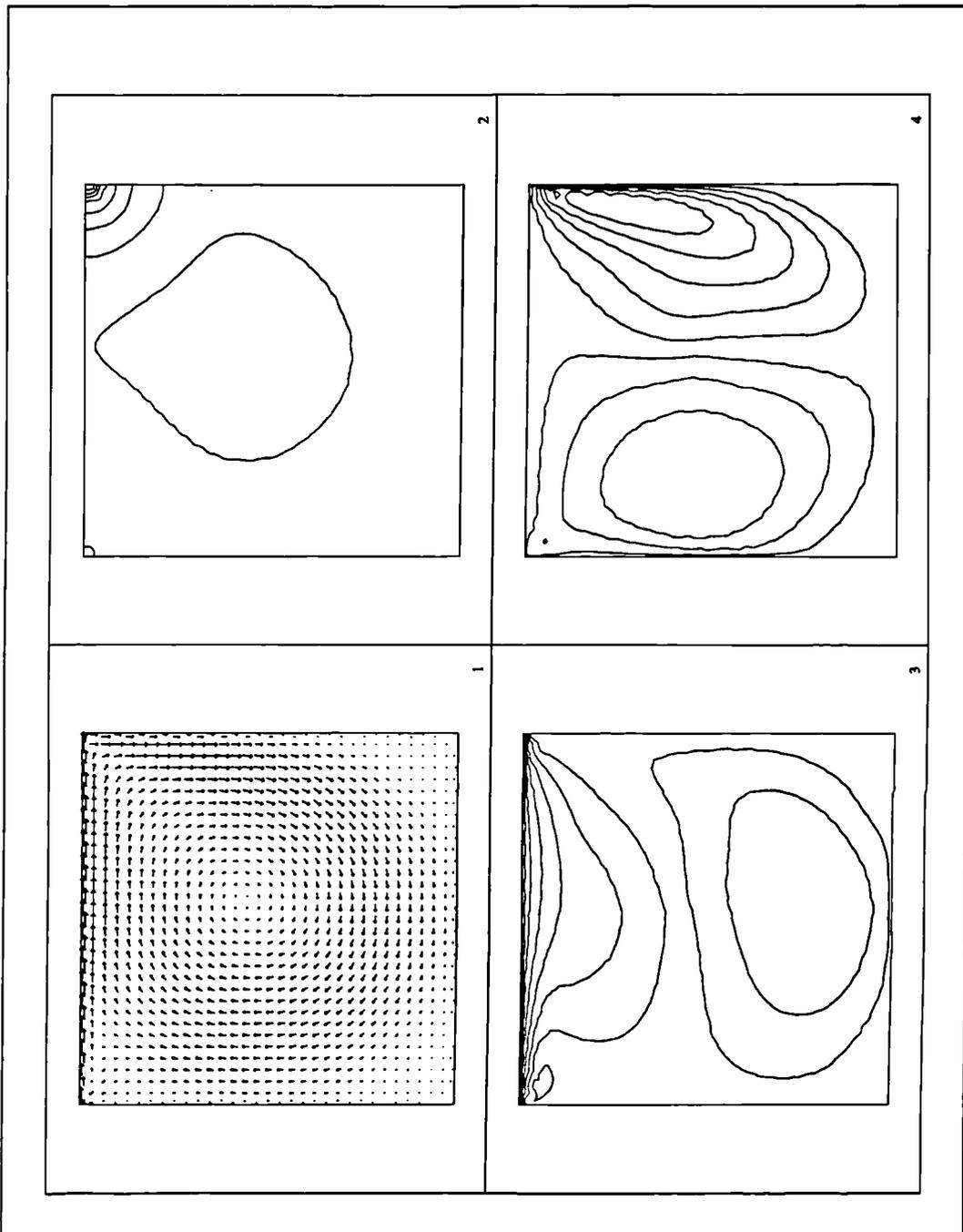


Figure 6.30 Octagon based mesh results for $Re=1000$.

- 1) Velocity vector.
- 2) Pressure.
- 3) U-velocity.
- 4) V-velocity.

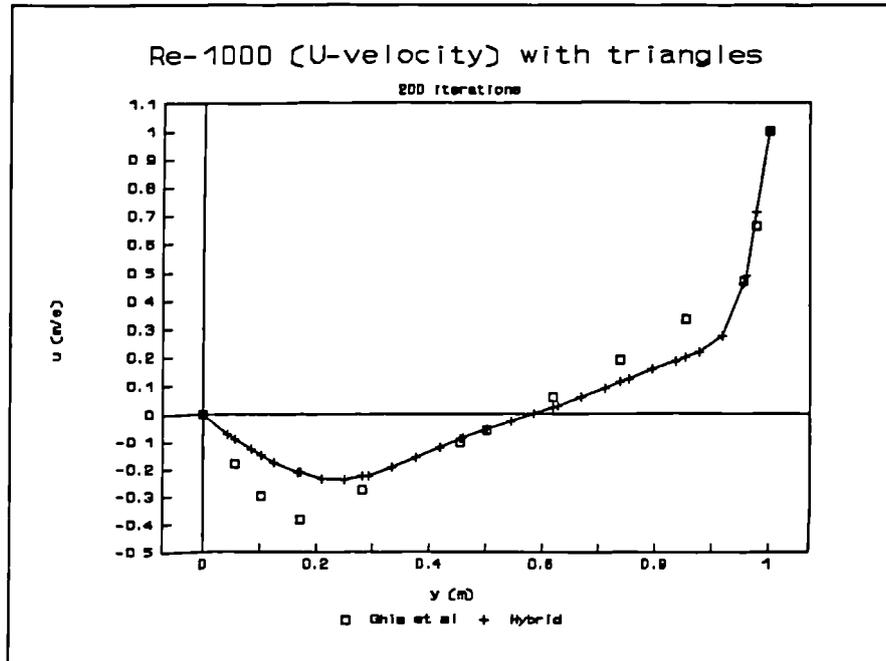


Figure 6.31a Triangle mesh results for Re-1000.

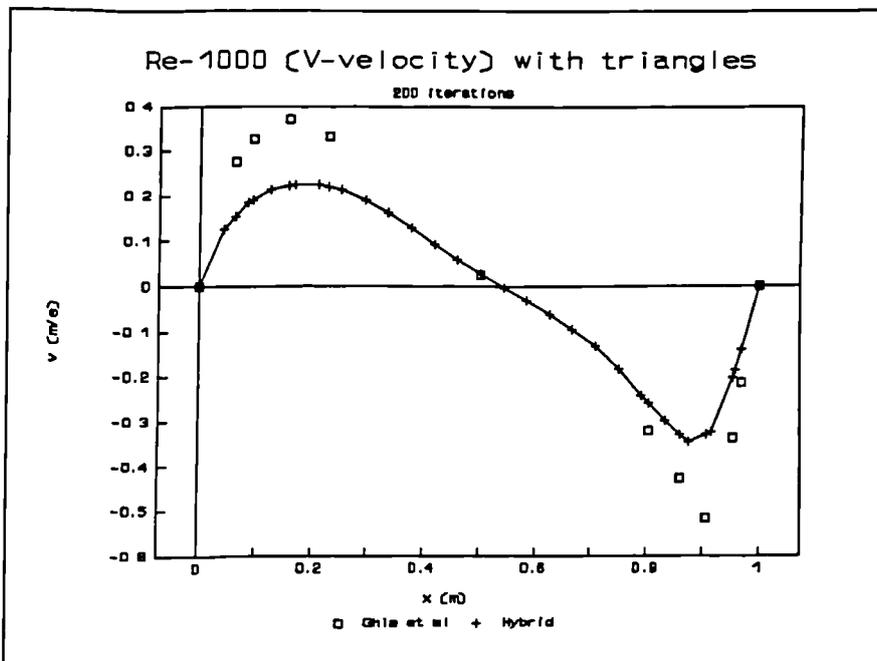


Figure 6.31b Triangle mesh results for Re-1000.

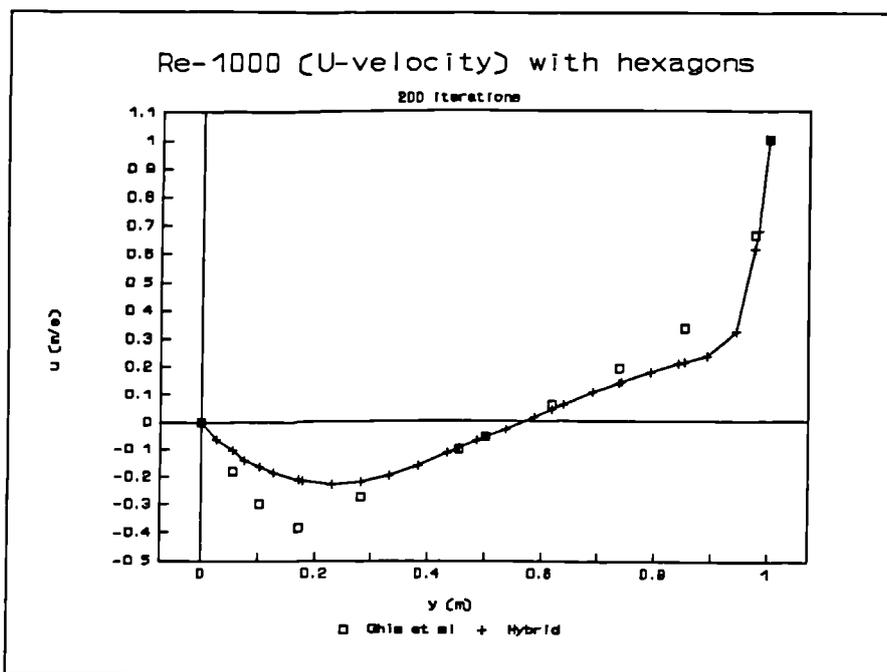


Figure 6.32a Hexagon based mesh results for Re-1000.

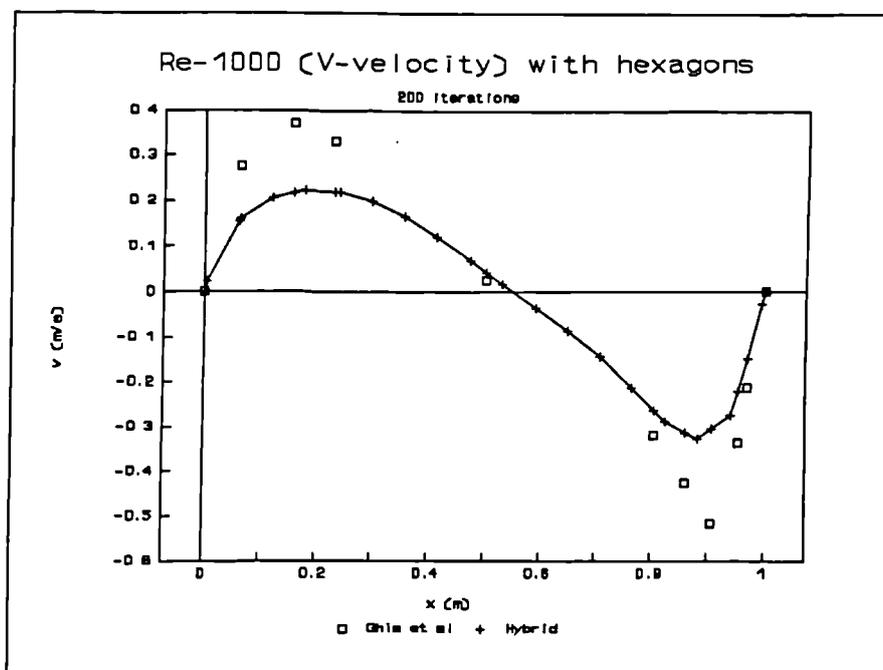


Figure 6.32b Hexagon based mesh results for Re-1000.

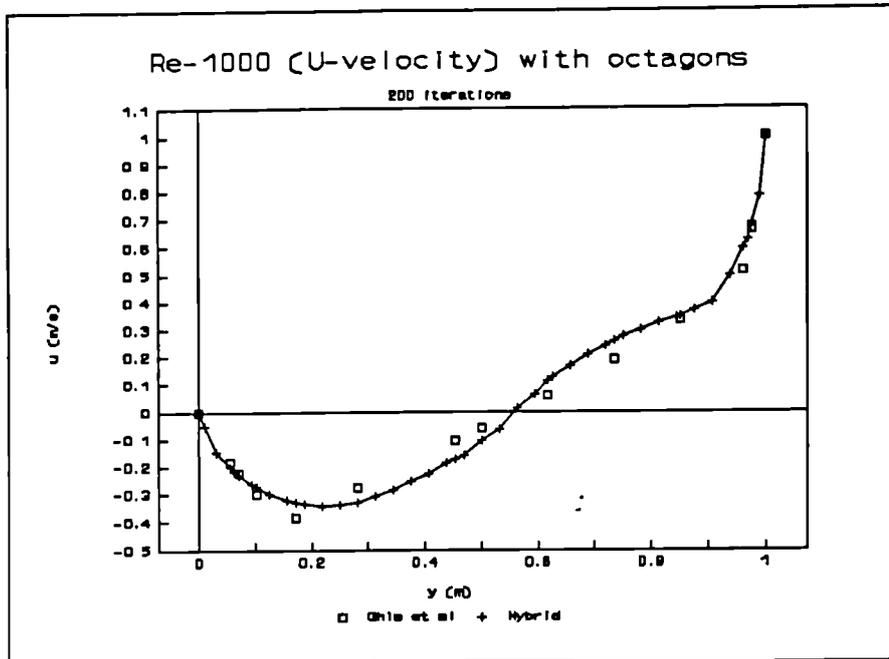


Figure 6.33a Octagon based mesh results for Re-1000.

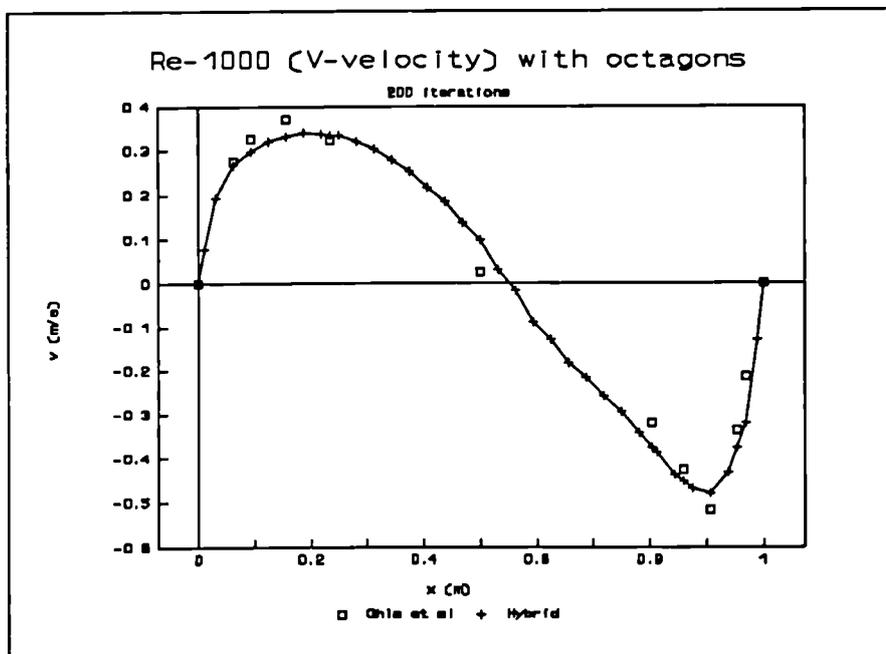


Figure 6.33b Octagon based mesh results for Re-1000.

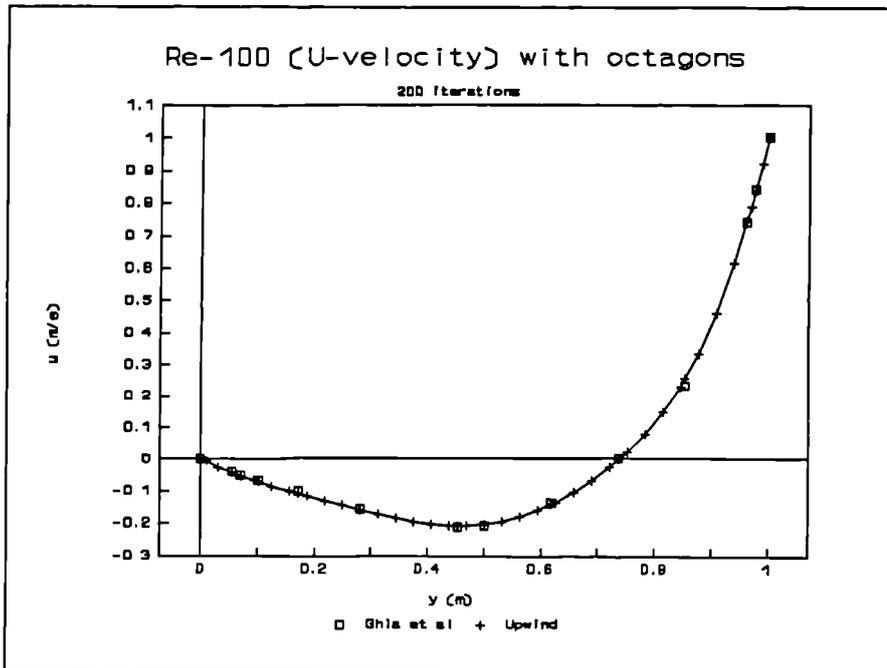


Figure 6.34a Octagon based mesh with upwind for Re-100.

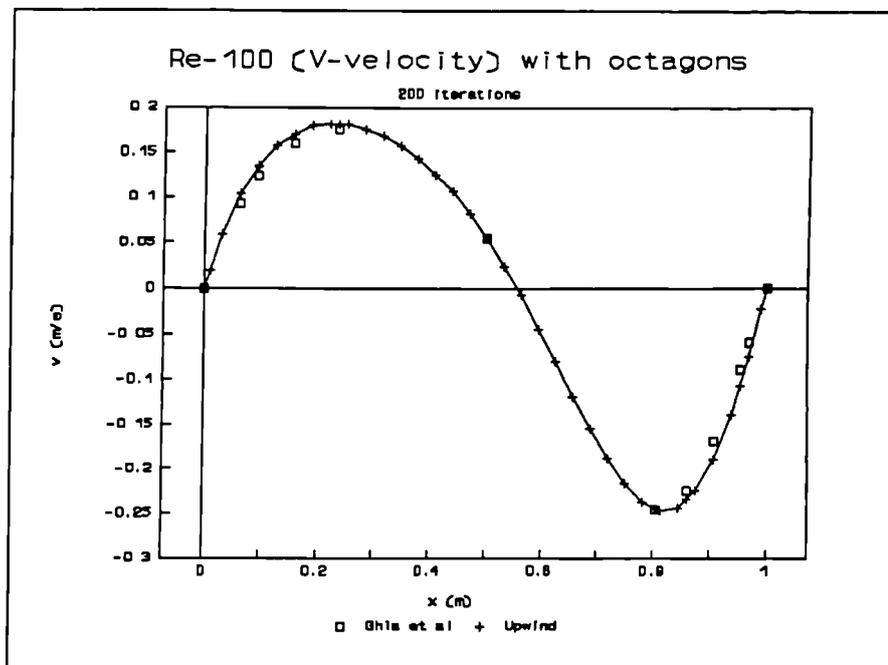


Figure 6.34b Octagon based mesh with upwind for Re-100.

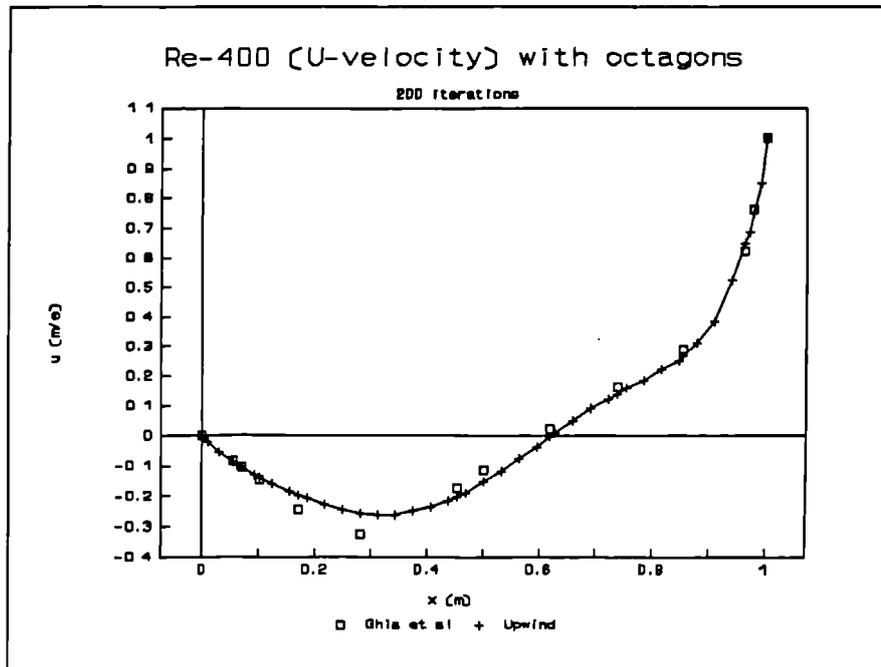


Figure 6.35a Octagon based mesh with upwind for Re-400.

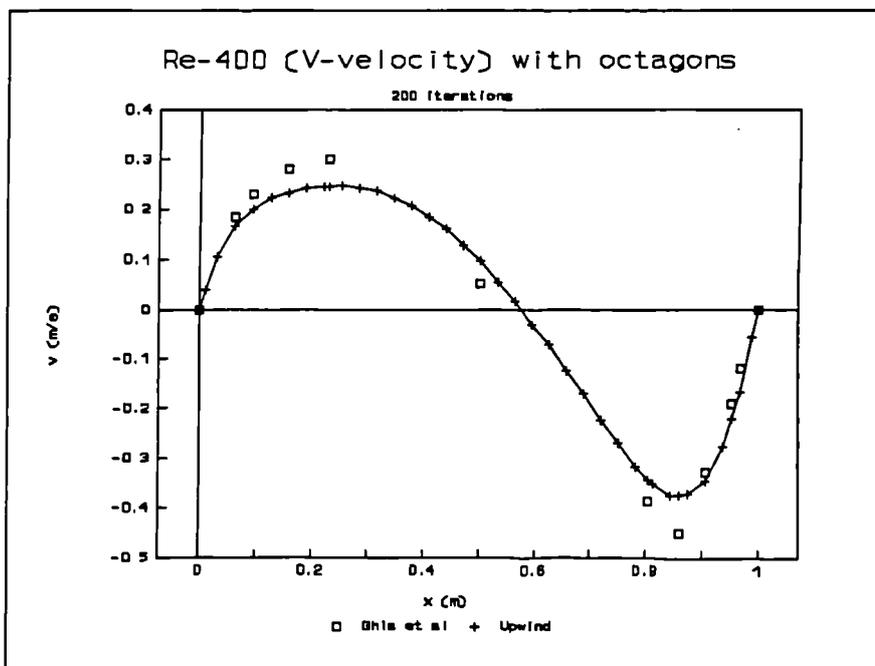


Figure 6.35b Octagon based mesh with upwind for Re-400.

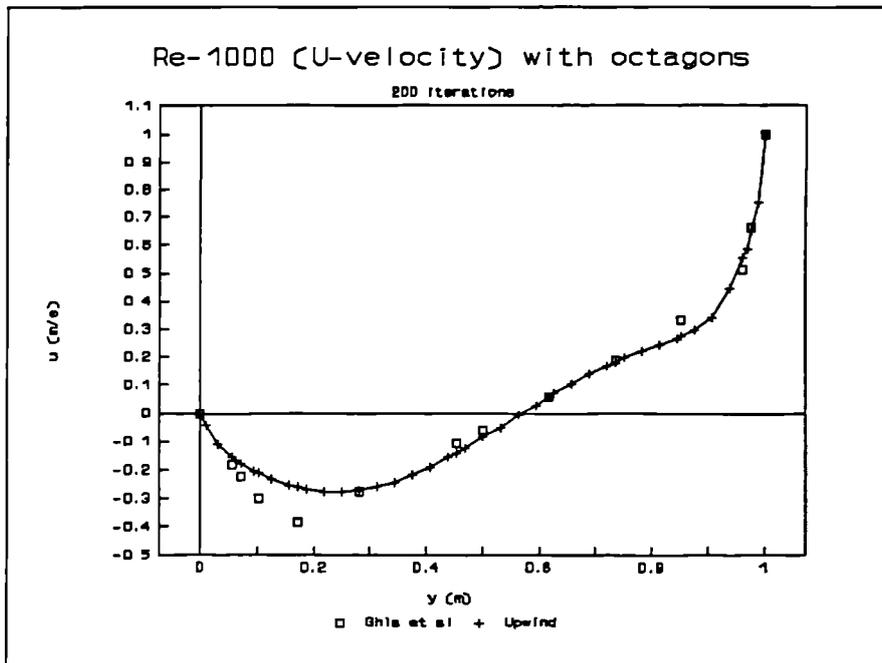


Figure 6.36a Octagon based mesh with upwind for Re-1000.

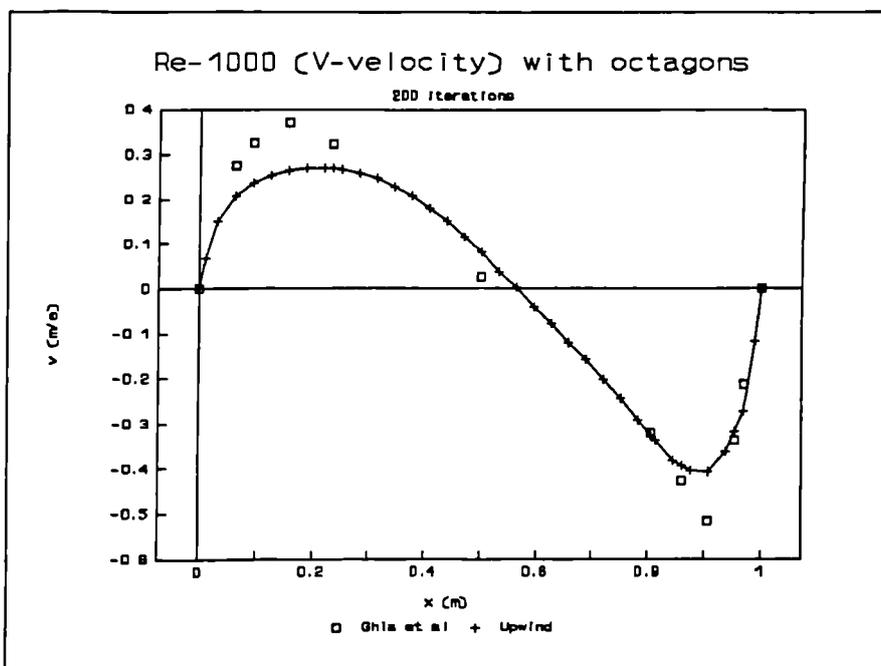


Figure 6.36b Octagon based mesh with upwind for Re-1000.

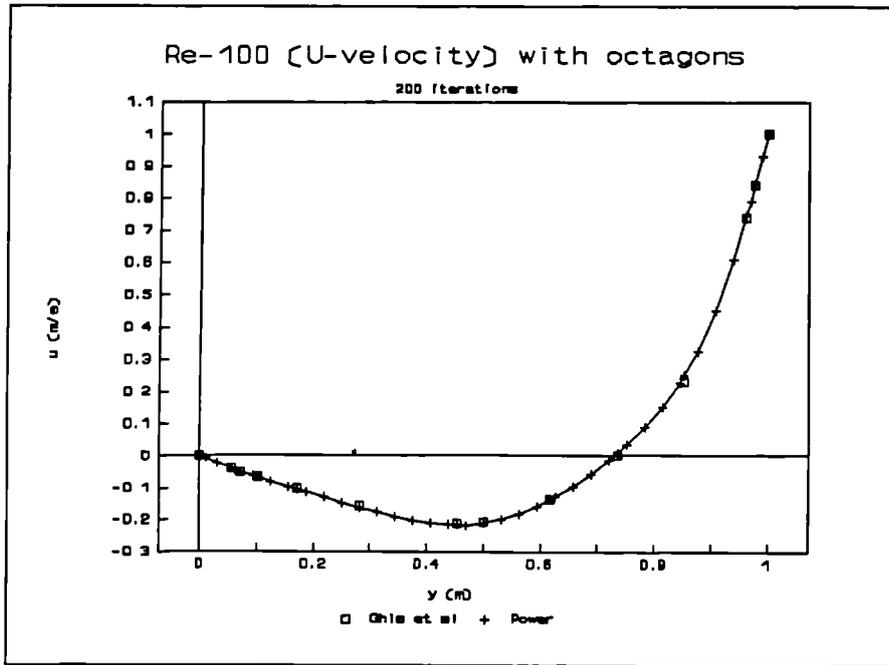


Figure 6.37a Octagon based mesh with power law for Re-100.

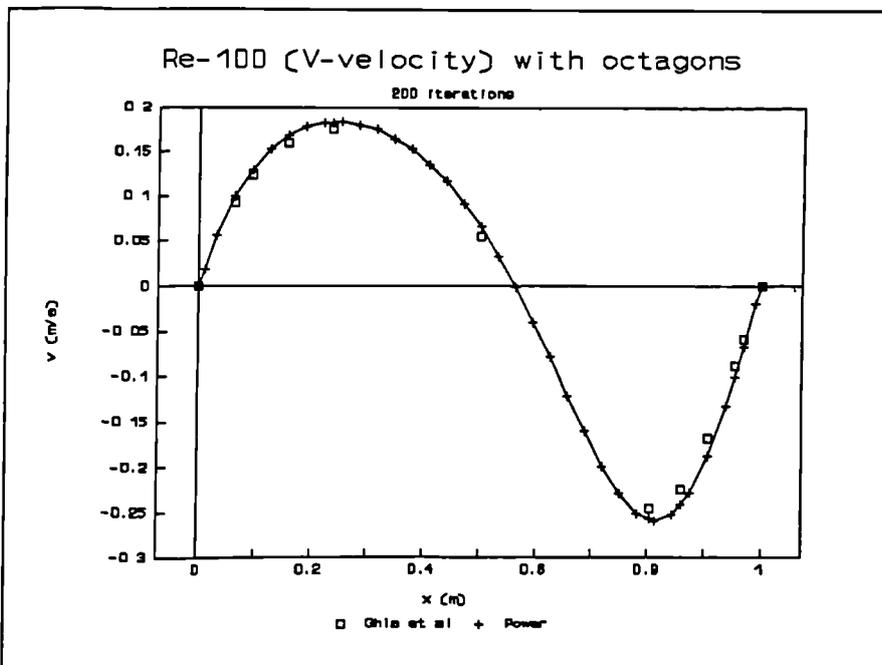


Figure 6.37b Octagon based mesh with power law for Re-100.

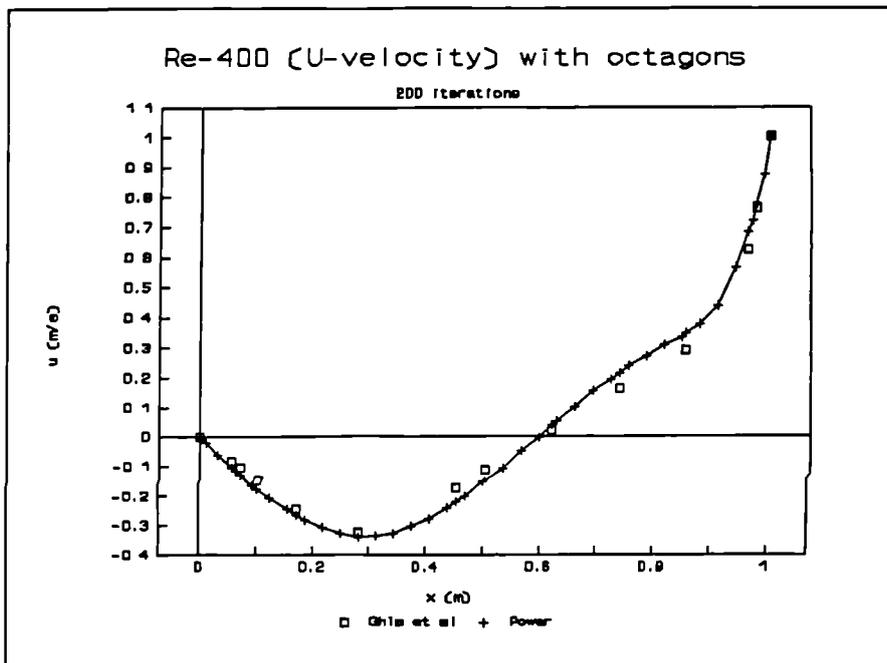


Figure 6.38a Octagon based mesh with power law for Re-400.

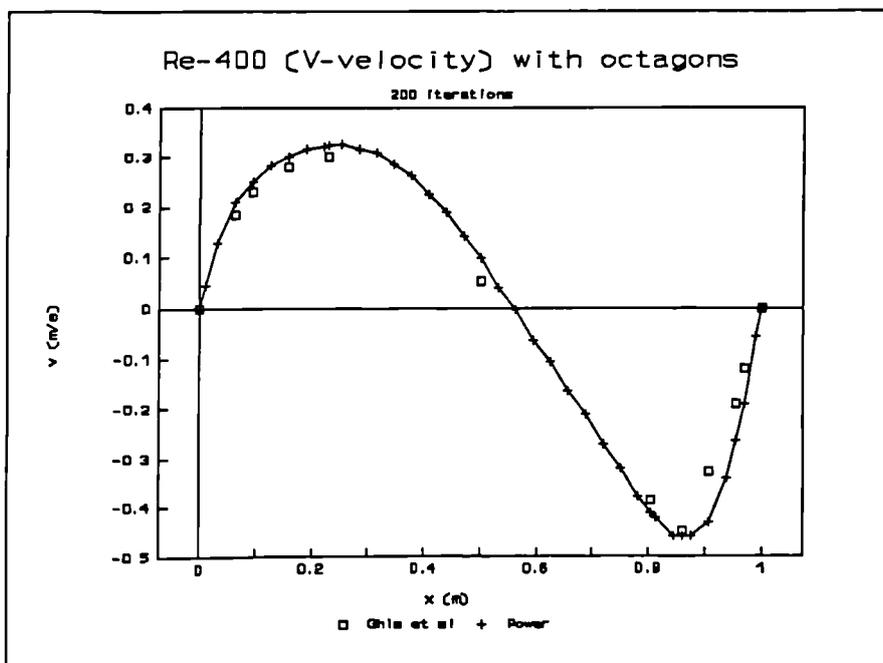


Figure 6.38b Octagon based mesh with power law for Re-400.

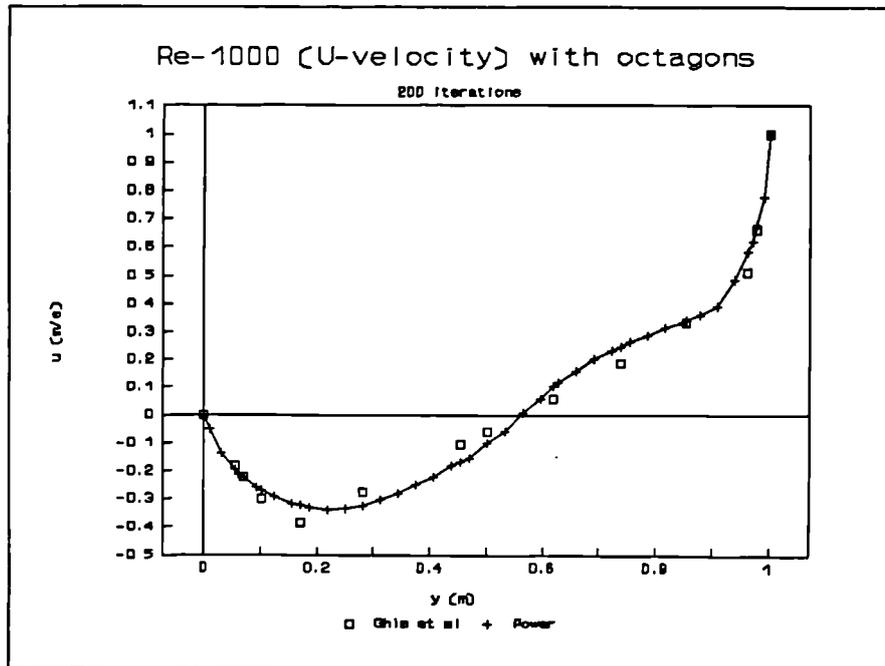


Figure 6.39a Octagon based mesh with power law for Re-1000.

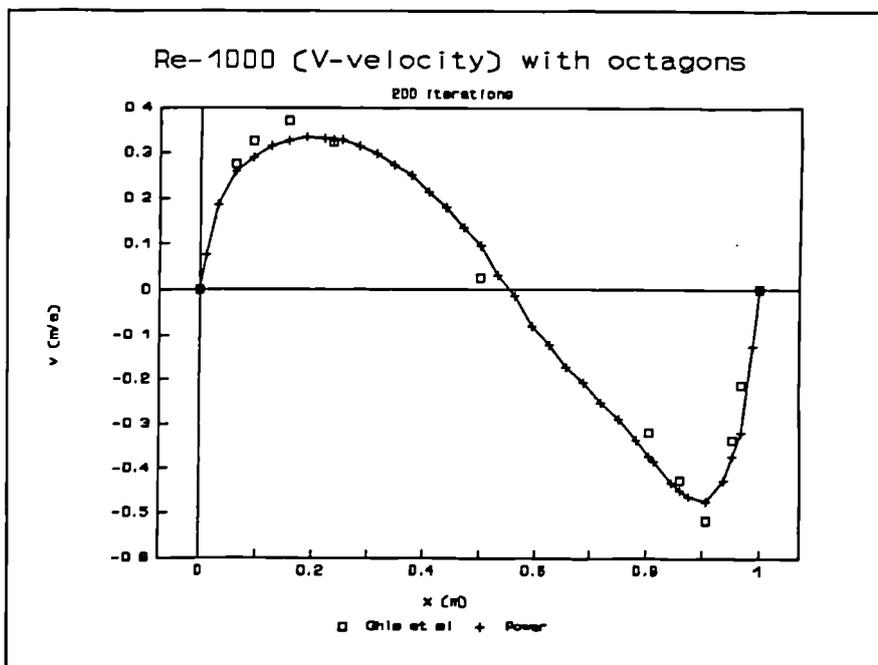


Figure 6.39b Octagon based mesh with power law for Re-1000.

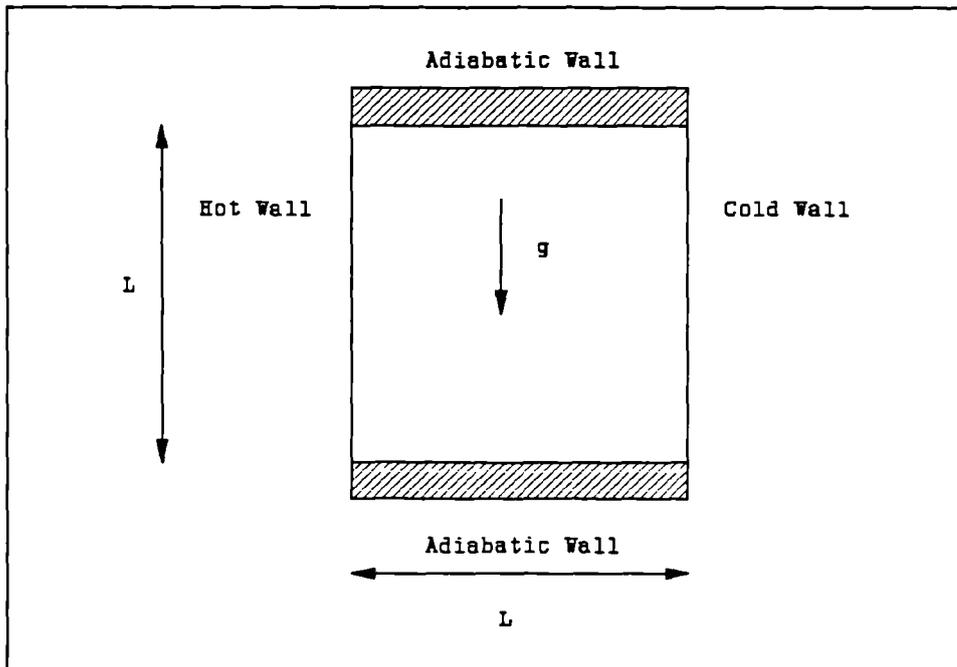


Figure 6.40 Natural convection driven cavity geometrical specification.

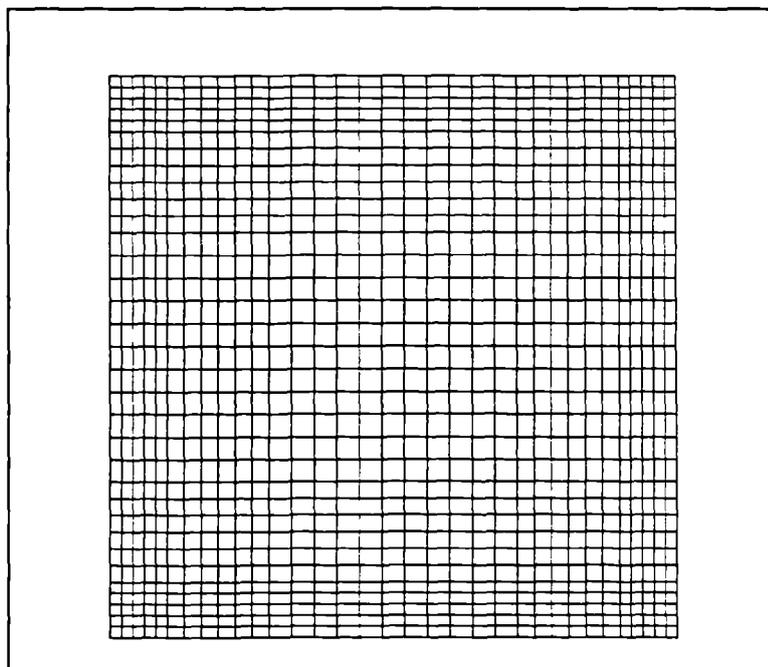


Figure 6.41 Non-uniform mesh.

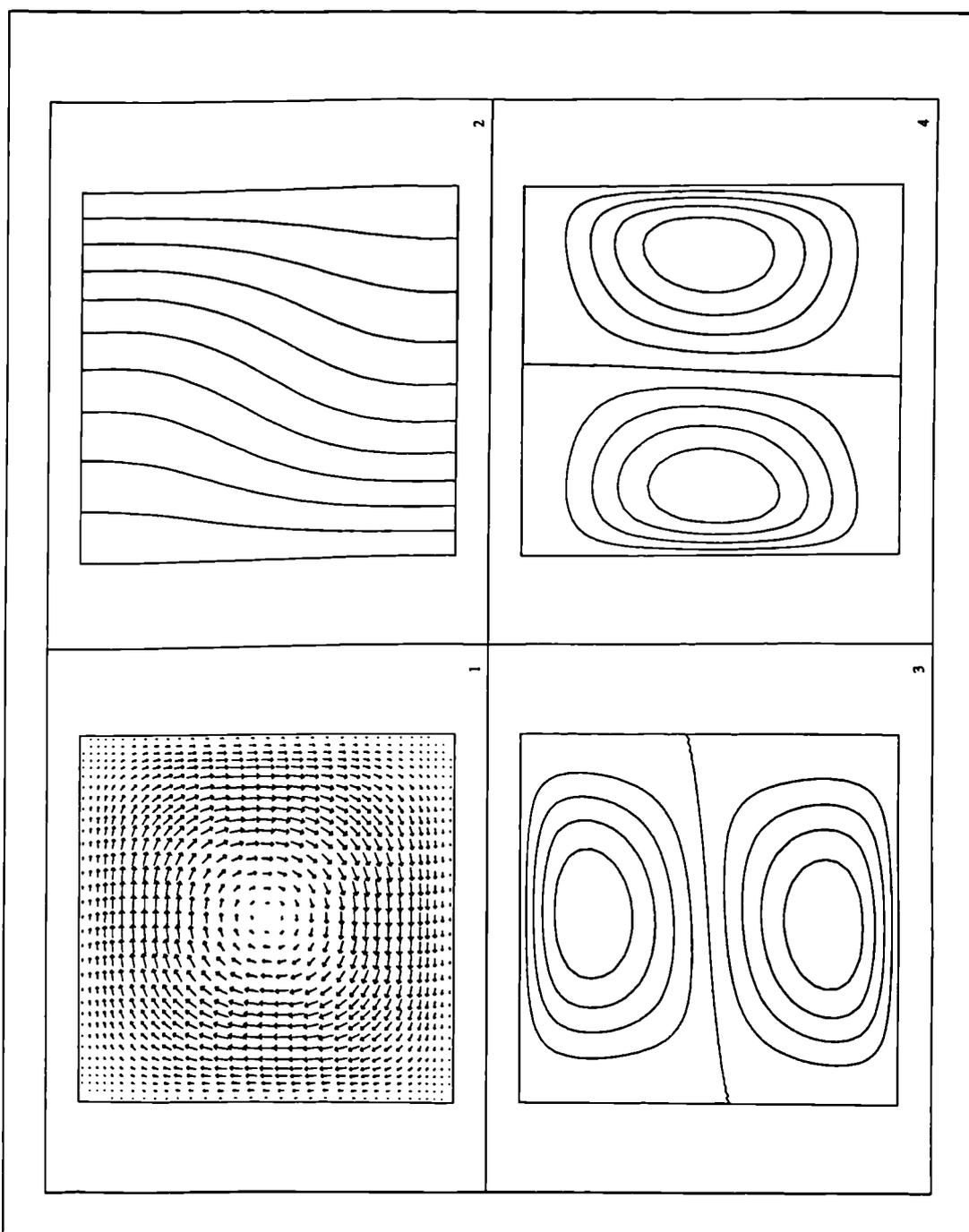


Figure 6.42 Non-uniform mesh results for $Ra-10^3$.

- 1) Velocity vector.
- 2) Temperature.
- 3) U-velocity.
- 4) V-velocity.

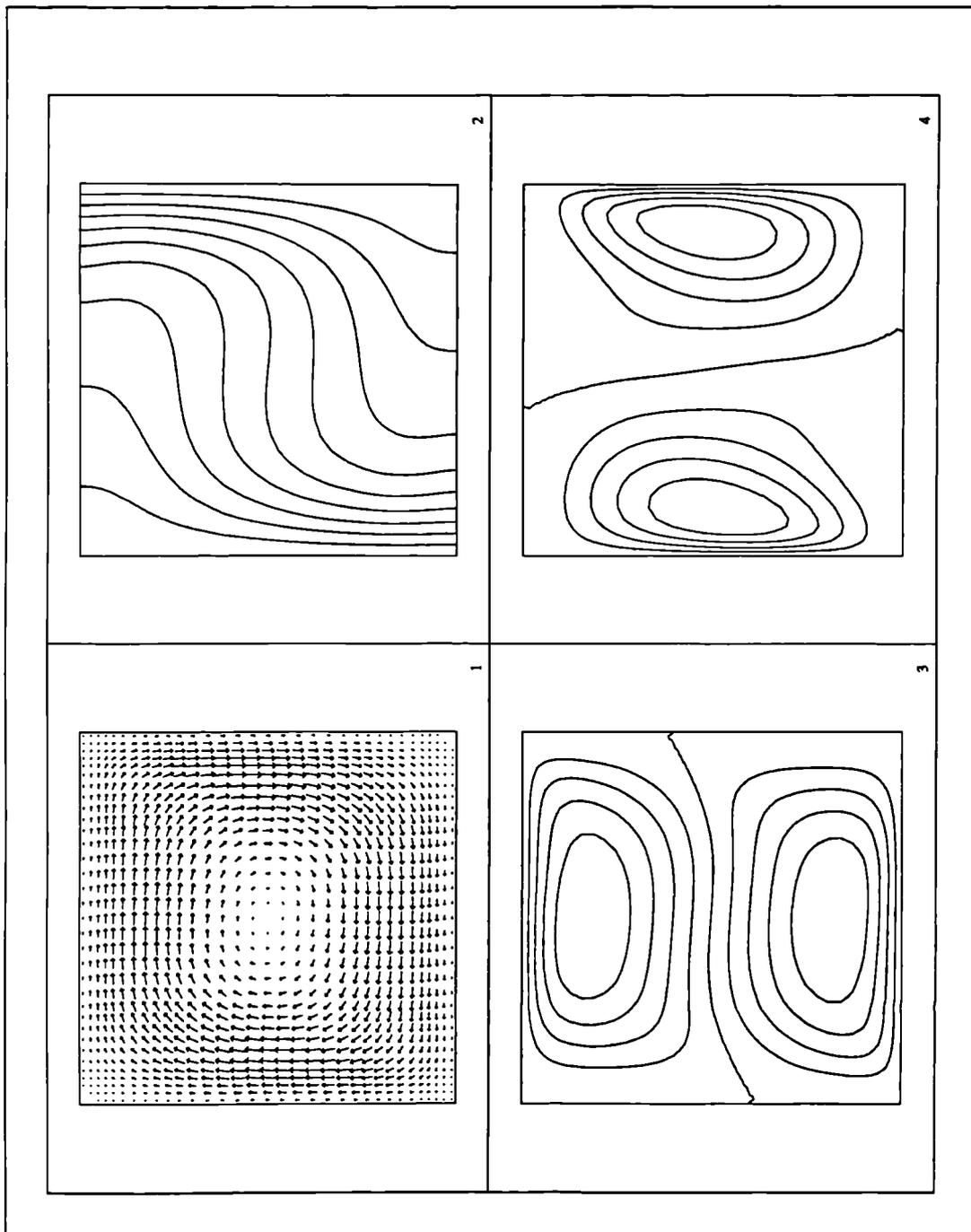


Figure 6.43 Non-uniform mesh results for $Ra-10^4$.

- 1) Velocity vector.
- 2) Temperature.
- 3) U-velocity.
- 4) V-velocity.

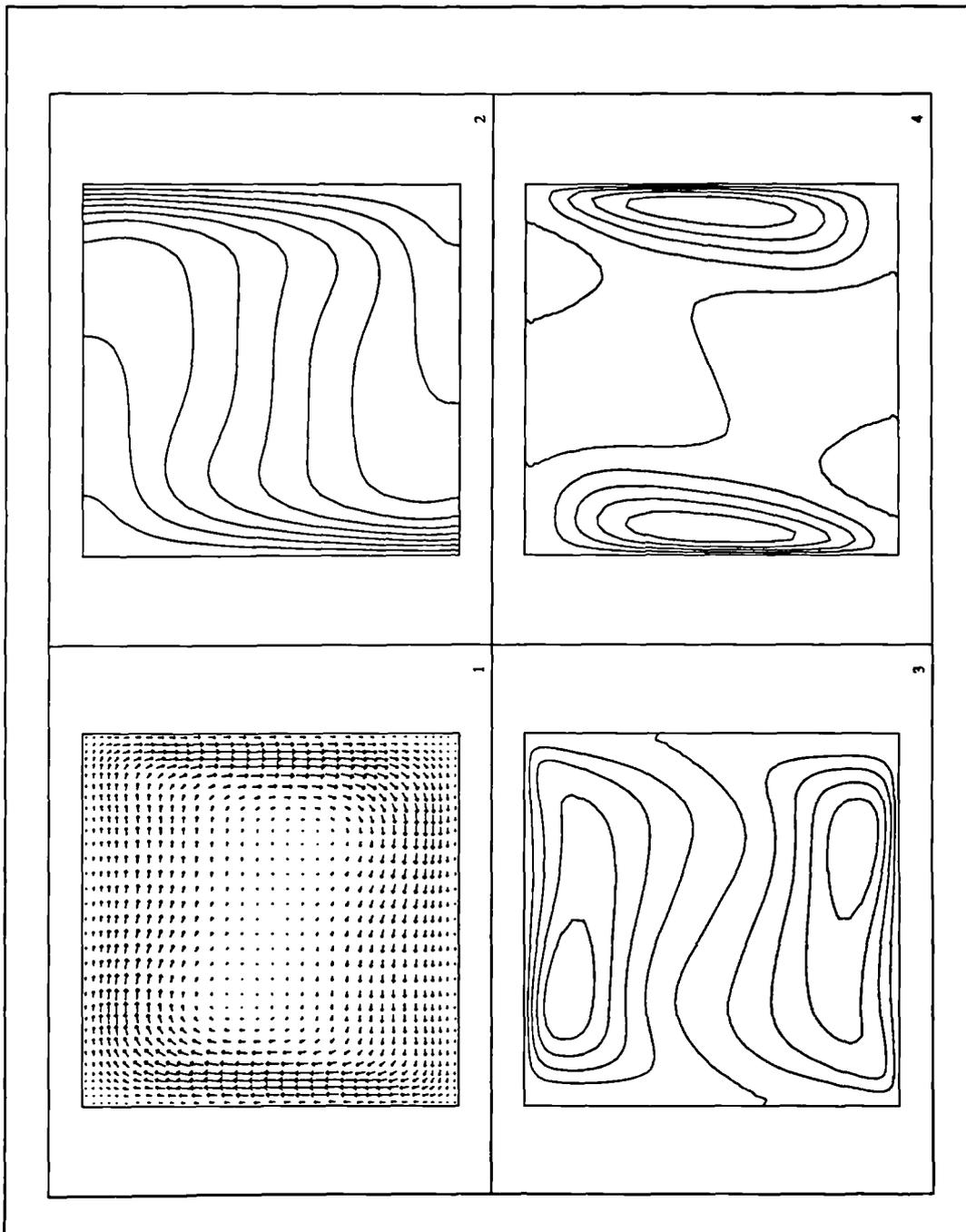


Figure 6.44 Non-uniform mesh results for $Ra-10^5$.

- 1) Velocity vector.
- 2) Temperature.
- 3) U-velocity.
- 4) V-velocity.

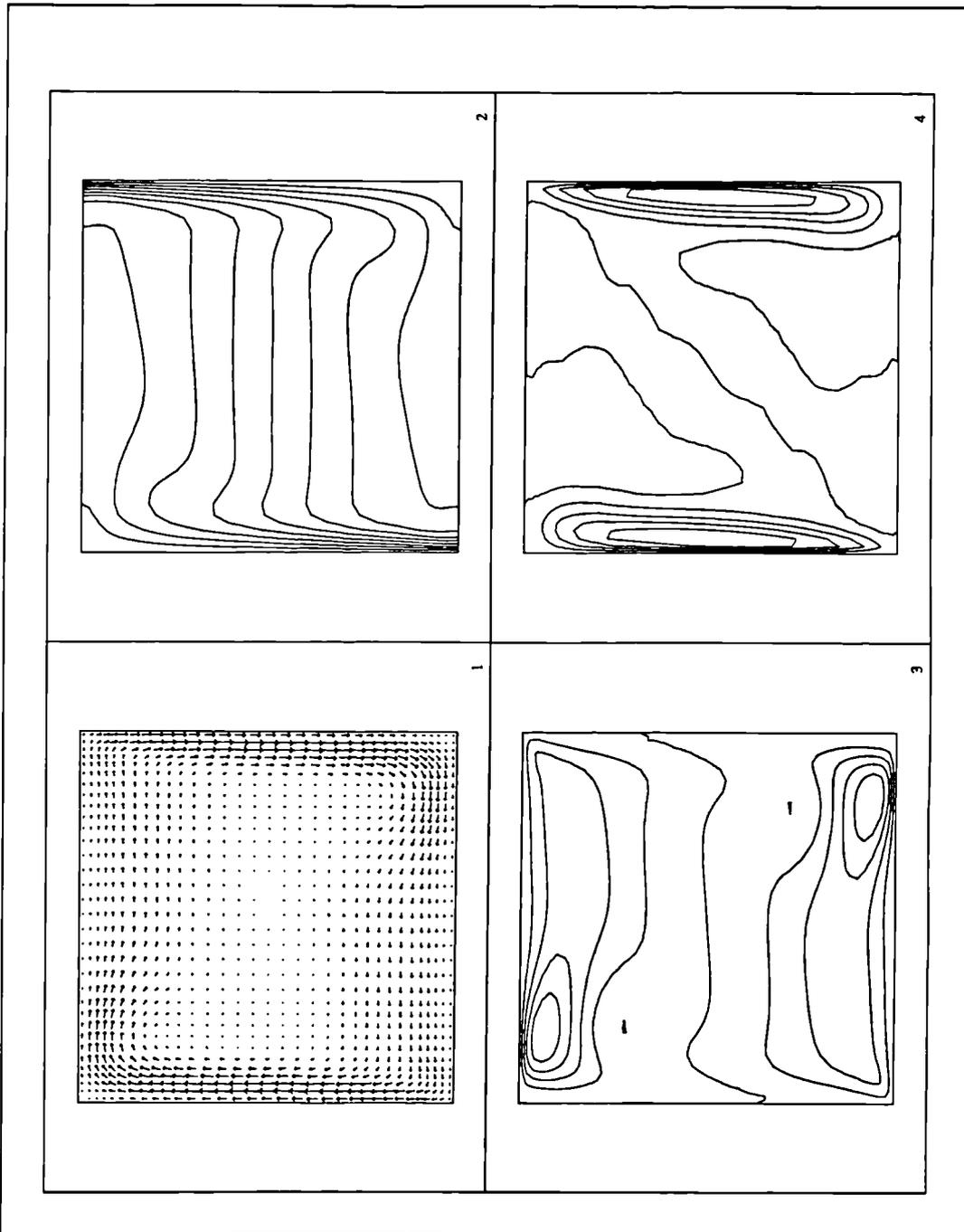


Figure 6.45 Non-uniform mesh results for $Ra-10^6$.

- 1) Velocity vector.
- 2) Temperature.
- 3) U-velocity.
- 4) V-velocity.

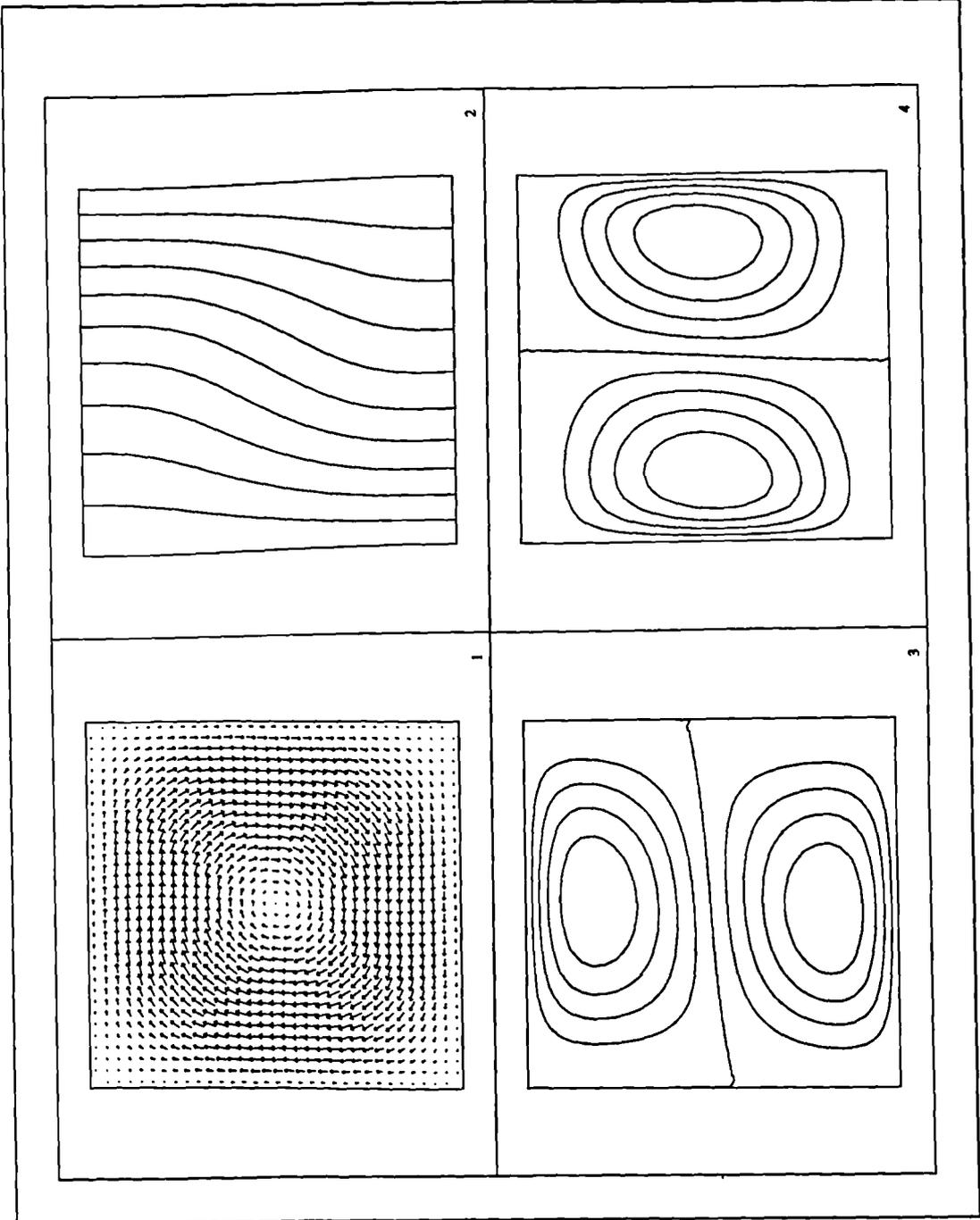


Figure 6.46 Uniform mesh results for $Ra-10^3$.

- 1) Velocity vector.
- 2) Temperature.
- 3) U-velocity.
- 4) V-velocity.

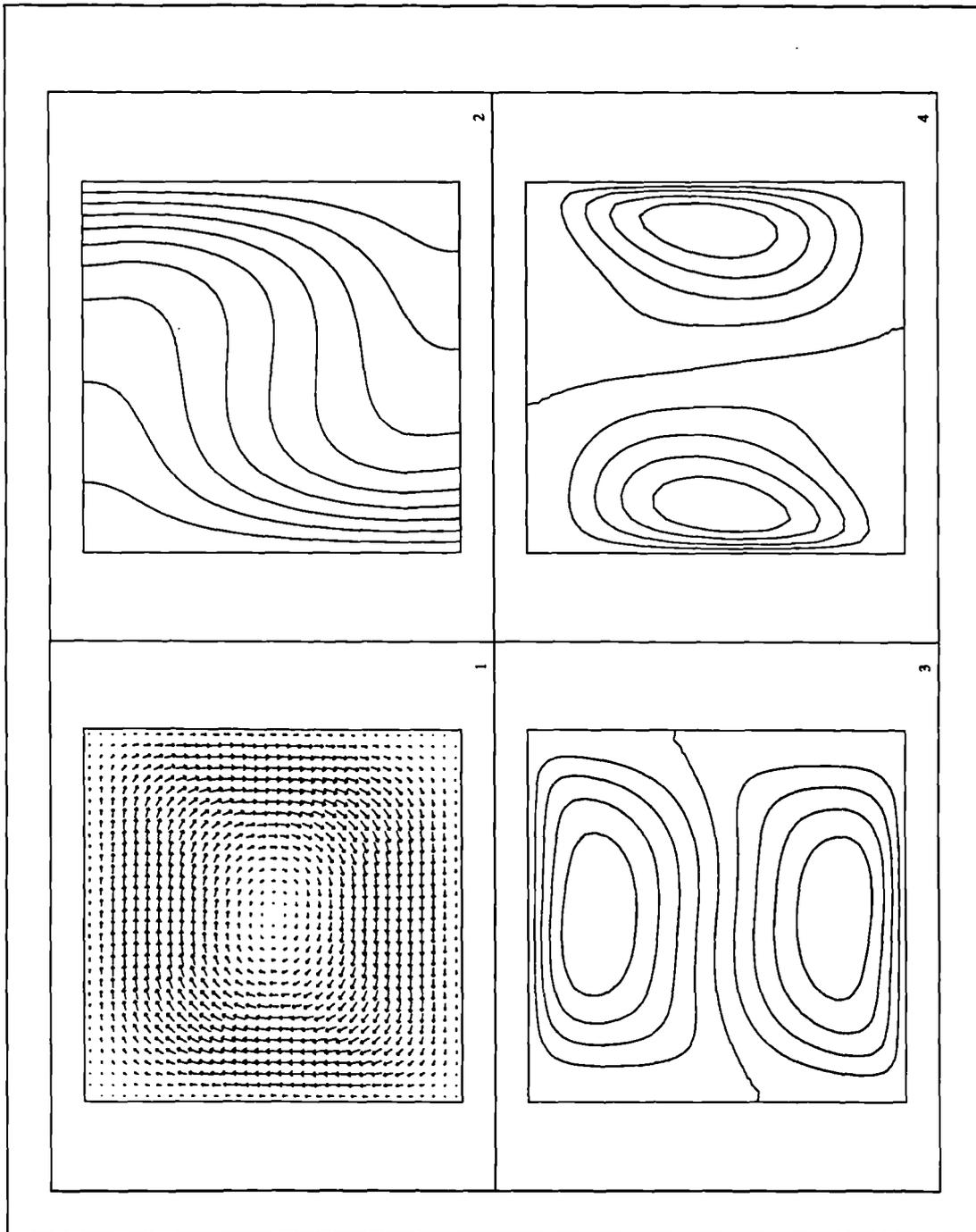


Figure 6.47 Uniform mesh results for $Ra=10^4$.

- 1) Velocity vector.
- 2) Temperature.
- 3) U-velocity.
- 4) V-velocity.

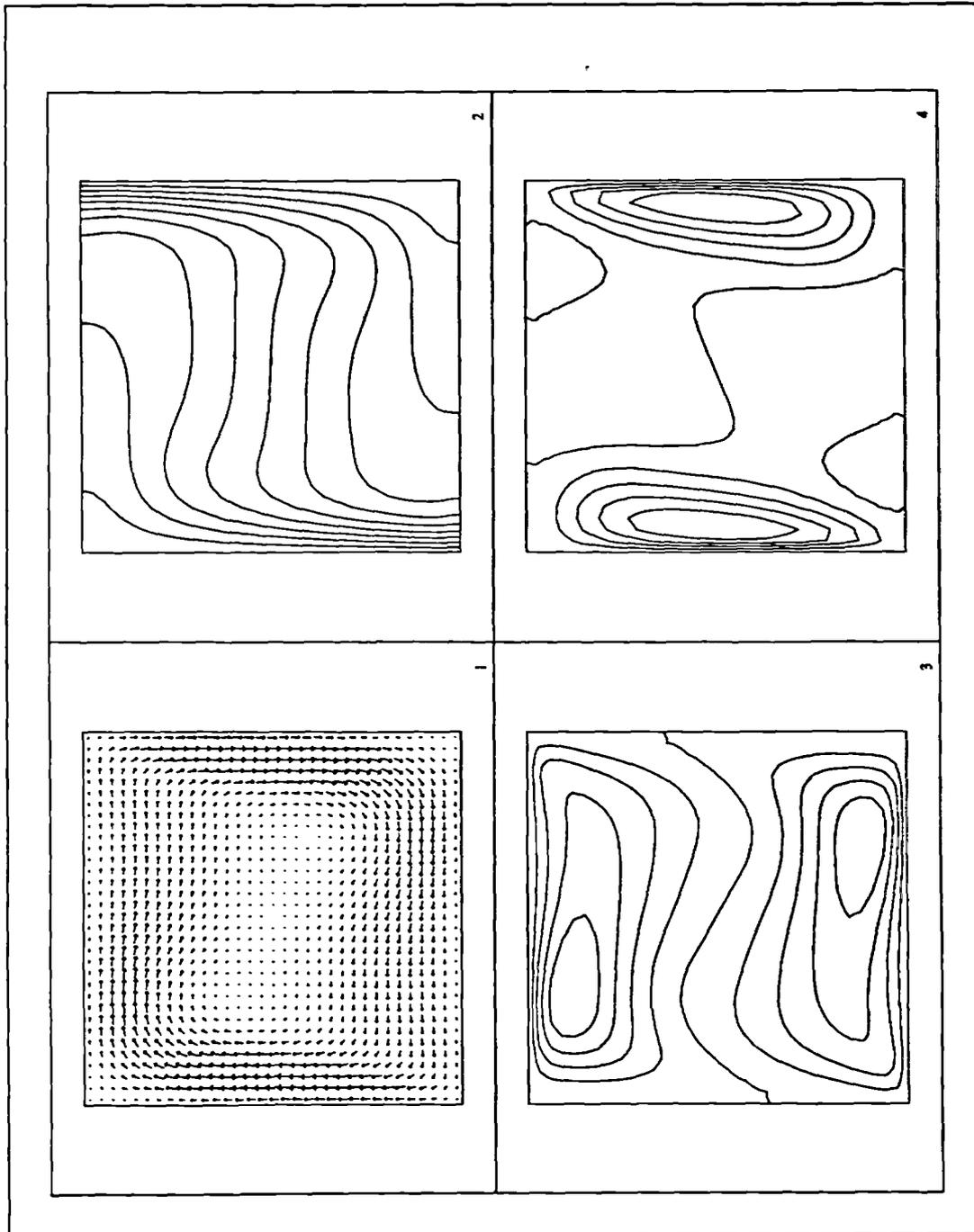


Figure 6.48 Uniform mesh results for $Ra-10^5$.

- 1) Velocity vector.
- 2) Temperature.
- 3) U-velocity.
- 4) V-velocity.

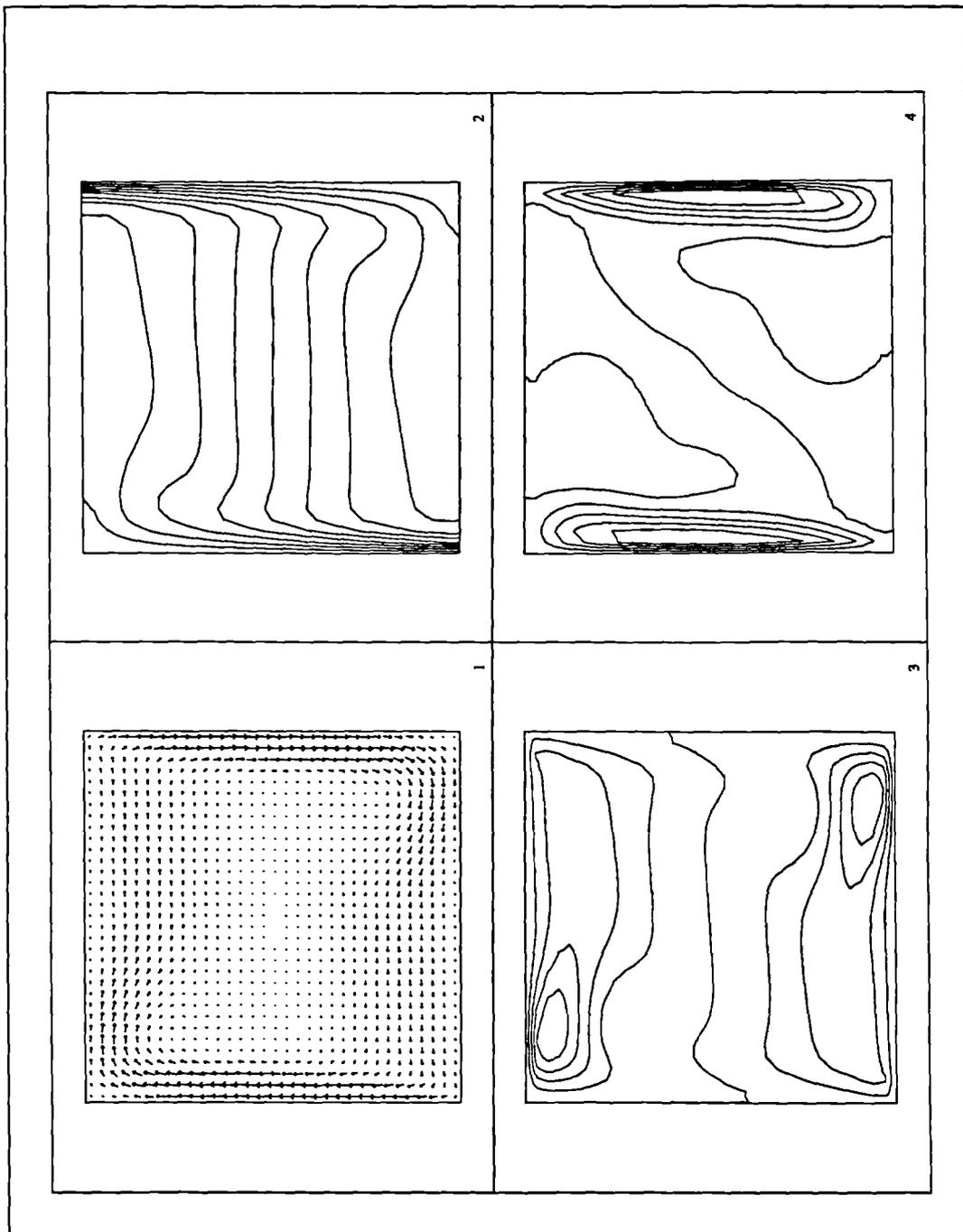


Figure 6.49 Uniform mesh results for $Ra-10^6$.

- 1) Velocity vector.
- 2) Temperature.
- 3) U-velocity.
- 4) V-velocity.

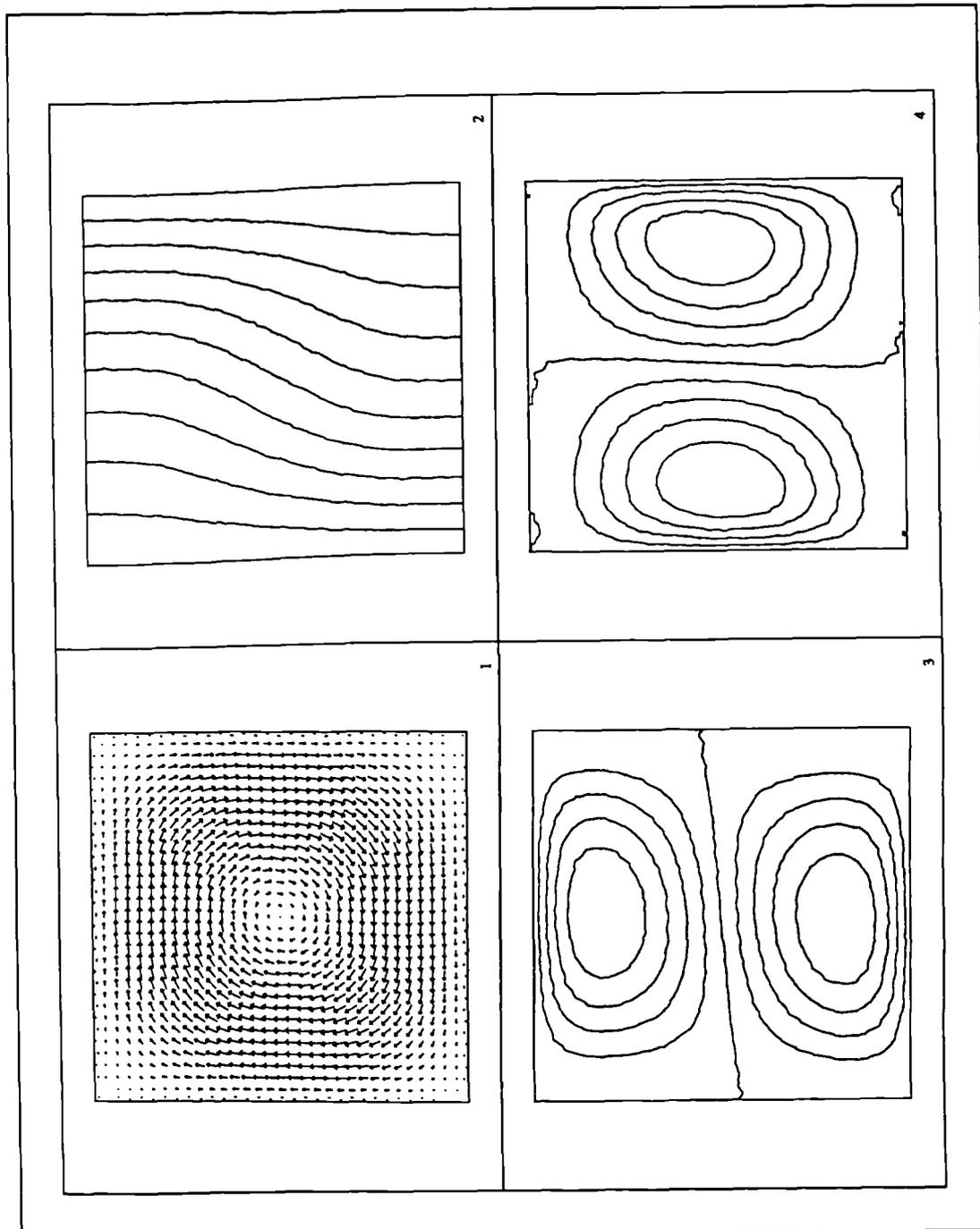


Figure 6.50 Octagon based mesh results for $Ra-10^3$.

- 1) Velocity vector.
- 2) Temperature.
- 3) U-velocity.
- 4) V-velocity.

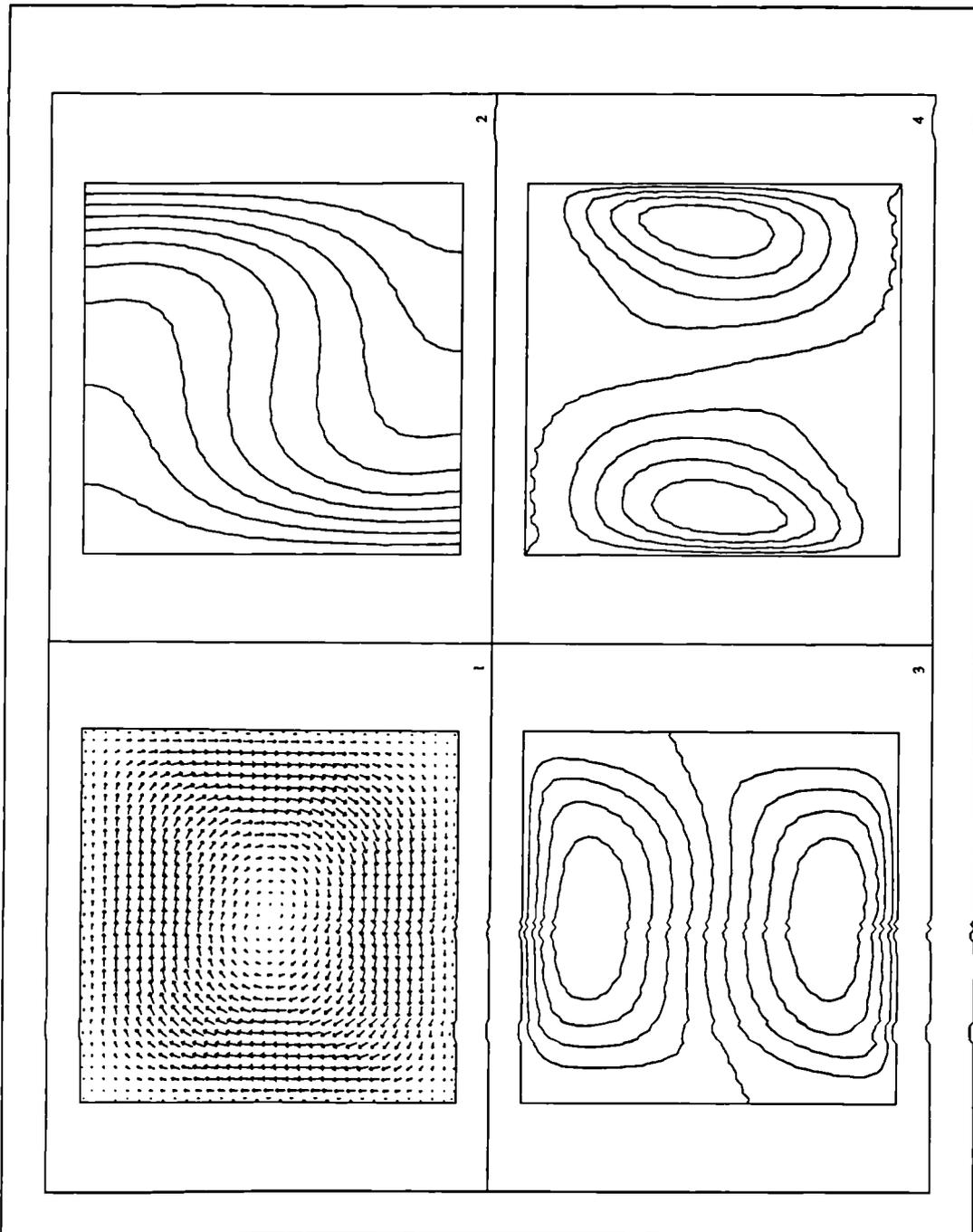


Figure 6.51 Octagon based mesh results for $Ra=10^4$.

- 1) Velocity vector.
- 2) Temperature.
- 3) U-velocity.
- 4) V-velocity.

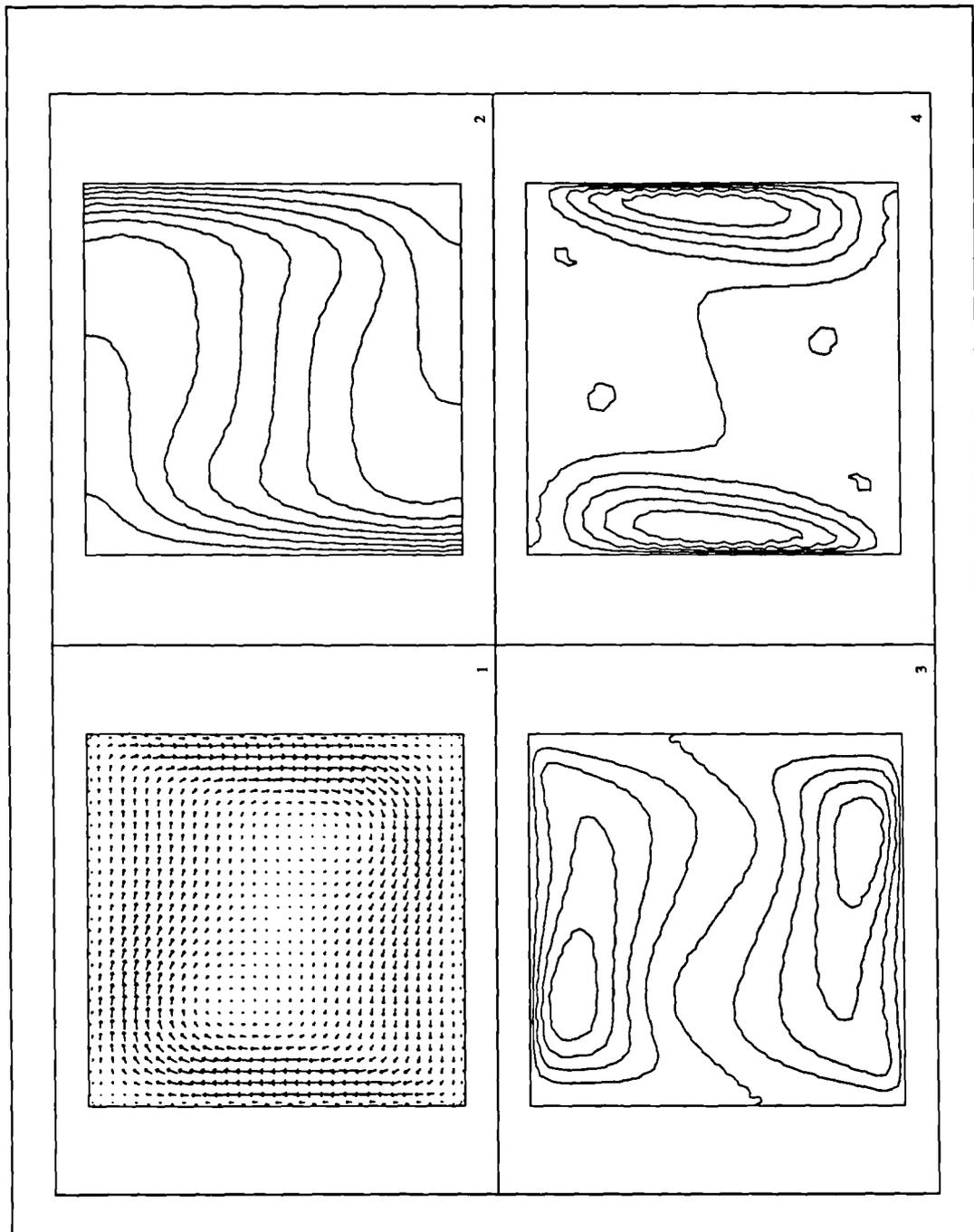


Figure 6.52 Octagon based mesh results for $Ra-10^5$.

- 1) Velocity vector.
- 2) Temperature.
- 3) U-velocity.
- 4) V-velocity.

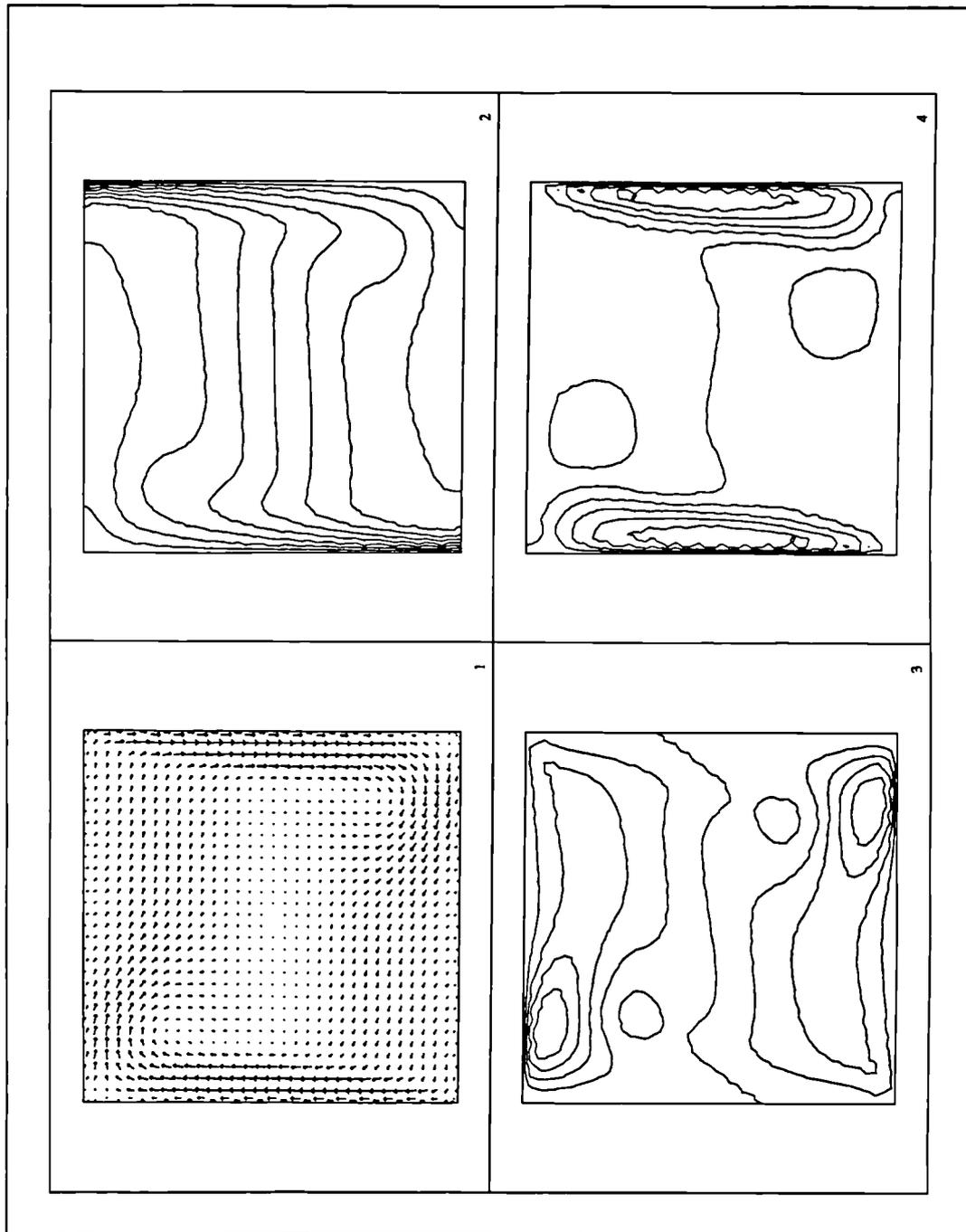


Figure 6.53 Octagon based mesh results for $Ra-10^6$.

- 1) Velocity vector.
- 2) Temperature.
- 3) U-velocity.
- 4) V-velocity.

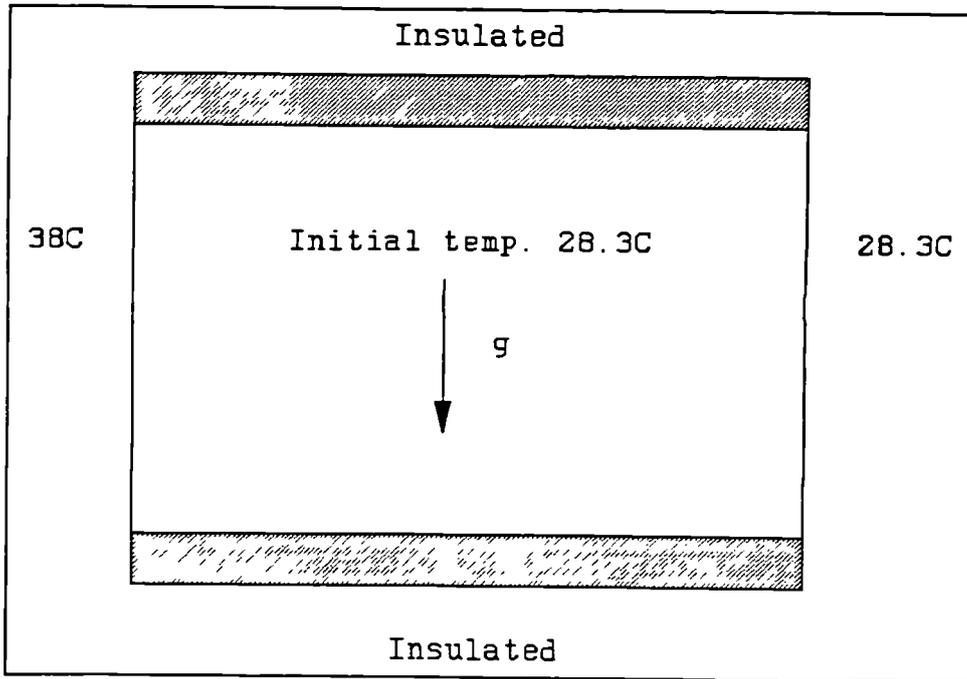


Figure 6.54 Geometrical specification for Gau and Viskanta.

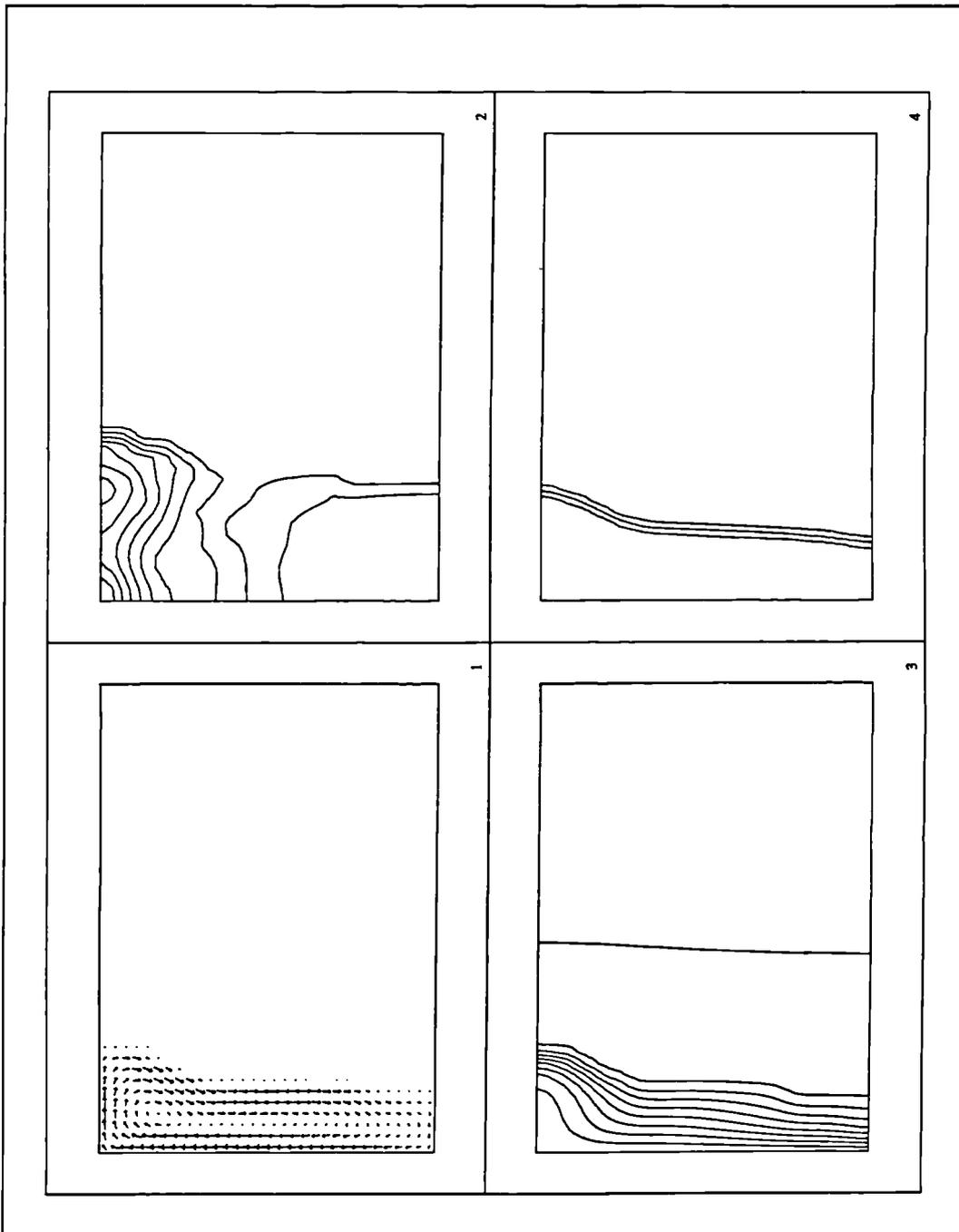


Figure 6.55 Melting of gallium at 3 minutes.

- 1) Velocity vector.
- 2) Pressure.
- 3) Temperature.
- 4) Liquid fraction.

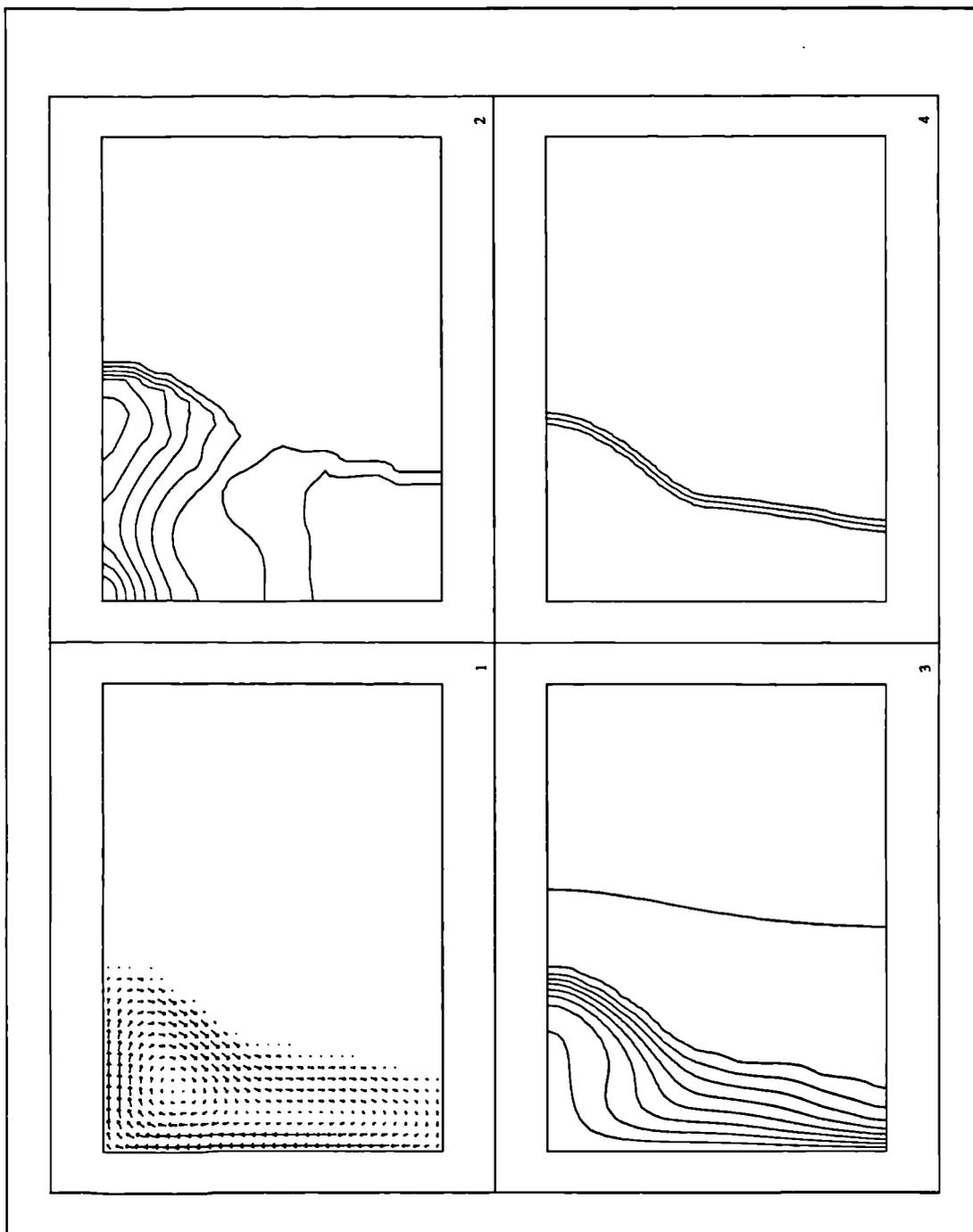
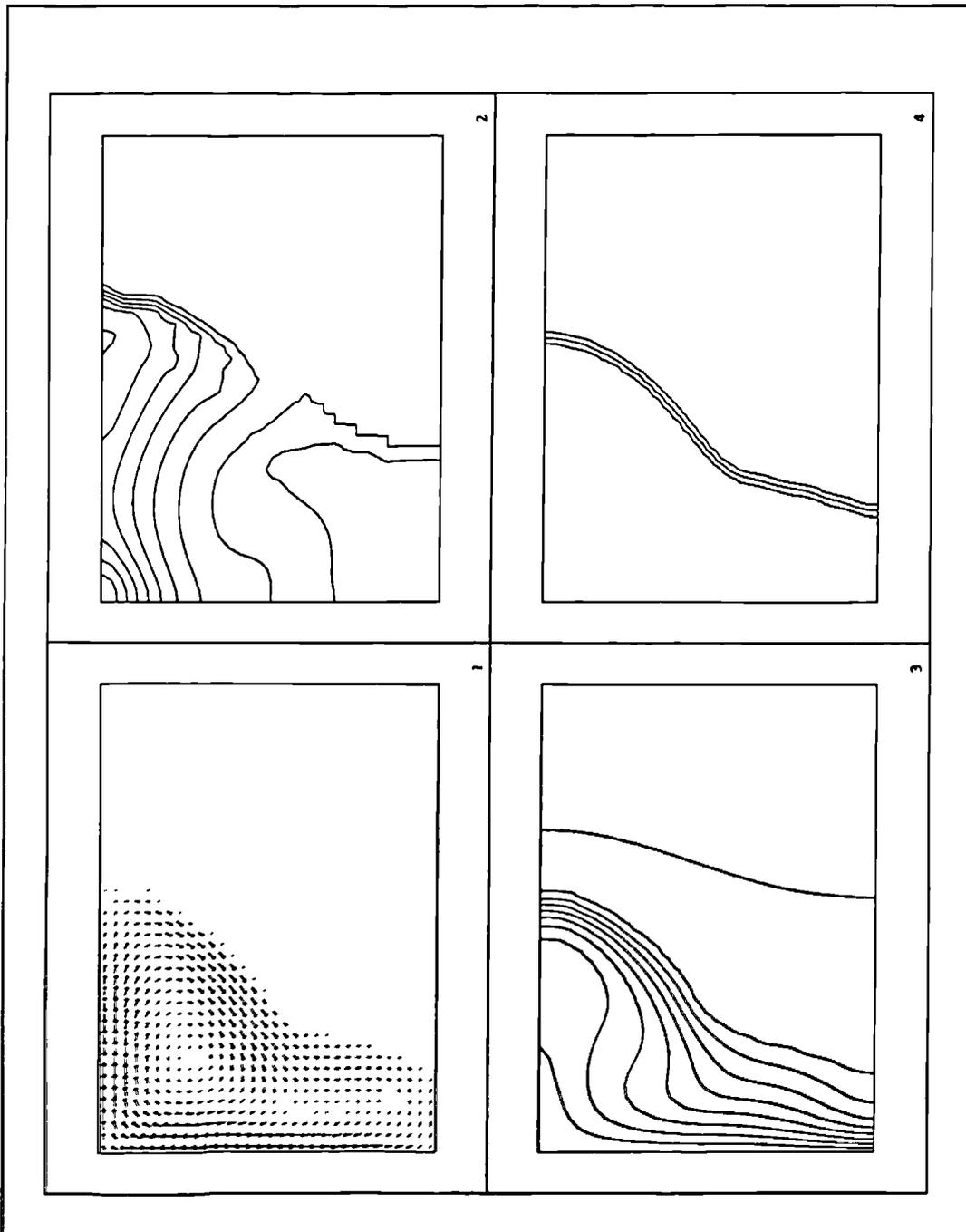


Figure 6.56 Melting of gallium at 6 minutes.

- 1) Velocity vector.
- 2) Pressure.
- 3) Temperature.
- 4) Liquid fraction.



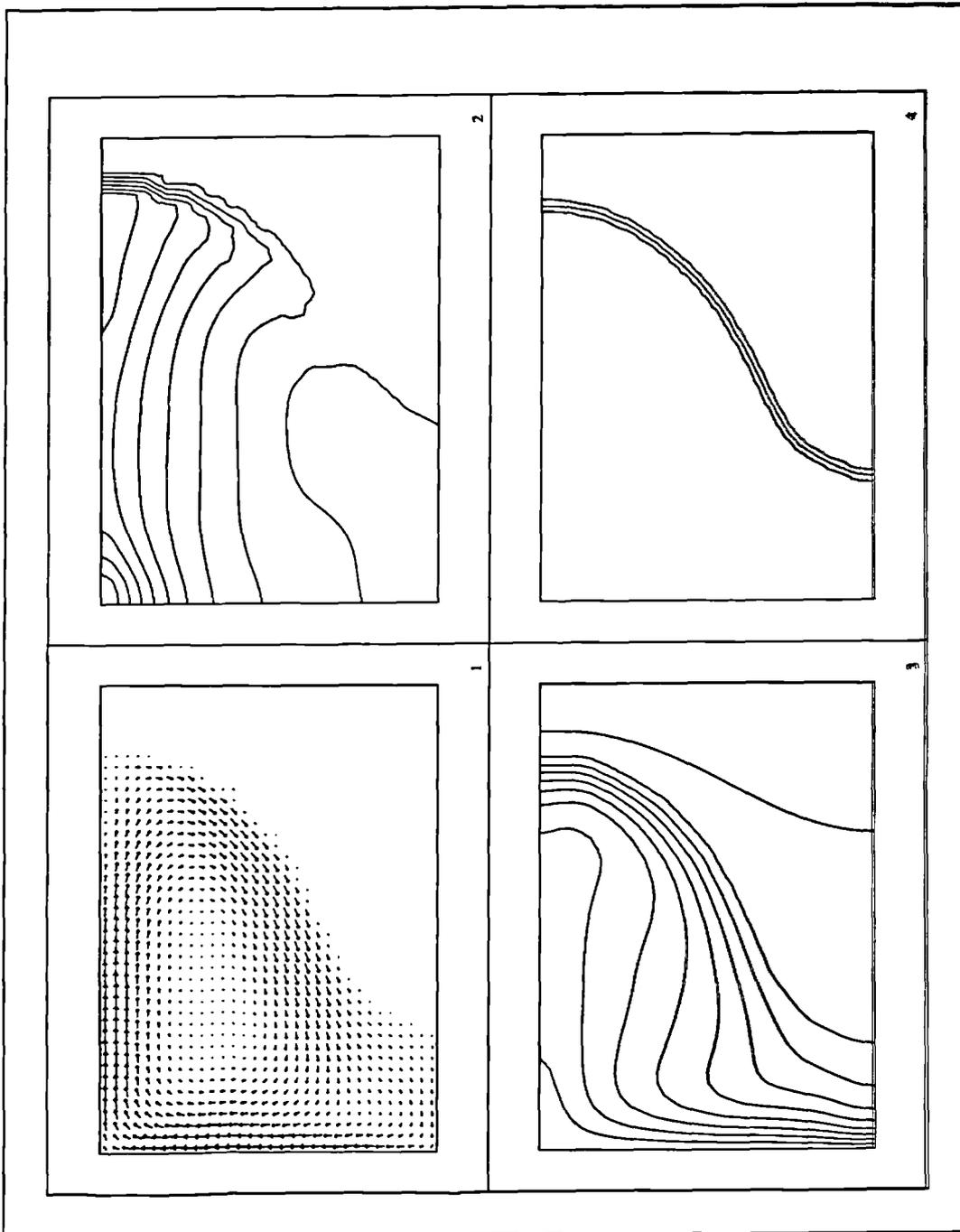


Figure 6.58 Melting of gallium at 19 minutes.

- 1) Velocity vector.
- 2) Pressure.
- 3) Temperature.
- 4) Liquid fraction.

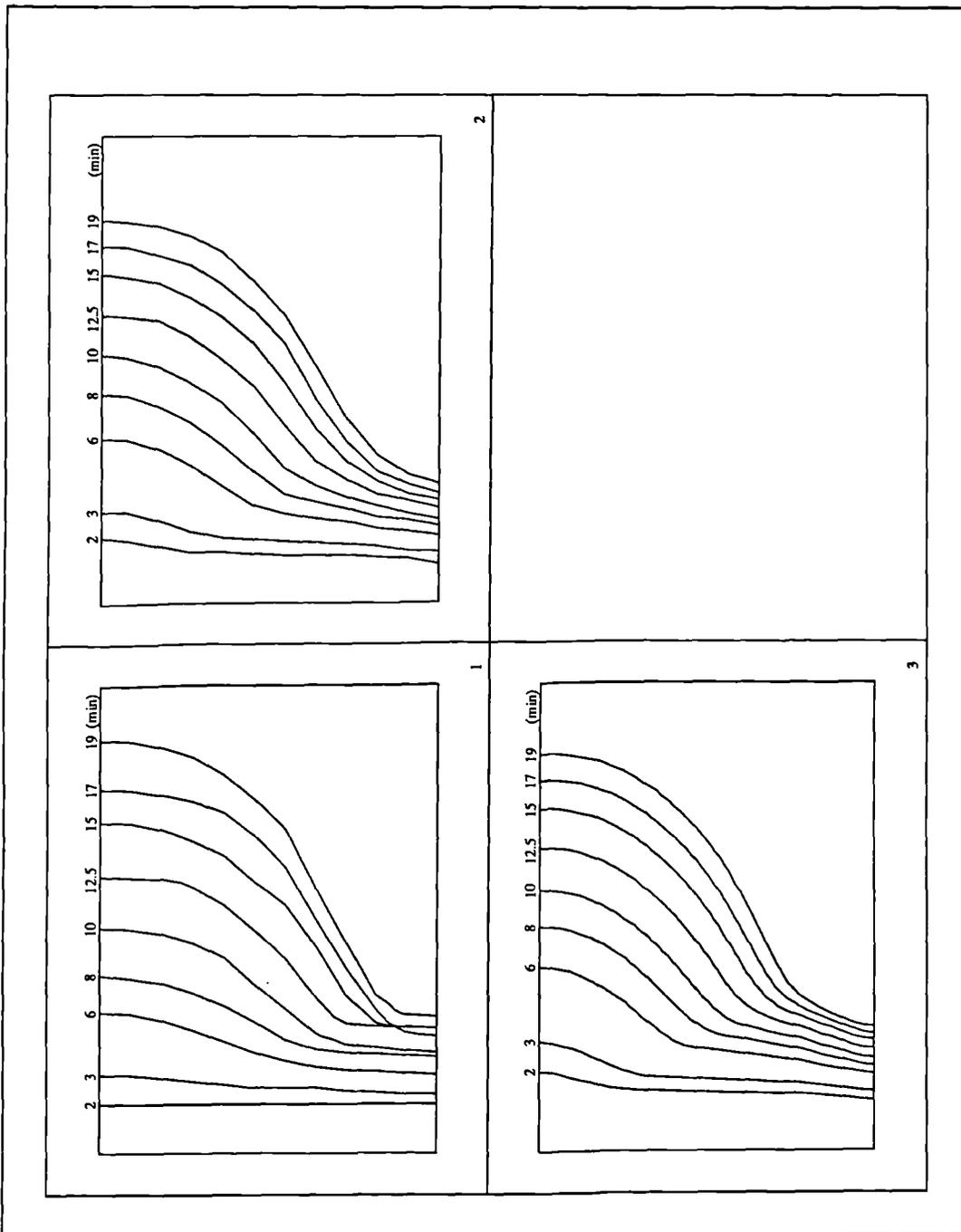


Figure 6.59 Melt fronts of the gallium problem.

- 1) Experimental results of Gau and Viskanta.
- 2) Numerical results of Brent et al.
- 3) Results from current study.

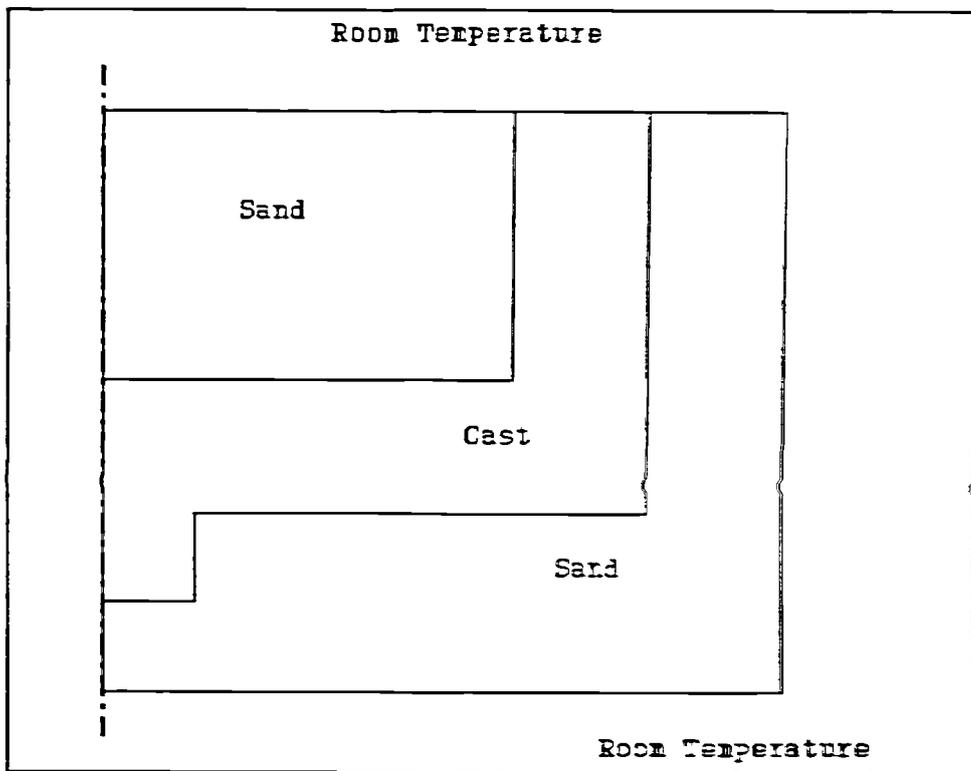


Figure 6.60 Geometrical specification of the integrated problem.

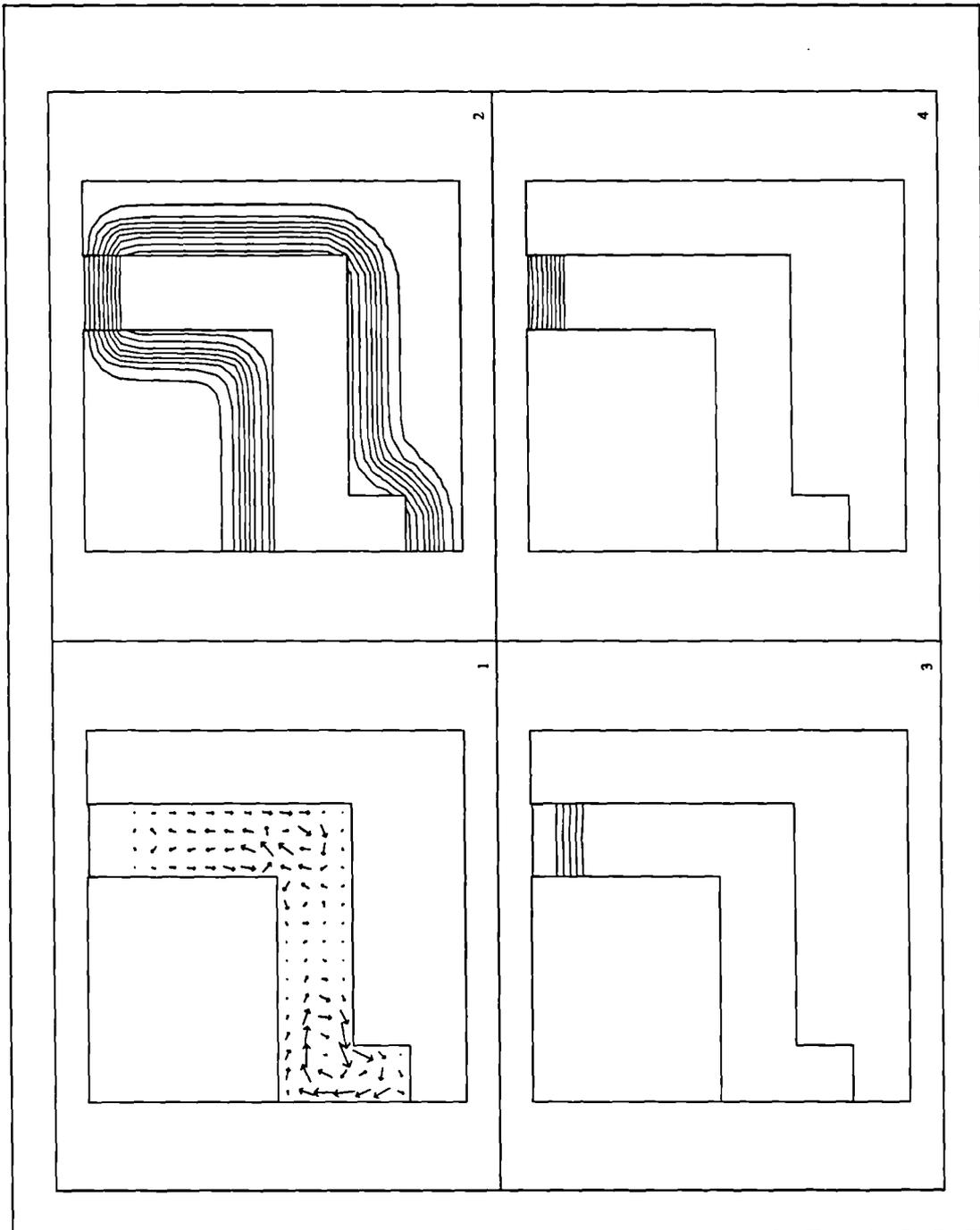


Figure 6.61 Integrated solution at 30 seconds.

- 1) Velocity vector.
- 2) Temperature.
- 3) Liquid fraction.
- 4) Residual stress.

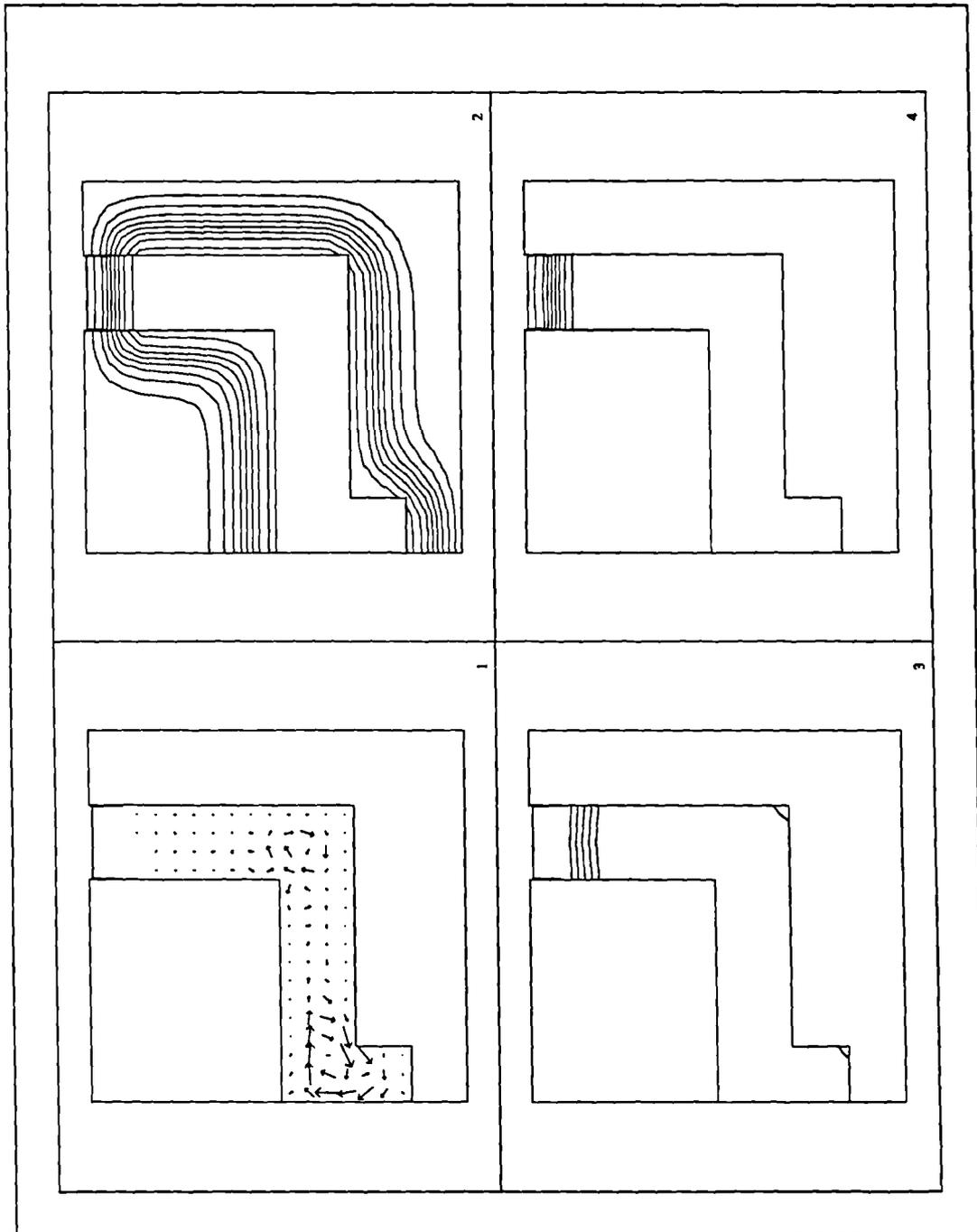
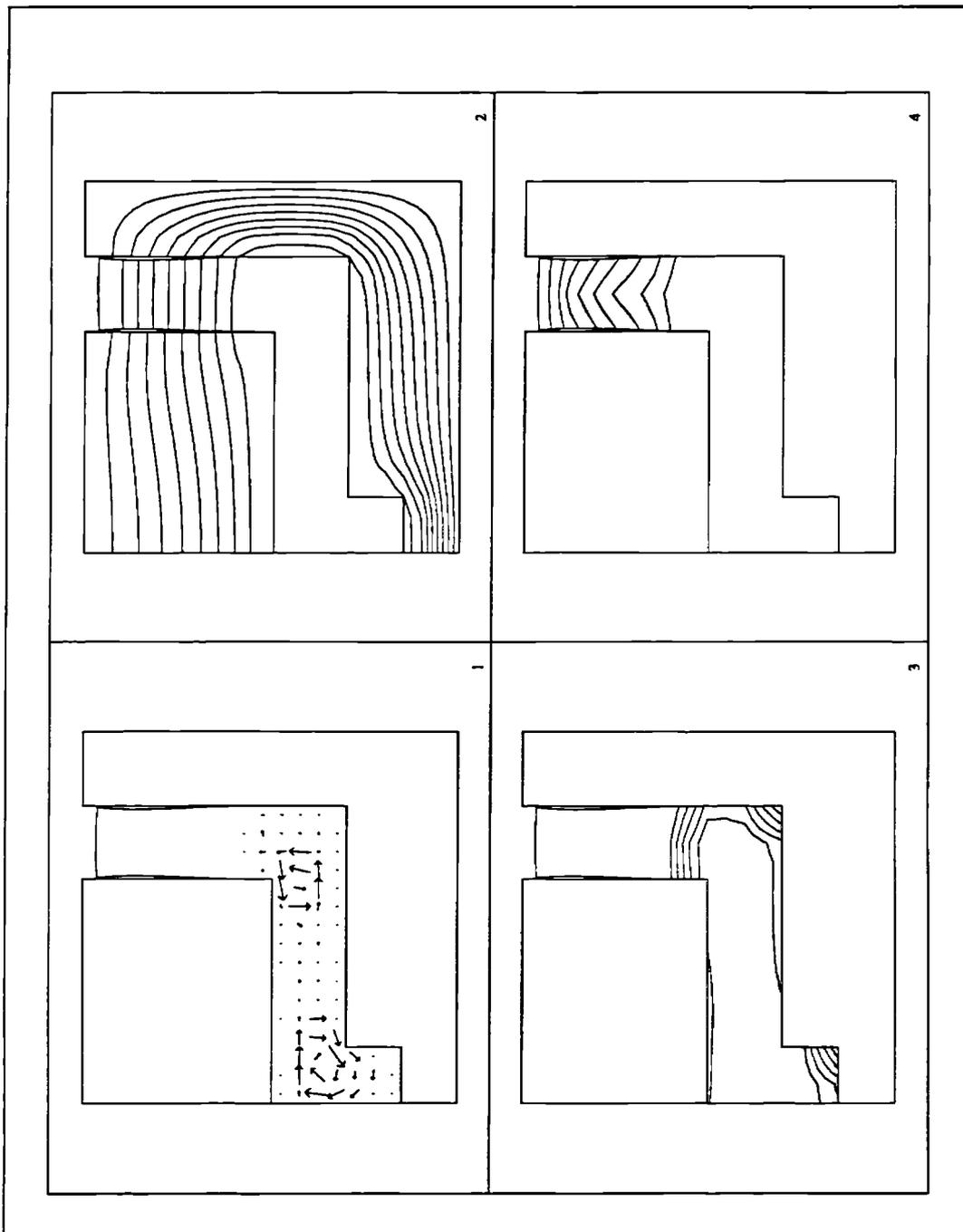


Figure 6.62 Integrated solution at 1 minute.

- 1) Velocity vector.
- 2) Temperature.
- 3) Liquid fraction.
- 4) Residual stress.



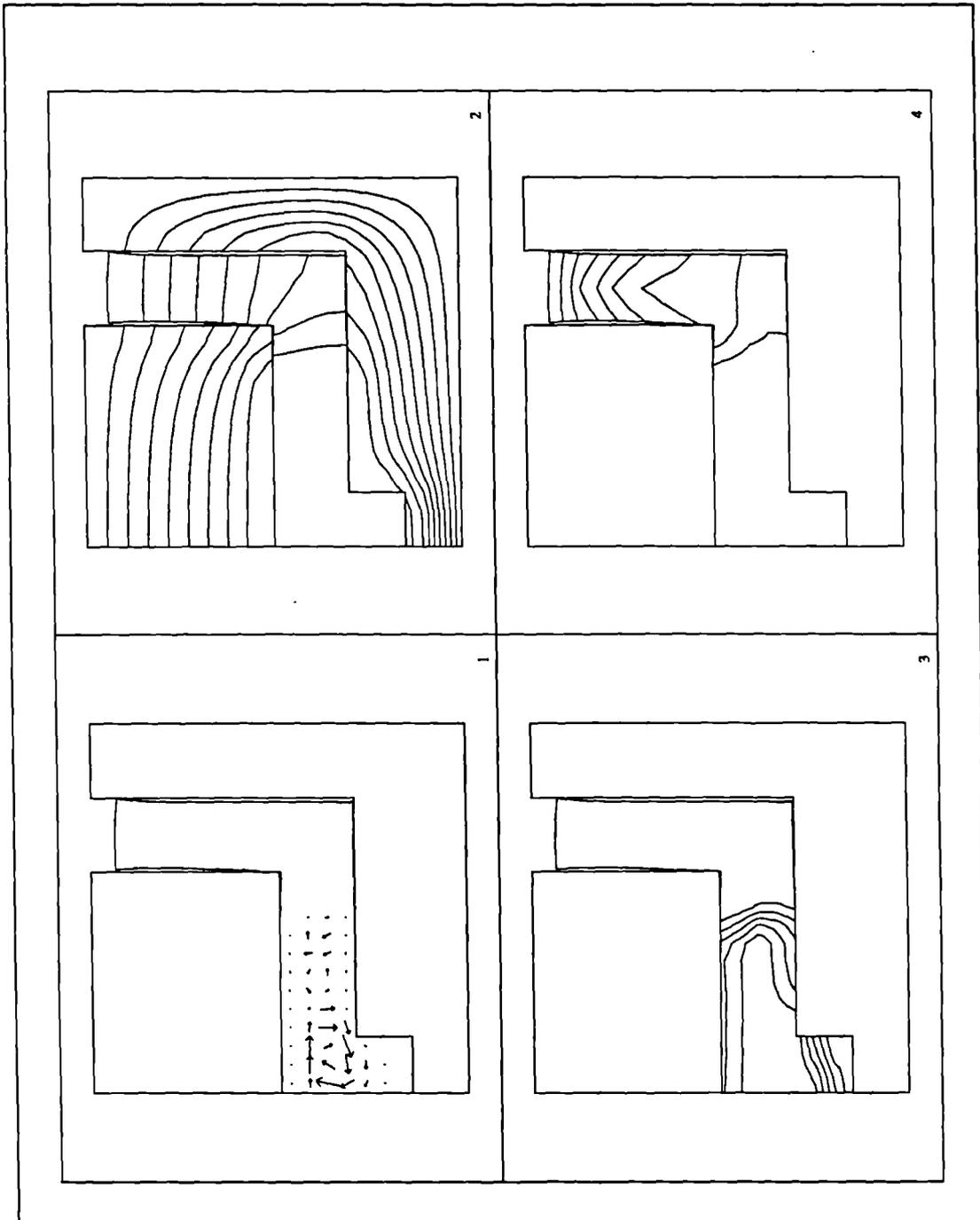


Figure 6.64 Integrated solution at 30 minutes.

- 1) Velocity vector.
- 2) Temperature.
- 3) Liquid fraction.
- 4) Residual stress.

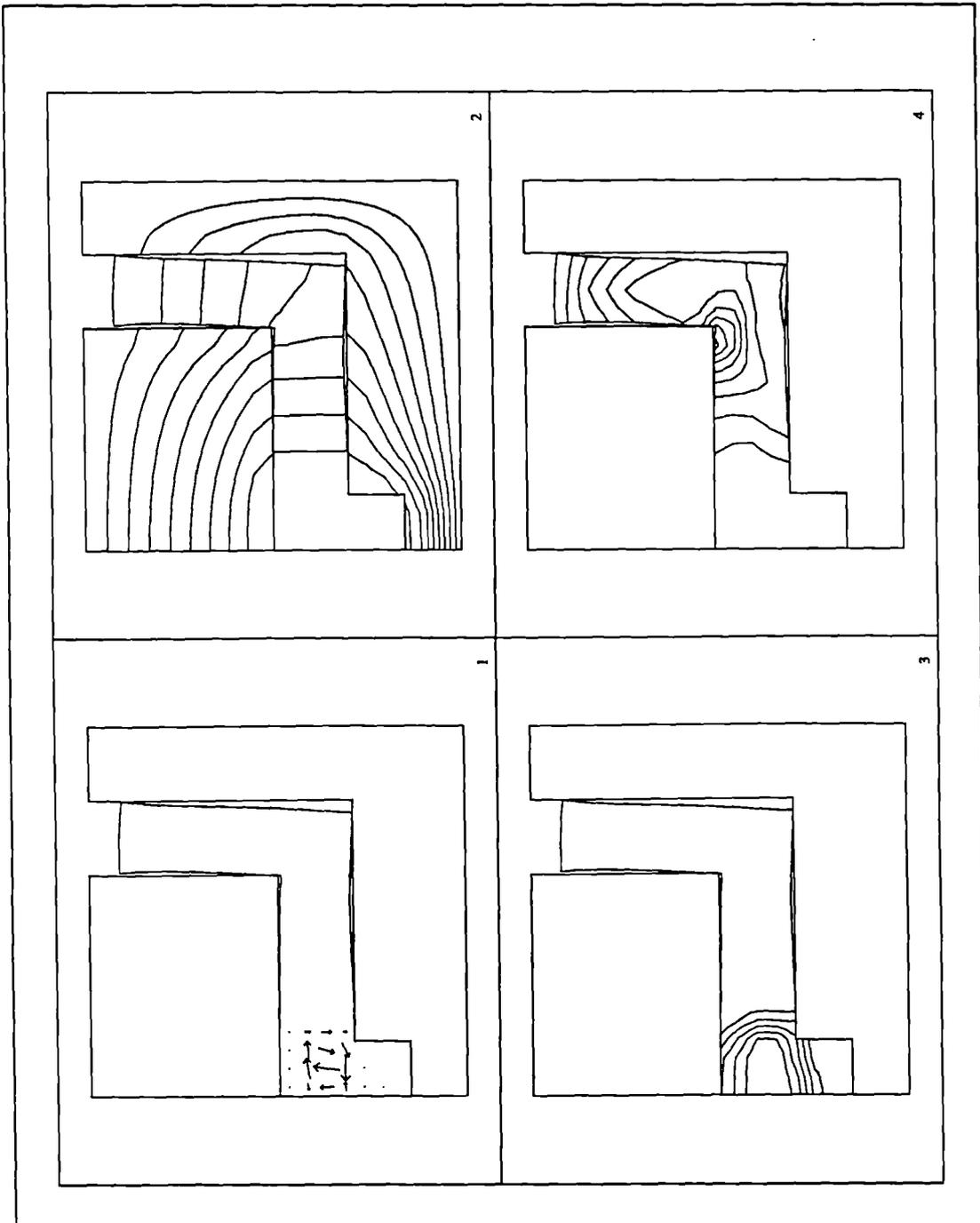


Figure 6.65 Integrated solution at 50 minutes.

- 1) Velocity vector.
- 2) Temperature.
- 3) Liquid fraction.
- 4) Residual stress.

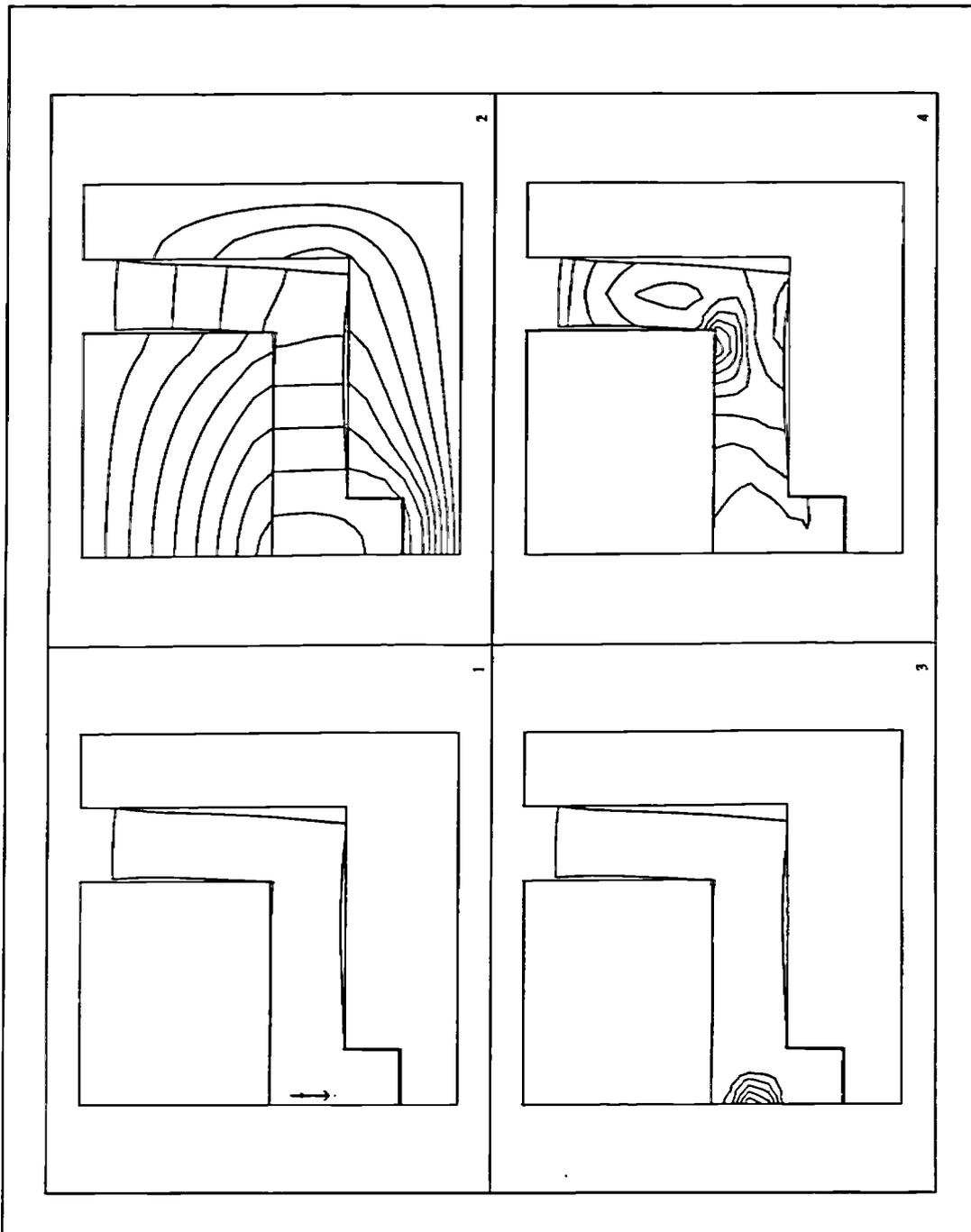


Figure 6.66 Integrated solution at 1 hour 5 minutes.

- 1) Velocity vector.
- 2) Temperature.
- 3) Liquid fraction.
- 4) Residual stress.

Chapter 7

7.0 Conclusions and Suggestions for further work	235
7.1 Conclusions	235
7.2 Suggestions for further work	238
7.2.1 Generating orthogonal high-order cells	238
7.2.2 Partial staggering to remove pressure oscillation	239
7.2.3 Cross-diffusion term for non-orthogonal meshes	241
7.2.4 Equation solvers	241
7.2.5 Extension to three-dimensions	242
Figures	243

7.0 Conclusions and Suggestions for further work

7.1 Conclusions

In this study, the examination and development of the Control Volume-Unstructured Mesh method (CV-UM) in context of solidification have been undertaken. It has been found that the two categories of the CV-UM approach, cell-centred and vertex-centred, can be deployed successfully in the modelling of casting processes. The vertex approach, generally known as the Control Volume-Finite Element (CV-FE) has already been used successfully in the fluid flow and heat transfer by workers such as Baliga and Patankar (1988), Schneider (1988) and Lonsdale and Webster (1989). For solidification, this study is the first to utilise the technique. The cell-centred approach on the other hand has previously been associated only with structured meshes [Patankar (1980)]. This has been extended to handle unstructured finite element meshes in this research and it is referred as the Irregular Control Volume method (ICV). The solidification algorithm used in the study is the one developed by Voller and Cross (1985), a method that is based on the cell-centred approach.

For solidification by conduction, the CV-FE solution has been found relatively insensitive to the orientation or quality of the mesh. Computational cost is 2-3 times more expensive than that of the structured one. And special considerations are needed for the cast/mould interface. For the ICV without the cross-diffusion term, the solution can be influenced by the orientation of the mesh unless it is reasonably orthogonal. With the simpler formulation the computational overhead is only, for quadrilateral, 20-

30% and for triangle, 30-80% more expensive than that of the structured method. This is a more cost effective unstructured mesh approach than the CV-FE method, specially when fluid flow calculation is included. For fluid flow and heat transfer simulations, the fluid flow calculation is the most CPU intensive. An efficient formulation would significantly bring down the computational cost.

One major advantage of the ICV method that came to light during the fluid flow and heat transfer investigation is the potential of high-order cells. These are basically polygonal cells of N sides, where each side accounts for the contribution from the neighbouring cell that shares the common interface. The results of using high-order cells have shown the same characteristics as the high-order flow directed schemes used in a structured mesh to obtain a higher degree of accuracy by reducing the numerical diffusion. For each high-order scheme, re-discretisation of the transport equations is necessary. This can add several more terms in to the equations, making it more complex in both formulation and coding. The high-order cells on the other hand have no such deficiency. This is because the technique is based on the contribution from each control surface. By adding extra surfaces to the cell, the formulation remains the same. The difficulty now is, how to generate the high-order cells automatically ? The current generation of automatic mesh generators are developed for the finite element method, where triangles are the norm. High-order cells can be formed using triangles, but this does not guarantee the orthogonality needed by the method if the cross-diffusion term is not to be included. The "circle-cell" idea, explained in the following section, would in theory guarantee the high-order cells to be orthogonal. This is just one of the

many suggestions made for further studies in the ICV method.

With the integration of all the major casting components such as filling, convective-diffusive solidification and residual stress fast becoming a reality, and with more and more physical phenomena such as holes, porosity, feeding effects and many others of interest to the foundry engineers being added to the integrated casting solution procedure for more complete prediction, an efficient formulation and algorithm is necessary as it can significantly reduce the computational cost. But still we are talking in terms of weeks and days instead of hours and minutes for a full simulation. This time scale for a single prediction hinders the software from becoming an effective design tool for the foundry engineers.

Parallel processing and vectorisation offers an inexpensive option in bringing computational time to a level that is of interest to the engineers. By vectorising, a speed-up factor up to five can be gained over the serial. This is a significant improvement, but the speed-up factor is not likely to go any higher for CFD computations [Ierotheou, Richards and Cross (1989)]. The limitation is due to the inherently scalar elements in the calculations and in the solution procedure. Some 20% of the calculations cannot be vectorised and so anchor the speed-up factor to about five. Parallelisation on the other hand has no such restriction. The limitation in computational performance has more to do with the number of processors available than with any significant constraints with essentially scalar operations [Cross, Johnson and Chow (1989)]. From first impressions, vectorisation seems limited when compared with the

possibilities that can be had with parallelisation. However, it is equally clear that a scalar processor with its own vector unit can gain a speed-up of five in performance. So the ideal computer set-up it seems, for large scale numerical computations, is a network of processors each with its own vector unit.

7.2 Suggestions for further work

7.2.1 Generating orthogonal high-order cells

From the investigation carried out in this study, the ICV method developed during the study is an extension of the successful control volume method that can now handle unstructured finite element meshes. The results obtained using the high-order cell technique show a lot of promise as an alternative substitute for the high-order schemes. For further investigations into the technique a fully- or semi- automatic high-order cell mesh generator is essential. The current generation of automatic mesh generators has been developed with finite element methods in mind, with the basic element/cell being the triangle. This may not necessarily be the best way to proceed in generating the high-order cells by combining triangles together, where the orthogonality of each cell is generally not guaranteed.

One idea for generating high-order cells that are orthogonal is the "circle-cell" technique. A polygonal cell with N degrees of freedom is not much different from a circle, It becomes a circle as N tend to infinity. It can be said that the shape of a circle is the

ideal geometrical figure for the control volume based methods. With equal distance to all the edges from the centre, the information stored at the centre gives a more representative value of the cell contents than a value stored off the centre, where spatial bias could have a significant affect on the solution. Using just circles to fill the physical domain, gaps will appear, see *Figure 7.1*. By drawing a tangent at the meeting of the two circles to the void spaces and with other touching circles, an intersection point would result in the void region. Connecting these intersection points together around the circle forms a polygon, see *Figure 7.2*. And with the neighbouring cell centre connecting it with its own centre's, it results in a connection that is perpendicular to the common surface. Therefore, an orthogonal cell is guaranteed. Mixing of various sizes of circles is straight forward, and adaption or grid refinement can be easily performed. At present, this is only an idea that show a lot of promise for generating high-order cells. A lot more work needs to be done in this area using the idea just described or any that high-order cells can be generated fully- or semi-automatically with cells that are essentially orthogonal.

7.2.2 Partial staggering to remove pressure oscillation

The pressure and the velocity components in this study have been stored at the same location, a non-staggered mesh approach. This approach has the advantage of a single mesh, unlike the classical structured CV for CFD that has a staggered mesh approach, utilising three meshes in two-dimensions and four meshes in three-dimensions. With

the staggered mesh approach, coding is generally more complex and the developer needs to keep track of the meshes for the stored information. This minor housekeeping task is a small price to pay considering the benefits gained in removing pressure oscillations, and in doing so, wall and inlet boundaries pressure need not be prescribed. The Rhie and Chow (1982) velocity approximation can be used to suppress the pressure oscillation in the non-staggered mesh approach, but a prescribed pressure at the wall and inlet boundaries is necessary in the calculations. As described in *Chapter 5*, the boundary pressure is extrapolated and the estimate will inevitably introduce some inaccuracy. This may have a significant influence in some applications. To eliminate the inaccuracy, some kind of mesh staggering is necessary. In an unstructured mesh, a fully staggered approach is not a viable option. This is due to the complexity of the three unstructured meshes in two-dimensions (four in three-dimensions) relationships between each other. This connectivity is generally not possible to achieve, but can be done in few special cases, such as in the quadrilateral (structured) mesh. If a fully staggered approach is not the answer then a partially staggered one may be [Chen (1991)]. By looking at a single element or cell, the pressure can be stored at the cell centre of the element and the velocity components at the corners, see *Figure 7.3*. Where a control volume is formed around the node at the corners, as in the vertex-centred approach, or better still, treat it as a separate mesh and use the cell-centred approach instead, *Figure 7.4*. This means one mesh for the velocity components and one for the pressure. This would remove the pressure oscillation, the need to prescribe a pressure at the wall and inlet boundaries and the need to interpolate fluxes. This needs to be addressed if boundary pressure has a dominant influence in the calculations.

7.2.3 Cross-diffusion term for non-orthogonal meshes

In the current study, for reason of precedence and simplification in formulation in extending the control volume method to handle unstructured meshes, the cross-diffusion term in the evaluation of diffusion fluxes has been left out. This is permissible if the mesh is orthogonal then the term disappears, or the non-orthogonality is not severe, where the cross-diffusion term is small compared with the other part of the diffusion fluxes.

Now with the idea of the ICV method developed and shown to work using orthogonal meshes, both structured and unstructured, the precedence to incorporate the cross-diffusion term in with the diffusion fluxes for non-orthogonal meshes becomes more prominent. This would remove the current constraint applied to meshes so that mesh orientation dependent results may be obtained.

7.2.4 Equation solvers

One area that can significantly improve the overall computational time is in the solution solver. In the current study no great amount of time was used in investigating solvers for CV-UM method. The iterative point-by-point JOR and SOR solvers were used in the study. They are not exactly state-of-the-art solvers but they are robust and reliable. Other iterative solvers that are currently use in CFD computations for which they have

shown great improvement over the JOR and SOR are; solvers such as line-by-line with block-correction [Settari and Aziz (1973)], strongly implicit procedure (SIP) [Store (1968)], conjugate gradient with pre-conditioner [Kightley and Jones (1985 , van der Vorst et al (1982, 1986))] are just a few that need to be investigated for their suitability with the CV-UM method.

7.2.5 Extension to three-dimensions

In this study, the ICV method has been developed and demonstrated to work successfully in two-dimensions for fluid flow and heat transfer in context of solidification. The next step now would be to extend it to three-dimensions, so complex 3D applications in casting and other fluid flow and heat transfer problems can be simulated.

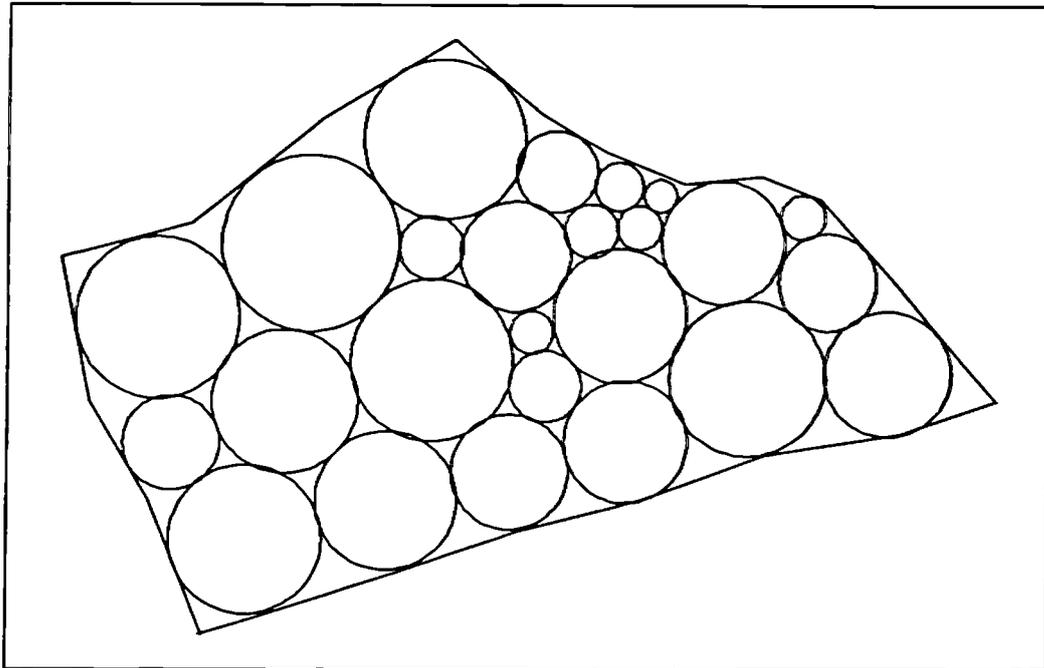


Figure 7.1 Using circles to fill the physical domain.

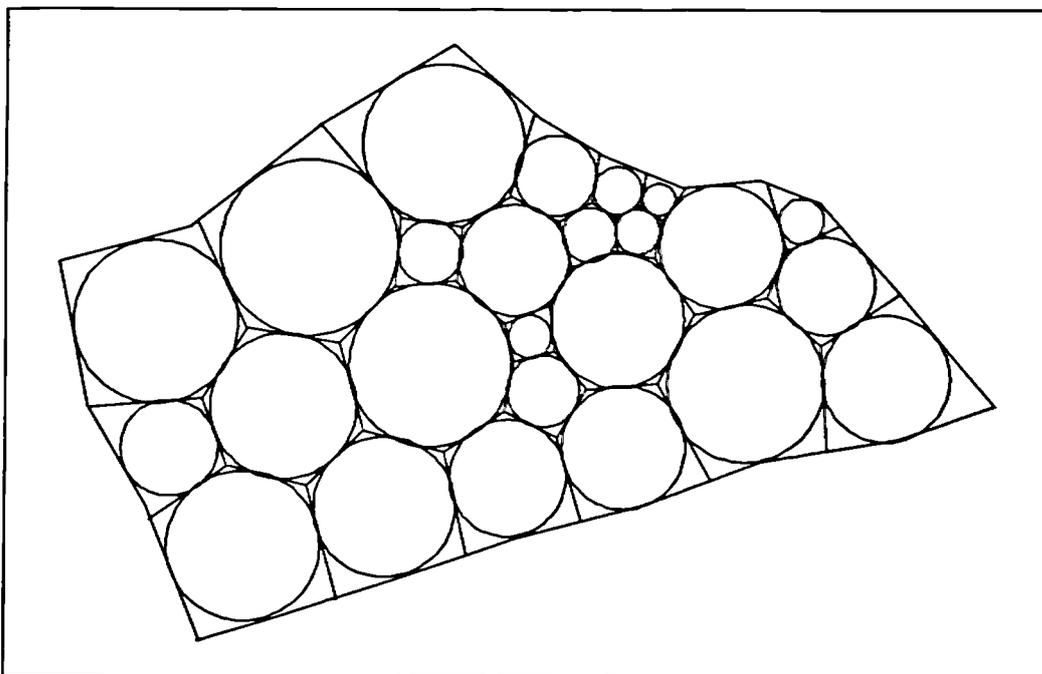


Figure 7.2 Filling the void spaces.

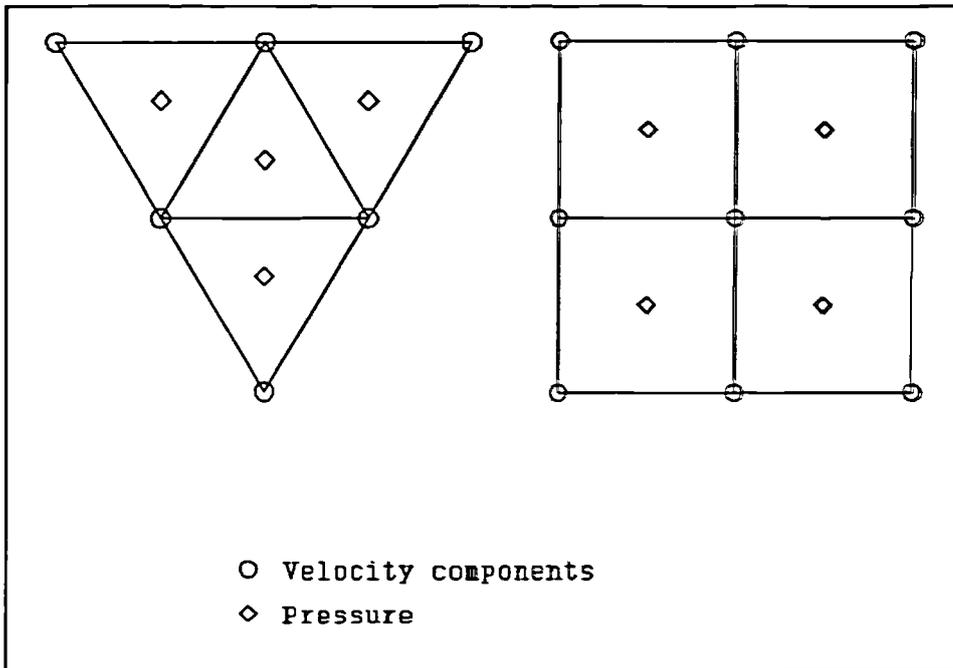


Figure 7.3 Partially staggered velocity-pressure arrangements.

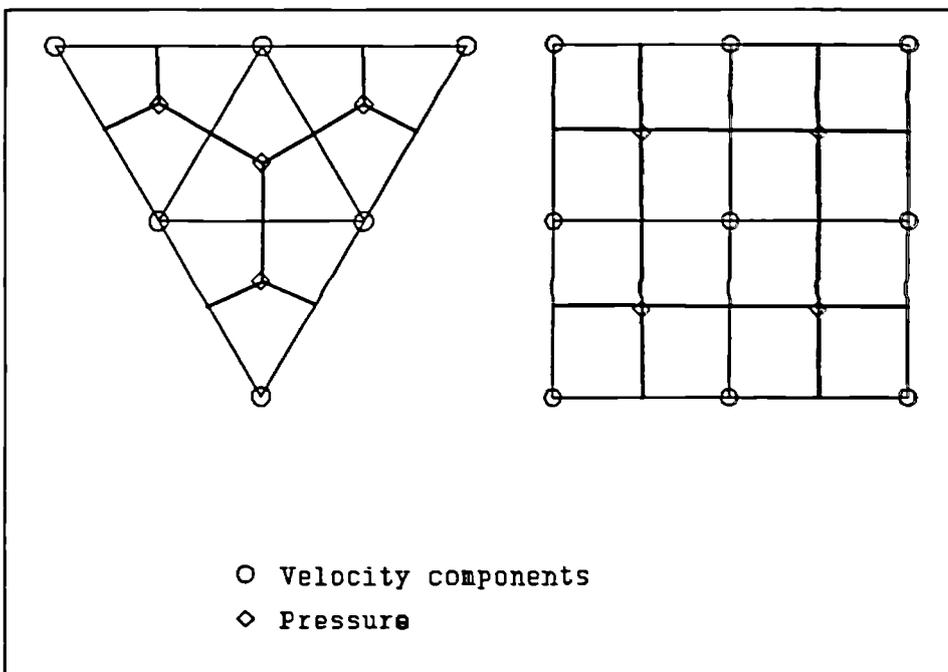


Figure 7.4 Partially staggered with two separate meshes.

REFERENCES**ACME**

ACME Casting Initiative: Advanced Numerical Modelling
Science and Engineering Research Council, grant no. GR/G/36265

Alexandre P, et al (1990)

Experimental and numerical analysis of thermomechanical behaviour of sand cast iron during solidification

Modelling of Casting, Welding and Advanced Solidification Processes - v

Pub. TMS, pp. 237-244.

ANSYS

Swanson Analysis System Inc.
Houston, PA, USA.

ASTEC

AEA Technology, Harwell, Oxon, UK.

Bailey C, Fryer Y D, Cross M and Chow P (1991)

Predicting the Deformation of Castings in Moulds Using a Control Volume Approach on Unstructured Meshes

IMA Conference on Mathematical Modelling of Materials Processing, University of Bristol, UK.

Baliga B R and Patankar S V (1980)

A New Finite Element Formulation for Convection-Diffusion Problems

Numerical Heat Transfer, vol 3, pp. 393-409.

Baliga B R and Patankar S V (1983)

A Control-Volume Finite-Element Method for Two-Dimensional Fluid Flow and Heat Transfer

Numerical Heat Transfer, vol 6, pp. 245-261.

-
- Baliga B R and Patankar S V (1988)
Elliptic System: Finite Element Method I
Handbook of Numerical Heat Transfer
Editors W. J. Minkowycz et al, pub Wiley.
- Bozman J D and Dalton C (1973)
Numerical Study of Viscous Flows in a Cavity
Journal of Computational Physics, vol. 12, p 348.
- Brent A D, Voller V R and Reid K J (1988)
Enthalpy-Porosity Technique for Modelling Convection-Diffusion Phase Change: Application to the Melting of a Pure Metal
Numerical Heat Transfer, vol. 13, pp. 297-318.
- Burgraff O R (1966)
Analytical and Numerical Studies of the Study of Steady Separated Flows
Journal of Fluid Mechanic vol. 24, pp. 113.
- Carman P C (1937)
Fluid flow through granular beds
Trans. Inst. Chem. Engrs, vol. 15, pp 150-166 (1937).
- Carslaw H S and Jaeger J C (1959)
The Conduction of Heat in Solids
2nd Edn, pub Open University Press.
- Chen J (1991)
The Numerical Solution of Complex Fluid Flow Phenomena
Ph.D. thesis, Dept. of Mechanical Engineering, University of Leeds, UK.
- Concus P (1967)
Numerical Solution of the Nonlinear Magnetostatic-Field Equation in Two Dimensions
Journal of Computational Physics, vol. 1, pp. 330-342.
-

Cross M, Bailey C, Chow P and Pericleous K (1992)
Towards an Integrated Control Volume Unstructured Mesh Code for the Simulation of All the Macroscopic Processes Involved in Shape Casting
Numerical Methods in Industrial Forming Processes - NUMIFORM 92
Editors J.-L. Chenot, R.D. Wood and O.C. Zienkiewicz, Pub Balkema.

Cross M, Johnson S and Chow P (1989)
Mapping Enthalpy-Based Solidification Algorithm Onto Vector and Parallel Architectures
Applied Mathematical Modelling, vol. 13, pp 702-709.

Cubberley W H (1979)
Metals Handbook - Properties and Selection: Nonferrous Alloys and Pure Metals
3rd edition, American Society of Metals, Metals Park, Ohio, pp. 763-737.

de Vahl Davis G and Jones I P (1983)
Natural Convection in a Square Cavity: A Comparison Exercise
Int. Journal of Numerical Methods in Fluids, vol. 3, pp. 227-248.

de Vahl Davis G (1983)
Natural Convection in a Square Cavity: A Bench Mark Numerical Solution
Int. Journal of Numerical Methods in Fluids, vol. 3, pp. 249-264.

de Vahl Davis G and Mallinson G D (1976)
An Evaluation of Upwind and Central Difference Approximations by a Study of Recirculation Flow
Computational Fluid, vol. 4, pp. 24-43.

Demirdzic I and Peric M (1990)
Finite Volume Method for Predication of Fluid Flow in Arbitrarily Shaped Domains with Moving Boundaries
Int. Journal for Numerical Methods in Fluids, vol. 10, pp771-790.

Duggin M J (1969)
The Thermal Conductivity of Liquid Gallium
Phys. Lett., vol. 29A(8), pp. 470-471.

FEAT

Nuclear Electric plc
Berkeley Nuclear Laboratories
Berkeley, Gloucestershire, England, UK.

FIDAP

Fluid Dynamics Intl. Inc.
Evanston, IL, USA

FLOW3D

AEA Technology, Harwell, Oxon, UK.

Fryer Y D, Bailey C, Cross M and Lai C-H (1991)
A Control Volume Procedure for Solving the Elastic Stress-Strain Equations on an Unstructured Mesh
Applied Mathematical Modelling, vol. 15, pp. 639-645.

Gau C and Viskanta R (1984)
Melting and Solidification of a Metal System in a Rectangular Cavity
Int. Journal of Heat Mass Transfer, vol. 27, pp. 113-123.

Ghia K N and Ghia U (1988)
Elliptic systems: Finite-Difference Method III
Handbook of Numerical Heat Transfer
Editors W. J. Minkowycz, E. M. Sparrow, G. E. Schneider and R. H. Pletcher, pub Wiley.

Ghia U, Ghia K N and Shin C T (1982)
Solution of the Incompressible Navier-Stokes Equations by a Coupled Strongly-Implicit Multi-Grid Method
Journal of Computational Physics, vol. 48, pp. 387-411.

Gordon W J and Hall C A (1973)
Construction of Curvilinear Coordinate Systems and Applications to Mesh Generation
International Journal for Numerical Methods in Engineering, vol. 7, pp 461-477.

Hibbitt, Karlson and Sorenson (1987)
Inc. ABAQUS User and Theory Manuals.

Ierotheou C S (1990)
The Simulation of Fluid Flow Processes Using Vector Processors
Ph.D. Thesis, School of Mathematics, Statistics and Computing, University of Greenwich
(formerly Thames Polytechnic), London, UK.

Ierotheou C S and Cross M (1987)
High Level Subroutines for the Masscomp Vector Processor
School of Mathematics, Statistics and Computing, University of Greenwich (formerly
Thames Polytechnic), London, UK.

Ierotheou C S, Richards C W and Cross M (1989a)
*Vectorisation of the SIMPLE Solution Procedure for CFD Problems - Part I: A Basic
Assessment*
Applied Mathematical Modelling, vol. 13, pp. 524-529.

Ierotheou C S, Richards C W and Cross M (1989b)
*Vectorisation of the SIMPLE Solution Procedure for CFD Problems - Part II: The
Impact of Using a Multigrid Method*
Applied Mathematical Modelling, vol. 13, pp. 530-536.

Jones I P (1979)
*A comparison Problem for Numerical Methods in Fluid Dynamics: the "Double-Glazing"
Problem*
Numerical Method in Thermal Problem
Editors R W Lewis and K Morgan, Pineridge Press, Swansea, UK.

Joshua L and Prince A (1990)
Coupled thermomechanical model for continuous bar casting
Modelling of Casting, Welding and Advanced Solidification Processes - v
Pub. TMS. pp. 229-236.

Kightley J R and Jones I P (1985)
*A Comparison of Conjugate Gradient Preconditionings for Three-Dimensional Problems
on a CRAY-1*
Comput. Phys. Comm., vol. 37, pp. 205-214.

-
- Lewis R W and Roberts P M (1987)
Finite Element Simulation of Solidification Problems
Modelling the Flow and Solidification of Metals
Editor T. J. Smith, pub Martinus Nijhoff.
- Lonsdale R D and Webster R (1989)
The application of finite volume methods for modelling three dimensional incompressible flow on an unstructured mesh
Numerical Methods in Laminar and Turbulent Flow
Editors C. Tayler et al, pub Pineridge Press, p1615.
- Markatos N C and Pericleous K A (1984)
Laminar and Turbulent Natural Convection in an Enclosed Cavity
Int. Journal of Heat Mass Transfer, vol. 27, No. 5, pp. 755-772.
- MASSCOMP (1984)
Masscomp Computers, Reference Manual, Order No. 075-00123-00-00, Rev. B,
Massachusetts, USA.
- Melaaen M C (1992)
Calculation of Fluid Flows with Staggered and Nonstaggered Curvilinear Nonorthogonal Grids - The Theory
Numerical Heat Transfer, Part B, vol. 21, pp. 1-19.
- Melaaen M C (1992)
Calculation of Fluid Flows with Staggered and Nonstaggered Curvilinear Nonorthogonal Grids - A Comparison
Numerical Heat Transfer, Part B, vol. 21, pp. 21-39.
- Michalek K P, Kelly J E and Dantzig J A (1986)
Modeling of In-Mold Heat Transfer In Continuous Casting of Steel
Modelling and Control of Casting and Welding Processes
Editors S. Kou and R. Mehrabian, pub AIME (1986).
- Patankar S V (1980)
Numerical Heat Transfer and Fluid Flow
Hemisphere, Washington, DC.
-

Patankar S V and Spalding D B (1972)
A Calculation Procedure for Heat, Mass and Momentum Transfer in Three-Dimensional Parabolic Flows

Int. Journal of Numerical Heat and Mass Transfer, vol. 15, pp. 1787-1806.

Patel M K (1987)

On the False-Diffusion Problem in the Numerical Modelling of Convection-Diffusion Process

Ph.D. Thesis, School of Mathematics, Statistics and Computing, University of Greenwich (formerly Thames Polytechnic), London, UK.

Pehlke R D, Jeyarajan A and Wada H (1982)

Summary of Thermal properties for Casting Alloys and Mould Materials

Dept. of Materials and Metallurgical Engineering, University of Michigan, USA.

PHOENICS

CHAM Ltd, England, UK.

Rathjen K A and Jiji L M (1971)

Heat conduction with melting of freezing in a corner

Journal of Heat Transfer, pp 101-109.

Rhie C M and Chow W L (1982)

A Numerical Study of the Turbulent Flow Past an Isolated Airfoil with Trailing Edge Separation

AIAA J., vol. 21, pp. 1525-1532.

Sammonds M, Lewis R W, Morgan K and Symberlist R (1985)

Finite Element Modelling of the Mold-Metal Interface in Casting Simulation with Coincident Nodes or Thin Element

Computational Techniques in Heat Transfer

Editors R. W. Lewis et al, pub Pineridge Press.

Samonds M and Waite D (1991)

3D Finite Element Simulation of Filling Transients in Metal Castings

IMA Conference on Mathematical Modelling of Materials Processing, University of Bristol, UK.

-
- Settari A and Aziz K (1973)
A generalization of the Additive Correction Methods for the Iterative Solution of Matrix Equations
SIAM Journal of Numerical Analysis, vol. 10, pp. 506-521.
- Schneider G E (1988)
Elliptic System: Finite Element Method II
Handbook of Numerical Heat Transfer
Editors W. J. Minkowycz et al, pub Wiley.
- Stone N L (1968)
Iterative Solution of Implicit Approximation of Multidimensional Partial Differential Equations
SIAM Journal of Numerical Analysis, vol. 5, pp. 530-558
- Tao L N (1967)
Generalised numerical solution of freezing of saturated liquid in cylinders and spheres
AIChE Jnl, vol. 13, pp 165-169.
- Thompson J F, Thames F C and Mastin C W (1974)
Automatic numerical generation of body-fitted curvi-linear coordinate system for fields containing any number of arbitrary 2D bodies
Journal Computational Physics, vol. 15, pp. 291-319.
- Thompson J F, Warsi Z U A and Mastin C W (1982)
Boundary-Fitted Coordinate System for Numerical Solution of Partial Differential Equations - A Review
Journal of Computational Physics, 47, pp. 108.
- Varga R S (1962)
Matrix Iterative Analysis
Prentice-Hall, Englewood Cliffs, N.J.
- Voller V R and Cross M (1983)
Use of the enthalpy method in the solution of Stefan problem
Numerical Methods in Thermal Problems
Editors R W Lewis and K Morgan, pub Pineridge Press.
-

- Voller V R and Cross M (1985)
Application of Control Volume Enthalpy Methods in the Solution of Stefan Problems
Computational Techniques in Heat Transfer
Editors R. W. Lewis and K. Morgan, pub Pineridge Press, pp. 245-276.
- Voller V R, Cross M and Markatos N C (1987)
An Enthalpy Method for Convection/Diffusion Phase Change
Int. Journal of Numerical Methods in Engineering, Vol. 24, pp. 271-284.
- Voller V R and Prakash C (1987)
A Fixed Grid Numerical Modelling Methodology for Convection-Diffusion Mushy Region Phase Change Problems
Int. Journal of Heat Mass Transfer, Vol. 30, No. 8, pp. 1709-1719.
- Voller V R and Swaminathan C R (1991)
General source-based method for solidification phase change
Numerical Heat Transfer, Part B, vol. 19 pp. 175-189.
- Walthier M K (1984)
Experimental verification of C.A.S.T.
Modelling and Control of Casting and Welding Processes
Editors S. Kou and K. Mehrabian, TMS-AIME.
- Winslow A M (1966)
Numerical Solution of Quasilinear Poisson Equation in a Nonuniform Triangle Mesh
Journal of Computational Physics, vol 1, pp. 149-172.
- Zeng C and Pehlke R D (1985)
Analysis of heat transfer at metal-sand mould boundaries and computer simulation of Gray Iron casting
AFSTrans, vol. 13, p 275.

