GRANT SPONSORED BY ADVANCED MICRO DEVICES (AMD)

USM PROJECT CODE NUMBER A0836

COMPREHENSIVE REPORT

Project Title: **Mold Flow FEA Simulation Software Development**

Project Leader: Assoc. Prof. Dr Ishak Hj Abdul Azid/ Prof. K.N.Seetharamu
School of Mechanical Engineering
Universiti Sains Malaysia
Engineering Campus
14300 Nibong Tebal
Penang

Co-Researcher: Assoc. Prof. Dr Ghulam Abdul Quadir
School of Mechanical Engineering
Universiti Sains Malaysia
Engineering Campus
14300 Nibong Tebal
Penang

Duration of Project: 1 June 2003 – 30 June 2005

Contents:

1.0 Introduction

1.1 Project Description

The present trend in electronics industry is towards the miniaturization of product designs. Basic concept is to make the products lighter, smaller, less expensive and at the same time to be faster, more powerful, reliable, user-friendly, more attractive (aesthetics look) with added functional features. Few examples of today's "shrinking" products include cellular phones, pagers, personal digital assistants (PDAs), laptops, personal notebook computers, camcorders, palmtop organizers, telecommunications equipments and automotive microelectronic components.

Electronic packaging is the dynamic process of physically locating, connecting, and protecting electronic components. The packaging of today's electronic equipment has become a major factor in the design and manufacture of the total system. Microelectronics packaging and interconnection technologies have undergone both evolutionary and revolutionary changes to serve the trend towards miniaturization in electronics devices. The requirements for smaller, more compact products and high density for high speed circuitry drive the design of packagers to higher input/output (I/O) and smaller package size. These in turn demand for higher interconnect density, number of I/O pins and more importantly stringent requirements for package production.

Integrated circuits (ICs) are processed on a large piece of semiconductor substrate called a wafer. Wafer sizes can vary from 3 to 12 inches in diameter. Depending on the size of an individual IC chip, there may be hundreds to thousands of IC chips on a wafer. Once the IC's fabrication process on the wafer is finished, the wafer is then cut by a fully automatic dicing saw whose blade is tipped with diamond tips. The process is called 'wafer dicing' process the end product of which is individual ICs or IC chips. Thus a piece of semiconductor material is transformed into a functional microelectronic part (Tummala, 2001). An IC chip cannot perform its designated function until it is packaged such that it is interconnected with the rest of the system and protected. Each chip has its unique packaging process. The package is generally fabricated independent of the IC chips. When the IC chip /die is ready to be packaged, it is bonded or attached to a substrate or lead frame. For packaging, this IC on substrate will be encapsulated by means of transfer molding or underfilling process.

Encapsulation is an electronic packaging technology that is typically done by means of low temperature polymers. Encapsulation or sealing provides an economical way to protect device packages by isolating the active devices from environmental pollutants and at the same time offering mechanical protection by structural coupling of the device to the constituent packaging materials into a robust package. Encapsulation materials are typically molded on to the IC or dispensed under the die, such as with flip chip ceramic ball grid array (BGA) packages. Former process is known as 'molding process' and the later one is called as 'underfilling process'. Encapsulation processes can be classified into two main types as transfer molding and liquid encapsulation.

In order to achieve higher electrical performance due to higher interconnect density of the present day electronic packages, the silicon die is attached to the package substrate and electrically connected through an array of solder bumps. Due to high coefficient of thermal expansion (CTE) differences between the silicon die and package substrates, large stresses are developed in the interconnects during temperature cycling and normal chip operations. To reduce these stresses [5], the stand-off region between die and package is encapsulated with epoxy/resins using the so-called underfill encapsulation process. Underfilling is most popular among liquid encapsulation processes and has been widely used to increase the thermal cycle fatigue life and to improve solder joint reliability of area array flip chip die or Chip Scale Package attachments both for use in internal packages and on the printed circuit boards (PCBs). It is the most critical operation in flip chip assembly process.

Underfilling materials can help distribute the shear stress on the solder bumps (Flip chip) or balls (CSPs) caused by the mismatch of Coefficient of Thermal Expansion (CTE) between the silicon and substrate or between the package and second level PCB. For chip scale packages, the effect of underfilling on reliability depends on package structure as well, such as leadless and laminated chip scale packages (CSPs). Good underfill materials and desired fillet geometry could prevent moisture, dust and other corrosive materials from reaching the surface of the die as well. Underfilling process will reduce the possibility of the fatigue cracks on the solder bumps or balls and enhance the overall mechanical strength of the assembly.

During the encapsulation process, the substrate is typically heated to approximately 80-90°C to reduce the underfilling material viscosity, resulting in improved underfill flow properties (Yang et al., 2003). The CTE of the underfilling material should match

that of the solder bumps and balls embraced. This CTE should be optimized not only at room temperature, but also at all package reliability testing temperature. The best option is to manufacture the underfilling materials with their glass transition temperature (Tg) at about 150°C or higher to maintain CTE constant through the reliability tests (Yang et al., 2003). Key factors affecting the underfilling quality and manufacturing cycle time are: the underfilling gap, the solder bump diameter, the number of bumps in the array, the layout of the bumps, and the bump pitch time. As the semiconductor technology progresses towards still higher levels of integration, high performance and increasing functionality, the design and fabrication of the package that will meet the requirements of modern and future microelectronic systems becomes increasingly complex and challenging. This makes encapsulation process much more complex and unpredictable. Therefore the analysis of encapsulation process is essential for a package design. For better design and optimization of the process, fluid flow analysis during mold filling process is necessary step for proper design of the package and in turn for defect free high volume manufacturing of electronic packages.

The epoxy resin is used as molding compound in transfer molding process. The epoxy resin is thermosetting material and flow of molten epoxy resin in a chip cavity is highly non-linear and transient analysis problem. The problem statement of the present study is to develop a numerical solution algorithm to analyze the 3-dimensional (3D) flow behavior of epoxy mold compound (EMC) in chip cavity for a given configuration of electronic package. The broad objectives of this research work are as follows.

1. To develop a numerical solution scheme based on finite element method (FEM) for the analysis of 2-dimensional (2D) flow in a mold cavity.
2. To study of Single chip and Multi chip packages using 2D model for the flow behavior of EMC.
3. To conduct the parametric study for 2 D model to know the effect of process parameters.
4. To develop a robust 3D flow simulation software code using FEM to predict the mold filling behavior of EMC for Generalized Newtonian Fluid case and to perform the parametric study.
5. To optimize of the gate sizes using neural networks and genetic algorithm.

1.2 Project Activities

The activities carried out in this project can be summarized as follows

Stage 1: Studying the process of encapsulation of electronic package (EP)

As part of the initial study, the process of mold filling in real EP environment was investigated. This was carried out in the AMD plant at Bayan Lepas, Pulau Pinang.

Stage 2: Modeling the molding process using hybrid characteristic based split (CBS) method and volume of fluid (VOF) technique

Characteristic based split (CBS) method was used to solve the Navier Stokes equations to get primitive variables namely velocity and pressure. The velocity field was used in volume of fluid (VOF) technique to trace the fluid flow at different time steps.

Stage 3: Modelling the single chip and multi chip packages using 2D model

The algorithm developed in stage 2 was applied to model the single and then multi chip packages for the flow behavior of EMC. The flow profile was studied and fully investigated.

Stage 4: Conducting the parametric study for 2 D model

The parametric study was carried out to know the effect of process parameters on the flow profile. The velocity, time taken and void can be determined by systematically changing the input parameters.

Stage 5: Development of a robust 3D flow simulation software code using FEM

The developed simulation software code for 2D model was extended to simulate 3D flow for EMC. The developed 3D flow simulation code can predict the mold filling behavior of EMC for generalized Newtonian fluid case and can also perform the parametric study.

Stage 6: Optimization of the gate sizes using neural networks and genetic algorithm.

The gate sizes of mold filling for a specific package can be optimized using neural networks and genetic algorithms. The gate sizes were optimized for the flow profile and also to avoid voids.

The various stages of the work and their results are reported in the theses produced from this research. Several excerpts of the reports are available in the Appendix A.

1.3 Project Benefits

Through this research, the numerical analysis of mold filling in electronic packaging material has been developed. The flow profiles have been determined for the mold filling of the electronic packaging material. Simulation software code was developed in which can enhance the teaching and research in the school of mechanical engineering especially in the subject of electronic packaging.

1.4 Project Duration

The project started in June 2003 and was completed in June 2005, which is for duration of two years.

1.5 Approved Grant Amount

The total amount approved by AMD for this project is RM 61,400.00.

1.6 Project Cost

The total amount spent for this project was RM 61,400.0.

2 Project Contribution/Achievement

Thesis and Publications

The contribution of the research in terms of theses and publications are as follows:

1. PhD Thesis titled: Computational Fluid Flow Analysis of Mold Filling Process in Electronic Packaging – Venkatesh M.Kulkarni (October, 2006).

Journal Papers:

1. Venkatesh M. Kulkarni, K. N. Seetharamu, Ishak Abdul Azid, P. A. Aswatha Narayana¶and Ghulam Abdul Quadir, "Numerical simulation of underfill encapsulation process based on characteristic split method", Int. J. Numer. Meth. Engng 2006; 66:1658–1671

Copy of the full paper is available in the Appendix B as attached to this report.

Conference Papers:

1. Venkatesh M.Kulkarni, Ishak A. Azid, K.N. Seetharamu, and P.A.Aswathanarayana, "An Analysis of three dimensional flow in Electronic Packages", 1st International Conference and 7th AUN/SEED-Net Field wise Seminar on Manufacturing and Material Processing 200(ICMM2006) , pp 677-682 ,March 14-16, 2006,Kuala Lumpur, Malaysia.

2. Venkatesh M.Kulkarni, Heng Chai Wei, Ishak A. Azid, K.N. Seetharamu, and P.A.Aswathanarayana, "Fluid Flow in Flip Chip Electronic Packages", 18th National & 7th ISHMT-ASME Heat and Mass Transfer Conference, January 4-6,2006, IIT Guwahati, India.

3. Venkatesh M.Kulkarni, Ishak A. Azid, K.N. Seetharamu, P.A.Aswathanarayana, "Numerical Model to analyze IC Chip Encapsulation Process", International Electronic Packaging Technical Conference and Exhibition (IPACK 2005), July 17-22, 2005, San Francisco, CA, USA.

4. Venkatesh M.Kulkarni,KN Seetharmu, P.A.Aswatha Narayana, I.A.Azid, & G.A.Quadir ,"Flow analysis for flip chip underfilling process using characteristic based split method",6th Electronic Packaging Technology Conference(EPTC) , pp.615-619, 8-10th Dec.04, Singapore.

5. C.W.Liang, Venkatesh M.Kulkarni, P.A.aswatha Narayana , G.A.Quadir,I.A.Azid & K.N.Seetharamu," Mould filling in Electronic Packaging", Proceedings of 6th International Conference on Electronic Materials and Packaging (EMAP 2004) , pp 529-534, Penang, Malaysia.

6. C. W. Liang, Venkatesh M.Kulkarni, P.A.Aswatha Narayana and K.N.Seetharamu, "Parametric studies in transfer molding for Newtonian fluids", Proceedings of Regional Conference on Environmental and Ecological Modeling (ECOMOD 2004), 15-16 Sept.2004, Penang, Malaysia.

7. Venkatesh M. Kulkarni, Ghulam A.Quadir,K .N.Seetharamu, P.A.A.Narayana and Ishak Abdul Azid, "Characteristic Based Split Algorithm used in Underfilling Encapsulation Process", European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS), 24-28 July 2004, Jyväskylä, Finland.

3 Conclusion

In this project, the mold filling software code of the electronic packages has been successfully developed. The mold filling process is a transient problem and one requires a numerical model to simulation the flow of molding compound into the chip cavity. This numerical simulation of a transfer mold filling is a viable tool in optimizing molding tool design and performance. Thus there is of great importance for computer aided engineering (CAE) for polymer process operations. The vast majority of these CAE tools concerned with the injection molding process. Very few commercial software packages are available for thermoset molding process which can take special care of transfer molding of IC packages. In this context, the development of application specific software code plays a significant rule. The main objective of all these packages is to simulate the flow filling profile in transfer molding in order to achieve the balanced mold filling.

In this research work, the transfer molding process was successfully modeled using hybrid CBS -VOF technique to simulate the flow in chip cavity and thus to get the flow filling profile. Characteristic based split (CBS) method was used to solve the Navier Stokes equations and the velocity field was used in volume of fluid (VOF) technique to trace the fluid flow at different time steps. The time taken by a molding compound to fill the chip cavity is called 'filling time' and is thus known easily from the proposed CBS-VOF algorithm. An effort has also been made in this work to optimize the gate for a specific package.

The outcome of this research can enhance current understanding of mold filling in electronic packaging. Future study can be investigated to expand the knowledge of flow profiles in mold filling of electronic package by studying the parameters involved in the process.

4 Acknowledgement

We would like to convey our sincere thanks to American Micro Devices (AMD) for the offer of the grant and the generous support in using their equipments that has enabled this research to be carried out and completed successfully.

Report prepared by:

.....
Assoc. Prof. Dr Ishak Hj Abdul Azid
School of Mechanical Engineering
Universiti Sains Malaysia
Engineering Campus
14300 Nibong Tebal
Penang

APPENDIX A

Extract from PhD Thesis entitled: Computational Fluid Flow Analysis of Mold Filling Process in Electronic Packaging – Venkatesh M.Kulkarni (October, 2006).

CHAPTER 3

ANALYSIS

3.0 Overview

In this chapter, the following items related to fluid flow analysis during mold filling process are studied.

- ❖ Need for Mold filling analysis
- ❖ Characteristic Based Split Scheme
- ❖ 2D Flow Analysis
- ❖ 3D Flow Analysis
- ❖ Front tracking method
- ❖ Optimization

3.1 Need for Mold Filling Analysis

Transfer molding of integrated circuits (ICs) is the most popular method for the manufacture of plastic electronic packages. Although it is quite mature technology, transfer molding is subjected to several manufacturing defects. The most common transfer molding defects are short shot, void formation, wire sweep, paddle shift and other stress induced problems (Manzoine,1990). Further more, the trend to produce faster, smaller and cheaper electronic devices is pushing the electronic packaging technology towards higher packaging density with thinner and smaller profile. This in turn has imposed even more requirements on molding process and material formulation. This makes the encapsulation process much more complicated and unpredictable.

Even though one can use trial and error method in industry, but still it is difficult to analyze the transfer molding process as it involves complex interactions between fluid flow, heat transfer and polymerization of epoxy molding compound (EMC). This necessitates analysis of

complex flow behavior of EMC. An increased demand for improved packages is largely responsible for the emphasis on flow modeling and analysis. This has made the computer aided engineering (CAE) as an effective tool to analyze the complicated flow phenomena inherent in the process of plastic encapsulation of microelectronics (PEM) (Nguyen, 1993; Turng, 1994; Chang et al., 1998). By predicting the flow patterns, one can avoid costly trial and error mold design procedure usually required when developing new high quality electronic packages.

Mold filling phase of Transfer Molding is a transient, non-isothermal process dependent on non-Newtonian flow behavior of epoxy molding compound. If the geometry of the part to be molded is complex in nature, then the analysis of mold filling process becomes extremely difficult. However analysis of molding process has been carried out by using simplified models. Mold filling time and void prediction are the most important parameters to be analyzed using mathematical models. The amount of warpage, wire sweep and paddle shift are secondary issues which can be evaluated based on mold filling results. Solution methodology for predicting a mold filling time involves solving flow governing equations to get primitive variable fields and then tracing fluid front using suitable front tracking method.

The Hele Shaw Model is the simplest and most widely used mathematical model to simulate the mold filling process. The Generalized Hele Shaw flow model introduced by Hieber and Shen (1978, 1980) provides simplified governing equations for non-isothermal, non-Newtonian and inelastic flows in cavities. More treatise on Hele Shaw flow model has been covered by Tucker III (1989). The present trend to have smaller electronic products with more features has made it necessary to analyze the mold filling process taking into real process conditions.

In this research work, a software code is developed to simulate both the 2D and 3D mold filling processes using the Characteristic Based Split method. It is well established algorithm to

solve the complex flow problems. It is based on finite element method and it is alternative to finite volume method to solve both compressible and incompressible flows. The pressure and velocity fields are obtained from CBS scheme and the velocity field is then used in the VOF technique, which is a most widely used front tracking algorithm to track the fluid front at different time intervals.

3.2 Characteristic Based Split Scheme

Mold filling process has been analyzed based on the convection theory of heat transfer process, where both heat and fluid flows interact with each other. Since it is complex to solve analytically, one has to necessarily rely on the numerical solution. The Finite Element Method is a popular numerical solution method, which is able to solve most of the complex problems in engineering world.

In this research work, the Characteristic Based Split (CBS) Method is used to obtain the solution for flow governing equations. By introducing the Characteristic Galerkin procedure and the split in momentum equations, the method becomes more stable and can be used to solve real flow problems of both compressible and incompressible nature. Hence this method is referred by a name 'Characteristic Based Split' method. For most of the fluid flow applications, the fluid is assumed as incompressible and the Navier-Stokes equations are used to represent the mathematical model. Split in momentum equations and subsequent velocity correction has been reported by Comini and Del (1972), Gresho and Sani (1999) and Ramasway et.al.(1992). The CBS procedure is efficient and flexible due to many extra provisions to improve stability and accuracy of incompressible flow calculations. The CBS method has been shown to be applicable to a wide variety of fluid dynamics problems ranging from incompressible flow to hypersonic flow.

In this research work computational flow analysis has been carried out by assuming the EMC as an incompressible fluid.

The general flow governing incompressible Navier Stokes equations are written as below.

Continuity Equation

$$\nabla . u = 0 \tag{3.1}$$

Momentum Equations

$$u_t + (u.\nabla)u = \frac{1}{\rho}\nabla p + \nu \nabla^2 u \tag{3.2}$$

where 'u' is the velocity vector , p is pressure and ν is kinematic viscosity.

The CBS scheme is implemented to obtain the solution of above equations. It consists of three basic steps which are explained below.

1. In the first step the pressure term from the momentum equation is dropped and an intermediate velocity or pseudo velocity is calculated.
2. In the second step, the pressure is obtained from a Pressure Poisson equation.
3. Finally intermediate velocities are corrected to get the actual velocity values.

The above three steps are cornerstones of the CBS scheme. These steps implemented using Finite Element Method. Any additional scalar quantities such as temperature and concentration can be added as a fourth step. The CBS scheme has been extensively covered in books by Zienkiewicz (2000) and Lewis et al. (2004). Both have been referred in this work.

In this research work, CBS method has been used to obtain the solution of both 2D and 3D Navier Stokes equations for a flow in a chip cavity during mold filling process. In next few pages, CBS scheme application to 2D and 3D has been explained in detail.

3.3 2D Mold Filling analysis

The flow governing Navier Stokes equations are written as below.

(i) Continuity Equation:

$$\frac{\partial u_1}{\partial x_1}+\frac{\partial u_2}{\partial x_2}=0 \tag{3.3}$$

where u_1,u_2 are velocities along x_1 and x_2 directions.

(ii)Momentum Equations:

Momentum equations in non-conservative form can be written as

x_1-Momentum Equation:

$$\frac{\partial u_1}{\partial t}+u_1\frac{\partial u_1}{\partial x_1}+u_2\frac{\partial u_1}{\partial x_2}=-\frac{1}{\rho}\frac{\partial p}{\partial x_1}+\nu\left(\frac{\partial^2 u_1}{\partial {x_1}^2}+\frac{\partial^2 u_1}{\partial {x_2}^2}\right) \tag{3.4}$$

x_2-Momentum Equation:

$$\frac{\partial u_2}{\partial t}+u_1\frac{\partial u_2}{\partial x_1}+u_2\frac{\partial u_2}{\partial x_2}=-\frac{1}{\rho}\frac{\partial p}{\partial x_2}+\nu\left(\frac{\partial^2 u_2}{\partial {x_1}^2}+\frac{\partial^2 u_2}{\partial {x_2}^2}\right) \tag{3.5}$$

where ρ is density and ν is the kinematic viscosity of the fluid.

iii) Energy Equation:

$$\frac{\partial T}{\partial t}+u_1\frac{\partial T}{\partial x_1}+u_2\frac{\partial T}{\partial x_2}=\alpha\left(\frac{\partial^2 T}{\partial {x_1}^2}+\frac{\partial^2 T}{\partial {x_2}^2}\right)+\frac{\eta\dot{\gamma}^2}{\rho C_p} \tag{3.6}$$

where α is the thermal diffusivity, η, is viscosity,and C_p is specific heat. $\dot{\gamma}$ is shear rate and is defined by the equation

$$\dot{\gamma} = \sqrt{\left(\frac{\partial u_1}{\partial x_1}\right)^2 + \left(\frac{\partial u_2}{\partial x_2}\right)^2 + \left(\frac{\partial u_3}{\partial x_3}\right)^2}$$

Now CBS method can be implemented by following above mentioned basic steps to get a solution to the convective heat transfer equations.

Step 1: Calculation of Intermediate Velocity or Momentum field

This step is carried out by removing the pressure term from momentum equations (3.4) and (3.5). Then the intermediate velocity component equations in semi-discrete form (Lewis et.al. 2004) is

Intermediate x_1 momentum equation:

$$\frac{\tilde{u}_1 - u_1^n}{\Delta t} + u_1 \frac{\partial u_1^n}{\partial x_1} + u_2 \frac{\partial u_1^n}{\partial x_2} = \nu \left(\frac{\partial^2 u_1}{\partial {x_1}^2} + \frac{\partial^2 u_1}{\partial {x_2}^2} \right)^n \qquad (3.7)$$

Intermediate x_2 momentum equation:

$$\frac{\tilde{u}_2 - u_2^n}{\Delta t} + u_1 \frac{\partial u_2^n}{\partial x_1} + u_2 \frac{\partial u_2^n}{\partial x_2} = \nu \left(\frac{\partial^2 u_2}{\partial {x_1}^2} + \frac{\partial^2 u_2}{\partial {x_2}^2} \right)^n \qquad (3.8)$$

In the CBS scheme, the Characteristic Galerkin method is used for temporal discretization of equations (3.7) and (3.8). The governing equations are discretized first in time according to a Taylor's series prior to the Galerkin spatial discretization. In the Characteristic Galerkin method, the temporal derivative is discretized along the characteristic, where the equation is self-adjoint in nature.

By applying Characteristic Galerkin Method, the above equations can be written as

$$\frac{\tilde{u}_1-u_1^n}{\Delta t}=-u_1\frac{\partial u_1^n}{\partial x_1}-u_2\frac{\partial u_1^n}{\partial x_2}+\nu\left(\frac{\partial^2 u_1}{\partial {x_1}^2}+\frac{\partial^2 u_1}{\partial {x_2}^2}\right)^n+u_1\frac{\Delta t}{2}\frac{\partial}{\partial x_1}\left[u_1\frac{\partial u_1}{\partial x_1}+u_2\frac{\partial u_1}{\partial x_2}\right]^n$$
$$+u_2\frac{\Delta t}{2}\frac{\partial}{\partial x_2}\left[u_1\frac{\partial u_1}{\partial x_1}+u_2\frac{\partial u_1}{\partial x_2}\right]^n \qquad (3.9)$$

$$\frac{\tilde{u}_2-u_2^n}{\Delta t}=-u_1\frac{\partial u_2^n}{\partial x_1}-u_2\frac{\partial u_2^n}{\partial x_2}+\nu\left(\frac{\partial^2 u_2}{\partial {x_1}^2}+\frac{\partial^2 u_2}{\partial {x_2}^2}\right)^n+u_1\frac{\Delta t}{2}\frac{\partial}{\partial x_1}\left[u_1\frac{\partial u_2}{\partial x_1}+u_2\frac{\partial u_2}{\partial x_2}\right]^n$$
$$+u_2\frac{\Delta t}{2}\frac{\partial}{\partial x_2}\left[u_1\frac{\partial u_2}{\partial x_1}+u_2\frac{\partial u_2}{\partial x_2}\right]^n \qquad (3.10)$$

Step 2: Calculation of Pressure

The pressure field is calculated from a pressure Poisson equation. It is obtained from the intermediate velocity field. If the pressure terms are not removed from momentum equations, we can directly get the actual velocities. Writing the semi-discrete form of the momentum equations without removing pressure term, we get

Semi-discrete x_1 momentum equation

$$\frac{u_1^{n+1}-u_1^n}{\Delta t}=-u_1\frac{\partial u_1^n}{\partial x_1}-u_2\frac{\partial u_1^n}{\partial x_2}+\nu\left(\frac{\partial^2 u_1}{\partial {x_1}^2}+\frac{\partial^2 u_1}{\partial {x_2}^2}\right)^n-\frac{1}{\rho}\frac{\partial p^n}{\partial x_1}+$$
$$+u_1\frac{\Delta t}{2}\frac{\partial}{\partial x_1}\left[u_1\frac{\partial u_1^n}{\partial x_1}+u_2\frac{\partial u_1^n}{\partial x_2}+\frac{\partial p^n}{\partial x_1}\right]^n+ \qquad (3.11)$$
$$+u_2\frac{\Delta t}{2}\frac{\partial}{\partial x_2}\left[u_1\frac{\partial u_1^n}{\partial x_1}+u_2\frac{\partial u_1^n}{\partial x_2}+\frac{\partial p^n}{\partial x_1}\right]$$

Semi-discrete x_2 momentum equation

$$\frac{u_2^{n+1}-u_2^n}{\Delta t}=-u_1\frac{\partial u_2^n}{\partial x_1}-u_2\frac{\partial u_2^n}{\partial x_2}+\nu\left(\frac{\partial^2 u_2}{\partial x_1^{\,2}}+\frac{\partial^2 u_2}{\partial x_2^{\,2}}\right)^n-\frac{1}{\rho}\frac{\partial p^n}{\partial x_2}+$$
$$+u_1\frac{\Delta t}{2}\frac{\partial}{\partial x_1}\left[u_1\frac{\partial u_2^n}{\partial x_1}+u_2\frac{\partial u_2^n}{\partial x_2}+\frac{1}{\rho}\frac{\partial p^n}{\partial x_2}\right]+ \qquad (3.12)$$
$$+u_2\frac{\Delta t}{2}\frac{\partial}{\partial x_2}\left[u_1\frac{\partial u_2^n}{\partial x_1}+u_2\frac{\partial u_2^n}{\partial x_2}+\frac{1}{\rho}\frac{\partial p^n}{\partial x_2}\right]$$

Subtracting eq.(3.9) from eq.(3.11) and eq.(3.10) from eq.(3.12), we get the following two equations.

$$\frac{u_1^{n+1}-\tilde{u}_1}{\Delta t}=-\frac{1}{\rho}\frac{\partial p^n}{\partial x_1}+u_1\frac{\Delta t}{2}\frac{\partial}{\partial x_1}\left(\frac{1}{\rho}\frac{\partial p}{\partial x_1}\right)^n+u_2\frac{\Delta t}{2}\frac{\partial}{\partial x_2}\left(\frac{1}{\rho}\frac{\partial p}{\partial x_1}\right)^n \qquad (3.13)$$

$$\frac{u_2^{n+1}-\tilde{u}_2}{\Delta t}=-\frac{1}{\rho}\frac{\partial p^n}{\partial x_2}+u_1\frac{\Delta t}{2}\frac{\partial}{\partial x_1}\left(\frac{1}{\rho}\frac{\partial p}{\partial x_1}\right)^n+u_2\frac{\Delta t}{2}\frac{\partial}{\partial x_2}\left(\frac{1}{\rho}\frac{\partial p}{\partial x_2}\right)^n \qquad (3.14)$$

If pressure is calculated from another source, then the intermediate velocities of step 1 can be corrected using equations (3.13) and (3.14). To have an independent pressure equation to get the pressure to be substitute into the above equations, we need to eliminate u_1^{n+1} and u_2^{n+1}. This can be done by using continuity equation.

Differentiating eq.(3.13) with respect to x_1 and eq.(3.14) with respect to x_2 and neglecting the third order terms, we get

$$\frac{\partial u_1^{n+1}}{\partial x_1}+\frac{\partial u_2^{n+1}}{\partial x_2}-\frac{\partial \tilde{u}_1}{\partial x_1}-\frac{\partial \tilde{u}_2}{\partial x_2}=-\frac{\Delta t}{\rho}\left(\frac{\partial^2 p}{\partial x_1^2}+\frac{\partial^2 p}{\partial x_2^2}\right) \qquad (3.15)$$

Now from the continuity equation, we have

$$\frac{\partial u_1^{n+1}}{\partial x_1}+\frac{\partial u_2^{n+1}}{\partial x_2}=0 \qquad (3.16)$$

Equation (3.14) is then reduces to the pressure Poisson equation as

$$\frac{1}{\rho}\left(\frac{\partial^2 p}{\partial x_1^2}+\frac{\partial^2 p}{\partial x_2^2}\right)=\frac{1}{\Delta t}\left(\frac{\partial \tilde{u}_1}{\partial x_1}+\frac{\partial \tilde{u}_2}{\partial x_2}\right) \tag{3.17}$$

Thus from equation (3.17), we can calculate the pressure field using the intermediate velocities.

Step 3: Velocity or Momentum correction

The velocity correction involves the pressure and intermediate velocity field. It has been derived during calculation of pressure in step 2. Equations (3.13) and (3.14) are the velocity correction terms in terms of pressure values. The actual momentum or velocity after applying velocity correction is given by

Actual momentum or velocity = intermediate momentum velocity + pressure value

$$\frac{u_1^{n+1}-\tilde{u}_1}{\Delta t}=\frac{\tilde{u}_1-u_1^n}{\Delta t}-\frac{1}{\rho}\frac{\partial p^n}{\partial x_1} \tag{3.18}$$

$$\frac{u_2^{n+1}-\tilde{u}_2}{\Delta t}=\frac{\tilde{u}_2-u_2^n}{\Delta t}-\frac{1}{\rho}\frac{\partial p^n}{\partial x_2} \tag{3.19}$$

Step 4: Temperature Calculation

Any number of steps can be added to the first three steps if the quantity of the interest is a scalar such as temperature, concentration or turbulent transport quantities.

In this case temperature can be calculated from energy equation and we can write the temperature calculation as fourth step for the completeness of the Navier Stokes equations.

Applying the Characteristic Galerkin method to the energy equation,(3.6),

$$\frac{T^{n+1}-T^n}{\Delta t}=-u_1\frac{\partial T^n}{\partial x_1}-u_2\frac{\partial T^n}{\partial x_2}+\alpha\left(\frac{\partial^2 T}{\partial x_1^{\,2}}+\frac{\partial^2 T}{\partial x_2^{\,2}}\right)^n+u_1\frac{\Delta t}{2}\frac{\partial}{\partial x_1}\left[u_1\frac{\partial T}{\partial x_1}+u_2\frac{\partial T}{\partial x_2}\right]^n$$
$$+u_2\frac{\Delta t}{2}\frac{\partial}{\partial x_2}\left[u_1\frac{\partial T^n}{\partial x_1}+u_2\frac{\partial T^n}{\partial x_2}\right]^n+\frac{\eta\dot{\gamma}^2}{\rho C_p} \qquad (3.20)$$

Now the CBS scheme is complete with temporal discretization and it just needs spatial discretization to get the finite element solution. The above semi-discrete equations can now be approximated in space using the standard Galerkin finite element procedure.

3.3.1 Spatial Discretization

The computational domain is subdivided into a mesh of linear triangular elements. Within an element each variable is approximated by a linear function, which can be expressed in terms of the variable value at each of the three nodes of the element:

$$\phi=\sum_{n=1}^{3}N_n\phi_n \qquad (3.21)$$

where N_n are the shape functions at each node 'n ' and ϕ_n is the value of the generic unknown quantity ϕ (u_i, p and T) at the node 'n'. The 2D linear triangular element is shown in Figure 3.1.

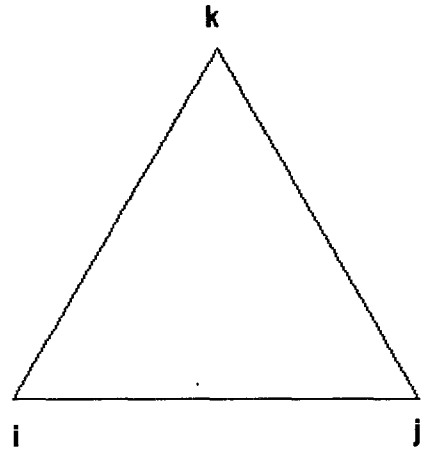


Figure 3.1 Linear triangular element used for 2D mold filling analysis

Using the Standard Galerkin procedure (Segerlind, 1984; Lewis et al., 1993), the weak form of the governing equations is obtained by weighting each of the above equations by the same shape functions introduced above. Shape functions are derived using the method explained in (Segerlind, 1984).

Step 1: Calculation of intermediate velocity

The weak form of the intermediate velocity equation x_1 component is given by

$$\int_\Omega [N]^T \frac{\tilde{u}_1 - u_1^n}{\Delta t} d\Omega = -\int_\Omega [N]^T u_1 \frac{\partial u_1^n}{\partial x_1} d\Omega - \int_\Omega [N]^T u_2 \frac{\partial u_1^n}{\partial x_2} d\Omega + \nu \int_\Omega [N]^T \left(\frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial^2 u_1}{\partial x_2^2} \right)^n d\Omega +$$
$$+ \int_\Omega [N]^T u_1 \frac{\Delta t}{2} \frac{\partial}{\partial x_1} \left[u_1 \frac{\partial u_1}{\partial x_1} + u_2 \frac{\partial u_1}{\partial x_2} \right]^n d\Omega + \qquad (3.22)$$
$$+ \int_\Omega [N]^T u_2 \frac{\Delta t}{2} \frac{\partial}{\partial x_2} \left[u_1 \frac{\partial u_1}{\partial x_1} + u_2 \frac{\partial u_1}{\partial x_2} \right]^n d\Omega$$

The weak form of the intermediate velocity equation x_2 component is given by

$$\int_{\Omega}[N]^T \frac{\tilde{u}_2 - u_2^n}{\Delta t} d\Omega = -\int_{\Omega}[N]^T u_1 \frac{\partial u_2^n}{\partial x_1} d\Omega - \int_{\Omega}[N]^T u_2 \frac{\partial u_2^n}{\partial x_2} d\Omega + \nu \int_{\Omega}[N]^T \left(\frac{\partial^2 u_2}{\partial x_1^2} + \frac{\partial^2 u_2}{\partial x_2^2} \right)^n d\Omega +$$

$$+ \int_{\Omega}[N]^T u_1 \frac{\Delta t}{2} \frac{\partial}{\partial x_1} \left[u_1 \frac{\partial u_2}{\partial x_1} + u_2 \frac{\partial u_2}{\partial x_2} \right]^n d\Omega + \qquad (3.23)$$

$$+ \int_{\Omega}[N]^T u_2 \frac{\Delta t}{2} \frac{\partial}{\partial x_2} \left[u_1 \frac{\partial u_2}{\partial x_1} + u_2 \frac{\partial u_2}{\partial x_2} \right]^n d\Omega$$

Now the integral of each term in above equations are obtained by using simple integral calculus rules.

$$\int_{\Omega}[N]^T \frac{\tilde{u}_1 - u_1^n}{\Delta t} d\Omega = \int_{\Omega}[N]^T [N] \frac{\{\tilde{u}_1\} - \{u_1\}^n}{\Delta t} d\Omega = [M] \frac{\Delta \{\tilde{u}_1\}}{\Delta t}$$

Where $[M] = \int_{\Omega}[N]^T [N] d\Omega =$ Element mass matrix (3.24)

$$\int_{\Omega}[N]^T u_1 \frac{\partial u_1^n}{\partial x_1} d\Omega + \int_{\Omega}[N]^T u_2 \frac{\partial u_1^n}{\partial x_2} d\Omega$$

$$= \int_{\Omega}[N]^T [N] \{u_1\} \frac{\partial [N]}{\partial x_1} \{u_1\}^n d\Omega + \int_{\Omega}[N]^T [N] \{u_2\} \frac{\partial [N]}{\partial x_2} \{u_1\}^n d\Omega$$

$$= [C] \{u_1\}^n$$

Where [C] is called 'Element Convection Matrix' and is given by

$$[C] = \left[\int_{\Omega}[N]^T [N] \{u_1\} \frac{\partial [N]}{\partial x_1} d\Omega + \int_{\Omega}[N]^T [N] \{u_2\} \frac{\partial [N]}{\partial x_2} d\Omega \right] \qquad (3.25)$$

Now consider the integration of second order terms. They can be evaluated by using Green's theorem.

$$\nu \int_{\Omega} [N]^T \left(\frac{\partial^2 u_1}{\partial x_1^{\,2}} + \frac{\partial^2 u_1}{\partial x_2^{\,2}} \right)^n d\Omega = -\nu \int_{\Omega} \frac{\partial [N]^T}{\partial x_1} \frac{\partial u_1^n}{\partial x_1} d\Omega + \nu \int_{\Gamma} [N]^T \frac{\partial u_1^n}{\partial x_1} n_1 d\Gamma$$
$$- \nu \int_{\Omega} \frac{\partial [N]^T}{\partial x_2} \frac{\partial u_1^n}{\partial x_2} d\Omega + \nu \int_{\Gamma} [N]^T \frac{\partial u_1^n}{\partial x_2} n_2 d\Gamma \qquad (3.26)$$

Surface Integrals constitute the diffusion term while line integrals corresponds to the forcing terms.

$$\text{Momentum Diffusion Term} = \nu \int_{\Omega} \frac{\partial [N]^T}{\partial x_1} \frac{\partial u_1^n}{\partial x_1} d\Omega + \int_{\Omega} \frac{\partial [N]^T}{\partial x_2} \frac{\partial u_1^n}{\partial x_2} d\Omega$$

$$= \left[\nu \int_{\Omega} \frac{\partial [N]^T}{\partial x_1} \frac{\partial [N]}{\partial x_1} d\Omega + \nu \int_{\Omega} \frac{\partial [N]^T}{\partial x_2} \frac{\partial [N]}{\partial x_2} d\Omega \right] \{u_1\}^n$$

$$= [K_m] \{u_1\}^n$$

where $[K_m]$ is the element diffusion matrix and is given by

$$[K_m] = \left[\nu \int_{\Omega} \frac{\partial [N]^T}{\partial x_1} \frac{\partial [N]}{\partial x_1} d\Omega + \nu \int_{\Omega} \frac{\partial [N]^T}{\partial x_2} \frac{\partial [N]}{\partial x_2} d\Omega \right] \qquad (3.27)$$

The forcing term or forcing vector is given by

$$f_1 = \nu \int_{\Gamma} [N]^T \frac{\partial u_1^n}{\partial x_1} n_1 d\Gamma + \nu \int_{\Gamma} [N]^T \frac{\partial u_1^n}{\partial x_2} n_2 d\Gamma \qquad (3.28)$$

where n_1 and n_2 are direction cosines of the outward normal **n**, Γ is the domain boundary of the domain Ω.

The integration of last two terms in equation (21) is given by

$$\int_\Omega [N]^T u_1 \frac{\Delta t}{2}\frac{\partial}{\partial x_1}\left[u_1\frac{\partial u_1}{\partial x_1}+u_2\frac{\partial u_1}{\partial x_2}\right]^n d\Omega + \int_\Omega [N]^T u_2 \frac{\Delta t}{2}\frac{\partial}{\partial x_2}\left[u_1\frac{\partial u_1}{\partial x_1}+u_2\frac{\partial u_1}{\partial x_2}\right]^n d\Omega$$

$$= \frac{\Delta t}{2}\left[\int_\Omega [N]^T[N]\{u_1\}\frac{\partial}{\partial x_1}\left[[N]\{u_1\}\frac{\partial [N]}{\partial x_1}+[N]\{u_2\}\frac{\partial [N]}{\partial x_2}\right]d\Omega\right]\{u_1\}^n +$$

$$+\frac{\Delta t}{2}\left[\int_\Omega [N]^T[N]\{u_2\}\frac{\partial}{\partial x_2}\left[[N]\{u_1\}\frac{\partial [N]}{\partial x_1}+[N]\{u_2\}\frac{\partial [N]}{\partial x_2}\right]d\Omega\right]\{u_1\}^n$$

$$= [K_s]\{u_1\}^n$$

$[K_s]$ is the element stabilization matrix and is given by

$$[K_s] = \frac{\Delta t}{2}\left[\int_\Omega [N]^T[N]\{u_1\}\frac{\partial}{\partial x_1}\left[[N]\{u_1\}\frac{\partial [N]}{\partial x_1}+[N]\{u_2\}\frac{\partial [N]}{\partial x_2}\right]d\Omega\right] + \frac{\Delta t}{2}\left[\int_\Omega [N]^T[N]\{u_2\}\frac{\partial}{\partial x_2}\left[[N]\{u_1\}\frac{\partial [N]}{\partial x_1}+[N]\{u_2\}\frac{\partial [N]}{\partial x_2}\right]d\Omega\right] \quad (3.29)$$

The element convection matrix, diffusion matrix, stabilization matrix and forcing vector are assembled to get assembled matrices.

Now the above step 1 of the CBS scheme can be written in matrix form as below.

$$[M]\frac{\Delta\{\tilde{u}_1\}}{\Delta t} = -[C]\{\tilde{u}_1\}^n - [K_m]\{\tilde{u}_1\}^n - [K_s]\{\tilde{u}_1\}^n + \{f_1\} \quad (3.30)$$

Following the same procedure for the intermediate velocity equation x_2, we get

$$[M]\frac{\Delta\{\tilde{u}_2\}}{\Delta t} = -[C]\{\tilde{u}_2\}^n - [K_m]\{\tilde{u}_2\}^n - [K_s]\{\tilde{u}_2\}^n + \{f_2\} \quad (3.31)$$

where

$$\Delta\{\tilde{u}_1\} = \{\tilde{u}_1\} - \{u_1\}^n \quad (3.32)$$

$$\Delta\{\tilde{u}_2\} = \{\tilde{u}_2\} - \{u_2\}^n \quad (3.33)$$

the Mass matrix [M], the convection matrix [C], the momentum diffusion matrix [K_m] and the stabilization matrix [K_s] are given by eq. (3.24), (3.25),(3.27)and (3.29) respectively.

The forcing vector f_2 is given by

$$f_2 = \nu \int_\Gamma [N]^T \frac{\partial u_2^n}{\partial x_1} n_1 d\Gamma + \nu \int_\Gamma [N]^T \frac{\partial u_2^n}{\partial x_2} n_2 d\Gamma \tag{3.34}$$

Note that the value of the forcing vectors f_1 and f_2 depends on the boundary edge of the element selected during the calculation.

Step 2: Calculation of Pressure

The weak form of the pressure equation is given by

$$\int_\Omega [N]^T \frac{1}{\rho}\left(\frac{\partial^2 p}{\partial x_1^2} + \frac{\partial^2 p}{\partial x_2^2}\right)^n d\Omega = \frac{1}{\Delta t}\int_\Omega [N]^T \left(\frac{\partial \tilde{u}_1}{\partial x_1} + \frac{\partial \tilde{u}_2}{\partial x_2}\right) d\Omega \tag{3.35}$$

Applying Green's theorem to the second order terms, we get

$$\int_\Omega [N]^T \frac{1}{\rho}\left(\frac{\partial^2 p}{\partial {x_1}^2} + \frac{\partial^2 p}{\partial {x_2}^2}\right)^n d\Omega = -\frac{1}{\rho}\int_\Omega \frac{\partial [N]^T}{\partial x_1}\frac{\partial p^n}{\partial x_1} d\Omega + \frac{1}{\rho}\int_\Gamma [N]^T \frac{\partial p^n}{\partial x_1} n_1 d\Gamma$$
$$-\frac{1}{\rho}\int_\Omega \frac{\partial [N]^T}{\partial x_2}\frac{\partial p^n}{\partial x_2} d\Omega + \frac{1}{\rho}\int_\Gamma [N]^T \frac{\partial p^n}{\partial x_2} n_2 d\Gamma$$

$$\text{Surface integral terms} = -\frac{1}{\rho}\int_\Omega \frac{\partial [N]^T}{\partial x_1}\frac{\partial p^n}{\partial x_1} d\Omega - \frac{1}{\rho}\int_\Omega \frac{\partial [N]^T}{\partial x_2}\frac{\partial p^n}{\partial x_2} d\Omega$$

$$= -\left[\frac{1}{\rho}\int_\Omega \frac{\partial [N]^T}{\partial x_1}\frac{\partial [N]}{\partial x_1} d\Omega + \frac{1}{\rho}\int_\Omega \frac{\partial [N]^T}{\partial x_2}\frac{\partial [N]}{\partial x_2} d\Omega\right]\{p\}^n$$

$$= -[K]\{p\}^n$$

where K- matrix is given by

$$K=-\left[\frac{1}{\rho}\int_{\Omega}\frac{\partial[N]^T}{\partial x_1}\frac{\partial[N]}{\partial x_1}d\Omega+\frac{1}{\rho}\int_{\Omega}\frac{\partial[N]^T}{\partial x_2}\frac{\partial[N]}{\partial x_2}d\Omega\right] \tag{3.36}$$

The forcing vector due to discretization of second order pressure terms is given by

$$\{f_3\}=\left(\frac{1}{\rho}\int_{\Gamma}[N]^T\frac{\partial[N\}}{\partial x_1}n_1 d\Gamma+\frac{1}{\rho}\int_{\Gamma}[N]^T\frac{\partial[N]}{\partial x_2}n_2 d\Gamma\right)\{p\}^n \tag{3.37}$$

Consider RHS of equation (3.35).

$$\frac{1}{\Delta t}\int_{\Omega}[N]^T\left(\frac{\partial\tilde{u}_1}{\partial x_1}+\frac{\partial\tilde{u}_2}{\partial x_2}\right)d\Omega.=\frac{1}{\Delta t}\int_{\Omega}[N]^T\frac{\partial\tilde{u}_1}{\partial x_1}d\Omega.+\frac{1}{\Delta t}\int_{\Omega}[N]^T\frac{\partial\tilde{u}_2}{\partial x_2}d\Omega.$$

$$=\frac{1}{\Delta t}\left[\int_{\Omega}[N]^T\frac{\partial[N]}{\partial x_1}d\Omega\right]\{\tilde{u}_1\}+\frac{1}{\Delta t}\left[\int_{\Omega}[N]^T\frac{\partial[N\}}{\partial x_2}d\Omega\right]\{\tilde{u}_2\}$$

$$=\frac{1}{\Delta t}[[G_1]\{\tilde{u}_1\}+[G_2]\{\tilde{u}_2\}]$$

where $[G_1]$ and $[G_2]$ are Gradient matrices. They are given by the expressions

$$[G_1]=\int_{\Omega}[N]^T\frac{\partial[N]}{\partial x_1}d\Omega \tag{3.38}$$

$$[G_2]=\int_{\Omega}[N]^T\frac{\partial[N]}{\partial x_2}d\Omega \tag{3.39}$$

The second step of CBS in which pressure is calculated can then be written as

$$[K]\{p\}^n=\frac{-1}{\Delta t}[[G_1]\{\tilde{u}_1\}+[G_2]\{\tilde{u}_2\}]+\{f_3\} \tag{3.40}$$

Step 3: Velocity Correction

Applying Galerkin Weighting method to eq.(3.18) , we get

$$\int_{\Omega}[N]^T\frac{u_1^{n+1}-\tilde{u}_1}{\Delta t}d\Omega=\int_{\Omega}[N]^T\frac{\tilde{u}_1-u_1^n}{\Delta t}d\Omega-\int_{\Omega}[N]^T\frac{1}{\rho}\frac{\partial p^n}{\partial x_1}d\Omega$$

$$\Rightarrow \int_{\Omega}[N]^T[N]\frac{\{u_1\}^{n+1}-\{\tilde{u}_1\}}{\Delta t}d\Omega = \int_{\Omega}[N]^T[N]\frac{\{\tilde{u}_1\}-\{u_1\}^n}{\Delta t}d\Omega - \frac{1}{\rho}\int_{\Omega}[N]^T\frac{\partial[N]}{\partial x_1}\{p\}^n d\Omega$$

$$\Rightarrow [M]\frac{\Delta\{u_1\}^{n+1}}{\Delta t} = [M]\frac{\Delta\{\tilde{u}_1\}}{\Delta t} - \frac{1}{\rho}[G_1]\{p\}^n$$

$$[M]\Delta\{u_1\}^{n+1} = [M]\Delta\{\tilde{u}_1\} - \frac{1}{\rho}\Delta t[G_1]\{p\}^n \tag{3.41}$$

where $\Delta\{u_1\}^{n+1} = \{u_1\}^{n+1} - \{\tilde{u}_1\}$ (3.42)

[M], [G_1] are mass matrix and Gradient matrix and are given by Eq. (3.24) and (3.38) respectively.

Similarly for x_2 momentum, after applying Galerkin Method to eq.(3.19), we get

$$[M]\Delta\{u_2\}^{n+1} = [M]\Delta\{\tilde{u}_2\} - \frac{1}{\rho}\Delta t[G_2]\{p\}^n \tag{3.43}$$

where $\Delta\{u_2\}^{n+1} = \{u_2\}^{n+1} - \{\tilde{u}_2\}$ (3.44)

[M], [G_2] are mass matrix and Gradient matrix and are given by Equations (3.24) and (3.39) respectively.

Step 4: Calculation of Temperature

Following the same treatment as that for step1, we get the following equation to calculate the temperature field.

$$[M]\frac{\Delta\{T\}}{\Delta t} = -[C]\{T\}^n - [K_t]\{T\}^n - [K_s]\{T\}^n + \{f_4\} + R \tag{3.44}$$

Where heat diffusion matrix [K_t] and forcing vector {f_4} are given by

$$[K_t] = \left[\alpha\int_{\Omega}\frac{\partial[N]^T}{\partial x_1}\frac{\partial[N]}{\partial x_1}d\Omega + \alpha\int_{\Omega}\frac{\partial[N]^T}{\partial x_2}\frac{\partial[N]}{\partial x_2}d\Omega\right] \tag{3.45}$$

$$\{f_4\}=\left(\alpha\int_\Gamma[N]^T\frac{\partial[N\}}{\partial x_1}n_1 d\Gamma+\alpha\int_\Gamma[N]^T\frac{\partial[N]}{\partial x_2}n_2 d\Gamma\right)\{T\}^n \tag{3.46}$$

$$R=\int_\Omega\frac{\eta\dot{\gamma}^2}{\rho C_p}d\Omega \quad = \text{Rheology matrix} \tag{3.47}$$

Now we can summarize the fundamental governing equations by four steps of CBS method as below.

Step 1: Intermediate Velocity Calculation

x_1-Component:

$$[M]\frac{\Delta\{\tilde{u}_1\}}{\Delta t}=-[C]\{\tilde{u}_1\}^n-[K_m]\{\tilde{u}_1\}^n-[K_s]\{\tilde{u}_1\}^n+\{f_1\}$$

x_2-Component:

$$[M]\frac{\Delta\{\tilde{u}_2\}}{\Delta t}=-[C]\{\tilde{u}_2\}^n-[K_m]\{\tilde{u}_2\}^n-[K_s]\{\tilde{u}_2\}^n+\{f_2\}$$

Step 2: Pressure Calculation

$$[K]\{p\}^n=\frac{-1}{\Delta t}[[G_1]\{\tilde{u}_1\}+[G_2]\{\tilde{u}_2\}]+\{f_3\}$$

Step 3: Velocity Correction

$$[M]\Delta\{u_1\}^{n+1}=[M]\Delta\{\tilde{u}_1\}-\frac{1}{\rho}\Delta t[G_1]\{p\}^n$$

$$[M]\Delta\{u_2\}^{n+1}=[M]\Delta\{\tilde{u}_2\}-\frac{1}{\rho}\Delta t[G_2]\{p\}^n$$

Step 4: Temperature Calculation

$$[M]\frac{\Delta\{T\}}{\Delta t} = -[C]\{T\}^n - [K_t]\{T\}^n - [K_s]\{T\}^n + \{f_4\} + R$$

The above four steps are the cornerstones of the CBS scheme for the solution of heat convection equations. Note that extending the above steps to conservation form and three dimensions is straight forward (Lewis et al., 2004). One can refer the papers (Zienkiewicz et al.,1999; Nithiarasu, 2003) for further insight of the procedure followed.

Various element matrices used in above equations are given by the following expressions.

$$[M] = \text{Mass Matrix} = \frac{A}{12}\begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} \tag{3.48}$$

The mass matrix may be lumped to simplify the solution procedure and also to save the calculation time. The lumped mass matrix for a linear triangular element is constructed by summing the rows and putting the sum along principle diagonal, while zeros being all other elements. Thus lumped mass matrix is given by

$$[M] = \text{Lumped Mass Matrix} = \frac{A}{3}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.49}$$

$$\text{Where } A = \text{Area of the TraingularElement} = \frac{1}{2}\begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix} \tag{3.50}$$

[C] = Convection Matrix

$$= \frac{1}{24}\begin{bmatrix} (usu+u_{1i})b_i & (usu+u_{1i})b_j & (usu+u_{1i})b_k \\ (usu+u_{1j})b_i & (usu+u_{1j})b_j & (usu+u_{1j})b_k \\ (usu+u_{1k})b_i & (usu+u_{1k})b_j & (usu+u_{1k})b_k \end{bmatrix} + \tag{3.51}$$

$$+\frac{1}{24}\begin{bmatrix} (vsu+u_{2i})c_i & (vsu+u_{2i})c_j & (vsu+u_{2i})c_k \\ (vsu+u_{2j})c_i & (vsu+u_{2j})c_j & (vsu+u_{2j})c_k \\ (vsu+u_{2k})c_i & (vsu+u_{2k})c_j & (vsu+u_{2k})c_k \end{bmatrix}$$

Where

$$\left.\begin{aligned} usu &= u_{1i}+u_{1j}+u_{1k} \\ vsu &= u_{2i}+u_{2j}+u_{2k} \end{aligned}\right\} \tag{3.52}$$

$[K_m]$ = Momentum Diffusion Matrix

$$= \frac{\nu}{4A}\begin{bmatrix} b_i^2 & b_ib_j & b_ib_k \\ b_ib_j & b_j^2 & b_jb_k \\ b_ib_k & b_jb_k & b_k^2 \end{bmatrix} + \frac{\nu}{4A}\begin{bmatrix} c_i^2 & c_ic_j & c_ic_k \\ c_ic_j & c_j^2 & c_jc_k \\ c_ic_k & c_jc_k & c_k^2 \end{bmatrix} \tag{3.53}$$

$[K_t]$ = Heat Diffusion Matrix

$$= \frac{\alpha}{4A}\begin{bmatrix} b_i^2 & b_ib_j & b_ib_k \\ b_ib_j & b_j^2 & b_jb_k \\ b_ib_k & b_jb_k & b_k^2 \end{bmatrix} + \frac{\alpha}{4A}\begin{bmatrix} c_i^2 & c_ic_j & c_ic_k \\ c_ic_j & c_j^2 & c_jc_k \\ c_ic_k & c_jc_k & c_k^2 \end{bmatrix} \tag{3.54}$$

$[K_s]$ = Stabilisation Matrix

$$= \frac{\Delta t}{24}u_{1av}usu\begin{bmatrix} b_i^2 & b_ib_j & b_ib_k \\ b_ib_j & b_j^2 & b_jb_k \\ b_ib_k & b_jb_k & b_k^2 \end{bmatrix} + \frac{\Delta t}{24}u_{1av}vsu\begin{bmatrix} b_ic_i & b_ic_j & b_ic_k \\ b_jc_i & b_jc_j & b_jc_k \\ b_kc_i & b_kc_j & b_kc_k \end{bmatrix} + \tag{3.55}$$

$$+\frac{\Delta t}{24}u_{2av}usu\begin{bmatrix} c_ib_i & c_ib_j & c_ib_k \\ c_jb_i & c_jb_j & c_jb_k \\ c_kb_i & c_kb_j & c_kb_k \end{bmatrix} + \frac{\Delta t}{24}u_{2av}vsu\begin{bmatrix} c_i^2 & c_ic_j & c_ic_k \\ c_ic_j & c_j^2 & c_jc_k \\ c_ic_k & c_jc_k & c_k^2 \end{bmatrix}$$

$$[K] = \frac{1}{4A\rho}\begin{bmatrix} b_i^2 & b_ib_j & b_ib_k \\ b_ib_j & b_j^2 & b_jb_k \\ b_ib_k & b_jb_k & b_k^2 \end{bmatrix} + \frac{1}{4A\rho}\begin{bmatrix} c_i^2 & c_ic_j & c_ic_k \\ c_ic_j & c_j^2 & c_jc_k \\ c_ic_k & c_jc_k & c_k^2 \end{bmatrix} \tag{3.56}$$

$[G_1]$=First GradientMatrix in x_1 direction

$$= \frac{1}{6}\begin{bmatrix} b_i & b_j & b_k \\ b_i & b_j & b_k \\ b_i & b_j & b_k \end{bmatrix} \tag{3.57}$$

$[G_2]$=SecondGradientMatrixin x_2 direction

$$= \frac{1}{6}\begin{bmatrix} c_i & c_j & c_k \\ c_i & c_j & c_k \\ c_i & c_j & c_k \end{bmatrix} \tag{3.58}$$

The force vectors f_1, f_2, f_3 and f_4 based on boundary edge ij are written as

$$\{f_1\} = \frac{\Gamma}{4A}\nu\begin{bmatrix} b_i u_{1i} + b_j u_{1j} + b_k u_{1k} \\ b_i u_{1i} + b_j u_{1j} + b_k u_{1k} \\ 0 \end{bmatrix}^n n_1 + \frac{\Gamma}{4A}\nu\begin{bmatrix} c_i u_{1i} + c_j u_{1j} + c_k u_{1k} \\ c_i u_{1i} + c_j u_{1j} + c_k u_{1k} \\ 0 \end{bmatrix}^n n_2 \tag{3.59}$$

$$\{f_2\} = \frac{\Gamma}{4A}\nu\begin{bmatrix} b_i u_{2i} + b_j u_{2j} + b_k u_{2k} \\ b_i u_{2i} + b_j u_{2j} + b_k u_{2k} \\ 0 \end{bmatrix}^n n_1 + \frac{\Gamma}{4A}\nu\begin{bmatrix} c_i u_{2i} + c_j u_{2j} + c_k u_{2k} \\ c_i u_{2i} + c_j u_{2j} + c_k u_{2k} \\ 0 \end{bmatrix}^n n_2 \tag{3.60}$$

$$\{f_3\} = \frac{\Gamma}{4A\rho}\begin{bmatrix} b_i p_i + b_j p_j + b_k p_k \\ b_i p_i + b_j p_j + b_k p_k \\ 0 \end{bmatrix}^n n_1 + \frac{\Gamma}{4A}\begin{bmatrix} c_i p_i + c_j p_j + c_k p_k \\ c_i p_i + c_j p_j + c_k p_k \\ 0 \end{bmatrix}^n n_2 \tag{3.61}$$

$$\{f_4\} = \frac{\alpha\Gamma}{4A}\begin{bmatrix} b_i T_i + b_j T_j + b_k T_k \\ b_i T_i + b_j T_j + b_k T_k \\ 0 \end{bmatrix}^n n_1 + \frac{\alpha\Gamma}{4A}\begin{bmatrix} c_i T_i + c_j T_j + c_k T_k \\ c_i T_i + c_j T_j + c_k T_k \\ 0 \end{bmatrix}^n n_2 \tag{3.62}$$

3.3.2 Time Step Calculation

The proper time step has to be chosen to obtain a stable numerical solution. The stability analysis for a given time discretization gives some idea about the time step restrictions of a

numerical scheme. In other words, time step governs the stability of a scheme. Therefore it is necessary to choose proper time step so that the solution converges to a value fast.

In general for convection diffusion problems of fluid mechanics, time steps are controlled by two wave speeds namely convection velocity and the real diffusion velocity introduced by the equations. The convection velocity is given by $|u| = \sqrt{u_1^2 + u_2^2}$

(3.63)

The diffusion velocity is given by '2k/ h', where 'h' is local element size.

The local time steps at each and every node can be calculated as

$$\Delta t = \min(\Delta t_c, \Delta t_d) \tag{3.64}$$

where $\Delta t_c, \Delta t_d$ are the convection time step limit and diffusion time step limits respectively .

The convection time step is computed using the equation

$$\Delta t_c = \frac{h}{|u|} \tag{3.65}$$

The diffusion time step contains two parts, one due to the kinematic viscosity and other due to the thermal diffusivity of the fluid. The diffusion time step is expressed as

$$\Delta t = \min\left(\frac{h^2}{2\nu}, \frac{h^2}{2\alpha}\right) \tag{3.66}$$

where ν, α are the kinematic viscosity and the thermal diffusivity respectively.

3.4 3D Flow Analysis

Majority of the models for the analysis of mold filling during IC Chip encapsulation process use the simple models which are based on the Hele Shaw approximation. The Hele Shaw model neglects the inertia and lateral velocity component for the melt flow in thin cavities. In other words, the mold filling process is simplified to the planar melt flows on the mid-planes of top and bottom mold halves with respective thickness approximations (Chang et al., 2004). In

addition to above, the Hele Shaw model also neglects 'fountain flow' in the vicinity of the melt front advancement during the mold filling. The fountain flow effect has significant influence on the flow history of a particle as well as the degree of conversion of the molding compound (Chang et al., 2004). More over, the higher integration and performance requirement of current electronic devices has forced the semiconductor industry to go far 3-dimensional (3D) packaging technologies which utilize the third or Z height dimension to provide a volumetric packaging.

Therefore it is necessary to develop a three dimensional analysis model which can accurately predict the mold filing process during IC encapsulation. In this research work, 3D mold filling process has been analyzed using hybrid FEM/VOF method.
Flow field is calculated using the Characteristic Based Split finite element method and the fluid front is tracked using 'Volume of Fluid' technique.

Governing Navier Stokes equations are written as below.

(i) Continuity Equation:

$$\frac{\partial u_1}{\partial x_1}+\frac{\partial u_2}{\partial x_2}+\frac{\partial u_3}{\partial x_3}=0 \qquad (3.67)$$

where u_1,u_2 and u_3 are velocities along x_1, x_2 and x_3 directions.

*(ii)*Momentum Equations:

Momentum equations in non-conservative form can be written as

x_1-Momentum Equation:

$$\frac{\partial u_1}{\partial t}+u_1\frac{\partial u_1}{\partial x_1}+u_2\frac{\partial u_1}{\partial x_2}+u_3\frac{\partial u_1}{\partial x_3}=-\frac{1}{\rho}\frac{\partial p}{\partial x_1}+\nu\left(\frac{\partial^2 u_1}{\partial {x_1}^2}+\frac{\partial^2 u_1}{\partial {x_2}^2}+\frac{\partial^2 u_1}{\partial {x_3}^2}\right) \quad (3.68)$$

x_2-Momentum Equation:

$$\frac{\partial u_2}{\partial t}+u_1\frac{\partial u_2}{\partial x_1}+u_2\frac{\partial u_2}{\partial x_2}+u_3\frac{\partial u_2}{\partial x_3}=-\frac{1}{\rho}\frac{\partial p}{\partial x_2}+\nu\left(\frac{\partial^2 u_2}{\partial {x_1}^2}+\frac{\partial^2 u_2}{\partial {x_2}^2}+\frac{\partial^2 u_2}{\partial {x_3}^2}\right) \quad (3.69)$$

x_3-Momentum Equation:

$$\frac{\partial u_3}{\partial t}+u_1\frac{\partial u_3}{\partial x_1}+u_2\frac{\partial u_3}{\partial x_2}+u_3\frac{\partial u_3}{\partial x_3}=-\frac{1}{\rho}\frac{\partial p}{\partial x_3}+\nu\left(\frac{\partial^2 u_3}{\partial {x_1}^2}+\frac{\partial^2 u_3}{\partial {x_2}^2}+\frac{\partial^2 u_3}{\partial {x_3}^2}\right) \quad (3.70)$$

where ρ is density and ν is the kinematic viscosity of the fluid.

iii) Energy Equation:

$$\frac{\partial T}{\partial t}+u_1\frac{\partial T}{\partial x_1}+u_2\frac{\partial T}{\partial x_2}+u_3\frac{\partial T}{\partial x_3}=\alpha\left(\frac{\partial^2 T}{\partial {x_1}^2}+\frac{\partial^2 T}{\partial {x_2}^2}+\frac{\partial^2 T}{\partial {x_3}^2}\right)+\frac{\eta\dot{\gamma}^2}{\rho C_p} \quad (3.71)$$

where α is the thermal diffusivity, η, is viscosity, $\dot{\gamma}$ is shear rate and C_p is specific heat.

The CBS method is used to get the solution of above equations. The basic steps of CBS method are written below.

Step 1: Calculation of Intermediate Velocity or Momentum field

Momentum equations in semi-discrete are written as below.

Intermediate x_1 momentum equation:

$$\frac{\tilde{u}_1-u_1^n}{\Delta t}+u_1\frac{\partial u_1^n}{\partial x_1}+u_2\frac{\partial u_1^n}{\partial x_2}+u_3\frac{\partial u_1^n}{\partial x_3}=\nu\left(\frac{\partial^2 u_1}{\partial {x_1}^2}+\frac{\partial^2 u_1}{\partial {x_2}^2}+\frac{\partial^2 u_1}{\partial {x_3}^2}\right)^n \quad (3.72)$$

Intermediate x_2- momentum equation:

$$\frac{\tilde{u}_2 - u_2^n}{\Delta t} + u_1 \frac{\partial u_2^n}{\partial x_1} + u_2 \frac{\partial u_2^n}{\partial x_2} + u_3 \frac{\partial u_2^n}{\partial x_3} = \nu \left(\frac{\partial^2 u_2}{\partial x_1^{\,2}} + \frac{\partial^2 u_2}{\partial x_2^{\,2}} + \frac{\partial^2 u_2}{\partial x_3^{\,2}} \right)^n \tag{3.73}$$

Intermediate x_3- momentum equation:

$$\frac{\tilde{u}_3 - u_3^n}{\Delta t} + u_1 \frac{\partial u_3^n}{\partial x_1} + u_2 \frac{\partial u_3^n}{\partial x_2} + u_3 \frac{\partial u_3^n}{\partial x_3} = \nu \left(\frac{\partial^2 u_3}{\partial x_1^{\,2}} + \frac{\partial^2 u_3}{\partial x_2^{\,2}} + \frac{\partial^2 u_3}{\partial x_3^{\,2}} \right)^n \tag{3.74}$$

By applying Characteristic Galerkin method to above equations, we get

Intermediate x_1 momentum equation:

$$\begin{aligned} \frac{\tilde{u}_1 - u_1^n}{\Delta t} = & -u_1 \frac{\partial u_1^n}{\partial x_1} - u_2 \frac{\partial u_1^n}{\partial x_2} - u_3 \frac{\partial u_1^n}{\partial x_3} + \nu \left(\frac{\partial^2 u_1}{\partial x_1^{\,2}} + \frac{\partial^2 u_1}{\partial x_2^{\,2}} + \frac{\partial^2 u_1}{\partial x_3^{\,2}} \right)^n + \\ & + u_1 \frac{\Delta t}{2} \frac{\partial}{\partial x_1} \left[u_1 \frac{\partial u_1}{\partial x_1} + u_2 \frac{\partial u_1}{\partial x_2} + u_3 \frac{\partial u_1}{\partial x_3} \right]^n + \\ & + u_2 \frac{\Delta t}{2} \frac{\partial}{\partial x_2} \left[u_1 \frac{\partial u_1}{\partial x_1} + u_2 \frac{\partial u_1}{\partial x_2} + u_3 \frac{\partial u_1}{\partial x_3} \right]^n + \\ & + u_3 \frac{\Delta t}{2} \frac{\partial}{\partial x_3} \left[u_1 \frac{\partial u_1}{\partial x_1} + u_2 \frac{\partial u_1}{\partial x_2} + u_3 \frac{\partial u_1}{\partial x_3} \right]^n \end{aligned} \tag{3.75}$$

Intermediate x_2 momentum equation:

$$\frac{\tilde{u}_2-u_2^n}{\Delta t}=-u_1\frac{\partial u_2^n}{\partial x_1}-u_2\frac{\partial u_2^n}{\partial x_2}-u_3\frac{\partial u_2^n}{\partial x_3}+\nu\left(\frac{\partial^2 u_2}{\partial x_1^{\,2}}+\frac{\partial^2 u_2}{\partial x_2^{\,2}}+\frac{\partial^2 u_2}{\partial x_3^{\,2}}\right)^n+$$
$$+u_1\frac{\Delta t}{2}\frac{\partial}{\partial x_1}\left[u_1\frac{\partial u_2}{\partial x_1}+u_2\frac{\partial u_2}{\partial x_2}+u_3\frac{\partial u_2}{\partial x_3}\right]^n+$$
$$+u_2\frac{\Delta t}{2}\frac{\partial}{\partial x_2}\left[u_1\frac{\partial u_2}{\partial x_1}+u_2\frac{\partial u_2}{\partial x_2}+u_3\frac{\partial u_2}{\partial x_3}\right]^n+ \qquad (3.76)$$
$$+u_3\frac{\Delta t}{2}\frac{\partial}{\partial x_3}\left[u_1\frac{\partial u_2}{\partial x_1}+u_2\frac{\partial u_2}{\partial x_2}+u_3\frac{\partial u_2}{\partial x_3}\right]^n$$

Intermediate x_3 momentum equation:

$$\frac{\tilde{u}_3-u_3^n}{\Delta t}=-u_1\frac{\partial u_3^n}{\partial x_1}-u_2\frac{\partial u_3^n}{\partial x_2}-u_3\frac{\partial u_3^n}{\partial x_3}+\nu\left(\frac{\partial^2 u_3}{\partial x_1^{\,2}}+\frac{\partial^2 u_3}{\partial x_2^{\,2}}+\frac{\partial^2 u_3}{\partial x_3^{\,2}}\right)^n+$$
$$+u_1\frac{\Delta t}{2}\frac{\partial}{\partial x_1}\left[u_1\frac{\partial u_3}{\partial x_1}+u_2\frac{\partial u_3}{\partial x_2}+u_3\frac{\partial u_3}{\partial x_3}\right]^n+$$
$$+u_2\frac{\Delta t}{2}\frac{\partial}{\partial x_2}\left[u_1\frac{\partial u_3}{\partial x_1}+u_2\frac{\partial u_3}{\partial x_2}+u_3\frac{\partial u_3}{\partial x_3}\right]^n+ \qquad (3.77)$$
$$+u_3\frac{\Delta t}{2}\frac{\partial}{\partial x_3}\left[u_1\frac{\partial u_3}{\partial x_1}+u_2\frac{\partial u_3}{\partial x_2}+u_3\frac{\partial u_3}{\partial x_3}\right]^n$$

Step 2: Calculation of Pressure

In the second step of CBS scheme, the pressure is calculated from a pressure Poisson equation. Continuing on the same lines of the method used in case of 2D analysis for pressure calculation and after simplification, we get

$$\frac{1}{\rho}\left(\frac{\partial^2 p}{\partial x_1^2}+\frac{\partial^2 p}{\partial x_2^2}+\frac{\partial^2 p}{\partial x_3^2}\right)=\frac{1}{\Delta t}\left(\frac{\partial \tilde{u}_1}{\partial x_1}+\frac{\partial \tilde{u}_2}{\partial x_2}+\frac{\partial \tilde{u}_3}{\partial x_3}\right) \qquad (3.78)$$

Step 3: Velocity or Momentum correction

In the third step, the pseudo velocity or intermediate velocity is corrected using the pressure value calculated in step 2 to get the real or actual velocities of the fluid flow particles. The relevant expressions for the velocity correction are written as below.

$$\frac{u_1^{n+1}-\tilde{u}_1}{\Delta t}=\frac{\tilde{u}_1-u_1^n}{\Delta t}-\frac{1}{\rho}\frac{\partial p^n}{\partial x_1} \tag{3.79}$$

$$\frac{u_2^{n+1}-\tilde{u}_2}{\Delta t}=\frac{\tilde{u}_2-u_2^n}{\Delta t}-\frac{1}{\rho}\frac{\partial p^n}{\partial x_2} \tag{3.80}$$

$$\frac{u_2^{n+1}-\tilde{u}_2}{\Delta t}=\frac{\tilde{u}_2-u_2^n}{\Delta t}-\frac{1}{\rho}\frac{\partial p^n}{\partial x_2} \tag{3.81}$$

Step 4: Temperature Calculation

In the fourth step, the temperature is calculated using energy equation.

Applying the Characteristic Galerkin method to the energy equation (3.71), we get

$$\begin{aligned}\frac{T^{n+1}-T^n}{\Delta t}=&-u_1\frac{\partial T^n}{\partial x_1}-u_2\frac{\partial T^n}{\partial x_2}-u_3\frac{\partial T^n}{\partial x_2}+\alpha\left(\frac{\partial^2 T}{\partial {x_1}^2}+\frac{\partial^2 T}{\partial {x_2}^2}+\frac{\partial^2 T}{\partial {x_3}^2}\right)^n+\\&+u_1\frac{\Delta t}{2}\frac{\partial}{\partial x_1}\left[u_1\frac{\partial T}{\partial x_1}+u_2\frac{\partial T}{\partial x_2}+u_2\frac{\partial T}{\partial x_3}\right]^n+\\&+u_2\frac{\Delta t}{2}\frac{\partial}{\partial x_2}\left[u_1\frac{\partial T^n}{\partial x_1}+u_2\frac{\partial T^n}{\partial x_2}+u_2\frac{\partial T}{\partial x_3}\right]^n+\\&+u_3\frac{\Delta t}{2}\frac{\partial}{\partial x_3}\left[u_1\frac{\partial T^n}{\partial x_1}+u_2\frac{\partial T^n}{\partial x_2}+u_2\frac{\partial T}{\partial x_3}\right]^n+\frac{\eta\dot{\gamma}^2}{\rho C_p}\end{aligned} \tag{3.82}$$

3.4.1 Spatial Discretization in 3D space domain

Now the above equations are discretized in space using the Galerkin finite element method. The linear four nodded tetrahedron element shown in Fig.2. is used in the 3D domain discretization.

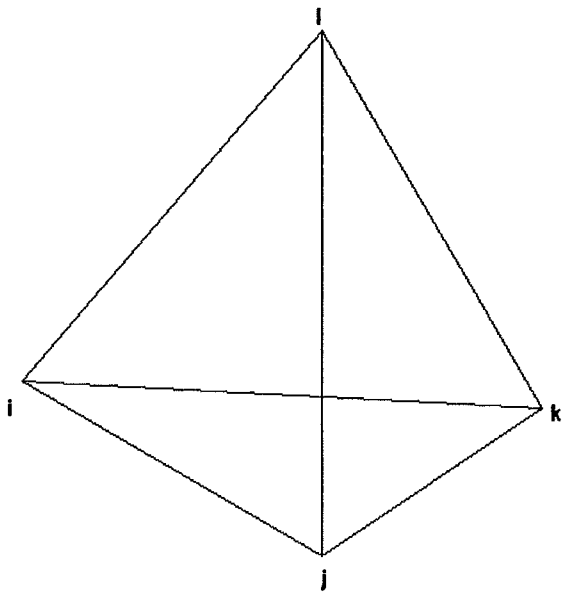


Fig. 3.2. Four noded tetrahedron used in 3D mold filling analysis.

The shape function variation for a four noded tetrahedron element can be written as

$$\phi = N_i\phi_i + N_j\phi_j + N_k\phi_k + N_l\phi_l \tag{3.83}$$

where N_i, N_j, N_k and N_l are the shape functions at each nodes i, j, k and k nodes respectively. ϕ_i, ϕ_j, ϕ_k and ϕ_l are the values of the quantity of interest like velocity 'u', pressure 'p' and temperature 'T' at the node 'i', 'j', 'k' and 'l' respectively.

Applying the Galerkin weighted residual method (Segerlind, 1984) to the above equations from (3.75) to (3.82), we get the weak form of the equations for four steps of the CBS Scheme as below.

Step 1: Calculation of intermediate velocity

X_1- intermediate velocity equation

$$\int_V [N]^T \frac{\tilde{u}_1 - u_1^n}{\Delta t} dV = - \int_V [N]^T u_1 \frac{\partial u_1^n}{\partial x_1} dV - \int_V [N]^T u_2 \frac{\partial u_1^n}{\partial x_2} dV - \int_V [N]^T u_3 \frac{\partial u_1^n}{\partial x_3} dV +$$
$$+ \nu \int_V [N]^T \left(\frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial^2 u_1}{\partial x_2^2} + \frac{\partial^2 u_1}{\partial x_3^2} \right)^n dV +$$
$$+ \int_V [N]^T u_1 \frac{\Delta t}{2} \frac{\partial}{\partial x_1} \left[u_1 \frac{\partial u_1}{\partial x_1} + u_2 \frac{\partial u_1}{\partial x_2} + u_3 \frac{\partial u_1}{\partial x_3} \right]^n dV + \quad (3.84)$$
$$+ \int_V [N]^T u_2 \frac{\Delta t}{2} \frac{\partial}{\partial x_2} \left[u_1 \frac{\partial u_1}{\partial x_1} + u_2 \frac{\partial u_1}{\partial x_2} + u_3 \frac{\partial u_1}{\partial x_3} \right]^n dV +$$
$$+ \int_V [N]^T u_3 \frac{\Delta t}{2} \frac{\partial}{\partial x_3} \left[u_1 \frac{\partial u_1}{\partial x_1} + u_2 \frac{\partial u_1}{\partial x_2} + u_3 \frac{\partial u_1}{\partial x_3} \right]^n dV$$

X_2 - intermediate velocity equation

$$\int_V [N]^T \frac{\tilde{u}_2 - u_2^n}{\Delta t} dV = - \int_V [N]^T u_1 \frac{\partial u_2^n}{\partial x_1} dV - \int_V [N]^T u_2 \frac{\partial u_2^n}{\partial x_2} dV - \int_V [N]^T u_3 \frac{\partial u_2^n}{\partial x_3} dV +$$
$$+ \nu \int_V [N]^T \left(\frac{\partial^2 u_2}{\partial x_1^2} + \frac{\partial^2 u_2}{\partial x_2^2} + \frac{\partial^2 u_2}{\partial x_3^2} \right)^n dV +$$
$$+ \int_V [N]^T u_1 \frac{\Delta t}{2} \frac{\partial}{\partial x_1} \left[u_1 \frac{\partial u_2}{\partial x_1} + u_2 \frac{\partial u_2}{\partial x_2} + u_3 \frac{\partial u_2}{\partial x_3} \right]^n dV + \quad (3.85)$$
$$+ \int_V [N]^T u_2 \frac{\Delta t}{2} \frac{\partial}{\partial x_2} \left[u_1 \frac{\partial u_2}{\partial x_1} + u_2 \frac{\partial u_2}{\partial x_2} + u_3 \frac{\partial u_2}{\partial x_3} \right]^n dV +$$
$$+ \int_V [N]^T u_3 \frac{\Delta t}{2} \frac{\partial}{\partial x_3} \left[u_1 \frac{\partial u_2}{\partial x_1} + u_2 \frac{\partial u_2}{\partial x_2} + u_3 \frac{\partial u_2}{\partial x_3} \right]^n dV$$

X_3 - intermediate velocity equation

$$\int_V [N]^T \frac{\tilde{u}_3 - u_3^n}{\Delta t} dV = - \int_V [N]^T u_1 \frac{\partial u_3^n}{\partial x_1} dV - \int_V [N]^T u_2 \frac{\partial u_3^n}{\partial x_2} dV - \int_V [N]^T u_3 \frac{\partial u_3^n}{\partial x_3} dV +$$
$$+ \nu \int_V [N]^T \left(\frac{\partial^2 u_3}{\partial x_1^2} + \frac{\partial^2 u_3}{\partial x_2^2} + \frac{\partial^2 u_3}{\partial x_3^2} \right)^n dV +$$
$$+ \int_V [N]^T u_1 \frac{\Delta t}{2} \frac{\partial}{\partial x_1} \left[u_1 \frac{\partial u_3}{\partial x_1} + u_2 \frac{\partial u_3}{\partial x_2} + u_3 \frac{\partial u_3}{\partial x_3} \right]^n dV +$$
$$+ \int_V [N]^T u_2 \frac{\Delta t}{2} \frac{\partial}{\partial x_2} \left[u_1 \frac{\partial u_3}{\partial x_1} + u_2 \frac{\partial u_3}{\partial x_2} + u_3 \frac{\partial u_3}{\partial x_3} \right]^n dV + \quad (3.86)$$
$$+ \int_V [N]^T u_3 \frac{\Delta t}{2} \frac{\partial}{\partial x_3} \left[u_1 \frac{\partial u_3}{\partial x_1} + u_2 \frac{\partial u_3}{\partial x_2} + u_3 \frac{\partial u_3}{\partial x_3} \right]^n dV$$

Now the integral of each term in above equations are obtained by using simple integral calculus rules.

$$\int_V [N]^T \frac{\tilde{u}_1 - u_1^n}{\Delta t} dV = \int_V [N]^T [N] \frac{\{\tilde{u}_1\} - \{u_1\}^n}{\Delta t} dV = [M] \frac{\Delta \{\tilde{u}_1\}}{\Delta t}$$

where [M] is a mass matrix and is given by the expression

$$[M] = \int_V [N]^T [N] \, dV \quad (3.87)$$

$$\int_V [N]^T u_1 \frac{\partial u_1^n}{\partial x_1} dV + \int_V [N]^T u_2 \frac{\partial u_1^n}{\partial x_2} dV + \int_V [N]^T u_3 \frac{\partial u_1^n}{\partial x_3} dV$$

$$= \int_V [N]^T [N]\{u_1\} \frac{\partial [N]}{\partial x_1} \{u_1\}^n dV + \int_V [N]^T [N]\{u_2\} \frac{\partial [N]}{\partial x_2} \{u_1\}^n dV + \int_V [N]^T [N]\{u_3\} \frac{\partial [N]}{\partial x_3} \{u_1\}^n dV$$

$$= [C]\{u_1\}^n$$

where [C] is called 'Element Convection Matrix' and is given by

$$[C] = \left[\int_V [N]^T [N]\{u_1\} \frac{\partial [N]}{\partial x_1} dV + \int_V [N]^T [N]\{u_2\} \frac{\partial [N]}{\partial x_2} dV + \int_V [N]^T [N]\{u_3\} \frac{\partial [N]}{\partial x_3} dV \right] \quad (3.88)$$

Now consider the integration of second order terms. They are evaluated by using Green's theorem.

$$\nu \int_V [N]^T \left(\frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial^2 u_1}{\partial x_2^2} + \frac{\partial^2 u_1}{\partial x_3^2} \right)^n dV = -\nu \int_V \frac{\partial [N]^T}{\partial x_1} \frac{\partial u_1^n}{\partial x_1} dV + \nu \int_A [N]^T \frac{\partial u_1^n}{\partial x_1} n_1 dA -$$

$$- \nu \int_V \frac{\partial [N]^T}{\partial x_2} \frac{\partial u_1^n}{\partial x_2} dV + \nu \int_A [N]^T \frac{\partial u_1^n}{\partial x_2} n_2 dA -$$

$$- \nu \int_V \frac{\partial [N]^T}{\partial x_3} \frac{\partial u_1^n}{\partial x_3} dV + \nu \int_A [N]^T \frac{\partial u_1^n}{\partial x_3} n_3 dA$$

Volume Integrals constitute the diffusion term while surface line integrals corresponds to the forcing terms.

$$\text{Momentum diffusion term} = \nu \int_V \frac{\partial [N]^T}{\partial x_1} \frac{\partial u_1^n}{\partial x_1} dV + \nu \int_V \frac{\partial [N]^T}{\partial x_2} \frac{\partial u_1^n}{\partial x_2} dV + \nu \int_V \frac{\partial [N]^T}{\partial x_3} \frac{\partial u_1^n}{\partial x_3} dV$$

$$= \nu \left[\int_V \frac{\partial [N]^T}{\partial x_1} \frac{\partial [N]}{\partial x_1} dV + \int_V \frac{\partial [N]^T}{\partial x_2} \frac{\partial [N]}{\partial x_2} dV + \int_V \frac{\partial [N]^T}{\partial x_3} \frac{\partial [N]}{\partial x_3} dV \right] \{u_1\}^n$$

$$= [K_m]\{u_1\}^n$$

where $[K_m]$ is the element diffusion matrix and is given by

$$[K_m]=\left[\nu\int_V\frac{\partial[N]^T}{\partial x_1}\frac{\partial[N]}{\partial x_1}dV+\nu\int_V\frac{\partial[N]^T}{\partial x_2}\frac{\partial[N]}{\partial x_2}dV+\nu\int_V\frac{\partial[N]^T}{\partial x_3}\frac{\partial u_1^n}{\partial x_3}dV\right] \quad (3.89)$$

Forcing term or forcing vector is given by

$$f_1=\nu\int_A[N]^T\frac{\partial u_1^n}{\partial x_1}n_1dA+\nu\int_A[N]^T\frac{\partial u_1^n}{\partial x_2}n_2dA+\nu\int_A[N]^T\frac{\partial u_1^n}{\partial x_3}n_3dA \quad (3.90)$$

where n_1, n_2 and n_3 are direction cosines of the outward normal **n**, 'A' is the boundary surface of the domain V .

The integration of last three terms in equation (3.84) is given by

$$\int_V[N]^T u_1\frac{\Delta t}{2}\frac{\partial}{\partial x_1}\left[u_1\frac{\partial u_1}{\partial x_1}+u_2\frac{\partial u_1}{\partial x_2}+u_3\frac{\partial u_1}{\partial x_3}\right]^n dV+$$

$$+\int_V[N]^T u_2\frac{\Delta t}{2}\frac{\partial}{\partial x_2}\left[u_1\frac{\partial u_1}{\partial x_1}+u_2\frac{\partial u_1}{\partial x_2}+u_3\frac{\partial u_1}{\partial x_3}\right]^n dV+$$

$$+\int_V[N]^T u_3\frac{\Delta t}{2}\frac{\partial}{\partial x_3}\left[u_1\frac{\partial u_1}{\partial x_1}+u_2\frac{\partial u_1}{\partial x_2}+u_3\frac{\partial u_1}{\partial x_3}\right]^n dV$$

$$=\frac{\Delta t}{2}\left[\int_V[N]^T[N]\{u_1\}\frac{\partial}{\partial x_1}\left[[N]\{u_1\}\frac{\partial[N]}{\partial x_1}+[N]\{u_2\}\frac{\partial[N]}{\partial x_2}+[N]\{u_3\}\frac{\partial[N]}{\partial x_3}\right]dV\right]\{u_1\}^n+$$

$$+\frac{\Delta t}{2}\left[\int_V[N]^T[N]\{u_2\}\frac{\partial}{\partial x_2}\left[[N]\{u_1\}\frac{\partial[N]}{\partial x_1}+[N]\{u_2\}\frac{\partial[N]}{\partial x_2}+[N]\{u_3\}\frac{\partial[N]}{\partial x_3}\right]dV\right]\{u_1\}^n+$$

$$+\frac{\Delta t}{2}\left[\int_V[N]^T[N]\{u_3\}\frac{\partial}{\partial x_3}\left[[N]\{u_1\}\frac{\partial[N]}{\partial x_1}+[N]\{u_2\}\frac{\partial[N]}{\partial x_2}+[N]\{u_3\}\frac{\partial[N]}{\partial x_3}\right]dV\right]\{u_1\}^n$$

$$=[K_s]\{u_1\}^n$$

$[K_s]$ is the element stabilization matrix and is given by

$$[K_s]=\frac{\Delta t}{2}\left[\int_V [N]^T[N]\{u_1\}\frac{\partial}{\partial x_1}\left[[N]\{u_1\}\frac{\partial[N]}{\partial x_1}+[N]\{u_2\}\frac{\partial[N]}{\partial x_2}+[N]\{u_3\}\frac{\partial[N]}{\partial x_3}\right]dV\right]+$$

$$+\frac{\Delta t}{2}\left[\int_V [N]^T[N]\{u_2\}\frac{\partial}{\partial x_2}\left[[N]\{u_1\}\frac{\partial[N]}{\partial x_1}+[N]\{u_2\}\frac{\partial[N]}{\partial x_2}+[N]\{u_3\}\frac{\partial[N]}{\partial x_3}\right]dV\right]+ \quad (3.91)$$

$$+\frac{\Delta t}{2}\left[\int_V [N]^T[N]\{u_2\}\frac{\partial}{\partial x_2}\left[[N]\{u_1\}\frac{\partial[N]}{\partial x_1}+[N]\{u_2\}\frac{\partial[N]}{\partial x_2}+[N]\{u_3\}\frac{\partial[N]}{\partial x_3}\right]dV\right]$$

The element convection matrix, diffusion matrix, stabilization matrix and forcing vector are assembled to get assembly matrices.

Now the above step 1 of the CBS scheme can be written in matrix form as below.

$$[M]\frac{\Delta\{\tilde{u}_1\}}{\Delta t}=-[C]\{\tilde{u}_1\}^n-[K_m]\{\tilde{u}_1\}^n-[K_s]\{\tilde{u}_1\}^n+\{f_1\} \quad (3.92)$$

Following the same procedure for x_2 and x_3 intermediate velocity equations, we get

$$[M]\frac{\Delta\{\tilde{u}_2\}}{\Delta t}=-[C]\{\tilde{u}_2\}^n-[K_m]\{\tilde{u}_2\}^n-[K_s]\{\tilde{u}_2\}^n+\{f_2\} \quad (3.93)$$

$$[M]\frac{\Delta\{\tilde{u}_3\}}{\Delta t}=-[C]\{\tilde{u}_3\}^n-[K_m]\{\tilde{u}_3\}^n-[K_s]\{\tilde{u}_3\}^n+\{f_3\} \quad (3.94)$$

where $\Delta\{\tilde{u}_1\},\Delta\{\tilde{u}_2\}$ have usual meaning and are given by equations (3.32) and (3.33) respectively. $\Delta\{\tilde{u}_3\}$ is given by

$$\Delta\{\tilde{u}_3\}=\{\tilde{u}_3\}-\{u_3\}^n \quad (3.95)$$

Mass matrix [M], convection matrix [C], momentum diffusion matrix $[K_m]$ and stabilization matrix $[K_s]$ are given by equations (3.87), (3.88), (3.89) and (3.91) respectively.

The forcing vector f_2 is given by

$$f_2 = \nu \int_A [N]^T \frac{\partial u_2^n}{\partial x_1} n_1 dA + \nu \int_A [N]^T \frac{\partial u_2^n}{\partial x_2} n_2 dA + \nu \int_A [N]^T \frac{\partial u_2^n}{\partial x_3} n_3 dA \qquad (3.96)$$

Similarly f_3 is expressed as

$$f_3 = \nu \int_A [N]^T \frac{\partial u_3^n}{\partial x_1} n_1 dA + \nu \int_A [N]^T \frac{\partial u_3^n}{\partial x_2} n_2 dA + \nu \int_A [N]^T \frac{\partial u_3^n}{\partial x_3} n_3 dA \qquad (3.97)$$

It is to be noted that the value of the forcing vectors f_1, f_2 and f_3 depend the boundary surface of the tetrahedron element selected during the calculation.

Step 2: Calculation of Pressure

The weak form of the pressure equation is given by

$$\int_V [N]^T \frac{1}{\rho}\left(\frac{\partial^2 p}{\partial x_1^2} + \frac{\partial^2 p}{\partial x_2^2} + \frac{\partial^2 p}{\partial x_3^2}\right)^n dV = \frac{1}{\Delta t} \int_V [N]^T \left(\frac{\partial \tilde{u}_1}{\partial x_1} + \frac{\partial \tilde{u}_2}{\partial x_2} + \frac{\partial \tilde{u}_3}{\partial x_3}\right) dV \qquad (3.98)$$

Applying Green's theorem to the second order terms, we get

$$\begin{aligned}\int_V [N]^T \frac{1}{\rho}\left(\frac{\partial^2 p}{\partial {x_1}^2} + \frac{\partial^2 p}{\partial {x_2}^2} + \frac{\partial^2 p}{\partial {x_3}^2}\right)^n dV = &-\frac{1}{\rho}\int_V \frac{\partial [N]^T}{\partial x_1}\frac{\partial p^n}{\partial x_1} dV + \frac{1}{\rho}\int_A [N]^T \frac{\partial p^n}{\partial x_1} n_1 dA - \\ &-\frac{1}{\rho}\int_V \frac{\partial [N]^T}{\partial x_2}\frac{\partial p^n}{\partial x_2} dV + \frac{1}{\rho}\int_A [N]^T \frac{\partial p^n}{\partial x_2} n_2 dA - \\ &-\frac{1}{\rho}\int_V \frac{\partial [N]^T}{\partial x_3}\frac{\partial p^n}{\partial x_3} dV + \frac{1}{\rho}\int_A [N]^T \frac{\partial p^n}{\partial x3} n_3 dA\end{aligned}$$

$$\begin{aligned}\text{Volume integral terms} &= -\frac{1}{\rho}\int_V \frac{\partial [N]^T}{\partial x_1}\frac{\partial p^n}{\partial x_1} dV - \frac{1}{\rho}\int_V \frac{\partial [N]^T}{\partial x_2}\frac{\partial p^n}{\partial x_2} dV - \frac{1}{\rho}\int_V \frac{\partial [N]^T}{\partial x_3}\frac{\partial p^n}{\partial x_3} dV \\ &= -\left[\frac{1}{\rho}\int_V \frac{\partial [N]^T}{\partial x_1}\frac{\partial [N]}{\partial x_1} dV + \right]\{p\}^n - \left[\frac{1}{\rho}\int_V \frac{\partial [N]^T}{\partial x_2}\frac{\partial [N]}{\partial x_2} dV\right]\{p\}^n - \\ &\quad -\left[\frac{1}{\rho}\int_V \frac{\partial [N]^T}{\partial x_3}\frac{\partial [N]}{\partial x_3} dV\right]\{p\}^n \\ &= -[K]\{p\}^n\end{aligned}$$

where the K- matrix is given by

$$K=-\left[\frac{1}{\rho}\int_V\frac{\partial[N]^T}{\partial x_1}\frac{\partial[N]}{\partial x_1}dV+\frac{1}{\rho}\int_V\frac{\partial[N]^T}{\partial x_2}\frac{\partial[N]}{\partial x_2}dV+\frac{1}{\rho}\int_V\frac{\partial[N]^T}{\partial x_3}\frac{\partial[N]}{\partial x_3}dV\right] \quad (3.99)$$

The forcing vector f_4 due to discretization of second order pressure terms is given by

$$\{f_4\}=\left(\frac{1}{\rho}\int_A[N]^T\frac{\partial[N\}}{\partial x_1}n_1dA+\frac{1}{\rho}\int_A[N]^T\frac{\partial[N]}{\partial x_2}n_2dA+\frac{1}{\rho}\int_A[N]^T\frac{\partial[N]}{\partial x_3}n_3dA\right)\{p\}^n \quad (3.100)$$

Consider RHS of equation (3.98).

$$\frac{1}{\Delta t}\int_V[N]^T\left(\frac{\partial\tilde{u}_1}{\partial x_1}+\frac{\partial\tilde{u}_2}{\partial x_2}+\frac{\partial\tilde{u}_3}{\partial x_3}\right)dV=\frac{1}{\Delta t}\int_V[N]^T\frac{\partial\tilde{u}_1}{\partial x_1}dV+\frac{1}{\Delta t}\int_V[N]^T\frac{\partial\tilde{u}_2}{\partial x_2}dV+\frac{1}{\Delta t}\int_V[N]^T\frac{\partial\tilde{u}_3}{\partial x_3}dV$$

$$=\frac{1}{\Delta t}\left[\int_V[N]^T\frac{\partial[N]}{\partial x_1}dV\right]\{\tilde{u}_1\}+\frac{1}{\Delta t}\left[\int_V[N]^T\frac{\partial[N\}}{\partial x_2}dV\right]\{\tilde{u}_2\}$$

$$+\frac{1}{\Delta t}\left[\int_V[N]^T\frac{\partial[N\}}{\partial x_3}dV\right]\{\tilde{u}_3\}$$

$$=\frac{1}{\Delta t}[[G_1]\{\tilde{u}_1\}+[G_2]\{\tilde{u}_2\}+[G_3]\{\tilde{u}_3\}]$$

where $[G_1]$, $[G_2]$and $[G_3]$ are gradient matrices and are given by the expressions

$$[G_1]=\int_V[N]^T\frac{\partial[N]}{\partial x_1}dV \quad (3.101)$$

$$[G_2]=\int_V[N]^T\frac{\partial[N]}{\partial x_2}dV \quad (3.102)$$

$$[G_3]=\int_V[N]^T\frac{\partial[N]}{\partial x_3}dV \quad (3.103)$$

The second step of CBS is then summarized as below.

$$[K]\{p\}^n = \frac{-1}{\Delta t}[[G_1]\{\tilde{u}_1\} + [G_2]\{\tilde{u}_2\} + [G_3]\{\tilde{u}_3\}] + \{f_4\} \quad (3.104)$$

Step 3: Velocity Correction

Applying Galerkin Weighting method to eq. (3.79) , we get

$$\int_V [N]^T \frac{u_1^{n+1} - \tilde{u}_1}{\Delta t} dV = \int_V [N]^T \frac{\tilde{u}_1 - u_1^n}{\Delta t} dV - \int_V [N]^T \frac{1}{\rho} \frac{\partial p^n}{\partial x_1} dV$$

$$\Rightarrow \int_V [N]^T [N] \frac{\{u_1\}^{n+1} - \{\tilde{u}_1\}}{\Delta t} dV = \int_V [N]^T [N] \frac{\{\tilde{u}_1\} - \{u_1\}^n}{\Delta t} dV - \frac{1}{\rho} \int_V [N]^T \frac{\partial [N]}{\partial x_1} \{p\}^n dV$$

$$\Rightarrow [M] \frac{\Delta\{u_1\}^{n+1}}{\Delta t} = [M] \frac{\Delta\{\tilde{u}_1\}}{\Delta t} - \frac{1}{\rho}[G_1]\{p\}^n$$

$$[M]\Delta\{u_1\}^{n+1} = [M]\Delta\{\tilde{u}_1\} - \frac{1}{\rho}\Delta t[G_1]\{p\}^n \quad (3.105)$$

[M], $[G_1]$ are mass matrix and gradient matrix and are given by eq. (3.89 and (3.101).

Similarly for x_2 momentum and x_3-momentum corrections are calculated by using Galerkin Method to eq.(3.88 and(3.89), we get

$$[M]\Delta\{u_2\}^{n+1} = [M]\Delta\{\tilde{u}_2\} - \frac{1}{\rho}\Delta t[G_2]\{p\}^n \quad (3.106)$$

$$[M]\Delta\{u_3\}^{n+1} = [M]\Delta\{\tilde{u}_3\} - \frac{1}{\rho}\Delta t[G_3]\{p\}^n \quad (3.107)$$

where

$$\Delta\{u_3\}^{n+1} = \{u_3\}^{n+1} - \{\tilde{u}_3\} \quad (3.108)$$

[M] is mass matrix, $[G_2]$ and $[G_3]$ are gradient matrices respectively and are given by equations (3.87), (3.102) and (3.103).

Step 4: Calculation of Temperature

Following the same procedure as that for step 1, we the following equation to calculate the temperature field.

$$[M]\frac{\Delta\{T\}}{\Delta t} = -[C]\{T\}^n - [K_t]\{T\}^n - [K_s]\{T\}^n + \{f_5\} \tag{3.109}$$

where heat diffusion matrix $[K_t]$ and forcing vector $\{f_5\}$ are given by

$$[K_t] = \alpha\left[\int_V \frac{\partial[N]^T}{\partial x_1}\frac{\partial[N]}{\partial x_1}dV + \int_V \frac{\partial[N]^T}{\partial x_2}\frac{\partial[N]}{\partial x_2}dV + \int_V \frac{\partial[N]^T}{\partial x_3}\frac{\partial[N]}{\partial x_3}dV\right] \tag{3.110}$$

$$\{f_5\} = \left(\alpha\int_A [N]^T\frac{\partial[N]}{\partial x_1}n_1 dA + \alpha\int_A [N]^T\frac{\partial[N]}{\partial x_2}n_2 dA + \alpha\int_A [N]^T\frac{\partial[N]}{\partial x_3}n_3 dA\right)\{T\}^n \tag{3.111}$$

Now, we summarize the four steps of CBS method for 3D flow analysis as below.

Step 1: Intermediate Velocity Calculation

x_1-Component:

$$[M]\frac{\Delta\{\tilde{u}_1\}}{\Delta t} = -[C]\{\tilde{u}_1\}^n - [K_m]\{\tilde{u}_1\}^n - [K_s]\{\tilde{u}_1\}^n + \{f_1\}$$

x_2-Component:

$$[M]\frac{\Delta\{\tilde{u}_2\}}{\Delta t} = -[C]\{\tilde{u}_2\}^n - [K_m]\{\tilde{u}_2\}^n - [K_s]\{\tilde{u}_2\}^n + \{f_2\}$$

x_3-Component:

$$[M]\frac{\Delta\{\tilde{u}_3\}}{\Delta t} = -[C]\{\tilde{u}_3\}^n - [K_m]\{\tilde{u}_3\}^n - [K_s]\{\tilde{u}_3\}^n + \{f_3\}$$

Step 2: Pressure Calculation

$$[K]\{p\}^n = -\frac{1}{\Delta t}[[G_1]\{\tilde{u}_1\} + [G_2]\{\tilde{u}_2\} + [G_3]\{\tilde{u}_3\}] + \{f_4\}$$

Step 3: Velocity Correction

$$[M]\Delta\{u_1\}^{n+1} = [M]\Delta\{\tilde{u}_1\} - \frac{1}{\rho}\Delta t[G_1]\{p\}^n$$

$$[M]\Delta\{u_2\}^{n+1} = [M]\Delta\{\tilde{u}_2\} - \frac{1}{\rho}\Delta t[G_2]\{p\}^n$$

$$[M]\Delta\{u_3\}^{n+1} = [M]\Delta\{\tilde{u}_3\} - \frac{1}{\rho}\Delta t[G_3]\{p\}^n$$

Step 4: Temperature Calculation

$$[M]\frac{\Delta\{T\}}{\Delta t} = -[C]\{T\}^n - [K_t]\{T\}^n - [K_s]\{T\}^n + \{f_5\}$$

The above four steps are the backbones of the CBS scheme for the solution of three dimensional equations.

Various element matrices used in above equations are given by the following expressions.

$$[M] = \text{Mass Matrix} = \frac{V}{20}\begin{bmatrix} 2 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 \\ 1 & 1 & 2 & 1 \\ 1 & 1 & 1 & 2 \end{bmatrix} \qquad (3.112)$$

The mass matrix may be lumped to simplify the solution procedure and hence to save the calculation time. Thus the lumped mass matrix is given by

$$[M] = \text{Lumped Mass Matrix} = \frac{V}{4}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (3.113)$$

The volume of the tetrahedron is expressed as

$$V=\frac{1}{6}\begin{bmatrix}1 & x_1 & y_1 & z_1\\ 1 & x_2 & y_2 & z_2\\ 1 & x_3 & y_3 & z_3\\ 1 & x_4 & y_4 & z_4\end{bmatrix} \qquad (3.114)$$

[C] = Convection Matrix

$$=\frac{1}{120}\begin{bmatrix}(usu+u_{1i})b_i & (usu+u_{1i})b_j & (usu+u_{1i})b_k & (usu+u_{1i})b_l\\ (usu+u_{1j})b_i & (usu+u_{1j})b_j & (usu+u_{1j})b_k & (usu+u_{1j})b_l\\ (usu+u_{1k})b_i & (usu+u_{1k})b_j & (usu+u_{1k})b_k & (usu+u_{1k})b_l\\ (usu+u_{1l})b_i & (usu+u_{1l})b_j & (usu+u_{1l})b_k & (usu+u_{1l})b_l\end{bmatrix}+$$

$$+\frac{1}{120}\begin{bmatrix}(vsu+u_{2i})c_i & (vsu+u_{2i})c_j & (vsu+u_{2i})c_k & (usu+u_{2i})c_l\\ (vsu+u_{2j})c_i & (vsu+u_{2j})c_j & (vsu+u_{2j})c_k & (usu+u_{2j})c_l\\ (vsu+u_{2k})c_i & (vsu+u_{2k})c_j & (vsu+u_{2k})c_k & (usu+u_{2k})c_l\\ (usu+u_{2l})c_i & (usu+u_{2l})c_j & (usu+u_{2l})c_k & (usu+u_{2l})c_l\end{bmatrix}+ \qquad (3.115)$$

$$+\frac{1}{120}\begin{bmatrix}(wsu+u_{3i})d_i & (wsu+u_{3i})d_j & (wsu+u_{3i})d_k & (wsu+u_{3i})d_l\\ (wsu+u_{3j})d_i & (wsu+u_{3j})d_j & (wsu+u_{3j})d_k & (wsu+u_{3j})d_l\\ (wsu+u_{3k})d_i & (wsu+u_{3k})d_j & (wsu+u_{3k})d_k & (wsu+u_{3k})d_l\\ (wsu+u_{3l})d_i & (wsu+u_{3l})d_j & (wsu+u_{3l})d_k & (wsu+u_{3l})d_l\end{bmatrix}$$

$$\text{Where } \begin{matrix}\{usu=u_{1i}+u_{1j}+u_{1k}+u_{1l}\}\\ \{vsu=u_{2i}+u_{2j}+u_{2k}+u_{2l}\}\\ \{wsu=u_{3i}+u_{3j}+u_{3k}+u_{3l}\}\end{matrix} \qquad (3.116)$$

and b_i, c_i and d_i are given by the following expressions.

$$b_i=\begin{vmatrix}1 & y_j & z_j\\ 1 & y_k & z_k\\ 1 & y_l & z_l\end{vmatrix};\ c_i=\begin{vmatrix}x_j & 1 & z_j\\ x_k & 1 & z_k\\ x_l & 1 & z_l\end{vmatrix};\ d_i=\begin{vmatrix}x_j & y_j & 1\\ x_k & y_k & 1\\ x_l & y_l & 1\end{vmatrix} \qquad (3.117)$$

Similarly b_j, c_j, d_j; b_k, c_k, d_k and b_l, c_l, d_l follows.

$[K_m]$ = Momentum Diffusion Matrix

$$= \frac{\nu}{36V}\begin{bmatrix} b_i^2 & b_i b_j & b_i b_k & b_i b_l \\ b_i b_j & b_j^2 & b_j b_k & b_j b_l \\ b_i b_k & b_j b_k & b_k^2 & b_k b_l \\ b_i b_l & b_j b_l & b_k b_l & b_l^2 \end{bmatrix} + \frac{\nu}{36V}\begin{bmatrix} c_i^2 & c_i c_j & c_i c_k & c_i c_l \\ c_i c_j & c_j^2 & c_j c_k & c_j c_l \\ c_i c_k & c_j c_k & c_k^2 & c_k c_l \\ c_i c_l & c_j c_l & c_k c_l & c_l^2 \end{bmatrix} + \quad (3.118)$$

$$+ \frac{\nu}{36V}\begin{bmatrix} d_i^2 & d_i d_j & d_i d_k & d_i d_l \\ d_i d_j & d_j^2 & d_j d_k & d_j d_l \\ d_i d_k & d_j d_k & d_k^2 & d_k d_l \\ d_i d_l & d_j d_l & d_k d_l & d_l^2 \end{bmatrix}$$

$[K_t]$ = Heat Diffusion Matrix

$$= \frac{\alpha}{36V}\begin{bmatrix} b_i^2 & b_i b_j & b_i b_k & b_i b_l \\ b_i b_j & b_j^2 & b_j b_k & b_j b_l \\ b_i b_k & b_j b_k & b_k^2 & b_k b_l \\ b_i b_l & b_j b_l & b_k b_l & b_l^2 \end{bmatrix} + \frac{\alpha}{36V}\begin{bmatrix} c_i^2 & c_i c_j & c_i c_k & c_i c_l \\ c_i c_j & c_j^2 & c_j c_k & c_j c_l \\ c_i c_k & c_j c_k & c_k^2 & c_k c_l \\ c_i c_l & c_j c_l & c_k c_l & c_l^2 \end{bmatrix} + \quad (3.119)$$

$$+ \frac{\alpha}{36V}\begin{bmatrix} d_i^2 & d_i d_j & d_i d_k & d_i d_l \\ d_i d_j & d_j^2 & d_j d_k & d_j d_l \\ d_i d_k & d_j d_k & d_k^2 & d_k d_l \\ d_i d_l & d_j d_l & d_k d_l & d_l^2 \end{bmatrix}$$

$[K_s]$ = Stabilization Matrix

$$= \frac{\Delta t}{2}\frac{u_{1av}}{144V} usu \begin{bmatrix} b_i^2 & b_i b_j & b_i b_k & b_i b_l \\ b_i b_j & b_j^2 & b_j b_k & b_j b_l \\ b_i b_k & b_j b_k & b_k^2 & b_k b_l \\ b_i b_l & b_j b_l & b_k b_l & b_l^2 \end{bmatrix} + \frac{\Delta t}{2}\frac{u_{1av}}{144V} vsu \begin{bmatrix} b_i c_i & b_i c_j & b_i c_k & b_i c_l \\ b_j c_i & b_j c_j & b_j c_k & b_j c_l \\ b_k c_i & b_k c_j & b_k c_k & b_k c_l \\ b_l c_i & b_l c_j & b_l c_k & b_l c_l \end{bmatrix} +$$

$$+ \frac{\Delta t}{2}\frac{u_{1av}}{144V} wsu \begin{bmatrix} b_i d_i & b_i d_j & b_i d_k & b_i d_l \\ b_j d_i & b_j d_j & b_j d_k & b_j d_l \\ b_k d_i & b_k d_j & b_k d_k & b_k d_l \\ b_l d_i & b_l d_j & b_l d_k & b_l d_l \end{bmatrix} + \frac{\Delta t}{2}\frac{u_{2av}}{144V} usu \begin{bmatrix} c_i b_i & c_i b_j & c_i b_k & c_i b_l \\ c_j b_i & c_j b_j & c_j b_k & c_j b_l \\ c_k b_i & c_k b_j & c_k b_k & c_k b_l \\ c_l b_i & c_l b_j & c_l b_k & c_l b_l \end{bmatrix} +$$

$$+ \frac{\Delta t}{2}\frac{u_{2av}}{144V} vsu \begin{bmatrix} c_i^2 & c_i c_j & c_i c_k & c_i c_l \\ c_j c_i & c_j^2 & c_j c_k & c_j c_l \\ c_k c_i & c_k c_j & c_k^2 & c_k c_l \\ c_l c_i & c_l c_j & c_l c_k & c_l^2 \end{bmatrix} + \frac{\Delta t}{2}\frac{u_{2av}}{144V} wsu \begin{bmatrix} c_i d_i & c_i d_j & c_i d_k & c_i d_l \\ c_j d_i & c_j d_j & c_j d_k & c_j d_l \\ c_k d_i & c_k d_j & c_k d_k & c_k d_l \\ c_l d_i & c_l d_j & c_l d_k & c_l d_l \end{bmatrix} +$$

$$+ \frac{\Delta t}{2}\frac{u_{3av}}{144V} usu \begin{bmatrix} d_i b_i & d_i b_j & d_i b_k & d_i b_l \\ d_j b_i & d_j b_j & d_j b_k & d_j b_l \\ d_k b_i & d_k b_j & d_k b_k & d_k b_l \\ d_l b_i & d_l b_j & d_l b_k & d_l b_l \end{bmatrix} + \frac{\Delta t}{2}\frac{u_{3av}}{144V} vsu \begin{bmatrix} d_i c_i & d_i c_j & d_i c_k & d_i c_l \\ d_j c_i & d_j c_j & d_j c_k & d_j c_l \\ d_k c_i & d_k c_j & d_k c_k & d_k c_l \\ d_l c_i & d_l c_j & d_l c_k & d_l c_l \end{bmatrix}$$

$$+ \frac{\Delta t}{2}\frac{u_{3av}}{144V} wsu \begin{bmatrix} d_i^2 & d_i d_j & d_i d_k & d_i d_l \\ d_j d_i & d_j^2 & d_j d_k & d_j d_l \\ d_k d_i & d_k d_j & d_k^2 & d_k d_l \\ d_l d_i & d_l d_j & d_l d_k & d_l^2 \end{bmatrix} \qquad (3.120)$$

$$[K] = \frac{1}{36V\rho} \begin{bmatrix} b_i^2 & b_i b_j & b_i b_k & b_i b_l \\ b_i b_j & b_j^2 & b_j b_k & b_j b_l \\ b_i b_k & b_j b_k & b_k^2 & b_k b_l \\ b_i b_l & b_j b_l & b_k b_l & b_l^2 \end{bmatrix} + \frac{1}{36V\rho} \begin{bmatrix} c_i^2 & c_i c_j & c_i c_k & c_i c_l \\ c_j c_i & c_j^2 & c_j c_k & c_j c_l \\ c_k c_i & c_k c_j & c_k^2 & c_k c_l \\ c_l c_i & c_l c_j & c_l c_k & c_l^2 \end{bmatrix} +$$

$$+ \frac{1}{36V\rho} \begin{bmatrix} d_i^2 & d_i d_j & d_i d_k & d_i d_l \\ d_j d_i & d_j^2 & d_j d_k & d_j d_l \\ d_k d_i & d_k d_j & d_k^2 & d_k d_l \\ d_l d_i & d_l d_j & d_l d_k & d_l^2 \end{bmatrix} \qquad (3.121)$$

$[G_1]$=First Gradient Matrix in x_1 direction

$$= \frac{1}{24}\begin{bmatrix} b_i & b_j & b_k & b_l \\ b_i & b_j & b_k & b_l \\ b_i & b_j & b_k & b_l \end{bmatrix} \tag{3.122}$$

$[G_2]$=Second Gradient Matrix in x_2 direction

$$= \frac{1}{24}\begin{bmatrix} c_i & c_j & c_k & c_l \\ c_i & c_j & c_k & c_l \\ c_i & c_j & c_k & c_l \end{bmatrix} \tag{3.123}$$

$[G_2]$=Third Gradient Matrix in x_3 direction

$$= \frac{1}{24}\begin{bmatrix} d_i & d_j & d_k & d_l \\ d_i & d_j & d_k & d_l \\ d_i & d_j & d_k & d_l \end{bmatrix} \tag{3.124}$$

The force vectors f_1, f_2, f_3 and f_4 based boundary surface ijk are written as

$$\{f_1\} = \frac{A\nu}{18V}\begin{bmatrix} b_i u_{1i} + b_j u_{1j} + b_k u_{1k} + b_l u_{1l} \\ b_i u_{1i} + b_j u_{1j} + b_k u_{1k} + b_l u_{1l} \\ b_i u_{1i} + b_j u_{1j} + b_k u_{1k} + b_l u_{1l} \\ 0 \end{bmatrix}^n n_1 + \frac{A\nu}{18V}\begin{bmatrix} c_i u_{1i} + c_j u_{1j} + c_k u_{1k} + c_l u_{1l} \\ c_i u_{1i} + c_j u_{1j} + c_k u_{1k} + + c_l u_{1l} \\ c_i u_{1i} + c_j u_{1j} + c_k u_{1k} + + c_l u_{1l} \\ 0 \end{bmatrix}^n n_2 +$$

$$+ \frac{A\nu}{18V}\begin{bmatrix} c_i u_{1i} + c_j u_{1j} + c_k u_{1k} + c_l u_{1l} \\ c_i u_{1i} + c_j u_{1j} + c_k u_{1k} + + c_l u_{1l} \\ c_i u_{1i} + c_j u_{1j} + c_k u_{1k} + + c_l u_{1l} \\ 0 \end{bmatrix}^n n_3 \tag{3.125}$$

$$\{f_2\}=\frac{A\nu}{18V}\begin{bmatrix} b_i u_{2i}+b_j u_{2j}+b_k u_{2k}+b_2 u_{2l} \\ b_i u_{2i}+b_j u_{2j}+b_k u_{2k}+b_l u_{2l} \\ b_i u_{2i}+b_j u_{2j}+b_k u_{2k}+b_l u_{2l} \\ 0 \end{bmatrix}^n n_1+\frac{A\nu}{18V}\begin{bmatrix} c_i u_{2i}+c_j u_{2j}+c_k u_{2k}+c_l u_{2l} \\ c_i u_{2i}+c_j u_{2j}+c_k u_{2k}+c_l u_{2l} \\ c_i u_{2i}+c_j u_{2j}+c_k u_{2k}+c_l u_{2l} \\ 0 \end{bmatrix}^n n_2+$$

$$+\frac{A\nu}{18V}\begin{bmatrix} d_i u_{2i}+d_j u_{2j}+d_k u_{2k}+d_l u_{2l} \\ d_i u_{2i}+d_j u_{2j}+d_k u_{2k}+d_l u_{2l} \\ c_i u_{2i}+d_j u_{2j}+d_k u_{2k}+d_l u_{2l} \\ 0 \end{bmatrix}^n n_3 \qquad (3.126)$$

$$\{f_3\}=\frac{A\nu}{18V}\begin{bmatrix} b_i u_{2i}+b_j u_{2j}+b_k u_{2k}+b_2 u_{2l} \\ b_i u_{2i}+b_j u_{2j}+b_k u_{2k}+b_l u_{2l} \\ b_i u_{2i}+b_j u_{2j}+b_k u_{2k}+b_l u_{2l} \\ 0 \end{bmatrix}^n n_1+\frac{A\nu}{18V}\begin{bmatrix} c_i u_{2i}+c_j u_{2j}+c_k u_{2k}+c_l u_{2l} \\ c_i u_{2i}+c_j u_{2j}+c_k u_{2k}+c_l u_{2l} \\ c_i u_{2i}+c_j u_{2j}+c_k u_{2k}+c_l u_{2l} \\ 0 \end{bmatrix}^n n_2+$$

$$+\frac{A\nu}{18V}\begin{bmatrix} d_i u_{2i}+d_j u_{2j}+d_k u_{2k}+d_l u_{2l} \\ d_i u_{2i}+d_j u_{2j}+d_k u_{2k}+d_l u_{2l} \\ c_i u_{2i}+d_j u_{2j}+d_k u_{2k}+d_l u_{2l} \\ 0 \end{bmatrix}^n n_3 \qquad (3.127)$$

$$\{f_4\}=\frac{A}{18\rho V}\begin{bmatrix} b_i p_i+b_j p_j+b_k p_k+b_l p_l \\ b_i p_i+b_j p_j+b_k p_k+b_l p_l \\ b_i p_i+b_j p_j+b_k p_k+b_l p_l \\ 0 \end{bmatrix}^n n_1+\frac{A}{18\rho V}\begin{bmatrix} c_i p_i+c_j p_j+c_k p_k+c_l p_l \\ c_i p_i+c_j p_j+c_k p_k+c_l p_l \\ c_i p_i+c_j p_j+c_k p_k+c_l p_l \\ 0 \end{bmatrix}^n n_2+$$

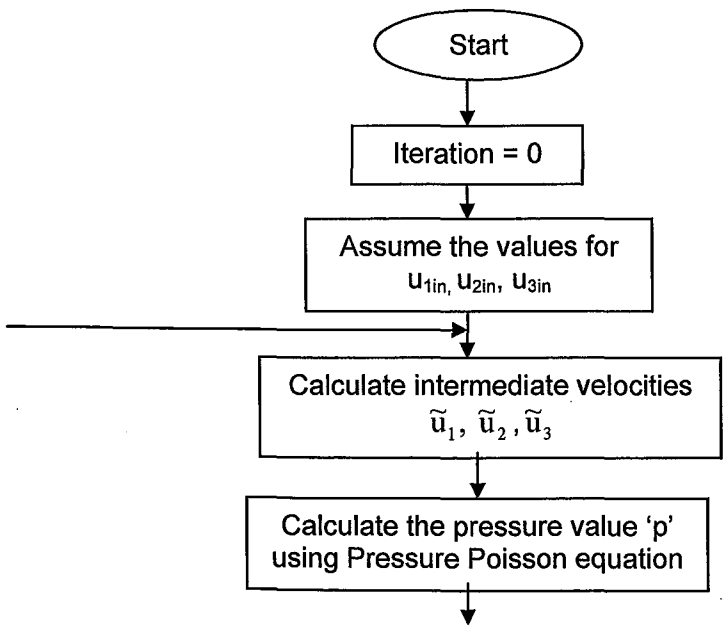
$$+\frac{A}{18\rho V}\begin{bmatrix} d_i p_i+d_j p_j+d_k p_k+d_l p_l \\ d_i p_i+d_j p_j+d_k p_k+d_l p_l \\ d_i p_i+d_j p_j+d_k p_k+d_l p_l \\ 0 \end{bmatrix}^n n_3 \qquad (3.128)$$

$$\{f_5\}=\frac{\alpha A}{18V}\begin{bmatrix} b_iT_i+b_jT_j+b_kT_k+b_lT_l \\ b_iT_i+b_jT_j+b_kT_k+b_lT_l \\ b_iT_i+b_jT_j+b_kT_k+b_lT_l \\ 0 \end{bmatrix}^n n_1+\frac{\alpha A}{18V}\begin{bmatrix} c_iT_i+c_jT_j+c_kT_k+c_lT_l \\ c_iT_i+c_jT_j+c_kT_k+c_lT_l \\ c_iT_i+c_jT_j+c_kT_k+c_lT_l \\ 0 \end{bmatrix}^n n_2+$$
$$+\frac{\alpha A}{18V}\begin{bmatrix} d_iT_i+d_jT_j+d_kT_k+d_lT_l \\ d_iT_i+d_jT_j+d_kT_k+d_lT_l \\ d_iT_i+d_jT_j+d_kT_k+d_lT_l \\ 0 \end{bmatrix}^n n_3 \qquad (3.129)$$

3.4.2 Time step Calculation

The time step restrictions are very much similar to those used in the 2D analysis. So one can refer section 3.3.2. for more details. The material is not presented to avoid repetition.

Commercially available MATLAB programming lanauage is used to develop the CBS finite element code . The program is called CBS flow code. The flowchart used for developing the code is shown in figure 3.3



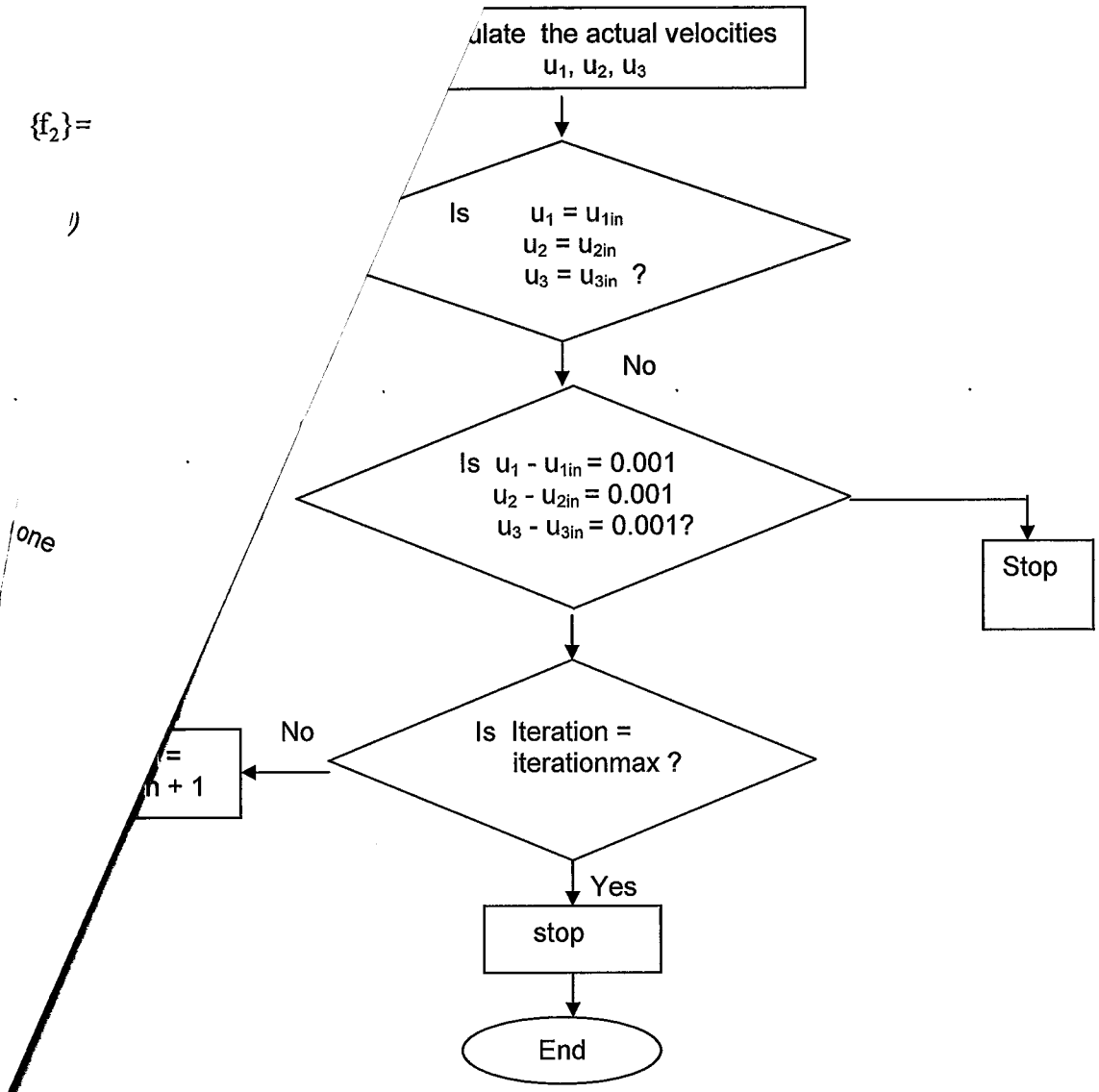


Figure 3.3. Flow chart for implementation of CBS flow program.

3.5 Volume of Fluid Technique

Mold filling process, in which a polymer material flow front advances through a mold, is a typical example of moving boundary problems. The major aim of the simulation of mold filling processes is to predict the trajectories of fluid particles including the development of the flow front. Adaptive grid methods have been commonly used in codes for two-dimensional simulation of molding processes. It involves the process of remeshing in which the mesh covers the fluid area and is extended upon every time step. Remeshing for each time step is tedious and timing

consuming process for fluid front tracking. In order to avoid remeshing and save the computation time, a novel method called the 'Volume-of-Fluid' technique is used to track the fluid front in chip cavity. The VOF based techniques can handle the most complex free surface flow problems.

The VOF method is a simple, but powerful, method to track free fluid surfaces and is based on the concept of a fractional volume of fluid (VOF). This method is shown to be more flexible and efficient than other methods for treating complicated free boundary configurations. In the VOF technique, introduced by Hirt and Nichols (1981) , the flow problem is solved on a fixed grid that covers the entire mold. The Volume-of-Fluid (VOF) method tracks the motion of a certain fluid volume through the computational domain, irrespective of whether the volume contains pure liquid, pure vapour or air or mixture of air and liquid or fluid. Within the scope of the VOF approach, the two-phase flow is treated as a homogeneous mixture and only a set of equations is used for the description.

In order to track the advancement of the interface position during the molding filling process, a volume fractional function 'F' is defined to describe the filling status of each element. The volume fraction function 'F' is a scalar quantity and for an element or cell 'F' is defined as

$$F(x,y,z,t) = \frac{\text{volume of fluid}}{\text{volume of the element}} \qquad (3.130)$$

Depending on the value of 'F' the filling status of an element ,whether it is filled by polymer melt or is empty or partially filled showing the position of flow front is determined.

$$\text{When } F=\begin{cases}1, & \text{the element is identified as filled with the polymer melt}\\ >0 \text{ and } <1, & \text{the interface is located within the element}\\ 0, & \text{the element is empty}\end{cases} \quad (3.131)$$

In the VOF method, the flow front is advanced by solving the following transport equation

$$F_t + (u.\nabla) F = 0 \quad (3.132)$$

Where 'u' is a velocity vector and is calculated by using the CBS scheme.

3.5.1 VOF technique for 2D mold flow analysis

The transport equation for 2D mold flow analysis using VOF is given by the following expression.

$$\frac{\partial F}{\partial t} + u_1 \frac{\partial F}{\partial x_1} + u_2 \frac{\partial F}{\partial x_2} = 0 \quad (3.133)$$

The solution of above equation gives the 'F' values for all elements in a computational domain. In order to get the converged solution and successful implementation of the front tracking algorithm, one has to choose the volume fraction function 'F' properly in the polymer melt-air zone and the subsequent evaluation of the new front from the predicted value of 'F'. The numerical solution of 'F' may have numerical oscillations since transport equation contained the convection velocity terms. To obtain accurate solution for front tracking algorithm, it is desired that the numerical oscillations be minimum or less. Hence the above equation is modified by to reduce the oscillations by introducing a new second term called 'artificial diffusion term". The new modified transport equation as suggested by Satya Prasad et al. (2000) is given by

$$\frac{\partial F}{\partial t} + u_1 \frac{\partial F}{\partial x_1} + u_2 \frac{\partial F}{\partial x_2} - \upsilon_{ad}\left[\frac{\partial^2 F}{dx_1^2} + \frac{\partial^2 F}{dx_2^2}\right] = 0 \quad (3.134)$$

Where υ_{ad} is a scalar quantity called 'artificial diffusivity' and is selected suitably.

The addition of artificial diffusivity has another important benefit in the sense that it causes partial slip of the polymer melt–air interface at the wall, by the pseudo-diffusion mechanism. It stabilizes numerical oscillation during simulation. It is used to tune the results so that spurious oscillations occurring in numerical results converge fast to a steady value. Suppose if $\upsilon_{ad} = 1$, '*F*' values ideally should be in the range (0,1) for all nodes of the domain. If '*F*' values are out of the range, artificial diffusivity value is adjusted by trial and error method so that '*F*' values fall in the range (0, 1) for front tracking method.

Discretization of equation (3.133) in time domain in an explicit manner (Satya Prasad et al., 2000) and application of the standard Galerkin weighted residual method leads to

$$\int_{\Omega}[N]^{T}F^{n+1}d\Omega = \int_{\Omega}[N]^{T}F^{n}d\Omega - \Delta t\int_{\Omega}N]^{T}\left[u_1\frac{\partial F}{\partial x_1}+u_2\frac{\partial F}{\partial x_2}-\upsilon_{ad}\left(\frac{\partial^2 F}{dx_1^2}+\frac{\partial^2 F}{dx_2^2}\right)\right]^{n}d\Omega \quad (3.134)$$

By considering 3 noded triangular element, integration of above equations and further simplification results in the following equation.

$$[M]\{F\}^{n+1} = [M]\{F\}^{n} - \Delta t[C]\{F\}^{n} - \frac{\Delta t\,\upsilon_{ad}}{4A}\begin{bmatrix} b_i^2 & b_ib_j & b_ib_k \\ b_ib_j & b_j^2 & b_jb_k \\ b_ib_k & b_jb_k & b_k^2 \end{bmatrix}\{F\}^{n} - \frac{\Delta t\,\upsilon_{ad}}{4A}\begin{bmatrix} c_i^2 & c_ic_j & c_ic_k \\ c_ic_j & c_j^2 & c_jc_k \\ c_ic_k & c_jc_k & c_k^2 \end{bmatrix}\{F\}^{n} \quad (3.135)$$

where [M] and [C] are lumped mass matrix and convection matrix and given by eq.(3.49) and (3.51), respectively, and A is area of a triangular element given by the equation (3.50). Note that the boundary integrals for the second order terms have not been shown in the above equation.

3.5.2 VOF technique for 3D mold flow analysis

The transport equation for 3D mold flow analysis using VOF is written as below.

$$\frac{\partial F}{\partial t}+u_1\frac{\partial F}{\partial x_1}+u_2\frac{\partial F}{\partial x_2}+u_3\frac{\partial F}{\partial x_3}=0 \tag{3.136}$$

The modified transport equation taking in to account the artificial diffusivity is given by

$$\frac{\partial F}{\partial t}+u_1\frac{\partial F}{\partial x_1}+u_2\frac{\partial F}{\partial x_2}+u_3\frac{\partial F}{\partial x_3}-\upsilon_{ad}\left[\frac{\partial^2 F}{dx_1^2}+\frac{\partial^2 F}{dx_2^2}+\frac{\partial^2 F}{dx_3^2}\right]=0 \tag{3.137}$$

The formulation procedure is the same as that followed for 2D flow analysis.

The discretization of the above equation in time domain in explicit form results in the following equation.

$$\begin{aligned}\int_V [N]^T F^{n+1} dV = \int_V [N]^T F^n dV - \Delta t \int_V N]^T \left[u_1\frac{\partial F}{\partial x_1}+u_2\frac{\partial F}{\partial x_2}+u_3\frac{\partial F}{\partial x_3}\right]^n dV \\ -\Delta t\int_V N]^T \upsilon_{ad}\left(\frac{\partial^2 F}{dx_1^2}+\frac{\partial^2 F}{dx_2^2}+\frac{\partial^2 F}{dx_3^2}\right)^n dV\end{aligned} \tag{3.138}$$

Evaluation of integrals using four noded tetrahedron element yields

$$[M]\{F\}^{n+1} = [M]\{F\}^{n} - \Delta t[C]\{F\}^{n} - \frac{\Delta t\, \upsilon_{ad}}{36V}\begin{bmatrix} b_i^2 & b_i b_j & b_i b_k & b_i b_l \\ b_i b_j & b_j^2 & b_j b_k & b_j b_l \\ b_i b_k & b_j b_k & b_k^2 & b_k b_l \\ b_i b_l & b_j b_l & b_k b_l & b_l^2 \end{bmatrix}\{F\}^{n} -$$

$$-\frac{\Delta t\, \upsilon_{ad}}{36V}\begin{bmatrix} c_i^2 & c_i c_j & c_i c_k & c_i c_l \\ c_i c_j & c_j^2 & c_j c_k & c_j c_l \\ c_i c_k & c_j c_k & c_k^2 & c_k c_l \\ c_i c_l & c_j c_l & c_k c_l & c_l^2 \end{bmatrix}\{F\}^{n} - \qquad (3.139)$$

$$-\frac{\Delta t\, \upsilon_{ad}}{36V}\begin{bmatrix} d_i^2 & d_i d_j & d_i d_k & d_i d_l \\ d_i d_j & d_j^2 & d_j d_k & d_j d_l \\ d_i d_k & d_j d_k & d_k^2 & d_k d_l \\ d_i d_l & d_j d_l & d_k d_l & d_l^2 \end{bmatrix}\{F\}^{n}$$

where [M] and [C] are lumped mass matrix and convection matrix for 3D analysis and are given by eq.(3.113) and (3.115), respectively, and V is volume of a tetrahedron element given by the equation (3.114). Here also the boundary integrals due to the integration of second order terms have not been shown in the above equation.

In the solution method, the velocities u_1, u_2, u_3 are first obtained from using the CBS method. Then the solution of volume fraction function $F(u_1,u_2, u_3, t)$ is advanced and new F values are obtained at all the spatial nodes. The location of the front at any instant is identified by plotting the contour of F.

3.6 Optimization

Optimization is the process of making something better. It is the process of adjusting the inputs or characteristics of a device, mathematical process or experiment to find the minimum or

maximum output or result (Haupt and Haupt, 2004). In other words, it is the best solution among available solutions for a particular problem in hand subjected to constraints.

In this research work, optimization of the mold filling process is carried out by using a hybrid neuro-genetic method. The methodology involves the coupling of artificial neural network (ANN) into Genetic Algorithm to get the optimum gate size of mold cavity. Its implementation consists of three steps.

i) The mold filling data is obtained from a parametric study conducted by varying the process variables using the Characteristic Based Split Finite Element Method for a given configuration of package.

ii) The data obtained from parametric study in first step is used to train an Artificial Neural Network (ANN). ANN will learn the complex relationship between the input data and output data and it correlates the input-output data by carefully emulating the human brain's ability to make decisions and draw conclusions through neuroscience. ANN predicts the mold filling time for various process parameter changes. Once a trained ANN model is achieved, it is possible for ANN to make output predictions for new sets of input parameters.

iii) ANN predicted data is used in Genetic Algorithm (GA) for optimization purpose. GA optimizes the gate size subjected to maximum pressure value and minimum mold filling time.

Because of Genetic Algorithms' flexibility to handle the function to be optimized, it is possible to use an ANN model in place of a closed form function used in calculus problems. Thus, a hybrid Neural Network-Genetic Algorithm scheme has been used in this research work where a trained ANN serves as an input-output model and subsequently, inputs data from ANN will be used for optimization using Genetic Algorithms. In the following few lines, brief information on both ANN and GAs is given.

3.6.1 Artificial Neural Network to predict the mold filling time

The purpose of using ANN for prediction of the mold filling time is to reduce the modeling and post processing effort and also the computational time during parametric studies of the mold filling process. This is mainly due to the fact that simulation of the mold filing process during transfer molding using the FEA/VOF technique takes considerable time and effort since modeling, solver and post-processing i.e. flow front tracking is laborious and tedious process. In order to carry out parameter study by changing the parameter of interest while keeping all other process variables constant, one has to start afresh and repeat the whole solution process to get the final front profiles. Since mold filling process is very complex in nature and takes 4-5 hours for each simulation case, ANN can be augmented to the solution process along with FEM to reduce the computation time and effort.

ANN is a powerful data modeling tool that is able to capture and represent complex input/output relationships. Artificial neural network computation is an alternative computation paradigm to the usual von Neumann machine computation based on a programmed instruction sequence to date (Hertz et al., 1991). It is inspired by the knowledge from the neuroscience and it draws its methods in large degree from statistical physics. The motivation for the development of neural network technology stemmed from the desire to develop an artificial system that could perform "intelligent" tasks similar to those performed by the human brain.

An artificial neural network (ANN) is an interconnected group of artificial neurons, akin to the vast network of neurons in the human brain, which uses a mathematical model for information processing based on a connectionist approach to computation. It is used to model complex relationships of highly non-linear dynamic nature between inputs and outputs or to find patterns in data. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network (http://en.wikipedia.org/wiki/Artificial_neural_networks).

A typical neural network is shown in Figure 3.4. It receives multi-input data, in this case 4 inputs and does the required computation to give the out put of interest.

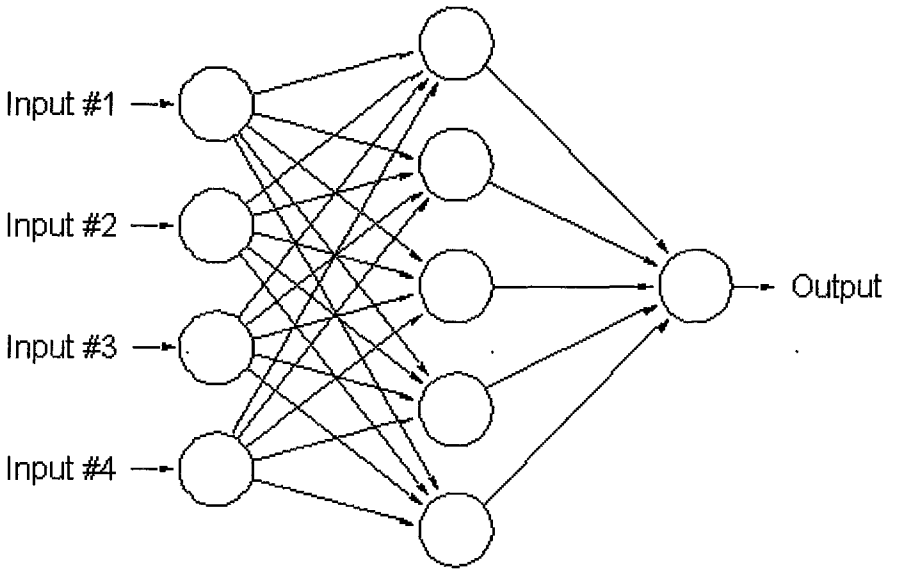


Figure 3.4. A typical neural network with 4 inputs and a single output.

(Source: http://smig.usgs.gov/SMIG/features_0902/tualatin_ann.fig3.gif)

ANN works by acquiring the knowledge required for the computation through learning process in much similar way the child learns at earlier stage of the development. Once ANN is trained, it establishes a correlation ship between input and output data. Then it computes the desired output for a specified input.

Learning in ANN is directed to developing new connections and / to modifying the strengths of the existing connections in order to correctly represent the environment. The learning paradigm of an artificial neural network depends very much on the network structure and the input data characteristics (Wu, 1994). Two popularly used learning models to train ANNs are

- The Back Propagation Model.

- The Self-Organization Model.

3.6.1.1 The Back Propagation Model

This type of ANN architecture is used for multilayered network and performs supervised learning. In this training model, a teacher is assumed to be present during the learning process and each example pattern used to train the network includes an input pattern together with a target or desired output pattern, the correct answer. During the learning process, a comparison can be made between the computed output by ANN and correct output to determine the error. The error is then used to change the network parameters which result in an improvement in performance. The error is made utilized through some form of computation and feedback to adjust the individual weights to reduce further error for training each next pair (Patterson, 1998). After iteratively adjusting the weights for all training patterns, the weight values may converge to a set of values needed to perform the required pattern recalls. Learning has been achieved when errors for all training patterns have been reduced to some acceptable levels for all patterns not in the training set.

The equations that describe the network training and operation are divided into two categories: feed forward calculations and error back propagation calculations.

The feed forward calculations, where each connection and all data flow go from left to right, are used both in the training mode and in the operation of the trained neural network (Eberhart and Dobbins,1990). Error back propagation calculations are applied only during the training process. During the training phase, the feed forward output state calculation is combined with backward error propagation and weight adjustment calculations that represent the network's learning or training.

3.6.1.2 The Self - organization Model

In this neural network architecture, the neural network has no feedback on the desired or correct output. The self organization model is trained without supervision. We present only the input data and the network organizes itself. There is no trainer to present target patterns. Therefore the system must learn by discovering and adapting to structured features in the input patterns, that is, by adapting to statistical regularities or clustering of patterns from the input training samples (Patterson, 1998).

This model is more biologically oriented than the back propagation model. One imoprtant feature of this model is that it is trained without supervision. This is similar to many of the neural cells in our brain, in other words nobody applies electronic stimuli to our neurons to train them to, say, learn to walk or to speak. The self organizing feature map implementation described by Kohonen might bear some resemblance to the way some areas or our brains are organized (Eberhart and Dobbins, 1990).

In this research work, the Error Back Propagation Network (EBPN) model has been used to predict the mold filling time by ANN.

3.6.2 Gate Size Optimization by Genetic Algorithm

The purpose of optimization in this study is to select the optimum gate size which has direct effect on mold filling stage during transfer molding. This in turn has bearing on the cycle time required for transfer molding process which consist of mold filling time and subsequent curing time for the IC package to be encapsulated.

When the input data for optimization are in scattered pattern, not connected by closed form of equation, the optimization by usual methods of calculus and linear or dynamic programming

techniques is not possible. Then Genetic Algorithms are good choice to optimize the desired data.

Genetic algorithms (GAs) are stochastic search and optimization methods based on the principles of genetics and Darwinian principle of natural selection and survival of the fittest. Genetic algorithms are typically implemented as a computer simulation in which solutions are represented in binary strings of 0s and 1s. Other different encodings are also possible. To use a genetic algorithm, one must represent a solution to the problem as a *genome* (or *chromosome*). The genetic algorithm then creates a population of solutions and applies genetic operators to evolve the solutions in order to find the best one(s).The evolution starts from a population of completely random individuals and happens in generations. In each generation, the fitness of the whole population is evaluated, multiple individuals are stochastically selected from the current population (based on their fitness), modified to form a new population, which becomes current in the next iteration of the algorithm.
(http://en.wikipedia.org/wiki/Genetic_algorithm#Initialization)

Genetic Algorithm differs substantially from more traditional search and optimization techniques. Zalzala and Fleming (1997) list the four most significant differences which are as follows.

1. GA searches a population of points in parallel, not a single point.
2. GAs use probabilistic transition rules, not deterministic ones.
3. GAs work on an encoding of the parameter set rather than the parameter set itself (except where real valued of individuals are used).
4. GAs do not require derivative information or other auxiliary knowledge ; only the objective function and corresponding fitness levels influence the directions of search.

It is important to note that the GA can provide a number of potential solutions to a given problem and the choice of a final solution is left to the user.

3.6.2.1 Working Procedure of a Genetic Algorithm

Working procedure of Genetic Algorithm is shown in the following steps

1. *Initialization*: Initially many individual solutions or chromosomes are randomly generated to form an initial population. The population size depends on the nature of the problem, but typically contains several possible solutions.
2. *Evaluation* : Evaluate the fitness function *f(x)* of each chromosome *x* in the population
3. *Selection*: A proportion of the existing population is selected during a certain fixed interval of time to produce a new population or generation. Individual solutions are selected through a *fitness-based* process. The selection process is stochastic in nature and is designed in such way that a small proportion of less fit solutions are selected. Roulette wheel selection and Tournament selection are the popular selection methods used in GA.
4. *Reproduction*: The next step is to generate a second generation population of solutions from those selected through genetic operators: crossover and mutation. The new generation individual typically shares many of the characteristics of its parent or initial generation. The reproduction process continues until a new population of solutions of appropriate size is generated.

These processes ultimately result in the next generation population of individuals or chromosomes that is different from the initial generation. Generally the average fitness will have increased by this procedure for the population, since only the best fit individuals from the first generation are selected for reproduction.

5. *Run*: The new generation population is used for a further run of the algorithm and many such generations are produced till one get the end condition satisfactorily.
6. *Termination:* The new population generation process is repeated until a termination conditions, such as converged solution, fixed number of generations, allotted commutation time/ money, stagnant solution for further reproduction and manual stop are reached.

The best fit solution after the termination process is the optimization solution for given problem based on the specified constraints at a particular interval of a time. The implementation and evaluation of the fitness function is an important factor in the speed and efficiency of the algorithm.

3.6.2.2 Genetic Algorithm Operators

In genetic algorithm, each new generation of population of individuals is reproduced using the following main operators.

1. Crossover
2. Mutation

The performance of GA is influenced mainly by these two operators. Before we can explain more about crossover and mutation, some information about encoding chromosomes will be given.

Encoding of a Chromosome : A chromosome should in some way contain information about solution that it represents. The most used way of encoding is a binary string. A chromosome then could look like the one shown in Figure 3.4.

Chromosome 1	1101100100110110
Chromosome 2	1101111000011110

Figure 3.4. Binary encoding of a chromosome

Each chromosome is represented by a binary string. Each bit in the string can represent some characteristics of the solution. Another possibility is that the whole string can represent a number.

Of course, there are many other ways of encoding. The encoding depends mainly on the solved problem. For example, one can encode directly integer or real numbers, sometimes it is useful to encode some permutations and so on.

3.6.2.2.1 Crossover

Crossover operates on selected genes from parent chromosomes and creates new offspring. The simplest way how to do that is to choose randomly some crossover point and copy everything before this point from the first parent and then copy everything after the crossover point from the other parent.

Crossover can be illustrated as below shown in Figure 3.5 (| is the crossover point).

Chromosome 1	11011 | 00100110110
Chromosome 2	11011 | 11000011110
Offspring 1	11011 | 11000011110
Offspring 2	11011 | 00100110110

Figure 3.5 Crossover illustration

There are other ways how to make crossover, for example we can choose more crossover points. Crossover can be quite complicated and depends mainly on the encoding of chromosomes. Specific crossover made for a specific problem can improve performance of the genetic algorithm.

3.6.2.2.2 Mutation

After a crossover is performed, mutation takes place. Mutation is intended to prevent falling of all solutions in the population into a local optimum of the solved problem. Mutation operation randomly changes the offspring resulted from crossover. In case of binary encoding we can switch a few randomly chosen bits from 1 to 0 or from 0 to 1. Mutation can be then illustrated in Figure 3.6.

Original offspring 1	1101111000011110
Original offspring 2	1101100100110110
Mutated offspring 1	1100111000011110
Mutated offspring 2	1101101100110110

Figure 3.6 Mutation illustration

The technique of mutation (as well as crossover) depends mainly on the encoding of chromosomes. For example when we are encoding permutations, mutation could be performed as an exchange of two genes.

3.7 Summary

The Characteristic Based Split (CBS) has been discussed in detail to solve the Navier Stokes Equations in non-conservative form for both 2D and 3D computational domains. The

Implicit Characteristic Galerkin method has been used in deriving the basic element stiffness matrices. A 2D mold flow problem was formulated using 3 noded linear triangular element and 4 noded tetrahedron element for 3D problem. The Volume-of-Fluid technique in explicit form has been used to track polymer melt flow in a mold during the mold filling process. The optimization procedure to optimize the mold flow process has been discussed using the hybrid neuro-genetic algorithm. Brief introduction to artificial neural networks (ANNs), ANN models and its implementation has been discussed in this chapter. Also introduction to Genetic Algorithm (GA), GA working procedure and GA operators has been highlighted in this chapter.

CHAPTER 4

RESULTS AND DISCUSSION

4.0 Overview

In this chapter, the results of this research work are presented in the following broad areas.

- ❖ Two Dimensional Mold Filling
- ❖ Parameter studies for 2D Mold filling
- ❖ Three Dimensional Mold Filling
- ❖ Parameter studies for 3D Mold filling
- ❖ Optimization

4.1 Two Dimensional Mold Filling

In this section the mold filling of epoxy molding compound in 2D mold cavity has been analyzed and simulated. The 2D flow in a mold cavity is analyzed using the hybrid CBS scheme / VOF technique. The finite element method is used as numerical tool to get the solution of primitive variables namely velocity and pressures. The velocities obtained are thus given as input to the volume of fluid method, a front tracking algorithm, to trace fluid front for a given

instant of time. The program has been coded in commercially available MATLAB programming language.

4.1.1 Code Verification

Before taking up the mold filing simulation analysis for IC chip cavities used in encapsulation process, the flow code is verified for its correctness. The results obtained from the present solution algorithm are validated with the results of Gethin and Abdullah (1997) for two cases of mold filling process namely rectangular cavity filling and a flat cavity filling in XY plane.

4.1.1.1 Simulation of rectangular cavity filling

In this case the problem of filling a rectangular cavity with uniform thickness is considered to study the flow behavior in a cavity. The rectangular cavity of size 100 x 30 x 3 mm considered in the analysis is shown in the Figure 4.1.

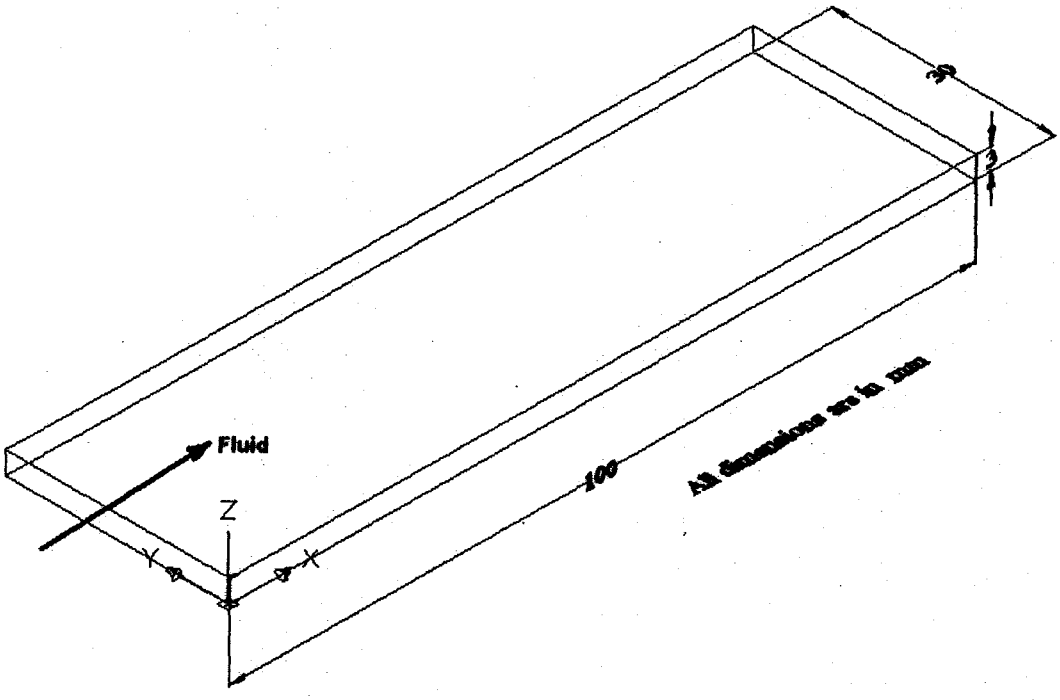


Figure 4.1 Dimensions of rectangular cavity used in analysis

The fluid enters the rectangular cavity with a uniform inlet velocity of 1m/s. For analysis purpose, the properties of the fluid considered are as below.

ρ_{fluid} = fluid density = 1000 kg / m^3, η_{fluid} = viscosity of fluid =0.005 Ns / m^2,

A zero pressure is applied at the boundary wall opposite to the fluid entry. No-slip boundary condition is considered for other walls. The boundary conditions are shown in Figure 4.2. The geometrical domain is discretized into linear triangular elements. The finite element mesh shown in Figure 4.3 consists of 663 nodes and 1200 elements.

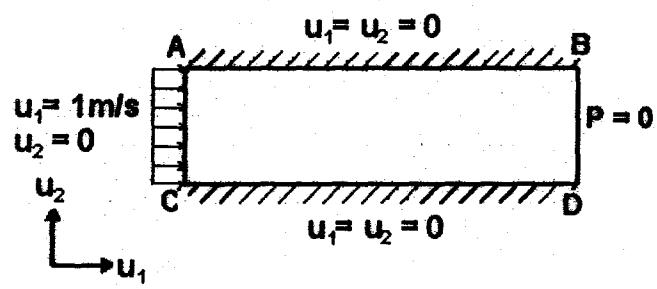


Figure 4.2 Boundary conditions for rectangular cavity filling problem.

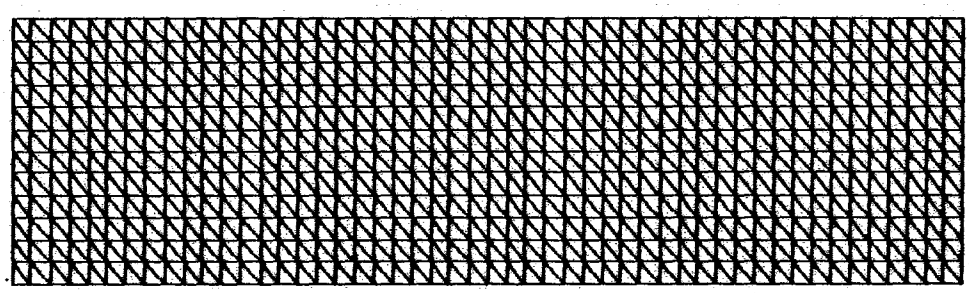


Figure 4.3 Uniform Finite Element Mesh for rectangular cavity filling

The position of free surface ' l_{free}' considering the slip on the walls is given by the expression

$$l_{free} = u_1 \times t \qquad (4.1)$$

where 't' is the time instant at which the front profile is tracked.

Since the boundary layer effect along the sides AB and CD is small, the predicted free surface of the flow front will be very close to the position 'l_{free}' given by equation (4.1). The cavity filling simulation results are obtained using a time step of 0.0005 seconds. The present simulated flow fronts agree with the analytical results from equation (4.1), since the difference is very small. The current simulated results of front profiles are compared with those obtained by Gethin and Abdullah (1997) under the identical conditions of geometry, flow velocity and boundary conditions. The compared results are shown in Figure 4.4. It is observed that the present simulated front profiles match almost each with those by Gethin and Abdullah (1997). This proves the validation of the present fluid flow algorithm to simulate the flow in simple cavities.

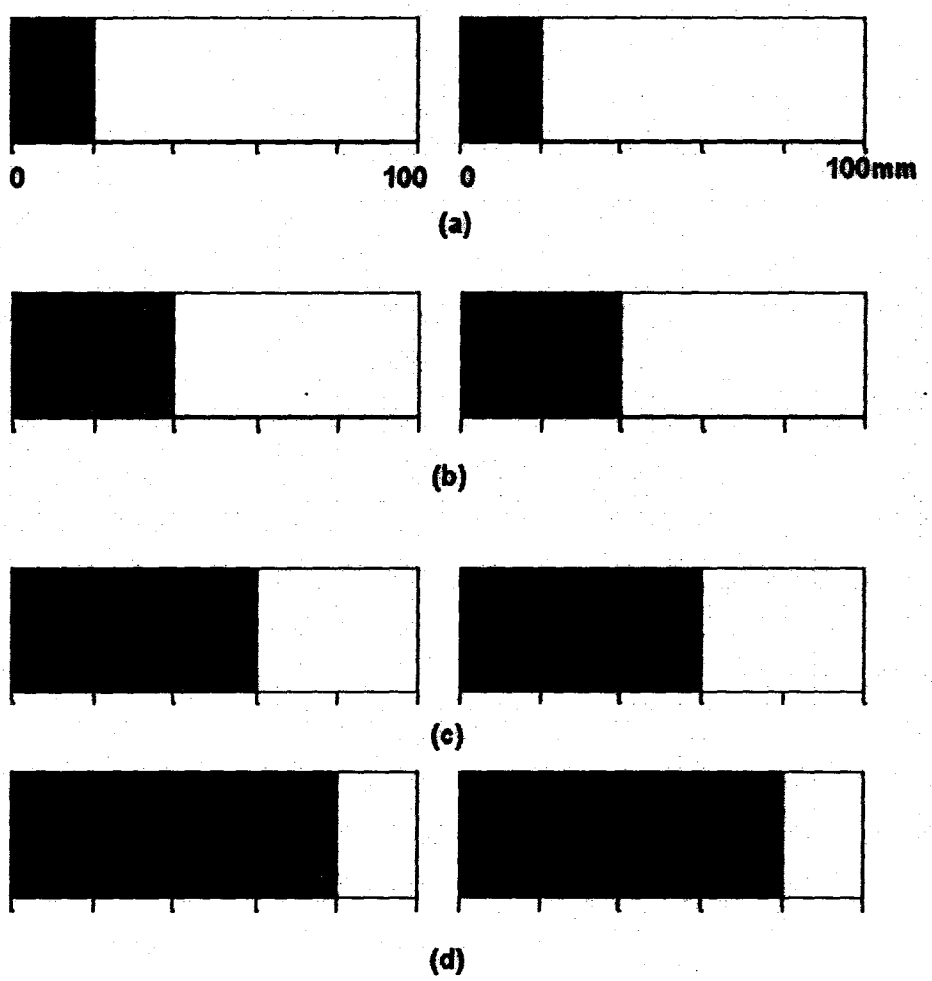


Figure 4.4 Comparison of flow front profiles with those by Gethin and Abdullah (1997) at various time steps (a) t = 0.02s, (b) t = 0.04s, (c) t = 0.06s and (d) t = 0.08s.

4.1.1.2 Simulation of flat cavity filling

To test the robustness of the current algorithm, it is also used to simulate the flow in casting process. For this, the Z-shaped cavity is considered and its geometry is shown in Figure 4.5. The problem is solved with similar type of conditions that has been considered for the plane rectangular cavity filing.

Thus ρ_{fluid} = 1000 kg / m^3, η_{fluid} = 0.005 Ns/m^2 and fluid inflow velocity = 1m/s.

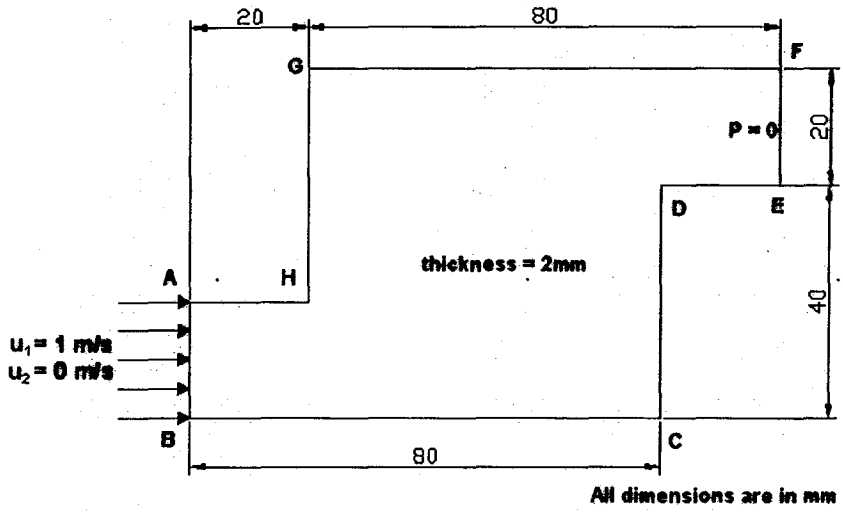


Figure 4.5 Geometrical dimensions and boundary conditions for flat cavity filling in x_1x_2 plane.

The following boundary conditions are considered in the analsysis.

At inlet (AB): u_1=1m/s, u_2 = 0.

At outlet (EF): Pressure 'P' = 0.

At Boundary walls (BC, CD, FG, GH, AH): u_1= u_2 = 0 (no slip condition)

The domain is discretized in space using linear triangular elements. The mesh shown in Figure 4.6 consists of 316 nodes and 550 elements.

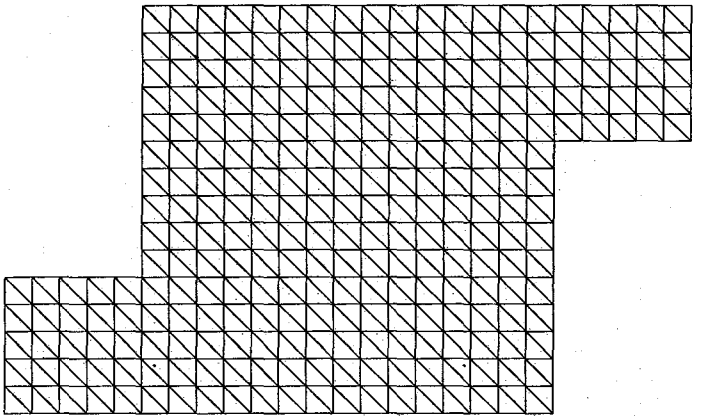


Figure 4.6 Finite element mesh for flat cavity filling

The velocity field is obtained from the CBS Scheme and is shown Figure 4.7.

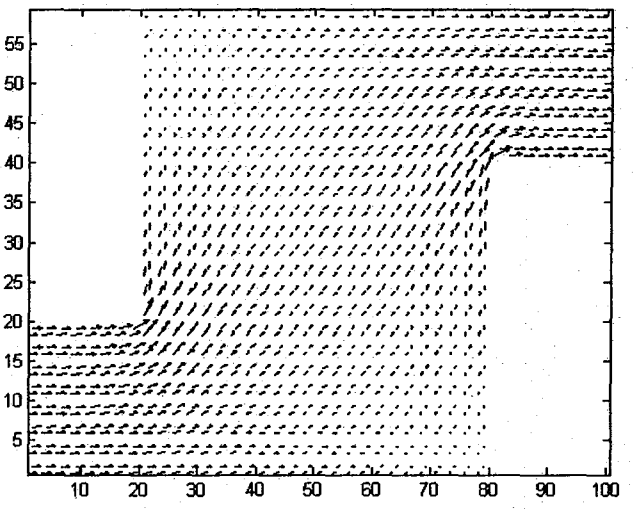


Figure 4.7 Velocity vectors plot for flow in x-y plane cavity at t = 0.22 s

From Figure 4.7, the velocity profile indicates that axial velocities exist at the entry and opposite end zones. When the flow progressively moving, all the velocity vectors at the middle

zone are heading towards the upper end with almost the same magnitudes in both x- and y-components. It is also observed that the velocity are vectors close to the wall region into the wall direction. The velocity field will be used in pseudo-concentration approach used to track the front of the fluid.

The velocity field is used in front tracking algorithm to get the volume fraction factors for each nodes and then front profiles are plotted to know the free surface movement at particular time of instant. The time at which the cavity fills completely is called 'filling time' and thus it can be known very easily from the proposed algorithm. The various front profiles at different time steps are shown in Figure 4.8.

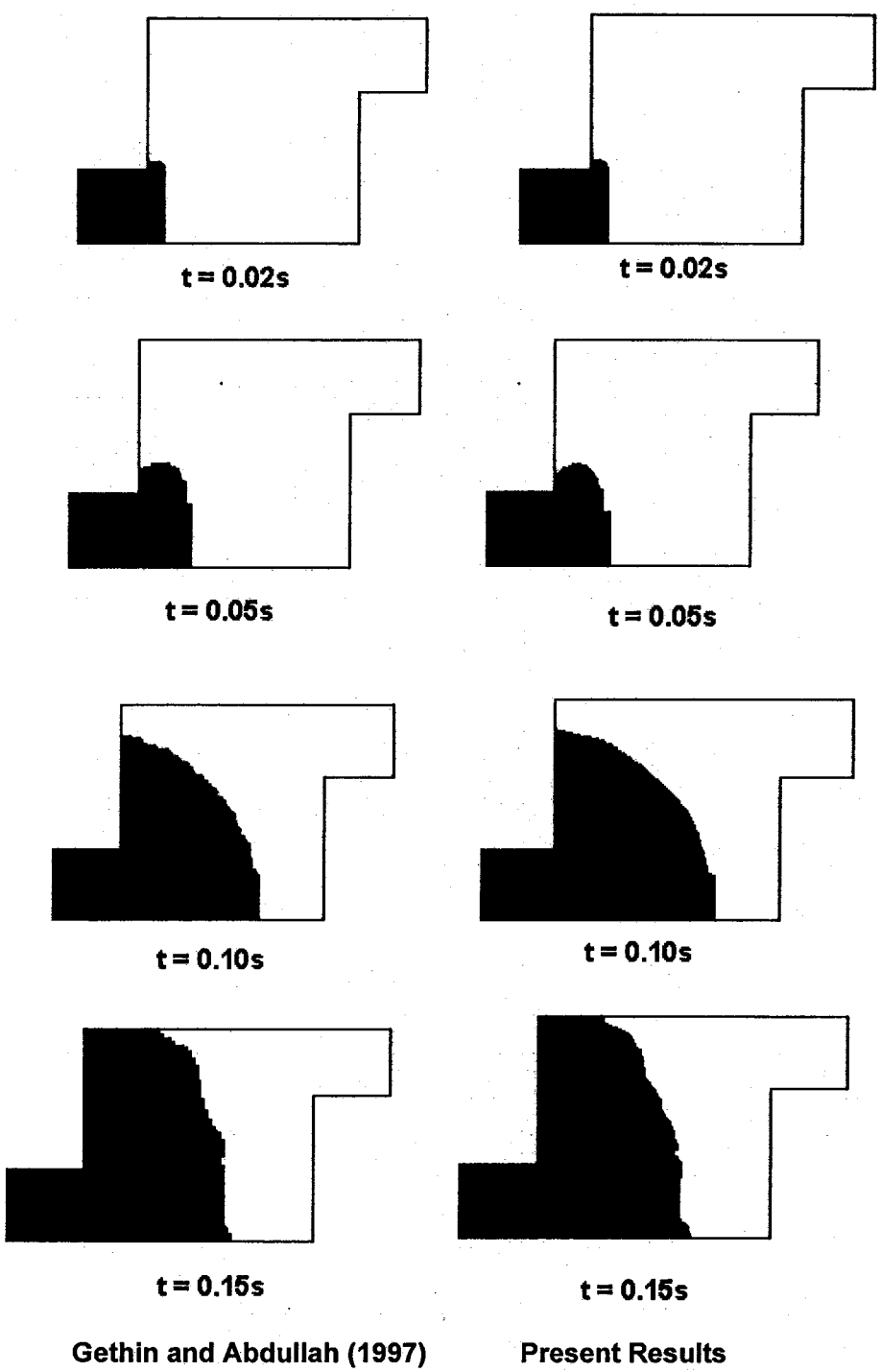


Figure 4.8a. Comparison of front profiles for flat cavity filling with those of Gethin & Abdullah (1997) at time steps t = 0.02 s, 0.05 s, 0.10 s and 0.15 s respectively.

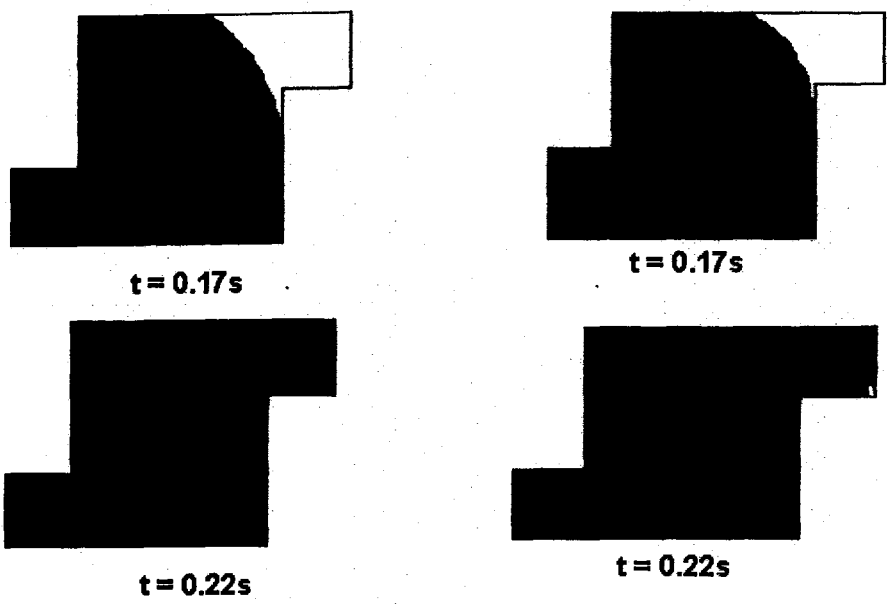


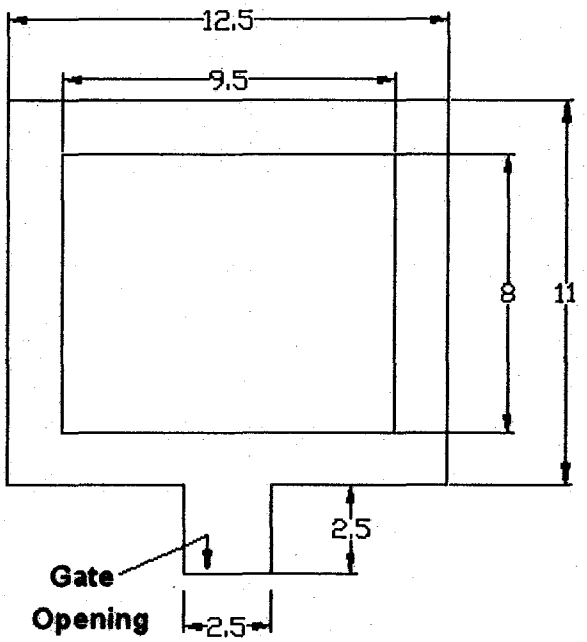
Figure 4.8b . Comparison of front profiles for flat cavity filling with those of Gethin & Abdullah (1997) at time steps t = 0.17 s and 0.15 s respectively.

In volume of fluid technique, a time step, Δt = 0.001s and artificial diffusivity, V_{ad} = 1.0 have been used. Comparison of front profiles at different instant is shown in Figure 4.8. It can be observed that at 0.02s, the flow is channeled uniformly into the cavity in the vicinity of entry zone. At 0.05s, the free surface starts to move upward and wetted the left sidewall. From 0.05s to 0.1 0s, the parabolic front pattern advances to the right and up portion of the cavity. The front contours plotted from the FEM analysis conform reasonably well up to 0.10s. At 0.15second, the flow has reached the top wall but the front center portion tends to move faster to the upper direction. The simulated filling time is 0.22s, which compares well to that predicted by Gethin and Abdullah (1997).

After establishing the robustness of the algorithm it is then applied to solve the molding filling problem in integrated circuit (IC) chip encapsulation process namely the transfer molding process. Cavity filling in IC chip is investigated in next few pages.

4.1.2 Mold Filling in Single chip cavity

In this section the analysis of mold filling in single chip cavity is analyzed. The geometrical description of the problem is shown in Figure 4.9.



(All Dimensions are in mm)

Figure 4.9 Single chip cavity geometry

The chip with dimensions 9.5 X 8mm with mold clearance of 1.5mm is considered in the analysis. The epoxy molding compound (EMC) enters the gate with a velocity of 1.75mm/s. The following properties of the fluid are incorporated in the model to analyze the flow filling behavior in single chip cavity.

Density of the fluid = 2.2e-6 kg/mm^3.

Viscosity of the fluid = 5e-3 kg-s/mm.

The analysis uses 2 dimensional linear triangular elements. The finite element mesh and boundary conditions are shown in Figure 4.10.

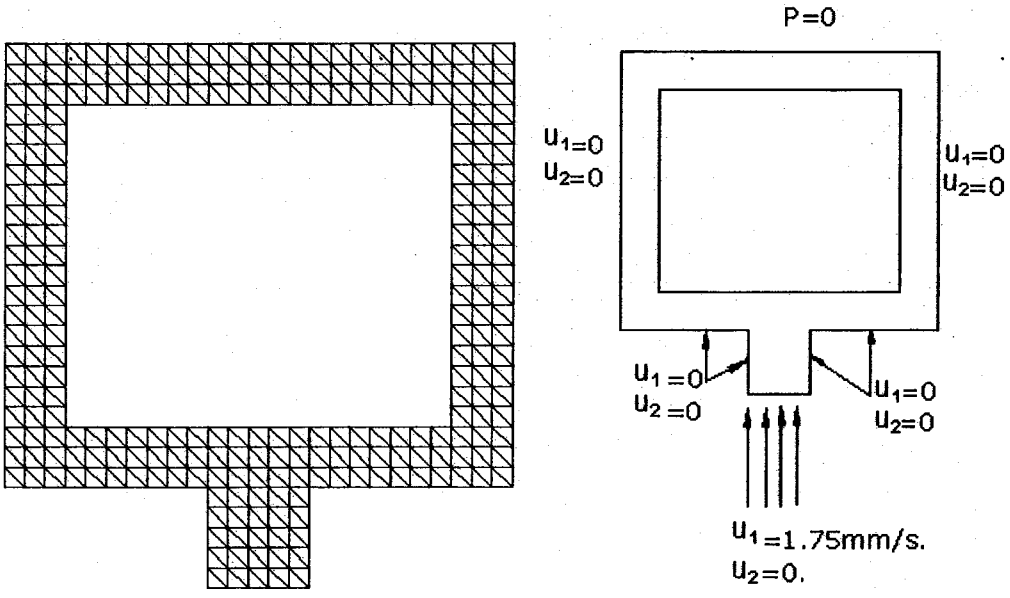
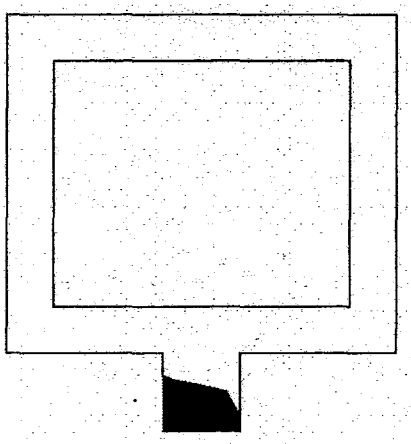
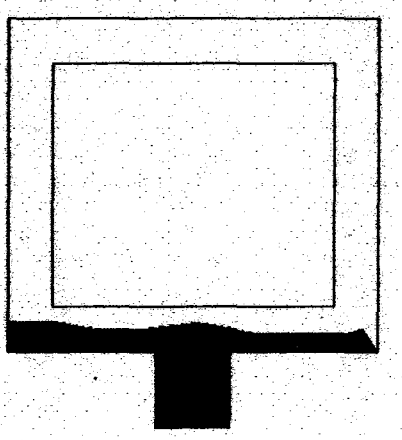


Figure 4.10. Finite Element Mesh and boundary conditions for single chip cavity filling.

The structured mesh consists of 358 nodes and 542 linear triangular elements. The fluid front profiles obtained from the simulation software code for various time steps are shown in Figure 4.11a and Figure 4.11b.

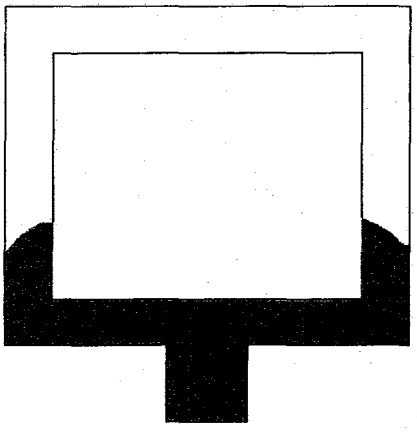


(a) Front Profile at 't = 0.015s'.

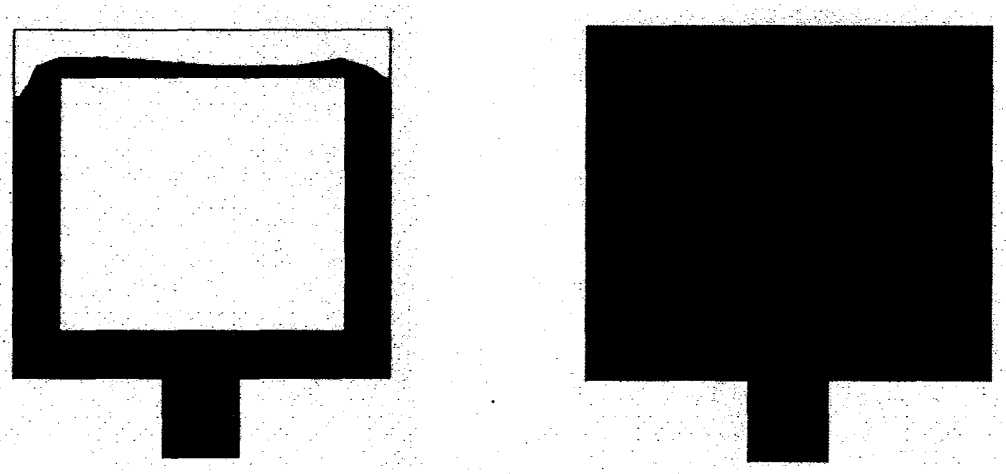


(b) 't = 0.05s'

Figure 4.11a. Front profiles for single chip cavity filling at time steps t = 0.015s & 0.05s.



(c) t = 0.5 s



(d) t= 0.85 s (e) filling time t = 1.21 s.

Figure 4.11b. Front profiles for single chip cavity filling at time steps t = 0.5s ,0.85s and 1.21s.

The filling time or transfer time taken by a epoxy molding compound to fill the chip cavity completely is 1.21seconds.

4.1.3 Mold Filling in multi chip cavity

After single chip cavity filling, the present simulation software code is used to simulate the fluid flow in actual electronic package. For this, a package with 3 X 3 array size is considered in the analysis. The package dimensions and boundary conditions are shown in Figure 4.12. The package is analyzed for flow filling in cavity in horizontal direction. The epoxy material enters the package with a transfer velocity of 1.75mm/s. The fluid properties considered are specific gravity =1.8 and viscosity $\eta = 4.5 Ns/m^2$.

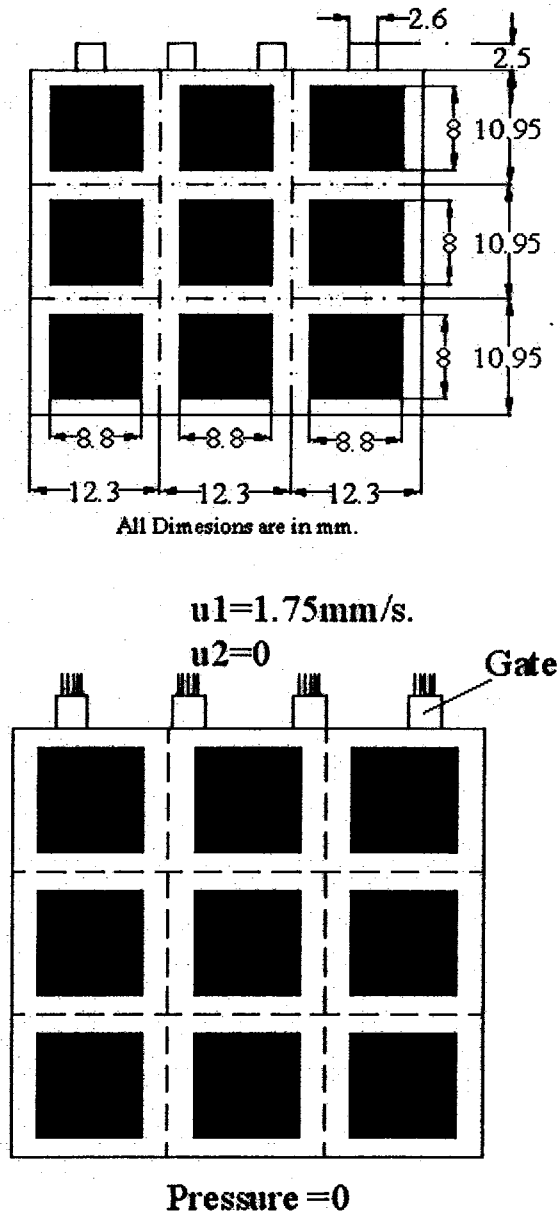


Fig 4.12 IC package dimensions and boundary conditions for multichip cavity filling.

The zero pressure condition is applied at the opposite wall since no slip condition and no shear stresses are applied. Finite element mesh is shown in Figure 4.13.

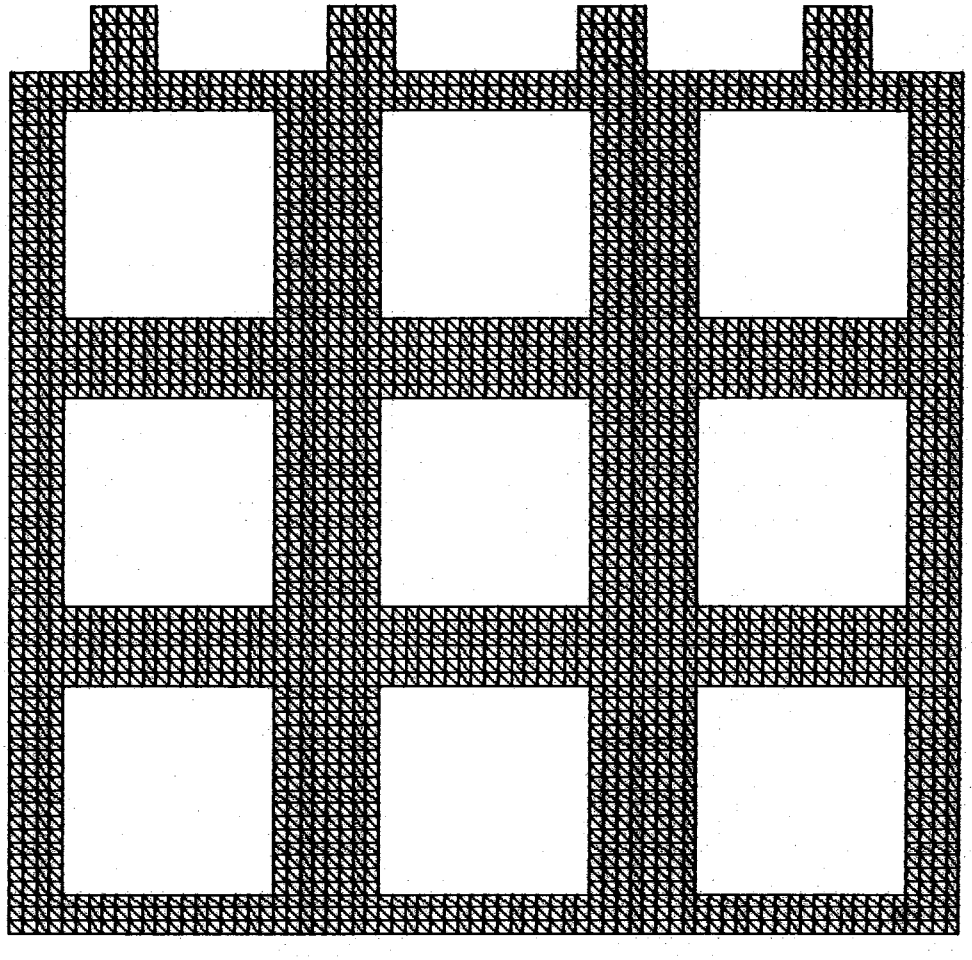
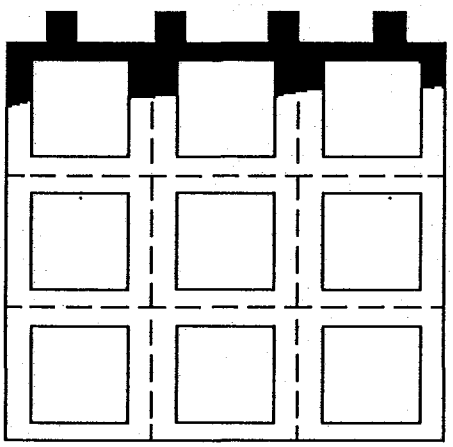
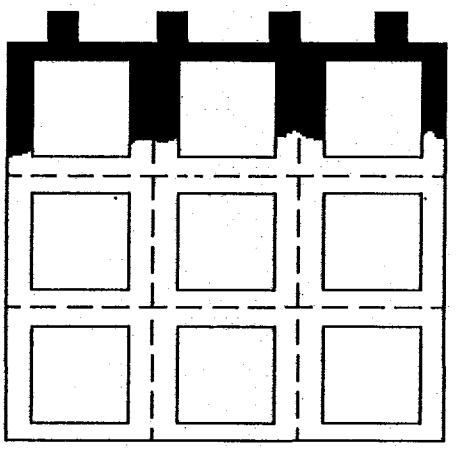


Fig 4.13 Finite Element Mesh for multichip cavity filling.

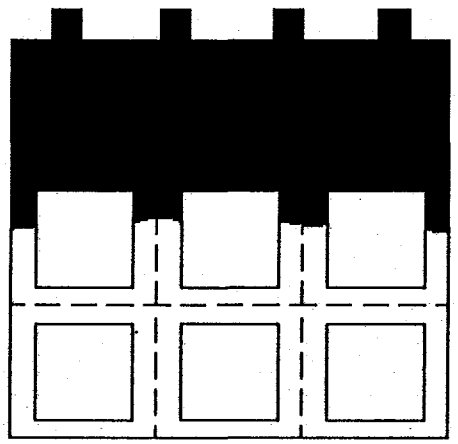
The finite element mesh consists of 4016 linear triangles and 2966 nodes The simulated front profile results for various time steps are shown in Figure 4.14.



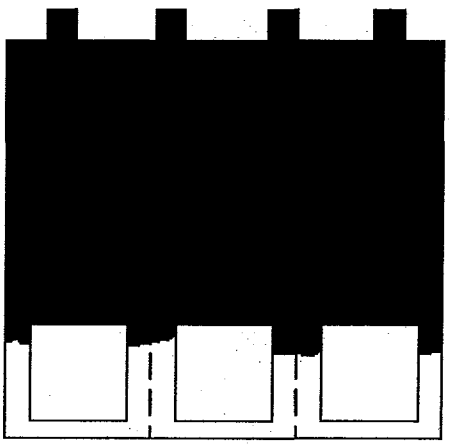
(a) t = 4s



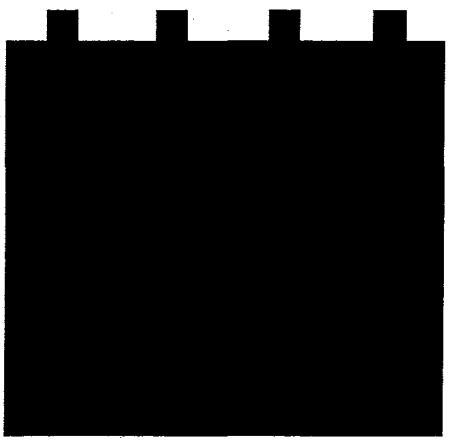
(b) t = 8s



(c) t = 12s



(d) t = 16s



(e) t =19s

Figure 4.14 Simulated Front Profiles for multichip cavity filling at time steps (a) t = 4 s, (b) t = 8 s , (c) t =12 s, (d) t =16 s and (e) t = 19 s.

4.2 Parameter studies for 2D Mold filling

The parameter studies are conducted to study the effect of process parameters on the cavity filling time. For this, the desired parameter of interest is varied while all other parameters are kept constant. In this section, the analysis is carried out to observe the fluid filling behavior in chip cavity for both single chip cavity filling and multi chip cavity filling process.

4.2.1 Parameter studies for 2D single chip cavity filling

Parameter studies are carried out for single chip cavity filling by varying the mold clearance while the gate dimensions are kept constant. Thus mold clearances are 1mm, 0.5mm, 0.2mm and 0.1mm.

Case 1: Mold clearance = 1mm.

The boundary conditions will remain the same as explained earlier. Only the finite element mesh changes. The mesh in Figure 4.15 consists of 288 nodes and 394 elements.

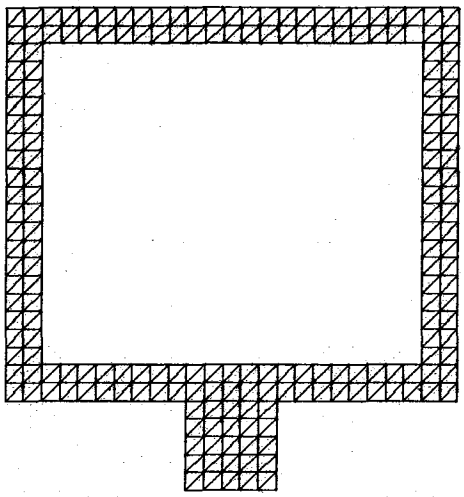


Figure 4.15 Finite Element Mesh for the case with 1mm clearance.

In this case the clearance of 1mm between the die and substrate is considered. Then the corresponding die size is 10.5 X 9mm. The epoxy molding compound takes 1.02 seconds to fill the chip cavity.

Case 2: Mold clearance = 0.5mm.

The finite element mesh consists of 210 nodes and 230 elements. The filling time in this case is 0.85 seconds.

Case 3: Mold clearance = 0.2mm.

The finite element mesh in this case consists of 1095 nodes and 1898 elements. The filling time in this case is 0.65 seconds.

Case 4: Mold clearance = 0.1mm.

This case was very crucial for simulating the mold filling results. Since the clearance between die and substrate is very small, the EMC experiences a stiff resistance for cavity filling.

A finer mesh has been adopted for mold clearance area and some what course mesh for gate compared to the clearance. The finite element mesh in this case consists of 962 nodes and 982 elements. The finite element mesh used is shown in Figure 4.16.

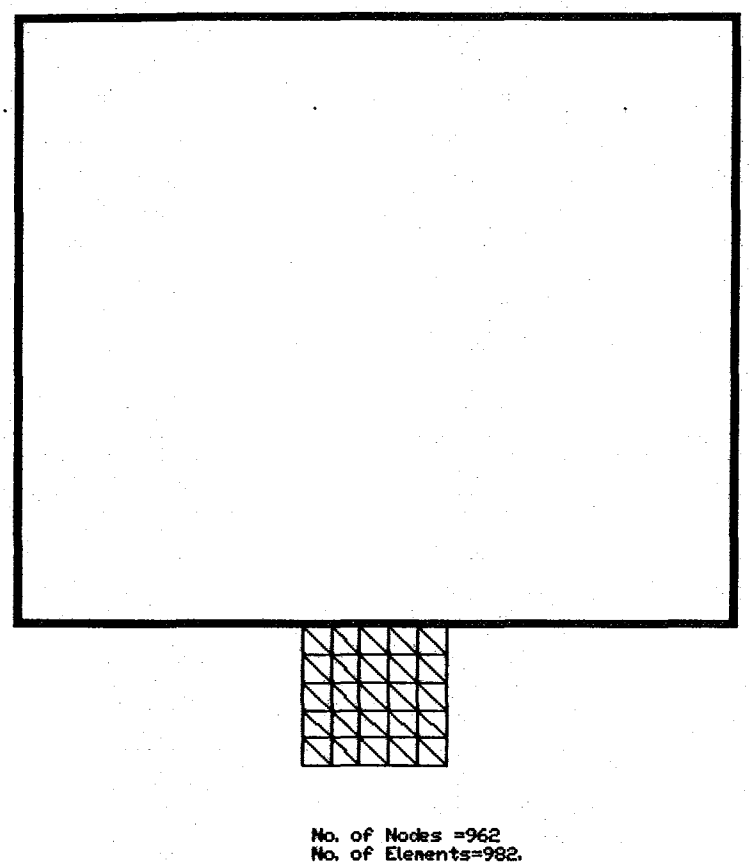
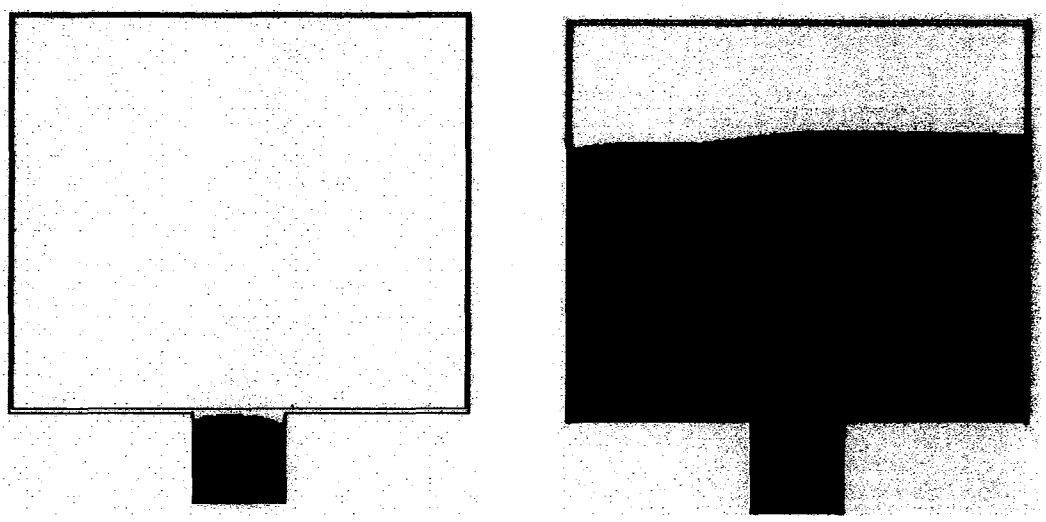


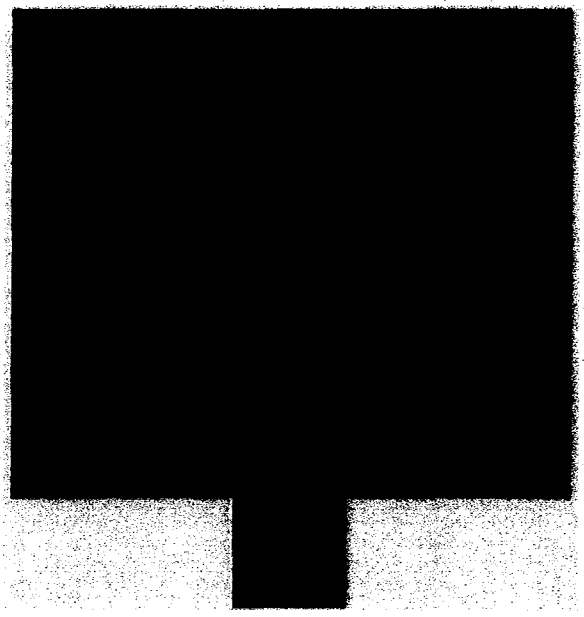
Figure 4.16. Finite Element Mesh for the case with 0.1mm clearance.

The various front profiles for different time steps are shown in Figure 4.17. The molding compound takes 0.43 seconds to fill the cavity completely.



a) Time step 't = 0.2s'. (b) Time step 't = 0.34s'.

Figure 4.17a. Front profiles for 100 micron clearance at t =0.2s and 0.34s.



c) Filling time't = 0.43s'

Figure 4.17. Front profiles for 100 micron clearance at t =0.43.

The results are summarized in the following Table 4.1.

Table 4.1. Filling time for various die sizes for single chip cavity

Sl.No.	Substrate size (in mm.)	Gate Size (in mm.)	Die Size (in mm.)	Mold clearance (in micron)	Filling Time (in seconds)
1	12.5 x 11	2.5 x 2.5	9.5 x 8	1500	1.21
2	12.5 x 11	2.5 x 2.5	10.5 x 9	1000	1.02
3	12.5 x 11	2.5 x 2.5	11.5 x 10	500	0.85
4	12.5 x 11	2.5 x 2.5	12.1x10.6	200	0.65
5	12.5 x 11	2.5 x 2.5	12.3x10.8	100	0.43

From Table 4.1, it is observed that filling time decreases as the mold clearance decreases.

4.2.2 Parameter studies for 2D multi-chip cavity filling

After confirming the capability of the present 2D algorithm for simulating the flow in a narrow region of micron size, it is then used for parametric study to simulate the flow in multichip cavity filling. Table 4.2 shows the results of parametric study (Filling time) carried out for above package.

Table 4.2. Tabulated results for 2D parametric study for a multichip package.

Sl.No	Parameters	Cavity filling time in seconds							
1	Gate width (mm)	2.6	2.4	2.2	2.0	1.8	1.6		
	Time(s)	19	20.2	22.4	24.1	26.2	27.8		
2	Gate length (mm)	2.8	2.6	2.5	2.4	2.2	2.0	1.8	1.6
	Time(s)	21.6	19	17.6	15.8	13.9	12.3	10.9	9.8
3	Velocity, u_1 (mm/s)	1.6	1.75	1.8	2.0	2.2	2.6	2.8	3.0
	Time(s)	20.5	19	18.4	17.2	15.4	13.8	12.5	11.2
4	Viscosity of the	4.5	4.3	4.0	3.8	3.6	4.6	4.8	5.0

	fluid, μ, (N.s/m^2)								
	Time(s)	19	17.6	16.3	14.1	11.8	20.2	21.3	22.5
5	Mold clearance (mm)	1.75	1.6	1.5	1.4	1.3	1.8	2.0	
	Time(s)	19	18.5	17.8	16	15.4	19.5	20.	

Table 4.2 indicates that the mold filling time decreases as the gate width is increased. Filling time varies according to the length of the gate. In other words, the filling time decreases as gate length decreases. Filling time increases with increase in viscosity value. As flow velocity is increased, the filling time decreases. Filling time decreases as the mold clearance decreases.

4.3 Three Dimensional Mold Filling

Three dimensional mold filling is analyzed using hybrid CBS- VOF technique. 4 noded tetrahedron elements are used in finite element analysis. The 3-D code is developed using commercially available MATLAB programming language (V7R14). Before analyzing the actual 3D packaging mold filling process, the code developed is verified by a known test case. The following sections cover 3D code verification and application case for 3D mold filling in an electronic package.

4.3.1 Code Verification

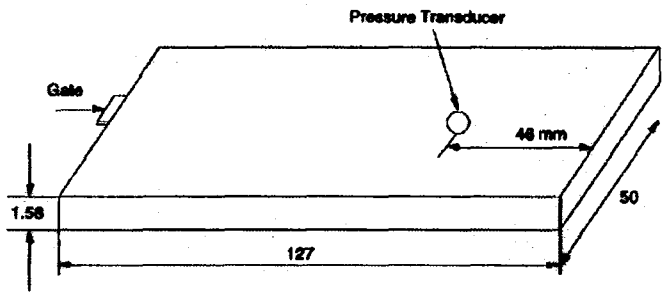
The validity of the algorithm for three dimensional is verified by considering mold filling case in a rectangular chip cavity of size 127 x 50 x 1.58mm. The chip cavity is shown in the Figure 4.18.

The properties of the materials are power law index 'n'=0.7401, viscosity =1.0e7 Pas, yield stress = 0.3771 Pa, shear rate = $300s^{-1}$, thermal conductivity = 0616 W/m-K and Cp = 1.11186 kJ/kg-K. The mold temperature was 173°C. Shear rate dependence of the viscosity is expressed by the Herhsel–Bulkey equation (Han and Wang, 2000) and is given by

$$\eta = \frac{\tau_y}{\dot{\gamma}} + K\dot{\gamma}^{(n-1)} \qquad (4.1)$$

Where τ_y is the yield stress, 'n' is power law index and $\dot{\gamma}$ is shear rate.

The dependence of viscosity on temperature was modeled using Williams-Landel-Ferry (WLF) form. The plunger pressure trace values simulated from the analysis are compared with the experimental values obtained by Han and Wang (2000). The simulation was carried out for only isothermal process. Hence energy equation is not used and temperature graphs are not plotted.



(Dimensions in mm.)

Figure 4.18 Rectangular chip cavity for 3D mold filling verification case

The simulated filling time value is compared with that obtained from the experiment. Thus the simulated filling time is 3.38s, while the experimental filling time is 3.3s. The calculated pressure traces are shown in Figure 4.19.

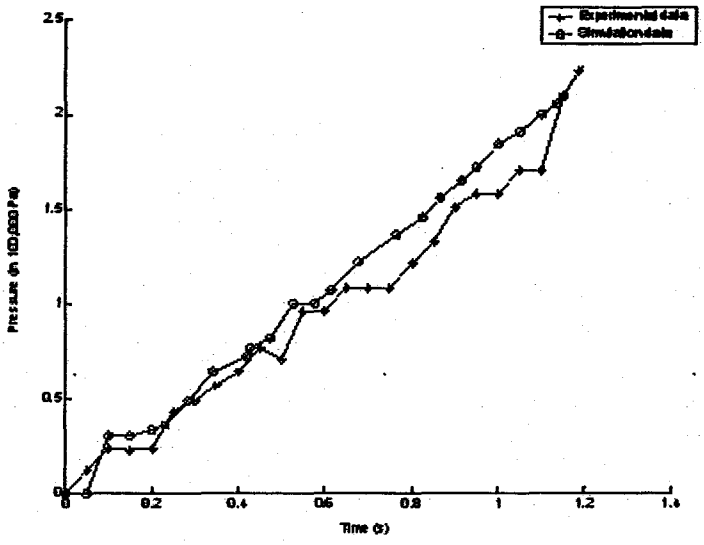


Figure 4.19 Experimental and calculated pressure traces during mold filling process

As can be seen from the Figure 4.19, pressure increases with time linearly during the filling stage because of the constant velocity of the ram. It can be seen that the agreement between experiment and simulation is good during the filling stage.

This proves the correctness of the present 3D algorithm.

4.3.2 Experimental verification for 3D Mold Filling

The proposed solution algorithm is also verified by conducting actual experiments for transfer molding process. The actual molding compound was used to investigate the flow behavior in chip cavity. The experiment was conducted at AMD/Spansion Shd Bhd engineering facility. The package considered for experimental purpose is shown in Figure 4.20. The package consists of a 12 chips of 8.3 x 6mm arranged in 4 rows on a substrate of 44 x 47 mm with 4 gates of dimensions 2.5 x 2.9 mm having film gate arrangement.

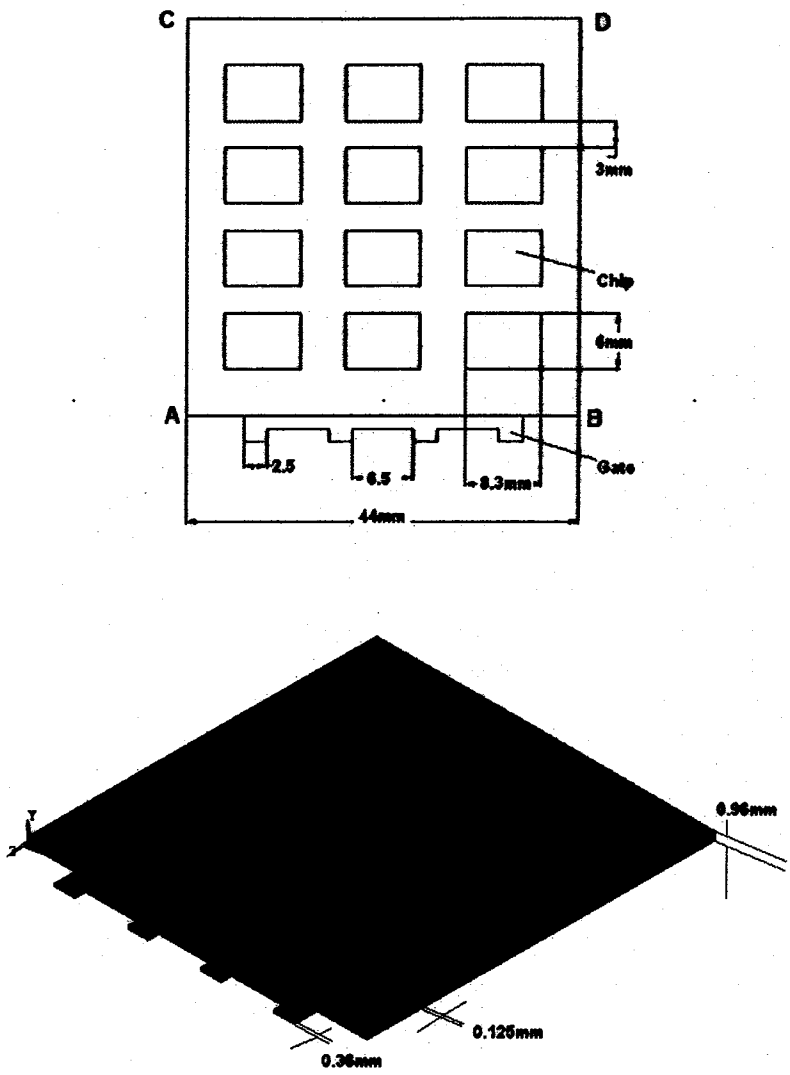


Figure 4.20. Package dimensions used in experimentation

To start the experiment, the epoxy molding compound (EMC) pallet was placed in plunger and held for couple of seconds for preheating it. During this process, the EMC was heated by mold heaters up to 178°C. After preheating, plunger activated until the stroke length set by the operator. Then large constant pressure was applied so as to avoid any air voids during the mold process and thus eventually make the molten EMC to set as a solid one. The press was held for sometime to apply packaging pressure and thus to make the epoxy compound cure to a

desired level under constant mold temperature. The package usually takes longer time to cure than that for molding. Once curing is over, the encapsulated package is ejected from the transfer molding press and is sent for further operations.

A package takes specific time and certain stroke length of the plunger to get filled completely for a specific speed of molten EMC. If one sets less stroke length, the package filling will be incomplete and the front profiles thus obtained are called "short-shots". In order to observe the mold compound filling pattern at different time steps, the short shots were taken for transfer speed of 4mm/s and the stroke length was varied as 6mm,8mm,10mm,12mm and 22mm. The plunger pressure was set as 70kg/cm^2. The experimental front profiles are then compared with the simulated profiles to verify the
correctness of the present algorithm. The simulation was run on a high performance computer with 64 bit AMD processor.

The boundary conditions used in the analysis are as below.

Inlet conditions (at gate): $u_1 = 4mm/s$, $u_2 = u_3 = 0$.

Along sides AB, BD and AC: $u_1 = u_2 = u_3 = 0$ (No slip condition).

Along the side CD: Pressure P = 0.

The finite element mesh is shown in figure 4.21. The mesh consists of 18085 elements and 38471 nodes.

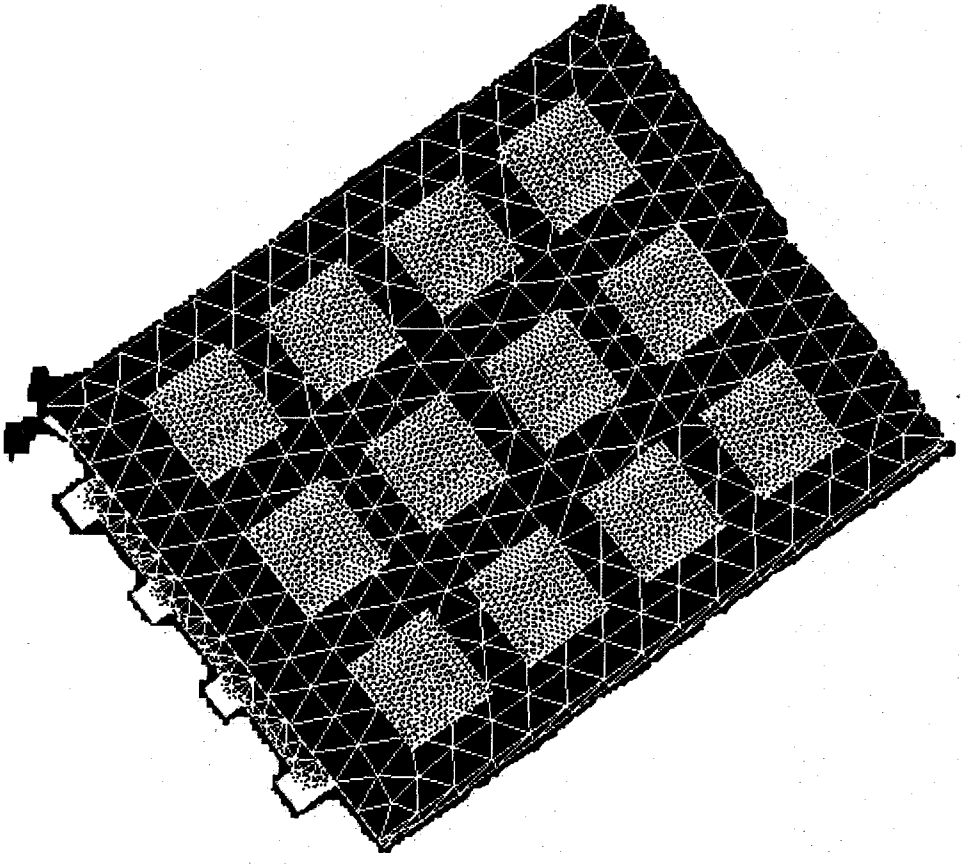


Figure 4.21. 3D Finite Element mesh with tetrahedron elements.

The Castro-Macosko equation was used to model the rheology of the molding compound in the computational model (Nguyen et.al., 2000). The model is described by the equation

$$\eta(\alpha,T) = b\exp(\frac{T_b}{T}).\left(\frac{\alpha_g}{\alpha_g-\alpha}\right)(C_1 + C_2\alpha) \qquad (4.2)$$

where η is the viscosity at the conversion α and temperature T, α_g is the conversion at the gel point, T_b is the activation energy dependent parameter and b, C_1, C_2 are the fitting constants.

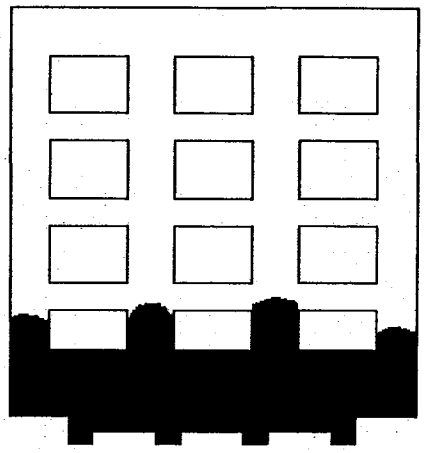
Transfer molding is more accurately approximated as an isothermal process because the temperature of the mold compound is raised rapidly to the mold temperature, then remains

constant for the reminder of the fill cycle. The values of the rheological parameters are shown in the Table 4.3.

Table 4.3 Castro- Macosko model parameters for isothermalprocessing of EMC

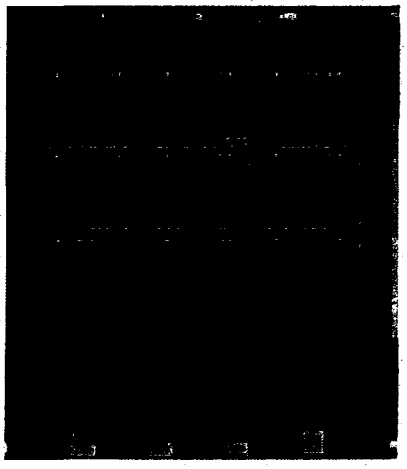
Parameter	Parameter value for Isothermal conditions
b(Pas)	$1.78X10^{-4}$
T_b (K)	5230
α_g	0.17
C_1	1.03
C_2	1.50

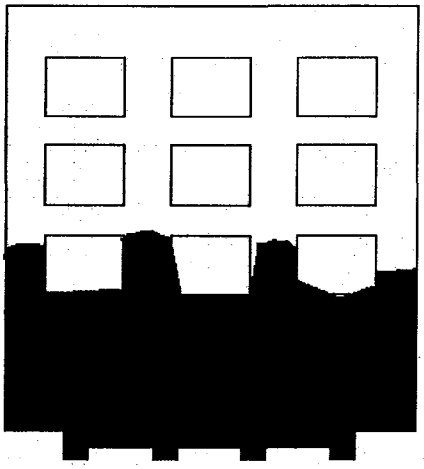
The simulated font profiles and their comparison with the experimental results are shown in the Figure 4.22.



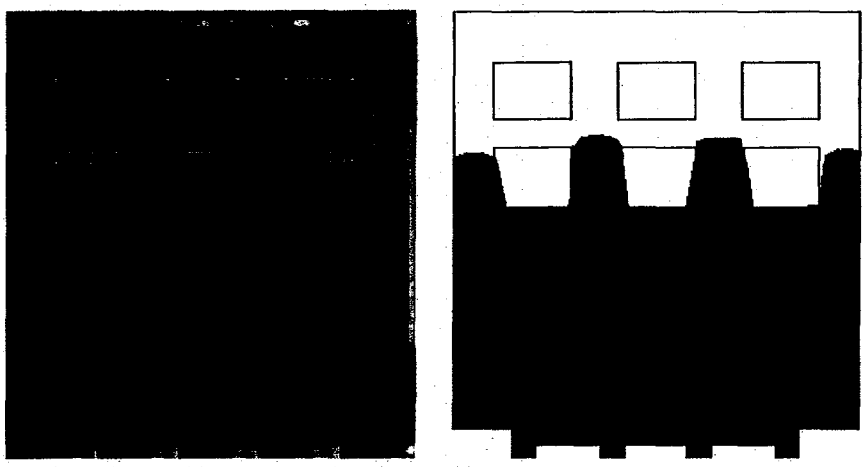
Experimental result Simulated Result

(a) Stroke = 6mm, time = 1.5s



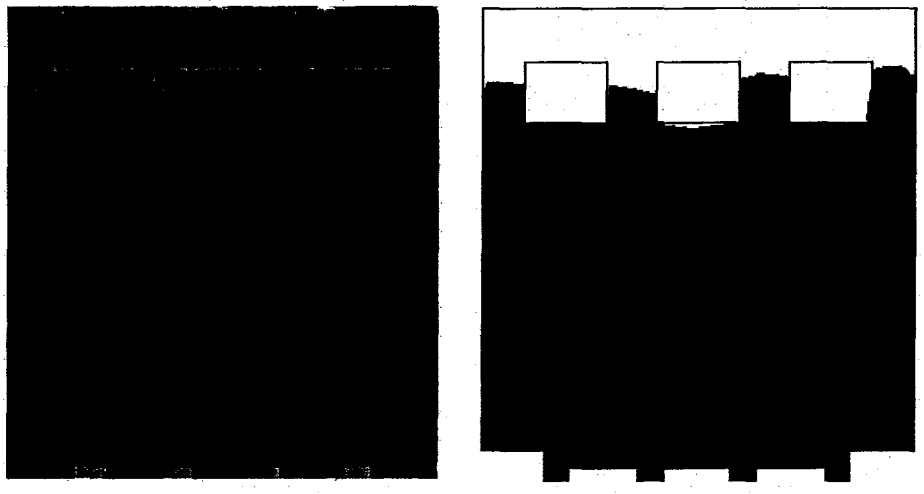


(b) Stroke = 8mm, time = 2s



(c)Stroke = 10mm, time = 2.5s

Figure 4.22a. Comparison of experimental and simulated front profiles.



(d) Stroke = 12mm, time = 3s

(e) Stroke = 22mm, time = 5.5s

Figure 4.22b. Comparison of experimental and simulated front profiles (continued).

4.3.3 Real Mold filling simulation for production setup (at velocity =3mm/s)

After validating the present methodology, it can be applied to solve the real problem of mold filling in an electronic package. Here again the same package considered in the experiment is used for mold filling simulation. The simulation is carried out for transfer speed of 3mm/s used in high volume manufacturing process of IC package encapsulation. Tetrahedron mesh used in the analysis is shown in Figure 4.23. The mesh consist of 18085 elements and 38471 nodes. The various front profiles obtained from simulation are shown in Figure 4.24.

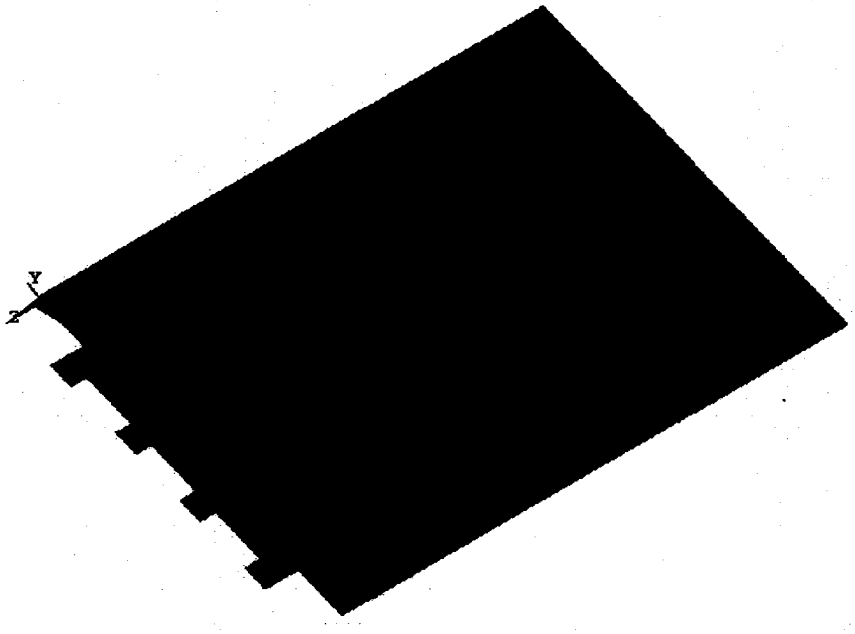
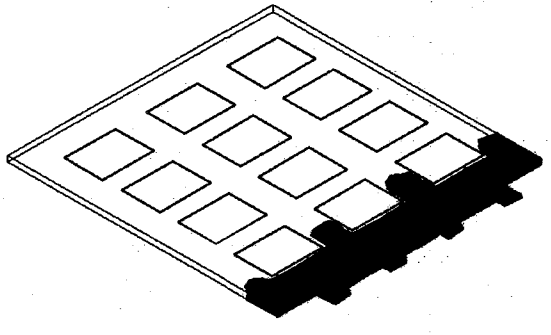
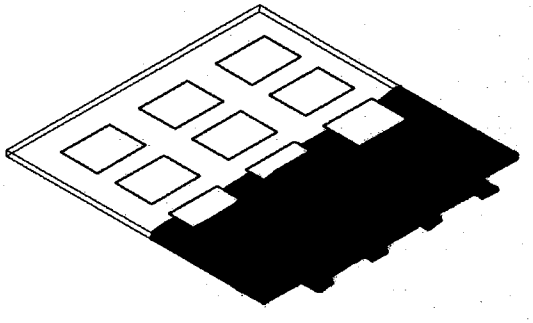


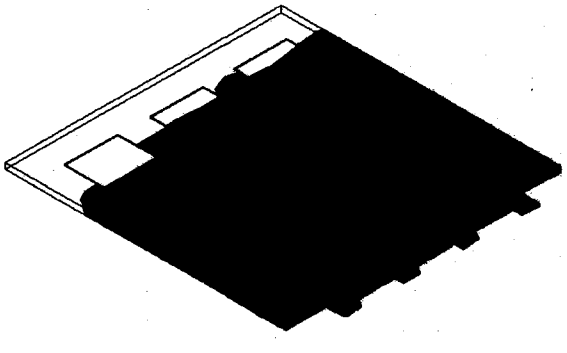
Figure 4.23. Finite element mesh with tetrahedron elements for 3D mold filling for velocity 3mm/s.

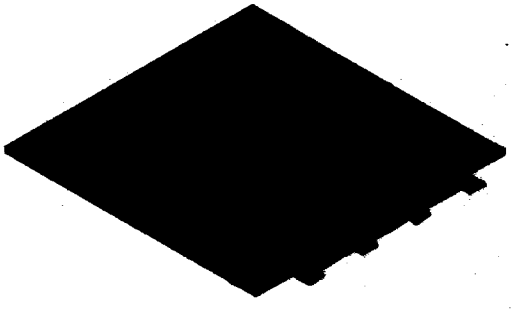


(a) Time step = 3.11s



(b) time step = 5.34s





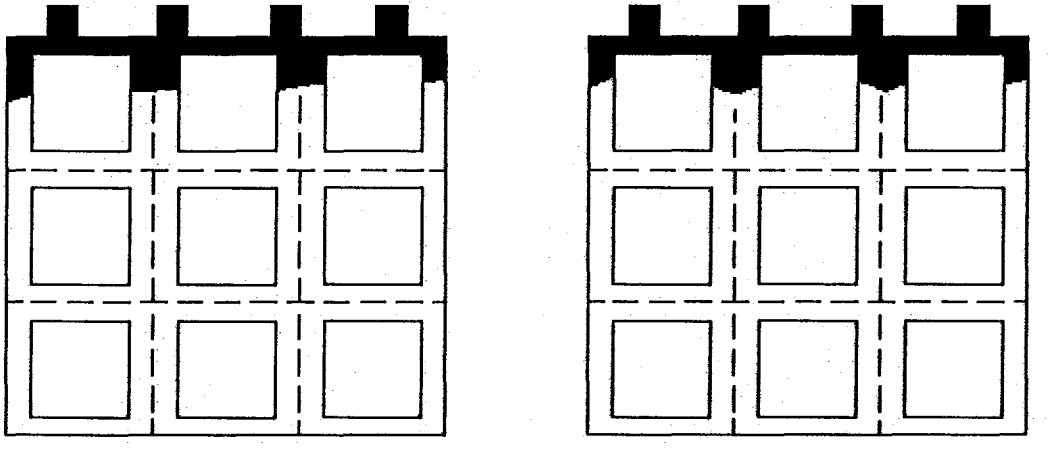
(c) Time step = 8.05s (d) Filling time step = 10.15s

Figure 4.24.Front profiles for different time steps at t = 3.11s, 5.34s, 8.05s and 10.15s.

The simulation predicts 10.15seconds to fill the package completely. The EMC in actual molding process in a factory takes 10second to fill the package and both experimental and simuated result agree very well , the error being 1.5%.

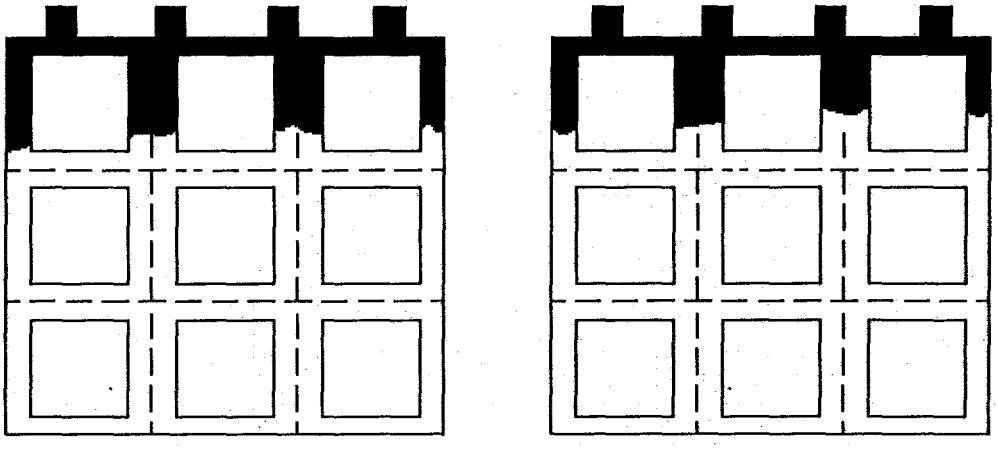
4.3.4 Comparison of 2D and 3D Simulation results.

The results of the 2D and 3D simulation were compared for the mold filling process in FTA 73 package used in 4.1.3 section under identical conditions of 1.75mm/s inlet flow velocity of EMC , 4.5Ns/m^2 viscosity and specific gravity 1.8. The compared front profiles for both the 2D and 3D results are shown in Figure 4.25.



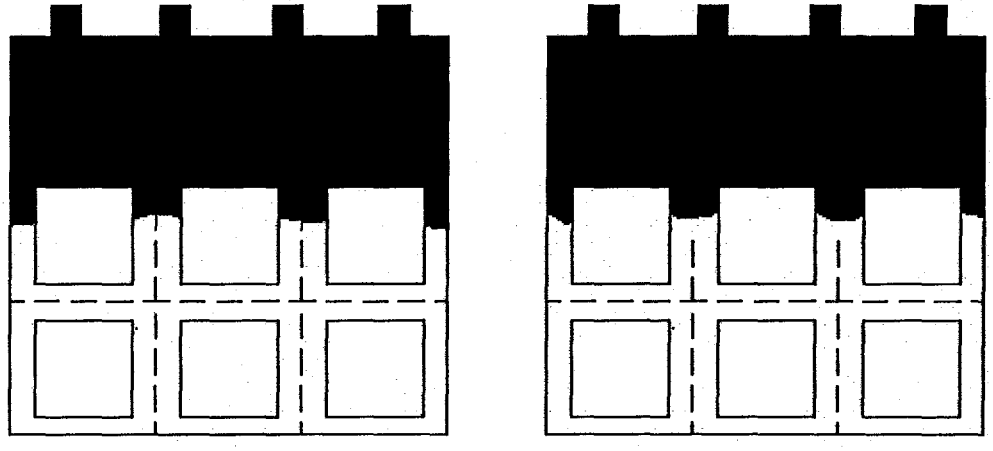
2D flow front profile 3D flow front profile

(a) time 't' = 4 seconds



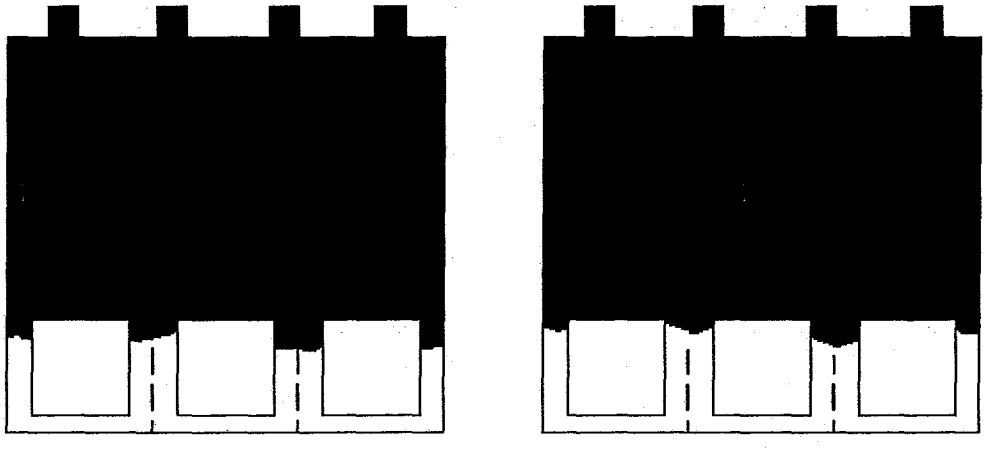
(b) t = 8s.

Figure 4.25a. Comparison of 2D and 3D flow front profiles at time steps t = 4s & 8s.

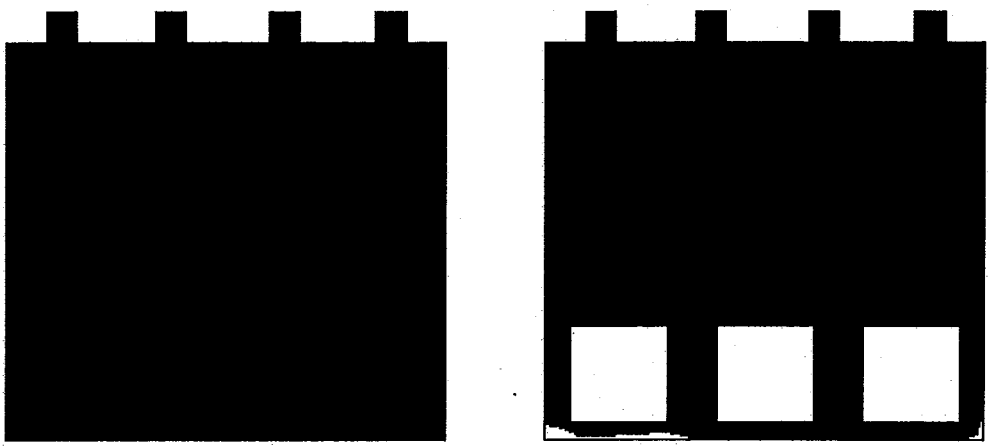


2D flow front profile 3D flow front profile

(c) t =12 s



(d) time 't' = 16 seconds



(e) time 't' = 19 seconds

Figure 4.25b. Comparison of 2D flow front and 3D flow front profiles at time steps t = 12s, 16s and 19s,

It is seen from the Figure 4.24 that the front profiles resemble each other at different time steps. But the 3D simulation results give accurate results than the 2D results considering thickness into the account. The 2D flow takes 19 seconds to fill the cavity completely but the 3D simulation gives the filling time as 19.8s. At 19s, the cavity fills completely in the 2D flow while the 3D result shows the slightly incomplete filling. One can use fairly 2D simulation for usual prediction of mold filling time. But if one wants the more accurate filling time, he/she can use the 3D flow simulation at the expense of computation time and laborious meshing.

4.4 Parameter studies for 3D Mold filling

The parameter study is conducted to know the effect of process variables on the mold filling time. The results are tabulated as shown in Table 4.4

Table 4.4. Results of parametric study for 3D flow simulation.

Sl. No	Parameters	Cavity filling time in seconds

1	Gate thickness (mm)	1.8	1.6	1.5	1.4	1.3	1.2	1.0	0.8
	Time(s)	10.35	10.27	10.15	9.95	9.87	9.78	9.63	9.51
2	Gate width (mm)	2.8	2.6	2.5	2.4	2.2	2.0	1.8	1.6
	Time(s)	9.80	9.95	10.15	10.33	10.54	10.78	11.2	11.65
3	Gate length (mm)	2.8	2.6	2.5	2.4	2.2	2.0	1.8	1.6
	Time(s)	11.2	10.53	10.15	9.8	9.63	9.54	9.45	9.37
4	Velocity, u_1 (mm/s)	5	4	3.5	3.0	2.5	2.0	1.8	1.5
	Time(s)	9.55	9.7	9.85	10.15	10.38	10.50	10.62	10.98
5	Viscosity of the fluid, μ, ($N.s/m^2$)	5.1	4.9	4.7	4.5	4.3	4.0	3.8	3.6
	Time(s)	11.2	10.70	10.43	10.15	10.0	9.85	9.60	9.40
6	Mold clearance (mm)	2.0	1.9	1.8	1.75	1.7	1.6	1.5	1.4
	Time(s)	9.65	9.8	10.0	10.15	10.22	10.45	10.75	11.21

The following conclusions are drawn for parametric studies conducted by varying process parameters.

1. Mold filling time increases as thickness and length of the gate are increased.
2. Filling time increase as viscosity of epoxy compound is increased.
3. Filling time decreases as width of the gate is increased.
4. As flow velocity increases, the filling time decreases.

4.5 Mold filling time prediction using Artificial Neural Network (ANN)

ANN is used as a tool to predict the data points within a known set of data. A total of 15 simulations have been conducted during the parametric study by varying the process parameters of mold filling process. From these simulation data, ANN is further used to predict additional data points using the trained data points. Training of the ANN for mold filling time prediction has been implemented using the feed-forward error back propagation network (EBPN) where 5 input layers, 2 hidden layers and 1 output layer have been assigned to the network. Once the network relation of ANN has been established, it is then used to predict the data of interest. The ANN simulation work has been carried out using ANN toolbox available in MATLAB programming language. The simulation will carry on until it fulfills either the iteration of 2000 epoch or when the

convergence criterion that has been set at 10^{-5} has been reached, whichever comes first.

The ANN results for mold filling time are compared with the resulats obtained from CBS/VOF method to ascertain the correctness of the ANN results. The accuracy depends upon the ANN model used, number of iterations and the convergence criterion set during the neural network training. The comparision of ANN predicted data and simulated results obtained from CBS /VOF method for gate thickness variation is

given in the table 4.5.

Table 4.5. Comparison of results for mold filling time between ANN predictions and simulation results from CBS/VOF method.

Gate length (mm)	Simulated CBS Scheme results	ANN predicted results	% Error
1.8	10.35	10.3034	0.45
1.6	10.27	10.2207	0.48
1.5	10.15	10.0992	0.50
1.4	9.95	9.9032	0.47
1.3	9.87	9.8186	0.52
1.2	9.78	9.7311	0.50
1.0	9.63	9.5837	0.48
0.8	9.51	9.4614	0.51

The comparison is also shown in figure 4.26.

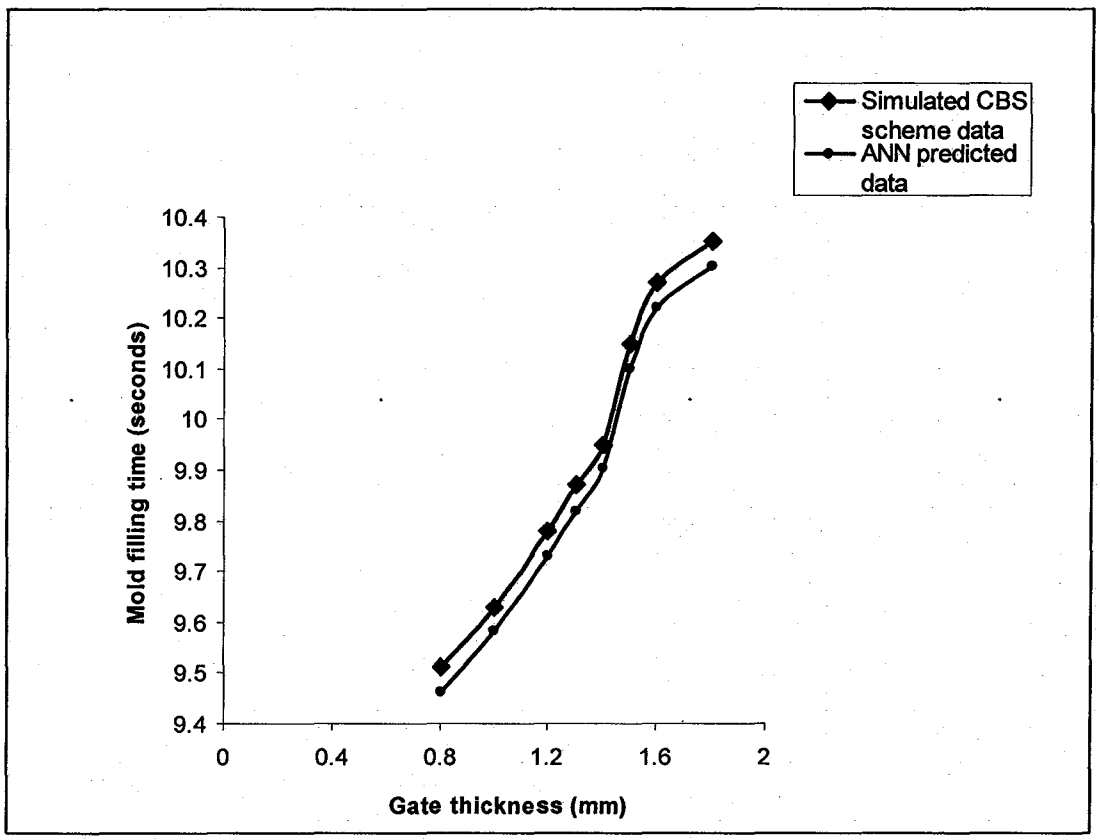


Figure 4.26. Comparison of results for mold filling time between ANN predictions and simulation results from CBS/VOF method.

It is seen from the table 4.4 that the error between the ANN predicted results and the simulated CBS/ VOF method is only 0.50 %. Thus the ANN predicts well compared to simulated results. It is observed that the ANN takes 2 minutes to predict the mold filling while simulation using CBS/VOF method takes 4hours and the savings in time and computational effort. After confirming the validity of ANN results, the trained network is used to obtain the mold filling time for the variation in the process quantity of interest like gate length, fluid velocity etc. The exhaustive data obtained from ANN is used in genetic algorithm for optimization purpose.

4.6 Optimization

The gate size of chip cavity is optimized using artificial neural network (ANN) assisted genetic algorithm (GA). Genetic algorithm is used to get the minimum filling time and the corresponding dimensions of gate gives the optimized gate size.

First, the developed hybrid CBS-VOF algorithm is used to get the filling time for given package in this case FPB 115 package. The Initial parametric study is conducted to study the effects of various process parameters by varying gate size, flow velocity, viscosity and other parameters of interest and thus package filling time is computed.

Mold filling process is transient, non-linear problem and the calculation of mold filling by numerical method such as CBS-VOF technique is laborious work and requires considerable amount of time. In order to minimize the computation time and reduce the laborious work, ANN is used to get the filling time. With the initial set of these parameters obtained by CBS-VOF method, artificial neural network (ANN) is used to study and establish the relationship between these process parameters with the mold filling time. For this, ANN is trained using error back propagation nnetwork (EBPN) model to get the non-linear relationship of filling time with process variables. One can notice the improvement in the computation time by using ANN. The normal FEM method takes roughly 4 hours including meshing, solution and post processing of results. On the other hand ANN takes 2-3 minutes to get the mold filling time for the given input of flow velocity, gate size, viscosity and density of EMC.

Once a trained ANN is established, results for parametric studies can be obtained very easily without much effort. A genetic algorithm program, available in MATLAB as took box, is used to search the input space of ANN for optimization. GA will generate the initial population of filling time for the input values of gate size 2.5 x 2.5mm, viscosity value 4.5Ns/m^2 and specific gravity of 1.5.

For genetic algorithm, the filling time is considered as fitness function or objective function with the objective of minimization of filling time. The optimization problem can be defined as

Minimize $t = f(X_1, X_2, X_3, P, \mu, \rho)$

Subjected to

$X_1, X_2, X_3 < 3.0$

$X_3 < 1.5$

$P \quad 75$

In the above formulation, 't ' represents the mold filling time, X_1, X_2, X_3 represent the gate sizes in mm and P the packing pressure in kg/cm^2. Depending upon the once experience and intuition the gate sizes are constrained to 3mm, 3mm and 1.5mm for particular package. The constrained pressure value is maximum pressure of the transfer molding machine known from the machine specifications.

For the present study, the optimization is carried out for isothermal condition with a mold temperature of 175°C. Majority of times the transfer molding is carried out at room temperature and the heating due to the flow is negligible and the process can be assumed to be isothermal one. Thus the temperature effect on the mold filling process is neglected. One can also include the temperature effect in the ANN training and temperature can be constrained to desired value. However, its effect will be minimum on the mold filling time.

The ANN helps to get exhaustive data of mold filling time for GA, which then optimizes the filling time using genetic principles. Thus the optimized filling time is 10.15s, the gate size is 2.5 X 2.5 X 1.2 mm for a maximum packing pressure of 70 kg/cm^2.

4.7 Summary

2D and 3D mold filling simulations were carried out using the proposed hybrid CBS-VOF scheme. Simulation codes were verified against the theoretical and experimental results given in literature. Experiments were conducted to for real mold filling process in industrial environment to know real fluid front profiles. The experimental front profiles were then compared with the 3D simulated results. Both agree well and thus establish the robustness of the present algorithm to solve the mold filling problem in electronic packaging. Gate size optimization has also been carried out using ANN coupled Genetic Algorithm. Based the simulation results and experimental verification, the main conclusions of this research work are present in the following chapter on 'conclusions'.

CHAPTER 5

CONCLUSIONS

5.0 Overview

Main conclusions drawn from the present research are presented under the following heads.

5.1 Highlights of the present work

5.2 Scope for future work

5.1 Highlights of the present work

The mold filling process in IC chip cavity has been analyzed using coupled characteristic based split (CBS) scheme with volume of fluid (VOF) technique. The CBS scheme, alternative to finite volume method, uses finite element method (FEM) to solve the flow governing Navier Stokes equations and thus to obtain primitive variable fields namely velocity and pressure fields. Both the 2D and 3D analyses were carried out considering the epoxy molding compound (EMC) as a generalized Newtonian fluid (GNF). EMC is assumed to be incompressible fluid since the density of highly viscous epoxy material does not changes unlike in the case of compressible flows.

The following main conclusions can be drawn from the present research work.

1. Conventionally the mold filling simulation has been modeled using Hele Shaw model for thin cavities, which neglects the thickness. But in this research work the mold filling process has been simulated using Characteristic based split algorithm, a roboust algorithm to solve complex fluid flow problem. CBS scheme has been applied for the first time in the recent history to analyse the mold filling process. It can analyse both the 2D and the 3D mold filling process.

2. The 2D analysis has been carried out using a 3 noded linear triangular element. The 2D model has been validated by comparing with the available theoretical and experimental results in the literature. Then the mold filling in single chip and multichip cavities have been analyzed. The molding compound takes 1.21 seconds to fill the single chip cavity

and 19 seconds for multichip cavity to fill completely and thus achieve balanced mold filling. The parametric study has been conducted to study the effect of variable on the mold filling time. The model is able to give good predictions.

3. The 3D model has been analyzed using 4 noded tetrahedron element. The 3D model has been verified using a literature data for pressure traces. Experiment was conducted in a local semiconductor industry to get the front profiles by varying the stroke length at constant flow velocity. These profiles were then compared with the profiles obtained from the 3D simulation model. Both results agree well and this establishes the correctness of proposed flow simulation model. Mold filling in a 3D multichip cavity has also been analyzed further as an application example. Parametric study has also been conducted to know the effect of flow velocity, viscosity and gate dimensions on the mold filling time.

4. The 2D and the 3D front profiles are compared and it is found that 3D analysis predict more accurate filling time than the 2D analysis but the difference their prediction of filling time is very small. This shows that one can reasonably predict the filling time near accurate value using a 2D model. On the other hand, the 3D simulation algorithm can get more accurate result at the cost of computation time and laborious meshing.

5. Artificial neural network (ANN) assisted genetic algorithm (GA) has been used to optimize the gate size. ANN predicts the mold filling time for a given input of flow velocity 3 mm/s and for different gate sizes. GA gives optimized gate as 2.5 x 2.5 mm.

5.2 Future Work

There is wide scope for future work in mold filling analysis for stacked die package where a number of dies are staked layer wise in vertical direction especially for flash memory packages used in mobile hand phones. The mold filling analysis data can be also used and exported for structural analysis work to predict wire sweep, warpage and delamination occurring in electronic packages. Research challenges are there to solve these critical problems of industry concern for growing miniaturization trend of IC chip industries. Research community has to strive hard in order to carryout the research on this mold filling process occurring chip manufacturing industry.

REFERENCES

Amit Kumar, Ghoshdastidar, P.S. and Muju,M.K.(2002). Computer simulation of transport processes during injection mold filling and optimization of the molding conditions. *Journal of Materials Processing Technology*, 120: 438-449.

Ashgriz, N., Poo, J.Y.(1991). FLAIR: Flux Line-Segment Model for Advection and Interface Reconstruction. *Journal of Computational Physics*, 93(2):449-468.

Brooks, A.N. and Hughes, T.J.R.(1982). Streamline upwind/Petrov-Galerkin formulation for convection dominated flows with particular emphasis on incompressible Navier –Stokes equation. Computer Methods in Applied Mechanics and Engineering, 33:199-259.

Castro, J.M. and Macosko, C. W.(1980).Kinetics and Rheology of Typical Polyurethane Reaction Injection Molding Systems. SPE Tech. Paper, 26 : 434-438.

Cerne, G., Petelin, S., and Tiselj,I. (2001).Coupling of the Interface Tracking and the Two-Fluid Models for the Simulation of Incompressible Two-Phase Flow. Journal of Computational Physics, 171:776–804.

Chang, R. Y., Yang, W. H., Chen, E., Lin, C. and Hsu, C. H. (1998). On the dynamics of air-trap in the encapsulation process of microelectronic package. *Proc. ANTEC'98 Conference.*

Chang, R.Y.,Yang, W.S., Chen, E.,Lin,C.and Hsu,C.H. On the dynamics of air trap in the encapsulation process of microelectronic package. Available: www.moldex3d.com/en/services/paper/antec/1998/0434.PDF.

Chang, R.Y. and Yang, W.H.(2001). Numerical simulation of mold filling in injection molding using a three-dimensional finite volume approach. International journal for numerical methods in fluids, 37(2):125-148.

Chang, R.Y., Yang, W.H., Hwang, S.J. and Francis Su F. (2004). Three-Dimensional Modeling of Mold Filling in Microelectronics Encapsulation Process. IEEE Transactions on Components and Packaging Technologies, 27(1), 200-209.

Chang, Y.S., Hwang, S.J., Lee, H.H. and Huang, D.Y. (2002). Study of P-V-T-C relation of EMC. *ASME Journal of Electronic Packaging,* 124: 371–373.

Chiang, H.H., Hieber, C.A. and Wang K.K.(1991a). A unified simulation of the filling and post filling stages in injection molding, Part I: Formulation, *Polymer Engineering Science*, 31:116-124.

Chiang, H.H., Hieber, C.A. and Wang K.K.(1991b), A unified simulation of the filling and post filling stages in injection molding, Part II: Experimental validation. , *Polymer Engineering Science*, 31: 125-139.

Chen, S. C., Chen, Y. C., Cheng, N.T. and Huang, M.S.(1998). Simulation of Injection-Compression Mold-Filling Process. *Int .Comm. Heat Mass Transfer.* 25(7): 907-917.

Chien, C.P., and Hwang, S.J.(2005). Prediction of Wire Sweep During the Encapsulation of IC Packaging With Wire Density Effect. *Journal of Electronic Packaging*, 127 (3):335-339.

Chern, I.L., Glimm, J., Mc Bryan, O., Plohr, B. and Yaniv, S. (1986). Front tracking for gas dynamics. *Journal of Computational Physics*, 62: 83 -110.

Chorin, A.J. (1968).Numerical solution of Navier strokes equations. *Mathematics of Computation*, 22: 745-762.

Chorin, A.J.(1980). Flame advection and propagation algorithm. *Journal of Computational Physics*, 35:1-11.

Chorin, A.J. (1985) Curvature and solidification. *Journal of Computational Physics*, 57(3): 472-490.

Codina, R., Vázquez, M. and Zienkiewicz, O.C.(1995). A fractional step method for compressible flows: boundary conditions and incompressible limit. *Proceedings of the international conference on Finite Element Fluids-New Trends and Applications,* Veneza, 15-21.

Codina, R., Vázquez, M. and Zienkiewicz, O.C.(1998).General algorithm for compressible and incompressible flows, Part III – A semi-implicit form. *International Journal of Numerical Methods in Fluids*, 27:12-32.

Comini G, Del,G.S.(1972). Finite Element solution of incompressible Navier Strokes equations. *Numerical Heat Transfer Part A*, 5: 463-473.

Donea, J. (1984). A Taylor -Galerkin method for convective transport problems.International Journal for numerical methods in fluids, 20:101-119.

Eberhart, R.C.and Dobbins, R.W. (1990) ed. *Neural network PC tools: a practical guide.* Academic Press, USA.

Fletcher, C.A.J.(1988). Computational techniques for fluid dynamics, Volume1 - Fundamentals and general techniques. Springer Verlag, 1988.

Gethin, D.T. and Abdullah, S. (1997).Finite element formulation for filling a thin section cavity. *International. Journal of Numerical Methods in Heat & Fluid Flow*, 7:344-366.

Gresho, P.M. and Sani, R.L. (1999). *Incompressible flow and the finite element method.* John Wiley & Sons, New York.

Haagh, G.A.A.V. and Van De Vosse, F.N. (1998).Simulation of Three-Dimensional Polymer Mould Filling Processes using A Pseudo-Concentration Method, 28: 1355–1369 (1998)

Han, K.H. and Im, Y.T. (1997).Compressible flow analysis of filling and post filling in injection molding with phase change effect. *Composite Structures*. 38(1-4):179-190.

Han, S. and Wang, K. K.(2000). Integrated flow analysis filling and post-filling stage of semiconductor encapsulation. *ASME J. Electronic Packaging*, 122(1):20-27.

Han, R, Shi, L. and Gupta, M.(2000) "Three-dimensional simulation of microchip encapsulation process," *Polymer Engineering. Science*, 40(3):776-785.

Harlow,F.H.and Welch,J.F.(1965). Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids*, 8; 2182-2189.

Haupt, R.L. and Haupt, S.E.(2004).Practical Genetic Algorithms. John Wiley & sons, New Jersey and Canada.

Hertz, J., Krogh, A. and Palmer, R.G. (1991). *Introduction to the theory of neural computation.* Addison-Wesley, USA.

Hieber, C. A. and Shen, S. F. (1978).Flow analysis of the non-isothermal two dimensional filling processes in injection molding. *Israel J. Tech.*, 16: 248–254.

Hieber, C.A. and Shen, S.F.(1980), A finite element/ finite difference simulation of Injection molding filling process, J. Non-Newtonian Fluid Mechanics, 7:1-32.

Henz,B.J.,Tamma,K.K.,Mohan,R.V. and Ngo,N.D.(2005).Process modeling of composites by resin transfer molding –Sensitivity analysis for non-isothermal consideations. *International Journal of Numerical Methods for Heat & Fluid Flow*, 15(7):631-653.

Hirsch, C. (1989). Numerical computation of internal and external flows.Vol.1, John Wiley, UK.

Hirt,C.W. and Nichols, B.D.(1981). Volume of Fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39: 201-225.

Hong, L.C., Hwang,S.J., Lee,H.H. and Hung D.Y.(2004). Simulation of Warpage Considering both Thermal and Cure Induced Shrinkage during Molding in IC Packaging. *4th International IEEE Conference on Polymers and Adhesives in Microelectronics and Photonics (Polytronic 2004)*, pp.78-84.

Hwang,S.J. and Yi-San Chang,Y.S.(2006). P–V–T–C Equation for Epoxy Molding Compound. IEEE Transactions on Components and Packaging Technologies, 29(1), 112-117.

Ip, K.W., Kwong, C.K. and Wong Y.W. (2003).Fuzzy regression approach to modeling transfer moulding for microchip encapsulation. *Journal of Materials Processing Technology*, 140: 147-151.

Kang,M.K.,Jung,J.J. and Lee W.I. (2000). Analysis of resin transfer moulding process with controlled multiple gates resin injection. Composites, part A ; Applied science and manufacturing. 31: 407-422.

Lau, J., Wong, C.P., Prince, J.L and Nakayama, W. (1998). *Electronic packaging design, materials, process and reliability,* McGraw Hill, New York.
Lewis,R.W., Morgan, K., Seetharamu, K.N. and Thomas, H.R.(1993).*Finite Element Method in Heat transfer analysis,* John Wiley, UK.

Lewis RW, Nithiarasu P, Seetharamu KN. *Fundamentals of the Finite Element Method for Heat and Fluid flow.* John Wiley & Sons, UK, 2004.

Mackerle, J. (1997). Finite-element analysis and simulation of polymers: a bibliography (1976–1996).Modeling and Simulation in Materials science and engineering, 5: 615–650.

Manzione, L. T. (1990). *Plastic Packaging of Microelectronic Devices.* Van Nostrand Reinhold, New York.

Mashayek, F., and Ashgriz, N. (1995). A Hybrid Finite Element -Volume of Fluid Method for Simulating Free Surface Flows and Interfaces. *International Journal of Numerical Methods in Fluids*, 20(12):1363-1380.

Massarotti,N., Nithiarasu,P. and Zienkiewicz,O.C.(1998). Characteristic-Based-Split (CBS) algorithm for incompressible flow problems with heat transfer. International Journal of Numerical Methods for Heat and Fluid Flow, 8(8): 969-990.

Miles,T.R., Rector,L.P. and Gong,S. (2000). Transfer Molding Encapsulation of Flip Chip Array Packages: Technical Developments in Material Design. www.amkor.com/products/notes_papers/TLOBI_Article1200.pdf.

Miyata, H (1986). Finite-difference simulation of breaking waves. *Journal of Computational Physics*, 65; 179 –214.

Muc, A.and Saj, P.(2004). Optimization of the reactive injection molding process. *Structural and Multidisciplinary Optimization*, 27(1-2):110-119.

Nguyen,L., Quentin,C., Bayyuk,S., Bidstrup –Allen, S.A. and Wang, S.T. (2000). Computational modeling and validation of the Encapsulation of Plastic Packages by Transfer Molding. ASME Journal of Electronic Packaging, 122:138-146.

Nguyen, L.T. (1993). Reactive flow simulation in transfer molding of IC packages. *Proceedings of 43rd Electronic Components Technology Conference*, 375–389.

Nithiarasu, P. (2002).On boundary conditions of the characteristic based split (CBS) algorithm for fluid dynamics. International Journal for Numerical Methods in Engineering,54(4): 523-536.

Nithiarasu, P. (2003). An efficient artificial compressibility (AC) scheme based on the characteristic based split (CBS) method for incompressible flows. International Journal for Numerical Methods in Engineering, 56(13):1815-1845.

Nithiarasu,P.(2004). A fully explicit characteristic based split (CBS) scheme for viscoelastic flow calculations. International Journal for Numerical Methods in Engineering, 60(5) : 949-978.

Patterson, D.W.(1998). Artificial Neural Networks - Theory and Applications. Prentice Hall PTR, New Jersey.

Pendelidis, I and Zou, Q.(1990). Optimization of injection molding design,Part I –Gate location optimization. *Polymer Engineering Science,* 30(10):873-882.

Satya Prasad, B., Seetharamu, K.N. and Sundararajan, T. (2000). Solidification simulation of castings using finite element method. A Ph.D. thesis, Department of Mechanical Engineering, IIT Madras, Chennai, India.

Rallison, J.M.and Acrivos, A. (1978). A numerical study of the deformation and burst of a viscous drop in an extensional flow. *J. of Fluid Mechanics*, 89:191-200.

Ramaswamy, B., Jue, T.C., Akin, J.E. (1992).Semi-implicit and explicit finite element schemes for coupled fluid thermal problems. *International Journal for Numerical methods in Engineering*, 34:675-696.

Segerlind LJ. *Applied Finite Element Analysis*, second edition, John Wiley and sons, 1984.

Shen,Y.K.,Chen, S.H. and Lee, H.C. (2004). Analysis of mold filling process on flip chip package. Journal of reinforced plastics and composites, 23(4):407-428.

Shen Chanyu, Zhai Ming and Gu Yuanxian (2005).An improved algorithm for the simulation of injection molding filling process. Journal of reinforced plastics and composites, 24(7):691-698).

Shin, S. and Lee, W.I. (2000).Finite element analysis of incompressible viscous flow with moving free surface by selective volume of fluid method. *International Journal of Heat and Fluid Flow*, 21:197-206.

Spalding, D.B. (1972). A novel finite difference formulation for differential equations involving both first order and second order derivatives. International Journal for Numerical Methods in Engineering, 4: 551-559.

Su,F., Hwang,S.J., Lee, H. H. and Huang,D.Y. (2000). Prediction of paddle shift via 3-D TSOP modeling," *IEEE Trans. Components and Packaging Technologies,* 23(4): 684-692.

Tong, K.W., Kwong, C.K.; Ip, K.W. (2003). Optimization of process conditions for the transfer molding of electronic packages, *Journal of Materials Processing Technology*, 138(1-3):361-365.

Tong, K.W., Kwong, C.K. and Yu, K.M. (2004). Process optimization of transfer molding for electronic packages using artificial neural networks and multi objective optimization techniques. *The International Journal of Advanced Manufacturing Technology*, 24(9-10):675-685.

Tucker III, C.L. (1989).ed. Fundamentals of Computer Modeling for Polymer Processing, Hanser Publishers, Munich Vienna New York.

Tummala, R.R. (2001). Fundamentals of Microsystems Packaging. McGraw-Hill, New York.

Turng, L. S. (1994).Computer aided engineering (CAE) for the microelectronic packaging process. *Advanced Computer Aided Engineering (CAE) Polymer Process.*, MD-49:191–208.

Wang, K.K., Shen, S.F., Cohen C, Hieber C.A., Kwon T.H., and Ricketson R.C. (1985). Computer aided design and fabrication of molds and computer control of injection molding, Progress report No.11, Cornell University , Ithaca, New York.

Yang,L., King,C.K. and Bernstein,J.B. (2003). Liquid dispensing encapsulation in semiconductor packaging. Microelectronics International, 20 (3) : 29-35.

Yeung, R.W. (1982). Numerical methods in free surface flows. *Annual Reviews of Fluid Mechanics*, 14:395-442.

Wu,J.K.(1996).Neural Networks and simulation methods. Marcel Dekkar, New York.

Zalzala, A. M. S. and Fleming, P. J. (1997) ed. *Genetic algorithms in engineering systems.* Institution of Electrical Engineers, London.

Zhao,Y. Hsiang Hui Tan and Baili Zhang (2002). A High-Resolution Characteristics-Based Implicit Dual Time-Stepping VOF Method for Free Surface Flow Simulation on Unstructured Grids. *Journal of Computational Physics*, 183, 233–273.

Zienkiewicz, O.C. and Codina, R.(1995 a). A general algorithm for compressible and incompressible flow, Part I -The split characteristic based scheme. *Int. J. Numerical Methods in Fluids*, 20: 869-85.

Zienkiewicz, O.C., Satya Sai, B.V.K., Morgan, K., Codina, R. and Vázquez, M.(1995 b).A general algorithm for compressible and incompressible flow, Part II - Tests on the explicit form. *International Journal of Numerical Methods in fluids*, 20: 887-913.

Zienkiewicz, O.C., Satya Sai, B.V.K., Morgan, K. and Codina, R.(1996). Split characteristic based semi-implicit algorithm for laminar/turbulent incompressible flows. *International Journal of Numerical Methods in Fluids*, 23:787-809.

Zienkiewicz, O.C., Nithiarasu,P., Codina, R., Vázquez, M. and Ortiz(1999). An efficient and accurate algorithm for fluid mechanics problems-the characteristic based split (CBS) algorithm. International journal for numerical methods in fluids, 31,359-392.

Zienkiewicz,O.C. and Taylor,R.L. (2000). The Finite Element Method, Vol.3- Fluid Dynamics. Butterworth and Henemann,UK.

Appendix B

Venkatesh M. Kulkarni, K. N. Seetharamu, Ishak Abdul Azid, P. A. Aswatha Narayana¶and Ghulam Abdul Quadir, "Numerical simulation of underfill encapsulation process based on characteristic split method", Int. J. Numer. Meth. Engng 2006; 66:1658–1671

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2006; **66**:1658–1671
Published online 12 April 2006 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme.1704

Numerical simulation of underfill encapsulation process based on characteristic split method

Venkatesh M. Kulkarni[1,‡], K. N. Seetharamu[2,*,†], Ishak Abdul Azid[1,§], P. A. Aswatha Narayana[2,¶] and Ghulam Abdul Quadir[3,‖]

[1]*School of Mechanical Engineering, Universiti Sains Malaysia, Engineering Campus, 14300 Nibong Tebal, Penang, Malaysia*
[2]*Centre for Fluid Flow and Energy Systems Research, M.S. Ramaiah School of Advanced Studies, Gnanagangothri Campus, New BEL Road, Bangalore 560 054, India*
[3]*Department of Mechanical Engineering, Al-Falah School of Engineering and Technology, Village Dauj, Faridabad 121 004, Haryana, India*

SUMMARY

Electronic packaging protects the integrated circuit chip from environmental and mechanical damages. Underfilling encapsulation is an electronic packaging technology used to reinforce the solder joints between chip and the substrate. For better mould design and optimization of the process, flow analysis during the encapsulation process is the first necessary step. This paper focuses on the study of fluid flow in underfilling encapsulation process as used in electronics industry. A two-dimensional numerical model was developed to simulate the mould filling behaviour in underfilling encapsulation process. The analysis was carried out by writing down the conservation equations for mass, momentum and energy for a two-dimensional flow in an underfilling area. The governing equations are solved using characteristic based split (CBS) method in conjunction with finite element method to get the velocity and pressure fields. The velocity field was used in pseudo-concentration approach to track the flow front. Pseudo-concentration is based on the volume of fluid (VOF) technique and was used to track fluid front for each time step.

A particular value of the pseudo-concentration variable was chosen to represent the free fluid surface which demarcates mould compound region and air region. Simulation has been carried out for a particular geometry of a flip-chip package. The results obtained are in good agreement with the available numerical and experimental values and thus demonstrate the application of the present numerical model for practical underfilling encapsulation simulations. Copyright © 2006 John Wiley & Sons, Ltd.

KEY WORDS: simulation; underfill encapsulation; characteristic based split (CBS) method; finite element method (FEM); volume of fluid technique (VOF); front tracking

*Correspondence to: K. N. Seetharamu, Centre for Fluid Flow and Energy Systems Research, M.S. Ramaiah School of Advanced Studies, Gnanagangothri Campus, New BEL Road, Bangalore 560 054, India.
†E-mail: knseetharamu@yahoo.com
‡E-mail: vmkulkarni1@yahoo.com
§E-mail: ishak@eng.usm.my
¶E-mail: paa_iitm@yahoo.co.in
‖E-mail: gaquadir@rocketmail.com

Received 17 March 2005
Revised 12 December 2005
Accepted 8 February 2006

1. INTRODUCTION

The continuing trend towards miniaturization of electronic equipment has revolutionized the assembly and packaging of electronic components. More user friendly and wider variety of functions are to be provided in these electronic devices. This means that more interconnections will have to be made in less space. Electronic packaging plays an important role to protect these active electronic components from electrical, mechanical and chemical hazards. The packaging also serves the functions of heat dissipation, signal timing, signal and power distribution [1]. Depending upon the type of the package material used for silicon integrated circuits, electronic packages are classified as ceramic packages, refractory glass packages and plastic packages. Plastic packaging is efficient since the assembly and process operations are conducted on lead frames, which typically cluster four to twelve devices together [1]. This configuration facilitates handling and makes the process more amenable to automation, thereby helping to keep costs low and market share high. Plastic packaging utilizes a metal lead frame and metallized circuit pattern to mount the silicon device and a fanout pattern of leads to the pins of the package. Moulded plastic material is used to protect the chip and lead frame from physical damage and contamination. Encapsulation is a packaging technique, provides both chemical and mechanical protection of the integrated circuit (IC) and is typically done by means of low temperature polymers. Present encapsulation techniques are based on flip-chip process, where the silicon die is attached to the package substrate and electrically connected through an array of solder bumps.

Underfilling encapsulation is used to reinforce the solder joints between chip and the substrate. Due to large difference in thermal expansion (CTE) between the silicon die and package substrates, large stresses are developed in interconnects during temperature cycling and normal chip operations [2]. To reduce these stresses, the stand-off region between die and package is encapsulated with epoxy moulding compound (EMC) using underfilling encapsulation process. During underfilling encapsulation process, polymers driven by capillary action are filled at a low speed between the IC chip and the substrate. For better mould design and optimization of the process, flow analysis during the encapsulation process is the first necessary step. In this paper, a general simulation algorithm based on characteristic based split algorithm to model the fluid flow in underfilling process has been established. It simulates the mould filling behaviour in the chip cavity. The flow analysis is carried out using a CBS assisted FEM/VOF hybrid technique to simulate the flow in a mould cavity. The paper highlights the importance of CBS scheme, its implementation and volume of fluid technique for front tracking.

The development of simulation capabilities for the analysis of underfill encapsulation process is of primary importance to understand the effects of the various process variables in the final underfill properties. Turng and Wang [3], Nguyen [4] and Tan *et al.* [5] have attempted to analyse the flow during the encapsulation process numerically. Daniel *et al.* [6] have presented large-scale numerical models for underfill encapsulation process. The numerical formulation consists of the finite element method coupled with the volume of fluid technique and is based on generalized Hele–Shaw equations. The flow of the encapsulant during the underfill encapsulation of flip-chips was studied by Sejin and Wang [7]. They developed models for both capillary driven encapsulation and forced injection encapsulation processes. The numerical analysis used a finite element method based on Hele–Shaw method for solving the flow field. Han and Wang [8] have analysed the fluid flow in a chip cavity during semiconductor encapsulation. The behaviour of the epoxy moulding compound was modelled by assuming the flow to be

generalized Newtonian third. They used Hele–Shaw approximation to analyse the flow in the chip cavity more accurately, particularly to model the flow through the openings in the lead frames.

Sejin and Wang [9] have also developed a process to study the pressurized underfill encapsulation of flip-chips. The process used a special mould to surround the chip to be encapsulated and injects the encapsulated material at elevated pressure. This pressurized encapsulation process reduced the fill time, was able to perform the encapsulation at room temperature, filled the cavity completely without any voids and increased the capability of handling viscous encapsulants relative to the customary dispensing process for the particular case used in the experiment.

The present trend to produce faster, smaller and cheaper electronics products is driving the packaging technology towards higher packaging density with thinner and smaller profile. This makes the plastic encapsulation process much more complicated and difficult to analyse. The conventional Hele–Shaw approximation is inadequate to analyse such a complex process. So the multipurpose solution algorithm such as characteristic based split method can be used to solve the problems with complex nature.

In this paper, characteristic based split (CBS) scheme is used to solve field equations. The characteristic based spilt method has advantages over other methods in respect to CPU time and accuracy. It takes 40% less CPU time compared to other schemes. Typically the moulding compound is modelled as a non-Newtonian fluid and treating EMC as a generalised Newtonian fluid (GNF) modified cross model with Arrhenius temperature dependence is adopted in the viscosity model. But in this paper, the flow is modelled by assuming Newtonian fluid. A part of this paper was presented in a conference [10]. Figures are reprinted with permission © 2004 IEEE. Velocity and pressure fields are thus obtained from CBS scheme, which uses finite element method to solve the governing partial differential equations. The velocity field is then used in pseudo-concentration algorithm to track the fluid front. The pseudo-concentration method is based on volume of fluid (VOF) technique.

2. ANALYSIS

The governing equations are written as

(i) *Continuity equation*:

$$\frac{\partial \mathbf{u}_1}{\partial \mathbf{x}_1} + \frac{\partial \mathbf{u}_2}{\partial \mathbf{x}_2} = 0 \tag{1}$$

(ii) *Momentum equations*:

x_1-Momentum equation:

$$\frac{\partial \mathbf{u}_1}{\partial \mathbf{t}} + \mathbf{u}_1 \frac{\partial \mathbf{u}_1}{\partial \mathbf{x}_1} + \mathbf{u}_2 \frac{\partial \mathbf{u}_1}{\partial \mathbf{x}_2} = -\frac{1}{\rho} \frac{\partial \mathbf{p}}{\partial \mathbf{x}_1} + \nu \left(\frac{\partial^2 \mathbf{u}_1}{\partial \mathbf{x}_1^2} + \frac{\partial^2 \mathbf{u}_1}{\partial \mathbf{x}_2^2} \right) \tag{2}$$

x_2-Momentum equation:

$$\frac{\partial \mathbf{u}_2}{\partial \mathbf{t}} + \mathbf{u}_1 \frac{\partial \mathbf{u}_2}{\partial \mathbf{x}_1} + \mathbf{u}_2 \frac{\partial \mathbf{u}_2}{\partial \mathbf{x}_2} = -\frac{1}{\rho} \frac{\partial \mathbf{p}}{\partial \mathbf{x}_2} + \nu \left(\frac{\partial^2 \mathbf{u}_2}{\partial \mathbf{x}_1^2} + \frac{\partial^2 \mathbf{u}_2}{\partial \mathbf{x}_2^2} \right) \tag{3}$$

(iii) *Energy equation*:

$$\frac{\partial \mathbf{T}}{\partial \mathbf{t}} + \mathbf{u}_1 \frac{\partial \mathbf{T}}{\partial \mathbf{x}_1} + \mathbf{u}_2 \frac{\partial \mathbf{T}}{\partial \mathbf{x}_2} = \alpha \left(\frac{\partial^2 \mathbf{T}}{\partial \mathbf{x}_1^2} + \frac{\partial^2 \mathbf{T}}{\partial \mathbf{x}_2^2} \right) \tag{4}$$

2.1. Characteristic based split method

The characteristic based split (CBS) method [11] is used to solve the real convection–diffusion equations. The CBS scheme is similar to the original Chorin split method [12], which was used in finite difference method. By introducing characteristic Galerkin procedure and the split in momentum equations, the method becomes more stable and can be used to solve real flow problems of both compressible and incompressible nature. For most of the fluid flow applications, the fluid can be considered as incompressible and the Navier–Stokes equations represent the mathematical model of the reality.

The characteristic based split (CBS) method [13–17] for incompressible flows is a variant of velocity correction method reported by many authors [18–20]. However, the CBS procedure is efficient and flexible due to many extra provisions to improve stability and accuracy of incompressible flow calculations. The CBS method has been shown to be applicable to a wide variety of fluid dynamics problems ranging from incompressible flow to hypersonic flow.

The CBS method consists of three basic steps. In the first step the pressure term from the momentum equation is dropped and an intermediate velocity is calculated. In the second step, the pressure is obtained from a pressure Poisson equation and finally intermediate velocities are corrected to get the actual velocity values. Any additional scalar quantities such as temperature and concentration can be added as a fourth step.

2.1.1. Time discretization. For temporal discretization of the CBS scheme [11], a semi-implicit characteristic Galerkin finite element method is used. The governing equations (i.e. momentum and energy) are discretized first in time according to a Taylor's series prior to the Galerkin spatial discretization. In characteristic Galerkin method, the temporal derivative is discretized along the characteristic, where the equation is self-adjoint in nature.

By implementing the above-mentioned steps, the solution to the convection–diffusion equations can be obtained easily. The three basic steps in their semi-discrete form can be written as

Step 1. *Calculation of intermediate velocity*:

Intermediate x_1 *momentum equation*:

$$\frac{\tilde{\mathbf{u}}_1 - \mathbf{u}_1^n}{\Delta \mathbf{t}} = -\mathbf{u}_1 \frac{\partial \mathbf{u}_1^n}{\partial \mathbf{x}_1} - \mathbf{u}_2 \frac{\partial \mathbf{u}_1^n}{\partial \mathbf{x}_2} + \nu \left(\frac{\partial^2 \mathbf{u}_1}{\partial \mathbf{x}_1^2} + \frac{\partial^2 \mathbf{u}_1}{\partial \mathbf{x}_2^2} \right)^n - \mathbf{u}_1 \frac{\Delta \mathbf{t}}{2} \frac{\partial}{\partial \mathbf{x}_1} \left[\mathbf{u}_1 \frac{\partial \mathbf{u}_1}{\partial \mathbf{x}_1} + \mathbf{u}_2 \frac{\partial \mathbf{u}_1}{\partial \mathbf{x}_2} \right]^n$$

$$-\mathbf{u}_2 \frac{\Delta \mathbf{t}}{2} \frac{\partial}{\partial \mathbf{x}_2} \left[\mathbf{u}_1 \frac{\partial \mathbf{u}_1}{\partial \mathbf{x}_1} + \mathbf{u}_2 \frac{\partial \mathbf{u}_1}{\partial \mathbf{x}_2} \right]^n \tag{5}$$

Intermediate x_2 momentum equation:

$$\frac{\tilde{u}_2 - u_2^n}{\Delta t} = -u_1 \frac{\partial u_2^n}{\partial x_1} - u_2 \frac{\partial u_2^n}{\partial x_2} + \nu \left(\frac{\partial^2 u_2}{\partial x_1^2} + \frac{\partial^2 u_2}{\partial x_2^2} \right)^n - u_2 \frac{\Delta t}{2} \frac{\partial}{\partial x_1} \left[u_1 \frac{\partial u_2}{\partial x_1} + u_2 \frac{\partial u_2}{\partial x_2} \right]^n$$
$$- u_2 \frac{\Delta t}{2} \frac{\partial}{\partial x_2} \left[u_1 \frac{\partial u_2}{\partial x_1} + u_2 \frac{\partial u_2}{\partial x_2} \right]^n \tag{6}$$

Step 2. *Pressure calculation*:

$$\frac{1}{\rho} \left(\frac{\partial^2 p}{\partial x_1^2} + \frac{\partial^2 p}{\partial x_2^2} \right)^n = \frac{1}{\Delta t} \left(\frac{\partial \tilde{u}_1}{\partial x_1} + \frac{\partial \tilde{u}_2}{\partial x_2} \right) \tag{7}$$

Step 3. *Velocity correction*:

$$\frac{u_1^{n+1} - \tilde{u}_1}{\Delta t} = -\frac{1}{\rho} \frac{\partial p^n}{\partial x_1} - u_1 \frac{\Delta t}{2} \frac{\partial}{\partial x_1} \left(\frac{1}{\rho} \frac{\partial p}{\partial x_1} \right)^n - u_2 \frac{\Delta t}{2} \frac{\partial}{\partial x_2} \left(\frac{1}{\rho} \frac{\partial p}{\partial x_1} \right)^n \tag{8}$$

$$\frac{u_2^{n+1} - \tilde{u}_2}{\Delta t} = -\frac{1}{\rho} \frac{\partial p^n}{\partial x_2} - u_1 \frac{\Delta t}{2} \frac{\partial}{\partial x_1} \left(\frac{1}{\rho} \frac{\partial p}{\partial x_1} \right)^n - u_2 \frac{\Delta t}{2} \frac{\partial}{\partial x_2} \left(\frac{1}{\rho} \frac{\partial p}{\partial x_1} \right)^n \tag{9}$$

Step 4. *Temperature calculation*:

$$\frac{T^{n+1} - T^n}{\Delta t} = -u_1 \frac{\partial T^n}{\partial x_1} - u_2 \frac{\partial T^n}{\partial x_2} + \alpha \left(\frac{\partial^2 T}{\partial x_1^2} + \frac{\partial^2 T}{\partial x_2^2} \right)^n - u_1 \frac{\Delta t}{2} \frac{\partial}{\partial x_1} \left[u_1 \frac{\partial T}{\partial x_1} + u_2 \frac{\partial T}{\partial x_2} \right]^n$$
$$- u_2 \frac{\Delta t}{2} \frac{\partial}{\partial x_2} \left[u_1 \frac{\partial T}{\partial x_1} + u_2 \frac{\partial T}{\partial x_2} \right]^n \tag{10}$$

2.1.2. Spatial discretization. Galerkin finite element method [21, 22] is used in spatial discretization. Linear interpolation functions are assumed in the analysis. The spatial variation for a linear triangular element may be written as below

$$\begin{aligned} u_1 &= N_i u_{1i} + N_j u_{1j} + N_k u_{1k} = [\mathbf{N}]\{\mathbf{u_1}\} \\ u_2 &= N_i u_{2i} + N_j u_{2j} + N_k u_{2k} = [\mathbf{N}]\{\mathbf{u_2}\} \\ p &= N_i p_i + N_j p_j + N_k p_k = [\mathbf{N}]\{\mathbf{p}\} \\ T &= N_i T_i + N_j T_j + N_k T_k = [\mathbf{N}]\{\mathbf{T}\} \end{aligned} \tag{11}$$

Shape functions are derived using the method explained in Reference [22]. The element convection, diffusion and other matrices are obtained using the above interpolation functions. Then element matrices are assembled to get assembled matrices. The above steps of the CBS scheme are written in matrix form as below.

Step 1. *Intermediate velocity calculation*:
x_1-*Component*:

$$[\mathbf{M}]\frac{\Delta\{\tilde{\mathbf{u}}_1\}}{\Delta t} = -[\mathbf{C}]\{\mathbf{u}_1\}^n - [\mathbf{K}_m]\{\mathbf{u}_1\}^n - [\mathbf{K}_s]\{\mathbf{u}_1\}^n + \mathbf{f}_1 \tag{12}$$

x_2-*Component*:

$$[\mathbf{M}]\frac{\Delta\{\tilde{\mathbf{u}}_2\}}{\Delta t} = -[\mathbf{C}]\{\mathbf{u}_2\}^n - [\mathbf{K}_m]\{\mathbf{u}_2\}^n - [\mathbf{K}_s]\{\mathbf{u}_2\}^n + \mathbf{f}_2 \tag{13}$$

Step 2. *Pressure calculation*:

$$[\mathbf{K}]\{\mathbf{p}\}^n = -\frac{1}{\Delta t}[[\mathbf{G}_1]\{\tilde{\mathbf{u}}_1\} + [\mathbf{G}_2]\{\tilde{\mathbf{u}}_2\}] + \mathbf{f}_3 \tag{14}$$

Step 3. *Velocity correction*:

$$[\mathbf{M}]\{\mathbf{u}_1\}^{n+1} = [\mathbf{M}]\{\tilde{\mathbf{u}}_1\} - \Delta t[\mathbf{G}_1]\{\mathbf{p}\}^n \tag{15}$$

$$[\mathbf{M}]\{\mathbf{u}_2\}^{n+1} = [\mathbf{M}]\{\tilde{\mathbf{u}}_2\} - \Delta t[\mathbf{G}_2]\{\mathbf{p}\}^n \tag{16}$$

Step 4. *Temperature calculation*:

$$[\mathbf{M}]\frac{\Delta\{\mathbf{T}\}}{\Delta t} = -[\mathbf{C}]\{\mathbf{T}\}^n - [\mathbf{K}_t]\{\mathbf{T}\}^n - [\mathbf{K}_s]\{\mathbf{T}\}^n + \mathbf{f}_4 \tag{17}$$

The above four steps are the cornerstones of the CBS scheme for the solution of convection equations. In this paper, isothermal process is assumed and hence Equation (17) can be neglected in the analysis.

2.2. *Volume of fluid technique*

Free surface flows and interfaces between two immiscible fluids or materials with different phases are observed in many natural and industrial processes. Different numerical techniques are developed to simulate these flows. However, due to the complexity of the problem, each technique is tailored to a particular category of flows. For instance, boundary integral techniques [23, 24] are mainly used for simulating inviscid irrotational flows. Finite element methods (FEM) and finite difference methods (FDM) are potentially applicable to generalized Navier–Stokes equations. However, they have to be coupled with a technique to track the advecting fluid boundaries and interfaces. The difficulty in the interface tracking is inherently related to the complexity of its topology. Therefore, techniques which can handle small surface deformations fail when applied to large interface distortions. For the simulation of the former category of flows, FEM is more popular.

More versatile and robust category of techniques for free surface flow modelling are the front tracking methods. Here an extra set of parameters is used to trace the fluid boundaries. Front tracking techniques are divided into two groups: surface-tracking and volume tracking methods. In general, the former class gives a more accurate description of the free surface, but the latter class can handle complicated liquid regions more easily. In surface tracking methods the position of the free surface is described in a direct way; either by specifying a set of marker points located on the free surface [25], or by introducing a height function which

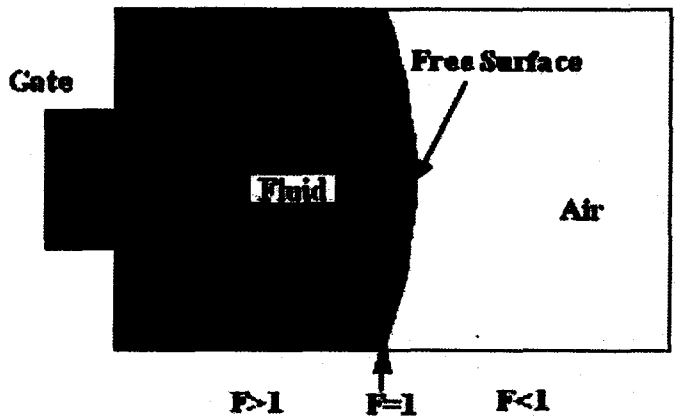


Figure 1. Representation of the free surface for pseudo-concentration Function $F(x_1, x_2, t)$.

explicitly describes the free-surface position. There are several problems associated with the surface tracking methods. The main problem is that the marker points will be non-uniformly distributed as the interface evolves. Also, relatively high computer storage is needed to maintain the interface continuous and smooth.

Volume tracking techniques define a tracer to follow the whole fluid region. The two commonly used techniques are MAC and VOF techniques. In Marker And Cell (MAC) technique [26, 27] hundreds of massless marker particles are added to the fluid. These particles are then advected in Lagrangian sense using the average of Eulerian velocities in their vicinity. In volume of fluid (VOF) technique [28–31] a volume fraction parameter, F is described for every one of the Eulerian grid cells. A cell is assumed to be filled with liquid if $F=1$, it is considered empty if $F=0$ and partially full if $0<F<1$. Different methods are developed to advect the volume fraction field and to reconstruct the fluid surface. VOF based techniques can handle the most complex free surface flow problems.

In this paper, pseudo-concentration approach [32] is used to track the advancement of fluid, once velocity field is known *a priori*. It is similar to VOF technique. The front tracking algorithm uses a pseudo-concentration function $F(x, y, t)$ which gives a smooth representation of the free surface as shown in Figure 1.

The transport of liquid front can be represented by the Euler equation as

$$\frac{\mathrm{dF}}{\mathrm{dt}} = \frac{\partial \mathrm{F}}{\partial \mathrm{t}} + \mathrm{u}_1 \frac{\partial \mathrm{F}}{\partial \mathrm{x}_1} + \mathrm{u}_2 \frac{\partial \mathrm{F}}{\partial \mathrm{x}_2} - \nu_{\mathrm{ad}} \left\{ \frac{\partial^2 \mathrm{F}}{\partial \mathrm{x}_1^2} + \frac{\partial^2 \mathrm{F}}{\partial \mathrm{x}_2^2} \right\} = 0 \tag{18}$$

where ν_{ad} is the artificial diffusivity to be selected suitably. Artificial diffusivity is a numerical oscillations smoothing factor. The artificial diffusion term has an important effect to cause partial slip of the liquid–air interface at the wall [31]. It stabilizes numerical oscillation during simulation. It is used to tune the results so that spurious oscillations occurring in numerical results converge fast to a steady value. Suppose if $\nu_{\mathrm{ad}}=1$, 'F' values ideally should be in the range (0,1) for all nodes of the domain. If 'F' values are out of the range, artificial diffusivity value is adjusted by trial and error method so that 'F' values fall in the range (0,1) for front tracking.

3. VERIFICATION

The general purpose solution algorithm was developed for the analysis of underfilling chip encapsulation process using commercially available MATLAB v 7 programming language. To illustrate the robustness of the algorithm proposed in this paper, mould filling process for a Z-shaped cavity is considered. The epoxy moulding compound (EMC) enters the mould cavity with an inlet velocity of 1 m/s and is ventilated at diagonally opposite corner. The inlet velocity value is given at the gate entry and the boundary opposite to the gate, where normally air vents are provided, is assigned with zero pressure assuming no slip condition. The problem is analysed for isothermal flow condition. The properties of fluid used are viscosity, $\eta = 0.05\,\mathrm{Ns/m^2}$ or $0.05\,\mathrm{Nsm^{-2}}$ and density, $\rho = 1000\,\mathrm{kg\,m^{-3}}$.

The geometry of mould cavity and applied boundary conditions are shown in Figure 2.

The structured finite element mesh with linear triangular elements is considered in the analysis. The domain is discretized into 1408 elements and 769 nodes. The thickness of the cavity is 2 mm. Thickness of the cavity is not taken into account in the present 2D algorithm as the filling in thickness direction represents a 3D mould filling process. The finite element mesh is shown in Figure 3.

The velocity vector plot from the analysis is shown in Figure 4.

The velocity field clearly indicates that axial velocities only exist at the entry and opposite end zones. When the flow progressively moves forward and sideward, all velocity vectors at middle zone are heading towards the side with almost same magnitudes in both x_1 and x_2 velocity components. It is observed that the velocity vectors close to the wall region act parallel to the wall direction. This velocity field is used in pseudo-concentration approach to track the front of the fluid. The front profile results obtained from front tracking algorithm are compared with those results of Gethin and Abdullah [33]. Comparison of front profiles at different time steps (instants) is shown in Figure 5.

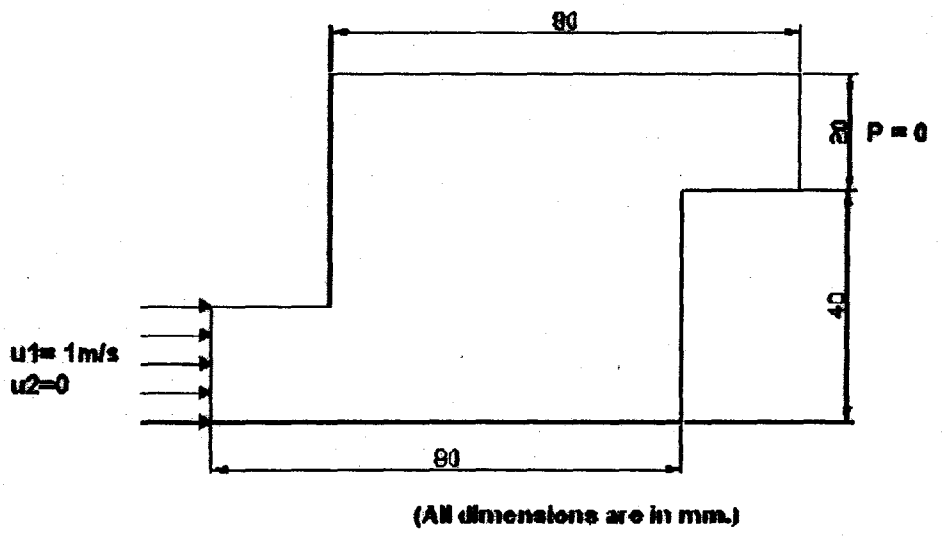


Figure 2. Mould cavity dimensions with boundary conditions.

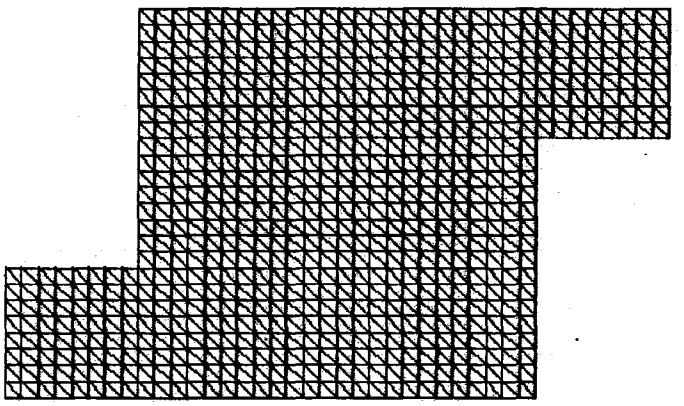


Figure 3. Finite element structured mesh with triangular elements.

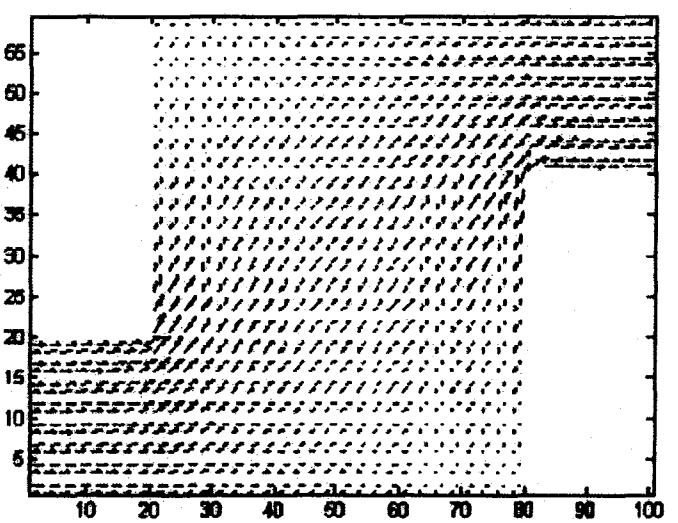


Figure 4. Velocity vector plot.

It can be observed that at 0.02 s, the flow is channelled uniformly into the cavity in the vicinity of entry zone. At 0.05 s, the free surface started to move sideward and wetted the left sidewall. From 0.05 to 0.10 s, the parabolic front pattern advances to the right and side portion of the cavity. The front contours plotted from the CBS scheme coupled with finite element method confirm reasonably well with the results of Gethin and Abdullah [33].

The simulated filling time is 0.23 s. It is in good agreement with the literature value of 0.22 s as predicted by Gethin and Abdullah [33]. The isothermal flow was analysed.

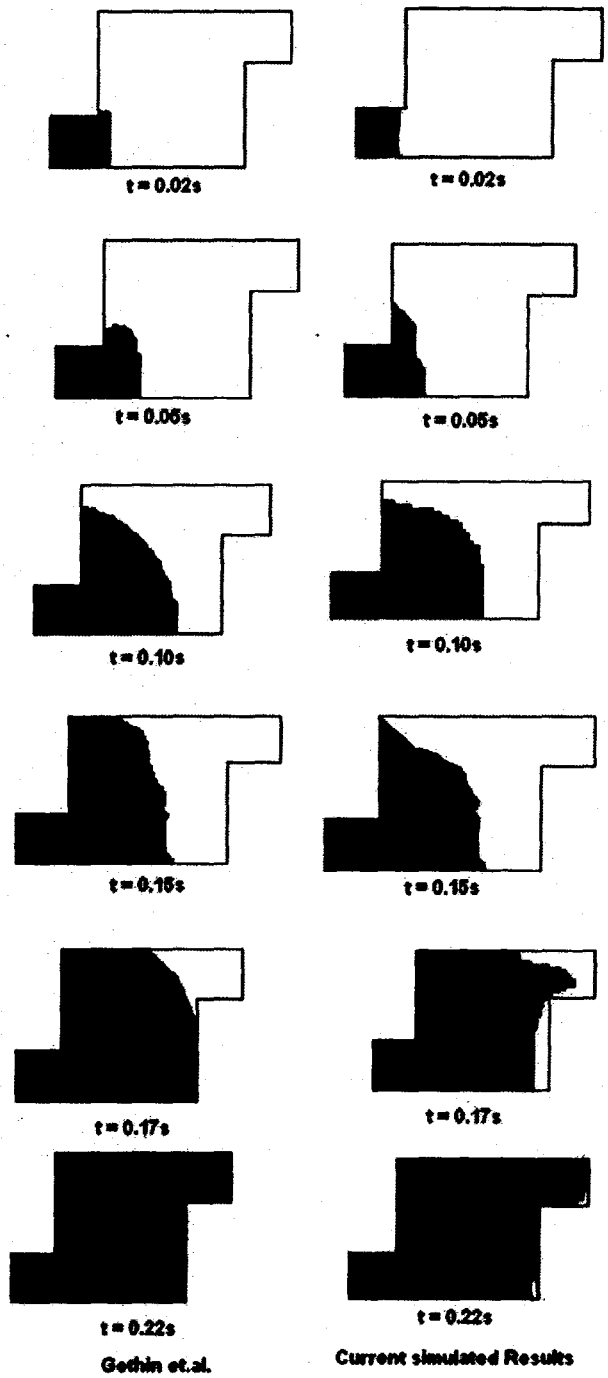


Figure 5. Various front profiles compared with that of Gethin and Abdullah [33].

4. APPLICATION EXAMPLE

The performance of the characteristic based split method used in the paper is demonstrated for a real problem of mould filling in flip-chip package used in electronic industry.

The problem of investigation is shown in Figure 6.

The epoxy material enters the package with a transfer velocity of 1.75 mm/s. The fluid properties considered are specific gravity = 1.8, viscosity, $\eta = 4.5\,N\,s/m^2$. The zero pressure condition is applied at the opposite wall since no slip condition and no shear stresses are applied. The inlet velocity value is given at the gate entry and the boundary opposite to the gate, where normally air vents are provided, is assigned with zero pressure value. Boundary conditions can be summarized as below.

4.1. Boundary conditions

Inlet conditions : $u_1 = 1.75$ mm/s, $u_2 = 0$.
Along sides AB, BD and AC : $u_1 = 0, u_2 = 0$ (no slip condition).
Along side CD: Pressure = 0.

The boundary conditions applied are also shown in Figure 7.

The epoxy material takes 27 s to fill completely the flip-chip package. The various fluid front profiles are shown in Figure 8.

The simulated results of application example are confirmed with data obtained from short shot experiments conducted in an industry. Simulated results of front profiles are in good

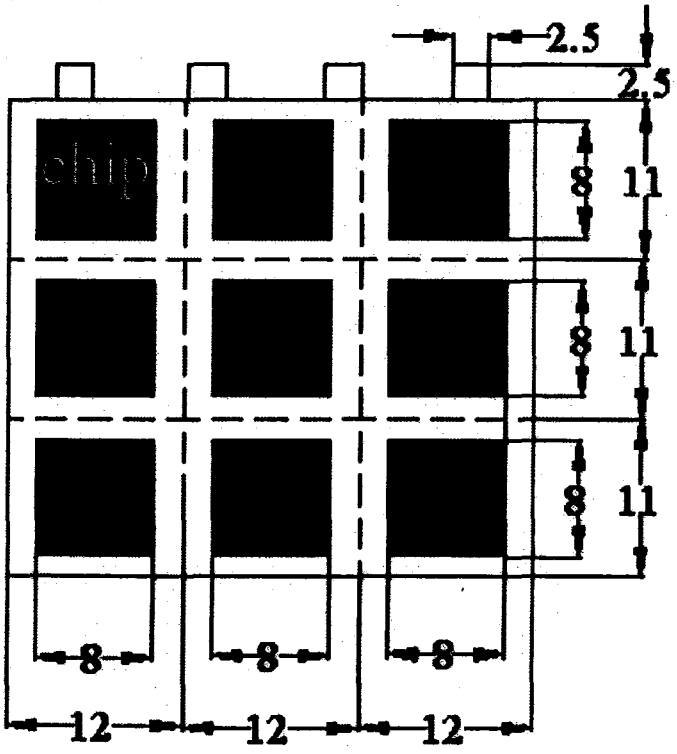


Figure 6. Dimensions of flip chip package.

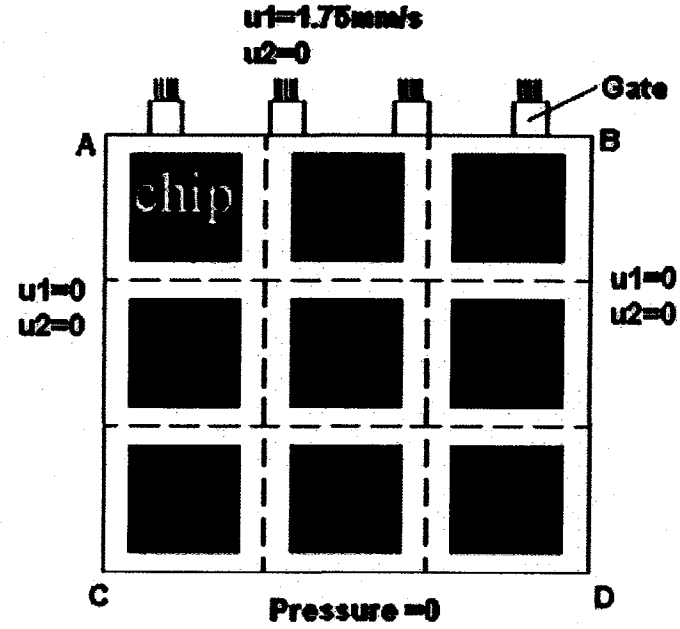


Figure 7. Boundary conditions for the analysis.

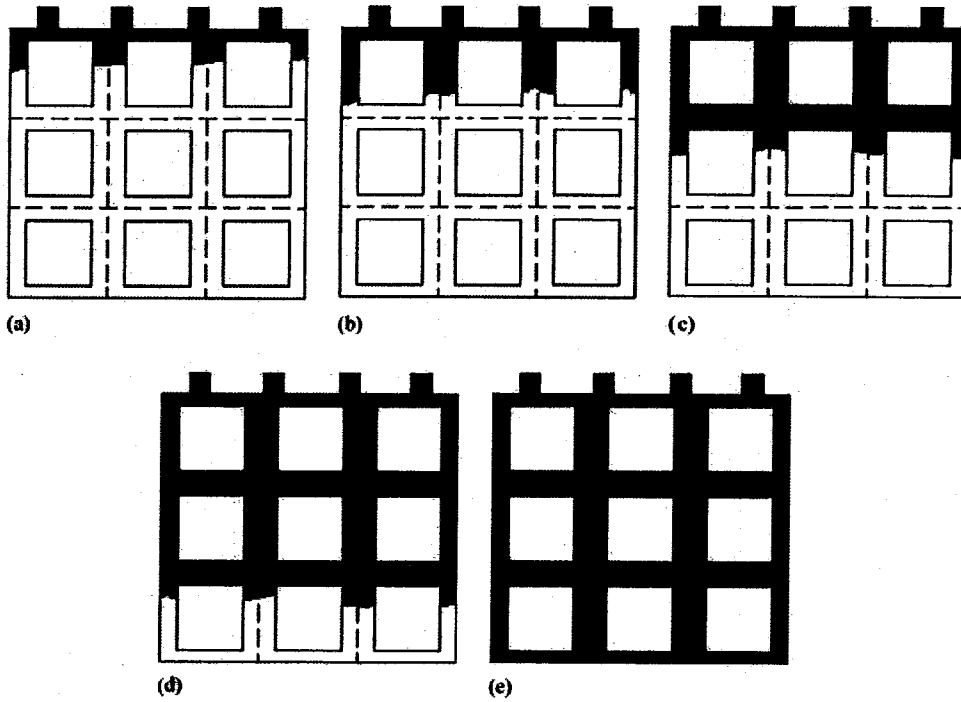


Figure 8. Various front profiles at different time steps: (a) Time step $t=5$ s; (b) $t=10$ s; (c) $t=16$ s; (d) time step $t=22$ s; and (e) filling time $t=27$ s.

agreement with those of numerical and experimental results. It is observed that the simulated filling time matches with the time taken by an epoxy material to fill the package.

5. CONCLUSIONS

The characteristic based split (CBS) scheme has been applied in its semi-implicit form for underfill encapsulation process used in electronic packaging. The robustness of the present algorithm is verified with realistic problem which is encountered in mould filling process. Simulated results of front profiles are in good agreement with those of numerical and experimental results. The agreement is good for low Reynolds numbers, which is the case for underfilling encapsulation process, due to high viscosity of epoxy resin. The implementation of CBS scheme is easy and straightforward. Moreover, it is proved to be an effective alternative for other computational fluid dynamics methods based on finite volume method as it takes less computational time. So, we can conclude that the present algorithm of CBS scheme coupled with pseudo-concentration method can be applied to analyse the mould filling in flip-chip packages.

NOMENCLATURE

$\mathbf{C}$	convection matrix
$\mathbf{f}$	force matrix
$\mathbf{G}$	gradient matrix
$\mathbf{K}$	K-matrix
$\mathbf{K}_s$	stabilization matrix
$\mathbf{K}_t$	heat diffusion matrix
$\mathbf{K}_m$	momentum diffusion matrix
$\mathbf{M}$	mass matrix
p	pressure, N/m^2
$\mathbf{t}$	time, s
T	temperature, K
u	velocity, m/s

Greek symbols

α	thermal diffusivity, m^2/s
ρ	density, kg/m^3
υ	kinematic viscosity, m^2/s

REFERENCES

1. Manzoine LT. *Plastic Packaging of Microelectronic Device.* Van Nostrand Reinhold: New York, 1990; 1–35.
2. Tummala RR. *Fundamentals of Microsystems Packaging.* McGraw-Hill: New York, 2001.
3. Turng LS, Wang VW. On the simulation of microelectronic encapsulation with epoxy molding compound. *Journal of Reinforced Plastic Composites* 1993; **12**:506–519.
4. Nguyen T. Flow simulation in IC chip encapsulation. *Electronic Component and Technology Conference*, Buena Vista, 1994.
5. Tan CW, Aswatha Narayana PA, Goh TJ. Flow and heat transfer during underfilling in flip chip electronic packages. *XVII National and VI ISHMT/ASME Heat and Mass Transfer Conference*, Chennai, India, 2004; 194–199.

6. Daniel P, Lei J, Sadasivan S, Sergei S. A FEM/VOF hybrid formulation for underfill encapsulation modeling. *Computers and Structures* 2003; **81**:879–885.
7. Sejin H, Wang KK. Analysis of the flow of encapsulant during underfill encapsulation of flip-chips. *IEEE Transactions on Components, Packaging and Manufacturing Technology-Part B* 1997; **20**(4):424–433.
8. Han S, Wang KK. Flow analysis in a chip cavity during semiconductor encapsulation. *Journal of Electronic Packaging, Transactions of ASME* 2000; **122**:160–166.
9. Sejin H, Wang KK. Study on the pressurized underfill encapsulation of flip chips. *IEEE Transactions on Components, Packaging and Manufacturing Technology-Part B* 1997; **20**(4):434–442.
10. Kulkarni VM, Seetharamu KN, Aswatha Narayana PA, Azid IA, Quadir GA. Flow analysis for flip chip underfilling process using characteristic based spit method. *6th Electronic Packaging Technology Conference EPTC* 2004; 615–619.
11. Lewis RW, Nithiarasu P, Seetharamu KN. *Fundamentals of the Finite Element Method for Heat and Fluid flow*. Wiley: U.K., 2004.
12. Chorin AJ. Numerical solution of Navier–strokes equations. *Mathematics of Computation* 1968; **22**:745–762.
13. Zienkiwicz OC, Nithiarasu P, Codina R, Vazquez M, Ortitz P. An efficient and accurate algorithm for fluid mechanics problems—the characteristic based split (CBS) algorithm. *International Journal for Numerical Methods in Fluids* 1999; **31**:359–392.
14. Nithiarasu P. An efficient artificial compressibility (AC) scheme based on the characteristic based split (CBS) method for incompressible flows. *International Journal for Numerical Methods in Engineering* 2003; **56**:1615–1845.
15. Nithiarasu P, Massarotti N, Mathur JS. Three dimensional convection heat transfer calculations using the characteristic based split (CBS) scheme. *XVII National and VI ISHMT/ASME Heat and Mass Transfer Conference*, IGCAR, Kalpakkam, India, 2004; 302–307.
16. Zienkiewicz OC, Taylor RL. *The Finite Element Method*, vol. 3 (5th edn). Butterworth and Heinemann: London, 2000.
17. Massarotti N, Nithiarsu P, Zienkiewicz OC. Characteristic based split (CBS) algorithm for incompressible flow problems with heat transfer. *International Journal of Numerical Methods for Heat and Fluid Flow* 1998; **8**:969–990.
18. Gresho P, Sani RL. *Incompressible Flow and the Finite Element Method*. Wiley: New York, 1999.
19. Comini G, Del Guidice S. Finite Element solution of incompressible Navier–Strokes equations. *Numerical Heat Transfer Part A* 1972; **5**:463–473.
20. Ramaswamy B, Jue TC, Akin JE. Semi-implicit and explicit finite element schemes for coupled fluid thermal problems. *International Journal for Numerical methods in Engineering* 1992; **34**:675–696.
21. Lewi RW, Morgan K, Seetharamu KN, Thomas HR. *Finite Element Method in Heat transfer analysis*. Wiley: New York, 1993.
22. Segerlind LJ. *Applied Finite Element Analysis* (2nd edn). Wiley: New York, 1984.
23. Yeung RW. Numerical methods in free surface flows. *Annual Reviews of Fluid Mechanics* 1982; **14**:395.
24. Rallison JM, Acrivos A. A numerical study of the deformation and burst of a viscous drop in an extensional flow. *Journal of Fluid Mechanics* 1978; **89**:191–200.
25. Chern IL, Glimm J, McBryan O, Plohr B, Yaniv S. Front tracking for gas dynamics. *Journal of Computational Physics* 1986; **62**:83–110.
26. Harlow FH, Welch JF. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids* 1965; **8**:2182–2189.
27. Miyata H. Finite-difference simulation of breaking waves. *Journal of Computational Physics* 1986; **65**:179–214.
28. Hirt CW, Nichols BD. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics* 1981; **39**:201–225.
29. Chorin AJ. Flame advection and propagation algorithm. *Journal of Computational Physics* 1980; **35**:1–11.
30. Chorin AJ. Curvature and solidification. *Journal of Computational Physics* 1985; **57**(3):472–490.
31. Ashgriz N, Poo JY. FLAIR: flux line-segment model for advection and interface reconstruction. *Journal of Computational Physics* 1991; **93**(2):449–468.
32. Prasad BS. Solidification simulation of casting using finite element method. *M.S. Thesis*, Indian Institute of Technology, Madras, 1999.
33. Gethin DT, Abdullah S. Finite element formulation for filling a thin section. *International Journal for Numerical methods for heat and fluid flow* 1993; **7**:344–366.

