

SIMULATION OF HEAT TRANSFER PHENOMENON IN FURNACE USING FLUENT-GAMBIT

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF

Bachelor of Technology

in

Mechanical Engineering

By

PEEYUSH AGARWAL

Under the guidance of

Dr. SWARUP KUMAR MAHAPATRA



**Department of Mechanical Engineering
National Institute of Technology
Rourkela
2009**



**National Institute of Technology
Rourkela
CERTIFICATE**

This is to certify that the thesis entitled “**SIMULATION OF HEAT TRANSFER
IN FURNACE USING FLUENT GAMBIT**”

Submitted by **MR.PEEYUSH AGARWAL** in partial fulfillment of the requirements for the award of Bachelor of technology Degree in Mechanical Engineering at the National Institute of Technology, Rourkela (Deemed University) is an authentic work carried out by him under my supervision and guidance. To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other University / Institute for the award of any Degree or Diploma

DATE:

Dr. SWARUP KUMAR MAHAPATRA

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA, 769008



ACKNOWLEDGEMENT

We deem it a privilege to have been the student of Mechanical Engineering stream
in
National Institute of Technology, ROURKELA
Our heartfelt thanks to Dr.S.K.MAHAPATRA, my project guide who helped us to
bring out
this project in good manner with his precious suggestion and rich experience.
We take this opportunity to express our sincere thanks to our project guide for
cooperation
in accomplishing this project a satisfactory conclusion.

Peeyush agarwal

Roll no: 10503026

MECHANICAL ENGINEERING, NIT ROURKELA

CONTENTS

CONTENTS.....	I
ABSTRACT.....	II
1. INTRODUCTION	
1.1 THE MINIATURIZATION	1
1.2 BASICS OF FURNACE	2
1.3 CFD PROCESS	5
1.4 GAMBIT CFD PREPROCESSOR.....	5
2. THEORY	
2.1 HOW DOES A CFD CODE WORK?	6
2.1.1. PRE-PROCESSING:.....	6
2.1.2. SOLVER.....	7
2.1.3 POST-PROCESSING	10
2.1.4. ADVANTAGES OF CFD.....	11
2.2 MATHEMATICAL EQUATIONS.....	11
2.2.1) ENERGY EQUATION	13
2.2.2) MOMENTUM EQUATIONS	17
2.2.3) TURBULENCE EQUATIONS	18
2.2.4) SPECIES EQUATIONS	18
2.2.5) PULL VELOCITY FOR CONTINUOUS CASTING	20
2.3 DISCRETIZATION METHODS IN CFD	22
2.3.1. FINITE DIFFERENCE METHOD (FDM):.....	22
2.3.2. FINITE VOLUME METHOD (FVM):	22
2.3.3. FINITE ELEMENT METHOD (FEM):.....	23
3 . PROBLEM DISCRPTION AND SOLUTION BY GAMBIT –FLUENT	
3.1)GEOMETRY AND MESH.....	25
3.2) ANALYSIS IN FLUENT.....	26
4) RESULTS	
4.1: COMPARISON OF CONVEGENCE CURVES.....	33
4.2: COMPARISON OF TIME TO BECOME STEADY.....	34

4.3: COMPARSION OF TEMPERATURE AT BOTTOM WALL.....	35
4.4: COMPARISON OF THE TEMPERATURE PROFILES.....	36
4.5: COMPARISON OF DENSITY PROFILE	37
4.6: COMPARISON OF SOLIDIFICATION-MELTING CURVE FOR VARYING TEMPERATURE OF FURNACE.....	38
5. CONCLUSION.....	39
BIBLIOGRAPY	

ABSTRACT

In this project work the simulation of heat transfer and the temperature curve in the furnace is computed out using gambit and fluent software. Comparison of temperature profiles of the material in the furnace using constant temperature heat source and linearly varying temperature of the heat source for unsteady state is done. Also the time for temperature to become steady is compared. The material used in the furnace is aluminum and conduction is the mode of heat transfer, the side walls are adiabatic and maintained at ambient temperature. Also the density variation and solidification-melting curve of material filled inside are compared for both conditions. The geometry used is very simple, similar to muffle furnace. The problem will be solved by using the software package FLUENT – GAMBIT.

FLUENT is a computational fluid dynamics (CFD) software package to simulate fluid problems. It uses the finite-volume method to solve the governing equations for a fluid. It provides the capability to use different physical models such as incompressible or compressible, in viscid or viscous, laminar or turbulent, etc. Geometry and grid generation is done using GAMBIT which is the preprocessor bundled with FLUENT.

INTRODUCTION

1.1 THE MINIATURIZATION

HEAT UP processing is the basic step for the workload in melting and heat treatment for further processing. It is also an energy-intensive process. Thus, correct prediction of the temperature variation and distribution in the workload is of significance to ensure the final quality of the parts and to reduce energy consumption and time as well.

There are some studies about the optimization of heat treating process in furnace.

FurnXpert'11 is developed to optimize furnace design and operation for many types of batch and continuous furnaces, such as the continuous belt furnace for sintering process in powder metallurgy. The program mainly focuses on the heat balance of the furnace, while, the load pattern of work pieces is just aligned load pattern in 2-dimension.

Han Xiaoliang [21] established mathematical model for the heating of workpiece in bogie hearth heat treatment furnace, while, the workpiece is assumed to be one dimensional and only single workpiece is considered.

Wan Nini [31] studied the heating up of steel tubes in continuous furnace annealing furnace. The influence of moving speed and thickness of steel tubes are studied. The heat transfer is also assumed to be one dimensional and the load pattern is simply aligned in the width direction. In the heat treatment of parts such as castings and forging, the load pattern is complicated with multi parts stacked in order or disorder and the shapes of work pieces are also irregular. Then it is necessary to model the heat transfer inside the workload.

In recent years, the computational fluid dynamics (CFD) based on conservation equations has become a viable technique for process simulation. So in this work we are going to compute the temperature profile generated. Using CFD and GAMBIT for a given temperature source, it can help to compute the energy required and to optimize it.

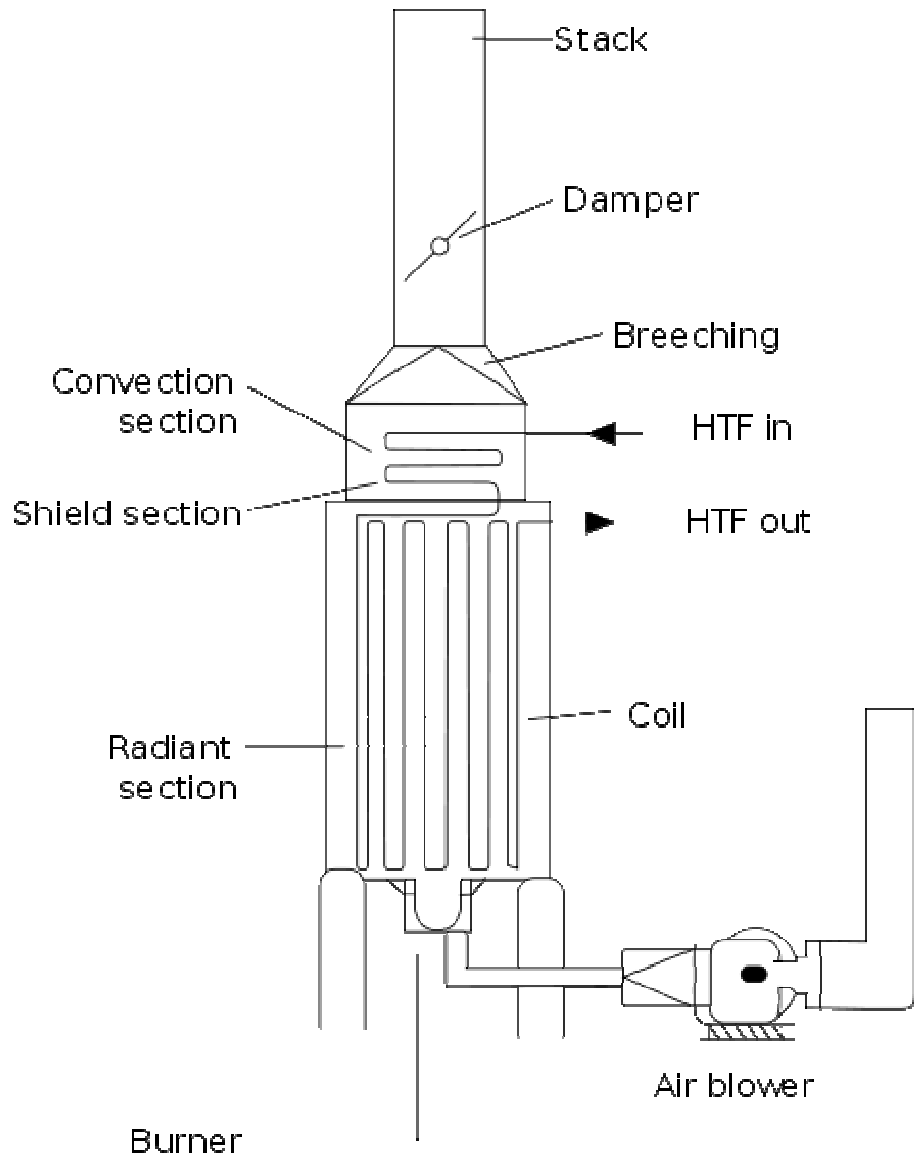
1.2 BASICS OF FURNACE:

A **furnace** is a device used for heating. The heat energy to fuel a furnace may be supplied directly by fuel combustion, by electricity such as the electric arc furnace, or through Induction heating in induction furnace

Industrial process furnaces:-

An industrial furnace or direct fired heater is equipment used to provide heat for a process or can serve as reactor which provides heats of reaction. Furnace designs vary as to its function, heating duty, type of fuel and method of introducing combustion air. However, most process furnaces have some common features.

Fuel flows into the burner and is burnt with air provided from an air blower. There can be more than one burner in a particular furnace which can be arranged in cells which heat a particular set of tubes. Burners can also be floor mounted, wall mounted or roof mounted depending on design. The flames heat up the tubes, which in turn heat the fluid inside in the first part of the furnace known as the radiant section or firebox. In this chamber where combustion takes place, the heat is transferred mainly by radiation to tubes around the fire in the chamber. The heating fluid passes through the tubes and is thus heated to the desired temperature. The gases from the combustion are known as flue gas. After the flue gas leaves the firebox, most furnace designs include a convection section where more heat is recovered before venting to the atmosphere through the flue gas stack.



Radiation section:

Radiation section is where the tubes receive almost all its heat by radiation from the flame. In a vertical, cylindrical furnace, the tubes are vertical. Tubes can be vertical or horizontal, placed along the refractory wall, in the middle, etc., or arranged in cells

Convection section

The convection section is located above the radiant section where it is cooler to recover additional heat. Heat transfer takes place by convection here, and the tubes are finned to increase heat transfer. The first two tube rows in the bottom of the convection section and at the top of the radiant section is an area of bare tubes (without fins) and are known as the shield section, so named because they are still exposed to plenty of radiation from the firebox and they also act to shield the convection section tubes, which are normally of less resistant material from the high temperatures in the firebox

Burner:

The burner in the vertical, cylindrical furnace as above is located in the floor and fires upward. Some furnaces have side fired burners, e.g.: train locomotive. The burner tile is made of high temperature refractory and is where the flame is contained in. Air registers located below the burner and at the outlet of the air blower are devices with movable flaps or vanes that control the shape and pattern of the flame, whether it spreads out or even swirls around

1.3 CFD PROCESS:

Preprocessing is the first step in building and analyzing a flow model. It includes building the model (or importing from a CAD package), applying the mesh, and entering the data. We used Gambit as the preprocessing tool in our project.

There are four general purpose products: FLUENT, Flowizard, FIDAF, and POLY FLOW. FLUENT is used in most industries All Fluent software includes full post processing capabilities.

1.4 GAMBIT CFD PREPROCESSOR:

Fast geometry modeling and high quality meshing are crucial to successful use of CFD.GAMBIT gives us both. Explore the advantage:

Ease of use

CAD/CAE Integration

Fast Modeling

CAD Cleanup

Intelligent Meshing

EASE-OF-USE

GAMBIT has a single interface for geometry creation and meshing that brings together all of Fluent's preprocessing technologies in one environment. Advanced tools for journaling let us edit and conveniently replay model building sessions for parametric studies.

THEORY

2.1 How does a CFD code work?

CFD codes are structured around the numerical algorithms that can be tackle fluid problems. In order to provide easy access to their solving power all commercial CFD packages include sophisticated user interfaces input problem parameters and to examine the results. Hence all codes contain three main elements:

1. Pre-processing.
2. Solver
3. Post-processing.

2.1.1. Pre-Processing:

This is the first step in building and analyzing a flow model. Preprocessor consist of input of a flow problem by means of an operator –friendly interface and subsequent transformation of this input into form of suitable for the use by the solver. The user activities at the Pre-processing stage involve:

- Definition of the geometry of the region: The computational domain.
- Grid generation the subdivision of the domain into a number of smaller, non-overlapping sub domains (or control volumes or elements Selection of physical or chemical phenomena that need to be modeled).
- Definition of fluid properties
- Specification of appropriate boundary conditions at cells, which coincide with or touch the boundary. The solution of a flow problem (velocity, pressure, temperature etc.) is defined at nodes inside each cell. The accuracy of CFD solutions is governed by number of cells in the grid. In general, the larger numbers of cells better the solution accuracy. Both the accuracy of the solution & its cost in terms of necessary computer hardware & calculation time are dependent on the fineness of the grid. Efforts are underway to develop CFD codes with a (self) adaptive meshing capability. Ultimately such programs will automatically refine the grid in areas of rapid variation.

GAMBIT (CFD PREPROCESSOR): GAMBIT is a state-of-the-art preprocessor for engineering analysis. With advanced geometry and meshing tools in a powerful, flexible, tightly-integrated, and easy-to use interface, GAMBIT can dramatically reduce preprocessing times for many applications. Complex models can be built directly within GAMBIT's solid geometry modeler, or imported from any major CAD/CAE system. Using a virtual geometry overlay and advanced cleanup tools, imported geometries are quickly converted into suitable flow domains. A comprehensive set of highly-automated and size function driven meshing tools ensures that the best mesh can be generated, whether structured, multiblock, unstructured, or hybrid.

2.1.2. Solver:

The CFD solver does the flow calculations and produces the results. FLUENT, FloWizard, FIDAP, CFX and POLYFLOW are some of the types of solvers. FLUENT is used in most industries. FloWizard is the first general-purpose rapid flow modeling tool for design and process engineers built by Fluent. POLYFLOW (and FIDAP) are also used in a wide range of fields, with emphasis on the materials processing industries. FLUENT and CFX two solvers were developed independently by ANSYS and have a number of things in common, but they also have some significant differences. Both are control-volume based for high accuracy and rely heavily on a pressure-based solution technique for broad applicability. They differ mainly in the way they integrate the fluid flow equations and in their equation solution strategies. The CFX solver uses finite elements (cell vertex numerics), similar to those used in mechanical analysis, to discretize the domain. In contrast, the FLUENT solver uses finite volumes (cell centered numerics). CFX software focuses on one approach to solve the governing equations of motion (coupled algebraic multigrid), while the FLUENT product offers several solution approaches (density-, segregated- and coupled-pressure-based methods)

The FLUENT CFD code has extensive interactivity, so we can make changes to the analysis at any time during the process. This saves time and enables to refine designs more efficiently. Graphical user interface (GUI) is intuitive, which helps to shorten the learning curve and make the modeling process faster. In addition, FLUENT's adaptive and dynamic mesh capability is

unique and works with a wide range of physical models. This capability makes it possible and simple to model complex moving objects in relation to flow. This solver provides the broadest range of rigorous physical models that have been validated against industrial scale applications, so we can accurately simulate real-world conditions, including multiphase flows, reacting flows, rotating equipment, moving and deforming objects, turbulence, radiation, acoustics and dynamic meshing. The FLUENT solver has repeatedly proven to be fast and reliable for a wide range of CFD applications. The speed to solution is faster because suite of software enables us to stay within one interface from geometry building through the solution process, to post-processing and final output.

The numerical solution of Navier–Stokes equations in CFD codes usually implies a discretization method: it means that derivatives in partial differential equations are approximated by algebraic expressions which can be alternatively obtained by means of the finite-difference or the finite-element method. Otherwise, in a way that is completely different from the previous one, the discretization equations can be derived from the integral form of the conservation equations: this approach, known as the finite volume method, is implemented in FLUENT ([FLUENT user's guide, vols. 1–5, Lebanon, 2001](#)), because of its adaptability to a wide variety of grid structures. The result is a set of algebraic equations through which mass, momentum, and energy transport are predicted at discrete points in the domain. In the freeboard model that is being described, the segregated solver has been chosen so the governing equations are solved sequentially. Because the governing equations are non-linear and coupled, several iterations of the solution loop must be performed before a converged solution is obtained and each of the iteration is carried out as follows:

(1) Fluid properties are updated in relation to the current solution; if the calculation is at the first iteration, the fluid properties are updated consistent with the initialized solution.

(2) The three momentum equations are solved consecutively using the current value for pressure so as to update the velocity field.

(3) Since the velocities obtained in the previous step may not satisfy the continuity equation, one more equation for the pressure correction is derived from the continuity

equation and the linearized momentum equations: once solved, it gives the correct pressure so that continuity is satisfied. The pressure–velocity coupling is made by the SIMPLE algorithm, as in FLUENT default options.

(4) Other equations for scalar quantities such as turbulence, chemical species and radiation are solved using the previously updated value of the other variables; when inter-phase coupling is to be considered, the source terms in the appropriate continuous phase equations have to be updated with a discrete phase trajectory calculation.

(5) Finally, the convergence of the equations set is checked and all the procedure is repeated until convergence criteria are met. (Ravelli et al., 2008)

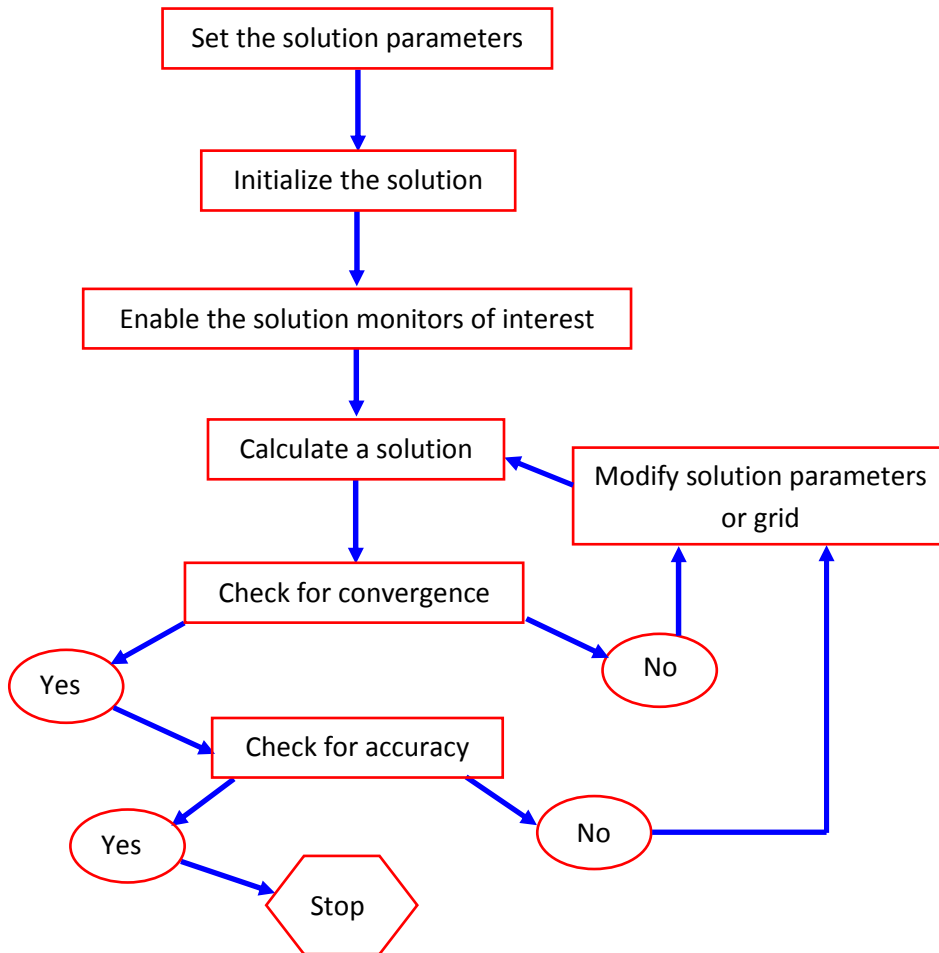


Fig.4. Algorithm of numerical approach used by simulation softwares

The conservation equations are linearized according to the implicit scheme with respect to the dependent variable: the result is a system of linear equations (with one equation for each cell in the domain) that can be solved simultaneously. Briefly, the segregated implicit method calculates every single variable field considering all the cells at the same time. The code stores discrete values of each scalar quantity at the cell centre; the face values must be interpolated from the cell centre values. For all the scalar quantities, the interpolation is carried out by the second order upwind scheme with the purpose of achieving high order accuracy. The only exception is represented by pressure interpolation, for which the standard method has been chosen. [Ravelli et al., 2008](#))

2.1.3 Post-Processing:

This is the final step in CFD analysis, and it involves the organization and interpretation of the predicted flow data and the production of CFD images and animations. Fluent's software includes full post processing capabilities. FLUENT exports CFD's data to third-party post-processors and visualization tools such as Enight, Fieldview and TechPlot as well as to VRML formats. In addition, FLUENT CFD solutions are easily coupled with structural codes such as ABAQUS, MSC and ANSYS, as well as to other engineering process simulation tools.

Thus FLUENT is general-purpose computational fluid dynamics (CFD) software ideally suited for incompressible and mildly compressible flows. Utilizing a pressure-based segregated finite-volume method solver, FLUENT contains physical models for a wide range of applications including turbulent flows, heat transfer, reacting flows, chemical mixing, combustion, and multiphase flows. FLUENT provides physical models on unstructured meshes, bringing you the benefits of easier problem setup and greater accuracy using solution-adaptation of the mesh. FLUENT is a computational fluid dynamics (CFD) software package to simulate fluid flow problems. It uses the finite-volume method to solve the governing equations for a fluid. It provides the capability to use different physical models such as incompressible or compressible, inviscid or viscous, laminar or turbulent, etc. Geometry and grid generation is done using GAMBIT which is the preprocessor bundled with FLUENT. Owing to increased popularity of

engineering work stations, many of which has outstanding graphics capabilities, the leading CFD are now equipped with versatile data visualization tools. These include

- Domain geometry & Grid display.
- Vector plots.
- Line & shaded contour plots.
- 2D & 3D surface plots.
- Particle tracking.
- View manipulation (translation, rotation, scaling etc.)

2.1.4. Advantages of CFD:

Major advancements in the area of gas-solid multiphase flow modeling offer substantial process improvements that have the potential to significantly improve process plant operations. Prediction of gas solid flow fields, in processes such as pneumatic transport lines, risers, fluidized bed reactors, hoppers and precipitators are crucial to the operation of most process plants. Up to now, the inability to accurately model these interactions has limited the role that simulation could play in improving operations. In recent years, computational fluid dynamics (CFD) software developers have focused on this area to develop new modeling methods that can simulate gas-liquid-solid flows to a much higher level of reliability. As a result, process industry engineers are beginning to utilize these methods to make major improvements by evaluating alternatives that would be, if not impossible, too expensive or time-consuming to trial on the plant floor. Over the past few decades, CFD has been used to improve process design by allowing engineers to simulate the performance of alternative configurations, eliminating guesswork that would normally be used to establish equipment geometry and process conditions. The use of CFD enables engineers to obtain solutions for

problems with complex geometry and boundary conditions. A CFD analysis yields values for pressure, fluid velocity, temperature, and species or phase concentration on a computational grid throughout the solution domain. Advantages of CFD can be summarized as:

1. It provides the flexibility to change design parameters without the expense of hardware changes. It therefore costs less than laboratory or field experiments, allowing engineers to try more alternative designs than would be feasible otherwise.

2. It has a faster turnaround time than experiments.

3. It guides the engineer to the root of problems, and is therefore well suited for troubleshooting.

4. It provides comprehensive information about a flow field, especially in regions where measurements are either difficult or impossible to obtain.

2.2) MATHEMATICAL EQUATIONS

2.2.1) Energy Equation

The enthalpy of the material is computed as the sum of the sensible enthalpy, h , and the latent heat, ΔH :

$$H = h + \Delta H \quad (1)$$

$$h = h_{\text{ref}} + \int_{T_{\text{ref}}}^T c_p dT \quad (2)$$

and h_{ref} = reference enthalpy

T_{ref} = reference temperature

c_p = specific heat at constant pressure

The liquid fraction, β , can be defined as

$$\beta = 0 \text{ if } T < T_{\text{solidus}}$$

$$\beta = 1 \text{ if } T > T_{\text{liquidus}}$$

$$\beta = \frac{T - T_{\text{solidus}}}{T_{\text{liquidus}} - T_{\text{solidus}}} \text{ if } T_{\text{solidus}} < T < T_{\text{liquidus}} \quad (-3)$$

Equation 3 is referred to as the lever rule .

The latent heat content can now be written in terms of the latent heat of the material, L :

$$\Delta H = \beta L \quad (4)$$

The latent heat content can vary between zero (for a solid) and L (for a liquid).

In the case of multi component solidification with species segregation; i.e., solidification or melting with species transport, the solidus and liquidus temperatures are computed instead of specified

$$T_{\text{solidus}} = T_{\text{melt}} + \sum_{\text{solute } i} K_i m_i Y_i \quad (5)$$

$$T_{\text{liquidus}} = T_{\text{melt}} + \sum_{\text{solute } i} m_i Y_i \quad (6)$$

where k_i is the partition coefficient of solute i , which is the ratio of the concentration in solid to that in liquid at the interface, y_i is the mass fraction of solute i , and m_i is the slope of the liquidus

surface with respect to y_i . It is assumed that the last species material of the mixture is the solvent and that the other species are the solutes.

For solidification/melting problems, the energy equation is written as

$$\frac{\partial}{\partial t}(\rho H) + \nabla \cdot (\rho \vec{v} H) = \nabla \cdot (k \nabla T) + S \quad (7)$$

where H = enthalpy (see Equation [1](#))

ρ = density

\vec{v} = fluid velocity

S = source term

The solution for temperature is essentially an iteration between the energy equation (Equation [7](#)) and the liquid fraction equation (Equation [3](#)). Directly using Equation [3](#) to update the liquid fraction usually results in poor convergence of the energy equation. In **FLUENT**, the method suggested by Voller and Swaminathan [[386](#)] is used to update the liquid fraction. For pure metals, where T_{solidus} and T_{liquidus} are equal, a method based on specific heat, given by Voller and Prakash [[385](#)], is used instead.

2 .2.2) Momentum Equations

The enthalpy-porosity technique treats the mushy region (partially solidified region) as a porous medium. The porosity in each cell is set equal to the liquid fraction in that cell. In fully solidified regions, the porosity is equal to zero, which extinguishes the velocities in these regions. The momentum sink due to the reduced porosity in the mushy zone takes the following form:

$$S = \frac{(1 - \beta)^2}{(\beta^3 + \epsilon)} A_{\text{mush}} (\vec{v} - \vec{v}_p) \quad (8)$$

where β is the liquid volume fraction, ϵ is a small number (0.001) to prevent division by zero, A_{mush} is the mushy zone constant, and \vec{v}_p is the solid velocity due to the pulling of solidified material out of the domain (also referred to as the pull velocity).

The mushy zone constant measures the amplitude of the damping; the higher this value, the steeper the transition of the velocity of the material to zero as it solidifies. Very large values may cause the solution to oscillate.

The pull velocity is included to account for the movement of the solidified material as it is continuously withdrawn from the domain in continuous casting processes. The presence of this term in Equation [8](#) allows newly solidified material to move at the pull velocity. If solidified

material is not being pulled from the domain, $\vec{v}_p = 0$. More details about the pull velocity are provided in Section [5](#).

2.2.3) Turbulence Equations

Sinks are added to all of the turbulence equations in the mushy and solidified zones to account for the presence of solid matter. The sink term is very similar to the momentum sink term (Equation [24.2-8](#)):

$$S = \frac{(1 - \beta)^2}{(\beta^3 + \epsilon)} A_{\text{mush}} \phi \quad (9)$$

where ϕ represents the turbulence quantity being solved (k , ϵ , ω , etc.), and the mushy zone constant, A_{mush} , is the same as the one used in Equation [8](#).

2.2.4) Species Equations

In the case of solidification/melting with species transport, the following species equation is solved:

$$\frac{\partial}{\partial t} (\rho Y_i) + \nabla \cdot (\rho [\beta \vec{v}_{\text{liq}} Y_{i,\text{liq}} + (1 - \beta) \vec{v}_p Y_{i,\text{sol}}]) = -\nabla \cdot \vec{J}_i + R_i \quad (10)$$

where \vec{J}_i is given by

$$\vec{J}_i = -\rho[\beta D_{i,m,liq} \nabla Y_{i,liq} + (1 - \beta) D_{i,m,sol} \nabla Y_{i,sol}] \quad (11)$$

Here Y_i is the average species mass fraction in a cell:

$$Y_i = \beta Y_{i,liq} + (1 - \beta) Y_{i,sol} \quad (12)$$

$Y_{i,liq}$ and $Y_{i,sol}$ are related by the partition coefficient K_i :

$$Y_{i,sol} = K_i Y_{i,liq} \quad (13)$$

\vec{v}_{liq} is the velocity of the liquid and \vec{v}_p is the solid (pull) velocity. \vec{v}_p is set to zero if pull velocities are not included in the solution. The liquid velocity can be found from the average velocity (as determined by the flow equation) as

$$\vec{v}_{liq} = \frac{(\vec{v} - \vec{v}_p(1 - \beta))}{\beta} \quad (14)$$

2.2.5) Pull Velocity for Continuous Casting

In continuous casting processes, the solidified matter is usually continuously pulled out from the computational domain, as shown in Figure [24.2.1](#). Consequently, the solid material will have a finite velocity that needs to be accounted for in the enthalpy-porosity technique.

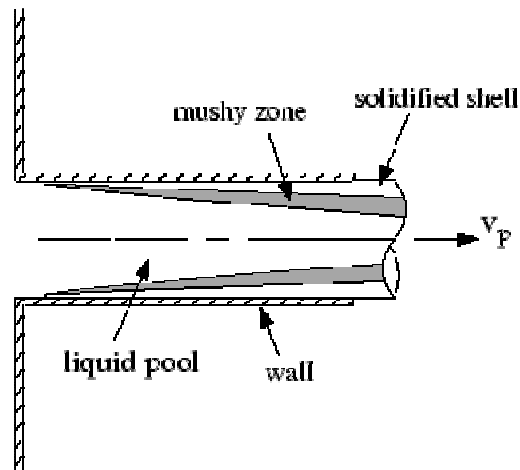


Figure 24.2.1: "Pulling" a Solid in Continuous Casting

As mentioned in Section [2](#), the enthalpy-porosity approach treats the solid-liquid mushy zone as a porous medium with porosity equal to the liquid fraction. A suitable sink term is added in the momentum equation to account for the pressure drop due to the porous structure of the mushy zone. For continuous casting applications, the relative velocity between the molten liquid and the solid is used in the momentum sink term (Equation [8](#)) rather than the absolute velocity of the liquid.

The exact computation of the pull velocity for the solid material is dependent on the Young's modulus and Poisson's ratio of the solid and the forces acting on it. **FLUENT** uses a Laplacian equation to approximate the pull velocities in the solid region based on the velocities at the boundaries of the solidified region:

$$\nabla^2 \vec{v}_p = 0 \quad (15)$$

FLUENT uses the following boundary conditions when computing the pull velocities:

- At a velocity inlet, a stationary wall, or a moving wall, the specified velocity is used.
- At all other boundaries (including the liquid-solid interface between the liquid and solidified material), a zero-gradient velocity is used.

The pull velocities are computed only in the solid region.

Note that **FLUENT** can also use a specified constant value or custom field function for the pull velocity, instead of computing it. See Section [2](#) for details.

(2.3) Discretization Methods in CFD

There are three discretization methods in CFD:

1. Finite difference method (FDM)
2. Finite volume method (FVM)
3. Finite element method (FEM)

2.3.1. Finite difference method (FDM): A finite difference method (FDM) discretization is based upon the differential form of the PDE to be solved. Each derivative is replaced with an approximate difference formula (that can generally be derived from a Taylor series expansion). The computational domain is usually divided into hexahedral cells (the grid), and the solution will be obtained at each nodal point. The FDM is easiest to understand when the physical grid is Cartesian, but through the use of curvilinear transforms the method can be extended to domains that are not easily represented by brick-shaped elements. The discretization results in a system of equation of the variable at nodal points, and once a solution is found, then we have a discrete representation of the solution.

2.3.2. Finite volume method (FVM): A finite volume method (FVM) discretization is based upon an integral form of the PDE to be solved (e.g. conservation of mass, momentum, or energy). The PDE is written in a form which can be solved for a given finite volume (or cell). The computational domain is discretized into finite volumes and then for every volume the governing equations are solved. The resulting system of equations usually involves fluxes of the conserved variable, and thus the calculation of fluxes is very important in FVM. The basic advantage of this method over FDM is it does not require the use of structured grids, and the

effort to convert the given mesh in to structured numerical grid internally is completely avoided. As with FDM, the resulting approximate solution is a discrete, but the variables are typically placed at cell centers rather than at nodal points. This is not always true, as there are also face-centered finite volume methods. In any case, the values of field variables at non-storage locations (e.g. vertices) are obtained using interpolation.

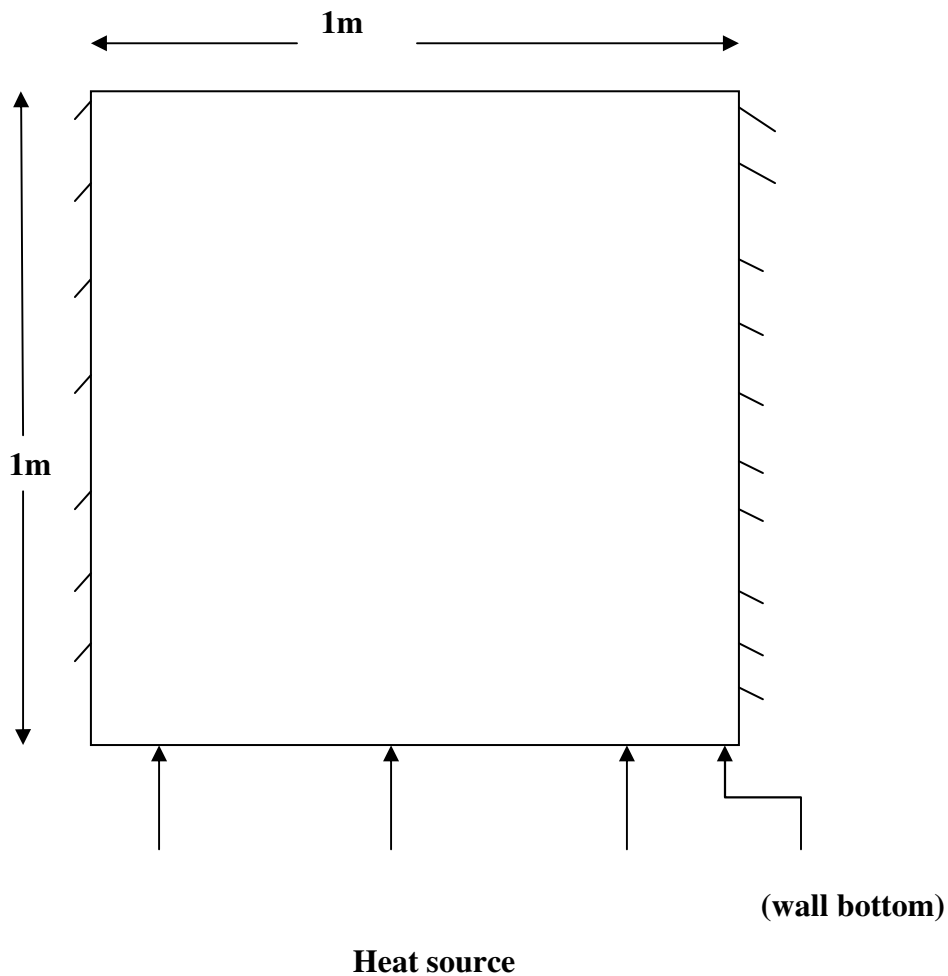
2.3.3. Finite element method (FEM): A finite element method (FEM) discretization is based upon a piecewise representation of the solution in terms of specified basis functions. The computational domain is divided up into smaller domains (finite elements) and the solution in each element is constructed from the basis functions. The actual equations that are solved are typically obtained by restating the conservation equation in weak form: the field variables are written in terms of the basis functions, the equation is multiplied by appropriate test functions, and then integrated over an element. Since the FEM solution is in terms of specific basis functions, a great deal more is known about the solution than for either FDM or FVM. This can be a double-edged sword, as the choice of basis functions is very important and boundary conditions may be more difficult to formulate. Again, a system of equations is obtained (usually for nodal values) that must be solved to obtain a solution.

Comparison of the three methods is difficult, primarily due to the many variations of all three methods. FVM and FDM provide discrete solutions, while FEM provides a continuous (up to a point) solution. FVM and FDM are generally considered easier to program than FEM, but opinions vary on this point. FVM are generally expected to provide better conservation properties, but opinions vary on this point also.

PROBLEM DISCRPTION AND SOLUTION
BY GAMBIT –FLUENT

Problem description

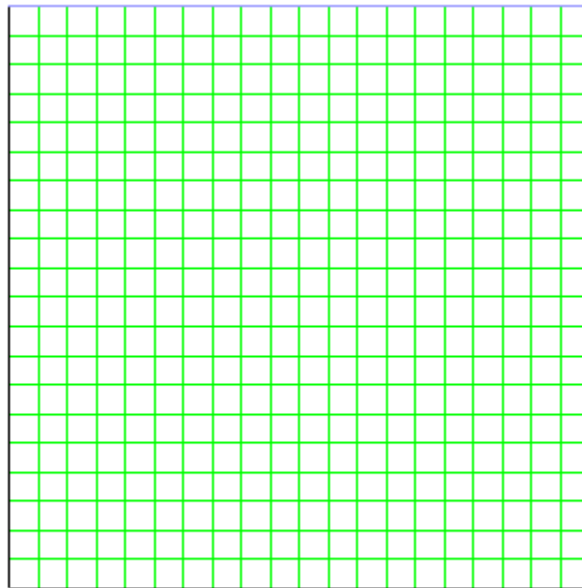
The problem is to simulate the heat transfer and the temperature curve in the furnace using gambit and fluent software. Comparison of temperature profiles of the material in the furnace using constant temperature heat source and linearly varying temperature of the heat source for unsteady state is found out. Also the time for temperature to become steady is compared. The material used in the furnace is aluminum and conduction is the mode of heat transfer, the side walls are adiabatic and maintained at ambient temperature. Also the density variation and solidification-melting curve of material filled inside are compared for both conditions



THE DEFAULT MATERIAL USED INSIDE IS ALUMINIUM

Simulation:

3.1) GEOMETRY AND MESH



Grid (Time=4.9000e+02)

May 08, 2009
FLUENT 6.2 (2d, segregated, lam, unsteady)

GAMBIT 2.2.30 was used for making 2D rectangular geometry with width of 1m and height 1 m, t geometry was later meshed into 20x20 cells.

3.2)ANALYSIS IN FLUENT

PROCEDURE: (2D VERSION OF FLUENT)

STEP 1: (GRID)

FILE → READ → CASE

The file channel mesh is selected by clicking on it under files and

Then ok is clicked.

The grid is checked.

GRID → CHECK

The grid was scaled to 1 in all x, y and z directions.

GRID → SCALE

The grid was displayed.

DISPLAY → GRID

Grid is copied in ms-word file.

STEP 2 :(Models)

The solver was specified.

DEFINE → MODELS → SOLVER

Solver is segregated

Implicit formulation

Space steady

Time steady

DEFINE → MODEL → ENERGY

Energy equation is clicked on.

DEFINE → MODELS → SOLIDIFICATION AND MELTING

STEP 3:(Materials)

the material selected was ALUMINIUM with properties.

Dynamic Viscosity, $\mu = 1.7894 \times 10^{-5}$

Density, $= 2700 - .1 * T$

Thermal Conductivity, $K = 237 \text{ W/mK}$

Specific heat, $C_p = 897 \text{ J/kg- K}$

Melting heat = 400000 j/kg

Solidus temperature = 933.47 k

Liquidus temperature = 933.47 k

STEP 4(Operating conditions)

Operating pressure = 101.325 KPa

Gravity = -9.81 m/s^2 in Y-direction

STEP 5:(Boundary conditions)

DEFINE → BOUNDARY CONDITIONS

ALUMINIUM

Set option is clicked

Default Interior

Set option is clicked

Velocity inlet.

Y -component of velocity= .001ml s.

Temperature =300 k

Wall out

Thermal conditions are temperature

Temperature =1400 k

Wall thickness= .2m

Heat generation=0

SIDE WALLS

Thermal conditions are temperature

Temperature =300k

Wall thickness= .2m

Heat generation=0

STEP 6: (custom field function)

Define temperature with x coordinates

Field function= grid

Define function to be , $T = (x + .5) * 1400 + 700$

STEP 6: (Solution)

SOLVE → CONTROLS → SOLUTIONS

All energy equation used.

Under relaxation factors

Pressure= 0.3

Density= 1

Body Force= 1

Momentum= .7

SOLVE → INITIALIZE

Compute from –all zones

Temperature=300 k

Click **INIT**

STEP 7: (adapt)

ADAPT → BOUNDARY ADAPTION

Boundary zone= wall out

CLICK → MARK → APPLY

STEP 8: (PATCH)

SOLVE → INITIALIZE → PATCH

Patch temperature with boundary wall out

STEP 9: (ITERATE)

SOLVE → ITERATE

Input 100 as the number of iterations and iterate was clicked.

For patch function and without patch function, Convergence was checked.

FILE → WRITE → DATA

STEP 10:(Displaying the preliminary solution)

Display of filled contours of velocity magnitude

DISPLAY → **CONTOURS**

Display of filled contours of temperature

DISPLAY → **CONTOURS**

~Temperature was selected and then

1. Static temperature.
2. Total temperature from drop down list was selected

~Display was clicked.

Display of filled contours of temperature

DISPLAY → **CONTOURS**

~density was selected and then

~Display was clicked.

Display of filled contours density

DISPLAY → **CONTOURS**

~solidification-melting was selected and then

1. liquid fraction.

~Display was clicked.

PLOTXY → **PLOT**

Y AXIS-TEMPERATURE

X AXIS-GRID

- 1.X AXIS= X CORDINATE
2. X AXIS= Y CORDINATE

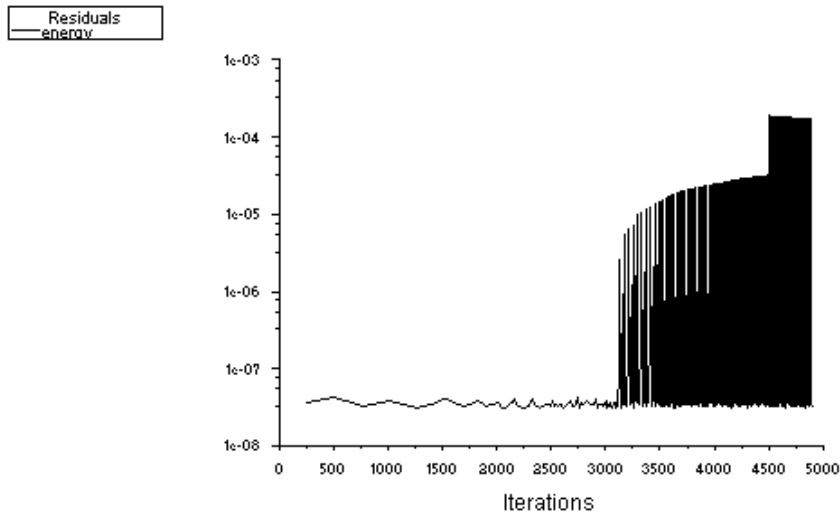
Find the temperature on a line varying x coordinates and y coordinates

RESULTS

The study work included the comparison of various graphs for the two conditions; the various graphs are plotted in the following pattern.

SLNO	FIG NO	DISCRIPTION
1	1,2	COMPARISON CONVEGENCE CURVES
2	3,4	COMAPARISON OF TIME TO BECOME STEADY
3	5,6	COMPARSION OF TEMPERATURE AT BOTTOM WALL
4	7,8	COMPARISON OF THE TEMPERATURE PROFILES
5	9,10	COMPARISON OF DENSITY PROFILE
6	11	COMPARISON OF SOLIDIFICATION-MELTING CURVE FOR VARYING TEMPERATURE OF FURNACE

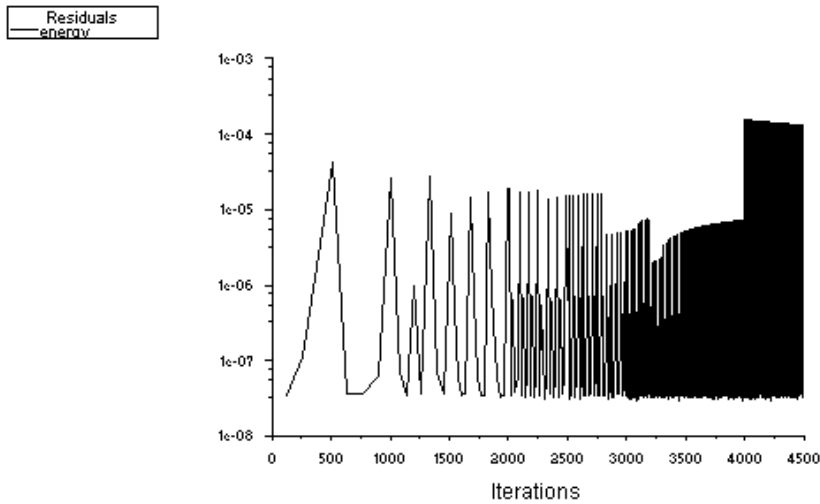
4.1: COMPARISON OF CONVEGENCE CURVES



Scaled Residuals (Time=4.9000e+02)

May 08, 2009
 FLUENT 6.2 (2d, segregated, lam, unsteady)

FIG1: THE GRAPH WAS CONVERGED WITH A CONSTANT TEMPERATURE OF BOTTOM WALL =1400 K



Scaled Residuals (Time=4.5000e+02)

May 08, 2009
 FLUENT 6.2 (2d, segregated, lam, unsteady)

FIG2: THE GRAPH WAS CONVERGED WITH LINEARLY VARYING TEMPERATURE OF BOTTOM WALL, $T = (x + .5) * 1400 + 700$, X IS THE CORDINATE IN X DIRECTION

4.2: COMPARISON OF TIME TO BECOME STEADY

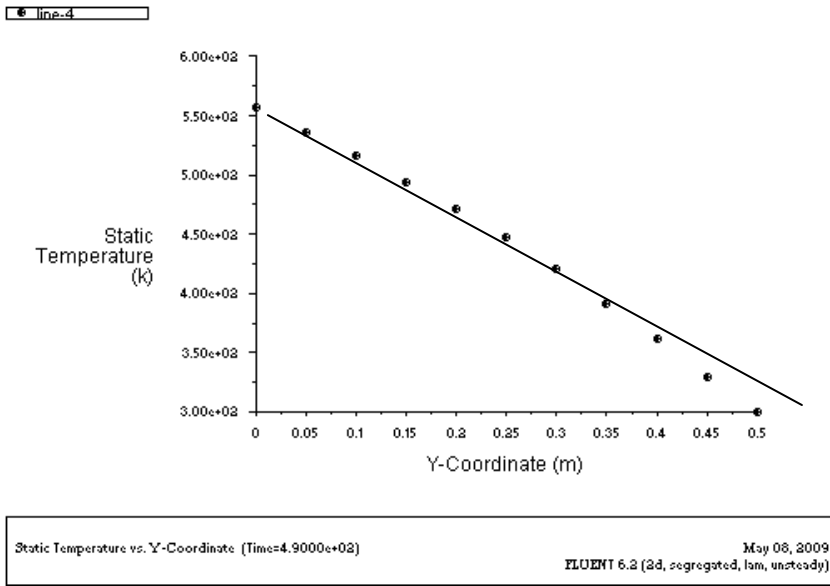


FIG 3: THE TIME FOR TEMPERATURE TO BECOME STEADY WITH A CONSTANT TEMPERATURE OF BOTTOM WALL =1400 K WITH Y COORDINATE, T=490 ITERATION WITH TIME STEP =1 SECOND

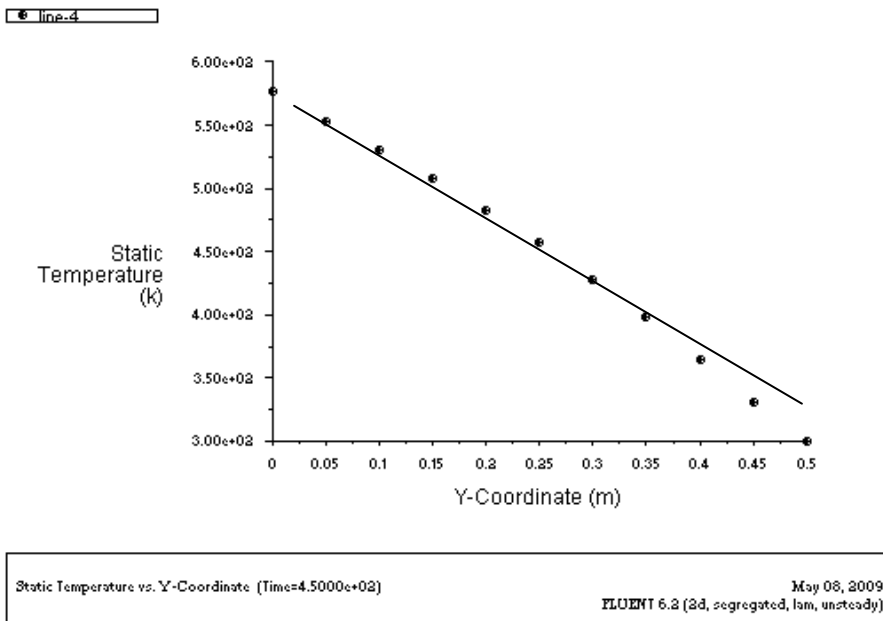


FIG 4: THE TIME FOR TEMPERATURE TO BECOME STEADY WITH A TEMPERATURE OF BOTTOM WALL, $T = (x + .5) * 1400 + 700$, FOR X=0, WITH Y COORDINATE, T=450 ITERATION WITH TIME STEP =1 SECOND

4.3: COMPARISON OF TEMPERATURE AT BOTTOM WALL

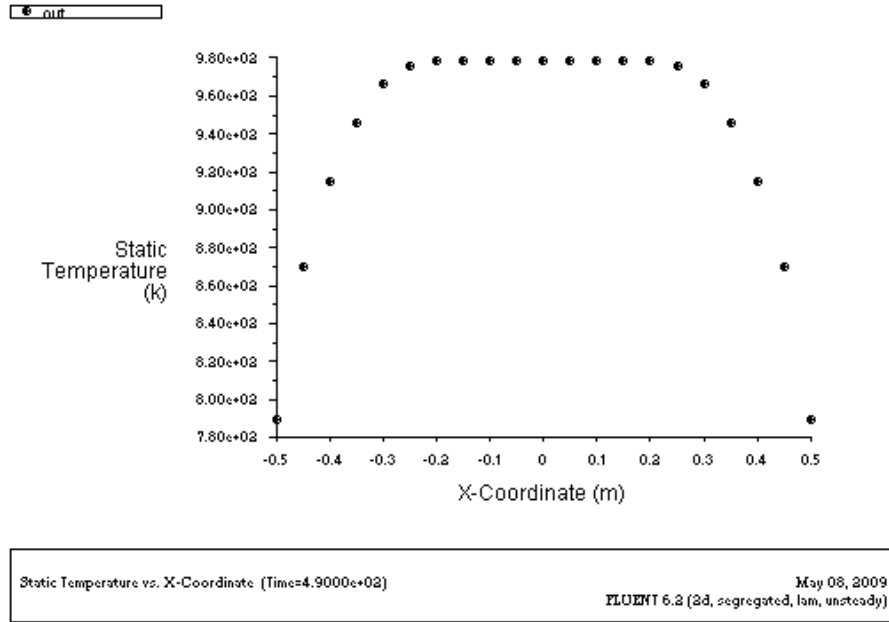


FIG 5: TEMPERATURE OF THE BOTTOM WALL AT STEADY CONDITION WITH WALL TEMPERATURE=1400K

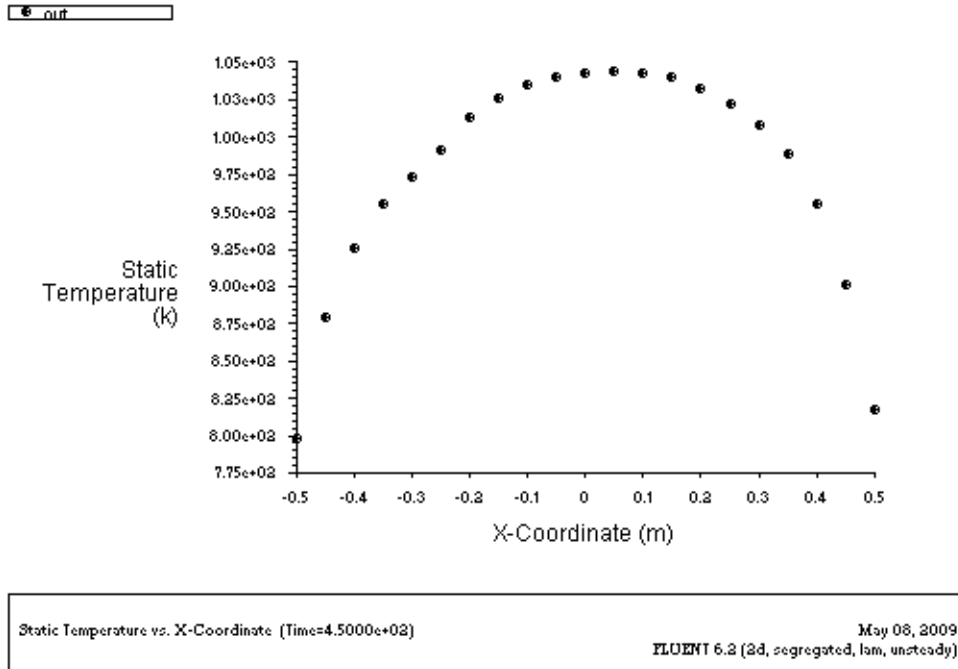


FIG 6 : THE TEMPERATURE OF THE BOTTOM WALL AT STEADY CONDITION WITH THE TEMPERATURE, $T = (x + .5) * 1400 + 700$

4.4: COMPARISON OF THE TEMPERATURE PROFILES

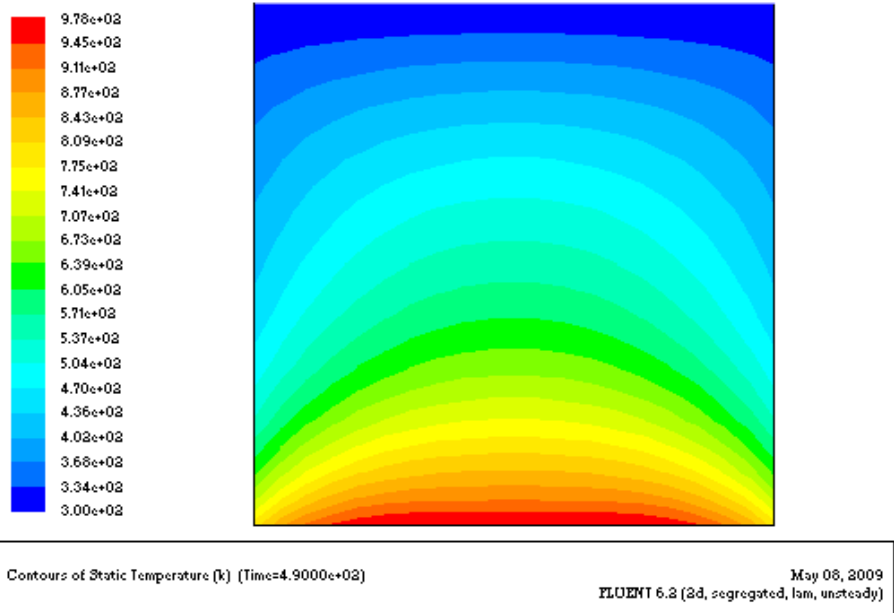


FIG 7: TEMPERATURE PROFILE OF THE FLUID INSIDE FURNACE AT CONSTANT BOTTOM WALL TEMPERATURE AND AT STEADY CONDITION

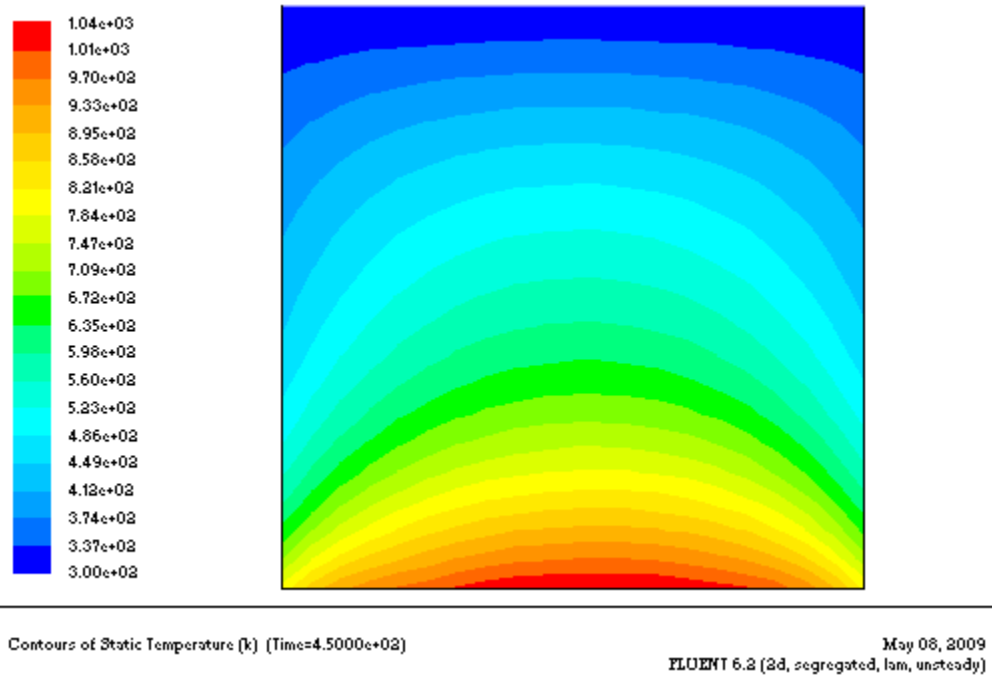


FIG8:TEMPERATURE PROFILE OF THE FLUID INSIDE FURNACE AT BOTTOM WALL TEMPERATURE $T = (x + .5) * 1400 + 700$ AND AT STEADY CONDITION

4.5: COMPARISON OF DENSITY PROFILE

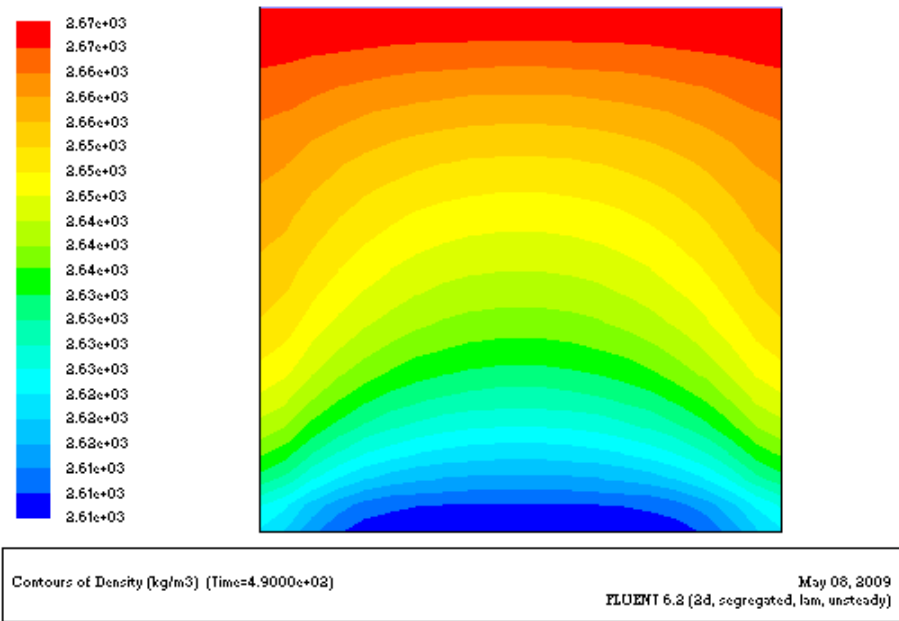
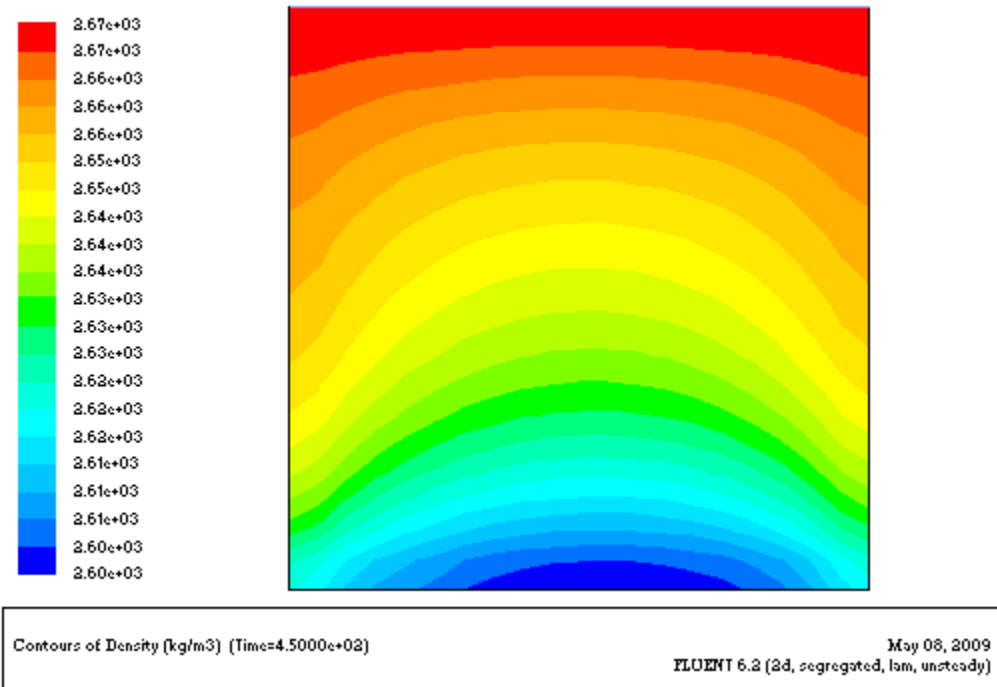
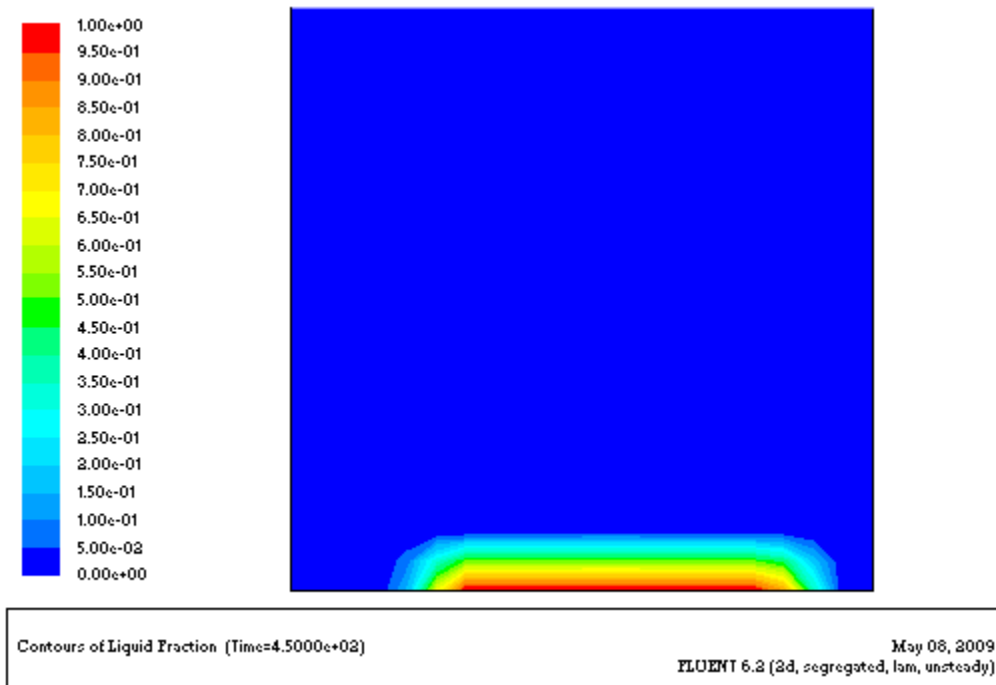


FIG9:THE DENSITY CURVE OF THE FLUID AT CONSTANT TEMPERATURE OF BOTTOM WALL, T =1400K AND AT STEADY CONDITION



THE DENSITY CURVE OF THE FLUID AT TEMPERATURE $T = (x + .5) * 1400 + 700$ OF BOTTOM WALL,

4.6: COMPARISON OF SOLIDIFICATION-MELTING CURVE FOR VARYING TEMPERATURE OF FURNACE



SOLIDIFICATION MELTING CURVE OF THE FLUID AT VARYING TEMPERATURE,

$$T = (x + .5) * 1400 + 700$$

COMPARISON CONVEGENCE CURVES

The graphs got converged properly for constant temperature at bottom wall also at linearly varying temperature of the bottom wall. The graph for constant temperature was converged for 490 steps of time step of 1 second, while the graph for linearly varying temperature was converged for 450 steps of a time step of 1 second.

COMPARISON OF TIME TO BECOME STEADY

The graph for temperature of a line drawn at the centre of the furnace at steady condition was compared, varying 'y' coordinates keeping 'x' coordinate to be fixed. It was found that the curve for constant bottom temperature =1400 k got steady at time of 490 seconds while the temperature got steady for linearly varying temperature at bottom wall at a time of 450 seconds.

COMPARISON OF TEMPERATURE AT BOTTOM WALL

The graph for temperature at the bottom wall for steady condition for a fixed temperature of 1400k was found, and the maximum temperature computed at the center was found to be 980 k while the maximum temperature at the bottom of the wall for linearly varying temperature was found to be 1050 k.

COMPARISON OF THE TEMPERATURE PROFILES

The graph for the temperature of the fluid at constant temperature and linearly varying temperature was found out. The maximum temperature was found at the bottom of the wall and at the middle of it which was 978 k and maximum temperature for linearly varying temperature was found out to be 1040 k

COMPARISON OF DENSITY PROFILE

The density for both the cases was almost similar with lesser density on the right side of the furnace in the linearly increasing temperature case.

COMPARISON OF SOLIDIFICATION-MELTING CURVE FOR VARYING TEMPERATURE OF FURNACE

It was found out that there was no liquefaction in case of constant temperature of bottom wall at temperature of 1400 K while the liquefaction was found out in linearly varying temperature at bottom of wall

RESULTS AND CONCLUSION

The temperature profiles, density, solidification-melting and temperature at the bottom of the wall and the time for the temperature to become steady for a constant temperature and linearly increasing temperature are compared. It was found that:

- The graph for constant temperature was converged for 490 steps of time step of 1 second, while the graph for linearly varying temperature was converged for 450 steps of a time step of 1 second.
- Temperature at the bottom wall for steady condition for a fixed temperature of 1400k was found, and the maximum temperature computed at the center was found to be 980 k while the maximum temperature at the bottom of the wall for linearly varying temperature was found to be 1050 k.
- The density for both the cases was almost similar with lesser density on the right side of the furnace in the linearly increasing temperature case.
- It was found out that there was no liquefaction in case of constant temperature of bottom wall at temperature of 1400 k while the liquefaction was found out in linearly varying temperature at bottom of wall.

BIBLIOGRAPHY

- [www.fluent.com/ software/ gambit.](http://www.fluent.com/software/gambit)
- Numerical Heat Transfer and Fluid Flow by Suhas V. Patankar
- www.wikipedia.org
- Heat transfer by– P.K NAG