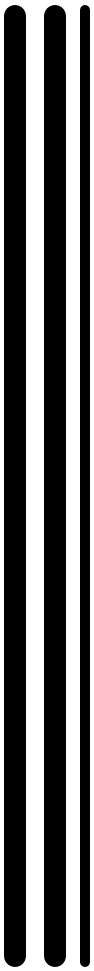
Metals Processing Simulation Laboratory
Department of Mechanical and Industrial Engineering
University of Illinois at Urbana-Champaign
Urbana, IL 68101

MODELING OF FLOW ASYMMETRIES AND PARTICLE ENTRAPMENT IN NOZZLE AND MOLD DURING CONTINUOUS CASTING OF STEEL SLABS



Sana Mahmood

Continuous Casting Consortium

Report

2006

Submitted to

Accumold
Algoma Steel
Baosteel
Corus
Labein
LWB Refractories
Nucor Steel Decatur
Mittal Riverdale Steel
Postech
Steel Dynamics Inc.

June 30, 2006

MODELING OF FLOW ASYMMETRIES AND PARTICLE ENTRAPMENT IN NOZZLE AND MOLD DURING CONTINUOUS CASTING OF STEEL SLABS

BY

SANA MAHMOOD

B.S., Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, 2003

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Mechanical Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2006

Urbana, Illinois

CERTIFICATE OF COMMITTEE APPROVAL

University of Illinois at Urbana-Champaign
Graduate College

June 30, 2006

We hereby recommend that the thesis by:

SANA MAHMOOD

Entitled:

MODELING OF FLOW ASYMMETRIES AND PARTICLE ENTRAPMENT IN NOZZLE AND MOLD DURING CONTINUOUS CASTING OF STEEL SLABS

Be accepted in partial fulfillment of the requirements for the degree of:

Master of Science

Signatures:

Director of Research - Brian G. Thomas

Head of Department - Huseyin Sehitoglu

Committee on Final Examination*

Chairperson -

Committee Member -

Committee Member -

Committee Member -

Committee Member -

Committee Member -

* Required for doctoral degree but not for master's degree

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

NOMENCLATURE

a_0	Liquid atomic radius (m)
A_c	Particle accelerating parameter
$\vec{A}$	Area vector of the boundary face of cell (m^2)
C	Weight concentration (wt pct)
C_A	Correction factor on added mass force
C_D	Coefficient of drag
d_p	Particle diameter (m)
D	Diffusion coefficient (m^2/s)
F_A	Added mass force (N)
F_B	Buoyancy force (N)
F_D	Drag force (N)
F_G	Gravitational force (N)
F_{Grad}	Surface energy gradient force (N)
F_I	Van der Waals interfacial force (N)
F_H	Basset history force (N)
F_L	Lift force (N)
F_{Press}	Pressure gradient force (N)
F_{stress}	Stress gradient force (N)
g	Gravitational acceleration vector (m^2/s)
G	Velocity gradient of a shear flow (1/s)

h_0	Distance between particle and dendrite tip (m)
K	Turbulent kinetic energy (m^2/s^2)
k	Distribution coefficient ($=C_s/C_l$)
m	Empirical constant (J/m^2)
m_p	Particle mass (kg)
M	Mass (kg)
n	Empirical coefficient (1/wt pct)
p	Pressure (Pa)
R_p	Particle radius (m)
Re	Reynolds number
Re_p	Particle reynolds number
Re_s	Reynolds number based on slip velocity
Re_G	Reynolds number based on velocity shear
R^f	Scaled residual error
r_d	Dendrite tip radius (m)
S_m	Source term
T_L	Liquidus temperature (K)
T_s	Solidus temperature (K)
U_{jet}	Average jet speed (m/s)
u_i	Velocity in cell i in x direction (m/s)
$\overline{u_i}$	Mean velocity components (m/s)

u_i'	Fluctuating velocity components (m/s)
u	Velocity component in x direction (m/s)
v_i	Velocity in cell i in y direction (m/s)
V_{sol}	Solidification interface advancing speed (m/s)
V_{avg}	Average Velocity (m/s)
$\vec{V}$	Velocity vector (m/s)
$\hat{V}_c$	unit vector in the direction of casting velocity
V_c	Casting speed (m/s)
v_p	Particle velocity (m/s)
v_f	Fluid velocity (m/s)
v	Velocity component in y direction (m/s)
w	Velocity component in z direction (m/s)
w_i	Velocity in cell i in z direction (m/s)
x	Mold width direction (m)
y	Mold thickness direction (m)
z	Casting direction (m)
α	Constant for surface energy gradient force
β	Constant for surface energy gradient force
ρ	Fluid density (kg/m^3)
ρ_p	Particle density (kg/m^3)

$\Delta\sigma_0$	Surface energy difference
x	Distance between dendrite tip center to particle center (m)
χ	Direction of the area normal vector (Solidification direction)
η	Direction of the of projected component of the sum of the buoyancy and flow drag force, lying in the tangential plane
h_b	Back flow zone fraction
u	Kinematic viscosity (m^2/s^2)
m_o	Fluid molecular viscosity (kg/ms)
m_t	Fluid turbulent eddy viscosity (kg/ms)
m_{eff}	Turbulent effective viscosity (kg/ms)
e	Turbulent dissipation rate (m^2/s^3)

Subscripts:

i	Coordinate directions (i = x, y, z)

CHAPTER 1. INTRODUCTION

The process of continuous casting of steel is used to cast 90% of the steel in the world [1], so small improvements in its operation can have a huge impact. The quality of steel is greatly affected by the flow pattern in the mold. The flow pattern within the mold depends on many complex phenomena including turbulent fluid flow and upstream parameters. Fig 1.1 shows a schematic of the process in the region of the mold. Molten steel is fed by a tundish, flow through a submerged entry nozzle before it enters the mold and begins to solidify. The flow rate can be controlled by either a stopper rod present at the beginning of the nozzle or a slide gate present within the submerged entry nozzle. This thesis focuses on the turbulent fluid flow and particle motion in part of the tundish region, submerged entry nozzle and vertical section of the mold.

One quality problem is the entrapment of inclusion particles. During the process impurity particles might enter the nozzle and then be carried by the flow into the mold. Alternatively, mold slag inclusions may become entrained at the meniscus, if the velocity there is too high. Inclusions from either source can become entrapped at the solidification front within the mold and cause sliver defects in the final steel product. Computational modeling of the process can help understand the flow, particle transport and entrapment phenomenon.

Chapter 2 of this thesis investigates the asymmetries in the flow pattern that can arise in the nozzle and the mold, including:

- Asymmetric flow entering the nozzle from tundish

- Asymmetric flow due to the presence of slide gate
- Asymmetric flow caused by various shapes of realistic nozzle clogs

Different 3-D nozzle geometries are modeled to see the affect of various causes of asymmetric flow in the nozzles. The steady state flow asymmetries are quantified by calculating certain defined characteristics. Asymmetric flow coming out from a clogged nozzle is introduced in the mold, to study the asymmetric flow pattern created in the mold.

Chapter 3 introduces the process of inclusion transport and entrapment within the mold. The hydrodynamic forces acting on the particle within the flow are explained. Based on these forces the inclusions are carried with the flow. A particle entrapment model developed to decide the fate of the particle close to the solidification front is explained. The model is based on force balance analysis performed on inclusions when close to the solidification front. The effect of changing various process parameters on forces present and entrapment is studied. The inclusion transport and entrapment model is then incorporated into a 3-D mold simulation. A few selected cases are then simulated to see the locations were particles get trapped in nozzle and mold.

Chapter 4 summarizes the combined conclusions drawn from both previous chapters and recommendations for future work are provided.

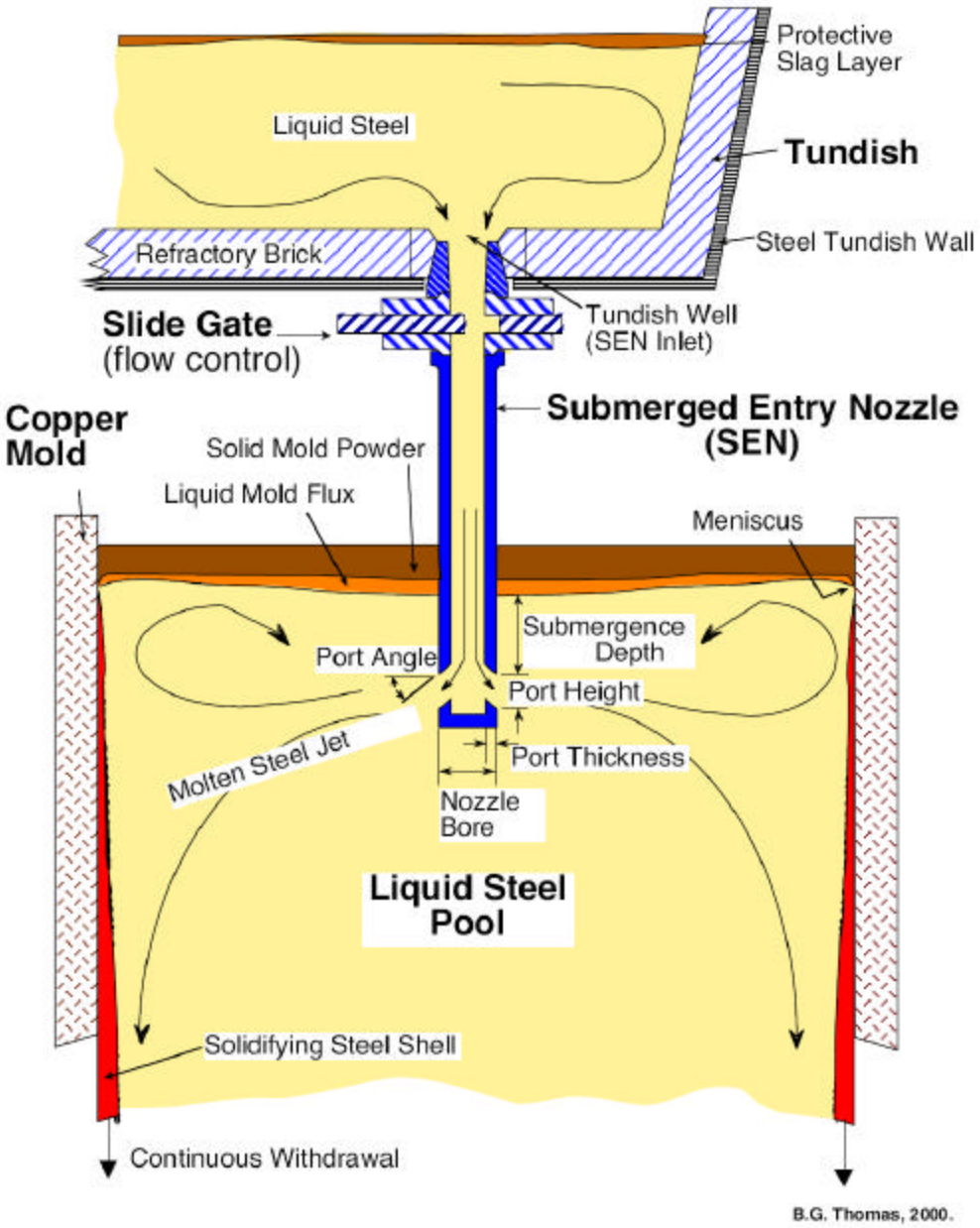


Fig 1.1. Schematic of tundish and mold region of continuous casting process [2].

1.1 REFERENCES

1. Wolf, M.M., *History of Continuous Casting*, in *Steelmaking Conf. Proc.* 1992, Iron & Steel Society, Warrendale, PA: Toronto, Canada. p. 83-137.
2. Thomas, B.G. and S.P. Vanka. *Study of Transient Flow Structures in the Continuous Casting of Steel*. in *NSF Design & Manufacturing Grantees Conference*. 2000. Vancouver, Canada: NSF, Washington, D.C.

CHAPTER 2. FLOW ASYMMETRIES

2.1 INTRODUCTION

The flow pattern in the nozzle greatly influences the continuous casting process. Asymmetric fluid flow in the nozzle can give rise to asymmetries in the mold flow pattern. This detrimental flow can cause quality problems in the final steel product [1]. Nozzle geometry, slide gate or stopper rod position, clogging, and other nozzle flow conditions can significantly affect the jet characteristics such as the jet speed and jet angle. These characteristics further affect the mold pattern. They influence the top surface fluctuations and the inclusion transport carried from these jets. The flow coming out from the nozzle enters the steel mold and this flow pattern affects the distribution of inclusion particles, relevant to the quality of steel.

Nozzle flow has been studied with computational models. Methods such as LES and Reynolds averaging have been used to model flow in 2D and 3D nozzle geometries [2-4]. Several studies have been done on how different nozzle parameters affect the flow exiting the ports into the mold [3, 5]. Najjar [3] modeled asymmetries in the nozzle ports by introducing an angled inlet velocity. This revealed a significant difference between side ports in velocities, outlet port angles (as much as 6 deg), and mass flow ratios (12%). Yokoya [6] modeled the asymmetries caused in the mold by an off-centered nozzle. Surface flow moved past the central nozzle from one side to the other at the top. Asymmetries caused by transients due to turbulent flow in otherwise steady-state conditions were quantified by Yuan in [7]. Asymmetric flow was found to

create surface level fluctuations and to enhance particle entrapment. Kubo in [8] simulated the movement of slide gate to observe time varying effect on mold flow pattern. Bai showed that the asymmetry introduced by the slide gate caused significant asymmetry between the ports entering the mold, including left-right asymmetry for a 0deg oriented gate and increased swirl for a 90deg oriented gate [2, 9].

In this thesis, the effect of various realistic geometric asymmetries in the nozzle on the time-averaged asymmetric flow leaving the nozzle ports is quantified. The analysis is made more realistic by modeling part of the tundish geometry as well as the slide gate or stopper rod, nozzle, and mold cavity. The asymmetric nature of the time-averaged flow leaving the nozzle ports causes a consistent flow bias across the top of the mold including asymmetric velocities across the mold top surface. The slag from the top surface then has greater chances to shear off with high surface velocities and enter the liquid steel [10]. These slag inclusions can get captured in the mold and thus cause sliver defects in the steel [11]. These phenomena are investigated using RANS models of turbulent flow.

In addition to asymmetric flow caused by the tundish and the slide gate, clogging is one of the primary factors creating asymmetries in nozzle flow [12]. Clogging arises from various causes such as inclusion build up, or solidification of liquid steel in a poorly preheated nozzle [13], as reviewed elsewhere [14]. Some previous work has been done to see the effect of artificial clog shapes at the slide gate region on asymmetric flow exiting the nozzle [1]. The port outflow conditions were found to depend strongly on the clogging.

This chapter investigates the importance of various causes of asymmetric fluid flow in a tundish nozzle, including:

- Asymmetric flow entering the nozzle from tundish
- Asymmetric flow due to the presence of slide gate
- Asymmetric flow caused by various shapes of realistic nozzle clogs

The asymmetries arising at the nozzle outlet ports are quantified by looking at the flow rates and jet characteristics. The effect of various clogs is compared. The clog shape was modeled based on clog samples obtained from POSCO [15]. Several actual clogged nozzles collected at the steel plant were cut into various sections and measurements of clog buildup were made [16]. These clogs were concentrated at the bottom of the nozzle in the region of the outlet ports. In addition, it has been observed that uneven clogging may develop within the nozzle bore [17] due to phenomena such as calcium build up. Such a clog has also been modeled.

Time-averaged asymmetric nozzle and mold flow patterns that arise from these different sources of asymmetry are presented and quantified in this chapter. A Reynolds-averaged turbulence model $(k-e)$ is used to model 3D time-averaged turbulent flow in different nozzle geometries. Before investigating the asymmetric flow patterns that can arise in the nozzle, the model is first verified. Yuan used LES to obtain flow solution for a typical nozzle [7] and validated from the predictions with experiments done on water models [18]. The nozzle used in [7] was modeled in this work for validating $(k-e)$ model by comparing results obtained from LES for the same nozzle. In the next chapter,

a similar flow validation will be done for flow in mold before inclusion transport and entrapment is modeled in it.

2.2 MODEL FORMULATION

The flow in the computational nozzle domains used in this chapter is 3-D and highly turbulent. The Reynolds number, based on the nozzle bore diameter [2] is of order of 10^4 as calculated in Appendix A.1.

For this level of turbulence, direct numerical solution of the unsteady Navier-Stokes equation for large 3-dimensional complex geometries is computationally exhaustive, in order to resolve all possible sub-scale eddies. Two alternative approaches can be used; Large eddy simulation (LES) and Reynolds averaging. The fundamental idea behind large eddy simulation is to resolve large scales of motion and model the dissipative effect of eddies smaller than a certain filter size, usually taken as the mesh size. Even LES is very computationally intensive, if accurate solutions are sought.

The Reynolds-averaged Navier-Stokes (RANS) equations represent transport equations for the averaged flow quantities and all the scales of turbulence are to be modeled. In this thesis, the RANS approach is used with the standard $(k-e)$ model. Hershey in [19] showed that results from separate simulations match well with a single domain comprising of both nozzle and mold. Thus for ease of convergence, nozzle and mold were separately modeled in this thesis. The values of velocity components, turbulent kinetic energy and turbulent dissipation, at the planes of the nozzle outlet ports were applied as mold inlet boundary conditions.

2.2.1 Governing Equations

The continuity equation solved is:

$$\nabla \bullet (\overrightarrow{rV}) = 0 \tag{2.1}$$

where, r is the density of steel and V is the velocity.

The momentum equation is:

$$\nabla \bullet (r\vec{V}\vec{V}) = -\nabla p + \nabla \bullet m_{eff}(\nabla \vec{V}) + r\vec{g} \tag{2.2}$$

where p is the static pressure, $r\vec{g}$ is the gravitational force and m_{eff} is defined as

$$m_{eff} = m_o + m_t \tag{2.3}$$

where, m_o is the molecular viscosity, and m_t is the turbulent viscosity.

In the Reynolds average approach used here, the variables in the Navier-Stokes equation are decomposed into mean and fluctuating components. For example for velocity components

$$u_i = \overline{u_i} + u_i' \tag{2.4}$$

where, $\overline{u_i}$ and u_i' are the mean and fluctuating components with i = 1,2,3 representing the 3 coordinate directions.

Similar expressions can be obtained for other variables and can then be substituted into the mass and momentum equations. Additional terms representing the effects of turbulence arise which need to be modeled for closure of the equations. The standard $(k-e)$ model developed by Launder and Spalding is used for this purpose [20]. Two

extra equations for turbulent kinetic energy (K) and turbulent dissipation rate (e) are solved. m_t is calculated as a function of these two quantities as:

$$m_t = rC_m \frac{K^2}{e} \tag{2.5}$$

The equations for turbulent kinetic energy (K) and turbulent dissipation rate (e) are given as:

$$\nabla \bullet (rKV) = \nabla \bullet ((m_o + \frac{m_t}{s_k})\nabla K) + G_k - re \tag{2.6}$$

$$\nabla \bullet (reV) = \nabla \bullet ((m_o + \frac{m_t}{s_e})\nabla e) + C_1 \frac{e}{K} G_k + C_2 r \frac{e^2}{K} \tag{2.7}$$

Where, G_k is represents generation of kinetic energy due to mean velocity gradients calculated as:

$$G_k = -r\overline{u_i' u_j'} \frac{\partial u_j}{\partial x_i} \tag{2.8}$$

The empirical constants are given by [20] :

$C_m = 0.09,\quad C_1 = 1.44,\quad C_2 = 1.92,\quad s_k = 1.0,\quad s_e = 1.3,$

2.2.2 Boundary Conditions

Nozzle Inlet

Across the inlet plane at the top of the nozzle, the inlet velocity value is set, based on the mass flow required to achieve the desired casting speed:

$$V_{avg} = \frac{A_{mold}}{A_{inlet}} * V_{cast} \tag{2.9}$$

Boundary values for the turbulent kinetic energy, k, and its Dissipation rate, e, also must be specified. The best inlet condition is to extend the domain upstream to model a portion of the bottom of the tundish, so that appropriate values evolve at the top of the nozzle. To simulate geometries with no tundish region, previous values have been

calculated from a mixing length model for turbulent pipe flow [21]. As the flow near the bottom of the tundish region is fairly slow relative to the nozzle inlet, its turbulent intensity is less, so small inlet values of ~ 10^{-5} are appropriate to be used for the turbulent kinetic energy and turbulent dissipation rate respectively.

Nozzle outlet / Mold inlet

A pressure boundary condition is used for the nozzle outlet boundary. The "mass flow boundary condition" available in Fluent is unreasonable in this work because it requires the unknown mass flow at each outlet port to be specified. The reference pressure at the outlet plane of the nozzle was set to atmospheric pressure of 101.325kPa, without regards to the submergence depth of the nozzle, but this has little influence on the solution [2]. The values of velocity components, turbulence kinetic energy and turbulence dissipation rate from the nozzle outlet are applied at the mold inlet as mold inlet conditions.

Mold outlet

A constant pressure condition is used at the mold outlet boundary, which represents a transverse plane through the strand that is deep enough below the mold that the fluid moves downward without recirculation.

Walls

In order to avoid excessive mesh refinement near the wall, a no slip boundary condition is specified and standard wall boundary functions are used in both the nozzle and mold [22]. As the flow remains fully turbulent throughout the nozzle and mold, this condition

is reasonable. Solidification of the shell in the mold was neglected, owing to its minor effect at the top surface of the thick mold of interest in this work.

A no slip boundary condition with wall laws for turbulence [22] is applied to all walls except at the mold top surface. A free slip condition (zero shear stress) is imposed on the mold top surface. This neglects the slight effect of slag layer, which tends to slow flow across the top.

2.2.3 Solution Procedure

A numerical grid of hexahedral cells is generated using Gambit which is directly read by Fluent [22]. The unstructured solver approach was not used, in spite of its ease of mesh generation, owing to its slow and difficult convergence. A structured mesh of hexahedral cells was carefully designed and adjusted many times to avoid any skewed elements, which tend to limit the maximum extent of convergence, or even to prevent it. The numbers of cells used vary from geometry to geometry and are stated for each case that follows.

The governing equations for mass, momentum, turbulence kinetic energy and turbulence dissipation rate, Eqs 2.1-2,2 and Eqs 2.6-2.7, are discretized using the finite-volume method. After discretization, the above conservation equations can be written in the form below:

$$a_p f_p = \sum_{nb} a_{nb} f_{nb} + b \qquad (2.10)$$

where, a_p represents the center coefficient, a_{nb} represents the neighboring coefficients,

b is the constant from the source term and f can be any scalar being solved.

Eqs 2.10 are solved using the commercial CFD program Fluent, version 6.1.22., with the "segregated solver", where the discretized governing equations are solved sequentially within each iteration. Each iteration is much faster and needs less memory than a direct solver, although more iterations are needed. First order upwinding scheme and implicit formulation is used to discretize the governing equations and the SIMPLE algorithm is used for pressure and velocity coupling [23, 24].

The residual error for each cell for each equation would be the difference between the right hand side and left hand side of Eq. 2.10. The scaled residual error for the whole domain, can be defined as follows for each variable, f:

$$R^f = \frac{\sum_{cellsP} \left| \sum_{nb} a_{nb} f_{nb} + b - a_p f_p \right|}{\sum_{cellsP} \left| a_p f_p \right|} \tag{2.11}$$

Zero velocity throughout the domain is adopted as the initial condition for all simulations. Plots of each of the 6 scaled residuals versus number of iterations are examined carefully. Convergence is taken to be achieved when the total scaled residual drops below 10^{-5}. Below this value, changes in the flow pattern were minor, as reported in the validation section. To see how to set up the case in Fluent for nozzle and mold, see Appendix A.2 and Appendix B.1 respectively.

2.3 JET CHARACTERISTICS

In addition to the velocity vectors, the flow exiting the nozzle ports is characterized

using 3 different parameters, according to previous work [2]. These characteristics include the jet angle, jet speed, and the relative size of back flow zone region. The solution at each side port outlet is extracted in excel format from Fluent. The solution extracted, comprises of calculated cell centered velocity magnitudes and face areas of the mesh, on the outlets. The equations used are given below:

Magnitude at any cell i is given as:

$$U_i = \sqrt{(u_i)^2 + (v_i)^2 + (w_i)^2} \tag{2.12}$$

Average velocity at any nozzle port in the outward x direction is

$$\bar{u} = \frac{\sum_{i \cdots ifoutflow} ((u_i) U_i (\Delta y)_i (\Delta z)_i)}{\sum_{i \cdots ifoutflow} (U_i (\Delta y)_i (\Delta z)_i)} \tag{2.13}$$

Average normal velocity at the outlet from a nozzle port in the width (y) direction is

$$\bar{v} = \frac{\sum_{i \cdots ifoutflow} ((v_i) U_i (\Delta y)_i (\Delta z)_i)}{\sum_{i \cdots ifoutflow} (U_i (\Delta y)_i (\Delta z)_i)} \tag{2.14}$$

Average velocity at a nozzle port in the downward (z) direction is

$$\bar{w} = \frac{\sum_{i \cdots ifoutflow} ((w_i) U_i (\Delta y)_i (\Delta z)_i)}{\sum_{i \cdots ifoutflow} (U_i (\Delta y)_i (\Delta z)_i)} \tag{2.15}$$

The average vertical jet angle exiting each nozzle port is

$$q_{zx} = \tan^{-1}\left(\frac{\bar{w}}{\bar{u}}\right) \tag{2.16}$$

The average jet speed is

$$U_{jet} = \sqrt{(\bar{u})^2 + (\bar{v})^2 + (\bar{w})^2} \tag{2.17}$$

The back flow zone fraction at each port is the portion of each port where re-circulating flow re-enters from the mold, and is dangerous when submergence is low, as it encourages downward suction of molten slag.

$$h_b = \frac{\sum_{all-i}((\Delta y_i)(\Delta z_i)) - \sum_{i-ifoutflow}((\Delta y_i)(\Delta z_i))}{\sum_{all-i}((\Delta y_i)(\Delta z_i))} \tag{2.18}$$

2.4 NOZZLE FLOW VALIDATION

The model in this work is first validated through comparison with the time-averaged flow pattern and velocities predicted by a Large Eddy Simulation model, which has been validated in previous work with numerous experimental measurements on water models and steel casters [18]. Specifically, a trifurcated submerged entry nozzle (SEN) with a tundish nozzle, a stopper rod and a tundish region was modeled for the same conditions as a previous analysis with the LES model which used a very refined (0.7 million cells) mesh. The model geometry is given in Fig 2.1 [7] and the operating conditions are given in Table 2.1.

2.4.1 Boundary Conditions

The inlet boundary condition was set to 0.0312m/s normal to the surface area of the cylindrical region representing a portion of the tundish bottom. The flow moves radially inwards towards the nozzle in this region. The turbulent kinetic energy and dissipation rate are both set to a magnitude of 1e-6. Other conditions are given in the previous section.

2.4.2 Mesh and Convergence

A mesh of approximately 223,000 hexahedral cells was generated to model the entire nozzle domain. Fig 2.2 and Fig 2.3 show mesh at the stopper rod and the nozzle bottom respectively. A highly dense mesh is present at the stopper rod. This is necessary in order to make a compromise between the mesh generated at the stopper rod and the bore section. The nozzle bore varies in geometry going down in casting direction. For this reason it is difficult to generate a good mesh using hexahedral cells in the bore section. Fig 2.4 shows the mesh at the cut sections A-A and B-B respectively. Fig 2.5 shows the mesh at the outlet ports. The solution converges in a few hours and about 700 iterations. Fig 2.5 shows a plot of variation in scaled residual error with iterations performed. The solution is well converged.

2.4.3 Results Comparison

Fig 2.7 show contour plots of fluid velocity in two mid planes in the nozzle. The velocity tends to increase as the fluid moves down in the nozzle, once a maximum is reached, it begins to slow down. The maximum velocity occurs at the region where the bore section changes. The bore diverges in the front view and converges in the side view as seen on these plots. Once the bore section has changed it stays the same in y direction, but continues to increase in x-direction, thus increasing the total flow area and decreasing the flow velocity. Fig 2.8 shows the velocity contours at the nozzle outlets. The maximum velocity of 0.8m/s is achieved at all ports. The flow is symmetric, because of the model setup. The mass flow rate from the left and right ports is 8.75kg/s and 8.77kg/s respectively. The slight difference between the outlet flowrates is because of numerical errors. This indicates that the asymmetric effect caused from the numerical

errors is negligible and that the asymmetries seen later in the results are real.

In order to compare with the previous results [7], the jet characteristics are calculated and listed in Table 2.2. The jet angle varies in time from ~30deg to 45deg for LES simulation [7]. The average jet angle calculated is 33deg for the $k-\varepsilon$ model which lies within the LES range. Fig 2.9(a) shows the velocity vector plots obtained for the submerged part of the nozzle in the center plane between the mold wide faces. The figure shows recirculation zones at the ports top and bottom. The back flow zone region was calculated to be approximately 9% for both side port outlets. A comparison of Fig 2.9(a) with Fig 2.9(b) shows that the velocity vectors seem to match well. Fig 2.10(a) and Fig 2.10(b) show the variation in fluid velocity components V_x and V_z at side port vertical centerline with port height obtained from $(k-e)$ model and LES. The outlet velocity profiles match well, with the back flow zone regions in both at the top and bottom. The maximum outward velocity (x) is achieved at a distance of 0.03m from the nozzle bottom from both models and the maximum downward velocity (z) is achieved 0.04m from the nozzle bottom from both models.

2.5 ASYMMETRIC FLOW CAUSED BY FLOW ACROSS TUNDISH BOTTOM

The first case investigated is the effect of asymmetric flow across the tundish bottom on asymmetry at the outlet ports. Even with a stopper rod that is perfectly aligned at the inlet of the nozzle, the flow entering the top (inlet) of the nozzle is not symmetric, owing to this flow across the bottom of the tundish. The geometries of the bifurcated

nozzle and stopper rod were provided by Nucor Steel Decatur LLC. The inner geometry of the nozzle and the stopper are modeled using a structured grid of 700,000 hexahedral cells. The modeled geometry can be seen in Fig 2.11.

To introduce the effect of asymmetry with a perfectly aligned stopper rod, flow in part of the tundish also needs to be modeled. The geometry of the part of the tundish was made so as to provide a specific casting speed of 3.6m/min. Part of the flow in tundish was modeled around the stopper rod. A cylinder segment of the flow in the tundish, around the stopper rod, was taken into account.

Unlike the symmetrical radial flow in the tundish region of the validation case, the flow in the tundish around the stopper rod was approximated as constant-velocity horizontal flow, moving from left (at 0.3m/s) to right (0.2 m/s) past the stopper rod. The difference (0.1m/s) is due to the mass flow going down into the nozzle. It is related to the casting speed, nozzle inlet cylinder (diameter, d= 2r = 279.8mm and height z=279.8mm), and mold dimensions as follows:

$$(V_{inlet} - V_{outlet})(2rz) = w * n * V_{casting} \tag{2.20}$$

where w*n are the mold cross-sectional dimensions. This is explained in the Appendix A.3 in more detail. The scaled residual error decreases monotonically, as seen in Fig 2.12.

2.5.1 Results

Fig 2.13 shows the velocity contours in two different planes over the entire length of the nozzle. The velocity increases as the nozzle converges into a smaller cross-sectional

area and then decreases as it diverges again just before leaving the outlet ports. Specifically, the downward velocity increases from approximately 2m/s at the nozzle inlet, to a maximum of approximately 3.15m/s in the nozzle bore.

Asymmetric flow is most prominent at the top portion of the nozzle as seen in Fig 2.13. An enlarged view of this is seen in Fig 2.14, where velocity vectors can be seen in the region of the stopper rod. Asymmetries diffuse away as the flow reaches the outlet. A velocity of 0.3m/s was given as the inlet of the tundish segment and a velocity of 0.2m/s was given as the outlet velocity at the tundish segment. This resulted in a higher mass flow rate at the left half z-plane than right half z-plane of the cylindrical tundish segment as seen in Fig 2.15. This scenario reverses as the fluid moves further in the downward z-direction. The incoming flow in the tundish, hits the stopper rod and flows around it as seen Fig 2.16 such that, ultimately when the flow reaches the stopper bottom, higher flow rate is in the right half plane. This transition of higher flow rate at left side to right side in z cross-sections can be seen in Fig 2.17– Fig 2.19. The mass flow rates in the left and right half planes were calculated in these three different z cross-sections, till fluid reaches the bottom of the stopper. These flow rates are listed in Table 2.3.

Fig 2.20 shows the bottom portion of the nozzle. The flow hits the bottom and diverges towards the outlet ports. The back flow regions can be seen at the top and bottom of the nozzle ports. These back flow regions occur due to the geometry of the nozzle bottom, where the fluid hits the solid surface, and the outlet ports are angled. Fig 2.21 is a plot of variation in velocity along port height, on vertical centerline. From this quantitative

plot of Fig 2.21 it can be seen that the asymmetries have died out till the flow reaches the outlet ports.

The jet characteristics, jet angle and back flow zone fraction are listed in Table 2.4. The solution at the port outlets is extracted in excel format from Fluent. Each port outlet has 660 cells. The jet velocities are symmetric to within 0.05m/s at the outlet ports. The jet angles match within about 1deg. Similarly, the same percentage of back flow region of 13.3% occurs at both ports. This shows that the asymmetries caused at the nozzle top vanish by the time they reach the bottom of the nozzle.

It should be noted that although the asymmetric inlet flow did not cause any significant asymmetries at the nozzle outlet ports, in the actual tundish, this asymmetric flow at the top can create asymmetric clogging within the nozzle and thus the flow at the outlet ports would not be symmetric. Asymmetries are also generated by misalignment of the stopper rod, and by the nature of turbulence itself, and these were not modeled.

2.6 ASYMMETRIC FLOW CAUSED BY SLIDE GATE

Next the asymmetry caused by the slide gate is investigated. The purpose of the slide gate is to control the flow rate. The model domain of a typical nozzle geometry (used at POSCO) includes some tundish region along with the nozzle as can be seen in Fig 2.23. Two different views of the nozzle bottom can be seen in Fig 2.22. This is useful for comparing with clogged nozzles, discussed later in the chapter. The mesh was taken from Zhang [25]. The mesh was improved by removing the tetrahedral cells placed at the slide gate. 96,000 structured hexahedral cells were used to mesh the nozzle

geometry. The mesh can be seen in Fig 2.24. The inlet velocity direction is chosen this time so as to ensure there is no asymmetry because of the flow in the tundish so as to isolate the asymmetries caused by the slide gate. Steel enters the tundish region in radial direction. The inlet velocity has been set as to a value of 0.014 m/s which gives a flow rate of 61.6kg/s to provide a casting speed of 1.74m/min for a mold of dimensions 1300mmx230mm. The inlet turbulence kinetic energy and dissipation rate values are set to 1e-4m^2/s^2 and 1e-4m^2/s^3 respectively. The outlet pressure is set to 101.325kPa. The liquid steel has a density set equal to 7020 kg/m^3 and a viscosity of 0.0067 kg/m-s. Fig 2.25 shows how well the scaled residual errors decrease with the iterations. The solution converges in about 700 iterations.

2.6.1 Results

The slide gate is symmetric in the x-z plane as shown in the Fig 2.26. This orientation of the slide gate, often referred to as "90 degree" orientation, causes flow to be symmetric in x-z plane but asymmetries are caused in the x-y plane as seen in Fig 2.27. Fig 2.28 shows the velocity vectors in the nozzle close to the outlet port region in x-z plane. The flow rate coming out from both the ports in such a nozzle would be the same.

The jet characteristics for the nozzle are given in Table 2.5, which shows the asymmetry between the ports. The flow itself from each port will not be symmetric about the vertical z axis. The Contours plots and vectors plots of the left and the right ports are shown in Fig 2.30. The velocity vector plots for the outlet ports clearly capture the recirculation zone present in x-y plane and their dominance in one corner.

Fig 2.29 quantifies the velocity variation on a vertical centerline for the two ports. This symmetry in velocity for both ports was expected as the slide gate is symmetric in x-z plane. The asymmetry existing about the vertical axis on each individual port is significant. In order to quantify this asymmetry each port is divided into two regions about the vertical z-axis. One half of each port lies on the negative y axis and other half on positive y axis. Flow rate in both halves is noted. The flow rates are listed in Table 2.6.

It is interesting to note that although the slide gate directs the flow towards the back of the nozzle (negative y direction), as can be seen in Fig 2.23, the flow rate exiting the ports is greater towards the front region of the ports (positive y direction). More importantly, Strong asymmetric recirculation currents, or "swirl" is created at the nozzle outlet ports because of this slide gate asymmetry. This is shown in Fig 2.30.

2.7 ASYMMETRIC FLOW CAUSED BY DIFFERENT CLOG SHAPES

Asymmetric flow is known to be caused by nozzle clogging [12, 26]. The effect of different shapes of nozzle clogs on asymmetric flow exiting the nozzle ports is studied here using the same nozzle geometry is used as in the previous section. Fig 2.31 shows sections cut through the bottom portion of three different nozzles [16]. As seen, the clogging is concentrated at the bottom, near the fluid exits the ports. It develops gradually on the side walls and builds up in the top regions of ports. The well of the nozzle is also prone to clogging.

Since the nozzle is clogged, the flow rate is decreased, and thus the inlet velocities coming from the tundish are decreased from 0.014m/s for the unclogged nozzles to 0.007m/s for all clogged nozzle simulations. This gives the flow rate of 30.8kg/s and the casting speed is reduced to only 0.87m/min. Each clogged nozzle modeled contains 80,000- 100,000 hexahedral cells. The scaled residual error arrived at from each simulation are given in Fig 2.38. Getting the scaled residual error below 10^{-6} ensures that the flows are properly converged and the asymmetries that would be seen in these flows are not due to numerical errors.

2.7.1 Clogs Studied

In the first clogged nozzle simulated (Clog1), the geometry of the flow region is restricted to represent an asymmetric clogging of the nozzle bore, as shown in Fig 2.32. Comparing Fig 2.32 with Fig 2.22 shows the shape of clog in the bore. The quantitative measurements on Fig 2.33 show the asymmetry present in the clog. Clogging is also present at the side walls at the nozzle outlet region and thus slightly decreases the width of the outlet ports.

The second clog shape (Clog 2(a)) is based directly on measurements taken in actual clogged alumina-graphite nozzles after 5 hours of casting ultra-low carbon steel with casting speed of 1.34m/min. Various sections cut through the clogged nozzle are shown in Fig 2.31. The flow region for this clog can be seen in Fig 2.34. A mixture of steel and alumina inclusions is observed to decrease the outlet port height and width, and to partially fill the bottom well. The next clog (Clog 2(b)) has the same flow region as

Clog 2(a), but with the slide gate position reversed. Fig 2.35 shows the change in geometry of the outlet ports for clog 2(a) with respect to original ports (with no clogging). This shows that the clogging at the ports is asymmetric.

The final clog shape (Clog 3), had the same flow region as clog 2(a) except that the clogging of the bottom well was increased from partially clogged, to completely clogged. The bottom flow regions modeled for the Clog 2(b) and Clog 3 can be seen in Fig 2.36 and Fig 2.37 respectively.

2.7.2 Results

Fig 2.39 and Fig 2.40 show the mid x-z plane and mid y-z planes respectively for all the cases modeled. Flow in the upper portion of all of the nozzles is similar to the un-clogged nozzle, as the effect of the clogs cannot propagate upstream. Fig 2.41 shows the velocity vector plots at the nozzle bottom in mid x-z plane for each clog. The flow is asymmetric about a center vertical axis. Fig 2.42 and Fig 2.43 show the velocity contours and velocity vectors respectively for left and right outlet ports for all 4 types of clogs. The flow patterns are different from left port to right port. In addition, asymmetries are present in each individual port as well about the vertical axis. In order to quantify the flow difference from left port to right port, velocity variation along port height at the port vertical centerline is given for each port in Figs 2.44 – 2.47. From these plots it is clear that flow asymmetry is significant for all types of clogs, and it is not easy to guess which clog shape has the most asymmetry. The jet characteristics for each clog shape are listed in Table 2.7. Flow rates through the left and right ports are noted for each clogged nozzle simulation, and are listed in Table 2.10. The ratios

$\frac{(Port\ Area)_{min}}{(Bore\ Area)_{min}}$ and $\frac{(Port\ Area)_{min}}{(Gate\ Area)_{min}}$ help in understanding flow in a given nozzle geometry. Clogging can alter these ratios significantly. For each different type of clog, the areas and the ratios are listed in Table 2.9 and Table 2.10.

Effect of Bore Clogging

For clog 1, which has the highest port to bore area, the recirculation zones at the port top enlarge to approximately 40% of the port area.

Usually the SEN for steel casting applications is designed with the combined area of the ports being larger than bore area. This is done to reduce the flow restriction caused by the ports and to accommodate some inclusion build-up in the port without affecting the flow of molten steel into the mold [1]. If the "port-to-bore" ratio for clog 1 had not increased because of bore clogging, the recirculation zones present would have accommodated some clog build-up at the top of the ports. Of course for better utilization of the ports, the ratio should not be this high.

Clog 1 experienced the maximum difference in flow rate between left and right ports of 10%, with 45% exiting from the left port and 55% from the right port. The jet speed coming out from the left and right ports is approximately the same and the horizontal jet angle is 3deg higher at the left port, which was located below the more severely clogged part of the bore. This shows asymmetric flow is very sensitive to slight asymmetries in the clog shape, when the bore is clogged severely.

Table 2.8 lists the mass flow rate from the front and back half of each port. It can be

seen that the mass flow rate at the front is only 0.5kg/s more than the back, compared to 8.6kg/s difference for slide gate nozzle with no clogging. The reduction in bore diameter has helped reduce the asymmetry created by the slide gate.

Effect of Port Clogging

For the other two types of clogs in which the flow region was the same and only the slide gate orientation was different, a flow rate difference of 6% is obtained. This is smaller than the difference for Clog 1 because the severity of the asymmetry of the clog is less (See Figs 2.33 vs 2.35). The back flow region became 0% compared to 13.48% back flow region in the slide gate nozzle without any clogging. The difference in jet characteristics of each port can be seen in Table 2.7. The Maximum difference between mass flow rate at front and back is about 1.6kg/s seen in Table 2.8. The reduction in nozzle width at the bottom has helped reduce the asymmetry created by slide gate.

Effect of Complete Well Clogging

Clog 3 (where the well is completely clogged) experienced significant flow asymmetry of 10%. A comparison of jet characteristics of Clog 3 and Clog2(a)/Clog2(b) (where the well is slightly clogged), given in Table 2.7 show that clogging the well completely increased the difference between horizontal jet angle for the ports. Also Clog 3 created a back flow region of 8% on one port and 0% on the other. Comparison of asymmetry in flow rate between Clog 2(a)/Clog 2(b) and Clog 3 given in Table 2.10. Although completely clogging the well as seen in Fig 2.39 produces a more symmetrical shape of flow region, the asymmetry increased. This suggests that using a well in the nozzle bottom helps to reduce asymmetry.

2.8 ASYMMETRIC FLOW IN THE MOLD

Flow in the entire mold cavity was modeled, neglecting the shell, to investigate the effect of asymmetric flow caused by nozzle clogging. The inlet velocities were taken from the nozzle simulation for the clogged bore (Clog 1). Fig 2.48 shows the mesh in the mold. The mesh is highly concentrated in the regions near the inlet of the mold. This can not be avoided, as the mesh at the mold inlet is to be the same as that at the nozzle outlet.

2.8.1 Results

The flow rate from the clogged nozzle was asymmetrical, with 45% from the left port and 55% from the right port. This difference caused significant asymmetries in the mold. Fig 2.49 shows the velocity contours on a plane mid way between the wide faces. The asymmetry between the two halves of the mold appears not much more than that present in transient runs with no clogging, done using LES [7]. A better view with velocity vectors in this plane in Fig 2.50 shows clearly the significant difference in the top roll pattern between the two halves of the mold.

A top view of velocity vectors on the top surface of the mold given in Fig 2.51 shows the significant difference between the two sides caused by asymmetries entering in the mold. The velocity flowing toward the SEN along the right hand side is so strong that liquid steel flows at high velocity between the nozzle outer wall and the mold, enters the left half of the mold, and disturbs the flow pattern on the left side, creating vortexes. The quantitative velocity profile in Fig 2.52 shows that the velocity across the top

surface is twice as high on the right side. The low velocity on the left can lead to meniscus freezing and hook defects in the steel surface. At higher casting speeds, the higher velocity on the right could shear off liquid slag into the liquid steel and thus cause inclusion defects [27]. In addition, the higher velocity induces greater level fluctuations, leading to other surface defects. Fig 2.53 shows the velocity contours on planes 5mm from left and right narrow faces.

Asymmetry is present not only between the two planes but also on each individual narrow plane about the vertical (z) axis. Fig 2.54 shows the variation in velocity down the mold length, at mid way between the mold center and narrow face. The asymmetry between the left and right side is significant. Yuan has also observed this difference in velocity between the left and right side of the mold, for transient analysis made using LES [7] but no significant asymmetry was observed at the top mold surface because of the transients.

2.9 CONCLUSIONS

The results from $(k-e)$ model compare well with time averaged results obtained from LES for the validation nozzle. A full nozzle is simulated and the almost negligible difference in jet characteristics present on both left and right hand side of the nozzle suggests the magnitude of the numerical errors present. Three different causes of asymmetry in nozzles were investigated:

- Asymmetric flow entering the nozzle from tundish
- Asymmetric flow due to the presence of slide gate
- Asymmetric flow caused by various types of nozzle clogs

No asymmetry is caused at the nozzle outlet due to asymmetry in the tundish flow. Asymmetry is present near the stopper rod region due to asymmetric flow in tundish and thus an asymmetric clog can develop in this region. The slide gate is oriented to avoid asymmetry between the left and right outlet ports of the nozzle, but generates significant swirl within each outlet. Flow entering the mold is asymmetric, with two thirds of the flow exiting the front of the ports with a horizontal angle of 31.9 deg from the left port and 35.4 deg from the right port. Increasing clog asymmetry naturally tends to increase flow asymmetry. Among the different clog shapes modeled, the most severe asymmetry is caused by nozzle clogged at the bore section and for nozzle clogging the bottom well entirely. The difference in flow rate between left and right port outlets is seen to be 10% for both these clog shapes. Clog 3 has same clog shape at the nozzle outlet ports as clog 2(a) but no well. The removal of asymmetrically clogged well created more asymmetry at the outlet ports then having no well at all. This suggests that well at nozzle bottom, helps create symmetry in flow at outlets.

The results from the nozzle clogged in the bore section, were introduced at the inlet of a mold with no shell. Significant asymmetry is present in at the top surface and also throughout the mold length. Vortexes are created on the left side of the top surface and it can cause flux entrapment. The effect is enhanced with increased casting speeds, lower submergence depths and asymmetric flow at the top surface [11]. A critical velocity of 0.3m/s at the top surface is theoretically suggested [11]. The surface velocity should be less than this critical velocity to prevent slag entrapment.

2.10 TABLES AND FIGURES

Table 2.1. Operating Conditions for validation nozzle.

Parameter / Property	
Nozzle Port Height x Thickness (mm x mm)	75 x 32 (inner bore)
Bottom nozzle Port Diameter (mm)	32
SEN Submergence Depth (mm)	127
Casting Speed (mm/s)	25.4
Fluid Kinematic Viscosity (m^2/s)	7.98×10^{-7}

Table 2.2. Jet characteristics at port outlets for validation nozzle.

	Left Port outlet	Right Port outlet
Average Velocity (Jet speed)(m/s)	0.68	0.675
Angle with the X-axis (deg)	33.76	33.08
Back Flow Zone Fraction (%)	9.2	9.4

Table 2.3. Mass flow rate showing progression of asymmetry caused by asymmetric flow in tundish.

Z-Plane	Left Half of Plane (Mass Flow Rate) Kg/s	Right Half of Plane (Mass Flow Rate) Kg/s
54mm above stopper bottom	29.480	25.322
19mm above stopper bottom	27.404	27.398
At stopper bottom	26.048	28.592

Table 2.4. Jet characteristics at port outlets for asymmetric flow in tundish.

	Left Port outlet	Right Port outlet
Average Velocity (Jet speed)(m/s)	2.56	2.55
Angle with the X-axis (deg)	31.15	30.69
Back Flow Zone Fraction (%)	13.3	13.3

Table 2.5. Jet characteristics at port outlets for nozzle with slide gate.

	Left Port outlet	Right Port outlet
Average Velocity (Jet speed)(m/s)	1.17	1.17
Angle with the X-axis (deg)	38.64	38.61
Back Flow Zone Fraction (%)	13.48	13.48

Table 2.6. Different mass flow rates for asymmetry caused by slide gate.

	Flow Rate (back half) (Region on negative Y-axis)	Flow Rate (front half) (Region on positive Y-axis)
Left Port	11.1 kg/s	19.7 kg/s
Right Port	11.1 kg/s	19.7 kg/s

Table 2.7. Jet characteristics at port outlets for all clog shapes.

Clog 1	Left Port outlet	Right Port outlet
Average Velocity (Jet speed)(m/s)	1.89	1.99
Angle with the X-axis (deg)	31.90	35.37
Back Flow Zone Fraction (%)	41.2	39.4
Clog 2(a)	Left Port outlet	Right Port outlet
Average Velocity (Jet speed)(m/s)	1.08	0.98
Angle with the X-axis (deg)	34.70	40.90
Back Flow Zone Fraction (%)	0	0
Clog 2(b)	Left Port outlet	Right Port outlet
Average Velocity (Jet speed)(m/s)	1.08	0.98
Angle with the X-axis (deg)	34.99	41.07
Back Flow Zone Fraction (%)	0	0
Clog 3	Left Port outlet	Right Port outlet
Average Velocity (Jet speed)(m/s)	1.09	1.03
Angle with the X-axis (deg)	34.05	43.46
Back Flow Zone Fraction (%)	0	8.9

Table 2.8 Mass flow rates in back half and front half of two different clog shapes.

	Flow Rate (back half) (Region on negative Y-axis)	Flow Rate (front half) (Region on positive Y-axis)
Clog 1 Left Port	6.6 kg/s	7.3 kg/s
Clog 1 Right Port	8.7 kg/s	8.2 kg/s
Clog 2(a) Left Port	9.0 kg/s	7.4 kg/s
Clog 2(a) Right Port	7.6 kg/s	6.8 kg/s

Table 2.9. Comparison of parameters for different clogged nozzles.

	Left Port Area (mm^2)	Right Port Area (mm^2)	Gate Area (mm^2)	Bore Area (mm^2)
Un-Clogged	7159	7159	2506	4141.8
Clog 1	3252	3622	2506	1256
Clog 2(a), 2(b), 3	2871	3666	2506	4141.8

Table 2.10. Comparison of flow rates from ports for different clogged nozzles.

	Clog Shape Description	$\frac{(Port\ Area)_{min}}{(Bore\ Area)_{min}}$	$\frac{(Port\ Area)_{min}}{(Gate\ Area)_{min}}$	Mass Flow rate at Left	Mass Flow rate at Right
Un-Clogged		1.728	2.857	50%	50%
Clog 1	Bore Clog	2.589	1.298	45%	55%
Clog 2 (a)	Port and Partial well Clog	0.693	1.146	53%	47%
Clog 2(b)	Clog 2(a) with Slide gate orientation reversed	0.693	1.146	53%	47%
Clog 3	Port and Complete well clog	0.693	1.146	55%	45%

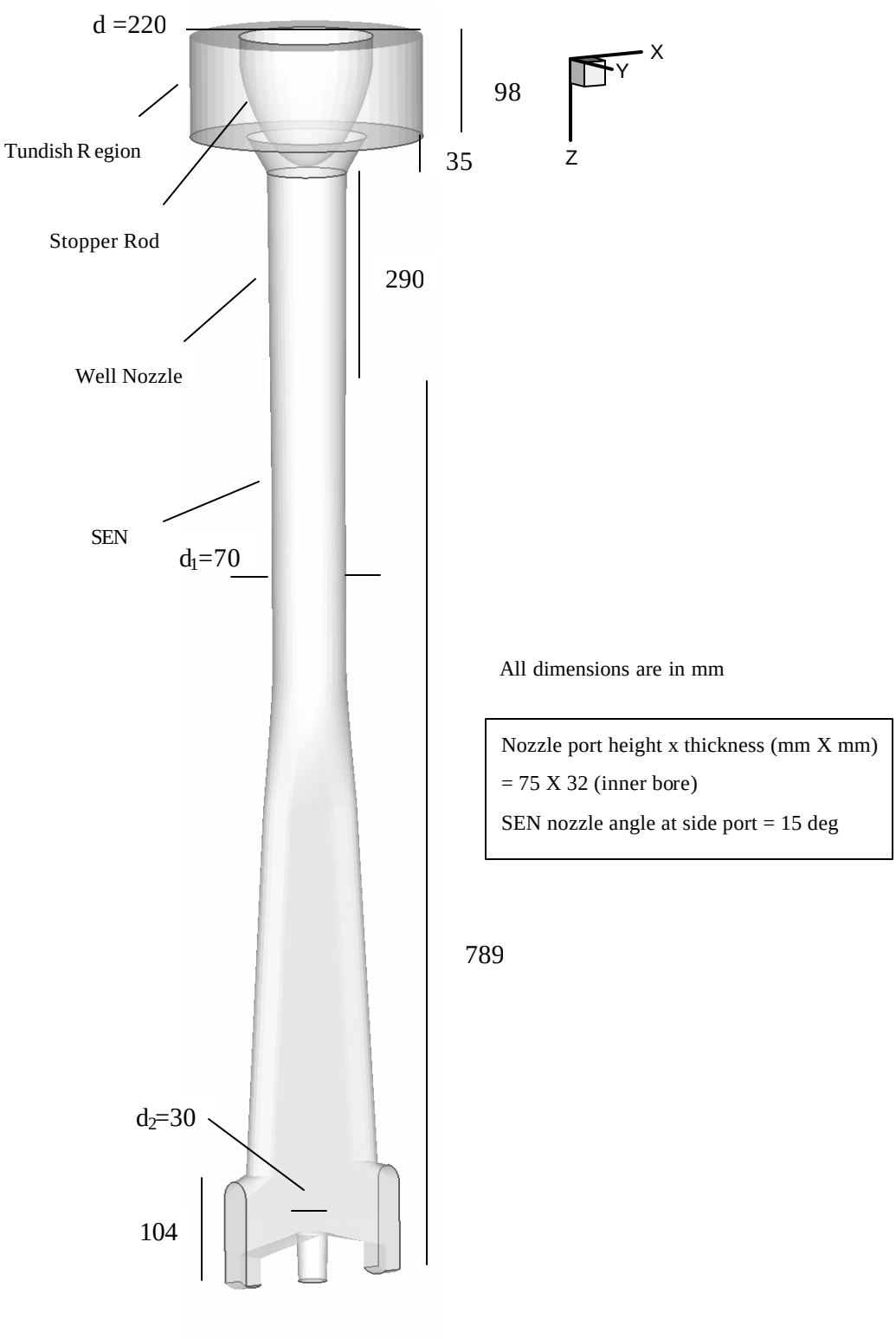


Fig 2.1. Geometry for the validation nozzle .

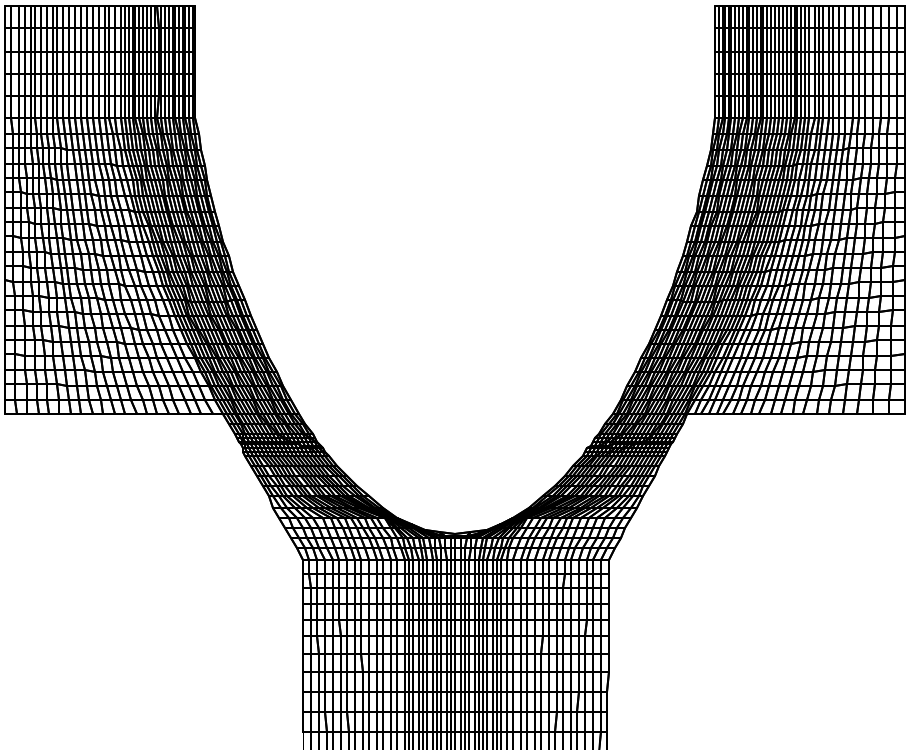


Fig 2.2. Mesh in the validation nozzle at the stopper rod region.

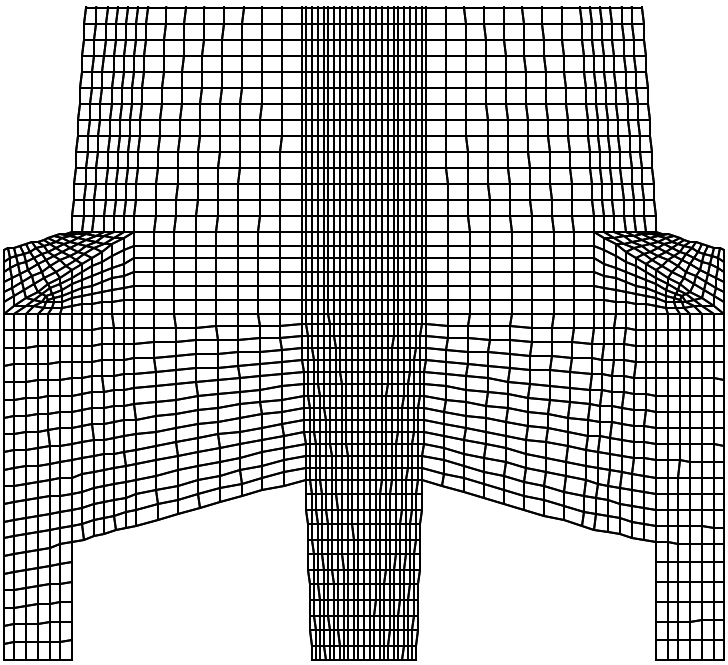


Fig 2.3. Mesh in the validation nozzle at the bottom.

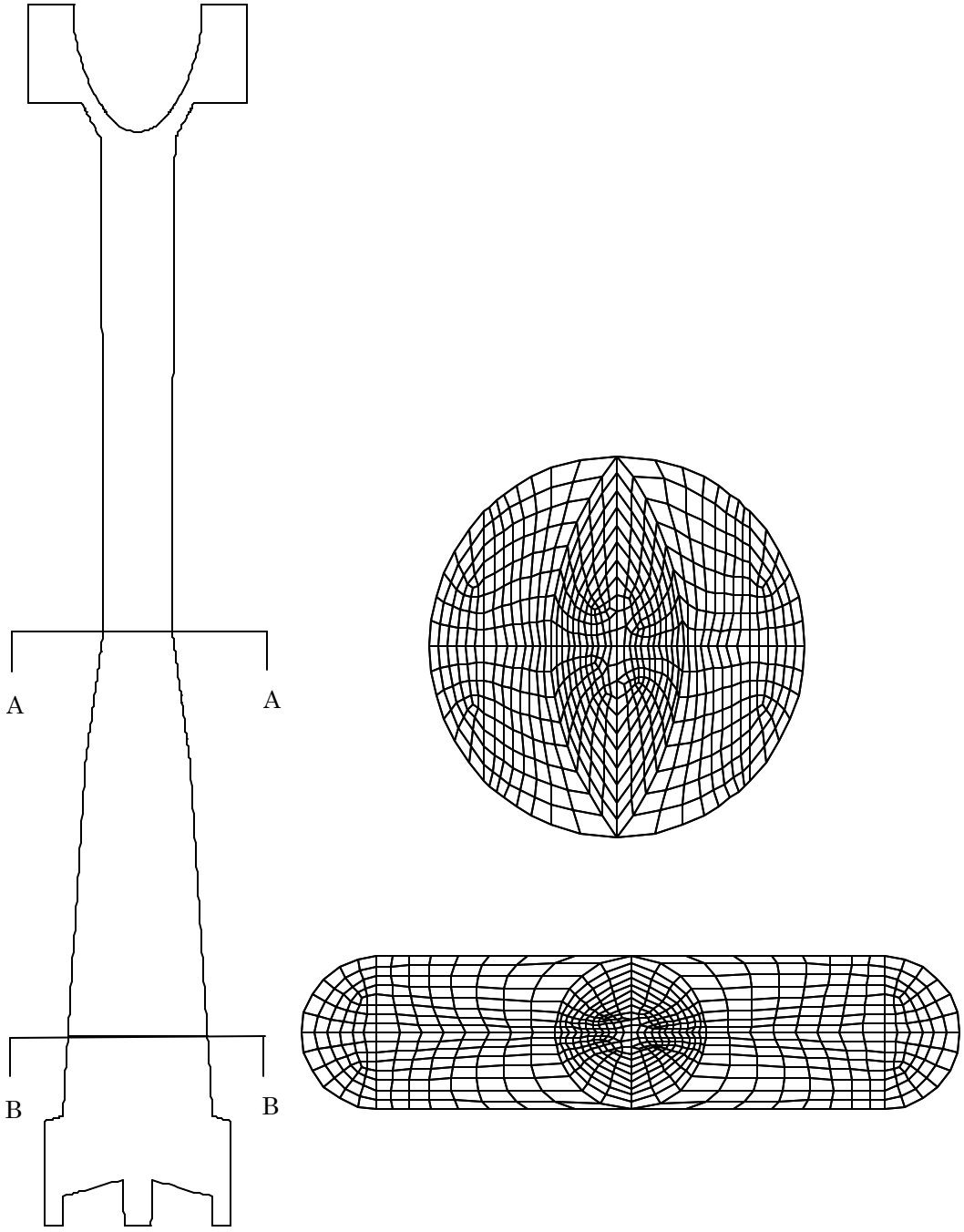
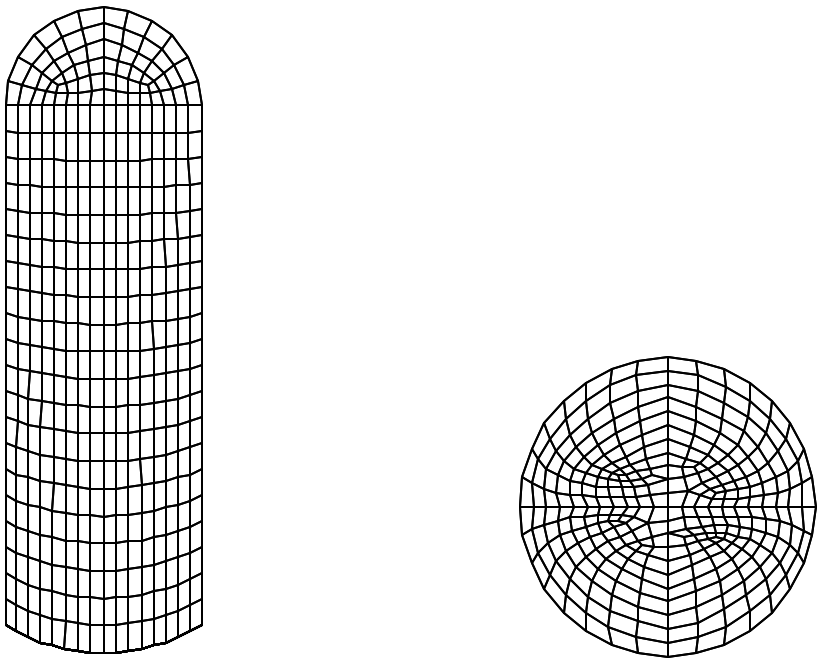


Fig 24. Validation nozzle mesh at different bore sections.



Side Port **Bottom Port**

Fig 2.5. Mesh at side port and bottom port for validation nozzle.

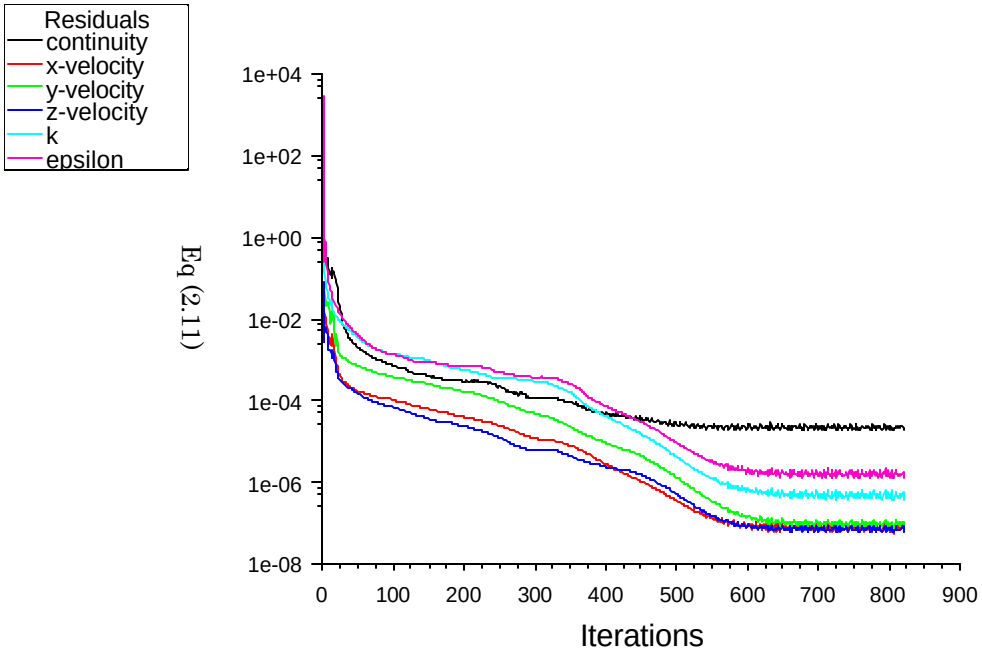


Fig 2.6. Variation in residual errors with number of iterations – validation case.

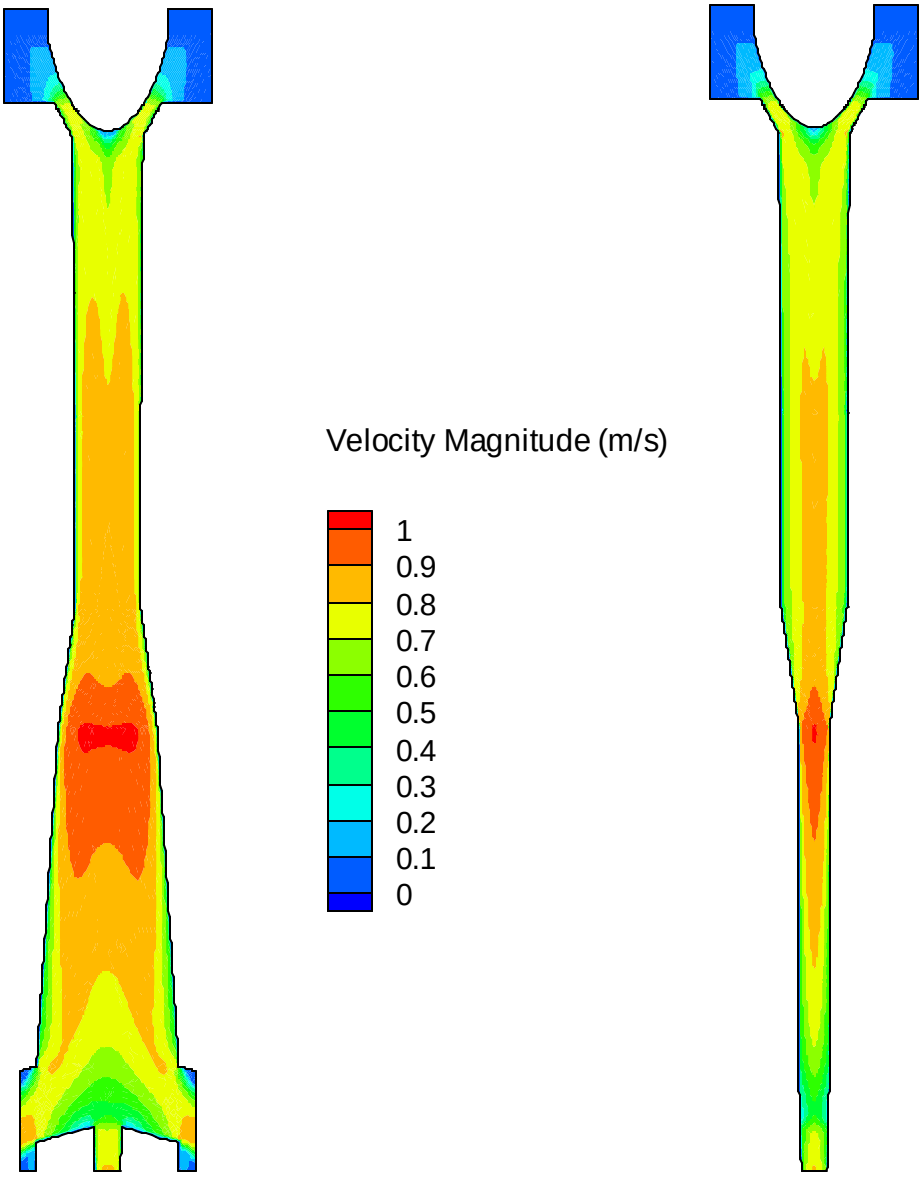


Fig 2.7. Velocity contours in front view and side view of validation nozzle.

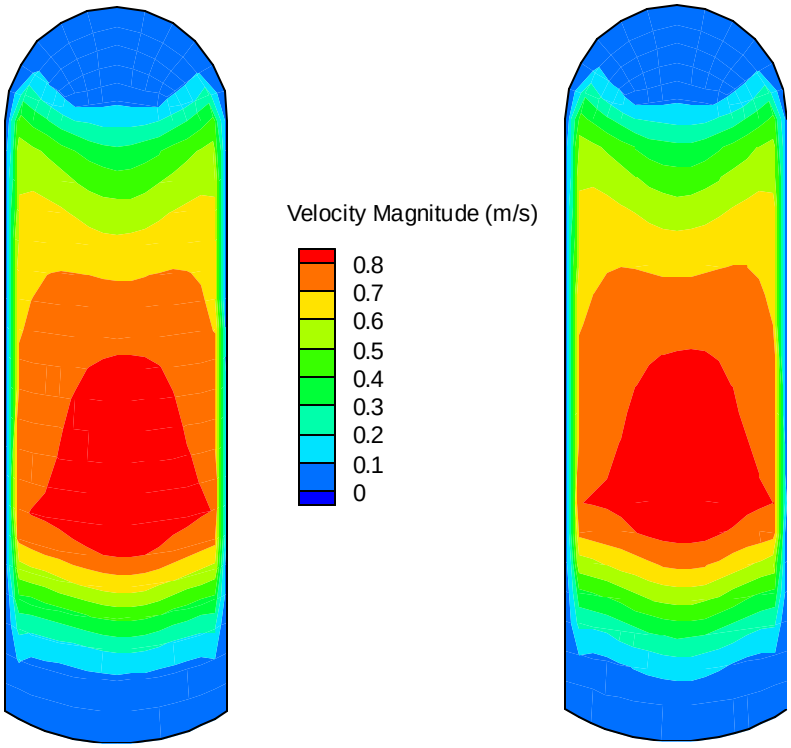


Fig 2.8. Velocity Contours at validation nozzle port outlets.

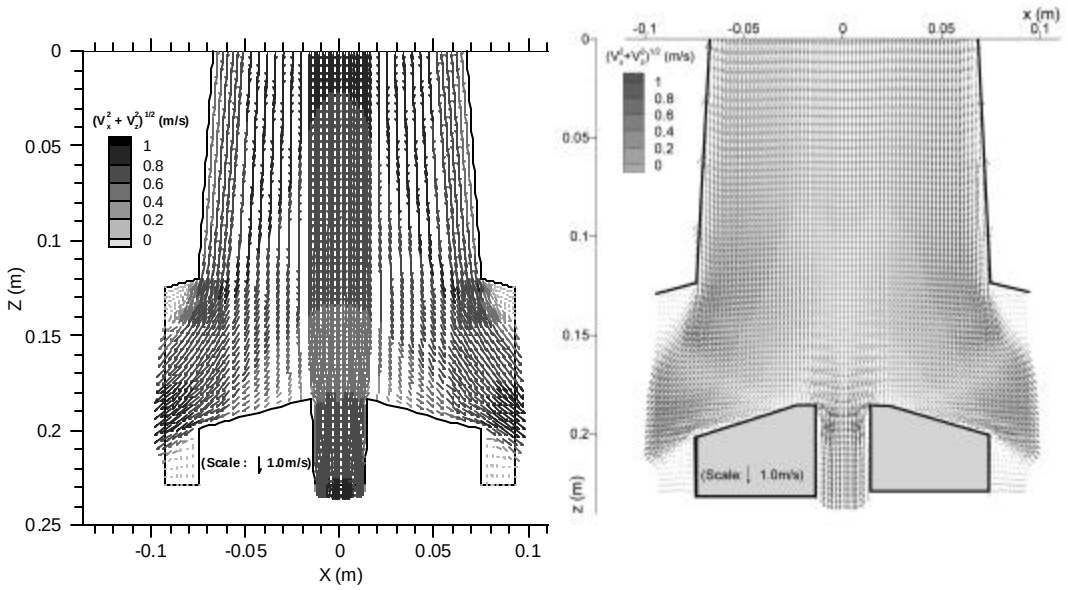


Fig 2.9. Velocity Vectors in validation nozzle near ports, comparing (a) this work with (b) previous work [7].

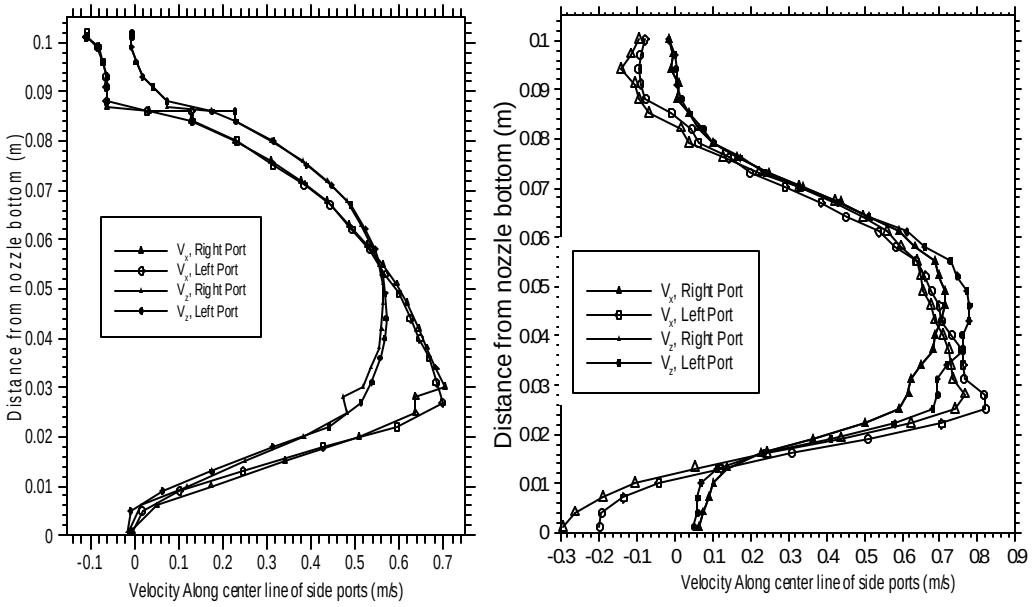


Fig 2.10. Variation in velocities at the validation nozzle ports centerlines, comparing (a) this work with (b) previous work [7].

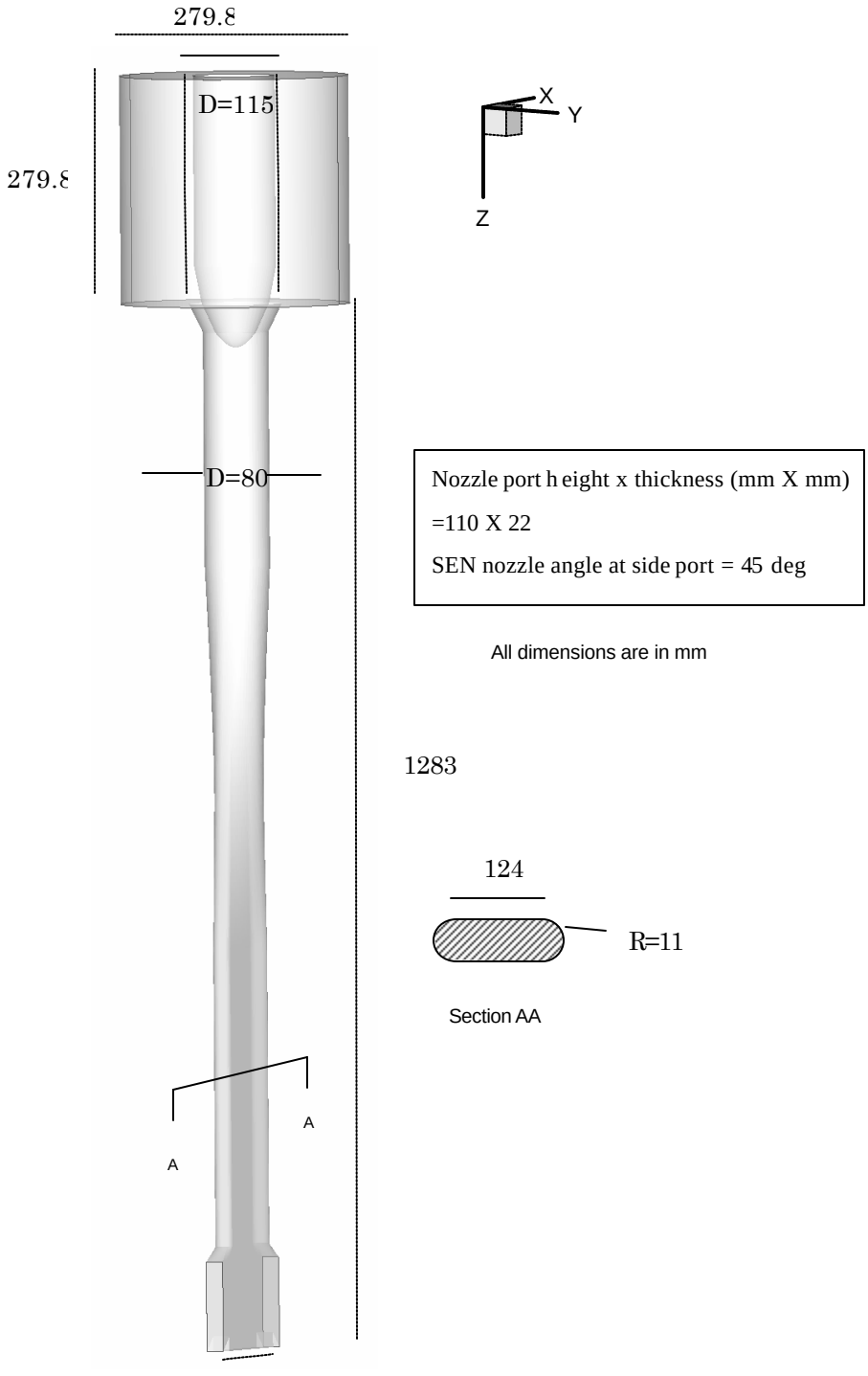


Fig 2.11. Geometry for nozzle to simulate asymmetric flow at tundish.

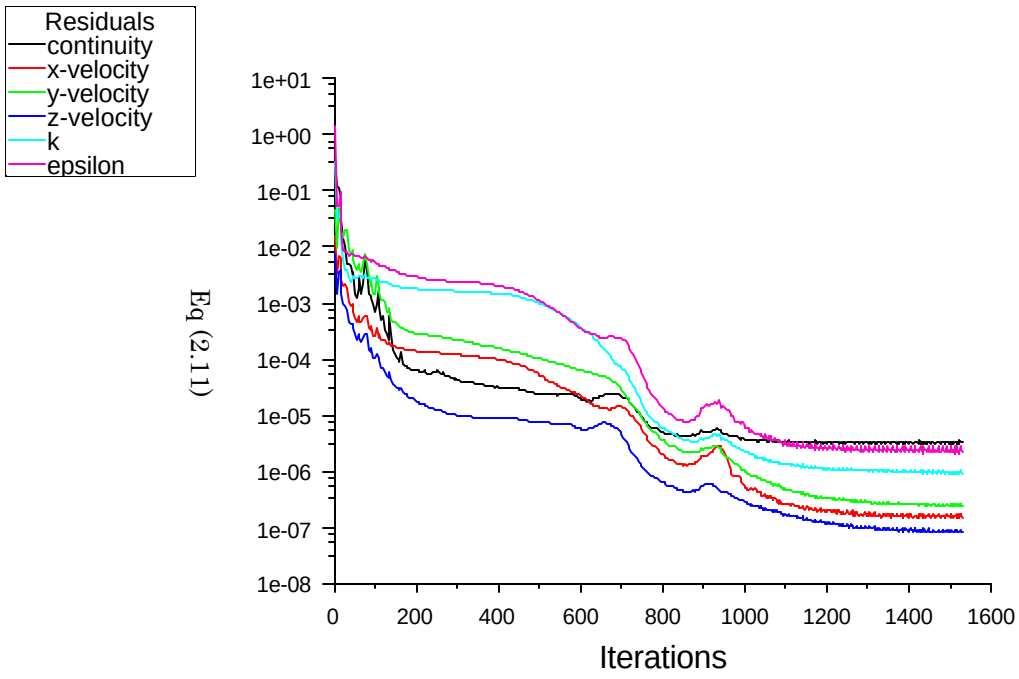


Fig 2.12. Scaled residual error for nozzle with asymmetric flow in tundish.

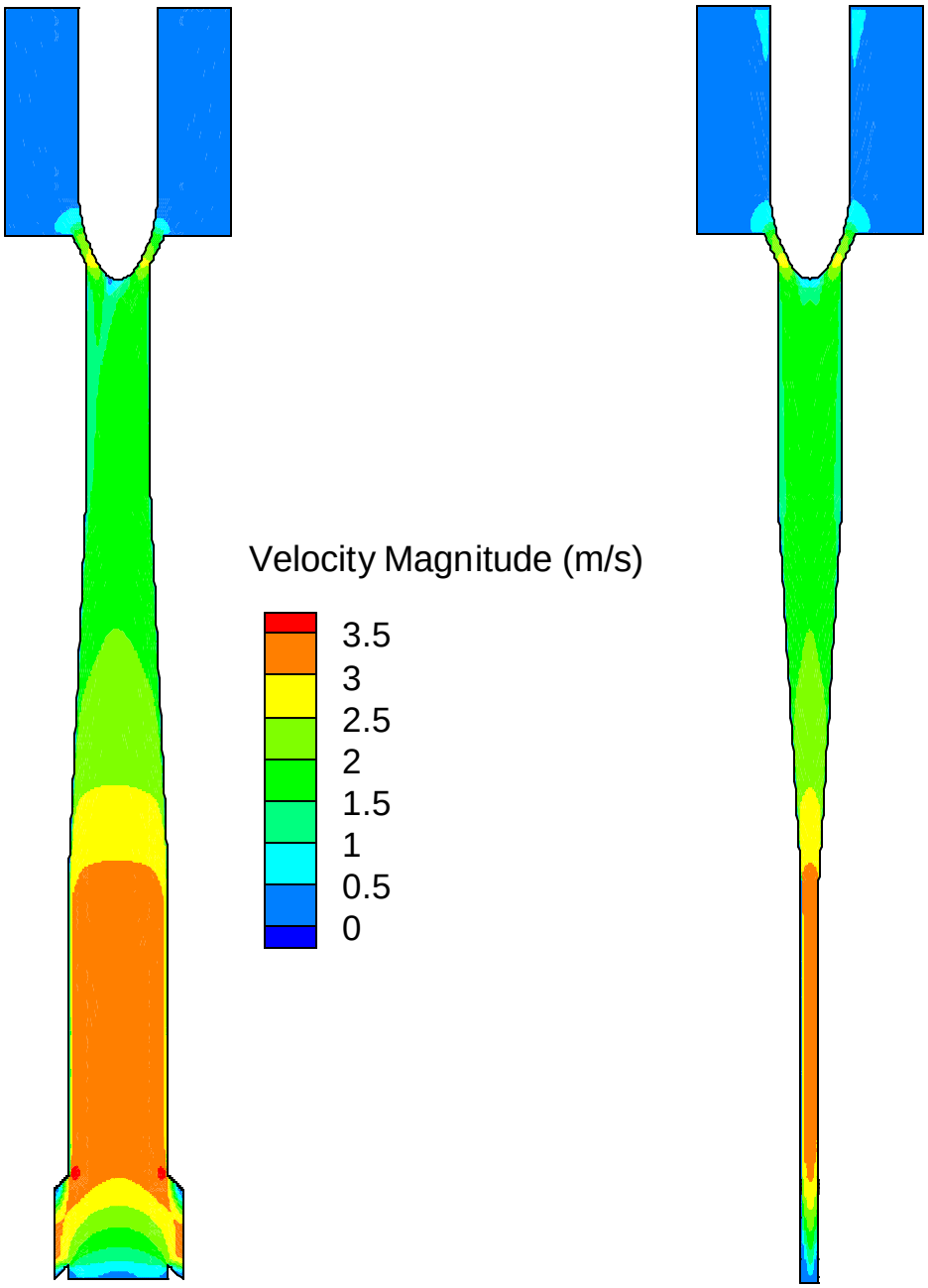


Fig 2.13. Velocity contours of mid plane front view and side view of the nozzle with tundish asymmetry.

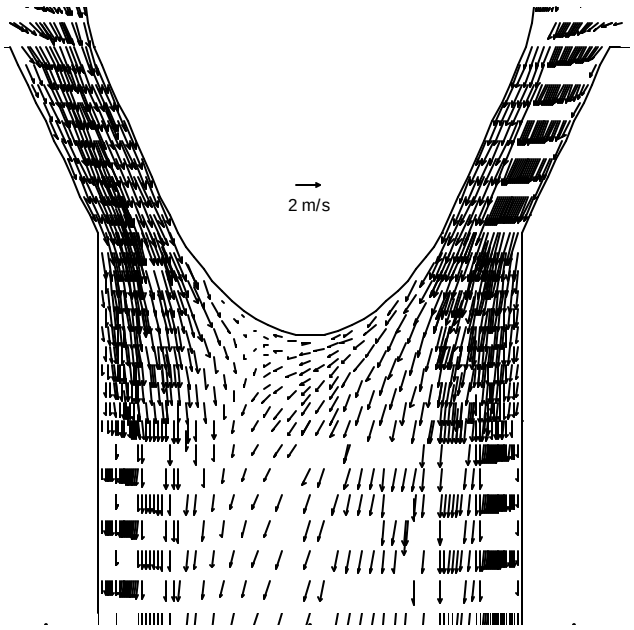


Fig 2.14. Enlarged view of velocity vectors at the bottom of the stopper rod (mid x-z plane).

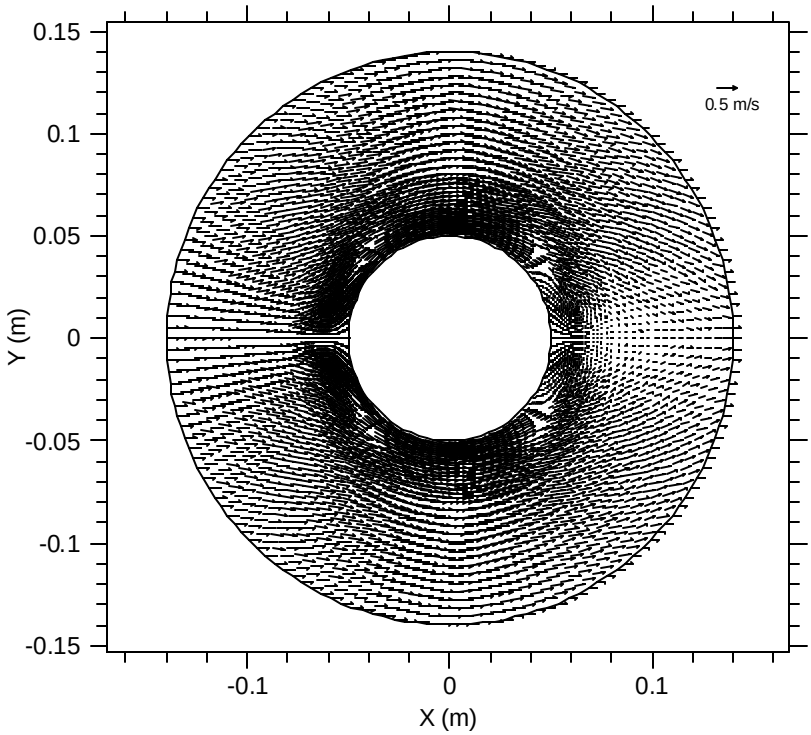


Fig 2.15. Velocity vectors in tundish on a z plane 100mm above the nozzle entry.

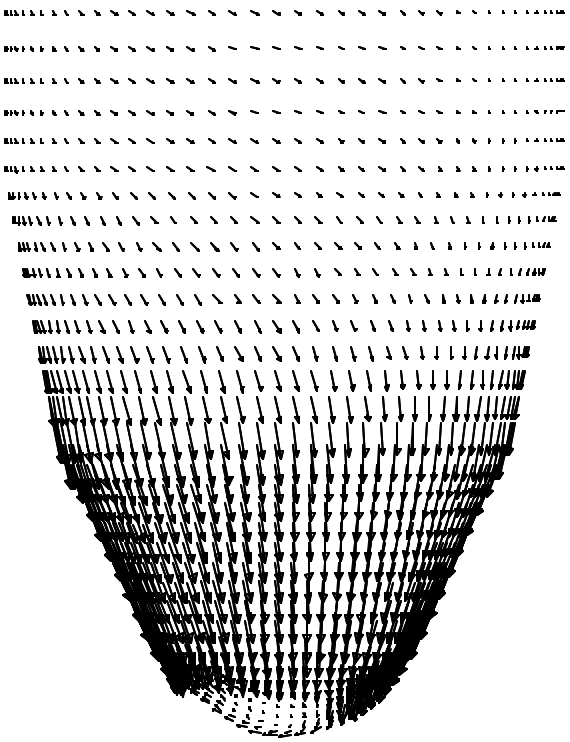


Fig 2.16. Velocity vectors close to the surface of the stopper rod.

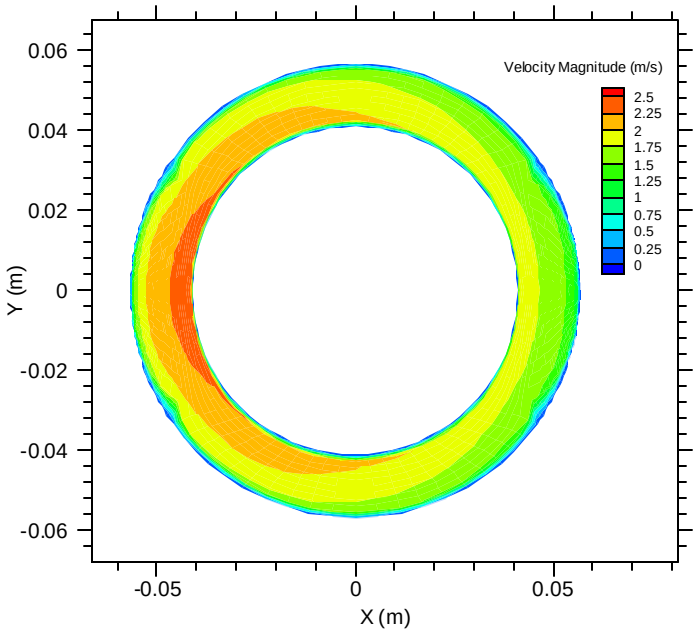


Fig 2.17. Velocity contours on a z plane 54mm above the stopper bottom.

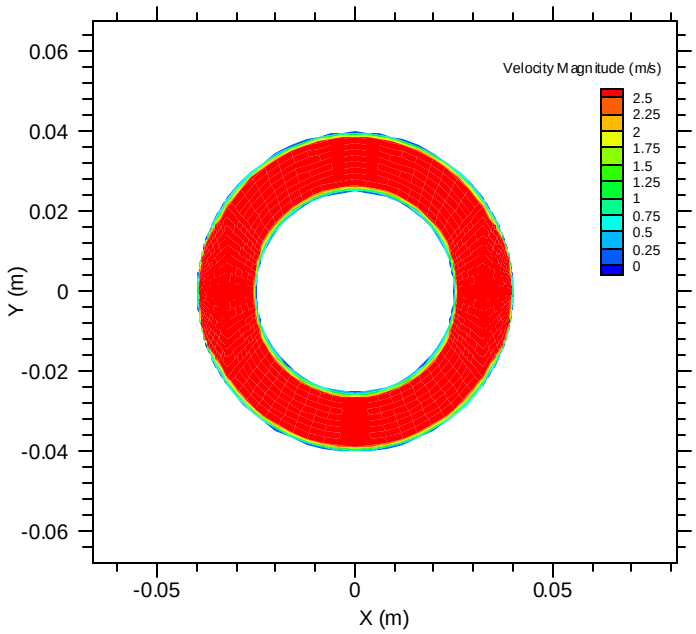


Fig 2.18. Velocity contours on a z plane 19mm above the stopper bottom.

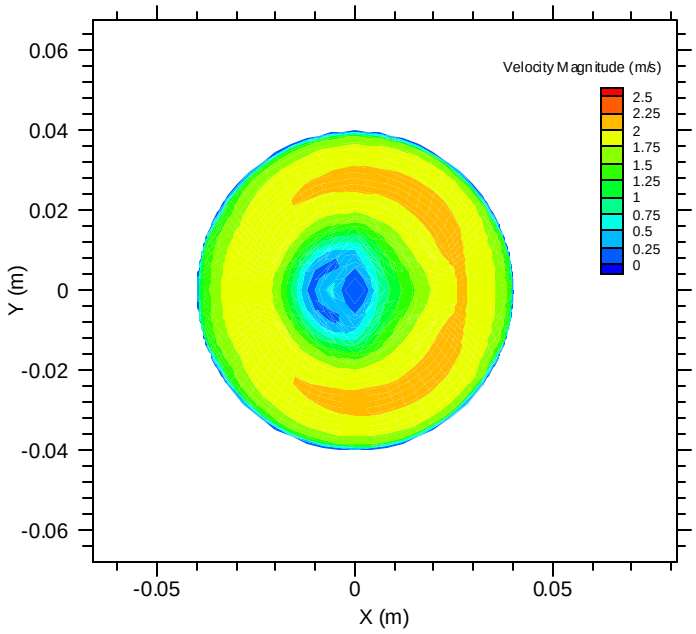
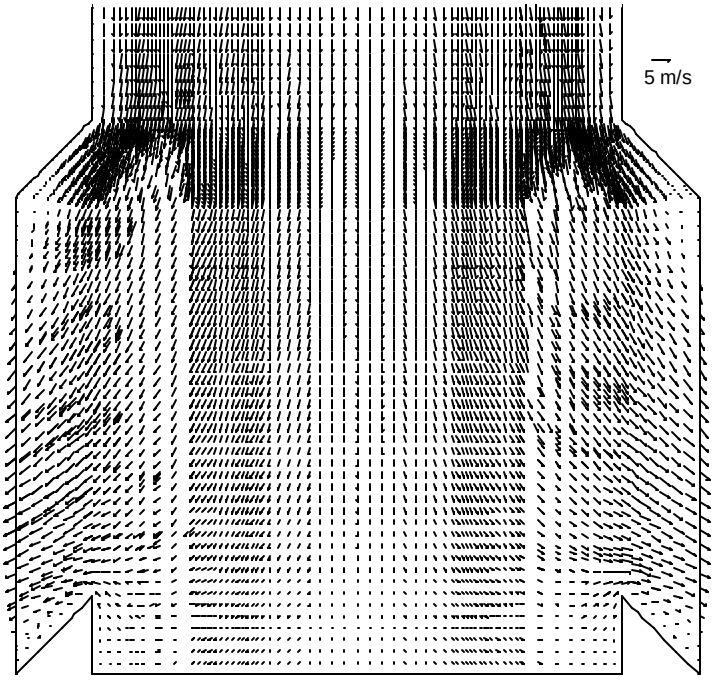
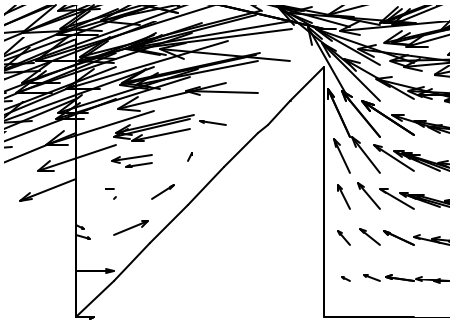


Fig 2.19. Velocity contours on a z plane at the bottom of the stopper rod.





Enlarged View of Bottom Left corner showing recirculation

Fig 2.20. Velocity vectors in the nozzle near outlet ports and an enlarged view of a recirculation region.

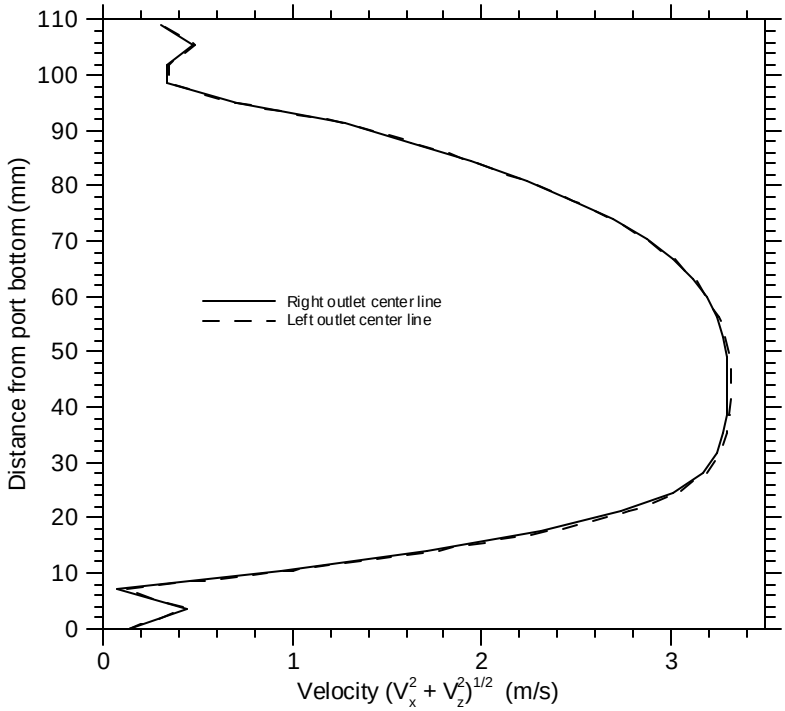
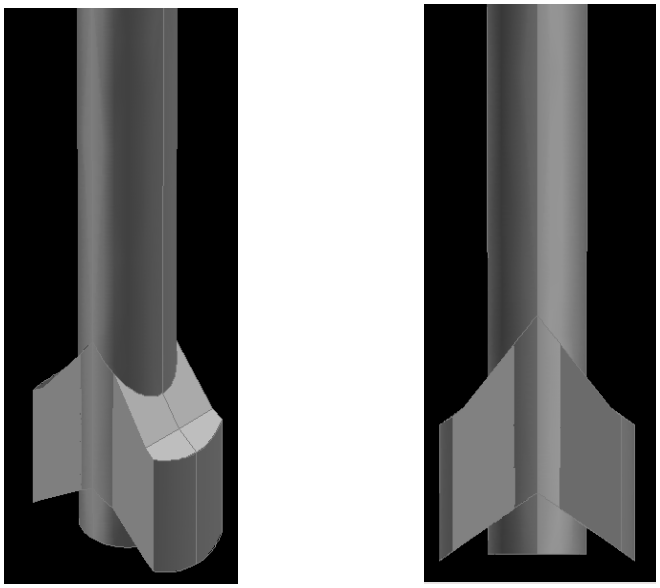


Fig 2.21. Velocity along nozzle port centerline on both sides.



(Isometric view) (Front view)

Fig 2.22. Isometric view and front view of the nozzle with slide gate (un-clogged).

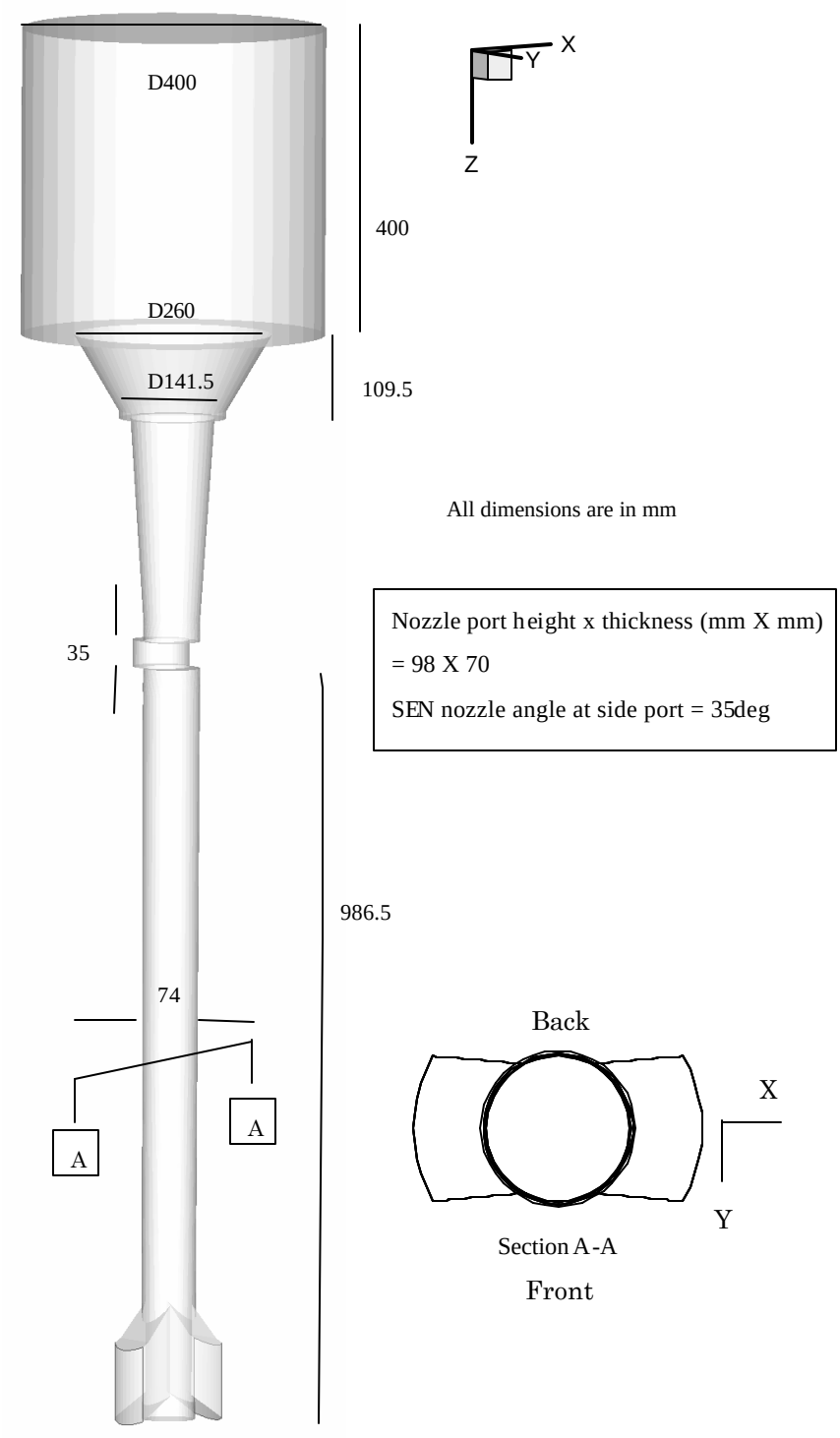


Fig 2.23. Geometry of the nozzle with slide gate .

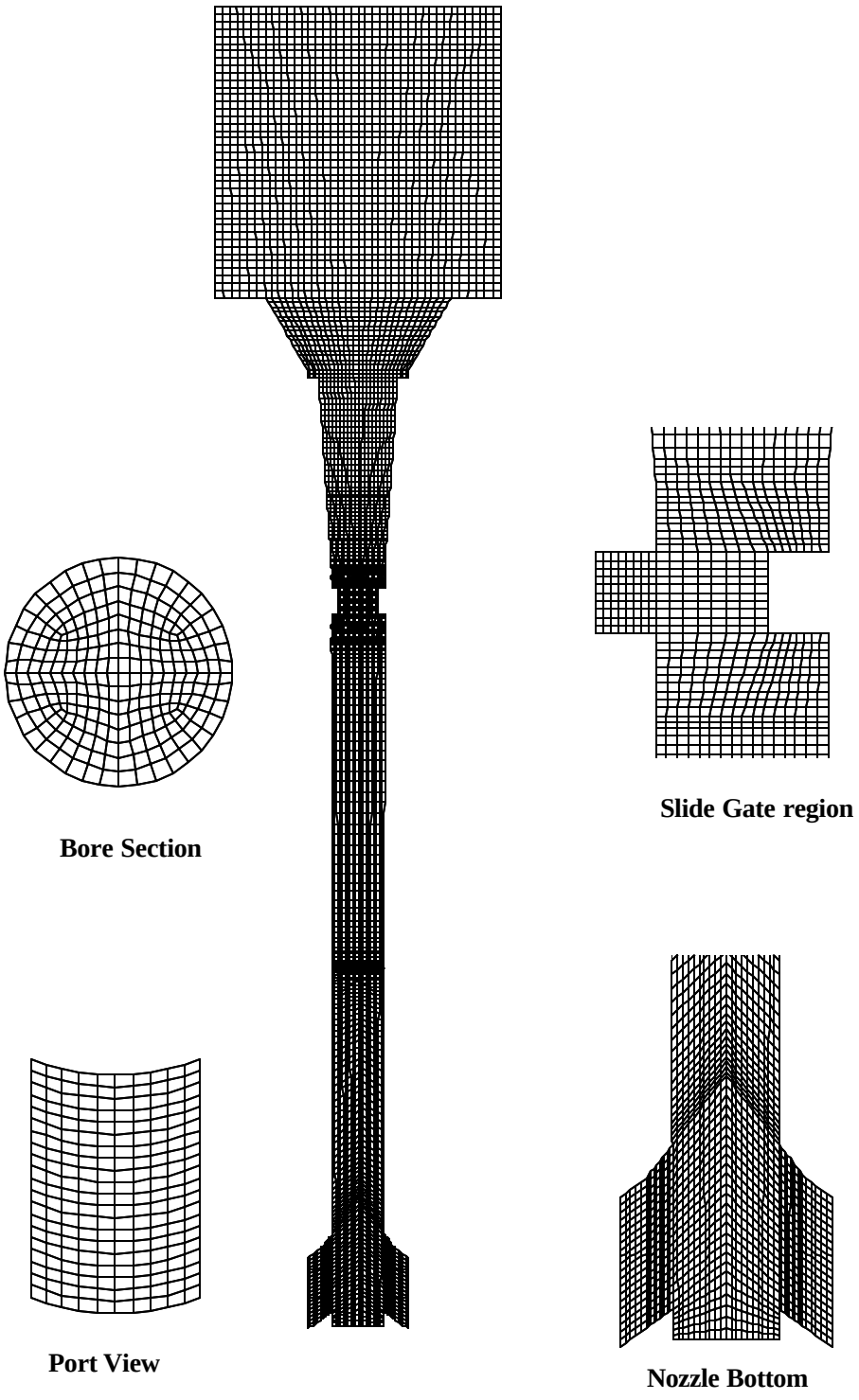


Fig 2.24. Mesh at different sections for nozzle with slide gate.

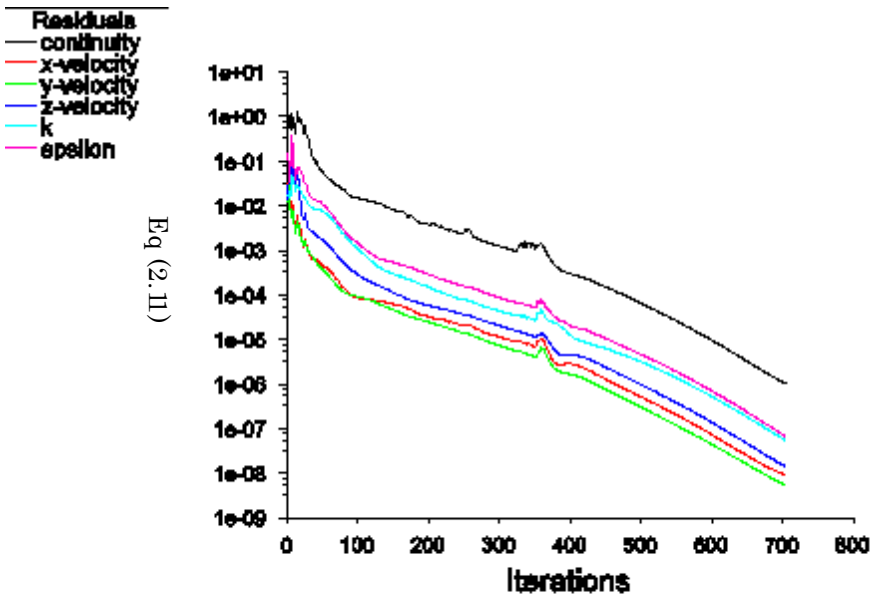


Fig 2.25. Plot of scaled residual error for original nozzle (unclogged) with slide gate .

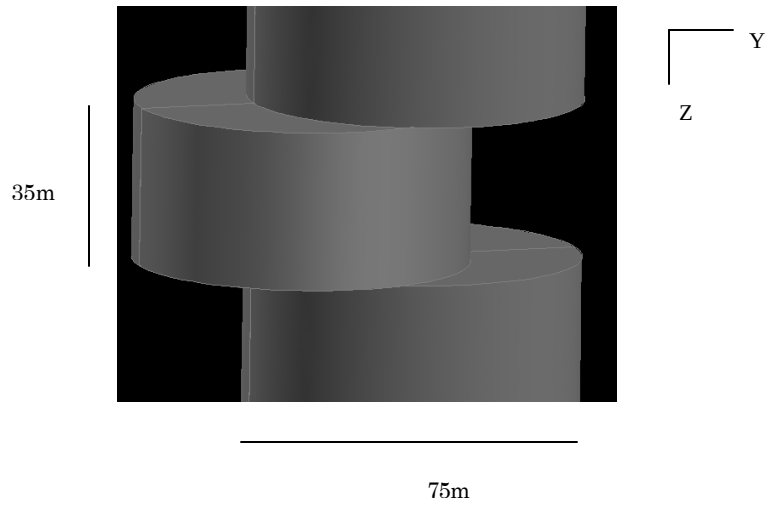


Fig 2.26. View of slide gate.

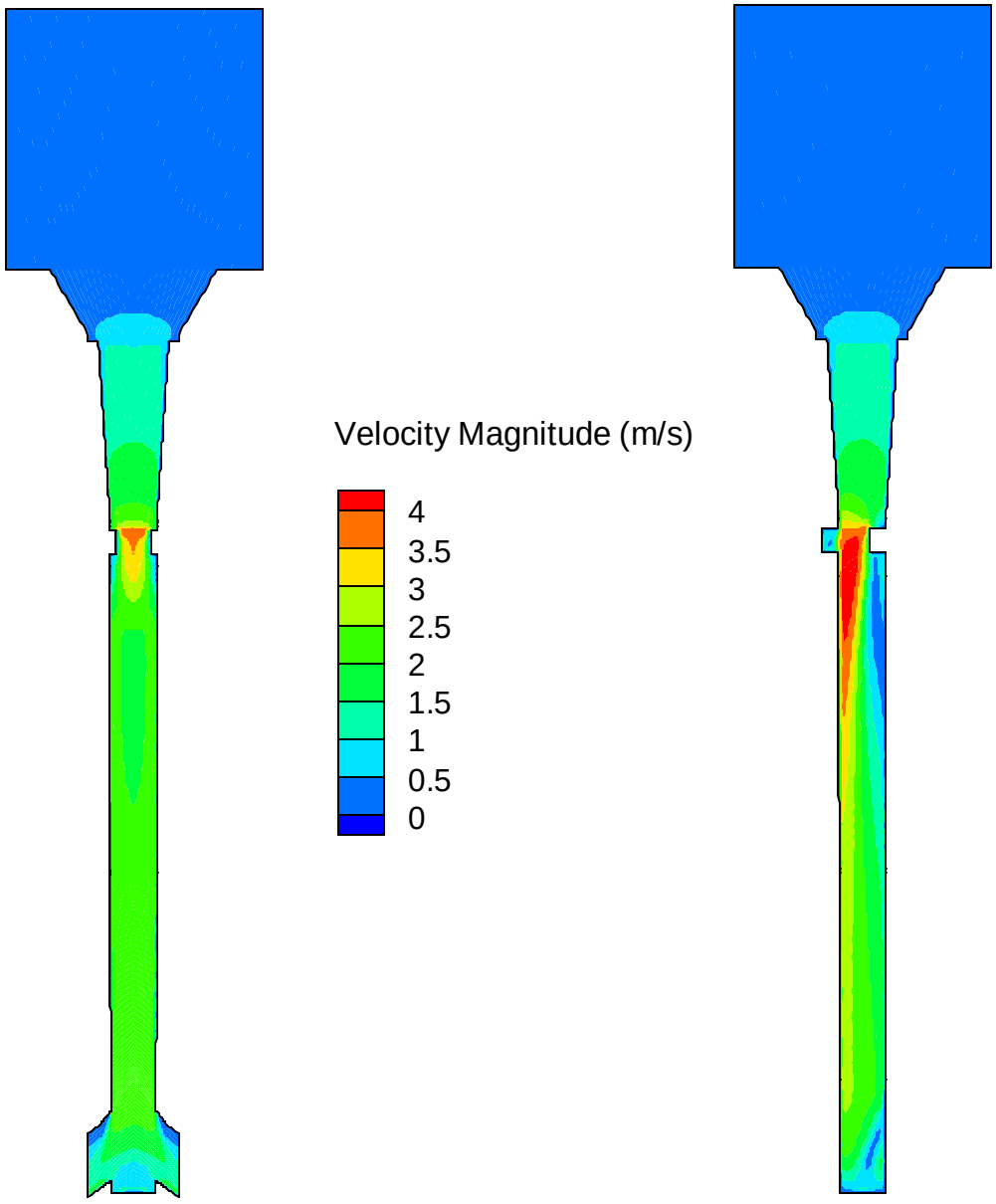


Fig 2.27. Velocity contours of mid plane front view and side view of the nozzle with slide gate .

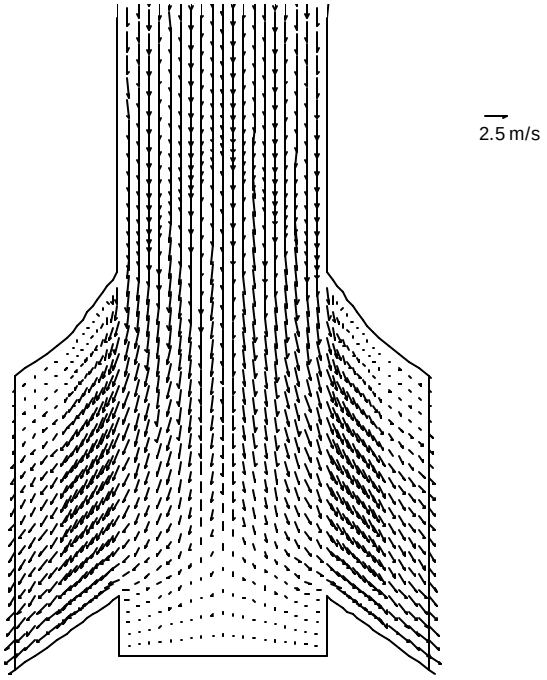


Fig 2.28. Velocity vectors at the nozzle bottom.

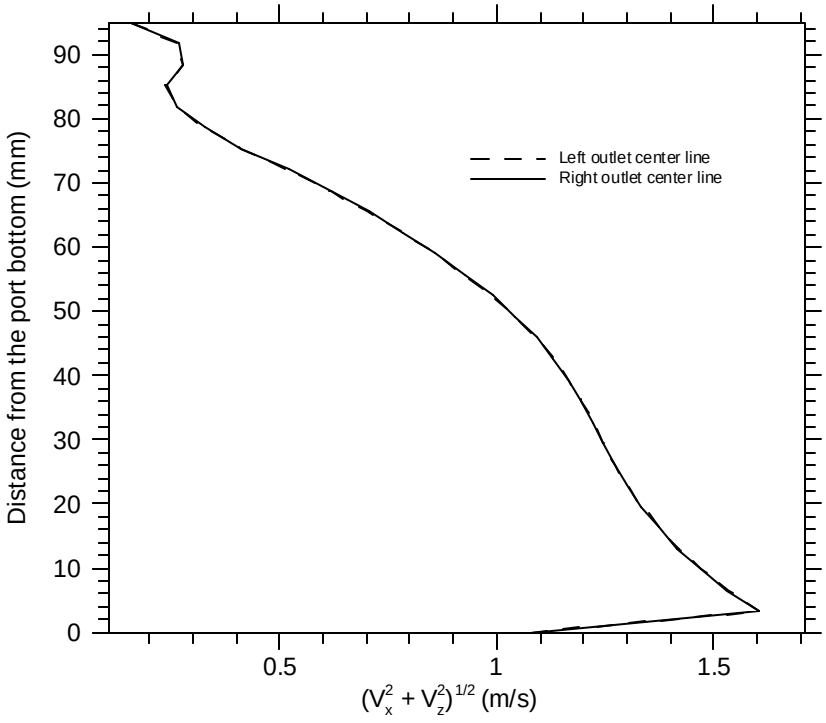
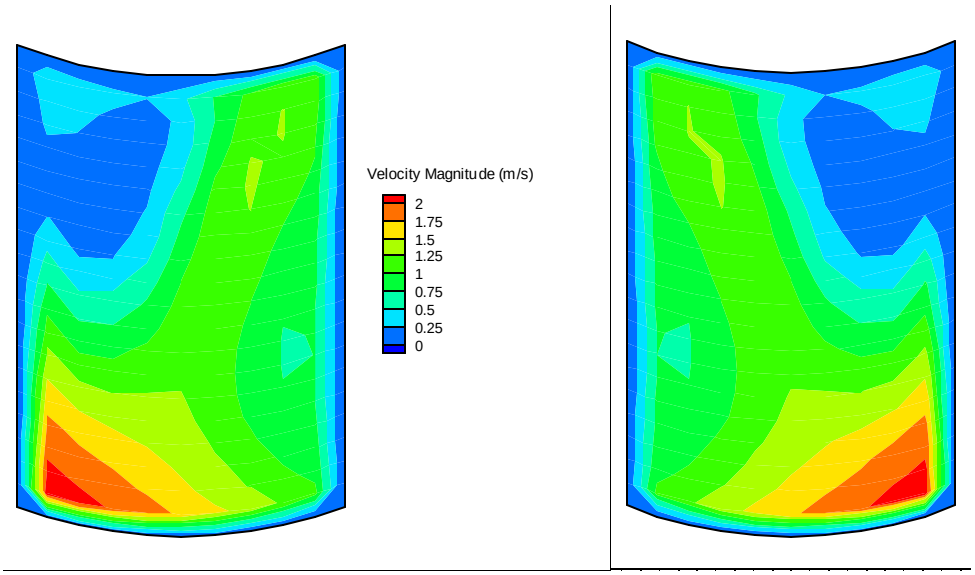


Fig 2.29. Velocity along nozzle port centerline on both sides.



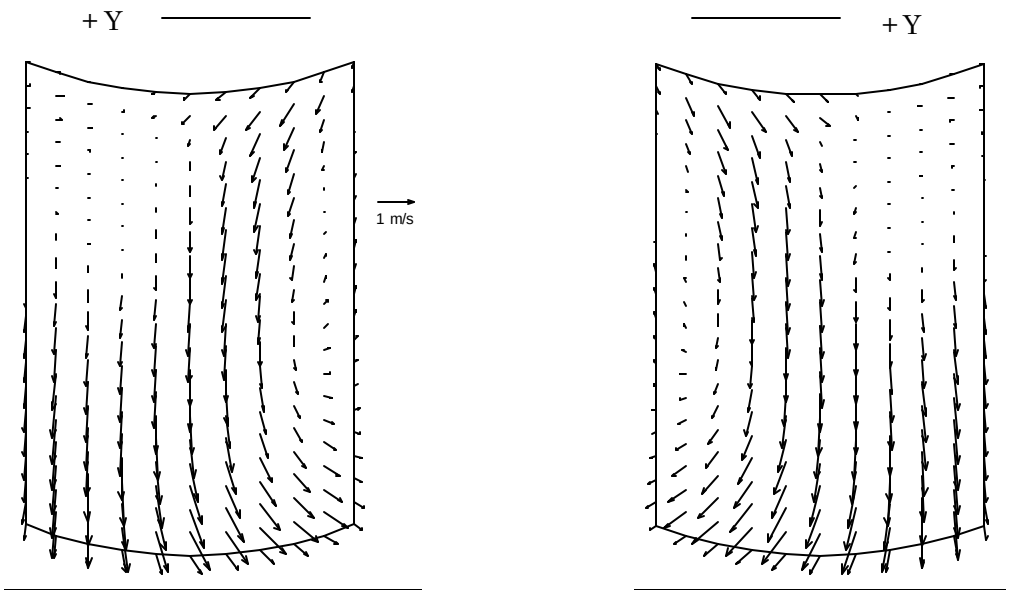


Fig 230. Velocity contours and velocity vectors at the outlet ports.

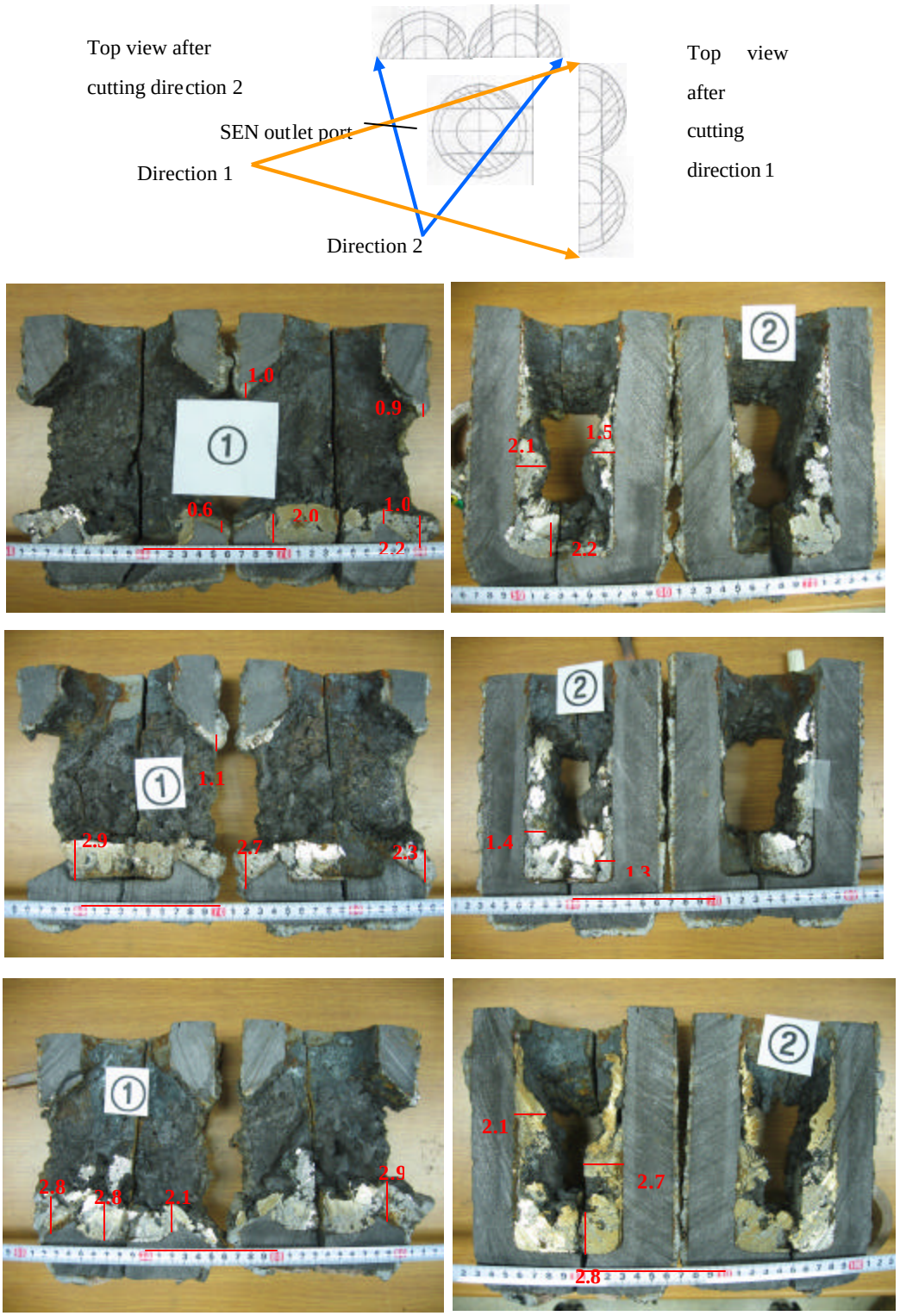
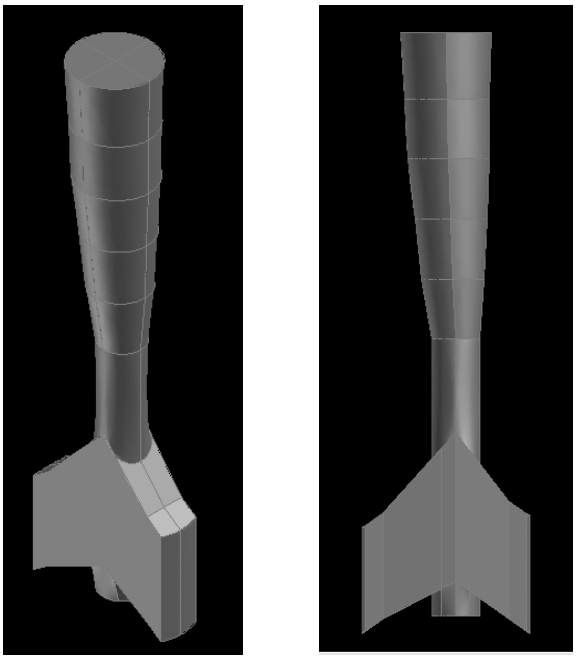


Fig 2.31. Different clog shapes as seen by cut sections of nozzles [16] with the dimensions on the figure listed in cm.



Isometric view **Front View**

Fig 2.32. Clog 1 at the nozzle bottom.

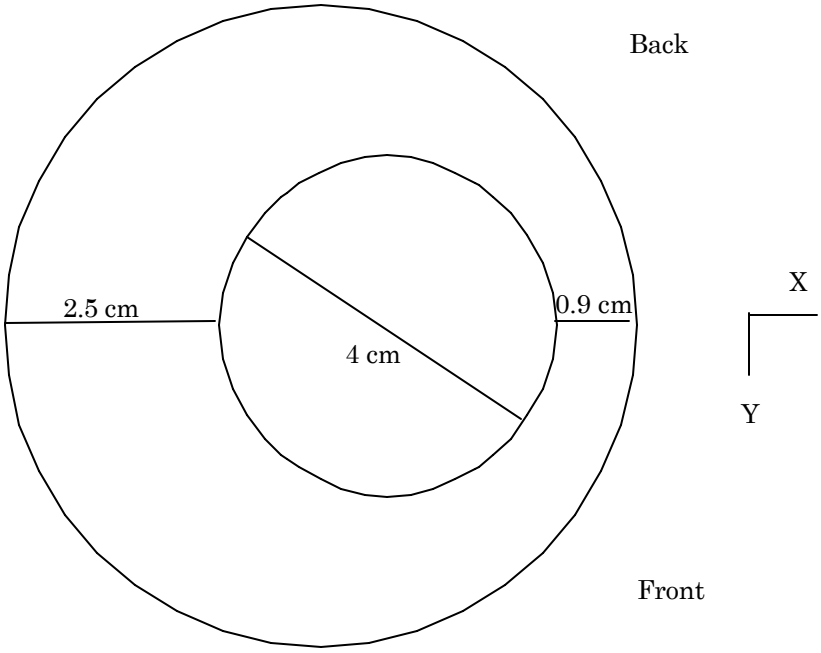
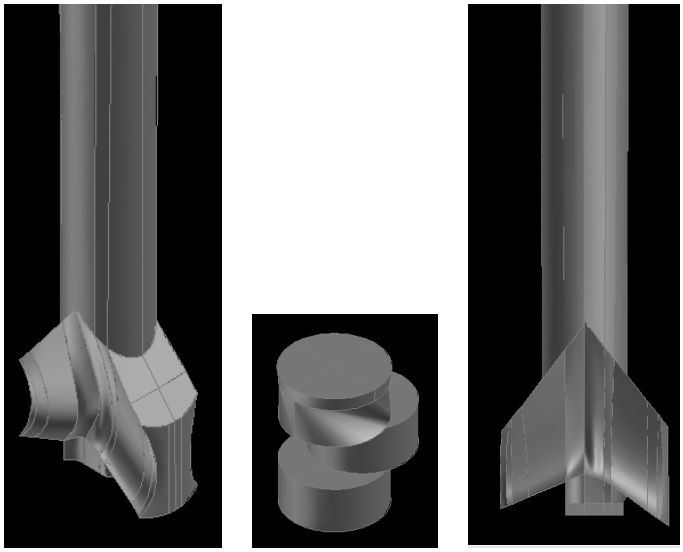
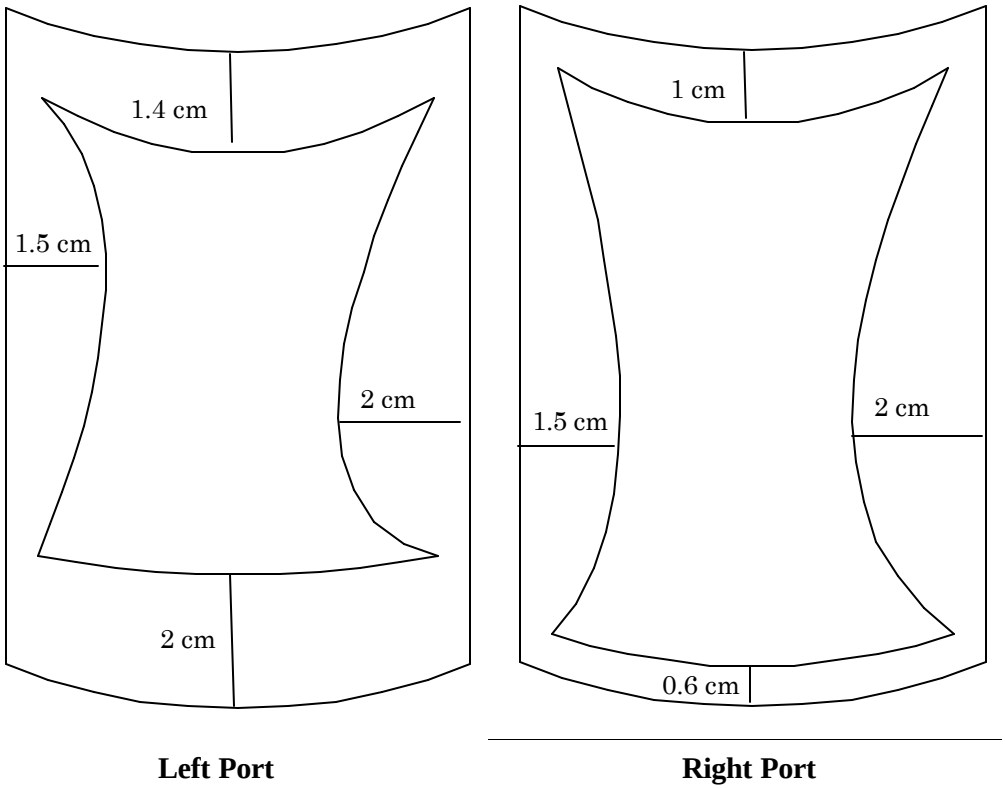


Fig 2.33. Asymmetrical reduction in bore diameter because of clogging (Clog 1).



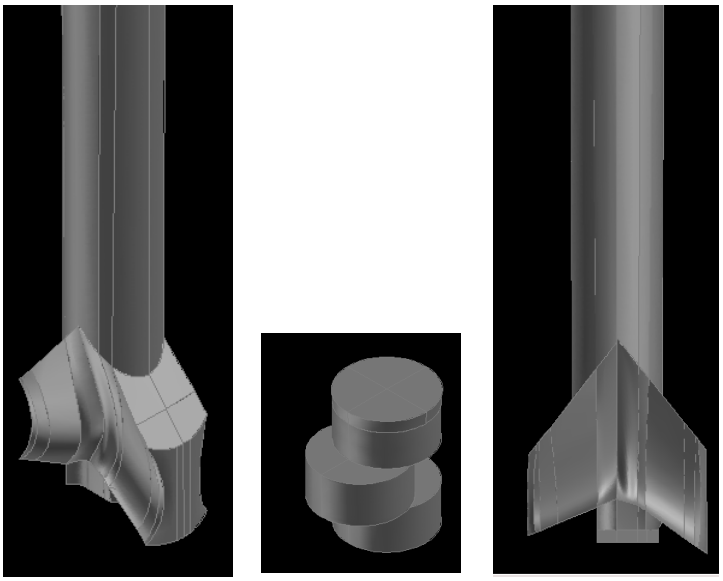
Isometric view **Front View**

Fig 2.34. Clog 2(a) at the nozzle bottom.



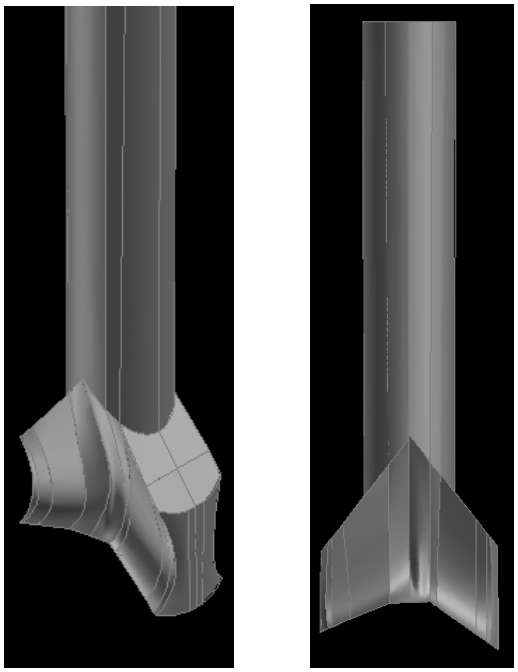
Left Port **Right Port**

Fig 2.35. Shape of outlet ports of Clog 2(a) with respect to original ports.



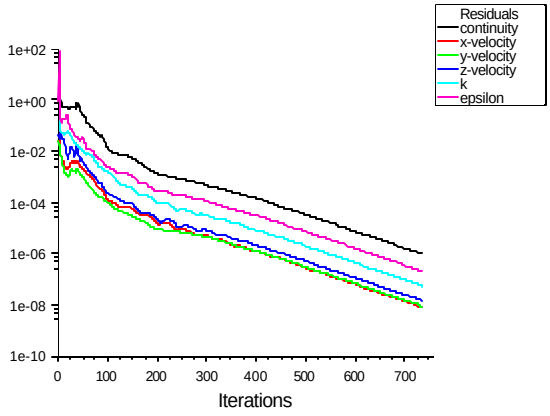
Isometric view **Front View**

Fig 2.36. Clog 2(b) at the nozzle bottom.

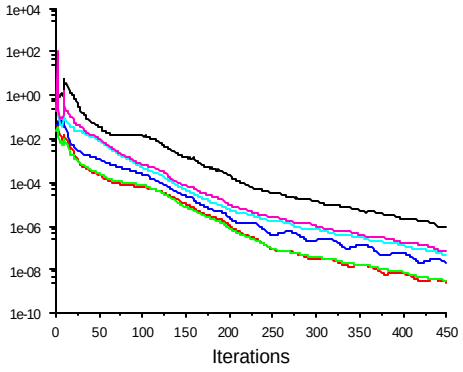


Isometric view **Front View**

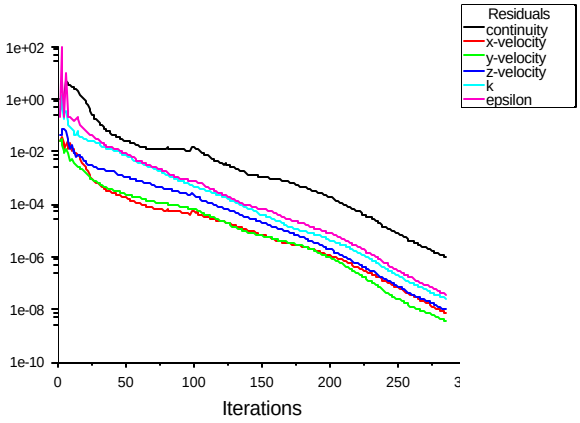
Fig 2.37. Clog 3 at the nozzle bottom.



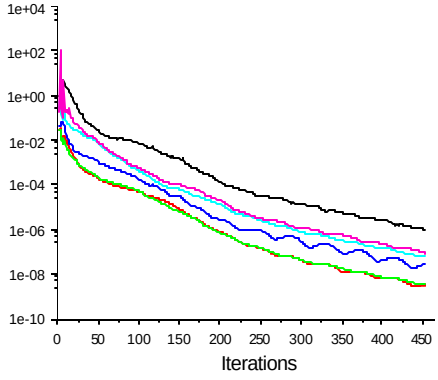
Clog 1



Clog 2(a)



Clog 2(b)



Clog 3

Fig 2.38. Scaled residual error for all different clog shapes.

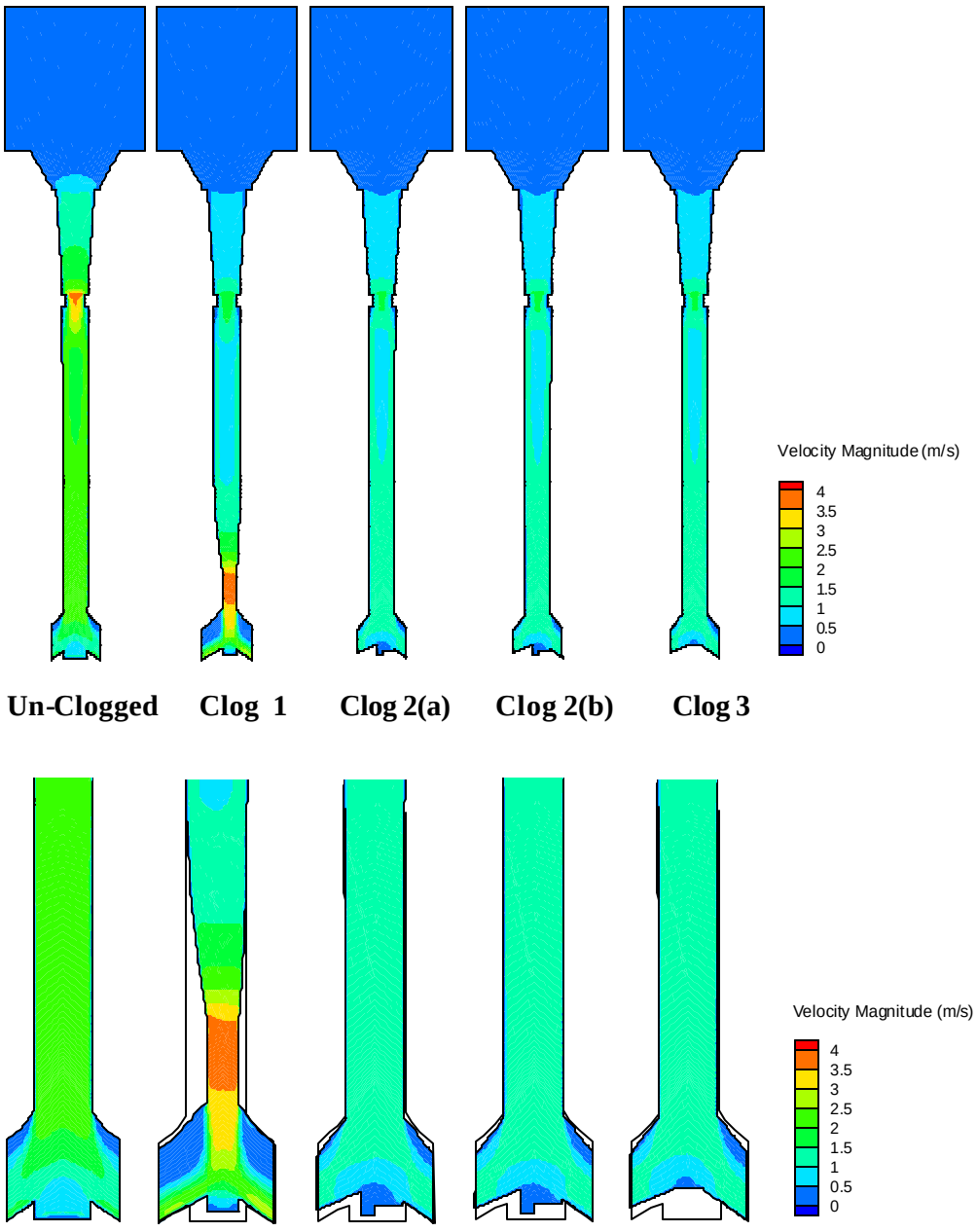


Fig 2.39. Velocity contours of mid x-z plane for all nozzle s.

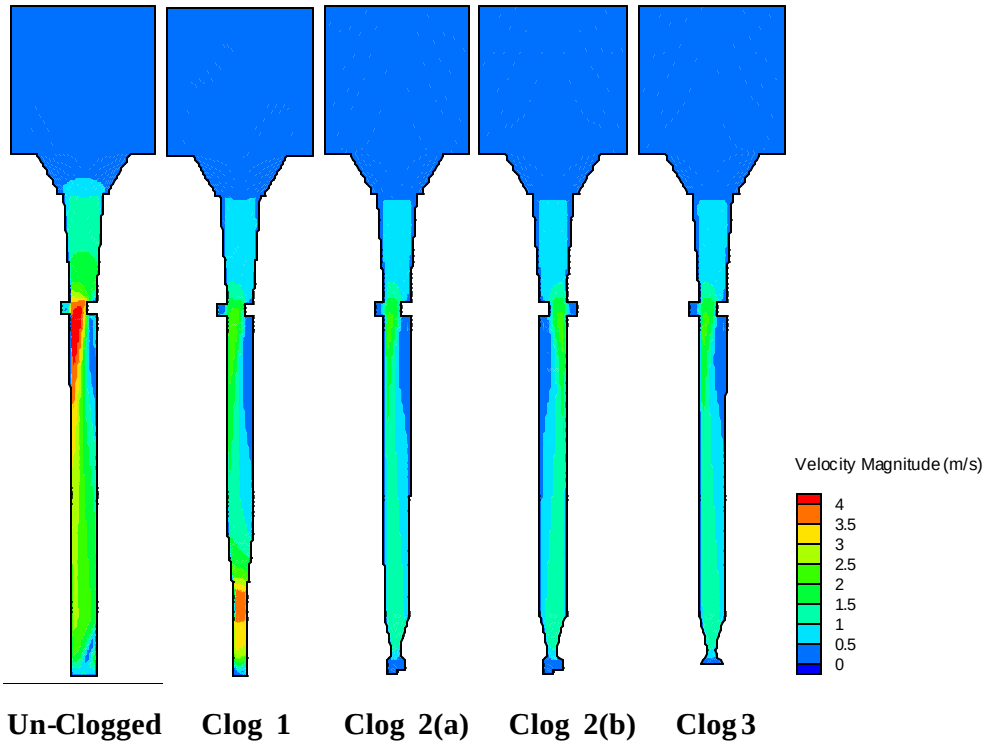
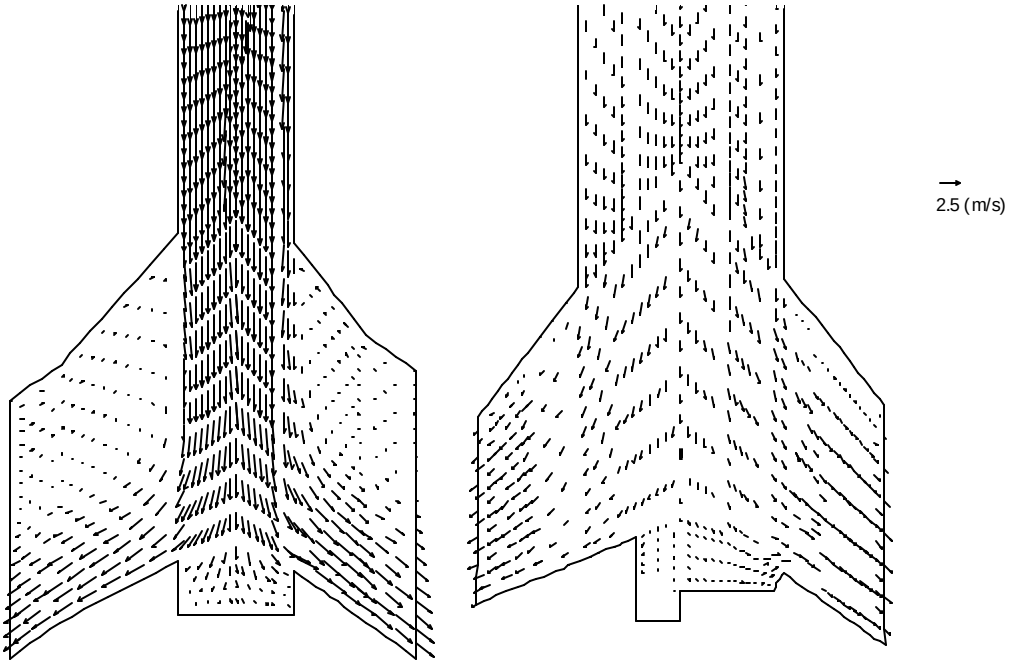
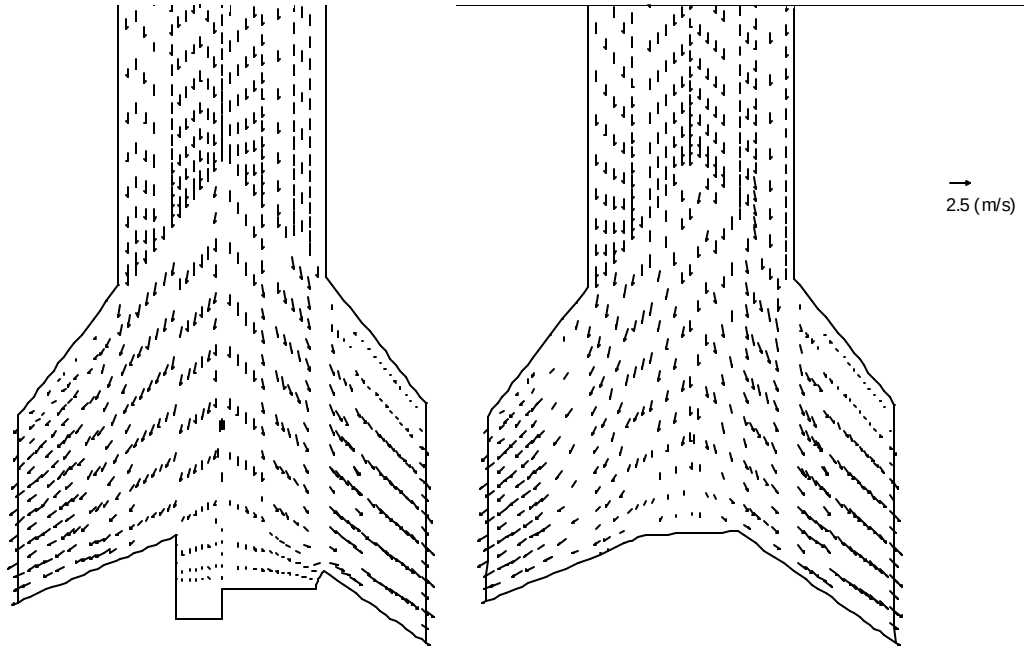


Fig 2.40. Velocity contours of mid x-z plane for all nozzles.

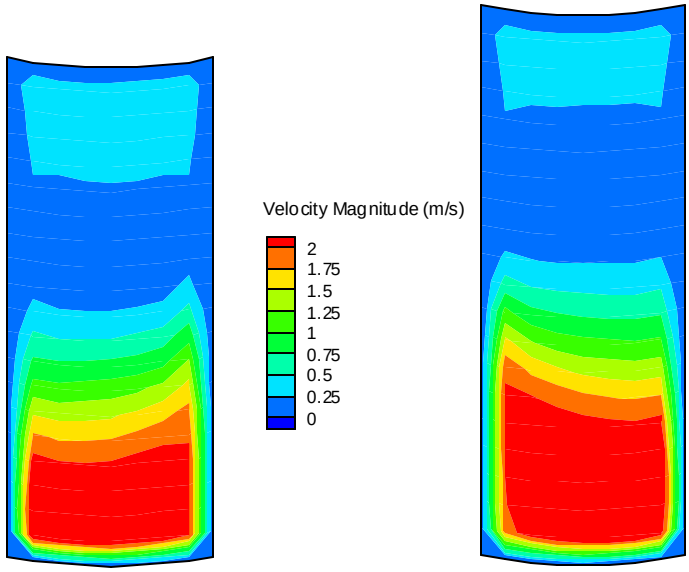


Clog 1 **Clog 2(a)**

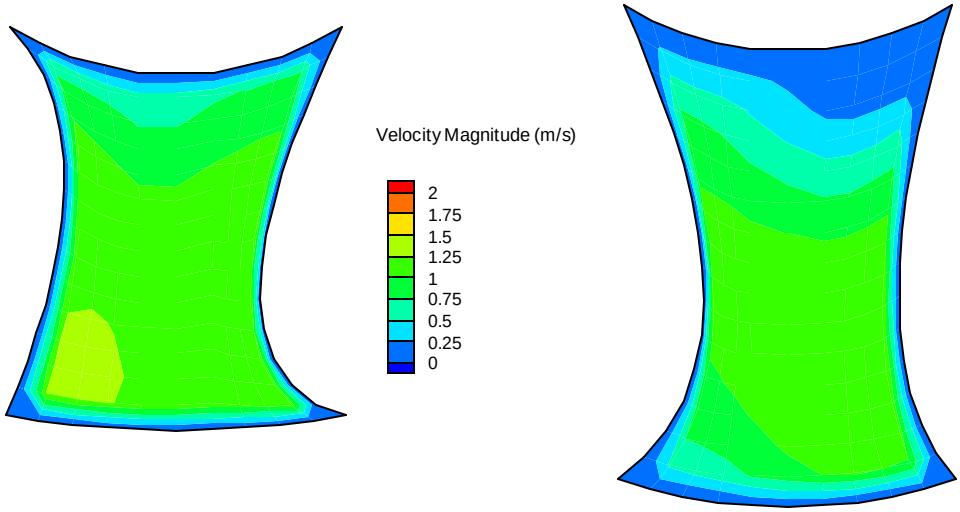


Clog 2(b) **Clog 3**

Fig 2.41. Velocity vectors at the nozzle bottom in mid x-z plane.

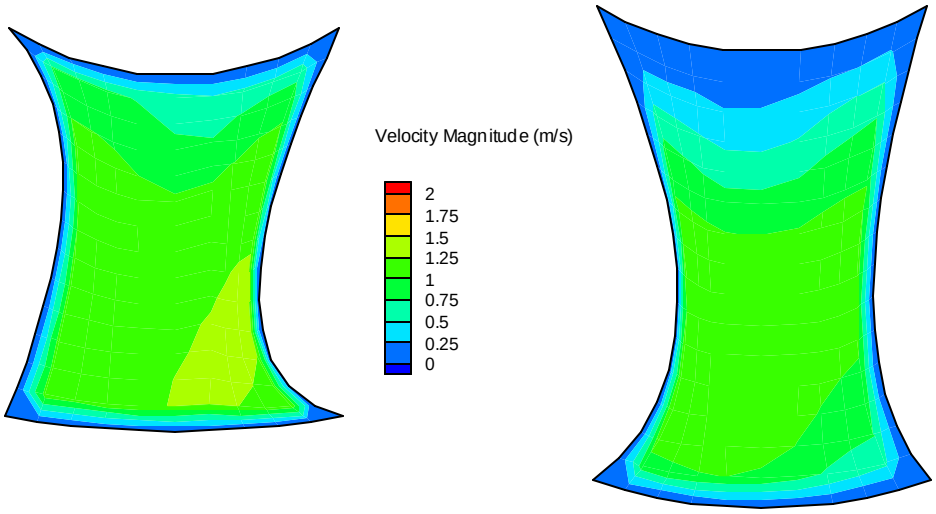


Clog 1, Left Port **Clog 1, Right Port**

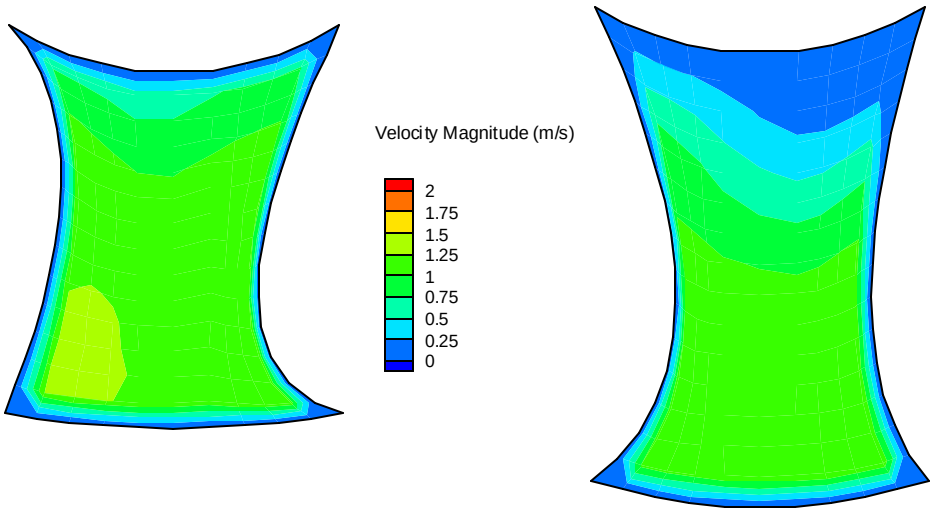


Clog 2(a), Left Port **Clog 2(a), Right Port**

Fig 2.42. Velocity contours at ports of clogged nozzles.



Clog 2(b), Left Port **Clog 2(b), Right Port**



Clog 3, Left Port **Clog 3, Right Port**

Fig 2.43. Velocity contours at ports of clogged nozzles.

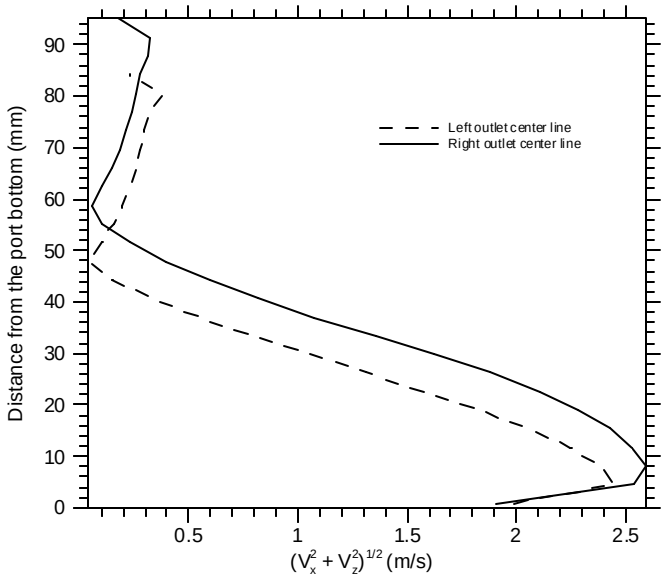


Fig 2.44. Velocity along nozzle port centerline on both sides for clog 1.

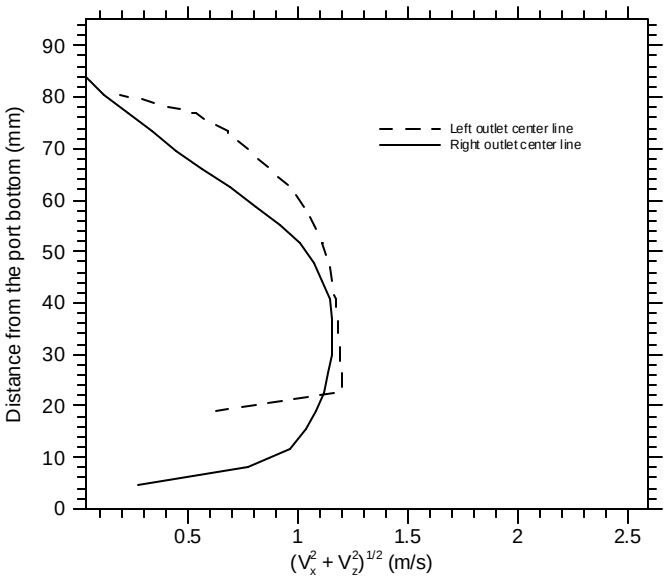


Fig 2.45. Velocity along nozzle port centerline on both sides for clog 2(a).

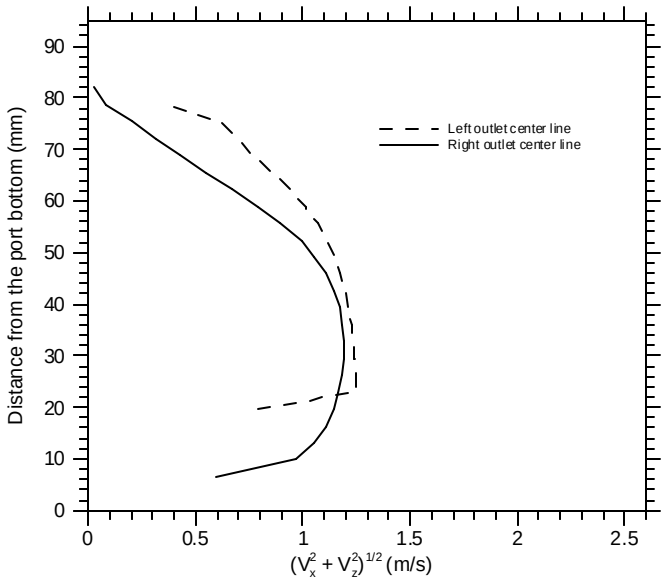


Fig 2.46. Velocity along nozzle port centerline on both sides for clog 2(b).

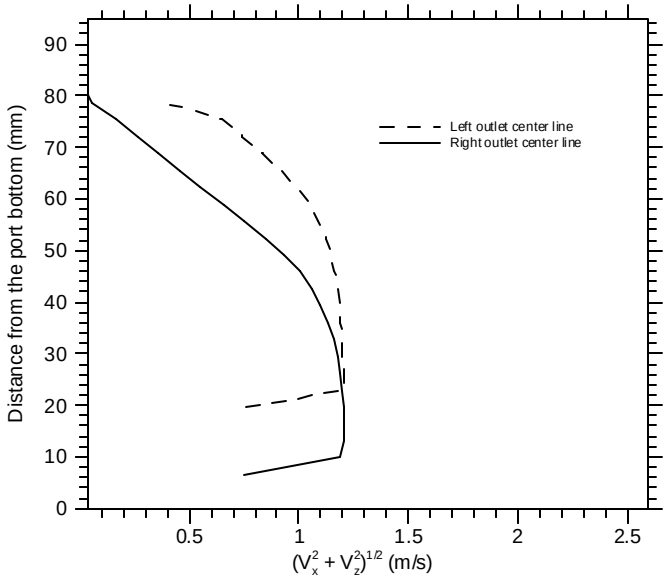


Fig 2.47. Velocity along nozzle port centerline on both sides for clog 3.

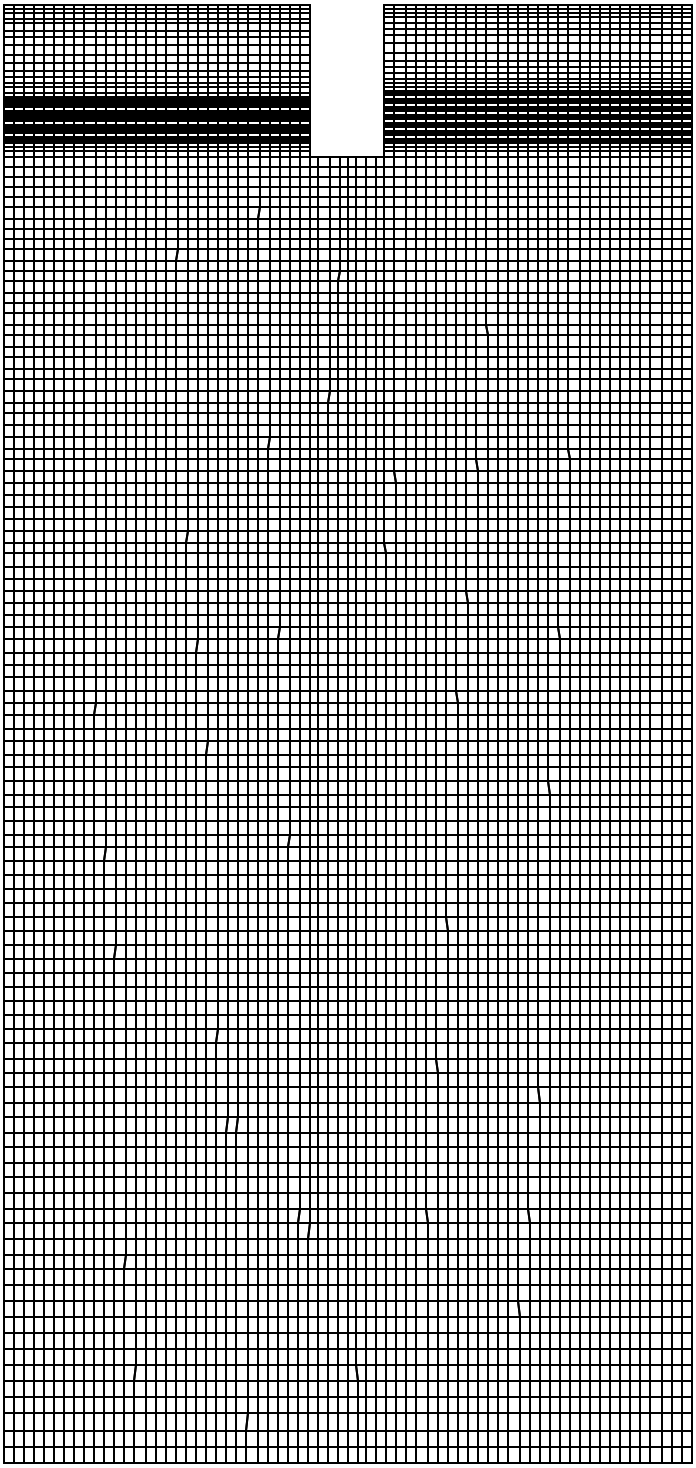


Fig 2.48. Mold mesh (with no shell incorporated).

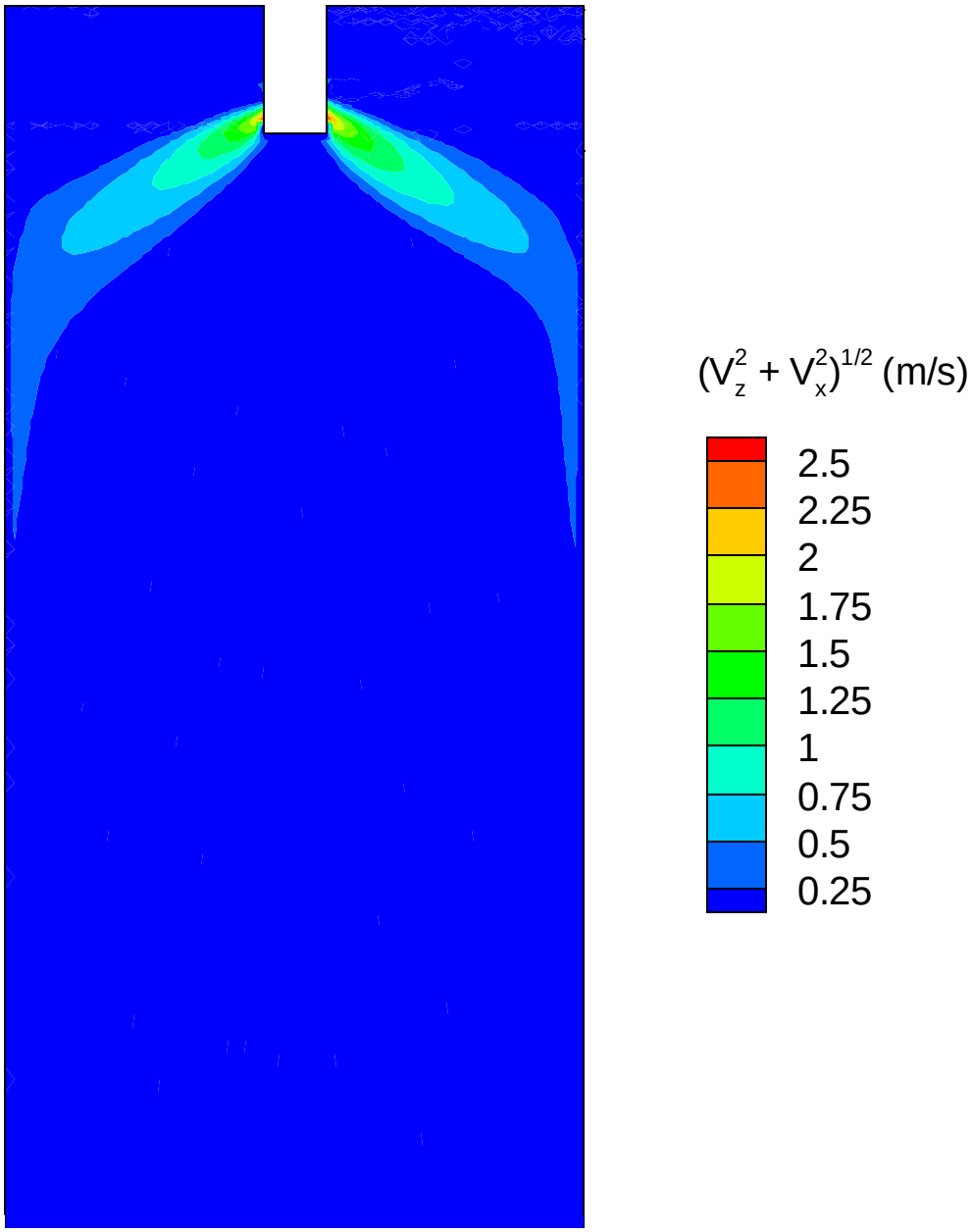


Fig 2.49. Velocity contours on plane mid way between wide faces.

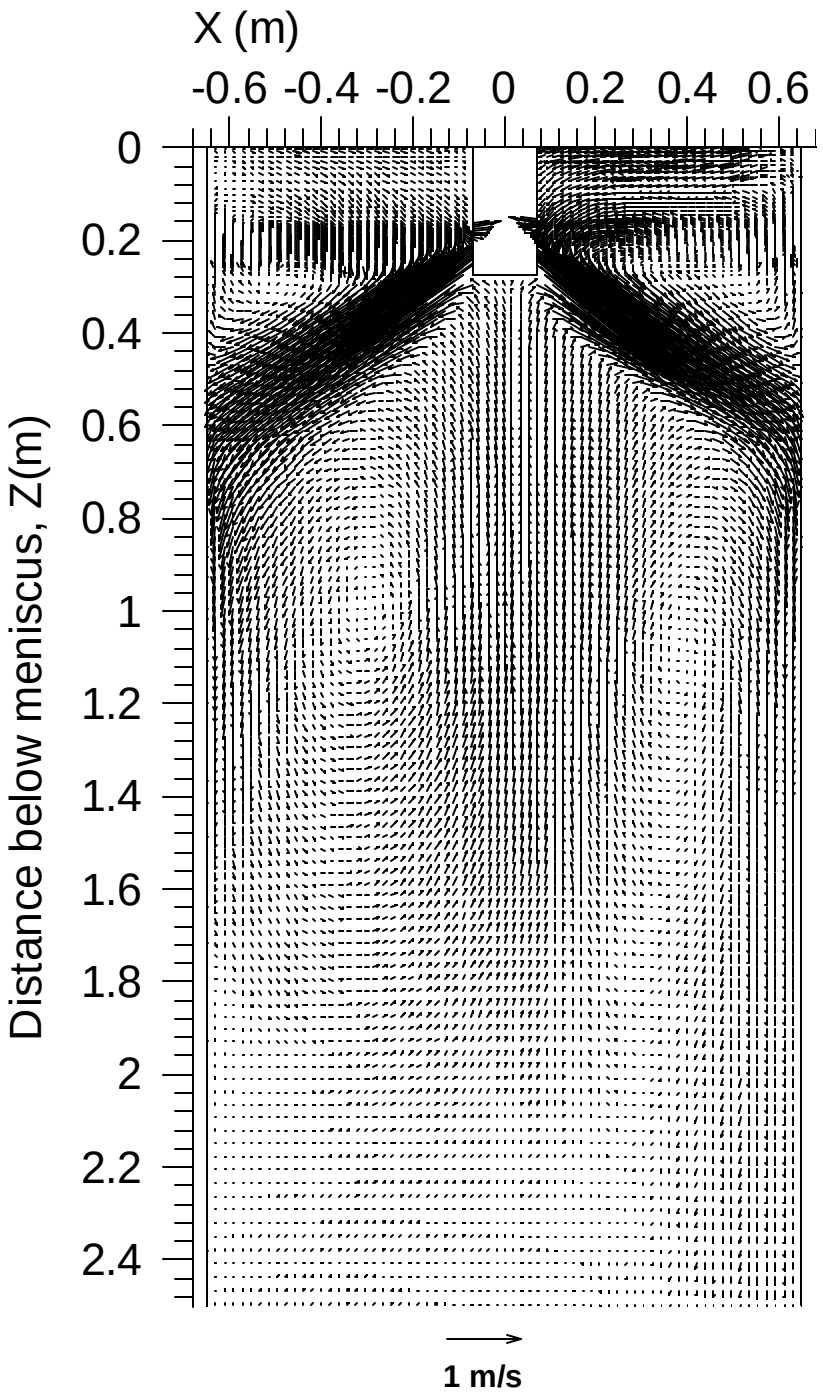


Fig 2.50. Velocity vectors on plane mid way between wide faces.

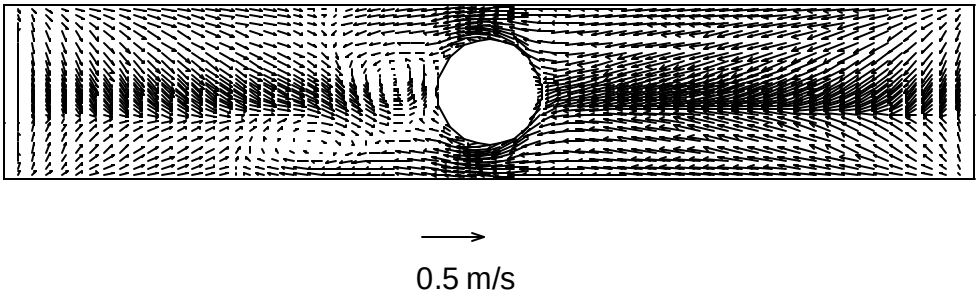


Fig 2.51. Velocity vectors at the top surface of the mold.

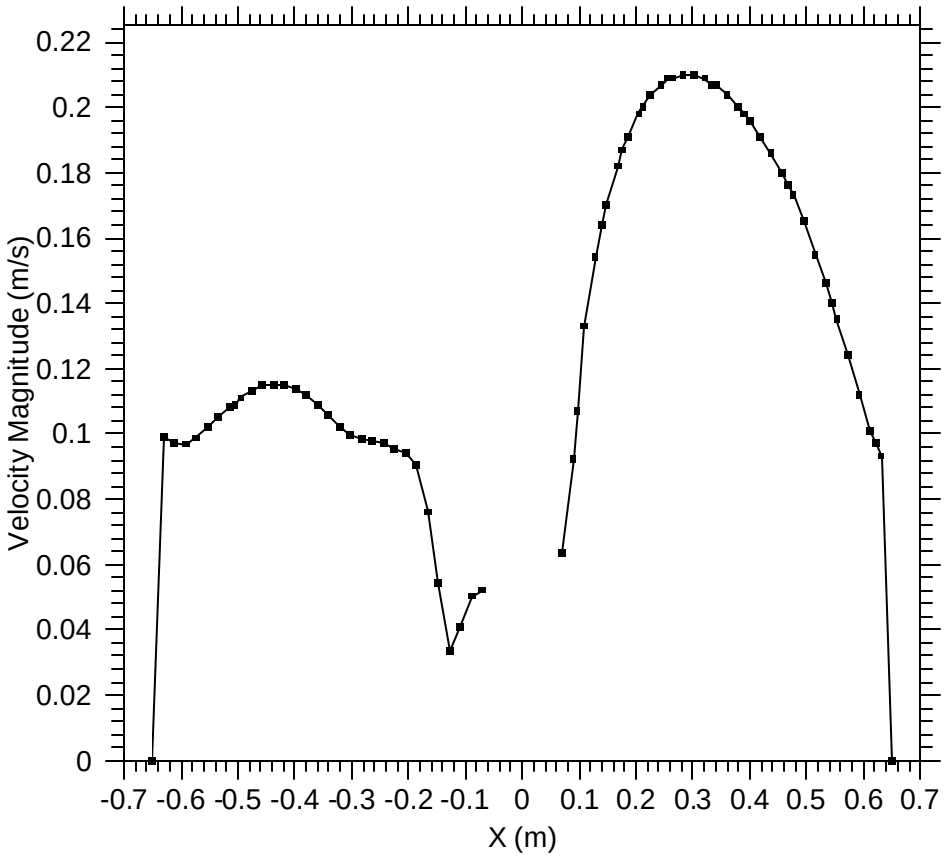


Fig 2.52. Velocity magnitude along center horizontal line at the top surface.

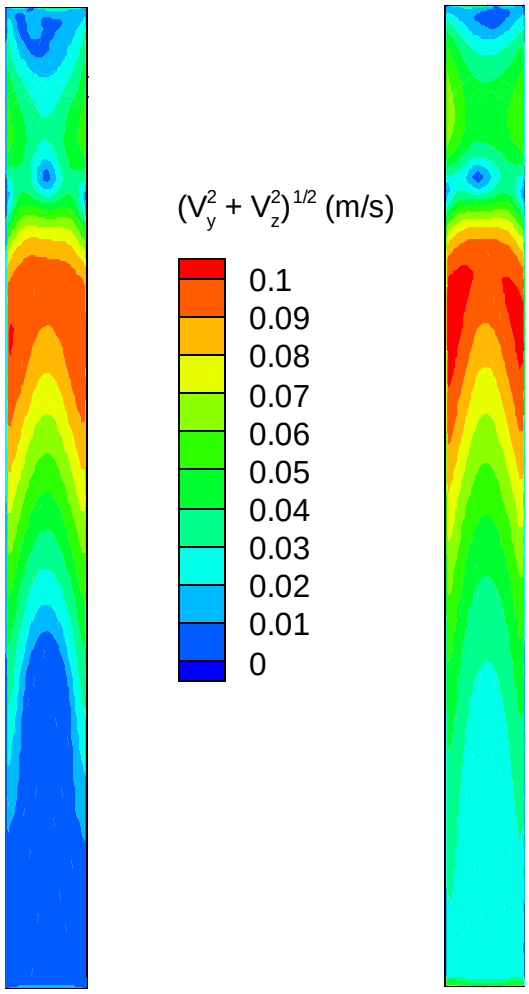


Fig 2.53. Velocity magnitude on planes 5mm from narrow faces.

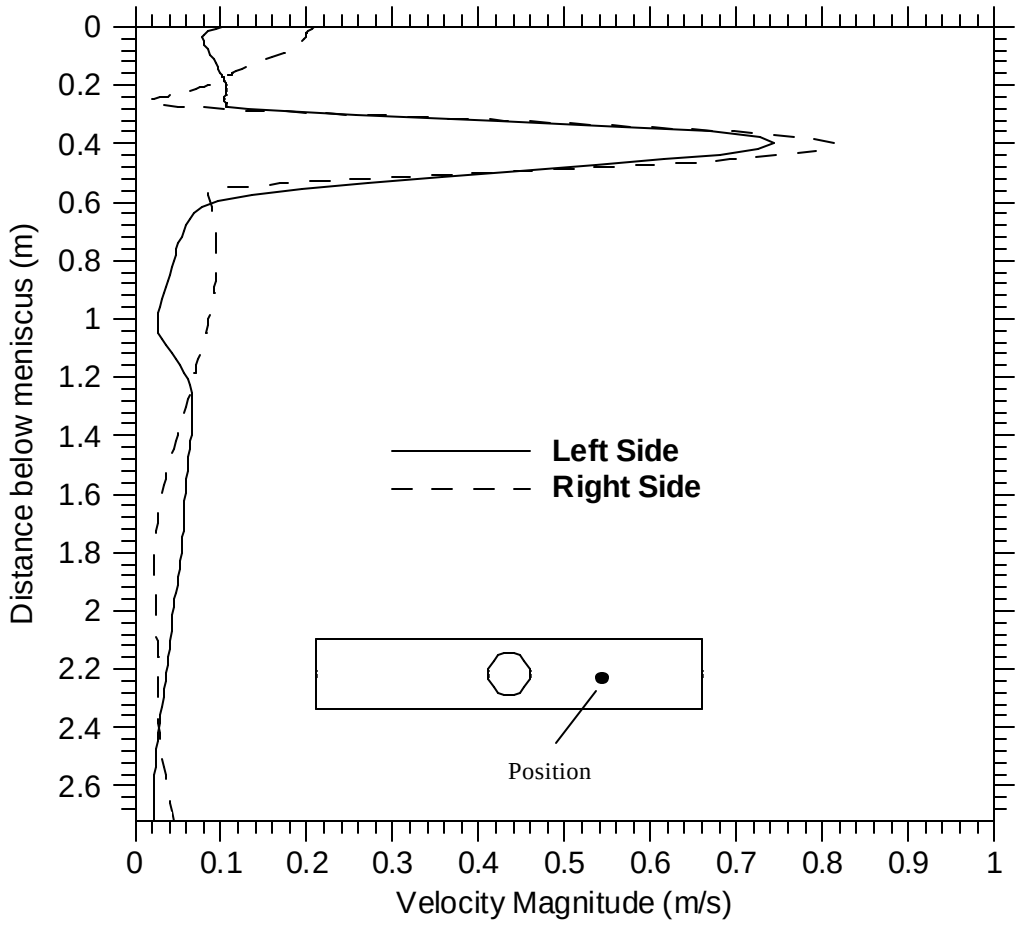


Fig 2.54. Velocity variation down the mold length at the position shown on both left and right side of the mold.

2.11 REFERENCES

1. Bai, H. and B.G. Thomas, *Effect of Clogging, Argon Injection, and Casting Conditions on Flow Rate and Air Aspiration in Submerged Entry Nozzles Steel,* in *83rd Steelmaking Conf. Proc.* 2000, ISS, Warrendale, PA: Pittsburgh, PA, March 2-29, 2000. p. 183-197.

2. Bai, H. and B.G. Thomas, *Turbulent Flow of Liquid Steel and Argon Bubbles in Slide-Gate Tundish Nozzles, Part I, Model Development and Validation.* Metall. Mater. Trans. B, 2001. **32B**(2): p. 253-267.

3. Najjar, F.M., B.G. Thomas, and D.E. Hershey, *Numerical study of steady turbulent flow through bifurcated nozzles in continuous casting*. Metall. Mater. Trans. B (USA), 1995. **26B**(4): p. 749-765.

4. Thomas, B.G., L.J. Mika, and F.M. Najjar, *Simulation of Fluid Flow Inside a Continuous Slab Casting Machine*. Metall. Trans. B, 1990. **21B**(2): p. 387-400.

5. Yokoya, S., et al., *Swirling Flow Effect in Immersion Nozzle on Flow in Slab Continuous Casting Mold.* ISIJ Int., 2000. **40**(6): p. 578-583.

6. Yokoya, S., et al., *Swirling Flow Effect in Off-center Immersion Nozzle on Bulk Flow in Billet Continuous Casting Mold*. ISIJ Int., 2001. **41**(10): p. 1215-1220.

7. Yuan, Q., B.G. Thomas, and S.P. Vanka, *Study of Transient Flow and Particle Transport during Continuous Casting of Steel Slabs, Part 1. Fluid Flow*. Metal. & Material Trans. B., 2004. **35B**(4): p. 685-702.

8. Noriko Kubo, J.K.a.T.I., *Simulation of Sliding Nozzle Gate Movements for Steel Continuous Casting* ISIJ International, 2001. **41**(10): p. 1221-1228.

9. Bai, H. and B.G. Thomas, *Turbulent Flow of Liquid Steel and Argon Bubbles in Slide-Gate Tundish Nozzles, Part II, Effect of Operation Conditions and Nozzle Design*. Metall. Mater. Trans. B, 2001. **32B**(2): p. 269-284.

10. Emling, W.H., et al. *Subsurface Mold Slag Entrainment in Ultra Low Carbon Steels.* in *77th Steelmaking Conference Proceedings*. 1994: ISS, Warrendale, PA.

11. Herbertson, J., et al., *Modelling of Metal Delivery to Continuous Casting Moulds*, in *Steelmaking Conf. Proc.* 1991, ISS, Warrendale, PA: Washington, D.C. p. 171-185.

12. Bai, H. and B.G. Thomas, *Effects of Clogging, Argon Injection and Continuous Casting Conditions on Flow and Air Aspiration in Submerged Entry Nozzles.* Metall. Mater. Trans. B, 2001. **32B**(4): p. 707-722.

13. Rackers, K.G. and B.G. Thomas, *Clogging in Continuous Casting Nozzles*, in *Steelmaking Conf. Proc.* 1995, ISS, Warrendale, PA: Nashville, TN. p. 723-734.

14. Thomas, B.G. and H. Bai, *Tundish Nozzle Clogging – Application of Computational Models*, in *Steelmaking Conf. Proc.* 2001, Iron and Steel Society, Warrendale, PA: Baltimore, MD. p. 895-912.

15. S.-H. Kim, B.G.T., S. Mahmood, H.-J. Shin, and G.-G. Lee, *"Effect of Porosity of Refractories and Af Flow Rate on Ar Bubble Sizes in the Mold", POSCO Technical Report*, in *POSCO: Pohang, Kyungbuk, South Korea*. 2005, POSCO: Pohang, Kyungbuk, South Korea.

16. Lee, G., *Private Communication*. 2005: Urbana-Champaign, IL.

17. Mizukami, K.S.a.Y., *Effect of Calcium Treatment of Molten Steel on Alumina Buildup on Submerged Entry Nozzles*. Journal of the Technical Association of Refactories, Japan, 2003. **23**(3): p. 156-163.

18. Yuan, Q., et al., *Computational and Experimental Study of Turbulent Flow in a 0.4-Scale Water Model of a Continuous Steel Caster.* Metall. & Mater. Trans., 2004. **35B**(5): p. 967-982.

19. Hershey, D.E., B.G. Thomas, and F.M. Najjar, *Turbulent Flow through Bifurcated Nozzles.* Int. J. Num. Meth. in Fluids, 1993. **17**(1): p. 23-47.

20. Launder, B.E. and D.B. Spalding, *Mathematical Models of Turbulence*. 1972: London Academic Press.

21. Creech, D., *Computational Modeling of Multiphase Turbulent Fluid Flow and Heat Transfer in the Continuous Slab Casting Mold*, in *Mechanical and Industrial Engineering*. 1999, University of Illinois at Urbana-Champaign: Urbana, IL.

22. FLUENT6.1-Manual. 2003, Fluent Inc., Lebanon, New Hampshire.

23. Patankar, S.V., *Numerical Heat Transfer and Fluid Flow*. 1980, New York, NY: McGraw Hill.

24. Patankar, S.V. and B.D. Spalding, *A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows.* Int. J. Heat Mass Transfer, 1972. **15**(10): p. 1787-1806.

25. Zhang, L., *Private Communication*. 2005: Champaign, IL.

26. Lai, W., M. Milone, and I. Samarasekera, *Meniscus Thermal Analysis as a Tool for Evaluating Slab Surface Quality*, in *Steelmaking Conf. Proc.* 2000, ISS, Warrendale, PA: Pittsburgh, PA. p. 461-476.

27. Cramb, A.W., et al., *The slag/metal interface and associated phenomena. A history of pneumatic steelmaking.* Iron and Steelmaker (USA), 1997. **24**(3): p. 77-83.

CHAPTER 3. PARTICLE ENTRAPMENT IN NOZZLE AND MOLD

3.1 INTRODUCTION

Turbulent fluid flow and the velocity distribution of molten steel in the continuous casting mold affects the steel quality, as it influences many important phenomena. Detrimental phenomena include high surface velocities, which shear off mold flux droplets, a non-level top surface profile, which retards flux infiltration into the gap, mold surface level fluctuations, which disrupts meniscus solidification, and particle transport [3]. Particle transport leads to inclusion entrapment into the solidifying shell and sliver defects in the final product, so is of great practical interest. Captured inclusions cause defects within the final steel product and thus degrade the steel quality. Zhang and Thomas [4] reviewed methods to lower the capture of impurity particles in steel.

Due to the difficulties of performing quantitative experiments in liquid steel, it is appropriate to use computational modeling to investigate particle behavior in continuous casting of steel. Much previous work has been done on modeling fluid flow in continuous casting of steel [5]. These models have been verified with experimental work performed on water models using PIV method [6]. Fluid particle flows have been simulated earlier with both Eulerian and Langrangian approaches [7-9]. However, much less work has been for steel continuous casting process [10-12]. Thomas in [13] has modeled transport of argon bubbles in the mold. Previous models assume that particles

touch in a wall are either captured or reflected. However, recent work by Yuan [2] has shown that particle capture varies greatly between these two extremes, depending on particle size and many other parameters.

Yuan [2] developed a model to quantitatively predict the particle removal rates. There are several forces that act on a particle when the particle approaches the solidification front in continuous casting of steel. Based on the magnitude of the forces that are present on the particle in this position, the particles fate can be determined. The particle can be captured, pushed away or rolled away from the dendrites. Particles with diameter smaller than the primary dendrite arm spacing are assumed to be instantly captured once they get between the arms (touch the boundary in the computational domain). For larger particles, a force balance analysis is done, to see if the particle can roll upward or downward about the dendrite tips, to escape. Yuan implemented this model of particle capture into an LES model to compute the transient flow and particle transport [14] and reasonably matched PIV measurements on 0.4 scale water models.

Large Eddy Simulation (LES) requires modeling long term transients with a fine grid and small time-step size, in order capture the large eddies accurately in a highly turbulent flow field. This requires large processing time and large computer memory storage, so very few cases have been examined. This work aims to incorporate this particle capture model into a Reynold's Average Navier Stokes (RANS) model, validate it by comparison with the previous LES results and apply it to investigate the effect of various process variables on particle capture.

A $(k-\varepsilon)$ model, which predicts time averaged velocities with reasonable accuracy, is used in this thesis to model flow in the steel mold. In order to proceed with this and to see particle transport, mold flow validation needs to be done. The results obtained from $(k-\varepsilon)$ model will be compared with the averaged results from LES obtained in [14], to validate the flow pattern. The model introduced by Yuan [2] for particle transport and entrapment presented is applied to simulate particle motion and entrapment in a steel mold.

The magnitude of the forces applied during the transport of bubbles and inclusions, and at the point of capture has been investigated and compared with LES results obtained by Yuan [2]. Lagrangian motion, used in this thesis to model particle motion, of liquid-particle flows can be categorized as either one way coupling, where the flow affects the particle motion or as two way coupling, where particles also affect the flow. In this thesis only one way coupling has been applied as done in [2].

A theoretical study of how particle entrapment is affected by variation in several parameters has been performed, before the criterion is added into the simulation. Discrete phase model (DPM) is used in Fluent to model the transport of particles in the mold computational domain and nozzle computational domain. Extensive user defined functions were written to modify the hydrodynamic forces acting on particles, to ensure that they match with those used by Yuan [2]. Further more boundary conditions for particle entrapment by the mold walls have to be modified to incorporate the forces acting on the particle when close to the solidifying front. Some user defined functions were written to enhance the post processing and in order to visualize results in Tecplot.

The hydrodynamic forces that have been applied on the particle are explained along with the forces that are present only when the particle is in close proximity of the solidification front. All particles assumed in this thesis are spherical and considered as point masses. Motion and capture of both inclusions and bubbles are modeled.

3.2 MODEL FORMULATION FOR FLOW IN MOLD WITH SHELL

The model domain is the nozzle, liquid region in the continuous casting mold and upper strand below the mold as shown in Fig 3.4. The solidifying steel shell is outside the domain, so the boundaries represent the dendrite tips at the liquidus temperature. The boundary shape was obtained from shell thickness predictions using CON1D [15] and was used previously by Yuan for LES modeling [14]. The shell thickness defining the boundary shape can be seen in Fig 3.5. Time averaged calculations are performed and only half of the mold is modeled, owing to symmetry about the z axis. 170,000 hexahedral cells were used in the mesh to ensure a well-shaped, structured mesh with refined cells at the model inlet (outlet ports of the nozzle) that has more refinement than needed in the central region, as can be seen in Fig 3.6. Achieving well-shaped cells was found to be essential in order to achieve reasonable convergence, however, so this level of refinement could not be avoided.

3.2.1 Governing Equations

The continuity equation solved is:

$$\nabla \bullet (\overrightarrow{\rho V}) = S_m \tag{3.1}$$

Where ρ is the density of steel, V is the velocity and S_m is the source term needed to model the solidifying shell.

$$S_m = -\frac{\rho_f V_c |A[1]|}{C\_VOLUME(c,t)} \tag{3.2}$$

Where,

ρ_f is the density of steel (kg/m^3)

V_c is the casting speed (m/s)

C_VOLUME(c,t) is the boundary cell volume (m^3)

A[1] is the of the downward (z) component of the negative boundary face area vector (m^2) defined as:

$$A[1] = -\vec{A} \bullet \frac{(\overrightarrow{V_c})}{|\overrightarrow{V_c}|} = \left|\vec{A}\right|\left|\hat{\overrightarrow{V_c}}\right| \cos\phi \tag{3.3}$$

where, $\vec{A}$ is the area vector of the boundary face of the cell from which the fluid is being extracted and $\hat{\overrightarrow{V_c}}$ is the unit vector in the direction of casting velocity, ϕ is the angle between the casting velocity and unit area vector as illustrated in Fig 3.5.

The momentum equation for incompressible flow is:

$$\nabla \bullet (\rho \vec{V} \vec{V}) = -\nabla p + \nabla \bullet \mu_{eff} (\nabla \vec{V}) + \rho \vec{g} \tag{3.4}$$

where p is the static pressure, $\rho \vec{g}$ is the gravitational force, μ_{eff} is defined as

$$\mu_{eff} = \mu_o + \mu_t \tag{3.5}$$

where, μ_o is the molecular viscosity, and μ_t is the turbulent viscosity.

For the momentum equation the source term is defined as follows:

$$S_m = -\frac{\rho_f V_c |A[1]|}{C\_VOLUME(c,t)} * V_i \tag{3.6}$$

Where V_i is the velocity in the cell in x, y or z direction and i = 1, 2 or 3 represents the 3 spatial coordinate directions (x, y, or z).

In the Reynolds average approach, the variables in Navier-Stokes equation are decomposed into mean and fluctuating components. For example for velocity components

$$u_i = \overline{u_i} + u_i' \tag{3.7}$$

Where, $\overline{u_i}$ and u_i' are the mean and fluctuating components with i = 1,2,3 representing the 3 coordinate directions.

Similar expressions can be obtained for other variables and can then be substituted into the mass and momentum equations. Additional terms representing the effects of turbulence arise which need to be modeled for closure of the equations. The standard $(k-\varepsilon)$ model developed by Launder and Spalding is used for this purpose [16]. Two extra equations for turbulent kinetic energy (K) and turbulent dissipation rate (ε) are solved. μ_t is calculated as a function of these two quantities as:

$$\mu_t = \rho C_\mu \frac{K^2}{\varepsilon} \tag{3.8}$$

The equations for turbulent kinetic energy (K) and turbulent dissipation rate (ε) are given as:

$$\nabla \bullet (\rho K V) = \nabla \bullet ((\mu_o + \frac{\mu_t}{\sigma_k})\nabla K) + G_k - \rho\varepsilon \tag{3.9}$$

$$\nabla \bullet (\rho \varepsilon V) = \nabla \bullet ((\mu_o + \frac{\mu_t}{\sigma_\varepsilon})\nabla \varepsilon) + C_1 \frac{\varepsilon}{K} G_k + C_2 \rho \frac{\varepsilon^2}{K} \quad (3.10)$$

Where, G_k is represents generation of kinetic energy due to mean velocity gradients calculated as:

$$G_k = -\rho \overline{u_i' u_j'} \frac{\partial u_j}{\partial x_i} \quad (3.11)$$

The empirical constants are given by [16] :

$C_\mu = 0.09,\ C_1 = 1.44,\ C_2 = 1.92,\ \sigma_k = 1.0,\ \sigma_\varepsilon = 1.3,$

As the shell solidifies, mass and momentum are extracted across the domain boundaries. This extraction decreases with distance down the mold. This extraction is modeled, by adding negative source terms (S_m) in the mass and momentum equations for cells adjacent to the boundary. These cells were declared as a different fluid continuum in Gambit in order to apply the user defined function in Fluent only for these cells. The User Defined Function (UDF) to impose these conditions in FLUENT is given in Appendix B.3.

3.2.2 Boundary Conditions

Nozzle Inlet

Across the inlet plane at the top of the nozzle, the inlet velocity value is set, based on the mass flow required to achieve the desired casting speed as follows:

$$V_{avg} = \frac{A_{mold}}{A_{inlet}} * V_{cast}$$

Boundary values for the turbulent kinetic energy, k, and its Dissipation rate, ε, also must be specified. The best inlet condition is to extend the domain upstream to model a

portion of the bottom of the tundish, so that appropriate values evolve at the top of the nozzle. To simulate geometries with no tundish region, previous values have been calculated from a mixing length model for turbulent pipe flow [17]. As the flow near the bottom of the tundish region is fairly slow relative to the nozzle inlet, its turbulent intensity is less, so small inlet values of ~ 10^{-5} are appropriate to be used for the turbulent kinetic energy and turbulent dissipation rate respectively.

Nozzle outlet / Mold Inlet

At the inlet of the mold, velocity components, turbulent kinetic energy and turbulent dissipation rate are obtained from the outlet of the nozzle. To conveniently map the outlet values of the nozzle to the inlet of the mold, the mesh and spatial coordinates of the nozzle outlet plane was chosen to match exactly with that of the mold inlet. Just half of the third (bottom) nozzle outlet is used.

Mold Outlet

A constant pressure condition is used at the outlet boundary, which represents a transverse plane through the strand that is deep enough below the mold that the fluid moves downward without recirculation.

Walls

A no slip boundary condition is specified and standard wall boundary functions are used [18] for both nozzle and mold. The boundary wall of the mold specifying the solidification front moves downwards at the casting speed.

Mold Top Surface

A free slip condition (zero shear stress) is imposed on the top surface, matching that of Yuan [14]. This neglects the slight effect of slag layer, which tends to slow flow across the top.

Symmetry Planes

Symmetry conditions are used at the mold center plane, as only half of the mold is modeled. Zero normal velocity and zero normal gradients of other variables used are imposed.

3.2.3 Solution Procedure

The geometry is meshed in Gambit, and the equations are solved with FLUENT. The commands to be used in Fluent, to achieve this are given in Appendix B.1. The solution procedure is given in Chapter 2, except that the mold mesh is more complicated, and the flow field is more complex, and so convergence was more difficult. Convergence depends on the under-relaxation factors for each of these equations.

$$\phi = \phi_{old} + \alpha \Delta \phi \tag{3.12}$$

Where, α is the under relaxation factor, ϕ is the variable value to be used for next iteration, ϕ_{old} is the old variable value, $\Delta\phi$ is the difference between the new variable value calculated and the old value of the variable. Small under relaxation factors, which use more of the old solution, avoid divergence, but need more iteration to reach convergence.

To make the residues continuously decrease and avoid divergence of the solution, the

under relaxation factors were changed as the solution progressed. Table 3.1 shows how the under relaxation factors for each variable were changed, after a prescribed number of iterations, in order to reduce the scaled residual error consistently, as shown in Fig 3.2.

3.3 TEST MOLD TO CHECK UDF AT BOUNDARIES FOR FLUID EXTRACTION

A test was performed to check the user defined function that imposes negative mass and momentum source terms for mass extraction pertaining to solidification. The test mold is a simple tapered rectangular block, shown in Fig 3.1. It converges easily because of the simplicity of its geometry. A layer of cells adjacent to the boundary walls is added. Mass and Momentum are extracted from these cells according to Eqs 3.1 and 3.4.

A uniform velocity of 0.05 m/s inlet is specified across the top surface (inlet), pressure outlet at the bottom and moving wall boundary condition with no slip on the sides. The side walls move downwards at 0.05m/s in the z direction.

Mass flow rates at the inlet and the outlet can be checked to see if convergence has been reached. Mass flow rates can also tell if the system is extracting the mass correctly or not. Fig 3.2 shows a plot of the scaled residual error as defined in Eq 2.11.

Fig 3.3 shows that the velocity is constant throughout the domain and equivalent to velocity at the inlet. If the fluid extraction was not done from the side walls, the velocity

would have increased in the tapered section to balance the flow rate at the outlet.

Calculation of dot product of velocity and area normal vector of the faces of the side walls, can give us the flow rate. This should equal the Net balance calculated from Fluent listed in Table 3.2.

A = area of the side walls, in the direction of downward velocity (z)

A = Area of top inlet – Area of bottom outlet

= (1 x 0.5) – (0.8 x 0.4) = 0.18m^2

Flow rate depleted from the sides = ρVA = 7000 x 0.05 x 0.18 = 63 kg/s

Thus, the solution is converged and is correct.

3.4 FLOW RESULTS AND VALIDATION

After verifying the user defined functions (udf) used for mass extraction on a test mold, they were applied on the mold shown in Fig 3.4. Fig 3.7 shows the velocity vector plot on the plane mid way between the two wide faces using $(k-\varepsilon)$. Two main double roll patterns can be seen in this plane, formed because the steel coming out from the side port hits the narrow face wall before splitting into upper and lower recirculation regions. A third narrow recirculation region is created in the lower region by the flow coming out from the bottom port. Fig 3.8 is similar vector plot for time averaged velocity, for the entire mold, obtained using LES [14]. It can be seen that some asymmetry exists between the left and the right hand side of the mold. The results in the vector plot for both $(k-\varepsilon)$ and LES match well.

Fig 3.9(a) gives a quantitative view of variation in velocity with distance below the meniscus on a line 293mm away from the mold center, lying on the plane mid way between the wide faces using $(k-\varepsilon)$. Fig 3.9(b) are the results from LES in [14] for the case where water is used as the liquid and the mold has no shell. The results have been plotted for different meshes, with computational cells ranging from 0.4 – 1.4 million cells. Along with this the results are obtained with and without using the sub grid scale (SGS) model to evaluate the eddy viscosity. The results from water model and steel caster will be different only near the boundaries and thus Fig 3.9(b) can be compared with Fig 3.9(a) [19]. From Fig 3.9(a) it can be seen that the jet hits the narrow wall at a height of 0.3m. This is comparable with Fig 3.9(b) where the jet hits at 0.32 – 0.35 m below the meniscus. The overall variation is velocity is also quite similar, thus suggesting that $(k-\varepsilon)$ model is gives similar time averaged results as LES model for flow in the mold.

Fig 3.10 quantifies the development of the center jet with time averaged downward velocity from LES [14] and downward velocity from $(k-\varepsilon)$ model. The development obtained from both the cases is similar, with the maximum difference seen at the place where the fluid leaves the bottom port. This could be because of not very accurate results obtained for the bottom port from the nozzle simulation. The results show that the jet velocity decreases abruptly as it leaves the bottom port. It is note worthy that the jet's downward velocity becomes almost zero at a distance 1m below the meniscus. The center jet moves downwards before it hits the shell incorporated wide face walls. Fig 3.11 shows the time averaged horizontal velocity on the top surface centerline for $(k-\varepsilon)$ model and LES model used in [14]. The results suggest that the maximum

velocity at this top surface center line is of about 0.225 – 0.24 m/s for both $(k-\varepsilon)$ and LES model. This maximum velocity is obtained at a distance of about 0.2m away from the SEN wall for both cases. Although the velocity pattern is the same for both the models, the difference between the two is more in the region after the maximum velocity has been achieved. Fig 3.12 shows the downward velocity on a horizontal line 0.5m below the meniscus obtained from $(k-\varepsilon)$. Values for this obtained from LES in [14] are also plotted. The trend and values obtained from both the models match well.

Fig 3.13 – Fig 3.15 show the velocity contour plots on different planes. Fig 3.13 shows the contours on a plane mid way between the wide faces. Fig 3.14 shows the velocity on a plane approximately 10mm from the narrow wall and Fig 3.15 shows the velocity on a plane 10mm from the wide face wall. The planes which are 10mm from the wall are not curved as the wall because of the shell thickness incorporated into the model, but are flat inclined planes approximately 10mm away from the walls. These plots showing the values of the velocity in the wall proximity can be useful later in suggesting the areas were particles are more susceptible to be trapped.

3.5 MODEL FORMULATION FOR PARTICLE TRANSPORT AND ENTRAPMENT

The particle transport is modeled by defining equations for particle trajectory. Some forces act on the particles only when it is close to the solidification front. Based on the balance of these force, capture criterion has been defined.

3.5.1 Equations for Particle Transport

The trajectories of inclusion particles in continuous casting of steel can be simulated by integrating the particle velocity (v_p):

$$\frac{dx}{dt} = v_p \tag{3.13}$$

The velocity can be found by integrating the force balance on the particle. This is given as follows:

$$m_p \frac{d\overrightarrow{v_p}}{dt} = \overrightarrow{F_D} + \overrightarrow{F_L} + \overrightarrow{F_{\mathrm{Press}}} + \overrightarrow{F_{Stress}} + \overrightarrow{F_A} + \overrightarrow{F_G} \tag{3.14}$$

Where $\overrightarrow{F_D}$ is the drag force, $\overrightarrow{F_L}$ is the shear lift force, $\overrightarrow{F_{\mathrm{Press}}}$ is the pressure gradient force, F_{Stress} is the stress gradient force, F_A is the added mass term, $\overrightarrow{F_G}$ is the gravitational force and m_p is the mass of the particle. Expressions for these forces are derived below, assuming spherical particles. Particle momentum is small, so particle accelerates to reach steady state very quickly, so acceleration term in Eq 3.14 is quite small.

Drag Force

$$F_D = \frac{1}{8}\pi d_p^2 \rho_f C_D \left|\overrightarrow{v_f} - \overrightarrow{v_p}\right| (v_f - v_p) \tag{3.15}$$

$$C_D = f_{\mathrm{Re}_p}\left(\frac{24}{\mathrm{Re}_p}\right) \tag{3.16}$$

$$\mathrm{Re}_p = \left| v_f - v_p \frac{d_p}{\nu} \right| \tag{3.17}$$

Where, v_f is the velocity of fluid, v_p is the velocity of the particle, C_D is known as

the drag coefficient, Re_p is the particles Reynolds number and f_{Re_p} is the correction factor due to a finite particle Reynolds number and can be found as follows [20]:

$$f_{\mathrm{Re}_p} = (1 + 0.15\,\mathrm{Re}_p^{0.687}) \tag{3.18}$$

Shear Lift Force

Velocity gradients in the fluid create different forces on opposite sides of a particle, which tend to "lift" the particle in the direction of the smaller force. Saffman [21] derived an expression for this lift force on solid spheres in an unbounded linear shear flow :

$$F = 6.46\mu a^2 U_s \,\mathrm{sgn}(G)\left[\frac{|G|}{\nu}\right]^{1/2} \tag{3.19}$$

Where a is the particle radius, μ is the dynamics viscosity, ν is kinematic viscosity, $G = \dfrac{du_1}{dy}$ is the wall normal gradient of the streamwise fluid velocity, sgn means sign of G, and $U_s = v_1 - u_1$ is the instantaneous streamwise velocity difference between the particle and the fluid. In the derivation of this equation it is assumed that the Reynolds number based on the slip velocity, $\mathrm{Re}_s = \dfrac{|U_s|d}{\upsilon}$ is assumed to be much smaller than that defined in terms of the velocity shear, $\mathrm{Re}_G^{1/2} = \left[\dfrac{|G|d^2}{\nu}\right]^{1/2}$, where $d = 2a$ is the particle diameter. This restriction was relaxed by Mclaughlin (1991) [22] and modified the expression for shear lift force.

$$F = -\frac{9}{\pi}\mu a^2 U_s \,\mathrm{sgn}(G)\left[\frac{|G|}{\nu}\right]^{1/2} J^u \tag{3.20}$$

where the function J^u is dependent upon the dimensionless parameter

$$\varepsilon = \mathrm{sgn}(GU_s)\frac{\mathrm{Re}_G^{1/2}}{\mathrm{Re}_s} = \mathrm{sgn}(G)\frac{\sqrt{|G|\nu}}{U_s} \tag{3.21}$$

The general expression for J^u is rather complicated, therefore Mie [23] reconstructed it using curve fitting for $0.1 \le \varepsilon \le 20$:

$$J(\varepsilon) = 0.6765\{1 + \tanh[2.5\log_{10}\varepsilon + 0.191]\}\{0.667 + \tanh[6(\varepsilon - 0.32)]\} \tag{3.22}$$

Although derived for solid spheres, this expression also holds for bubbles if their surfaces have been covered with small solid inclusions, which typically happens due to collisions with tiny alumina particles in the molten steel.

Pressure Gradient and Stress Gradient forces

The pressure gradient force contributes to the hydrostatic component of the buoyancy force, due to difference in particle and fluid density. This force can be important when the particle density is comparable or lighter than the fluid. These two forces can be calculated as follows [20]

$$\overrightarrow{F_{press}} + \overrightarrow{F_{Stess}} = -\frac{\pi d_p^3}{6}\frac{D\overrightarrow{v_f}}{Dt}\rho_f \tag{3.23}$$

The material derivative present in the above equation is written in terms of velocity gradients.

Added Mass Force

The added mass force arises from the acceleration of the surrounding fluid by the particle moving through it. It can be expressed as follows: [20, 24]

$$\overrightarrow{F_A} = \frac{C_A\rho\pi d_p^3}{12}\left(\frac{D\overrightarrow{v_f}}{Dt} - \frac{d\overrightarrow{v_p}}{dt}\right) \tag{3.24}$$

$$C_A = 2.1 - \frac{0.132}{0.12 + Ac^2} \tag{3.25}$$

$$Ac = \frac{\left|\overrightarrow{v_f} - \overrightarrow{v_p}\right|^2}{d_p \frac{d\left|\overrightarrow{v_f} - \overrightarrow{v_p}\right|}{dt}} \tag{3.26}$$

where C_A is the correction factor due to the acceleration effect and Ac is a constant called the acceleration parameter.

Gravitational Force

The gravitational force arises due to buoyancy due to the density difference is equivalent to the weight of the particle and can be written as follows:

$$F_G = \frac{1}{6}\pi d_p^3 \rho_p \vec{g} \tag{3.27}$$

3.5.2 Forces Close to Solidification Front

There are three further forces that are important in determining the capture of a particle along with the previous mentioned hydrodynamic forces when it is very close to a dendritic solidification front. These are:

- Lubrication force
- Van der Waals interfacial force
- Surface Energy Gradient force

These forces are computed assuming that the particle is in equilibrium at the dendritic interface. If the forces balance, the particle is pushed along with solidification front. Otherwise, the direction of the imbalance indicates if the particle is captured or released.

Lubrication force

This force arises because fluid must flow between the gap between the particle and the dendrite tip, in order to keep the particle moving with the solidification front. The thickness of this gap is much smaller than the diameter of both the dendrite tip and the particle but the gap is large than a critical distance which will be defined later. The force acts on the particle along particle's radius towards dendrite tip. The force can be written as: [25, 26]

$$F_{\text{lub}} = 6\pi\mu V_{sol} \frac{R_p^2}{h_o}\left(\frac{r_d}{r_d + R_p}\right)^2 \tag{3.28}$$

where V_{sol} is the solidification velocity, R_p is the particles radius, r_d is the radius of the dendrite tip and h_o is the distance between the dendrite tip and the particle. This expression represents the maximum of this force, as fluid may also flow around the dendrite tips and between the dendrite trunks in a 3-D dendritic array.

Van der Waals interfacial force

This force arises from the weak attraction between all atoms in the particle and dendrite tip, and is negligible unless the distances become very small. The function for this force has been obtained from [27] which describes the force on a spherical particle in front of a solidifying interface with a convex curvature of radius r_d.

$$F_I = 2\pi\Delta\sigma_o \frac{r_d R_p}{r_d + R_p}\frac{a_o^2}{h_o^2} \tag{3.29}$$

$\Delta\sigma_o = \sigma_{sp} - \sigma_{sl} - \sigma_{pl}$, where σ is the surface energy, the subscripts s, p and l denote solid, particle and liquid respectively, a_o is the atomic diameter of the liquid and h_o is

the distance between the dendrite tip and the particle. This force pushes the particle away from the dendrite tip

Surface Energy Gradient force

The surface tension of liquid steel changes with its temperature and composition. Yuan in [2] has explained that in deoxidized steels, sulfur is the major solute that is interfacially active. Rejection of solute ahead of the solidifying interface causes the sulfur content on the cold side of the particle to be higher than on the hot side. The lower sulfur content decreases the surface tension acting around that half of the particle, relative to the outside, high-tension half. The resultant force tends to push the particle towards the solidification front. Following Kaptay's work [28], for force on a spherical particle in front of a planar interface, Yuan [2] derived the following expression for this force acting on a spherical particle close to a hemispherical dendrite tip.

$$F_{Grad} = -\frac{m\beta\pi R_p}{\xi^2}\left\{\frac{\left(\xi^2 - R_p^2\right)}{\beta}\ln\left[\frac{\left(\xi + R_p\right)\left[\alpha\left(\xi - R_p\right)+\beta\right]}{\left(\xi - R_p\right)\left[\alpha\left(\xi + R_p\right)+\beta\right]}\right] + \frac{2R_p}{\alpha} - \frac{\beta}{\alpha^2}\ln\left[\frac{\alpha\left(\xi + R_p\right)+\beta}{\alpha\left(\xi - R_p\right)+\beta}\right]\right\} \tag{3.30}$$

$$\alpha = 1 + nC_o \tag{3.31}$$

$$\beta = nr_d\left(C^* - C_o\right) \tag{3.32}$$

$$\xi = R_p + r_d + h_o \tag{3.33}$$

C^* can be found from the expression below

$$\frac{V_{sol}r_d}{2D_s} = \frac{C^* - C_o}{C^*(1-k)} \tag{3.34}$$

Where, m and n are empirical constants with values of 0.17J/m^2 and 844(mass %)$^{-1}$. C_o is the sulfur content of steel, D_s is the diffusion coefficient of sulfur in steel, k is the

distribution coefficient and h_o is the distance between the dendrite tip and the particle. This force acts along the particle radius towards the dendrite tip radius. Following Kaptay [28], the negative sign in the above equation represents attraction towards the interface.

3.5.3 Capture Criterion at Solidification Front

When particles reach a domain boundary, a criterion must be applied as the boundary condition. This work adapts the methodology developed by Yuan [2] for a transient model into the current RANS model. Entrapment is determined by a criterion based on particle size and force balance [2].

Particles with diameter smaller than the PDAS at the solidification front, are captured once they hit the boundary wall. For particles with diameter greater than the PDAS, the following capture criterion is used to evaluate their fate.

Fig 3.16 shows the forces acting on a particle close to the solidification front that are used in the capture criterion. Whenever particle with diameter greater than PDAS touches the boundary (solidifying shell), forces are compared in the χ direction, which is the normal vector to the boundary face of the solidification front (see Fig. 3.5). The boundary face direction represents the direction of growth of the solidification front, which depends on the rate of shell solidification. The χ direction is normal to the solidification front at the point where the particle touches the shell.

$$\vec{\chi} = \frac{\vec{A}}{|\vec{A}|} \tag{3.35}$$

where $\vec{A}$ can be seen in Fig 3.5.

If the net force in the χ direction is away from the interface, then particle pushing will occur, and the fluid flow will likely sweep the particle, so it is not captured. This is rarely found to be the case, however, for the conditions simulated to date. If the net force in the χ direction is towards the solidification front, then the next step is to balance forces in the η direction (direction of particle motion tangential to the solidification front), to determine if particle rotation can occur.

The η direction lies in the plane tangential to the local solidification front (domain boundary) and is the projected component of the sum of the buoyancy and drag forces in that plane. It represents the direction that a particle can most easily rotate around the dendrite tips and be transported away with the flowing liquid. The effect of the local roughness on this rotation direction is ignored. The roughness effect depends on the local arrangement of the dendrites, but this has been shown to average out (ie the 2-D force balance presented here is in between the easiest and most difficult directions of rotation in the a 3-D array of hexagonal dendrites). If the previous step indicates that the particle is captured in the normal direction, then a force balance in the tangential (η) direction is conducted. First the η direction (unit vector) is evaluated from fluid flow direction

$$\vec{\eta} = \frac{(\overrightarrow{F_D} + \overrightarrow{F_B}) - [(\overrightarrow{F_D} + \overrightarrow{F_B}) \bullet \vec{\chi}](\vec{\chi})}{\left|(\overrightarrow{F_D} + \overrightarrow{F_B}) - [(\overrightarrow{F_D} + \overrightarrow{F_B}) \bullet \vec{\chi}](\vec{\chi})\right|} \tag{3.36}$$

This model requires as input data as a function of position over the solidification front (domain boundary): the Primary dendrite arm spacing (PDAS), the solidification front

velocity (V_{sol}) and the dendrite tip radius (r_d).

The algorithm followed for this is given below:

If $(2 * R_p \le PDAS)$

Where R_p is the particle radius

Particle is captured otherwise the force balance analysis below is checked:

If

$$\overrightarrow{F_L} + (\overrightarrow{F_B} \bullet \vec{\chi}) + (\overrightarrow{F_D} \bullet \vec{\chi}) > 2 * (\overrightarrow{F_{Lub}} - \overrightarrow{F_{grad}} - \overrightarrow{F_{vand}}) * \cos(\theta) \quad (3.37)$$

where,

$$\theta = acr\sin\left[0.5\mathrm{PDAS} \Big/ \left(R_p + r_d\right)\right] \quad (3.38)$$

particle is pushed away

Else rotation is checked

If both $\overrightarrow{F_D} \bullet \vec{\eta}$ **and** $\overrightarrow{F_B} \bullet \vec{\eta}$ **are in the same direction, (ie same sign) then if**

$$(\left|\overrightarrow{F_D} \bullet \vec{\eta}\right| + \left|\overrightarrow{F_B} \bullet \vec{\eta}\right|) * \cos(\theta) + (\left|\overrightarrow{F_D} \bullet \vec{\chi}\right| + \left|\overrightarrow{F_B} \bullet \vec{\chi}\right|) * \sin(\theta) + \overrightarrow{F_L} \le (\overrightarrow{F_{Lub}} - \overrightarrow{F_{grad}} - \overrightarrow{F_{vand}}) * \sin(2\theta)$$

(3.39)

Particle is captured

Else if $\overrightarrow{F_D} \bullet \vec{\eta}$ **and** $\overrightarrow{F_B} \bullet \vec{\eta}$ **are in the opposite direction, (ie have different signs) then if**

If $\left|\overrightarrow{F_D} \bullet \vec{\eta}\right| > \left|\overrightarrow{F_B} \bullet \vec{\eta}\right|$ **, then if,**

$$(\left|\overrightarrow{F_D} \bullet \vec{\eta}\right| - \left|\overrightarrow{F_B} \bullet \vec{\eta}\right|) * \cos(\theta) + (\left|\overrightarrow{F_D} \bullet \vec{\chi}\right| + \left|\overrightarrow{F_B} \bullet \vec{\chi}\right|) * \sin(\theta) + \overrightarrow{F_L} \le (\overrightarrow{F_{Lub}} - \overrightarrow{F_{grad}} - \overrightarrow{F_{vand}}) * \sin(2\theta)$$

(3.40)

Particle is captured

else, i.e $\left|\overrightarrow{F_B} \bullet \vec{\eta}\right| > \left|\overrightarrow{F_D} \bullet \vec{\eta}\right|$**, then if,**

$$(\left|\overrightarrow{F_B} \bullet \vec{\eta}\right| - \left|\overrightarrow{F_D} \bullet \vec{\eta}\right|) * \cos(\theta) + (\left|\overrightarrow{F_D} \bullet \vec{\chi}\right| + \left|\overrightarrow{F_B} \bullet \vec{\chi}\right|) * \sin(\theta) + \overrightarrow{F_L} \leq (\overrightarrow{F_{Lub}} - \overrightarrow{F_{grad}} - \overrightarrow{F_{vand}}) * \sin(2\theta)$$

(3.41)

Particle is captured

This process can be followed through a flow chart in Fig 3.17.

3.5.4 Incorporating Turbulence

The flow solution obtained earlier is at steady state and thus the fluctuating velocity components need to be artificially added in order to simulate particle motion accurately. The effects of turbulence on the dispersion of particles due to turbulent eddies present, in the continuous phase, are also included. The dispersion of particles due to turbulence can be predicted by either using the particle cloud model or stochastic tracking (random walk) model.

The cloud method, tracks a cloud of particles about a mean trajectory. To understand the cloud method to represent the turbulent dispersion of particles, a simulation is used. In this model the particles disperse about a mean trajectory. The mean trajectory is based on the average motion of particles in a cloud of certain diameter. The cloud itself expands due to turbulent dispersion. The particle end positions are shown in Fig 3.46. It can be seen that the particles concentrate about the mean trajectory and become dependent on the motion of the other particles. The motion thus becomes too streamlined. Thus, this method poorly depicts particle motion in a real caster. Furthermore, the model is computationally intensive, requiring about 24 hours to track

only 1250 particles, with no capture criterion (the user defined functions needed for larger particles would further slow down the computations).

Stochastic methods are used in random walk to include the effects of turbulent velocity fluctuations. The turbulent dispersion of particles is predicted by integrating the trajectory equations for individual particles, using the instantaneous fluid velocity, $\bar{u} + u'(t)$, along the particle path. In the random walk method that is used in this thesis, interaction of particles with fluid phase turbulent eddies is simulated. The values of u', v' and w' that exist during the lifetime of an eddy are sampled by assuming that they obey a Gaussian probability distribution, therefore

$$u' = \zeta\sqrt{\overline{u'^2}} \tag{3.42}$$

Assuming isotropy,

$$\sqrt{\overline{u'^2}} = \sqrt{\overline{v'^2}} = \sqrt{\overline{w'^2}} = \sqrt{\frac{2k}{3}} \tag{3.43}$$

where k is the kinetic energy of the fluid.

This method is faster computationally, and also performs better than the cloud model, as shown in the next sections, so was adopted for the rest of this work.

3.5.5 Solution Procedure

First the 3-D fluid flow equations, Eqs. 3.1, 3.4 and 3.9-3.10, are solved for the time-averaged flow field in the nozzle and mold using FLUENT [29]. Then, particles are introduced at the inlet plane of the nozzle or mold. The Discrete phase model in Fluent is used to achieve the transport of particles. The case set up in Fluent for particles

is given in Appendix B.4. Particles are placed at random positions by generating a file in appropriate format for FLUENT, using a matlab program. The Matlab code and file format are provided in the Appendix B.5. During the simulation, the entrapment criterion is evaluated each time a particle touches a boundary. This event triggers FLUENT to evaluate a user defined function, provided in the Appendix B.8, which evaluates the procedure given in Section 3.53 and the flow chart Fig 3.17. Appendix B.6 shows how to extract the end positions of entrapped particles from Fluent and Appendix B.7 shows how to visualize them. Particles smaller than the smallest PDAS are always captured, so the UDF is not needed for simulations involving such small particles. Primary dendrite arm spacing (PDAS) varies down the mold length and is incorporated as the function representing the measured PDAS [30]. Solidification front velocity also varies down the mold length and is incorporated by using the data obtained from [2].

3.6 VALIDATION AND COMPARISON OF FORCES ACTING ON THE PARTICLE

To compare the magnitude of the hydrodynamic forces in this k-e model with those obtained in previous work with LES [2], data was extracted from the results for two different particles released at two positions into the mold with no turbulence. This was done with hydrodynamic forces using velocity gradients extracted from Fluent using its predefined macros. Their trajectories are re-plotted for the three different drag conditions listed below

- Built-in drag force function of Fluent as defined in “Bubbles, Drops, and Particles" (1978) [1] (no user-defined function included)

- Drag force function as defined by Yuan [2] (added with user defined function)
- Drag force function as defined in “Bubbles, Drops, and Particles" (1978) [1] (added with user defined function)

These trajectories can be seen in Fig 3.18 and Fig 3.19. The trajectories almost overlap, showing that the user defined function is written correctly.

The shear lift force built into Fluent is invoked only for sub micron particles so a user defined function is made for the shear lift force. Fig 3.20 and Fig 3.21 show the trajectories of particles of different diameters, including the Saffman shear lift force used in Fluent and the shear lift force used by Yuan in [2]. For the smaller particle of 100μm, the trajectories vary only slightly, but for particle diameter of 400μm, the trajectories are affected by the type of shear lift force used.

The pressure gradient, stress gradient, added mass, and buoyancy forces are already present in the discrete phase particle trajectory model of Fluent. The basset history force is not included in this work because Yuan [2] found that this force is small and can be neglected.

Fig 3.22 shows how the particle trajectory is affected by particle size. Larger particles tend to rise more. Fig 3.23 shows how particle trajectory is affected by particle density. It can be seen that particles with lower density rise more. The magnitudes of these forces were calculated in an Excel spread sheet, based on the values of variables and gradients extracted from pre-defined macros in Fluent using user defined functions.

Magnitude of the hydrodynamic forces acting on alumina particles of 400µm and 100µm, from LES simulation, can be seen in Fig 3.25 and Fig 3.27 respectively [2]. The magnitudes obtained from Fluent for 400µm alumina particle and 100µm alumina particle are presented in Fig 3.24 and Fig 3.26 respectively. A comparison between results from LES [2] and k-e model in Fluent, show that the values of these forces are quite similar if a mean value of these forces is looked at, as it is not known where in the flow these particles were released for LES modeling. The material derivative present in Eq 3.23 and 3.24 is written in terms of velocity gradients. Fluent has predefined macros to get the values of velocity gradients and thus, this force can be extracted then from Fluent. The Fb force on the plots is the net force due to buoyancy and gravity. The two main prominent forces are the drag and buoyancy forces. The basset history force is small and was not calculated with Fluent. The drag force and buoyancy forces become more prominent with increasing particle diameter. Fig 3.28 shows these forces for a slag particle of 400µm. Note that for k-e model, these hydrodynamic forces were tracked for approximately 1.5sec of the particle trajectory time in the mold.

Fig 3.29 shows the magnitudes of the forces present when the particle is close to the solidification front. It can be seen that the surface gradient force is the most dominant of the three forces, as shown on Fig 3.29. It will further be seen how varying different parameters affect these forces and how they contribute towards determining particle entrapment.

3.7 PARAMETERS AFFECTING PARTICLE CAPTURE

The capture criterion, given in the force balance Eqs 3.35 - 3.41, incorporates a great

deal of information on the entrapment behavior of particles at a metal solidification front, and the likely-hood of a fraction of particles trapped in a given set of process conditions.

This section investigates the effect of the following parameters on the forces, the force balance, and the conditions for particle entrapment:

- Particle diameter
- Particle density
- Primary dendrite arm spacing (PDAS)
- Sulfur concentration
- Dendrite tip radius (rd)
- Solidification front velocity (Vsol)
- Solidification front angle: angle of solidifying shell (wall) with the horizontal (ø) (decreases with distance below meniscus, with decreasing machine radius)
- Cross flow velocity

Schematic Fig 3.30 and Fig 3.31 show how the direction of forces change, depending on the distance of a particle down in the curved portion of a caster, and on whether it is on the inner radius or the outer radius. Fig 3.32 shows the direction of forces when the flow is in horizontal direction along the caster wall, with buoyancy still vertically up. Plots are made to study how particle capture is affected in such scenarios.

The effect of the process parameters discussed above on the critical cross-flow velocity for particle capture are investigated by varying their value in the capture criterion

equations. Graphs were constructed as a function of particle diameter to evaluate the force balance equations to find the "critical" velocity that exactly satisfies the capture criterion equation. Lines are constructed to divide the velocity – diameter space into regions where particles are captured, or can escape by rotating away. As the equations for capture criterion are non-linear, the goal seek function was used in excel.

The results are presented in Figs 3.33 – 3.42. These figures represent maps that illustrate the conditions when particles are captured or not. Each graph based on vertical flow reveals that a "capture window" exists, that divides the graph into 3 regions. To the left, (inside the capture window) is the region where particles with diameter less than the PDAS get captured. Above this capture window is the region where particle can rotate downwards and below this capture window is the region where particle can rotate upwards to escape. The two lines which divide the latter 2 regions narrows with increasing particle diameter, but usually extend indefinitely. This tiny narrow region represents the flow conditions where the particle is exactly suspended in front of the solidification front, which is moving downward at the casting speed. To achieve this, the upward terminal velocity of the particle relative to the fluid due its buoyancy, must be exactly matched by the downward vertical flow velocity in a reference frame moving downwards at the casting speed. The downward flow velocity in the lab frame of reference is found by adding the casting speed to the y-axis velocity on these graphs. In other words, the y axis velocity represents the downward vertical velocity across the solidification front (in the usual lab frame of reference) subtract the casting speed.

Capture depends greatly on the orientation of the moving wall of solidifying dendrites,

and the direction of the cross-flow along the surface of that wall. Particles are easiest to capture if there is the horizontal component of cross-flow velocity is zero, and if the vertical cross-flow velocity is exactly opposite to the rising velocity of the particle due to buoyancy. This is how the previous graphs Figs 3.33 – 3.42 were constructed. Graphs with vertical cross-flow velocity include both positive and negative regions because the result differs greatly if flow is in the same or opposite direction to the direction of buoyancy, which tends to encourage the particle to drift upwards. Alternatively, graphs have been constructed for with horizontal cross-flow velocity, where flow is perpendicular to the buoyancy force.

Particle diameter

The inclusions diameters in steel casters can vary from 10um – 500um and the bubbles diameters can be as large as even 2000um or even higher. The effect of particle diameter is included in Figs 3.33 – 3.38.

As particle size increases the chances for the particle to get captured decrease. Every particle smaller than the PDAS is captured if it reaches the solidification front, regardless of fluid velocity. Once a particle enters between the dendrite tips, the chances of turbulent flow taking it back out again are assumed to be negligible. Larger particles are also captured, if the cross flow velocity is small, because the forces acting at the dendritic front are unable to cause particle pushing for any of the cases investigated in the case of vertical flow. A larger particle can more easily rotate about the dendrite tip radius, so has a lower critical velocity. Very large particles are captured only if the velocity across the dendrites is zero.

Particle density

The inclusion composition affects capture in this work only by affecting the particle density. Inclusions can be slag particles that have been sheared off from the top slag layer, due to velocity fluctuations at the top surface of the mold. These can also be alumina particles or argon gas bubbles that have been introduced from the outlet of the nozzle.

The particle density greatly affects particle capture through its effect on the particle buoyancy force. Density is controlled by the particle composition, and is listed in Table 3.3 for the three different cases investigated. Increasing the density of the particle decreases its buoyancy force and therefore lessens the chances for the particle to drift upwards. Argon, which has almost zero density, represents an extreme case where the buoyancy force is the maximum possible. A comparison of Slag, Alumina and Argon particles for vertical flow along a vertical wall is given in Fig 3.33.

With decreasing density, the capture window translates to higher downward vertical velocities (to balance the higher buoyancy force), but the height of the window decreases only slightly. Fig 3.42 shows the capture and non-capture area on the plot, with a horizontal flow in the direction shown in Fig 3.32. The argon bubble is more likely to rotate about the dendrite tip, than slag, because it has a higher buoyancy force.

The direction of rotation is in the η direction, which is in the direction of the net force of the drag from the cross-flow velocity and the buoyancy (upwards). The direction

increases from vertical (at zero cross-flow velocity) to almost horizontal (at large cross-flow velocity).

Primary dendrite arm spacing (PDAS)

Primary dendrite arm spacing depends on several parameters such as the cooling rate at the boundaries, composition of steel, and solidification front velocity to name a few. Solidification theory has been developed to predict the arm spacing [31]. Experimental measurements of the variation in PDAS down the mold length [32] is shown in Fig 3.43. Carbon content has a major effect on the PDAS [30]. Temperature gradients also have a large effect [2].Yuan [2] combined these facts to calculate PDAS based on the temperature gradients at the solidification front calculated at the model domain boundaries by the LES model. The predictions roughly match the measured arm spacings, as can be seen in Fig 3.43. The temperature equations are not being solved in this thesis for $k-\varepsilon$ model and therefore for modeling purposes, the measured data has been smoothened and the variation of PDAS as seen in Fig 3.44 is then used, when the force balance analysis is incorporated in the computational model.

The effect of different PDAS is shown in Fig 3.34 for argon bubbles (in vertical flow along vertical walls). The larger the PDAS, the larger is the area of capture on Fig 3.34. The measured values of PDAS shown in Fig 3.43 go up to 210μm. It has been seen that with lower carbon concentration or lower cooling rate in steel, the PDAS increases [30]. Thus, it is important to see the theoretical behavior of particle with PDAS larger than 210μm.

Sulfur Concentration

Yuan in [2] showed that sulfur is the major solute contributing to the surface gradient force in killed steel, where oxygen content is low. Fig 3.29 shows that the surface gradient force has the highest magnitude among forces that become active when particles are near the dendrites.

Sulfur concentration in the solidifying steel affects particle capture by causing the gradient in surface tension, and it corresponding force towards the solidification front. This work investigates the effect of increasing sulfur concentrations from a typical low level of 0.0028 wt pct to a typical high level of 0.03 wt pct. Fig 3.36 – Fig 3.38 show that the effect on increasing the ease particle capture is noticeable, but not as large as previous variables investigated. It can be seen that higher sulfur increases particle capture. For example, 100μm particles require downward cross-flow to increase from 0.06 m/s to 0.08 m/s to avoid capture. Thus, high sulfur steels might capture more inclusions.

Dendrite tip radius and solidification front velocity

The dendrite tip radius varies according to the solidification conditions, which include: front velocity, composition, liquidus and solidus temperature. A general expression by Kurz and Fisher [31] to evaluate the tip radius is given below:

$$r_{tip} = 2\pi \left(\frac{D_o \, \sigma_{sl} / S}{V_{sol} k \Delta T_o} \right)^{1/2} \tag{3.44}$$

$$\Delta T_o = (T_L - T_s) \tag{3.45}$$

where, D_o is the diffusion coefficient of the solute, σ_{sl} is the specific solid-liquid interface energy, k is the distribution coefficient, and S is the melting entropy.

The solidification velocity depends on the balance between superheat delivery by the flowing steel jets in the mold, and the extraction of heat from the cold side of the solidifying shell. Its variation down the mold, and between the narrow and wide face can be predicted by CON1D [2]. It decreases abruptly near the top of the mold, and then becomes fairly constant down the mold.

The solidification velocity affects the force of lubrication that acts where the dendrite tips nearly touch to particle. In the real caster, the solidification front velocity varies as shown in Fig 3.45 for the validation mold used with casting speed of 25.4mm/s. Fig 3.35 shows the behavior of argon bubble under two different conditions of solidification velocity and tip radius. Solidification velocity was varied from 200μm to 500μm. The dendrite tip radius of 2.13μm and 3.3μm were investigated. Increasing the solidification velocity causes a higher force of lubrication. However, the magnitude of the lubrication force was found to be negligible for all cases investigated here. The surface gradient force is an order of magnitude higher than the lubrication force, even at with low sulfur concentration.

Solidification front angle (ϕ)

Most casters are vertical for some distance below the meniscus, but then begin to curve, eventually becoming horizontal before cutting into slabs. This change in orientation greatly affects the direction of forces such as buoyancy and needs to be taken into

account for a force balance. These differences are responsible for some of the significant differences in capture between the inside and outside radius and between vertical and straight-mold casters. Fig 3.16 shows the direction of forces acting on the particle when the solidification front angle is 90deg. Fig 3.30 and Fig 3.31 show the direction of forces acting on the particle when it touches the inner and outer radius respectively, where the solidification front angle is not 90deg. Keeping in view the capture criterion explained earlier and Fig 3.31 for the outer radius, Eqs 3.37-3.41 can be rewritten for this case. Note the directions of χ , η and the forces in this case shown on Fig 3.31. The direction of χ it is affected by ø and plays an important role in the equations. .

Outer Radius:

If $F_B \cos\phi + F_L > 2(F_{Lub} - F_{Grad} - F_I)\cos\theta$ (3.46)

Particle is pushed away.

Else check if the particle can rotate about a dendrite tip

$(F_D - F_B \sin\phi)\cos\theta + (F_L + F_B \cos\phi)\sin\theta \le (F_{Lub} - F_{Grad} - F_I)\sin 2\theta$ if

$F_D > F_B \sin\phi$ (3.47)

or

$(F_B \sin\phi - F_D)\cos\theta + (F_L + F_B \cos\phi)\sin\theta \le (F_{Lub} - F_{Grad} - F_I)\sin 2\theta$, if $F_B \sin\phi > F_D$

(3.48)

the particle is captured.

Previous results have investigated vertical walls. Inclining the solidification front has a

great influence on particle capture. On the inside radius, more incline of the angle of the wall (decreasing, positive values of ϕ) causes easier capture of particles, as seen in Fig 3.39.

On the outside radius, Fig 3.40 shows that the particles can drift away from the dendrite tips, in the radial direction. A critical particle size exists that if exceeded will avoid capture for any flow condition. At lower ϕ the buoyancy component encourages the movement of particles upwards, towards the inner radius.

Cross flow velocity

The cross-flow velocity is the difference between the actual fluid velocity at the solidification front, and the casting speed. The cross-flow velocity determines the direction of the drag force on the particle, and varies over the solidification front according to the local flow field. Increasing the cross-flow velocity allows easier rotation of the particle about the dendrite tip, allowing it avoid capture at that location. These show how changing the above mentioned parameters can alter the fate of the particle.

Horizontal Flow:

The particles can also be entrapped even when the flow is in horizontal direction which is more likely to be present on the wide faces of the strand. The direction of the flow across the wide face determines the drag force direction and can vary greatly with position and casting conditions. On the other hand, the buoyancy force always acts vertically in the direction opposite to the gravitational force, irrespective of the flow

direction. To determine the critical cross flow velocity required to enable the particle to rotate, force balance analysis needs to be performed. A net force being the sum of drag and buoyancy force is determined. The dendrite tips are then assumed to be placed in line with the net force direction. The particle can either then move in this net force direction or get captured by the dendrite arms. To illustrate this, orientation 1 in Fig 3.32 shows a particle close to the wide wall with a horizontal flow direction. Once the sum of drag and buoyancy force is determined, the problem can then be visualized as shown in orientation 2 in Fig 3.32. Thus, the vertical flow equations where the drag and buoyancy are in the same direction are needed to determine the fate of this particle but the particle will not move in vertical direction but rather in the net force direction.

3.8 3D FLOW, PARTICLE TRANSPORT AND CAPTURE

The full 3D fluid flow and particle transport model was applied to investigate particle entrapment in a typical continuous casting nozzle and mold. The particles are introduced in the validation mold and trifurcated nozzle when the flow solution has already been obtained for both. The domain boundaries represent the solidification front, so particles touching a domain boundary are subjected to the capture criterion. The solidification front shape is given in Fig 3.5. The nozzle flow pattern was obtained earlier in Chapter 2 and was validated. The velocity vector plot at the nozzle bottom can be seen in Fig 3.47.

3.8.1 Nozzle Entrapment

To simulate particle attachment to the nozzle walls and the initial stages of clogging, 20,000 spherical particles are introduced at random positions at the inlet of the tundish

region. The particles are always captured if they hit a nozzle wall. The diameters and densities of the four groups of particles modeled are given in Table 3.4.

The entrapped positions of the particles are shown in Fig. 3.47, and indicate potential regions for clogging. 39% of the particles are trapped at the nozzle walls, with most of them concentrated at the region of the well nozzle and the bottom of the SEN. This is much more than the 16% that typically occurs in practice, [33] and indicates that a better entrapment criterion is needed at the nozzle walls. Fig 3.48 shows the particle positions at the nozzle outlet. The distribution is reasonably random, except that naturally, none of the particles exit through the recirculation regions at the top or bottom of these ports.

3.8.2 Mold Entrapment

Particles were added in the mold through the mold inlet ports. From the nozzle simulation, only 12% of the particles exiting the ports into the mold through the bottom outlet port while the rest 88% passed through the side ports. Particles were introduced into the mold at the port inlet surfaces with these same fractions.

Small Particles

Yuan introduced 40,000 small particles into the mold using LES [34] in 9sec. For $(k-\varepsilon)$ model, with half the mold simulation, 20,000 particles were introduced into the mold, with the same diameters and densities as used by Yuan. The break down of particle groups is the same as that used in the nozzle and is in Table 3.4.

Fig 3.49 (a) shows the particle positions in mold with LES [34] 2 sec after they begin to enter the mold. Fig 3.49 (b) was extracted from the k-e model results, as explained in the Appendix B.9. The general behavior of particle transport and dispersion in the two models agree well. However, the percentage of particles reaching the top surface differs. About 8% particles are predicted to reach the top surface of the mold with LES and only 3.5% reach the top surface with $(k-\varepsilon)$ model. Fig 3.50 shows the entrapped particle positions in the mold and on the narrow face using $(k-\varepsilon)$ model. A further 3% of the particles are trapped at the top of the narrow face close to the meniscus.

Large Particles

Six different groups of large particles were simulated, as given in Table 3.5. Fig 3.51–Fig 3.53 shows the entrapped position of particles for each Alumina, Argon and Slag particles for different particle sizes. Table 3.6 lists the percentage of particles removed by the top surface. It can be seen that with seen that with increasing particle diameter, the fraction of particles removed by the top surface increases. This fraction also increases with decreasing particle density. These results are consistent with the expectations from the entrapment criterion. In addition, however, the larger particles and the lower density particles also tend to float towards the top surface relative to the flow pattern, giving them an extra reason for their higher removal rates.

Fig 3.54 shows the position on the strand surface where alumina particles of various sizes were predicted to become entrapped on the narrow mold face. The number of particles trapped near the jet region and below decreases with increasing particle size, owing to the smaller particle capture window, as shown in Figs 3.33 - 3.38. The number

of particles trapped at the top of the narrow face increases, showing the same trend experienced at the top surface with increasing particle size.

Yuan simulated the same groups of Alumina particles using LES [2]. The percentage of particles removed by the top surface using LES for Alumina particles are listed in Table 3.7. The entrapment locations and qualitative trends consistently agree between the two models. However, it can be seen that the $(k-\varepsilon)$ model under predicts the number of particles removed by the top surface. Both models show that most of the particles are entrapped in or just below the mold, so concentrate near the surface. Furthermore, the majority of particles entering the mold are entrapped, so mold fluid flow design should focus on meniscus stability, while upstream refining operations should focus on removing inclusions before they enter the mold. This finding agrees with previous work, [3] and shows that the k-e model can be used as at least a qualitative tool to gain insight into particle entrapment.

3.8.3 Discussion

Although the time averaged velocities of from both LES and $(k-\varepsilon)$ model matched well as seen earlier in this chapter, the particle motion is highly dependent on turbulent velocity fluctuations. The $(k-\varepsilon)$ model assumes turbulent velocity fluctuations are isotropic, based on the average kinetic energy. However, the velocities in each spatial direction in the mold are very different, as compared in Figs 3.55 – 3.57. As revealed by LES, the turbulent velocity fluctuations are strongly dependent on the velocity components. This explains the smaller fraction of particles reaching the top surface in the k-e model, compared with the LES. The larger component of the fluctuating velocity

towards the narrow and wide faces associated with the isotropic turbulence of the k-e model makes contact with the walls more likely in the k-e model. Turbulence in the LES model was strongly skewed in the flow direction, so particles generally moved further with the flow before touching the solidification front.

These findings suggest that a more accurate description of the turbulence, and the corresponding fluctuating velocity components is required in order to improve the accuracy of the RANS models in predicting particle entrapment.

3.9 CONCLUSIONS

A method to incorporate decreasing fluid mass in the mold was implemented in Fluent. The $(k-\varepsilon)$ model prediction for steady flow in the mold match well with the time averaged results from LES. The variation in several parameters on the particle capture criterion was investigated parametrically as a function of cross velocity at the shell / liquid interface. Along with this, comparison of different hydrodynamic forces in the mold with $(k-\varepsilon)$ model and LES model were compared.

Increasing primary dendrite arm spacing has the most important effect increasing particle capture. Small particles are always captured when they touch the solidification front. Particle composition (density: bubble vs. inclusion) shifts the capture window. Bubbles escape more easily than solid inclusions in stagnant flow regions, but their capture depends on the flow pattern. Although steels with low sulfur content tend to have less particle entrapment, the effect is small. The increased ease of particle capture on the inner radius is a large effect (relative to vertical or outer radius).

Values of the hydrodynamic forces are quite similar for both models, if a mean value of these forces is looked at, as it is not known where in the flow these particles were released for LES modeling

Particle transport and entrapment model was incorporated into Fluent and several simulations were performed for particles of different sizes and densities. The positions where the particle get trapped in the nozzle, are potential areas where clogging can occur. The mold simulations quantify how particles with larger diameters and lower densities are more likely to reach the top surface. Specifically, 41% of 400μm argon particles reach the top surface compared to only 5% of 100μm argon particles. Similarly, the fraction of 400μm diameter the number of argon particles (density $\simeq$ 0 kg/m^3) that reach the top surface is 3 times greater than that of slag particles (density = 5000kg/m^3).

The results for $(k-\varepsilon)$ model for particle removal percentages do not match with those obtained from LES. Magnitude of velocity components in all spatial directions suggests that assumption of isotropic velocity fluctuations used in $(k-\varepsilon)$ model under-predicts the number of particles reaching the top surface by a factor of about 1/2.

3.10 TABLES AND FIGURES

Table 3.1. Specified variation in under-relaxation factors.

	Pressure	Density	Body Force	Momentum	Turbulence Kinetic Energy	Turbulence Dissipation Rate	Turbulent Viscosity	Number Of Iterations
α	0.5	1	1	0.6	0.7	0.7	0.8	0
α	0.5	0.8	0.8	0.5	0.7	0.7	0.8	40
α	0.5	0.7	0.7	0.4	0.6	0.6	0.7	100
α	0.5	0.6	0.6	0.4	0.6	0.6	0.6	220
α	0.4	0.5	0.5	0.4	0.5	0.5	0.5	4485
α	0.3	0.5	0.5	0.3	0.4	0.4	0.4	5618

Table 3.2. Steel properties and flow imbalance in test mold.

Steel density	7000 Kg/m3
Steel Viscosity	0.006 kg/m-s
Flow rate in	176 kg/s
Flow rate out	113 kg/s
Net balance	63 kg/s

Table 3.3. Densities of different particles.

Particle	**Density (kg/m^3)**
Argon	≃ 0
Alumina	2700
Slag	5000

Table 3.4. Groups of small spherical particles simulated in nozzle and mold.

Number of Particles	Diameter (μm)	Density (kg/m^3)
5000	10	5000
5000	40	5000
5000	10	2700
5000	40	2700

Table 3.5. Groups of large spherical particles used in mold.

Number of Particles	Diameter (μm)	Density (kg/m^3)
5000	100	2700
5000	250	2700
5000	400	2700
5000	100	≃ 0
5000	250	≃ 0
5000	400	≃ 0
5000	100	5000
5000	250	5000
5000	400	5000

Table 3.6. Percentage of particles removed by the top surface by $(k-\varepsilon)$ model.

Diameter (μm)	Density (kg/m^3)	% of particles removed by top surface
100	≃ 0	4.86
250	≃ 0	16.50
400	≃ 0	41.00
100	2700	4.62
250	2700	11.20
400	2700	25.50
100	5000	3.84
250	5000	6.54
400	5000	12.84

Table 3.7. Percentage of particles removed by the top surface by LES model [2].

Diameter (μm)	Density (kg/m^3)	% of particles removed by top surface
100	2700	12.58
250	2700	42.5
400	2700	69.89

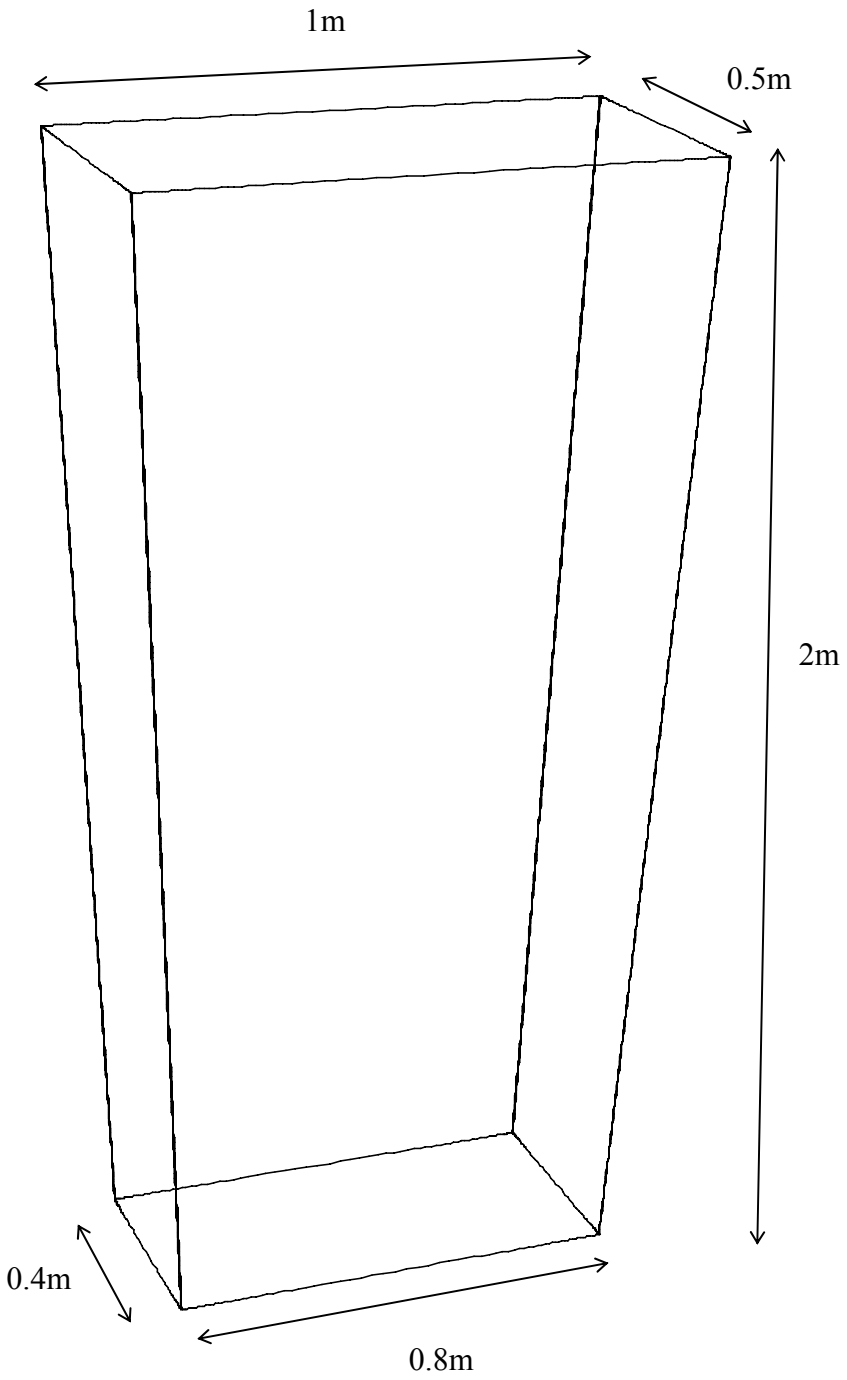


Fig 3.1. Geometry of test mold.

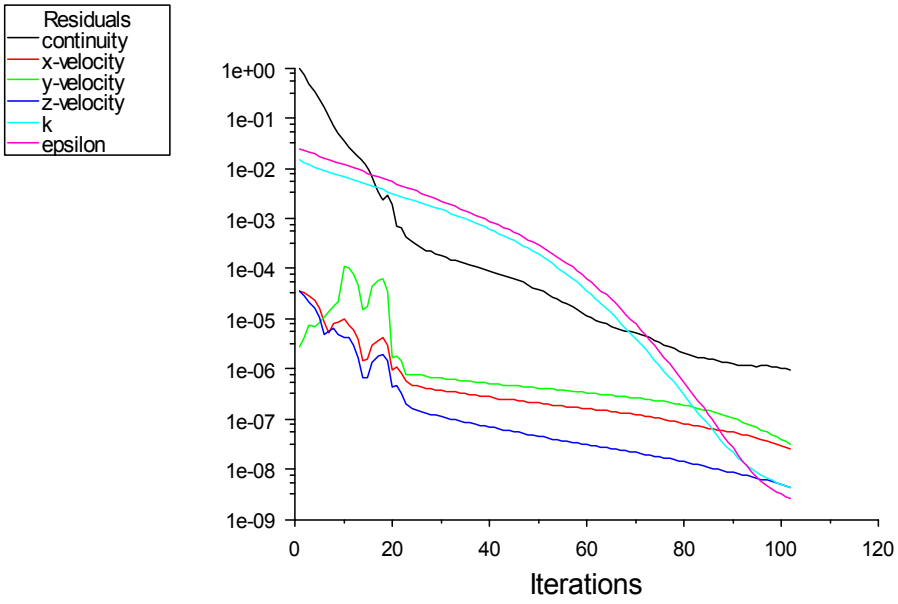


Fig 3.2. Plot of scaled residual error for the test mold.

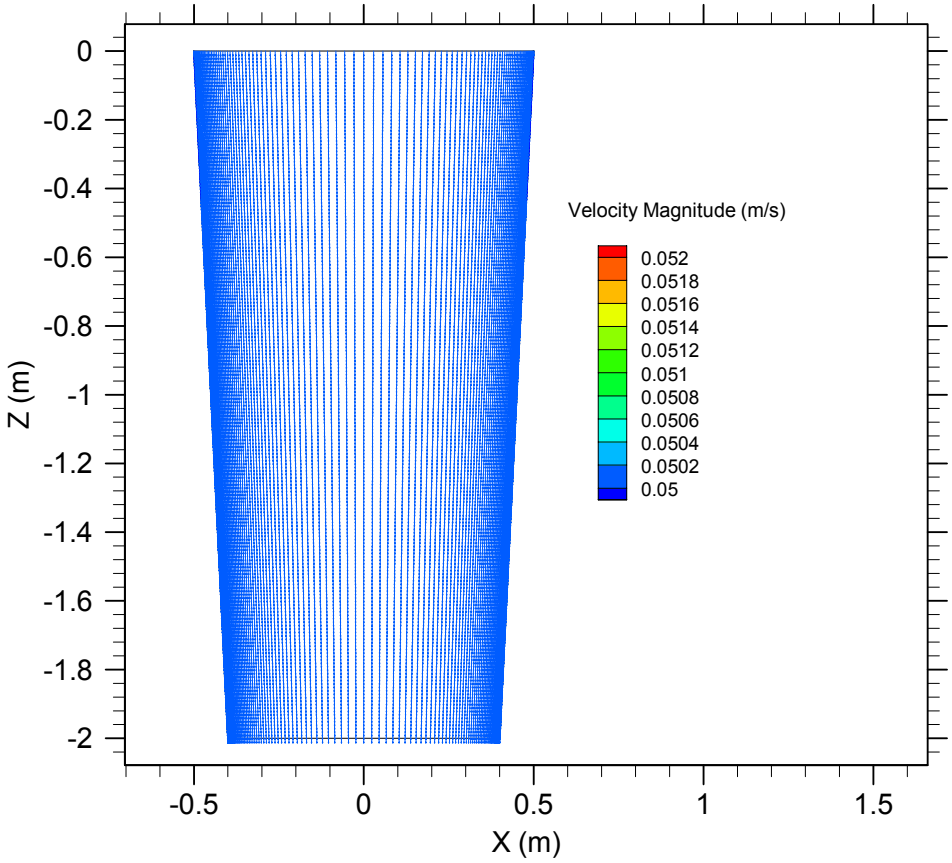


Fig 3.3. Velocity vectors on the plane mid way between wide faces of test mold.

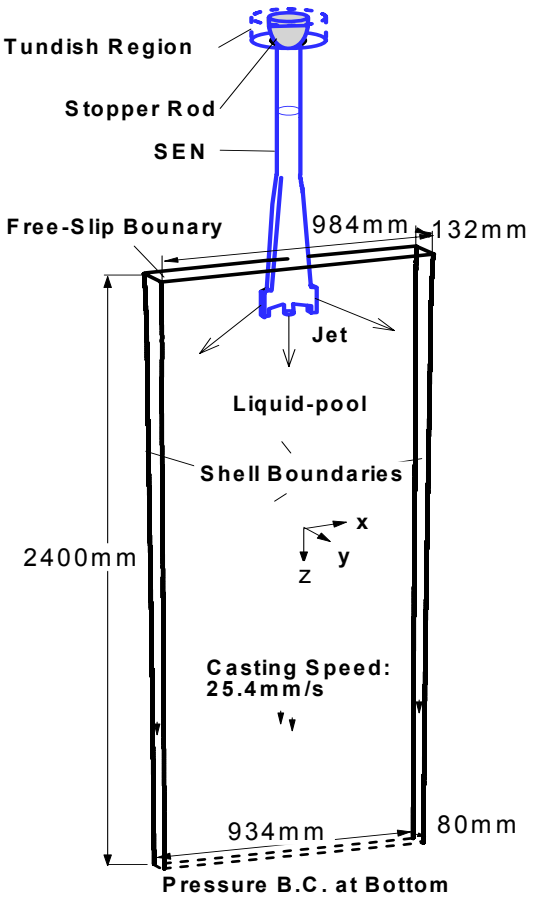


Fig 3.4. Mold domain with shell used for simulations [2].

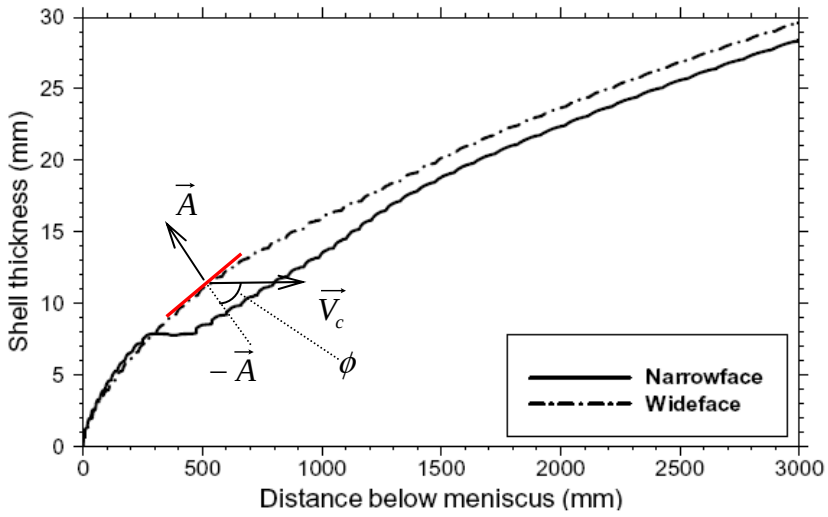


Fig 3.5. Predicted shell thickness from CON1D [15].

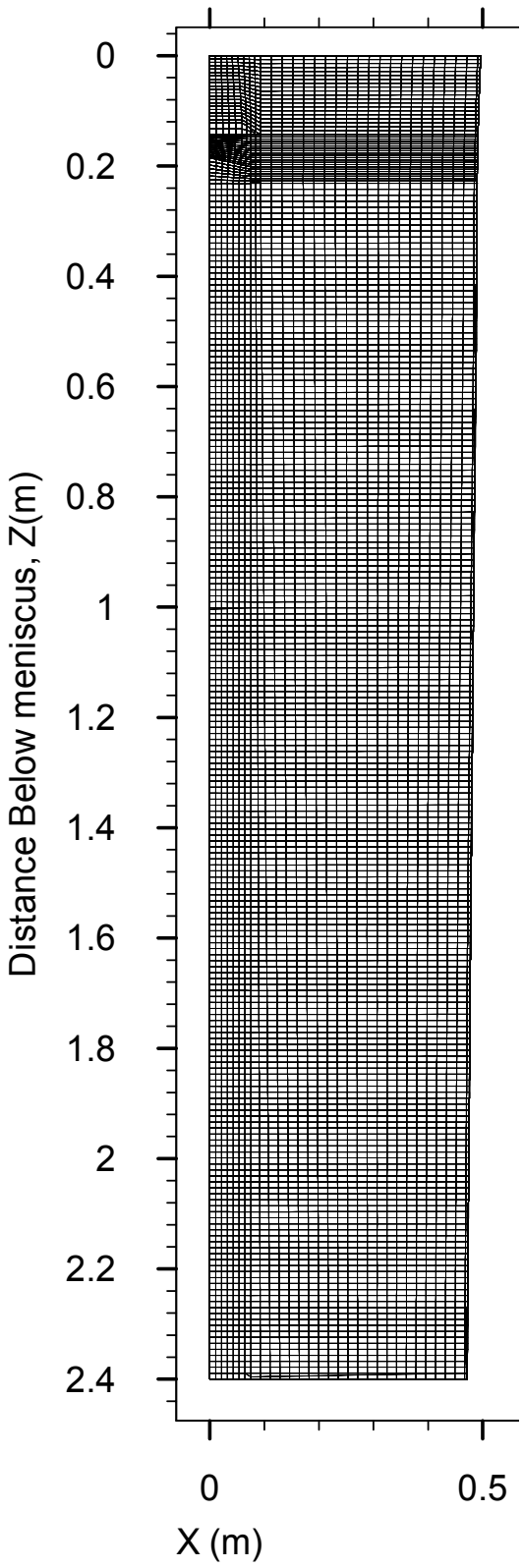


Fig 3.6. Mold mesh with shell included.

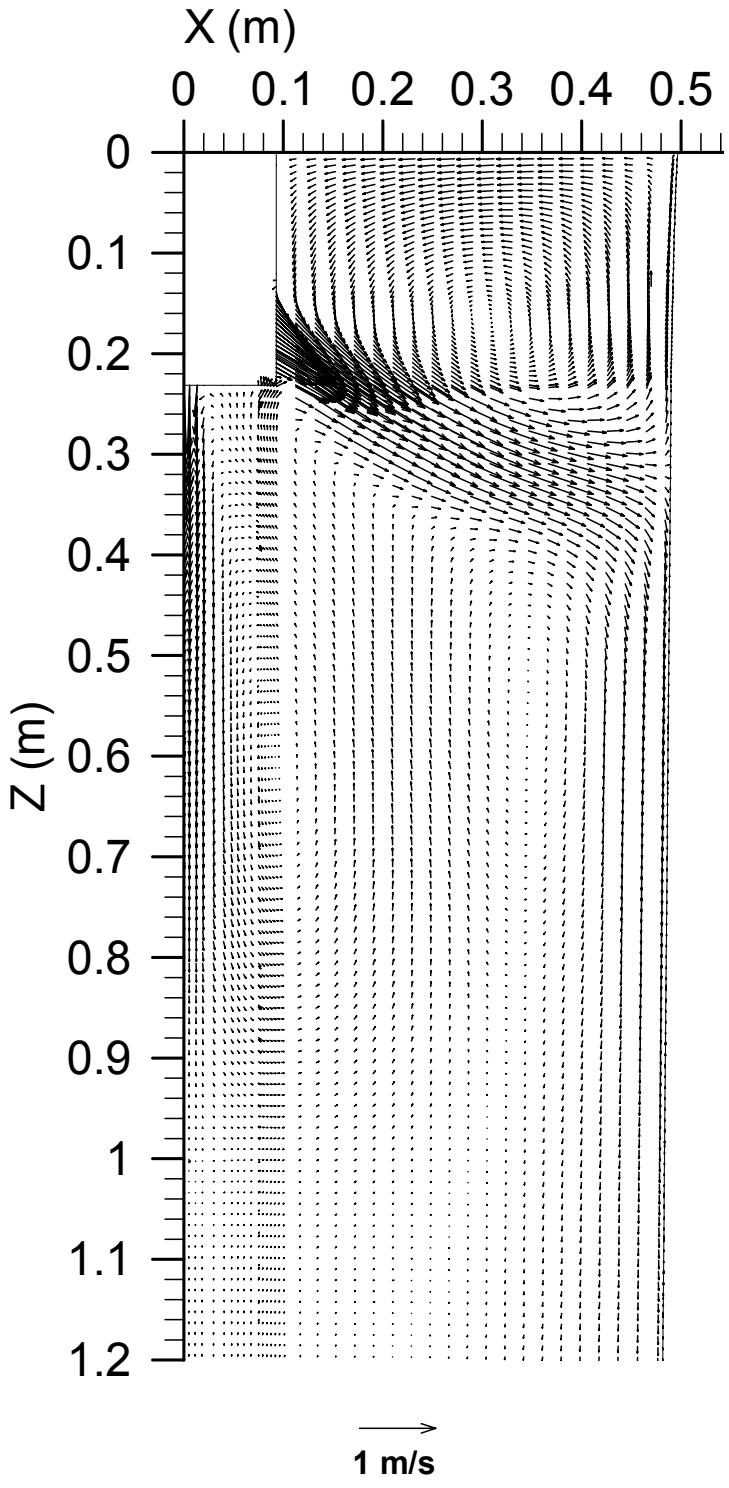


Fig 3.7. Velocity vectors on a plane mid way between wide faces using $(k-\varepsilon)$.

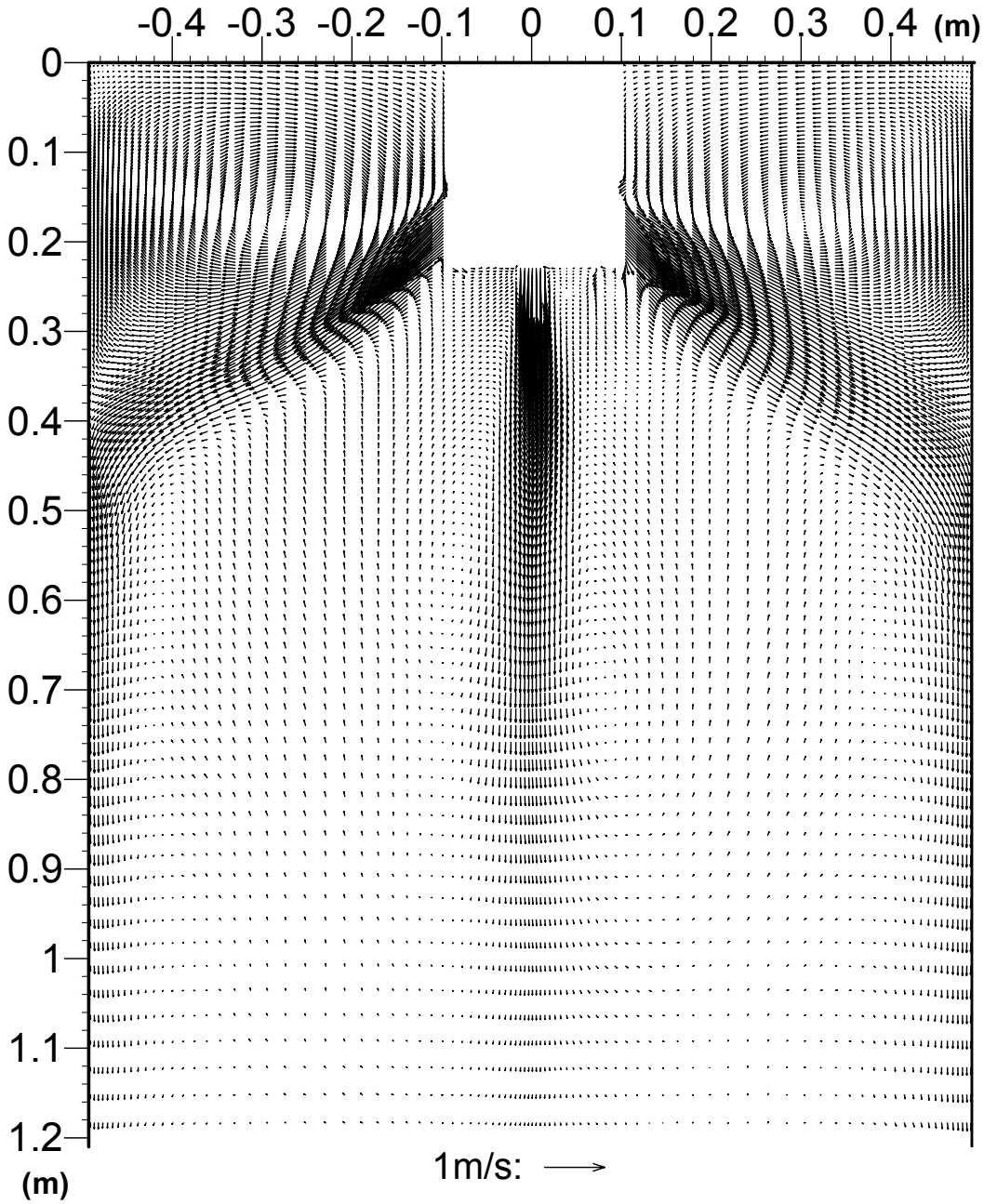


Fig 3.8. Velocity vectors on a plane mid way between wide faces using LES [14].

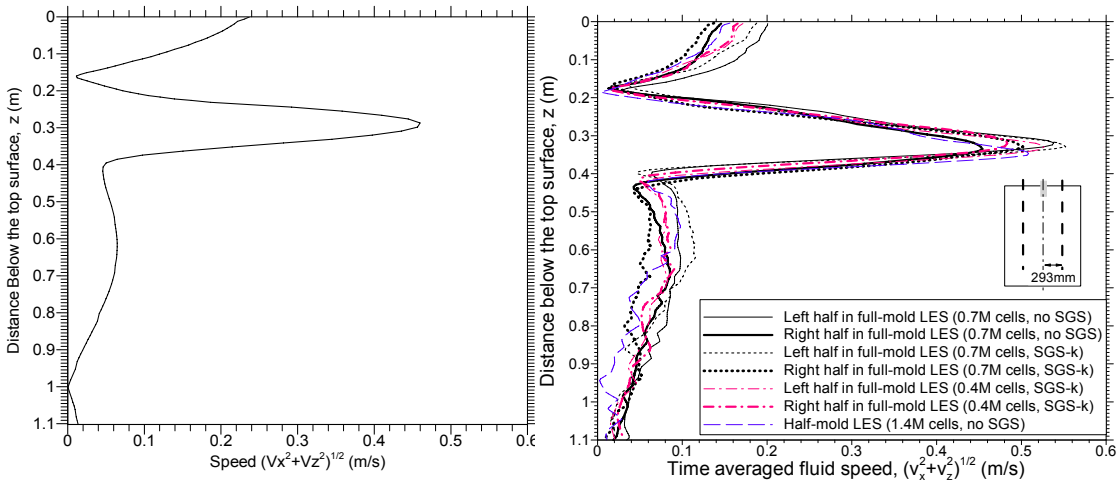


Fig 3.9. Velocity down the mold on a line 293mm from the center, comparing (a) this work with (b) previous work [14].

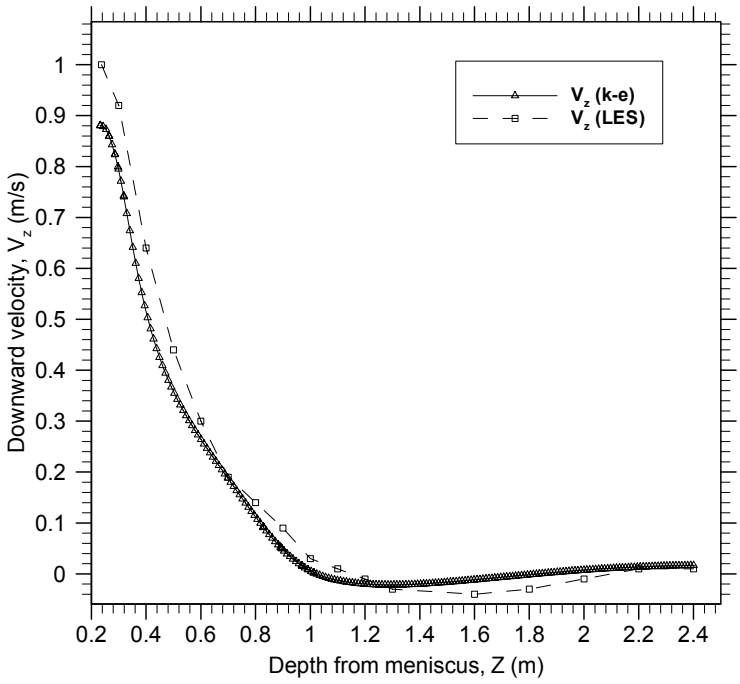


Fig 3.10. Velocity along center jet center line using LES [14] and $(k-\varepsilon)$ model.

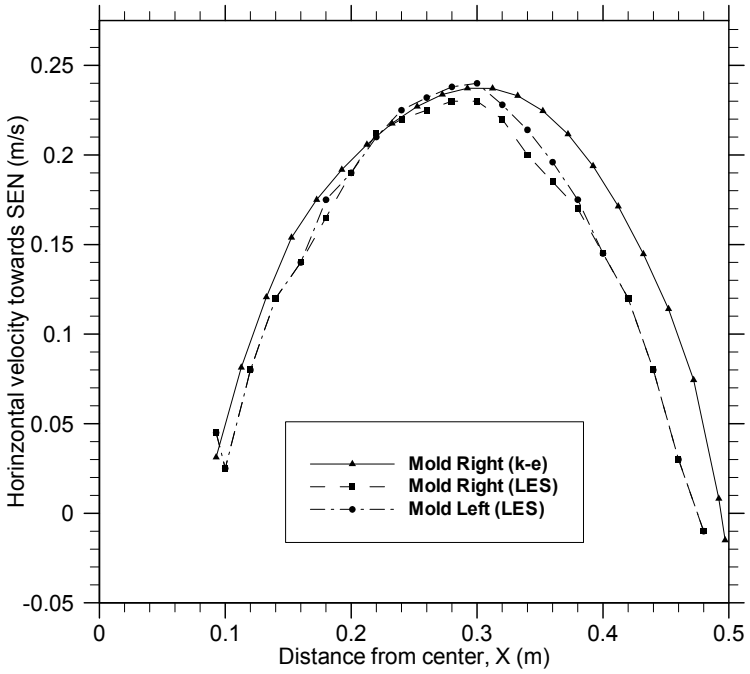


Fig 3. 11. Horizontal velocity towards SEN using LES [14] and $(k-\varepsilon)$ model.

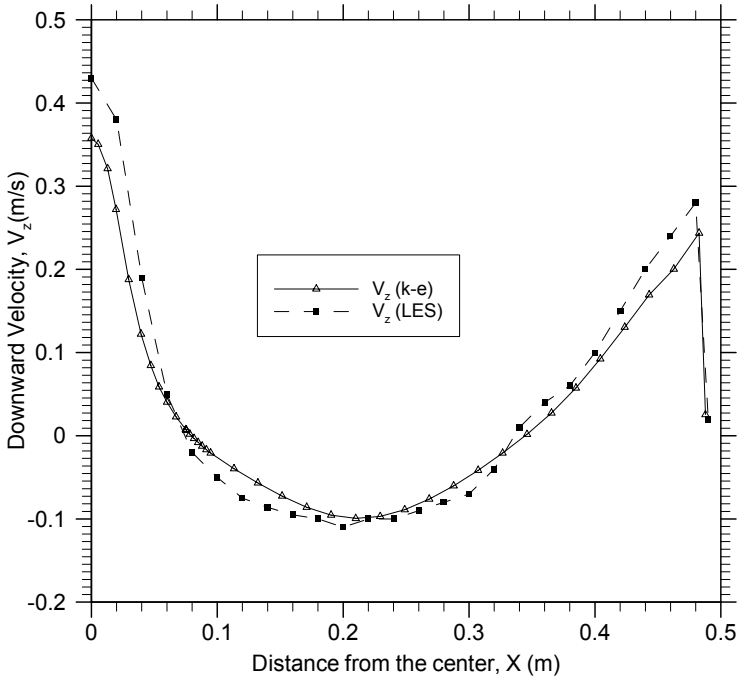


Fig 3.12. Velocity along a horizontal line 0.5 m below meniscus using LES [14] and $(k-\varepsilon)$ model.

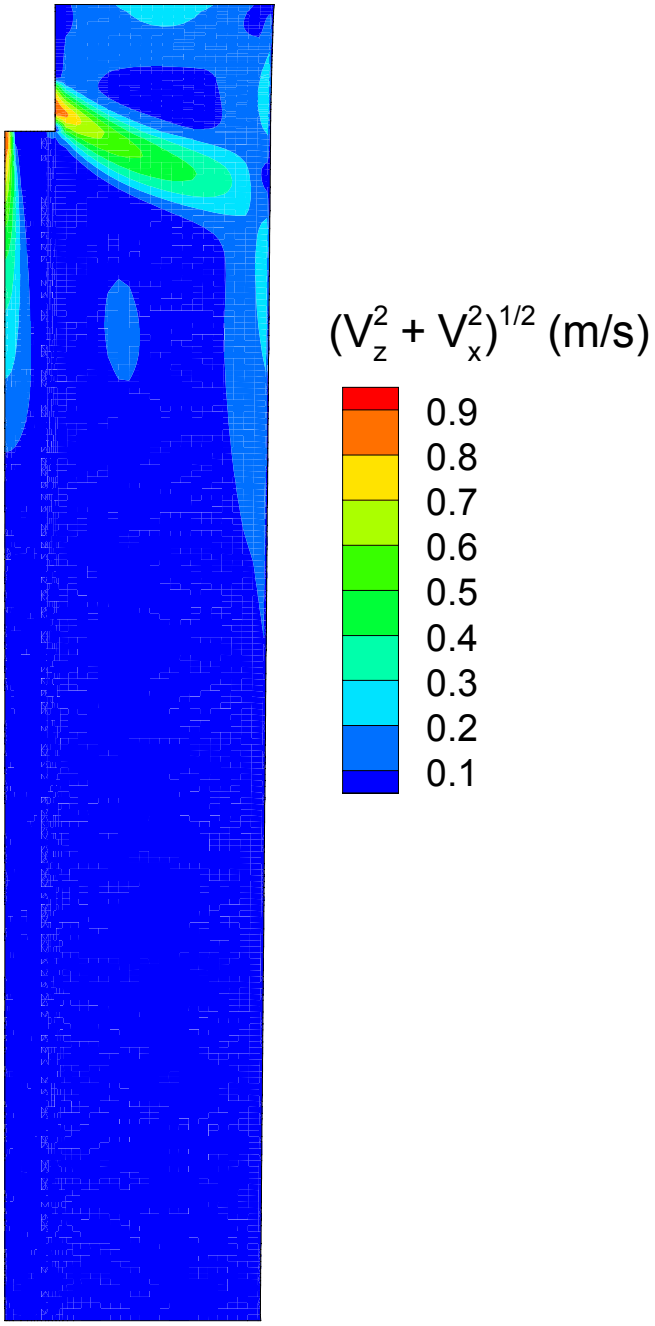


Fig 3.13. Velocity contours on a plane mid way between wide faces.

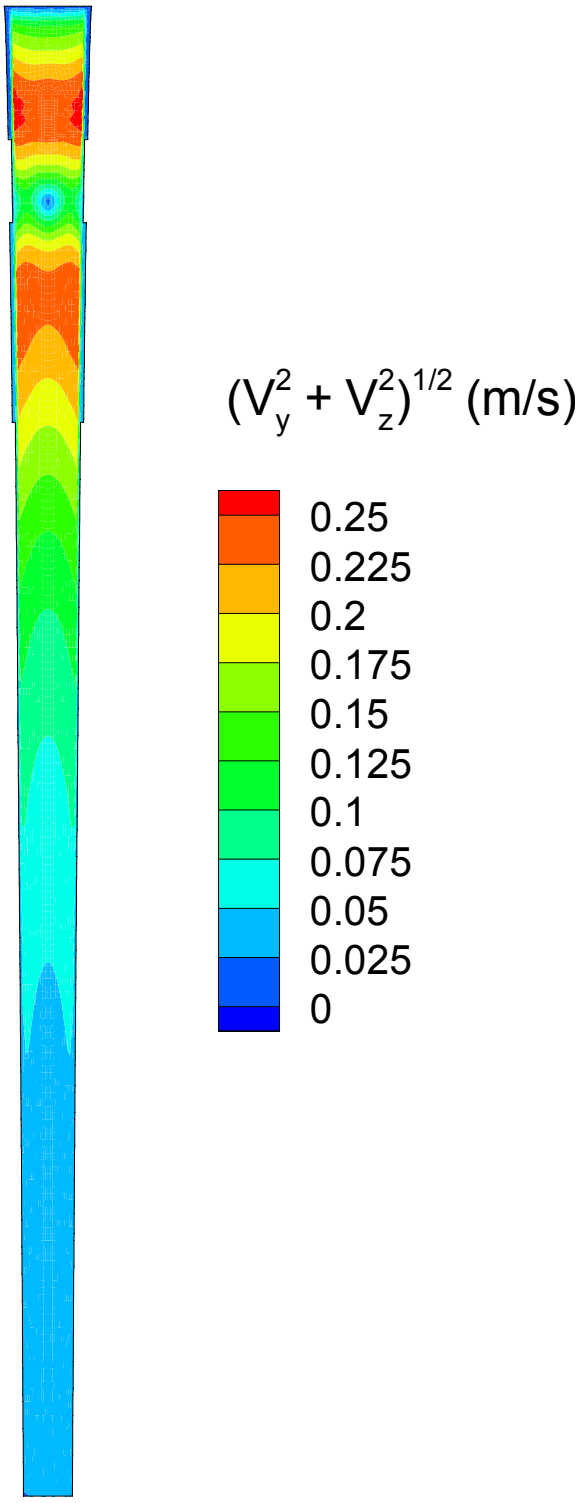


Fig 3.14. Velocity contours on a plane approximately 10mm from narrow face.

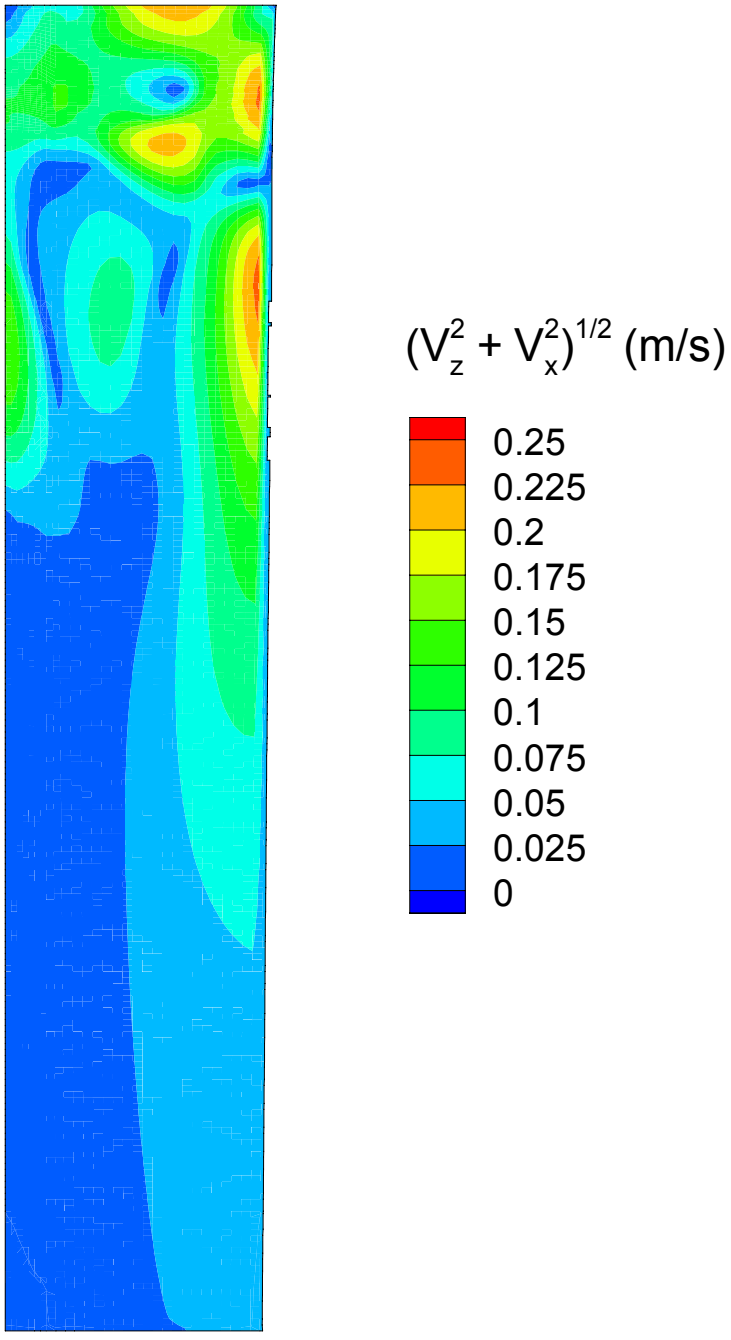
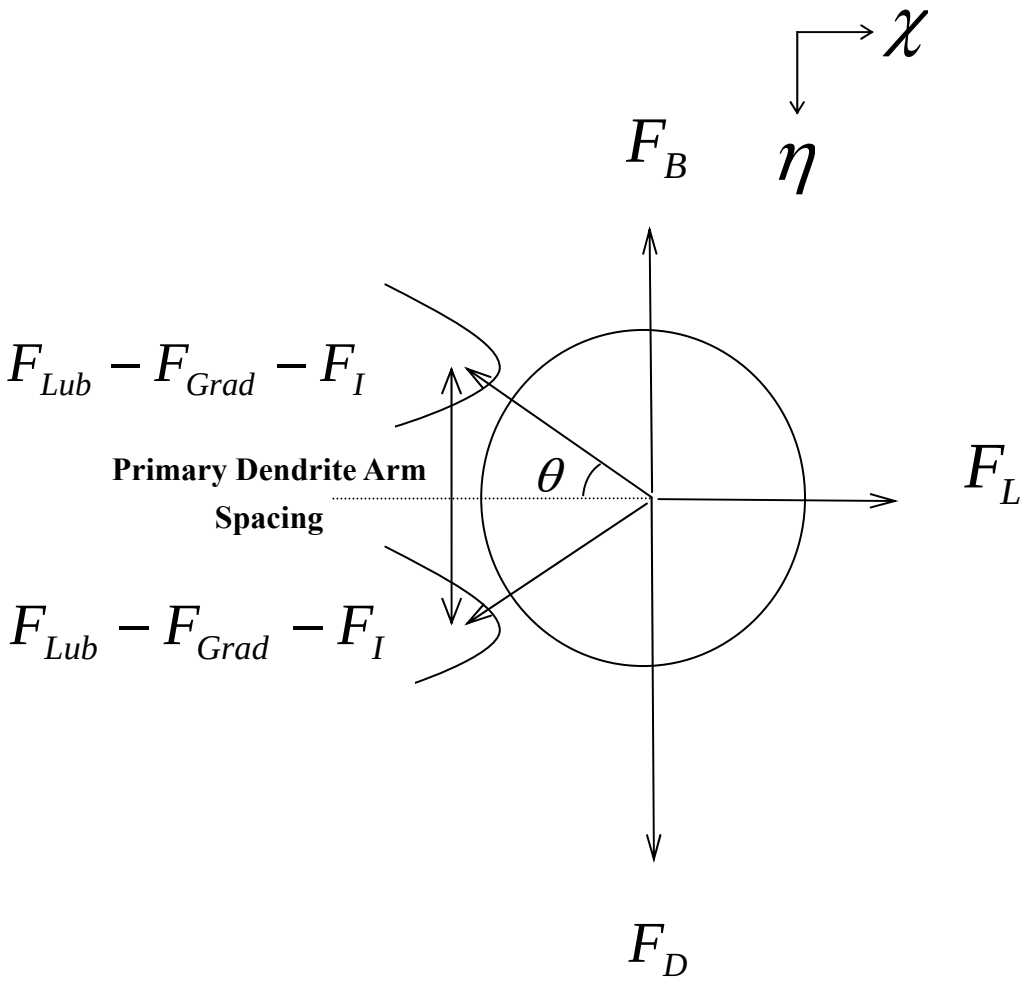


Fig 3.15. Velocity contours on a plane approximately 10mm from wide face.



F_D = Drag force

F_B = Buoyancy force

F_{Lub} = Lubrication force

F_{Grad} = Surface Gradient force

F_I = Van der Waals Interfacial force

Fig 3.16. Force balance at vertical solidification front.

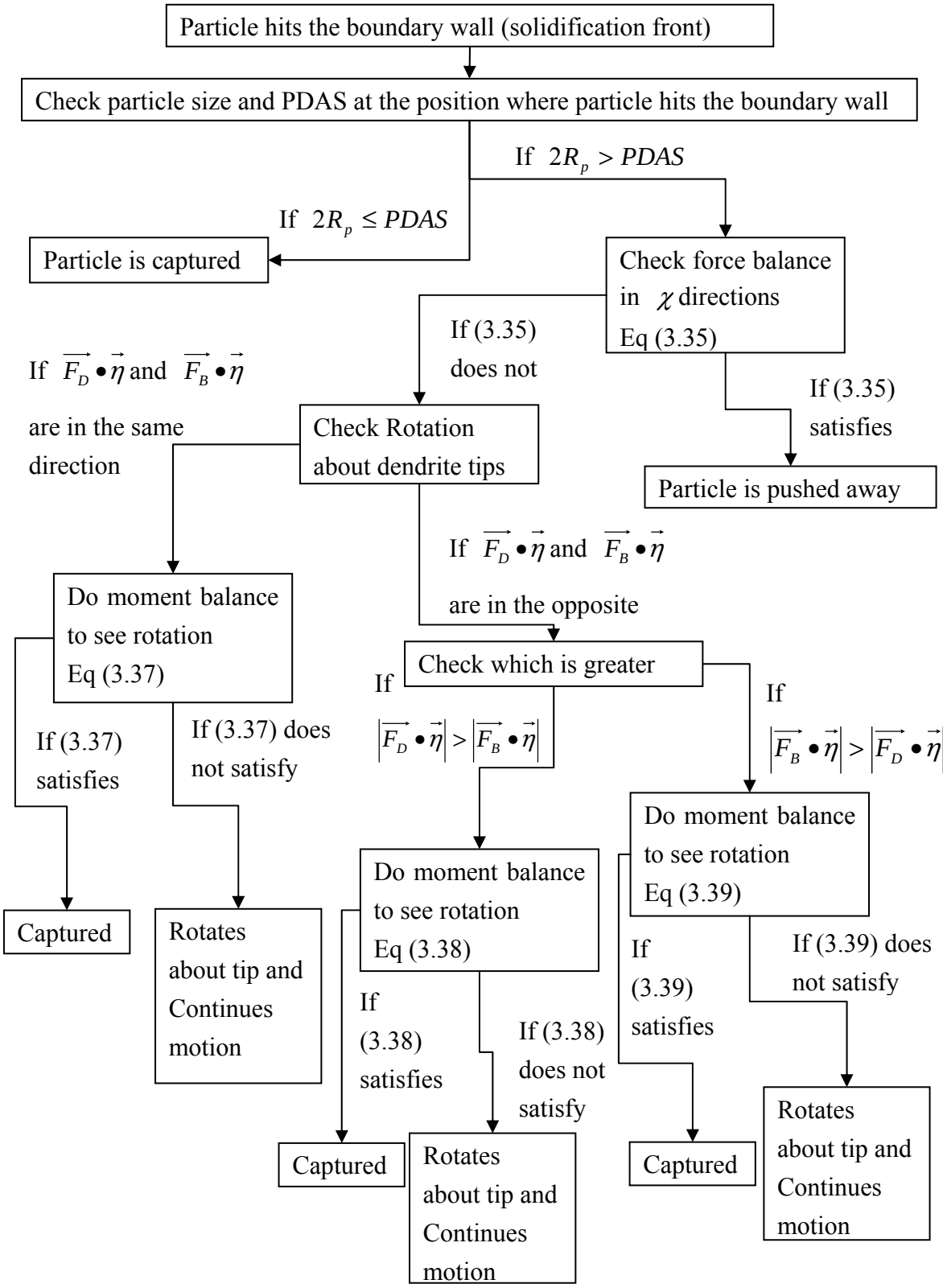


Fig 3.17. Flow Chart for Capture Criterion.

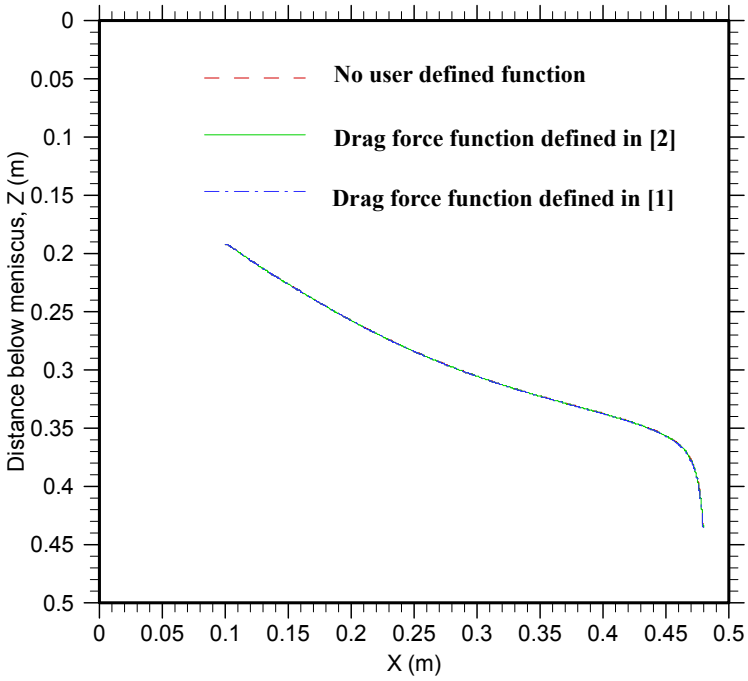


Fig 3.18. Trajectory with different drag force functions (0.4mm dia, ρ= 5000kg/m^3).

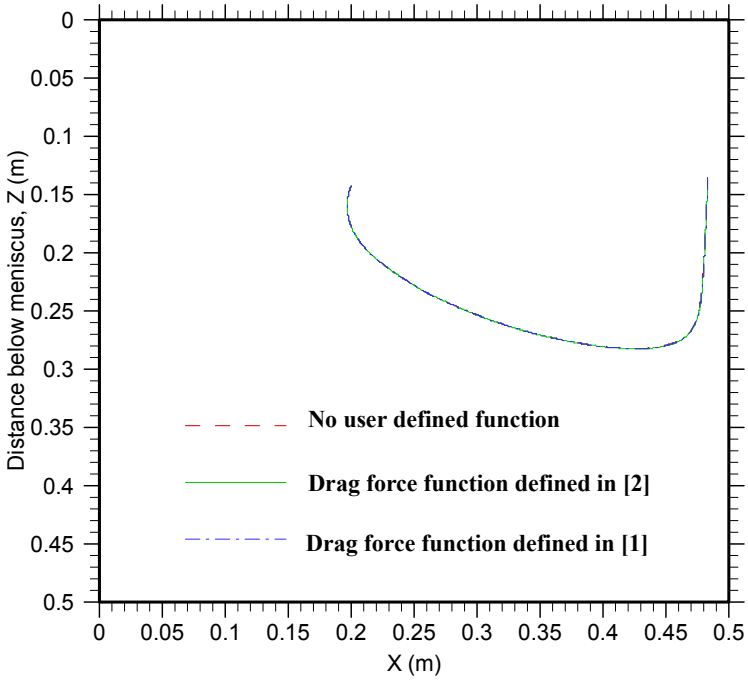


Fig 3.19. Trajectory with different drag force functions (0.1mm dia, ρ=2700kg/m^3).

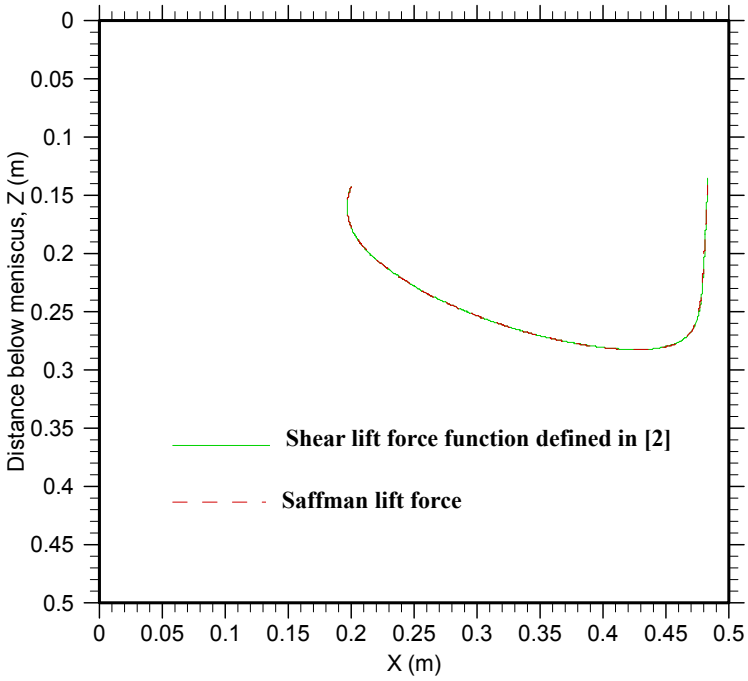


Fig 3.20. Trajectory with different shear lift force (0.1mm dia, ρ=2700kg/m^3).

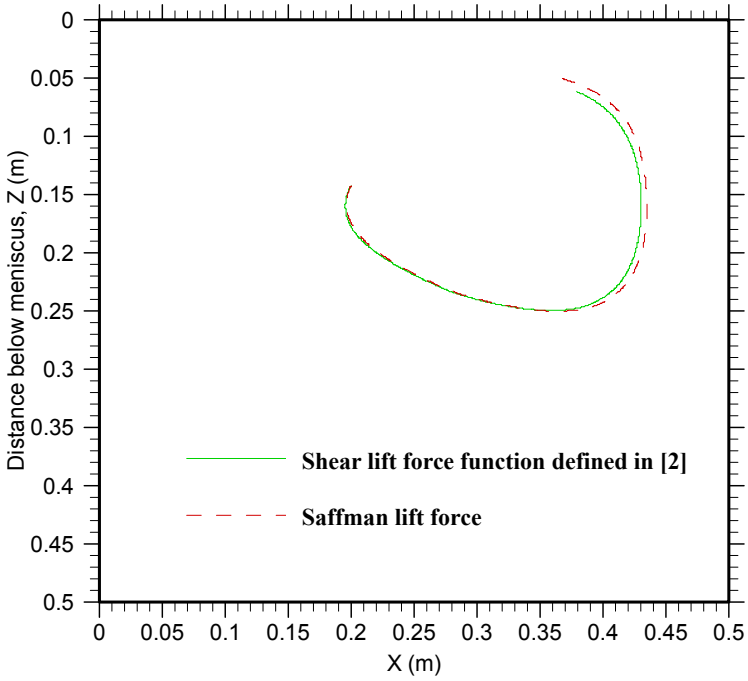


Fig 3. 21. Trajectory with different shear lift force (0.4mm dia, ρ=2700kg/m^3).

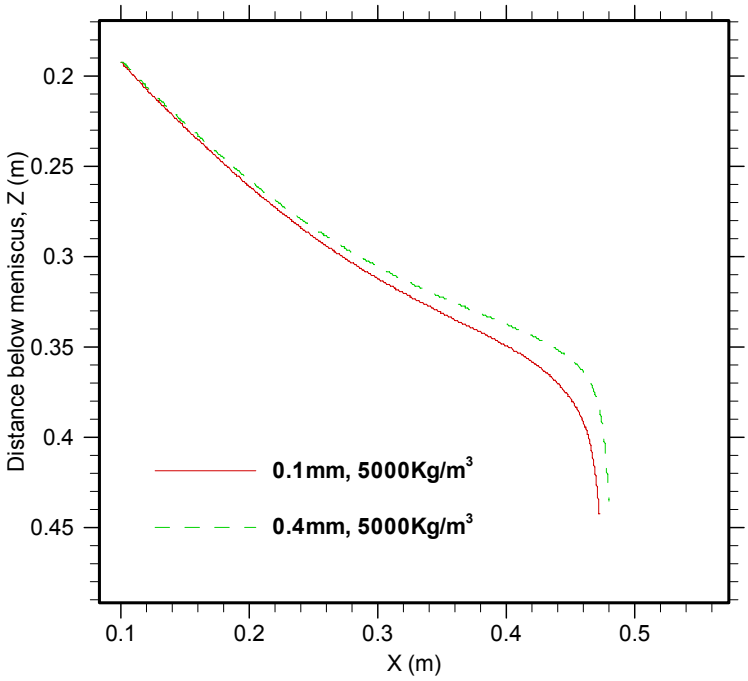


Fig 3.22. Effect on particle trajectory with varying particle diameter.

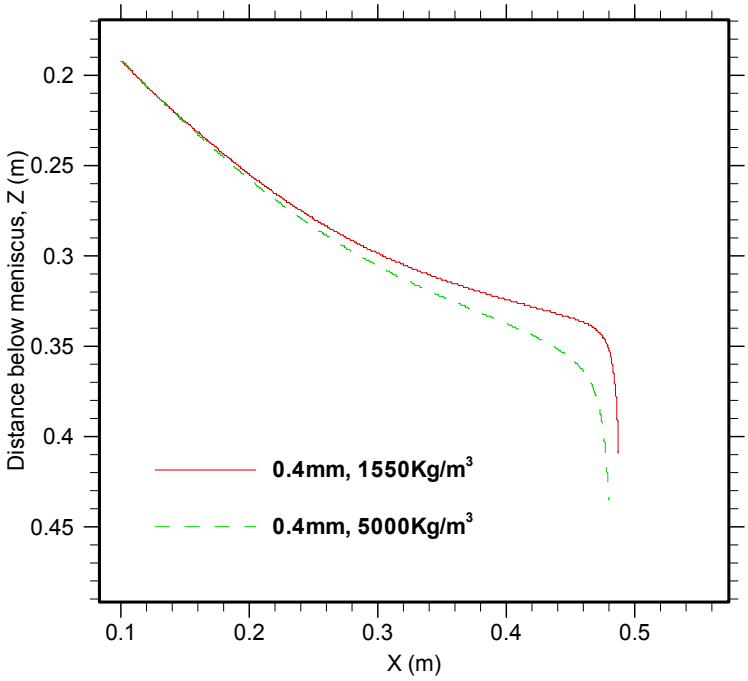


Fig 3.23. Effect on particle trajectory with varying particle density.

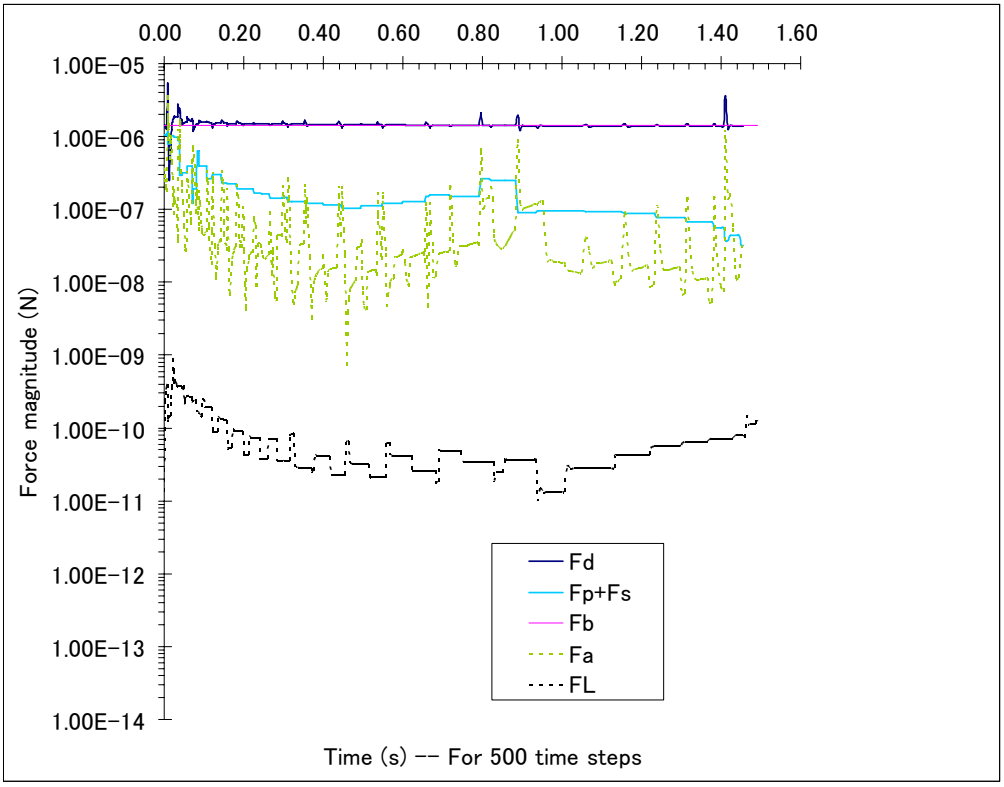


Fig 3.24. Magnitude of hydrodynamic forces $(k-\varepsilon)$ (0.4mm dia, ρ=2700kg/m^3).

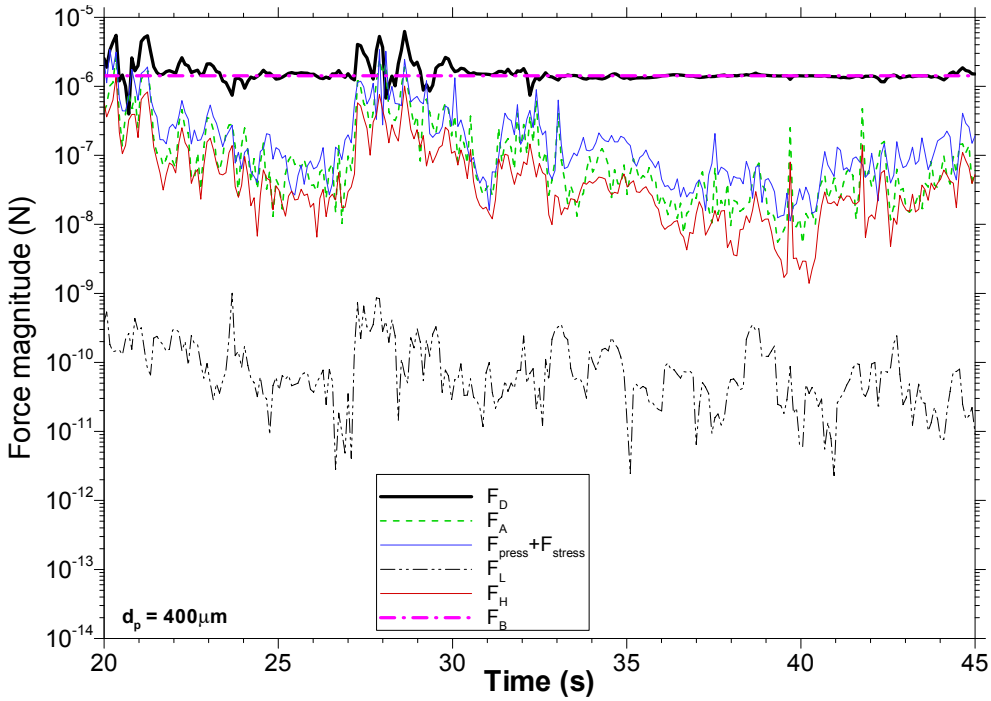


Fig 3.25. Magnitude of hydrodynamic forces [2] (0.4mm dia, ρ=2700kg/m^3).

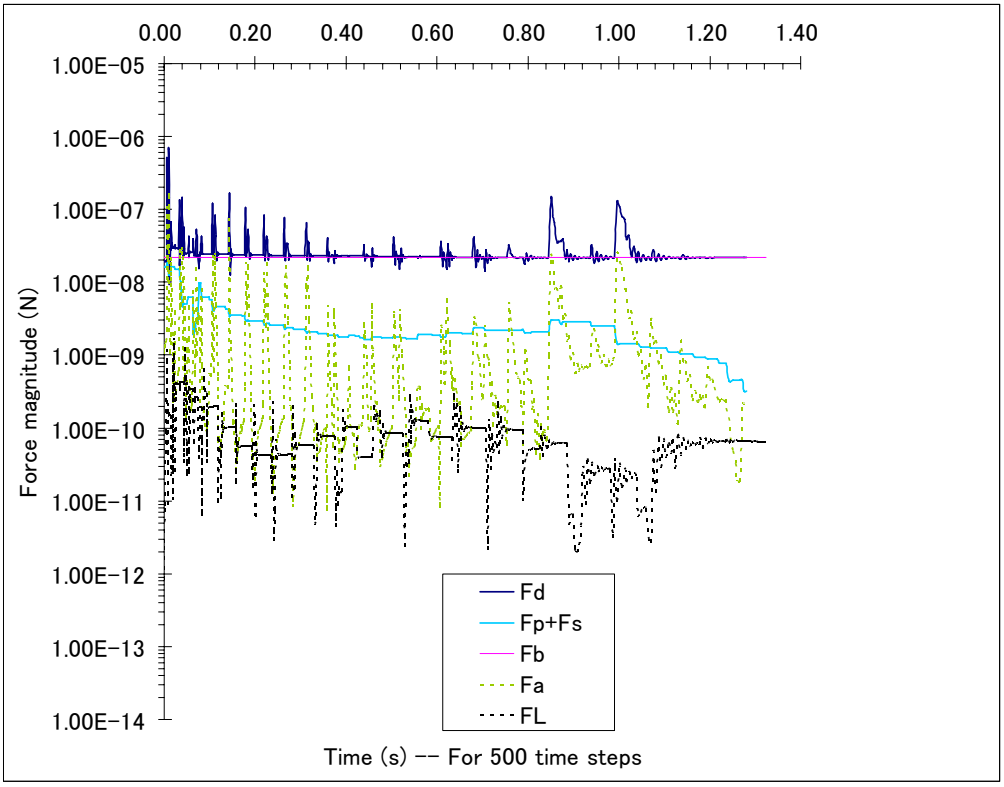


Fig 3.26. Magnitude of hydrodynamic forces $(k-\varepsilon)$ (0.1mm dia, ρ=2700kg/m^3).

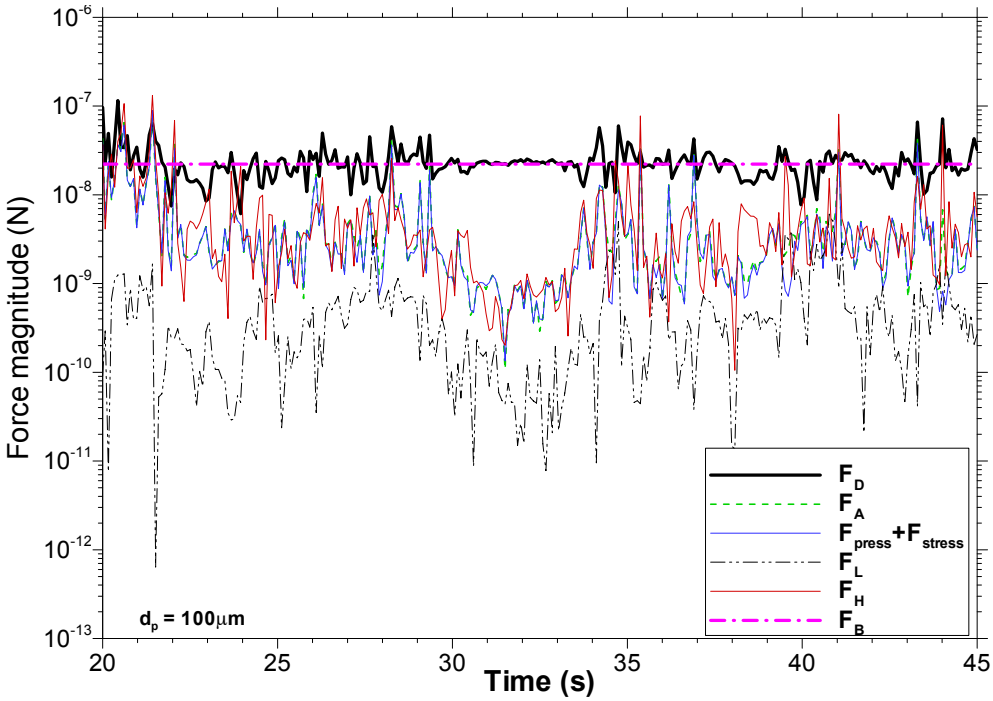


Fig 3.27. Magnitude of hydrodynamic forces [2] (0.1mm dia, ρ=2700kg/m^3).

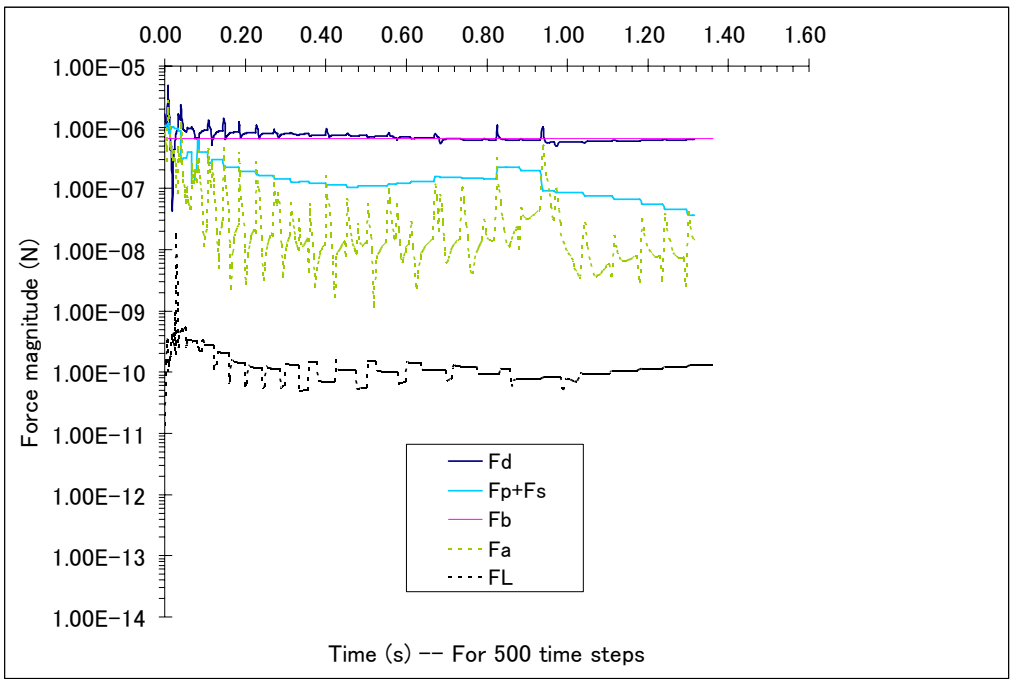


Fig 3.28. Magnitude of hydrodynamic forces $(k-\varepsilon)$ (0.4mm dia, ρ= 5000kg/m^3).

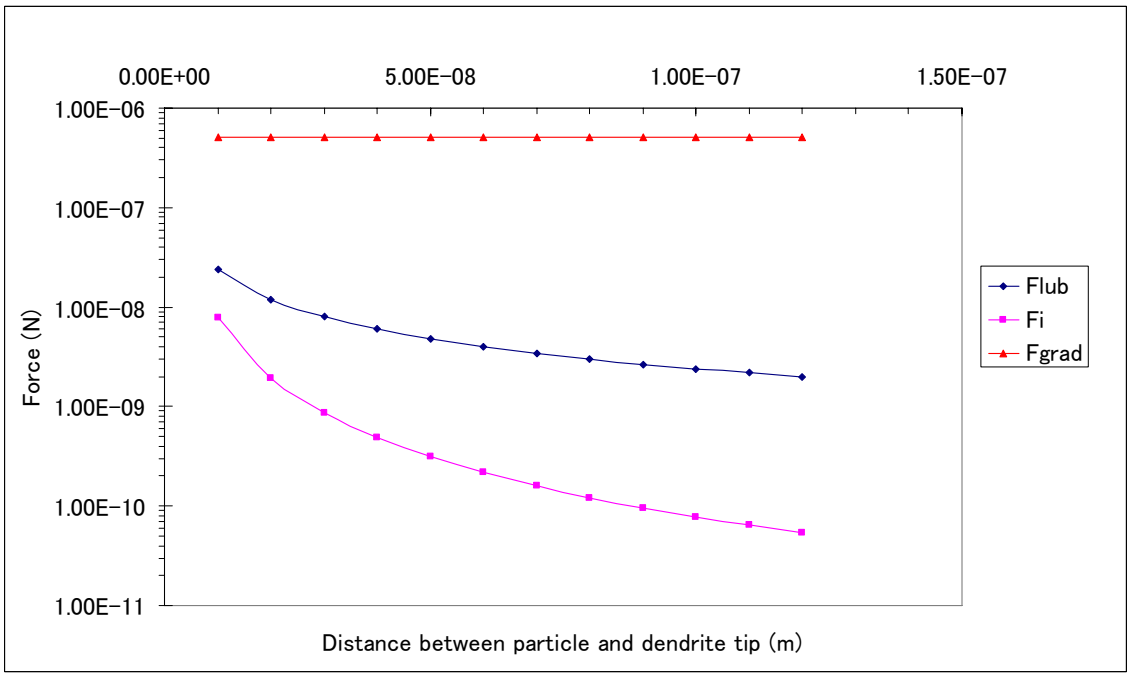


Fig 3.29. Magnitude of lubrication, interfacial and gradient force.

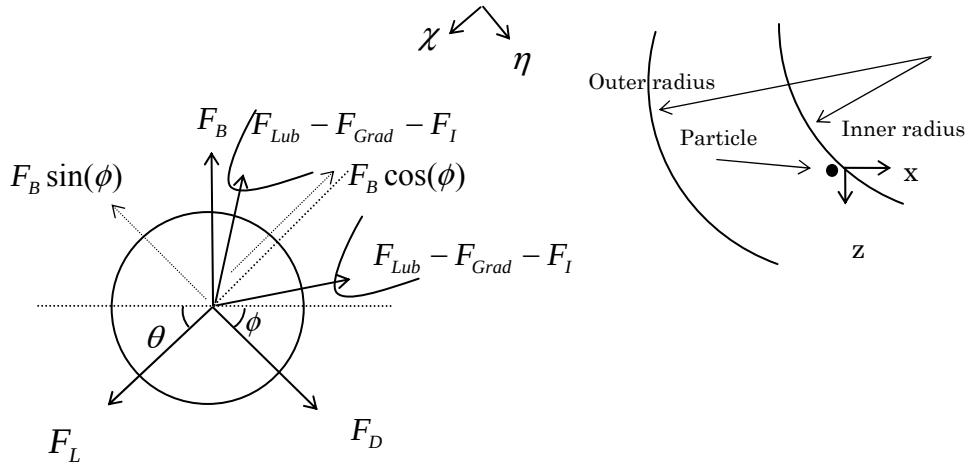


Fig 3.30. Force balance on particle at the inner radius on curved section.

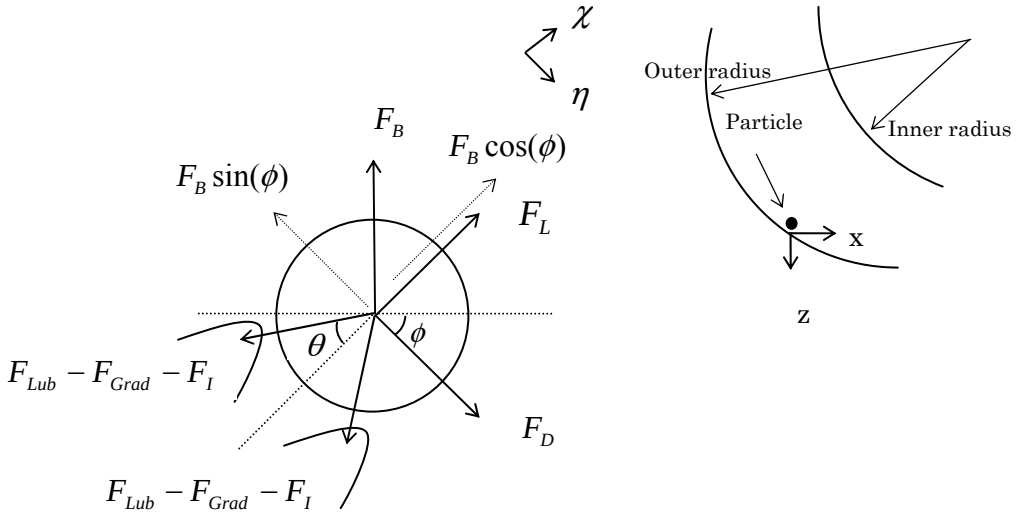
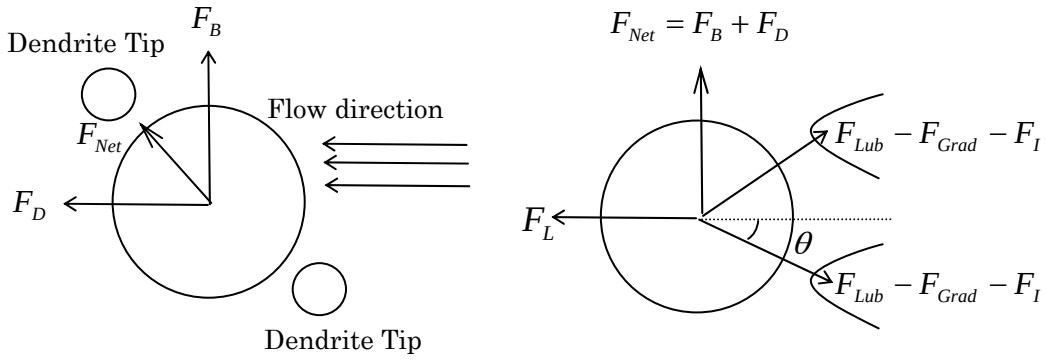


Fig 3.31. Force balance for particle near the outer radius on curved section.



Orientation 1 **Orientation 2**

Fig 3.32. Particle between dendrite arms with horizontal flow.

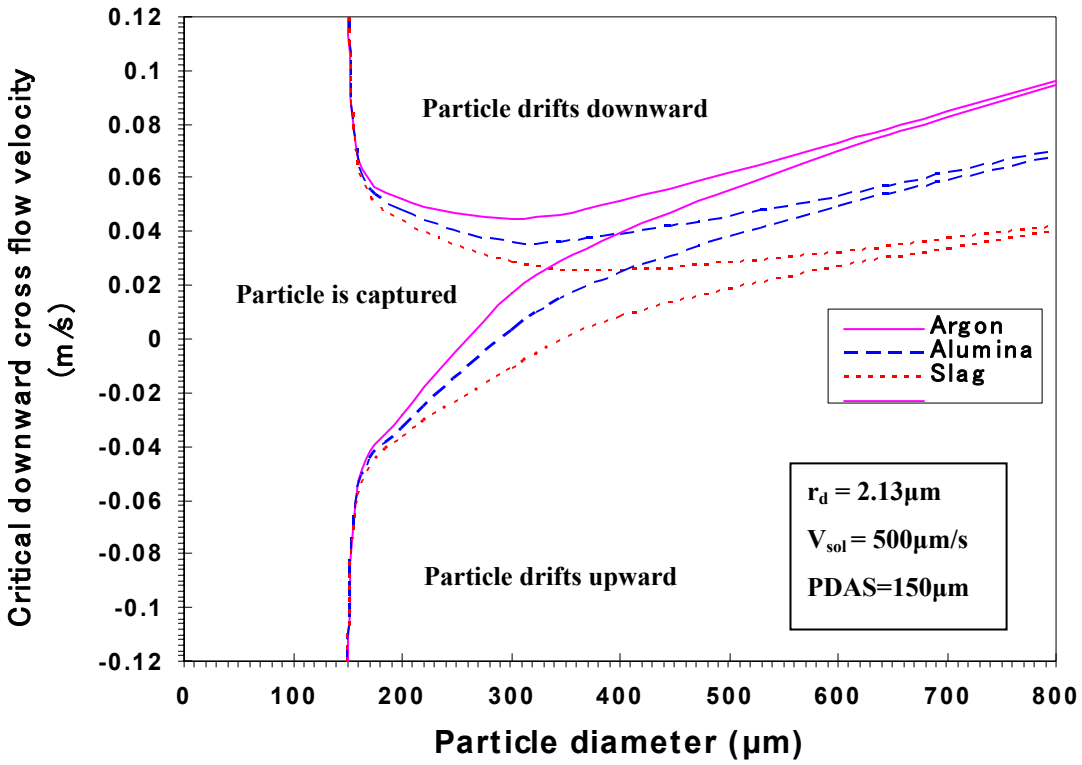


Fig 3. 33. Particle Comparison effect on Particle Capture.

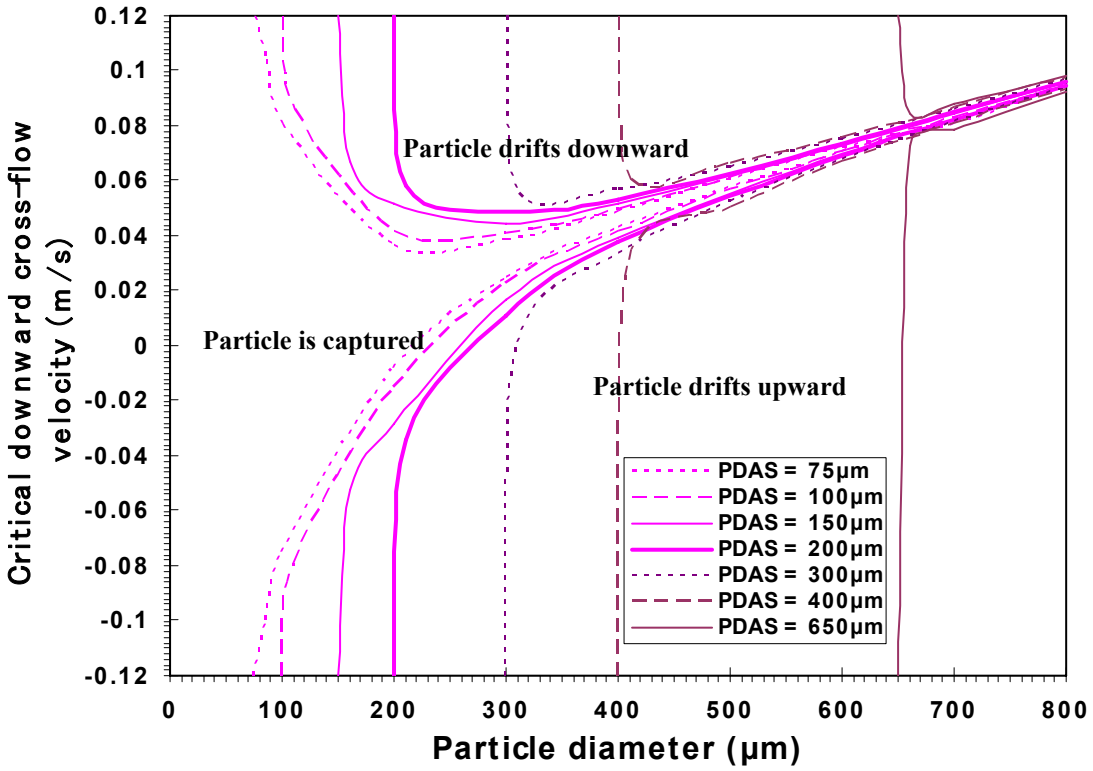


Fig 3.34. Effect on Argon bubble for various PDAS (rd=2.13μm, Vsol=500μm/s).

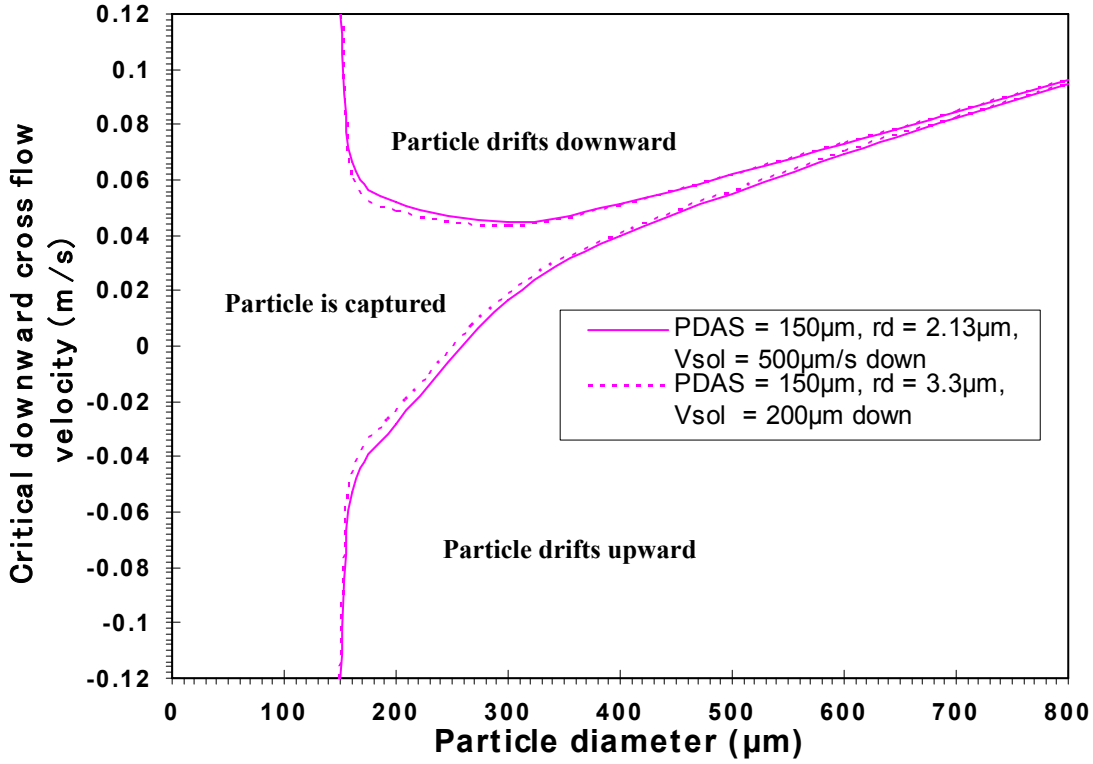


Fig 3.35. Comparison between two different conditions for Argon gas bubble.

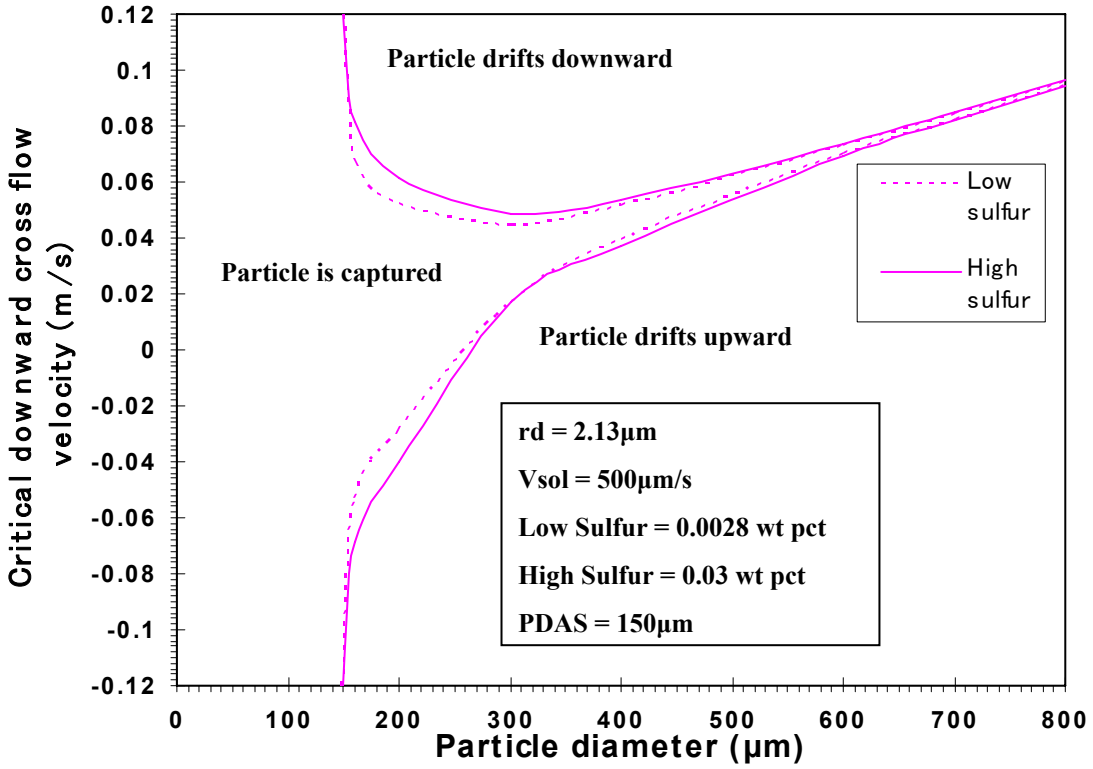


Fig 3.36. Behavior of Argon bubble in low and high sulfur content (PDAS=150µm).

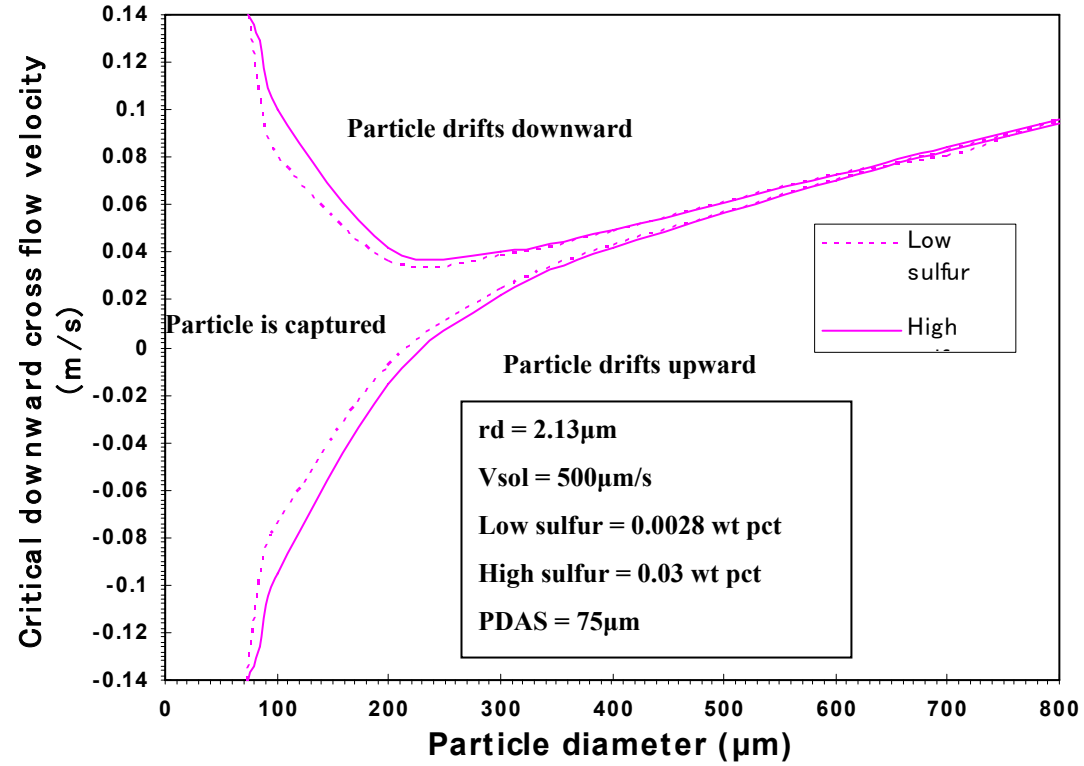


Fig 3.37. Behavior of Argon bubble in low and high sulfur content (PDAS=75µm).

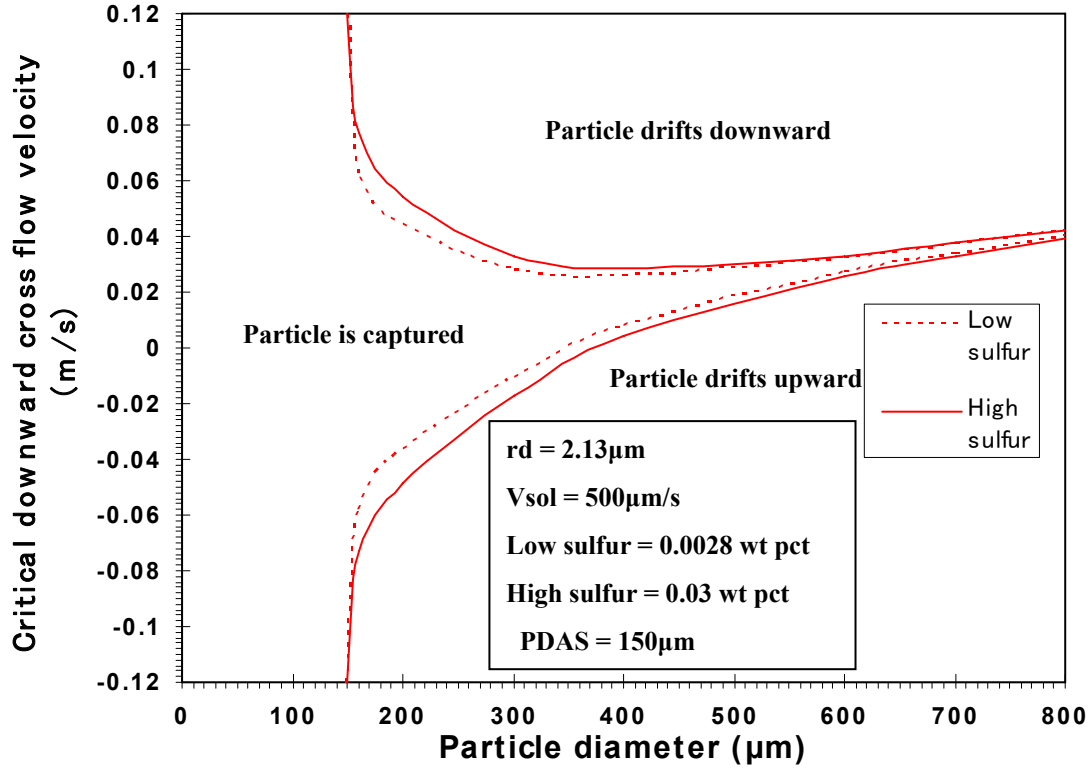


Fig 3.38. Behavior of Slag particle in low and high sulfur content (PDAS = 150um).

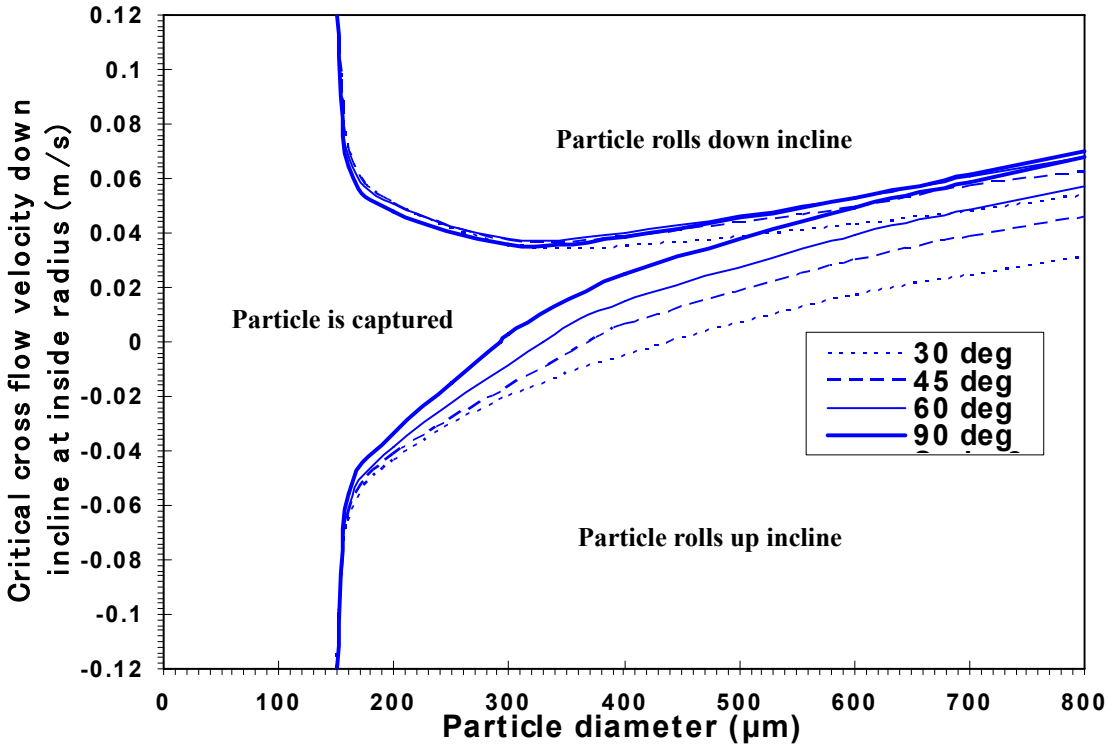


Fig 3.39. Behavior of Alumina particles for different ϕ on the inner radius.

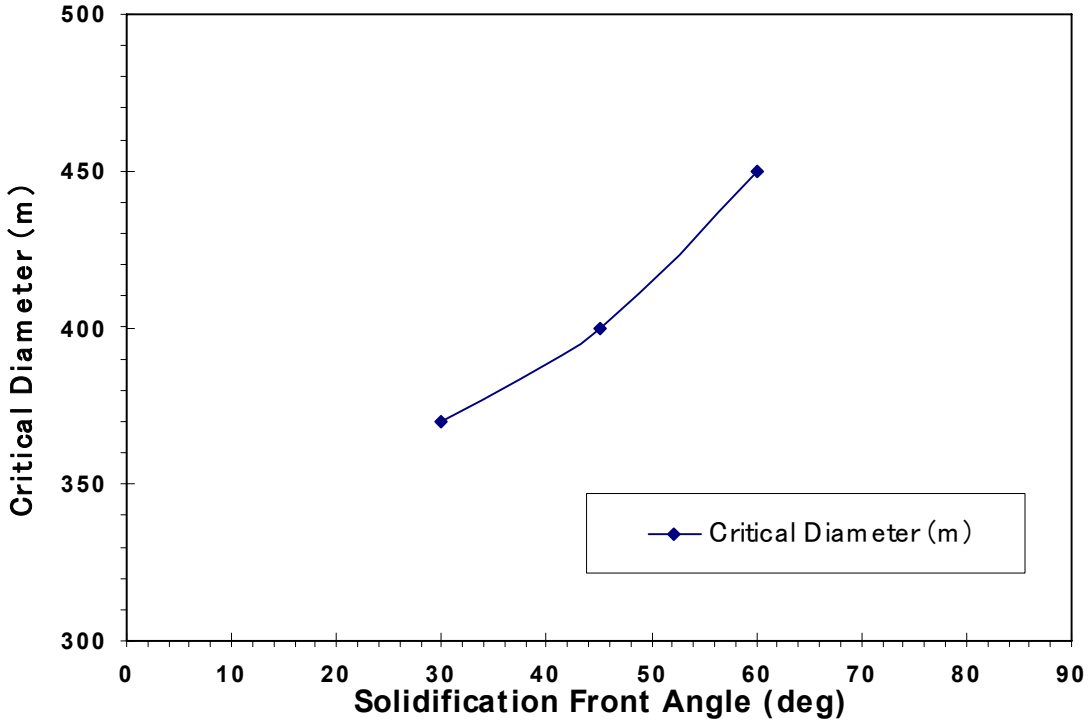


Fig 3.40. Critical diameter for which particle will escape for any cross flow velocity different ϕ at outer radius.

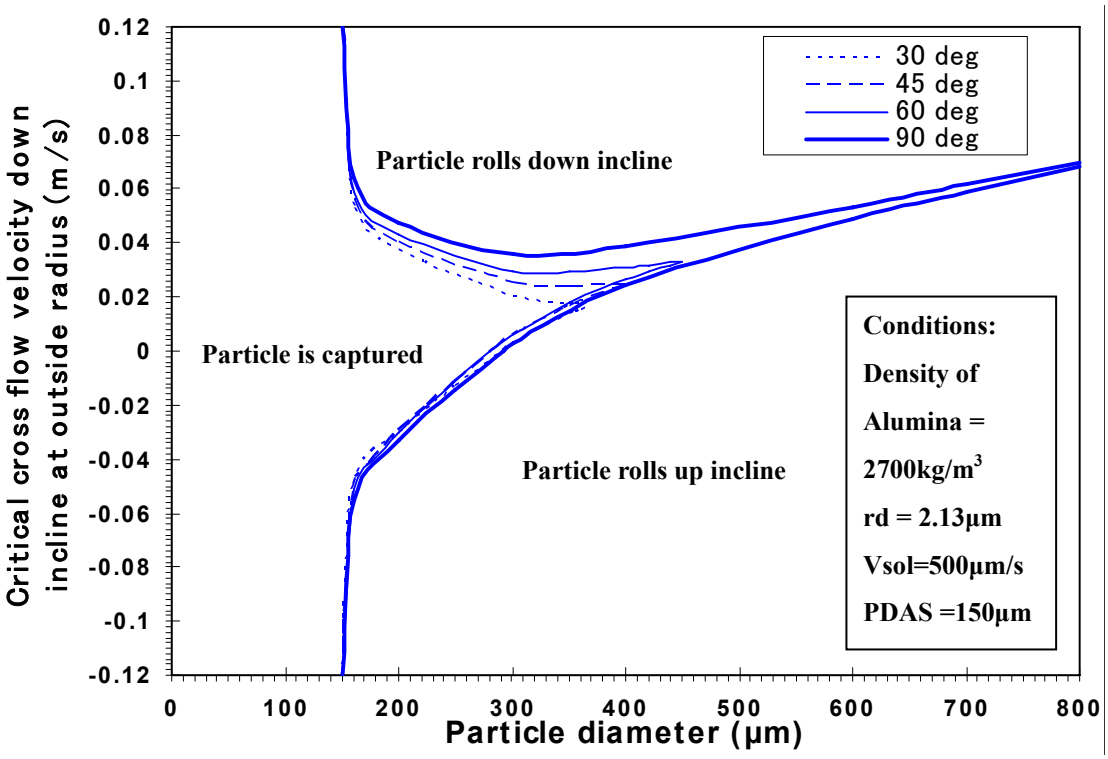


Fig 3.41. Behavior of alumina particles for different ϕ on the outer radius.

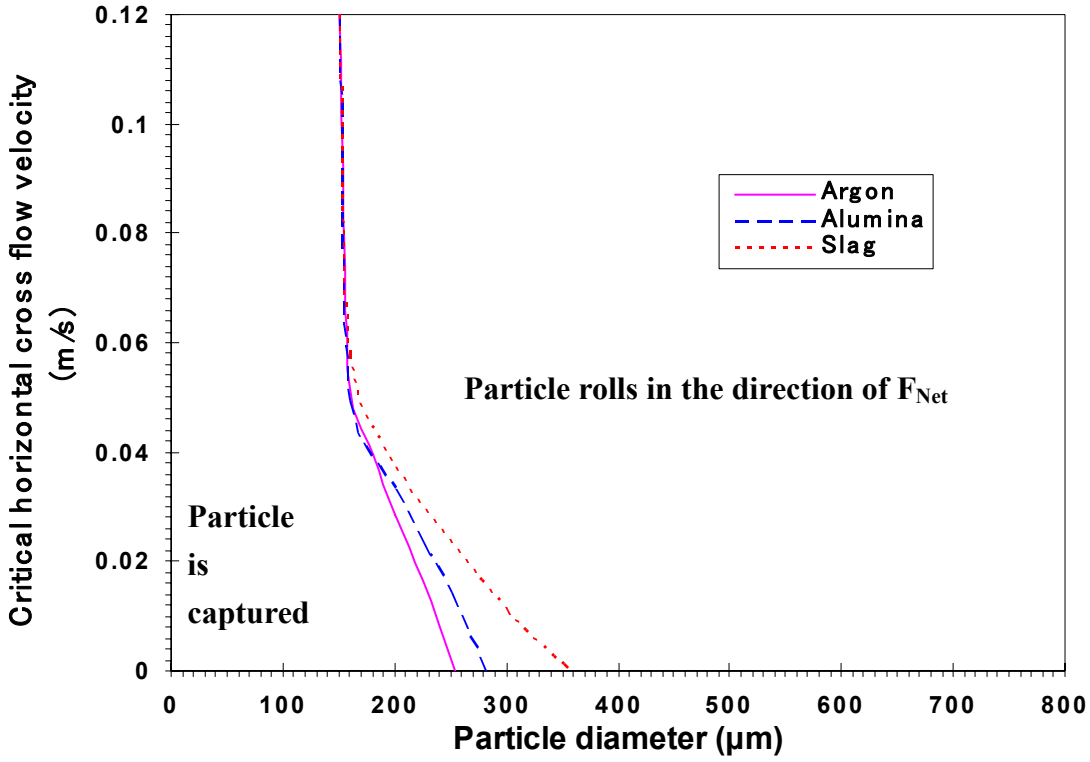


Fig 3.42. Critical Horizontal cross flow velocity for particles of different densities.

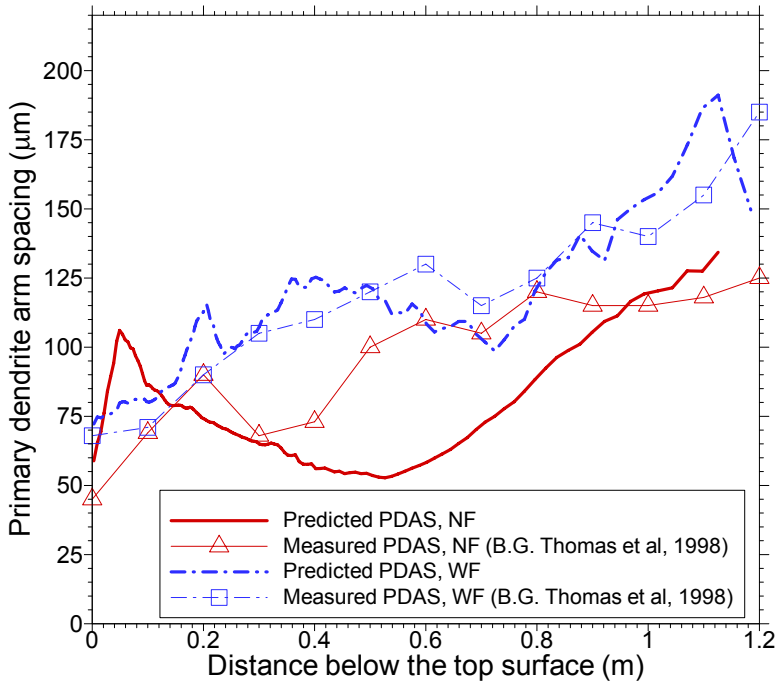


Fig 3.43. Measured [30] and predicted values [2] of primary dendrite arm spacing (PDAS) down the mold.

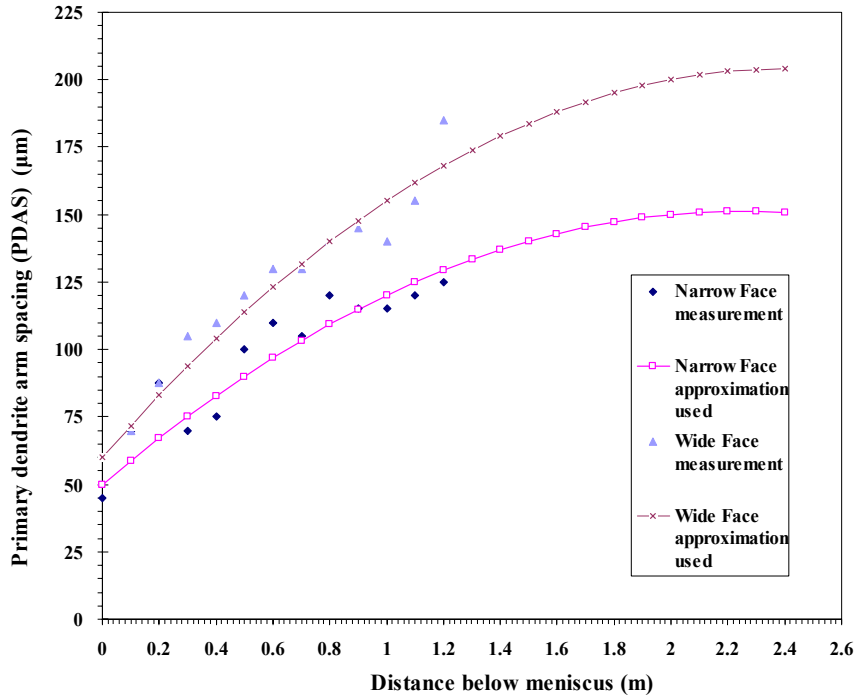


Fig 3.44. Approximation of PDAS down the mold used in computational model.

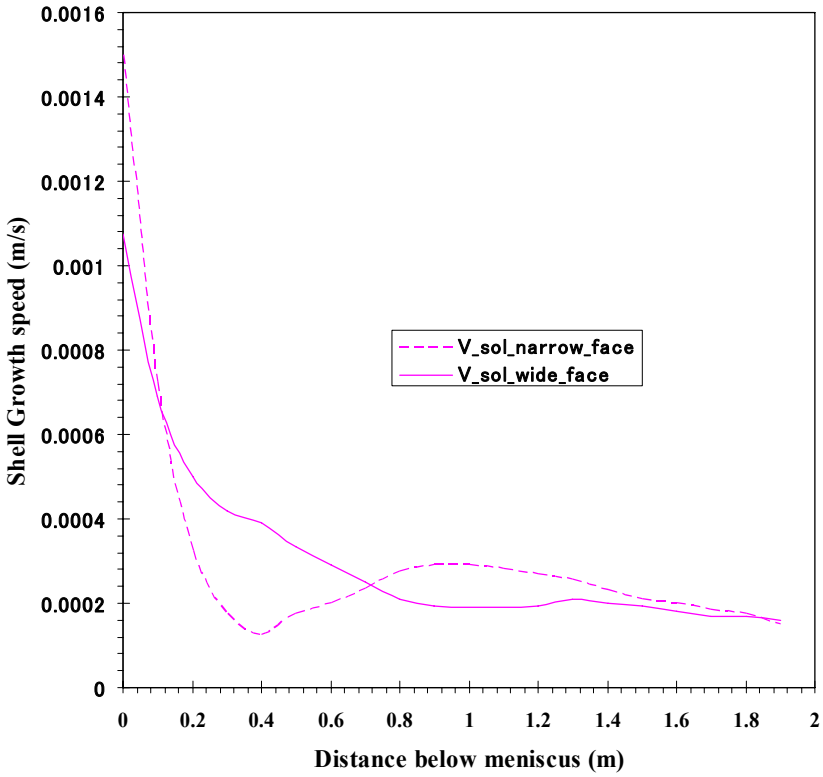


Fig 3.45. Variation of solidification velocity down the mold used in computational model.

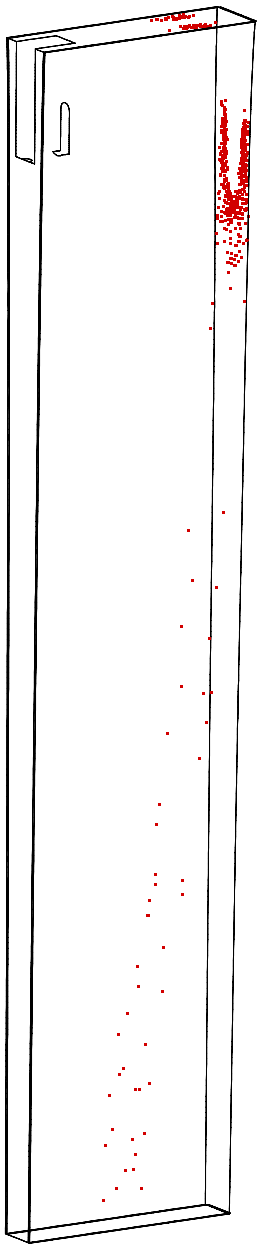
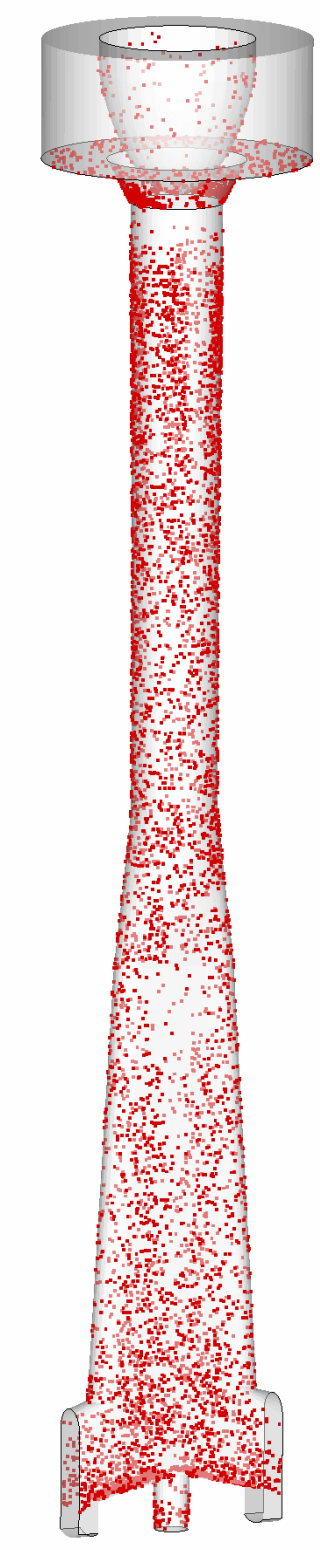
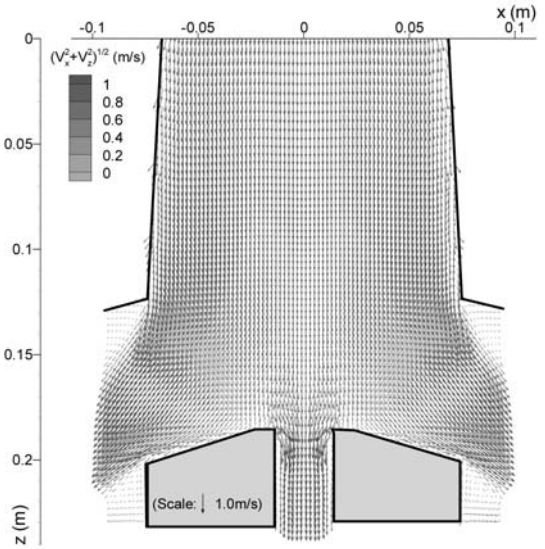
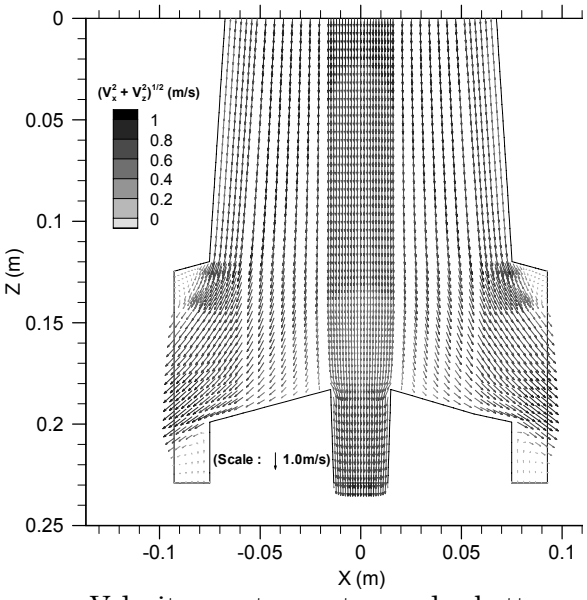


Fig 3.46. Trapped positions of 1250 particles entering from the side outlet using cloud method.





Velocity vectors at nozzle bottom from LES



Velocity vectors at nozzle bottom from $(k-\varepsilon)$

Fig 3.47. Position of entrapped particles in the nozzle using $(k-\varepsilon)$ model and velocity vectors obtained with LES [14] and $(k-\varepsilon)$ model.

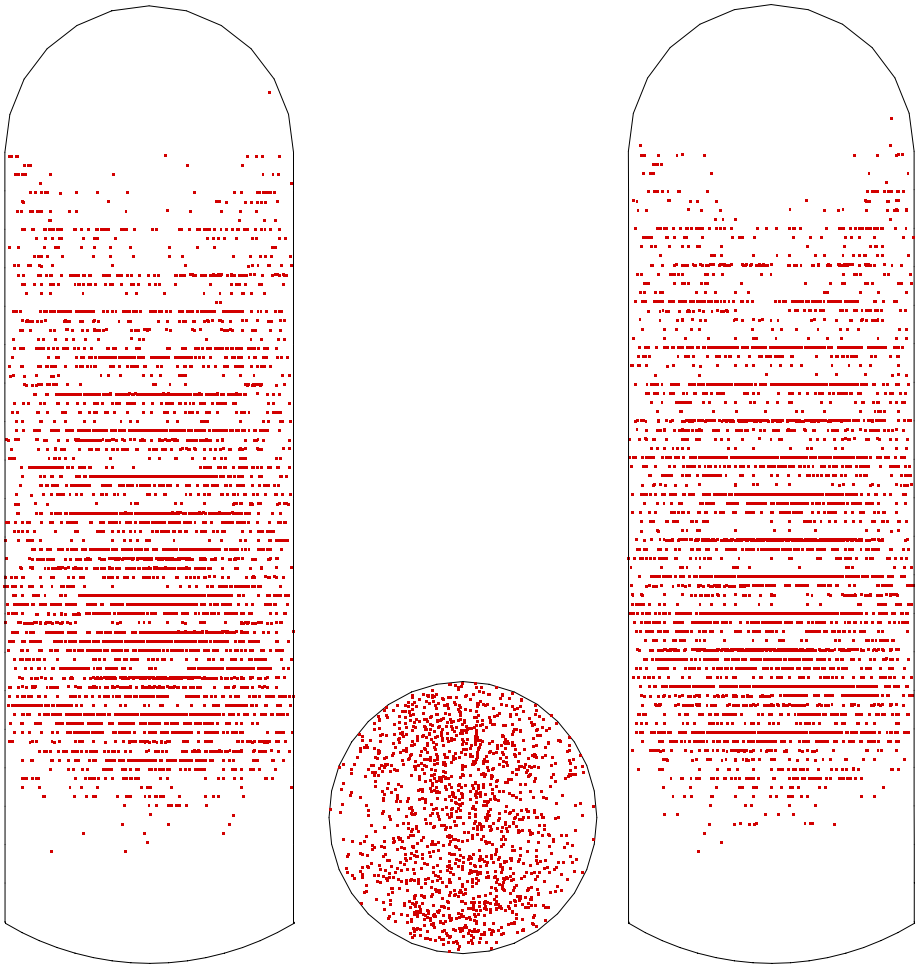


Fig 3.48. Position of particles at the nozzle outlet using $(k-\varepsilon)$ model.

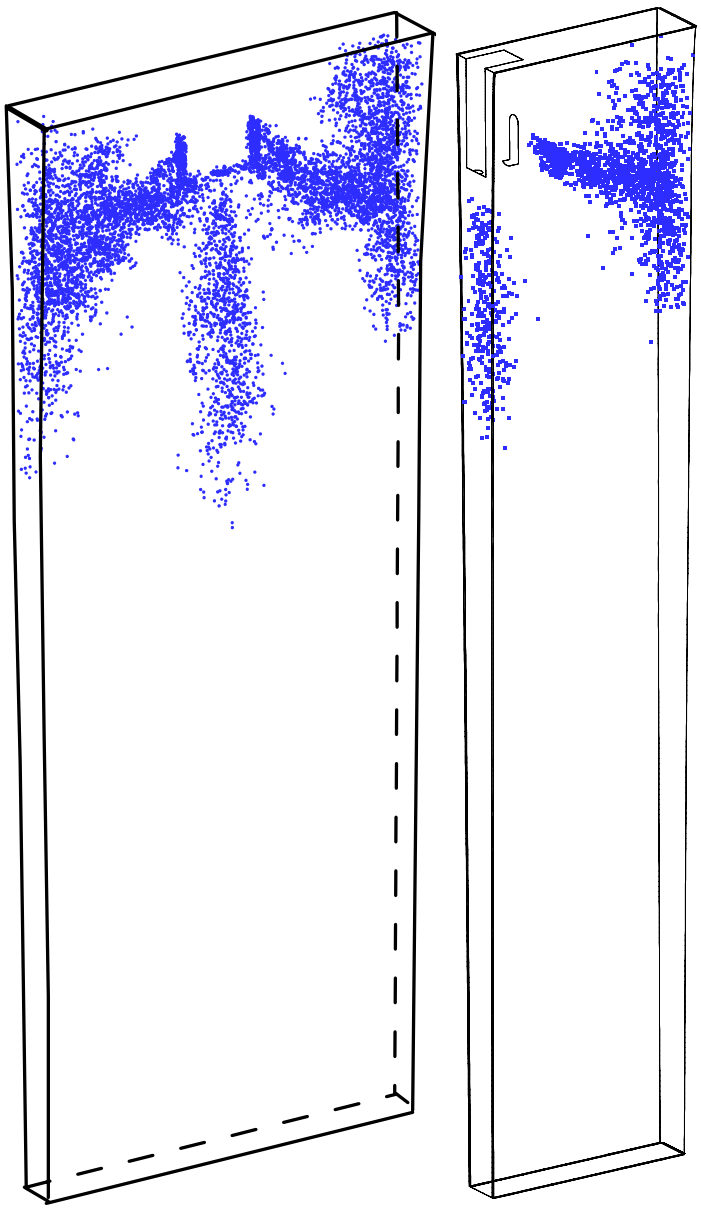


Fig 3.49. Position of particles after 2sec in full mold using LES model [34] and half mold using $(k-\varepsilon)$ model.

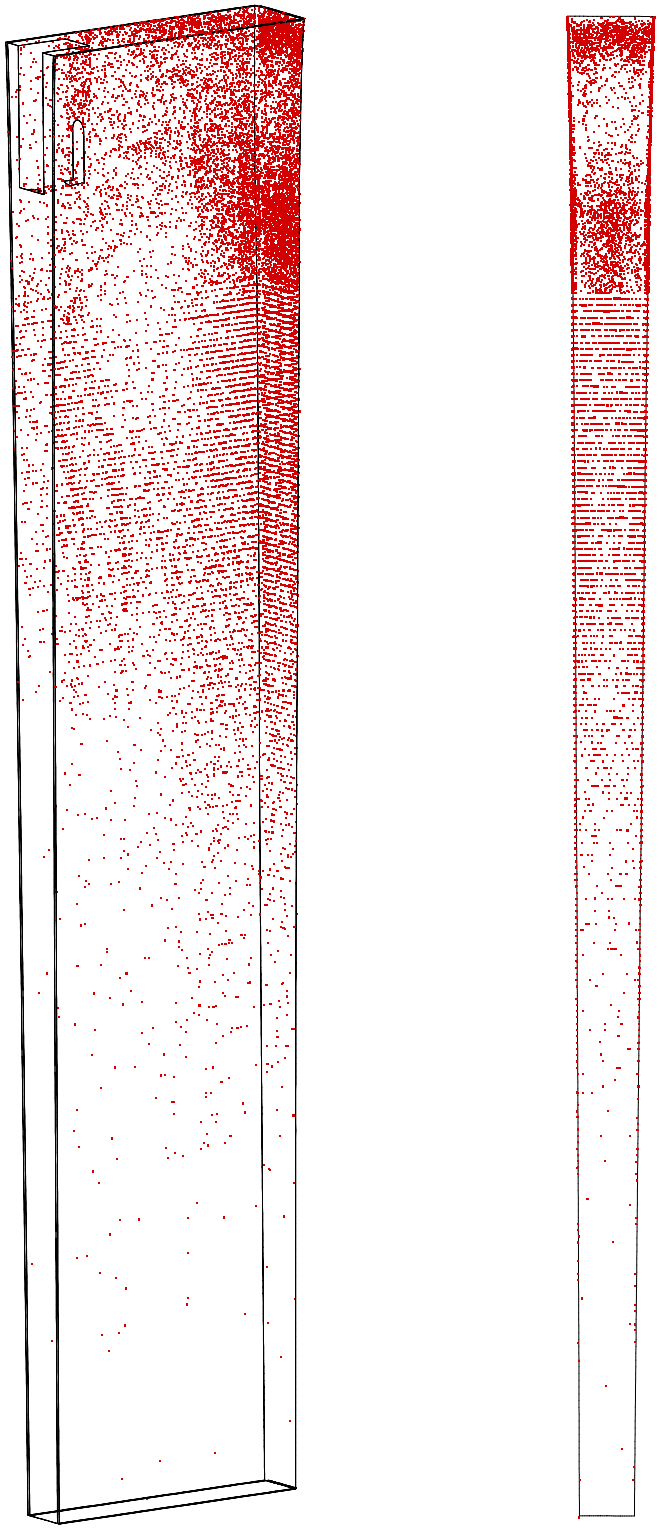


Fig 3.50. Entrapped positions of small particles (10um and 40um diameter) using $(k-\varepsilon)$ model in the entire mold and on narrow face.

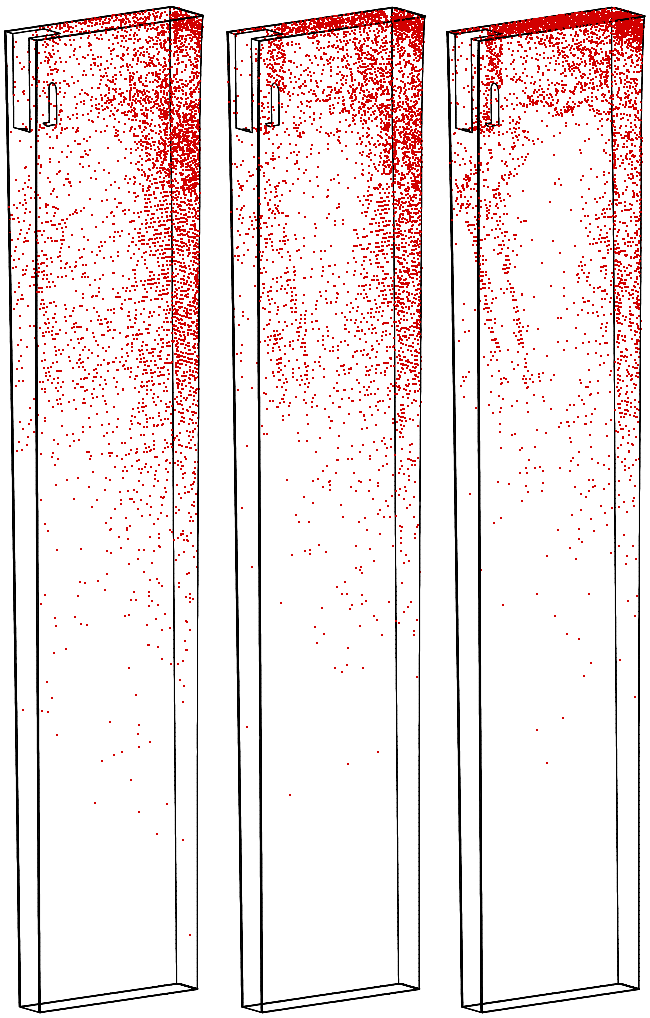


Fig 3.51. Entrapped positions of Alumina particles of 100µm, 250µm and 400µm diameters using $(k-\varepsilon)$ model.

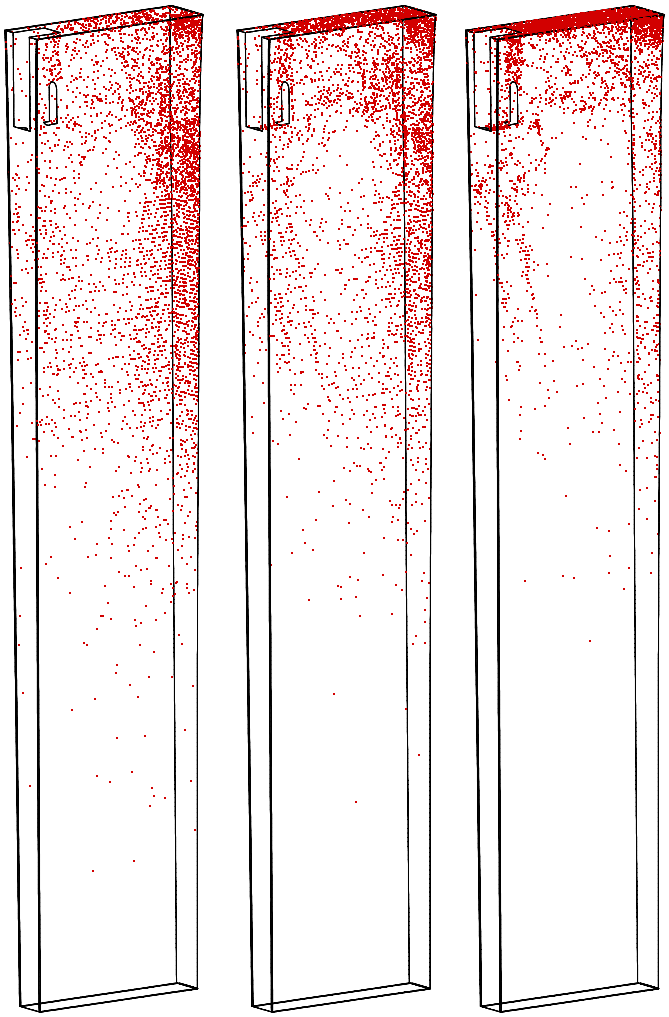


Fig 3.52. Entrapped positions of Argon particles of 100µm, 250µm and 400µm diameters using $(k-\varepsilon)$ model.

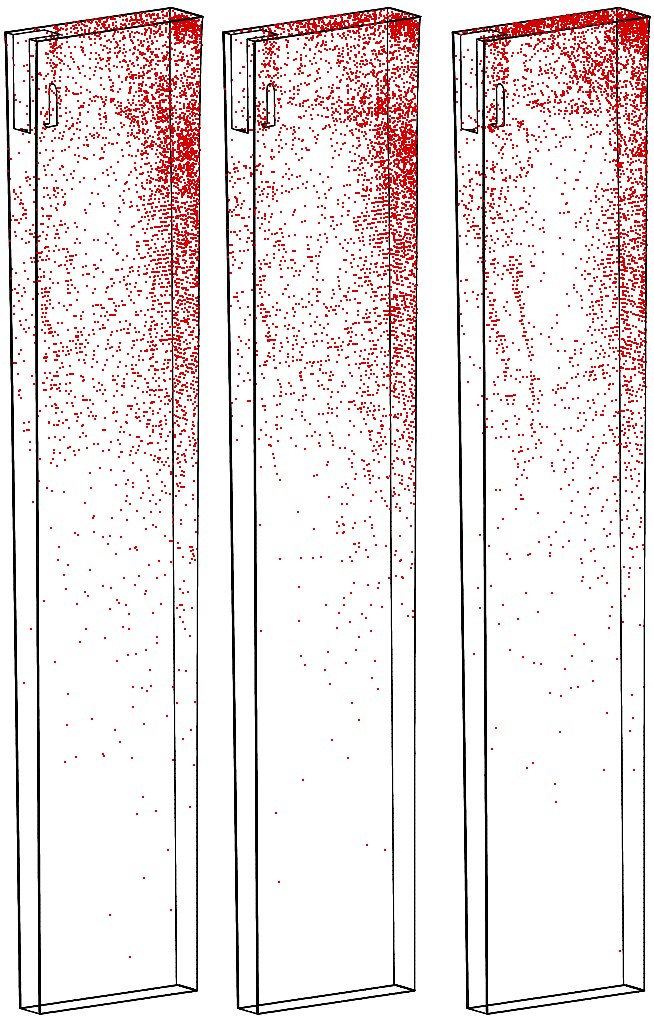


Fig 3.53. Entrapped positions of Slag particles of 100µm, 250µm and 400µm diameters using $(k-\varepsilon)$ model.

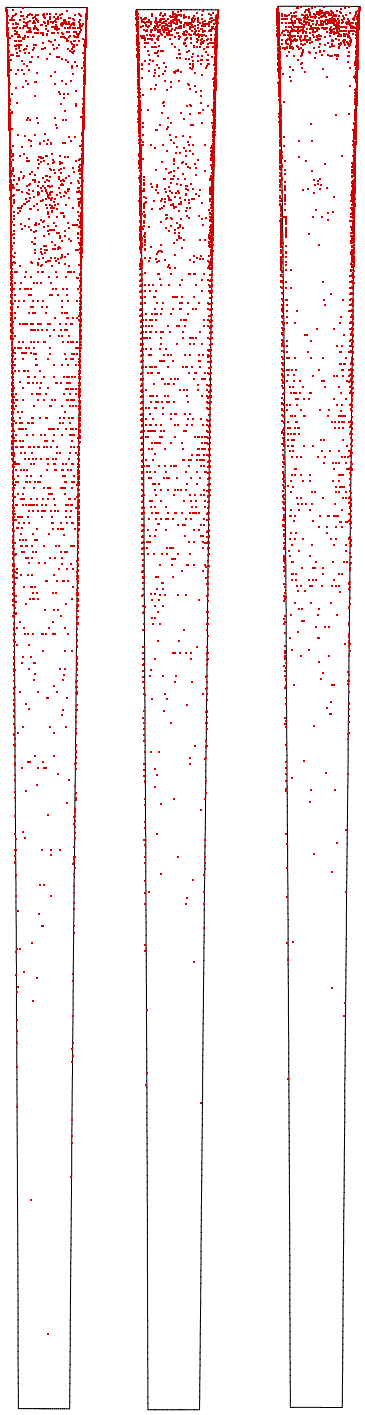


Fig 3.54. Entrapped positions of Alumina particles of 100µm, 250µm and 400µm diameters on the narrow face of the mold using $(k-\varepsilon)$ model.

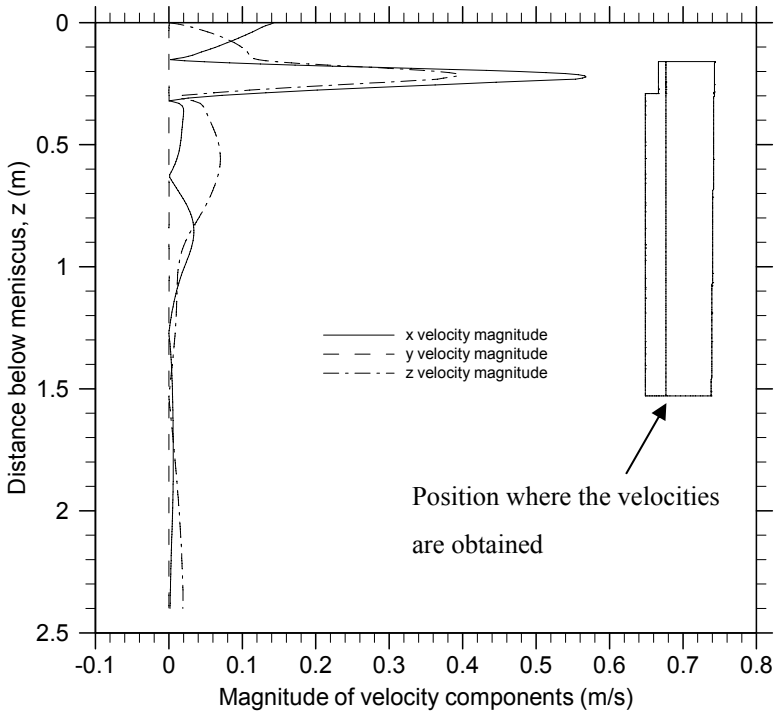


Fig 3.55. Magnitude of velocity components down the mold on a line 51mm from the SEN.

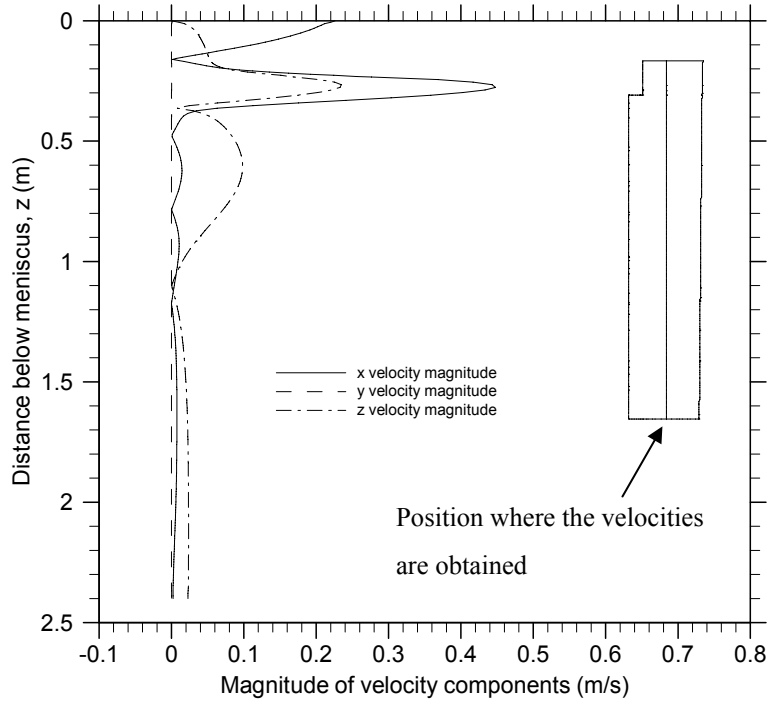


Fig 3.56. Magnitude of velocity components down the mold on a line 155mm from the SEN.

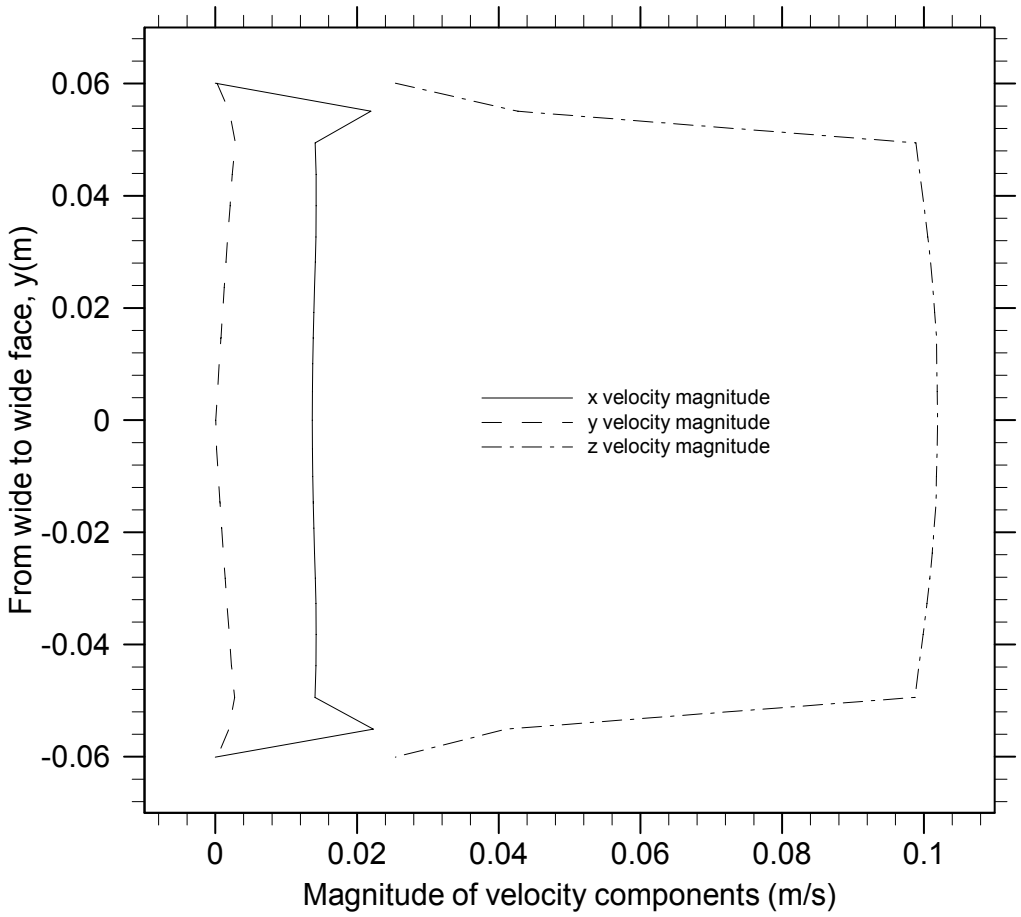


Fig 3.57. Magnitude of velocity components 600 mm below meniscus, 245mm from narrow face.

3.11 REFERENCES

1. Clift, R., J.R. Grace, and M.E. Weber, *Bubbles, Drops and Particles*. 1978, New York: Academics Press, INC.

2. Yuan, Q., *Transient Study of Turbulent Flow and Particle Transport During Continuous Casting of Steel Slabs*, in *Mechanical & Industrial Engineering*. 2004, University of Illinois at Urbana-Champaign, IL: Urbana, IL. p. 196.

3. Thomas, B.G., *Modeling of Continuous Casting Defects Related to Mold Fluid Flow.* AIST Transactions, 2006. **3**(5).

4. Zhang, L. and B.G. Thomas, *State of the Art in Evaluation and Control of Steel Cleanliness.* ISIJ International, 2003. **43**(3): p. 271-291.

5. Thomas, B.G. and L. Zhang, *Review: Mathematical Modeling of Fluid Flow in Continuous Casting.* ISIJ Internat., 2001. **41**(10): p. 1181-1193.

6. Yuan, Q., et al., *Computational and Experimental Study of Turbulent Flow in a 0.4-Scale Water Model of a Continuous Steel Caster.* Metall. & Mater. Trans., 2004. **35B**(5): p. 967-982.

7. Squire, K.D. and J.K. Eaton, *Particle Response and Turbulence Modification in Isotropic Turbulence.* Physics of Fluids, 1990. **2**(7): p. 1191-1203.

8. Wang, Q. and K.D. Squire, *Large Eddy Simulation of Particle Deposition in a Vertical Turbulent Channel Flow.* International Journal of Multiphase Flow, 1996. **22**(4): p. 667-683.

9. Elghobashi, S. and G.C. Truesdell, *Direct Numerical Simulation of Particle Dispersion in a Decaying Isotropic Turbulence.* Journal of Fluid Mechanics, 1992. **242**: p. 655-700.

10. Aboutalebi, M.R., M. Hasan, and R.I.L. Guthrie, *Coupled Turbulent Flow, Heat, and Solute Transport in Continuous Casting Processes.* Metall. Mater. Trans. B, 1995. **26B**(4): p. 731-744.

11. Sussman, R.C., et al., *Inclusion Particle Behavior in a Continuous Slab Casting Mold*, in *10th Process Technology Conference Proc.* 1992, Iron and Steel Society, Warrendale, PA: Toronto, Canada, April 5-8, 1992. p. 291-304.

12. Grimm, B., et al., *Inclusions in Continuously Cast Steel Slabs-Numerical Model and Validation.* Steel Res., 1999. **70**(10).

13. Thomas, B.G., A. Denissov, and H. Bai, *Behavior of Argon Bubbles during Continuous Casting of Steel*, in *Steelmaking Conf. Proc.* 1997, ISS, Warrendale, PA.: Chicago, IL. p. 375-384.

14. Yuan, Q., B.G. Thomas, and S.P. Vanka, *Study of Transient Flow and Particle Transport during Continuous Casting of Steel Slabs, Part 1. Fluid Flow.* Metal. & Material Trans. B., 2004. **35B**(4): p. 685-702.

15. Meng, Y. and B.G. Thomas, *Heat Transfer and Solidification Model of Continuous Slab Casting: CON1D.* Metal. & Material Trans., 2003. **34B**(5): p. 685-705.

16. Launder, B.E. and D.B. Spalding, *Mathematical Models of Turbulence*. 1972: London Academic Press.

17. Creech, D., *Computational Modeling of Multiphase Turbulent Fluid Flow and Heat Transfer in the Continuous Slab Casting Mold*, in *Mechanical and Industrial Engineering*. 1999, University of Illinois at Urbana-Champaign: Urbana, IL.

18. FLUENT6.1-Manual. 2003, Fluent Inc., Lebanon, New Hampshire.

19. Yuan, Q., *Private Communication* 2006: Champaign, IL.

20. Crowe, C., M. Sommerfild, and Y. Tsinji, *Multiphase Flows with Droplets and Particles*. 1998: CRC Press. 71.

21. Saffman, P.G., *The Lift on a Small Sphere in a Slow Shear Flow.* J. Fluid Mech., 1965. **22**(2): p. 385-400.

22. Wang, Q., et al., *On the Role of the Lift Force in Turbulence Simulations of Particle Deposition.* Int. J. Multiphase Flow, 1997. **23**(4): p. 749-763.

23. Mei, R., *Brief Communication: An Approximate Expression for the Shear Lift Force on a Spherical Particle at Finite Reynolds Number.* International Journal of Multiphase Flow, 1992. **18**(1): p. 145-147.

24. Odar, F., *Verification of the Proposed Equation for Calculation of the Forces on a Sphere Accelerating in a Viscous Flow.* Journal of Fluid Mechanics, 1966. **25**: p. 591-592.

25. Uhlmann, D.R. and B. Chalmers, *Interaction Between Particles and a Solid-Liquid Interface.* Journal of Applied Physics, 1964. **35**(10): p. 2986-2993.

26. Shangguan, D., S. Ahuja and D.M. Stefanescu, *An Analytical Model for the Interaction between an Insoluble Particle and an Advancing Solid/Liquid Interface.* Metallurgical Transactions A, 1992. **23A**(2): p. 669-680.

27. Potschke, J. and V. Rogge, *On The Behavior of Foreign Particle at An Advancing Solid-Liquid Interface.* Journal of Crystal Growth, 1989. **94**: p. 726-738.

28. Kaptay, G. and K.K. Kelemen, *The Force Acting on a Sphere Moving towards a Solidification Front due to an Interfacial Energy Gradient at the Sphere/Liquid Interface.* ISIJ International, 2001. **41**(3): p. 305-307.

29. FLUENT6.1. 2003, Fluent Inc., Lebanon, New Hampshire, www.fluent.com.

30. El-Bealy, M. and B.G. Thomas, *Prediction of Dendrite Arm Spacing for Low Alloy Steel Casting.* Metall. Mater. Trans., 1996. **27B**(4): p. 689-693.

31. Kurz, W. and D.J. Fisher, *Dendrite Growth at the Limit of Stability: Tip Radius and Spacing.* Acta Metall., 1981. **29**: p. 11-20.

CHAPTER 4. CONCLUSIONS AND RECOMMENDATIONS

4.1 CONCLUSIONS

Steady-state, three-dimensional computations were performed to study asymmetric fluid flow, particle transport, and entrapment in the nozzle and mold of a steel continuous caster. The flow results obtained from $(k-\varepsilon)$ model compared well with time averaged results obtained from LES for both the validation nozzle and mold. Flow in nozzle geometry and mold geometry was simulated separately. A method to incorporate decreasing fluid mass in the mold was implemented in Fluent.

Three different causes of asymmetry in nozzles were investigated:

- Asymmetric flow entering the nozzle from tundish
- Asymmetric flow due to the presence of slide gate
- Asymmetric flow caused by various types of nozzle clogs

The asymmetric pattern in the nozzle in turn created asymmetric flow pattern in the mold. The differences created in the top surface of the mold, due to different jet characteristics on left and right nozzle ports, were investigated.

Asymmetry created near the nozzle stopper rod due to asymmetry in the tundish flow, dies out when fluid reaches the nozzle bottom and thus no asymmetry is caused at the nozzle outlets. The slide gate is oriented to avoid asymmetry between the left and right

outlet ports of the nozzle, but generates significant swirl within each outlet. Increasing clog asymmetry naturally tends to increase flow asymmetry. Among the different clog shapes modeled, the most severe asymmetry was caused by nozzle clogged at the bore section and for nozzle clogging the bottom well entirely. The difference in flow rate between the left and right port outlets was 10% for both of these clog shapes. Having a well at the nozzle bottom helps create symmetry in flow at the port outlets.

The asymmetric flow pattern created at the nozzle outlets from one of the clog shapes was introduced into the mold. This caused significant asymmetry at the top surface and also throughout the mold length. Vortexes were seen on the left side of the top surface that could cause flux entrapment.

A particle transport and entrapment model was applied. Particles were added into the mold after validating mold flow pattern. Hydrodynamics forces acting upon the particles were calculated and they were comparable with those obtained from LES. Before the particle entrapment model was incorporated into simulation, the effect of several parameters on entrapment was investigated by evaluating the capture criterion.

Increasing primary dendrite arm spacing had the most important effect on increasing particle capture. Small particles were always captured when they touch the solidification front based on their size and PDAS. Particle composition (density: bubble vs. inclusion) shifted the capture window. Bubbles escape more easily than solid inclusions in stagnant flow regions, but their capture depends on the flow pattern. Although steels with low sulfur content tend to have less particle entrapment, the effect was relatively

small. The increased ease of particle capture on the inner radius was a large effect (relative to vertical or outer radius).

Several simulations of 3-D flow, particle transport, and entrapment were performed for particles of different sizes and densities. The positions where the particles get trapped in the nozzle are potential areas where clogging can occur. In the mold simulations, the particles with larger diameters and lower densities were more likely to reach the top surface

Although the particle distribution evolution was qualitatively reasonable, the $(k-\varepsilon)$ model results for particle removal percentages did not match quantitatively with those obtained from LES. The magnitude of the velocity components in all spatial directions suggested that the assumption of isotropic velocity fluctuations used in $(k-\varepsilon)$ model was responsible for the discrepancy.

4.2 FUTURE WORK

In a real caster, whenever the nozzle gets clogged, the slide gate is further opened, to maintain the same flow rate and thus the same casting speed. More simulations should be done for clogged nozzles without reducing the flow rate, to see how the flow in the mold is affected. In addition, transient simulations should be performed to see the relative importance of transient asymmetries caused by turbulence, and time-averaged asymmetries caused by geometric features.

Reynolds stress model (RSM) should be used rather than $(k-\varepsilon)$ model, as it takes

into account the non-isotropy in velocity fluctuations. Further more simulations should be performed, to see the affect of varying casting speed, mold curvature, carbon content, and cooling rate. All these parameters affect the primary dendrite arms spacing down the mold and cross-flow velocities. Electromagnetic stirring can be introduced into the mold, to see the behavior of particles.

APPENDIX A

A.1 Reynolds Number Calculation

The Reynolds number is defined as

$$\mathrm{Re} = \frac{V * D}{n} \tag{A.1}$$

Where, V is the fluid velocity, D is the characteristic length and n is the kinematic viscosity.

For a circular pipe flow, the characteristic length can be defined as the diameter of the pipe. Keeping this in view, the Re number for the nozzle can be calculated based on its bore diameter. At the inlet plane at the top of the nozzle, the inlet velocity value is set, based on the mass flow required to achieve the desired casting speed:

$$V_{avg} = \frac{A_{mold}}{A_{inlet}} * V_{cast} \tag{A.2}$$

To maintain a casting speed of 0.0254m/s, for a mold of width 984mm and thickness 132mm, the inlet speed at the inlet diameter of 70mm, of the nozzle should be 0.857m/s. Using this speed, Re = 75,000 = 7.5*104.

A.2 How to Set Up Case for Nozzle in Fluent

1) Open "Fluent 3ddp"
2) Read nozzle mesh file made by gambit (file should be with extension .msh)
3) Rescale if necessary
   Grid – Scale …
4) File – write – case file
   Save case file
5) Define – models – solver – leave what is by default (segregated solver, implicit scheme, steady) -- ok
6) Define – models – viscous – select k-epsilon model -- leave what is by default (standard wall function are used)
7) Define – material – create steel with specified viscosity and density – change/create – either overwrite air when asked else make sure to go back in the materials panel to select steel that was created.
8) Define – operating conditions – check gravity – enter value its value for the right

direction in grid

9) Define – Boundary conditions – select inlet (velocity inlet) and specify inlet velocity, turbulent kinetic energy and turbulent dissipation rate values – check ok
10) Define – Boundary condition – select outlet (pressure outlet condition) – leave value present by default
11) Define – Boundary condition – select wall boundaries (wall conditions) – leave default
12) Solve – controls – solution – leave default values
13) Solve – initialize – initialize… -- select compute from all zones – click initialize
14) Solve – monitors – residual – check on print and plot – select storage accordingly (increase later during the solution if it does not converges in number if iterations given) – keep convergence criterion 10e-5 or 10e-6
15) Solve – iterate – choose number of iterations – click iterate
16) When converged, save case and data file

A.3 Tundish Region Modeled for Asymmetry in Tundish

Mold cross sectional area perpendicular to the velocity (slab thickness): 0.09m x 1.45m

Casting Velocity: 3.6m/min

Keeping in view Fig A.1

Mass flow rate into the inlet half of the tundish cylinder = $\int_0^z \int_0^l Vr * dl * dz$ (A.3)

where,

Vr is the velocity in the radial direction

dl is the length of the sector segment

dz is the unit height of the cylinder

$$= 2 * \int_0^z \int_0^{90} V * Cosq * r * dq * dz$$
$$= 2 * V * r * z \quad (A.4)$$

USE: Mass Flow input = Mass Flow output

Steel Slab cross-sectional dimensions = 0.09m x 1.45m

$$(2 * V_{inlet} * r * z) - (2 * V_{outlet} * r * z) = V_{casting} * (0.09) * (1.45) \quad (A.5)$$

$$2 * \Delta v * r * z = \frac{3.6}{60} * 0.09 * 1.45$$

$$\Delta v = 0.3 - 0.2 = 0.1 m/s$$

$$r * z = 0.03915\, m^2$$

d diameter of cylindrical segment

z is the height of the cylindrical segment

Use d = z for the cylinder

d = z = 279.8mm

The depth of the stopper rod was chosen so as to make sure that the area of the annulus is half of that of the area of the outer circle as shown in Fig A.2.

d = diameter of the outer circle

r = radius of the inner circle

$$\frac{(\Pi * d^2)}{4} - \Pi * r^2 = \frac{1}{2}(\frac{\Pi * d^2}{4}) \qquad \text{(A.6)}$$

d = 115mm (given in nozzle geometry)

Therefore, r = 40.78mm

A.4 Figures

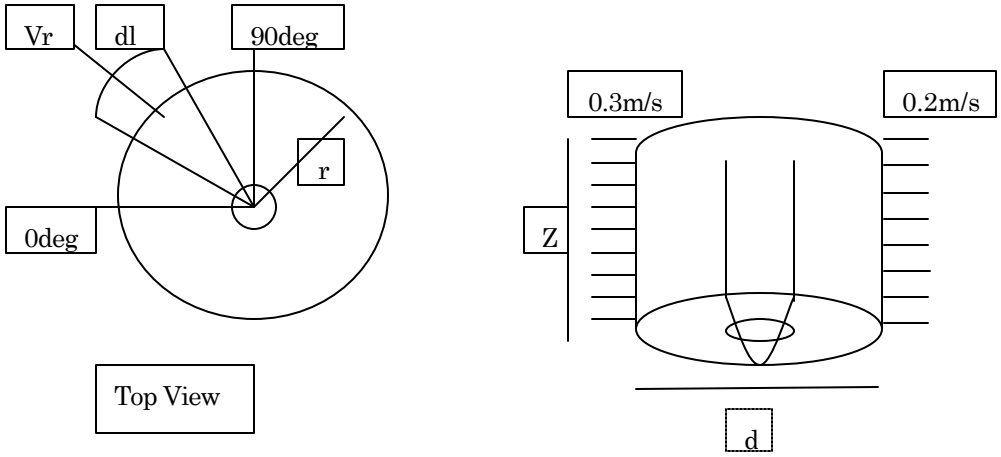


Fig A.1. Two different views of modeled tundish region.

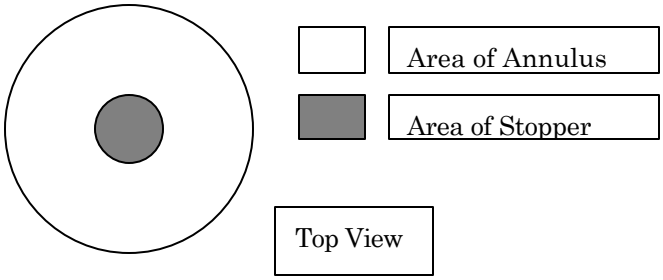


Fig A.2. Explanation of area of annulus and area of stopper rod when viewing from the top.

APPENDIX B

B.1 Case Set Up for Mold

1) Open “Fluent 3ddp”
2) Read mold mesh file made by gambit (file should be with extension .msh)
3) Rescale if necessary
   Grid – Scale …
4) File – write – case file
   Save case file
5) Define – models – solver – leave what is by default (segregated solver, implicit scheme, steady) -- ok
6) Define – models – viscous – select k-epsilon model -- leave what is by default (standard wall function are used)
7) Define – material – create steel with specified viscosity and density – change/create – either overwrite air when asked else make sure to go back in the materials panel to select steel that was created.
8) Define – operating conditions – check gravity – enter value its value for the right direction in grid
9) Define – user defined functions – compiled – Add the file with udf for fluid extraction to represent solidification – ensure to give a new library name each time you add the udf – Build and load – click ok to the information pop up – ensure that no errors were generated written in the main window – the main window will now show the following three

   mass_source
   x_momentum_source
   y_momentum_source
   z_momentum_source

   These are the 4 source terms need to be added only for the cells adjacent to the boundary of the mold, where the fluid solidifies.
10) See B.2, to see how to get the required nozzle outlet values into the mold case
11) Select the fluid zone defining the cells which adjacent to the boundary – check on source terms and add the mass and momentum source terms accordingly – out turbulent kinetic energy and dissipation rate values as zero (constant) – check on laminar zone – click ok

12) Define – Boundary conditions – select inlet (velocity inlet) and specify inlet velocity, turbulent kinetic energy and turbulent dissipation rate values with drop down menus appearing, selecting the profiles obtained from nozzle simulations – check ok
13) Define – Boundary condition – select outlet (pressure outlet condition) – leave value present by default
14) Define – Boundary condition – select wall boundaries (wall conditions) – leave default except for top surface wall,

    -- go to momentum, set shear condition in as specified shear – set zero shear in all directions
15) Solve – controls – solution – set the under-relaxation factors as explained in chapter 3, which shows how they should be likely changed after a number of iterations for the solution convergence to be easier
16) Solve – initialize – initialize… -- select compute from all zones – click initialize
17) Solve – monitors – residual – check on print and plot – select storage accordingly (increase later during the solution if it does not converges in number if iterations given) – keep convergence criterion 10e-5 or 10e-6
18) Solve – iterate – choose number of iterations – click iterate
19) When converged, save case and data file

B.2 How to Get Inlet Values for Mold

The values of velocity components, turbulent kinetic energy and turbulent dissipation rate obtained at the nozzle outlet need to be placed at the mold inlet.

1) Once the flow solution for the nozzle has been obtained, go to

    File – Write – profile – select the velocity components, turbulent kinetic energy and turbulent dissipation rate for the outlets needed – *Write*
2) Profile files will be generated in the folder specified.
3) Set up the case file for mold simulation, go to

    Read – profile – select the profile files generated earlier
4) The profile files read will appear as dropdown menus in the inlet boundary condition panel to be selected as the inlet condition.

B.3 User Defined Function for Boundary Cells

The code for the user defined functions used for extracting fluid from the boundary cells in given below:

//

```
/************************************************************************/
/* UDF for specifying a mass source term to represent steel
*/
/* solidification                                                       */
/************************************************************************/
#include "udf.h"
#include "sg.h"
#include "math.h"
#define casting_velocity 0.0254
#define wall_id 4
//casting speed =25.4mm/s
DEFINE_SOURCE(mass_source, c, t, dS, eqn) //for mass source term
{
real mass, source, area_face;
int i;
real A[ND_ND];
face_t f;
cell_t cc;
Thread *tf;
  c_face_loop(c,t,i)
 {
        f = C_FACE(c,t,i);
        tf = C_FACE_THREAD(c,t,i);
  if(THREAD_ID(C_FACE_THREAD(c,t,i)) == wall_id )
   {
   F_AREA(A,f,tf);
   area_face = A[1];
   source = -7000 * casting_velocity * fabs(area_face)/C_VOLUME(c,t);
   dS[eqn] = 0;
       }
  }
   return source;
   return dS[eqn];
}

DEFINE_SOURCE(x_momentum_source, c, t, dS, eqn) //for x-momentum source
```

```
term
{
real mass, source, area_face;
int i;
real A[ND_ND];
face_t f;
cell_t cc;
Thread *tf;
   c_face_loop(c,t,i)
 {
        f = C_FACE(c,t,i);
        tf = C_FACE_THREAD(c,t,i);
  if(THREAD_ID(C_FACE_THREAD(c,t,i)) == wall_id )
   {
   F_AREA(A,f,tf);
   area_face = A[1];
   source = -7000 * casting_velocity *
fabs(area_face)*C_U(c,t)/C_VOLUME(c,t);
   dS[eqn] = -7000 * casting_velocity * fabs(area_face)/C_VOLUME(c,t);
        }
  }
   return source;
   return dS[eqn];
}

DEFINE_SOURCE(y_momentum_source, c, t, dS, eqn) //for y-momentum source
term
{
real mass, source, area_face;
int i;
real A[ND_ND];
face_t f;
cell_t cc;
Thread *tf;
  c_face_loop(c,t,i)
 {
```

```
        f = C_FACE(c,t,i);
        tf = C_FACE_THREAD(c,t,i);
  if(THREAD_ID(C_FACE_THREAD(c,t,i)) == wall_id )
   {
   F_AREA(A,f,tf);
   area_face = A[1];
   source = -7000 * casting_velocity *
fabs(area_face)*C_V(c,t)/C_VOLUME(c,t);
   dS[eqn] = -7000 * casting_velocity * fabs(area_face)/C_VOLUME(c,t);
        }
  }
   return source;
   return dS[eqn];
}

DEFINE_SOURCE(z_momentum_source, c, t, dS, eqn) //for z-momentum source
term
{
real mass, source, area_face;
int i;
real A[ND_ND];
face_t f;
cell_t cc;
Thread *tf;
  c_face_loop(c,t,i)
 {
        f = C_FACE(c,t,i);
        tf = C_FACE_THREAD(c,t,i);
  if(THREAD_ID(C_FACE_THREAD(c,t,i)) == wall_id )
   {
   F_AREA(A,f,tf);
   area_face = A[1];
   source = -7000 * casting_velocity *
fabs(area_face)*C_W(c,t)/C_VOLUME(c,t);
   dS[eqn] = -7000 * casting_velocity * fabs(area_face)/C_VOLUME(c,t);
        }
```

```
  }
   return source;
   return dS[eqn];
}
/////////////////////////////////////////////////////////////////
```

B.4 Steps Needed to Add Particles in Mold

1) Make sure you have the solved steady state solution for flow in the mold

2) Open the data file for solved mold flow

3) Make sure that the input file "v_sol_to_read2.txt" is present in the same folder as mold flow solution data file. This input file basically has the vallues of solidification front velocities for narrow and wide face walls, needed to be used by the udf's that will be added.

4) To go

Define – user defined function – compiled – Add source files

Source file number 1 "trying_debug_8.c" (containing the mass extraction source terms for mold)

Source file number 2 "May25_boundary_1_close_1.c" (containing the Drag force function, shear lift force function and boundary conditions for particles larger than 40um)

Build – load

5) Initiate discrete phase

Define – Models – Discrete Phase

6) Keep the Drag law as spherical for 10um and 40um particles. For particles larger than 40um diameter, select the user defined drag force function "particle_drag_force".

7) Check on the Saffman lift force for 10um and 40um particles. For particles larger than 40um diameter, select the user defined shear lift force function "DPMBF_Lift_and_Gravity" in the *body force* menu. Although the name includes the word gravity, gravity is not defined in this function, but is present already because of gravity in the *operating conditions* panel.

8) To initiate particles go to

Injections on the discrete phase control panel – *create* – select injection type as *file* – click on *file* – select the file (read Appendix B.5**)**

9) Go to

Turbulent dispersion on the same panel – check *stochastic model* and *random walk*

10) Click *ok* to close all panels one by one

11) Note that the material for the particle has not yet been set up and the default as anthracite is present. The new material needs to be created first. Go to

Define – Materials – select material type as *inert particle* – change the name and put the correct density – click change/create – when asked for overwrite, say no.

This will create the new material but this needs to be placed in the *set injections panel.*

12) Go to

Define – Models – discrete phase – injection – select the injection created earlier and click *set*

Change the material to the new one defined and hit *ok.*

13) The boundary conditions now need to be set. Go to

Define – Boundary conditions – select all the wall zone types one by one to set them correctly

Set – DPM – select boundary condition type as trap for all particles of diameter $\leq$40um.

For particles larger than 40um in diameter, *set user defined* and then select

“bc_nozzle_walls” for nozzle walls

“bc_surface_top” for top surface

“bc_reflect” for Mold walls

“bc_nozzle_walls” and “bc_surface_top” simply trap the particle whenever it touches the nozzle walls. For nozzle walls set reflect can also be used if particles are to be reflected from the nozzle walls, rather than selecting this user defined function. “bc_reflect” for Mold walls actually does the force balance analysis for the larger particles.

14) Go to

Sove – iterate – iterate for only 1 or 3 iterations, just to tell Fluent that the DPM model has been incorporated. The trajectories are not calculated at this point.

15) Go to

Display – particle tracks – select the *injection* – check *summary* – click *Display*

Note:

If there are more than 500 particles, it is recommended not the do (15), as this shows the trajectories of all the particles, fro the beginning to the end and consumes a lot of time. It is also hard to grasp anything from the end picture produced, with too many

trajectories on the same plot.
16) The important thing is to see the end postions of the particles and (15) will not produce that for particles of size $\leq$40um diameter. For particles of size $\leq$40um diameter, read Appendix B.6. This is the point where Fluent, calculates the particle trajectories and can take considerable time for larger particles, because of the user defined functions.
17) Either (15) or (16) can be used to get the trapped positions of larger particles. Larger particles had the user defined functions incorporated for the boundaries, which generate the .txt files for trapped particle positions in the same folder as the case and data files.
18) To display the entrapped particle positions, see Appendix B.7.

B.5 To Place Particles at Inlet

The particles initial conditions can be read from an external file to describe the injection distribution. The file needs to be created in the following format:
((x y z u v w diameter temperature mass-flow) name)
Where, x, y, z are the coordinates giving particle's position and u, v, w give the particles initial velocity. Each particle can be given a name as well, but it is optional.

Below is a part of a file generated:

```
(( 7.279307e-003 -7.900000e-001 8.775367e-003     0     0     0 1.000000e-004 300 1.000000e-004) )
(( 1.084618e-003 -7.900000e-001 -2.552804e-003    0     0     0 1.000000e-004 300 1.000000e-004) )
(( 2.450586e-003 -7.900000e-001 -6.857388e-003    0     0     0 1.000000e-004 300 1.000000e-004) )
(( 3.772908e-004 -7.900000e-001 5.819572e-003     0     0     0 1.000000e-004 300 1.000000e-004) )
(( 1.037934e-002 -7.900000e-001 2.581638e-003     0     0     0 1.000000e-004 300 1.000000e-004) )
(( 8.521723e-003 -7.900000e-001 3.318768e-003     0     0     0 1.000000e-004 300 1.000000e-004) )
(( 4.068821e-003 -7.900000e-001 3.667279e-003     0     0     0 1.000000e-004 300 1.000000e-004) )
(( 1.474070e-004 -7.900000e-001 -1.660559e-004    0     0     0 1.000000e-004 300 1.000000e-004) )
(( 9.572481e-003 -7.900000e-001 -2.350705e-003    0     0     0 1.000000e-004 300 1.000000e-004) )
```

The mass-flow can be set to any value, as it is only used in unsteady particle tracking, where fluid flow is solved at the same time along with the particle trajectories. The temperature can be set to any value as well, as the temperature equation is not being solved. The initial velocities as can be seen in the file are set to 0. This does not matter significantly, as the particles quickly change their velocities, depending on the flow velocity. The name to any particle has not been given, as it was optional. The particle positions are random positions at the mold inlet.

Make sure to place this file in the same folder as the case and data file for the solved mold flow.

A code was written in Matlab to write the file in the format explained above. The code needs to be modified based on the inlet coordinates of the mold. The code is given below:

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
clear

fid = fopen('400_27_b_May_3.txt', 'wt'); %open a writable file

if (fid == -1)
    error('cannot open file for writing');
end

%%Getting random position for particles on the bottom inlet of validation
%%mold
r = (12*rand(1,500))/1000; %500 particles placed at the bottom inlet
rand('state',sum(100*clock));
theta = (-90 + 180*rand(1,500))*(22/7)/180;

for (i = 1 : 500)
    x(i) = r(i)*cos(theta(i));
    z(i) = r(i)*sin(theta(i));
end

y = -790/1000;

u = 0;
v = 0;
w = 0;
diam = 0.4/1000;
temp = 300;
massflow = 0.0001;
```

```
for (i = 1 : 500) %%writing the random postion in the specific file format
fprintf(fid,'(( %3d %3d %3d %3d %3d %3d %3d %3d %3d) )\n' , x(i), y, z(i), u, v, w, diam, temp,
massflow);
end

%%Getting random position for particles on the right side inlet of
%%validation mold
xs = 0.095;

ys = (-690 - 90*rand(1,2000))/1000;%2000 particles placed at the side inlet
rand('state',sum(100*clock));
zs = (-15 + 30*rand(1,2000))/1000;

us = 0;
vs = 0;
ws = 0;
diams = 0.4/1000;
temps = 300;
massflows = 0.0001;

for (i = 1 : 2000) %%writing the random postion in the specific file format
fprintf(fid,'(( %3d %3d %3d %3d %3d %3d %3d %3d %3d) )\n' , xs, ys(i), zs(i), us, vs, ws, diams, temps,
massflows);
end

fclose (fid); %closing the file
```

//

B.6 To Get Particle Positions at Boundaries

1) Go to

Report – discrete phase – sample

Select the injections and select the boundaries on which to see the particle positions.

Compute

Files will be generated with the names of the boundaries selected with .dpm extension

in the same folder as the case and data file.

2) These files can be opened in excel sheet.

B.7 To Display Entrapped Particles in Tecplot

1) Extract the values of just x,y and z coordinates from the .dpm files or the .txt file for large particles.

2) Create a .txt file with these coordinates in the format shown below:

x y z

The coordinates for each particle should be in the different line.

3) Change the extension of the file to .plt

4) Open the file in Tecplot

5) *Select 3-D – click ok to message for aspect ratio*

6) To fix the aspect ratio, go to

Axis – edit axis – select xyz dependency and place x to y ratio as 1 and y to z ratio as 1

7) To get the mold boundaries into Tecplot, go to Fluent

File – export – check tecplot – select the boundaries to be exported.

Write – give the file extension as .plt

8) Before you open this file in the same tecplot as the particles file. Go to

data – data set info – change the V1, V2, V3 names to X, Y, Z

This needs to be done as the file generate by Fluent has variable X, Y, Z as the boundary components.

9) Now open the .plt file for boundaries in the same Tecplot

File – load data file – check add to current data

10) Then use Tecplot to represent the particles as *scatter* and close scatter show for the mold boundaries in *plot attributes*.

11) The color and size of scatters can be changed and the color for the boundaries.

12) Close the axis show from *edit axis* for better visualization of the particles.

13) The figure can be rotated in with the rotation buttons provided on the left side.

B.8 Code for User Defined Functions for Particle Trajectory and Entrapment

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

/*************************************************************/
/* trying to initialize particles on a certain surface  */
```

```
/*             square channel                    */
/*****************************************************************/


// Disable the warning c4996 from compiler using fscanf and fopen
#pragma warning(disable : 4996)


#include "udf.h"
#include "mem.h"
#include "sg.h"
#include "math.h"
#include "surf.h"
#include "dpm.h"
#include "stdio.h"

#define wall_id 4 // outer mold walls ID number

double Dragforce[3];
double Liftforce[3];
double Cross_vel[3];
double Eta[3];
double Net_force_eta[3];

double Vel_diff_mag2;
double Drag_coeff;
double Drag_help;
double Rep;

int file_read = 1;

#define SIZE  20 // size for v_sol interpolation
#define SIZE_T_N 15 // size for PDAS interpolation

double x_c_vsol[SIZE], y_c_vsol[SIZE], y_c_vsol_w[SIZE]; // global
declaration of solidification velocity vectors file narrow
```

```
/*////////////////////////////////////////////////////////////////////
/////////////////*/

/*function to read the solidification velocity on narrow face and wide face
from a file*/
void reading_v_sol(double x_c[SIZE], double y_c[SIZE], double y_c_w[SIZE])
{
        FILE *fr;
        int i;
        double x_val, y_val, y_val_w;
        // x_val = distance below meniscus
        // y_val = Vsol value on narrow face
        // y_val_w = Vsol value on wide face

        char line[80];

        // reading from the file v_sol_to_read.txt
        fr = fopen ("v_sol_to_read2.txt", "rt");
        i=0;
        while(fgets(line, 80, fr) != NULL)
        {
         sscanf (line, "%lf %lf %lf", &x_val, &y_val, &y_val_w);

         x_c[i] = x_val;
         y_c[i] = y_val;
         y_c_w[i] = y_val_w;

         i = i+1;
        }
         fclose(fr);
}

/*////////////////////////////////////////////////////////////////////
/////////////////*/
```

```
/*function to find the solidification velocity on narrow face and wide face
by interpolation*/
double finding_vsol_inter(double y_pos, double x_c_v[SIZE], double
y_c_v[SIZE])
{
	double v_sol;
	double x1, x2, y1, y2, x;

	int i;

	// Doing linear interpolation by first finding between which two
points y_pos exits
	x = -(y_pos + 0.5576);  // to get the distance in term of distance
below meniscus

	for (i = 0; i < SIZE; i++)
	{
		if ( (x_c_v[i] < x) && (x < x_c_v[i+1]) )
		{
			x1 = x_c_v[i];
			x2 = x_c_v[i+1];
			y1 = y_c_v[i];
			y2 = y_c_v[i+1];

			v_sol = ((x-x1)*y2 + (x2-x)*y1)/(x2-x1);

			return (v_sol);
		}

	}

	v_sol = y_c_v[SIZE]; // incase the particle hits way below in the
mold
	return (v_sol);
```

```
}

/*////////////////////////////////////////////////////////////////////
//////////////////*/

/*function to get the PDAS on the narrow face and wide face for a certian
distance below meniscus*/
double finding_PDAS_inter(double y_pos, int PDAS_face)
{

	double PDAS;
	double x;

	// Doing linear interpolation by first finding between which two
points y_pos exits
	x = -(y_pos + 0.5576);



	if (PDAS_face == 1)
	{
		PDAS = -2*pow(10,-5)*pow(x,2) + 9*pow(10,-5)*x +
5*pow(10,-5); // narrow face wall
	}
	else
	{
		PDAS = -2.5*pow(10,-5)*pow(x,2) + 0.00012*x + 6*pow(10,-5);
// wide face wall
	}

	return (PDAS);
}

/*////////////////////////////////////////////////////////////////////
//////////////////*/

/*Macro that is used when the particle hits the nozzle walls to determine
```

```
its position*/
DEFINE_DPM_BC(bc_nozzle_walls, p, t, f, f_normal, dim)
{
       FILE *fin;

              fin = fopen ("nozzle_boundary_hits.txt", "a");
              fprintf (fin, "P_POS(p)[0] is %e\t P_POS(p)[1] is %e\t
P_POS(p)[2] %e\n" , P_POS(p)[0], P_POS(p)[1], P_POS(p)[2]);
              fclose(fin);
              return (DPM_BC_TRAP);
}

/*////////////////////////////////////////////////////////////////////
/////////////////*/

/*Macro that is used when the particle hits the top surface to determine
its postition*/
DEFINE_DPM_BC(bc_surface_top, p, t, f, f_normal, dim)
{
       FILE *fis;

              fis = fopen ("surface_boundary_hits.txt", "a");
              fprintf (fis, "P_POS(p)[0] is %e\t P_POS(p)[1] is %e\t
P_POS(p)[2] %e\n" , P_POS(p)[0], P_POS(p)[1], P_POS(p)[2]);
              fclose(fis);
              return (DPM_BC_TRAP);
}

/*////////////////////////////////////////////////////////////////////
/////////////////*/

/*Macro that is used everytime the particle hits the mold boundary walls
to determine its fate*/
DEFINE_DPM_BC(bc_reflect, p, t, f, f_normal, dim)
{
       FILE *fi;
```

```
        FILE *fib;

        real x[ND_ND];

        int i, idim;
        double yoyo, yoyo2, yoyo3;
        int signyo2, signyo3;

    double Vsol, Rp, rd, F_lub, ho;
        double s_e, a_not, F_vand;
        double alpha, beta, zeta, n, Co, C_star, F_grad;
        double Ds, k, first_term, second_term, m;
        double B_W_force[3]; // net buoyancy and weight force
        double theeta, F_tot_x[3], PDAS, Xi[3], Net_force_eta[3];
        double Rel_vel[3];
        double F_tot_x_try;

        double y_pos; // y position in the mold where particle hits

        double Velocity_diff[3];
        double Cross_vel2[3];
        int PDAS_face;

        idim = dim;

        y_pos = P_POS(p)[1];

        if (file_read == 1)
        {
               reading_v_sol(x_c_vsol, y_c_vsol, y_c_vsol_w); // reading
text file for solidification velocity
               file_read = 2; // To read only once in the program
        }


               if ( f_normal[0] > f_normal[2]) // narrow face
```

```
                    {
                    PDAS_face = 1;
                    Vsol = finding_vsol_inter(y_pos, x_c_vsol,
y_c_vsol); // interpolation for solidification velocity (narrow face)
                    PDAS = finding_PDAS_inter(y_pos, PDAS_face); //
Finds the PDAS value from a function (narrow face)
                    }
             else // wide face
                    {
                    PDAS_face = -1;
                    Vsol = finding_vsol_inter(y_pos, x_c_vsol,
y_c_vsol_w); // interpolation for solidification velocity (wide face)
                    PDAS = finding_PDAS_inter(y_pos, PDAS_face); //
Finds the PDAS value from a function (wide face)
                    }


       // without even doing any force analysis, if particle diameter is
smaller than PDAS, trap the particle if...
       if (P_DIAM(p) < PDAS)
       {
              fib = fopen ("wall_boundary_hits.txt", "a");
              fprintf (fib, "P_POS(p)[0] is %e\t P_POS(p)[1] is %e\t
P_POS(p)[2] %e\n" , P_POS(p)[0], P_POS(p)[1], P_POS(p)[2]);
              fclose(fib);
              return (DPM_BC_TRAP);

       }

       fi = fopen ("Mar12_boundary.txt", "a");

       /*lubrication force begins*/
       Rp = P_DIAM(p)/2; // particle radius
       rd = 0.0000033; // dendrite tip radius

       //ho is distance between dendrite tip and particle raduis .. it is
```

```
assumed that this is much smaller than Rp and rd
    ho = 6.22093*pow(10,-8); // for 200um
        ho = 7.84*pow(10,-8); // for 400um

        //ho for 100um particle according to Kaptay should be 4.9e-8

        F_lub = 6.0*M_PI*0.006*Vsol*(pow(Rp,2)/ho)*pow((rd/(Rp+rd)),2);
// lubrication force
        /*lubrication force ends*/

        /*Interfacial force begins*/
        s_e = 0.963; // surface energy force
        a_not = 2.5*pow(10,-10); // atomic diameter of the liquid

        F_vand = 2*M_PI*s_e*((rd*Rp)/(rd+Rp))*pow(a_not,2)/pow(ho,2); //
vanderwall interfacial force
        /*interfacial force ends*/

        /*surface energy gradient force b
        egins*/
        n = 840; // (1/mass%)
        Co = 0.0028; // (mass%)
        alpha = 1+ (n*Co);
        Ds = 3.4*pow(10,-9);  // diffusion coefficient (m2/s)
        k = 0.05; // Distribution coefficient (Cs/Cl)
        C_star = Co / (1 - ((Vsol*rd)/(2*Ds))*(1-k));
        beta = n*rd*(C_star- Co);
        zeta = Rp + rd + ho;
        m = 0.171;

        first_term = -(m*beta*M_PI*Rp/pow(zeta,2)) *
(  ((pow(zeta,2)-pow(Rp,2))/beta) * log( ((zeta+Rp)*
(alpha*(zeta-Rp)+beta)) / ((zeta-Rp)* (alpha*(zeta+Rp)+beta)) ) );

        // second term has second and third term
        second_term = -(m*beta*M_PI*Rp/pow(zeta,2)) *  ( (2*Rp/alpha) -
```

```
(beta/pow(alpha,2))*log( (alpha*(zeta+Rp)+beta) /
(alpha*(zeta-Rp)+beta) ) );

        F_grad = first_term + second_term;
        /*Surface energy gradient force ends*/

        /* net buoyancy and weight force begins */
        { // ??
        cell_t c = P_CELL(p); // get the cell the particle is currently in
        { // ??
        Thread *t = P_CELL_THREAD(p); // get the thread the particle is
currently in

        face_t f;
        Thread *tf;

        B_W_force[0] = 0.0;
        B_W_force[1] = ( C_R(c,t)- P_RHO(p) ) * (4.0/3.0)* M_PI *
pow(P_DIAM(p)/2,3) * 9.81; // upwards if the particle density is less than
fluid density
        B_W_force[2] = 0.0;
        /* net buoyancy and weight force ends */



        C_CENTROID(x,c,t);

        Velocity_diff[0] = C_U(c,t) - P_VEL(p)[0];
        Velocity_diff[1] = C_V(c,t) - P_VEL(p)[1];
        Velocity_diff[2] = C_W(c,t) - P_VEL(p)[2];

        Dragforce[0] = Drag_help * (Velocity_diff[0]) /
NV_MAG(Velocity_diff);
        Dragforce[1] = Drag_help * (Velocity_diff[1]) /
NV_MAG(Velocity_diff);
        Dragforce[2] = Drag_help * (Velocity_diff[2]) /
NV_MAG(Velocity_diff);
```

```
        /*SETTING ESCAPE CRITERION*/

        c_face_loop(c,t,i)
        {

          f = C_FACE(c,t,i);

          tf = C_FACE_THREAD(c,t,i);

        if(THREAD_ID(C_FACE_THREAD(c,t,i)) == wall_id )
               {

                        Xi[0] = -f_normal[0]; // unit normal vector // face
normal vector
                        Xi[1] = -f_normal[1];
                        Xi[2] = -f_normal[2];

                        //finding the Eta direction ( Sum of Bouyancy and Drag
force )

                        Cross_vel[0] = B_W_force[0] + Dragforce[0];
                        Cross_vel[1] = B_W_force[1] + Dragforce[1];
                        Cross_vel[2] = B_W_force[2] + Dragforce[2];


                        //take dot product of Cross_vel with Xi .. then
multiply this number with Xi (unit vector) subtract this from Cross_vel
vector
                        // to get Cross_vel2

                        Cross_vel2[0] = Cross_vel[0] -
NV_DOT(Cross_vel,Xi)*Xi[0];
                        Cross_vel2[1] = Cross_vel[1] -
NV_DOT(Cross_vel,Xi)*Xi[1];
```

```
                    Cross_vel2[2] = Cross_vel[2] -
NV_DOT(Cross_vel,Xi)*Xi[2];


                    Eta[0] = Cross_vel2[0]/NV_MAG(Cross_vel2); //
getting unit vector
                    Eta[1] = Cross_vel2[1]/NV_MAG(Cross_vel2);
                    Eta[2] = Cross_vel2[2]/NV_MAG(Cross_vel2);

             }

      }

      theeta = asin(0.5*PDAS/(Rp+rd));


      }//?
      }//?


      F_tot_x_try = NV_MAG(Liftforce) +  NV_DOT(B_W_force,Xi) +
NV_DOT(Dragforce,Xi) - 2*(F_lub - F_grad - F_vand)*cos(theeta);


      fclose(fi); // closing file

      if (F_tot_x_try > 0.0)
      {
             //Message("particle pushed away: \n");  //particle pushed
away

             return DPM_BC_REFLECT;

      }
```

```
        else // check for rotation
        {

                /*fi = fopen ("Mar12_boundary.txt", "a");
                fprintf (fi, "1 \n");
                fclose(fi);*/

                yoyo2 = NV_DOT(Dragforce,Eta);

                if (yoyo2 > 0)
                {
                        signyo2 = 1;
                }
                else
                {
                        signyo2 = -1;
                }

                yoyo3 = NV_DOT(B_W_force,Eta);

                if (yoyo3 > 0)
                {
                        signyo3 = 1;
                }
                else
                {
                        signyo3 = -1;
                }


                if (signyo2 == signyo3) // both drageta and boyeta in the
same direction
                {


                        if      ((NV_DOT(Dragforce,Eta)*signyo2 +
```

```
NV_DOT(B_W_force,Eta)*signyo3)*cos(theeta) + (NV_MAG(Liftforce)+
NV_DOT(Dragforce,Xi)+ NV_DOT(B_W_force,Xi))*sin(theeta) < (F_lub - F_grad
- F_vand)*sin(2*theeta))
                    {


                           fib = fopen ("wall_boundary_hits.txt", "a");
                           fprintf (fib, "P_POS(p)[0] is %e\t
P_POS(p)[1] is %e\t P_POS(p)[2] %e\n" , P_POS(p)[0], P_POS(p)[1],
P_POS(p)[2]);
                           fclose(fib);

                           return DPM_BC_TRAP;
                    }
                    else
                    {

                           return DPM_BC_REFLECT;
                    }


             }

             else // drageta and boyeta in opposite directions

             {

                    if ( (signyo2*yoyo2) > (signyo3*yoyo3) )
                    {


                           if( (NV_DOT(Dragforce,Eta)*signyo2 -
NV_DOT(B_W_force,Eta)*signyo3)*cos(theeta) + (NV_MAG(Liftforce)+
NV_DOT(Dragforce,Xi)+ NV_DOT(B_W_force,Xi))*sin(theeta) < (F_lub - F_grad
- F_vand)*sin(2*theeta))
                           {
```

```
                                    fib = fopen ("wall_boundary_hits.txt",
"a");
                                     fprintf (fib, "P_POS(p)[0] is %e\t
P_POS(p)[1] is %e\t P_POS(p)[2] %e\n" , P_POS(p)[0], P_POS(p)[1],
P_POS(p)[2]);
                                     fclose(fib);

                                     return DPM_BC_TRAP;

                              }

                              else
                              {

                                     return DPM_BC_REFLECT;
                              }

                       }

                       else
                       {

                              if ((NV_DOT(B_W_force,Eta)*signyo3 -
NV_DOT(Dragforce,Eta)*signyo2)*cos(theeta) + (NV_MAG(Liftforce)+
NV_DOT(Dragforce,Xi)+ NV_DOT(B_W_force,Xi))*sin(theeta) < (F_lub - F_grad
- F_vand)*sin(2*theeta))
                              {


                                    fib = fopen ("wall_boundary_hits.txt",
"a");
                                     fprintf (fib, "P_POS(p)[0] is %e\t
P_POS(p)[1] is %e\t P_POS(p)[2] %e\n" , P_POS(p)[0], P_POS(p)[1],
P_POS(p)[2]);
```

```
                                fclose(fib);
                                return DPM_BC_TRAP;
                        }

                        else
                        {

                                return DPM_BC_REFLECT;
                        }

                }


        }

        }

                return DPM_BC_REFLECT;
}

/*////////////////////////////////////////////////////////////////////
/////////////////*/


DEFINE_DPM_DRAG(particle_drag_force, Re, p) // Macro for Drag force
{
        real w;
        double fe, Cd,  drag_force, Vel_diff[3], Vel_diff_mag, Us; //
variable declarations

        double drag_check;
        double vf;

        FILE *f; // file declaration
```

```
        f = fopen ("Mar12_drag.txt", "a"); // file open

        Rep = Re;

        { // ??
        cell_t c = P_CELL(p); // get the cell the particle is currently in
        { // ??
        Thread *t = P_CELL_THREAD(p); // get the thread the particle is
currently in

        fe = (1 + 0.15*pow(Re,0.687)); // friction coefficient (Quan's
Thesis)

        Cd = fe*(24/Re); // Drag coefficient (Quan's Thesis)

        /* Another Drag force function seen in Fluent referenced to a paper*/

/*      if (Re < 0.01)
    {
    drag_force=18.0;
    return (drag_force);
    }
  else if (Re < 20.0)
    {
    w = log10(Re);
    drag_force = 18.0 + 2.367*pow(Re,0.82-0.05*w) ;
    return (drag_force);
    }
  else*/
    /* Note: suggested valid range 20 < Re < 260 */
 /*   {
    drag_force = 18.0 + 3.483*pow(Re,0.6305) ;
    return (drag_force);
    }*/

        //Cd = ( (24/Re)*(1 + 0.1862*pow(Re,0.6529)) ) +  ((0.437353*Re) /
```

```
(7178.74+Re)); // Cd as defined in Fluent

        drag_force = 18.0 * Cd * Re / 24.0 ;
        drag_check =
(M_PI/8.0)*C_R(c,t)*Cd*pow((Re*C_MU_L(c,t)/C_R(c,t)),2);

        Drag_help = drag_check;

        Vel_diff_mag2 = (Re * C_MU_L(c,t)) / (P_DIAM(p) * C_R(c,t));
        Drag_coeff = Cd;

        vf = (Re*(C_MU_L(c,t)/C_R(c,t))/P_DIAM(p)); // this is vf-vp


        } // ??
        } // ??


        fclose(f); // closing file

        return(drag_force); // returning value */

}

/*////////////////////////////////////////////////////////////////////
/////////////////*/

DEFINE_DPM_BODY_FORCE(DPMBF_Lift_and_Gravity, p, i)
{

        /* Calculating Shear Lift Force  */

        FILE *fi;

        double G, particle_dia, Reg, J, e, L_star, L_w, Lift, signG,
Lift_v[3];
```

```
double Us, Gx;
int signe;
int ind;

ind = i;
Gx = 0;

fi = fopen ("Mar12_boundary_Lift_1.txt", "a");

{ // ??
cell_t c = P_CELL(p);
{ // ??
Thread *t = P_CELL_THREAD(p);

particle_dia = P_DIAM(p);


Us = Rep * (C_MU_L(c,t)/C_R(c,t)) / particle_dia;

if (i == 0)
{
G = C_DVDX(c,t) + C_DWDX(c,t);
Gx = G;
}
else if (i == 1)
{
G = C_DUDY(c,t) + C_DWDY(c,t);
}
else
{
G = C_DUDZ(c,t) + C_DVDZ(c,t);
}


if (G > 0)
{
```

```
		signG = 1;
	}
	else
	{
		signG = -1;
	}

	Reg = signG*G*pow(particle_dia,2)/(C_MU_L(c,t)/C_R(c,t)); // find
the value of viscosity again

	e = pow(Reg,0.5) / Rep;

	if (e<0)
	{
		signe = -1;
	}
	else
	{
		signe = 1;
	}



		if (0.1 < (signe*e) < 20)

		{
			J = 0.6765*(1+tanh( (2.5*log10(e)) + 0.191 ))*(0.667
+ tanh(6*(e-0.32)));

		}

		else //else if (e < 0.1) (signe*e) <<1

		{
			J =
-32.0*pow(M_PI,2)*pow(signe*e,5)*log(1/pow(e,2));
```

```
            }


        Lift =
(-9.0/M_PI)*C_MU_L(c,t)*pow(P_DIAM(p)/2,2)*Us*signG*pow((signG*G)/(C_M
U_L(c,t)/C_R(c,t)),0.5)*J;

        Lift_v[ind] = Lift;

        Liftforce[ind] = Lift;



        } // ??
        } // ??

        fclose(fi);

        return (Lift/P_MASS(p));

}

/*////////////////////////////////////////////////////////////////////
/////////////////*/

//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

B. 9 Particle Positions after 2 sec of Their Motion

In Fluent, for steady state discrete phase model, each particle is tracked individually and time step taken for each particle will be different and not constant through out that particles trajectory. To see position of all the particles at a certain instant, some post processing needs to be done.

Fluent generates a file giving time steps taken and the position obtained at each time step for all the particles. A Matlab code can be written to obtain the position of all the particles at a certain instant from this file.

Yuan introduced 20,000 particles over a period of 9 sec in from the outlet ports of the nozzle. These were 4 groups of 5000 particles each.

To see the position of the particles after 2 sec of their introduction into the mold, we need to follow a certain method in Fluent.

Knowing that 20,000 particles were introduced in 9secs into the mold, then assuming constant flow rate of particle introduction, it can be said that only approximately 4440 particles enter the mold in 2 sec. Thus, we have 4 groups of 1110 particles that need to enter in the first 2 sec.

We will make 10 further sub batches for each of the above 4 batches. Particle time steps taken and position obtained in that time step is available for all the particles, throughout the particles at least 2 sec of motion in the mold for these batches.

We would then determine the positions of all the 10 number batches at 2 sec. The 9 number batches would have been introduced with a delay in unsteady state, and therefore would be currently at 1.8 sec of their motion. Similarly, we need to get the positions of the 8 number batches at 1.6 sec and so on.

For further more accuracy, we can discretize the time delay to a lower value of 0.1sec.
Matlab code as shown below is used to perform the function explained above.
//

```
clear

load All_batch_1_files.dat

i = 1;
k = 1;

for m = k : 75045
    if (All_batch_1_files(m,1) == 0)

        k = m;
```

```
            while (All_batch_1_files(k,1) < 2.6)

                time_c(i,1) = All_batch_1_files(k,1);
                time_c(i,2) = All_batch_1_files(k,2);
                time_c(i,3) = All_batch_1_files(k,3);
                time_c(i,4) = All_batch_1_files(k,4);

                k = k + 1;

            end

            i = i + 1;
    end


end

fid = fopen('All_batch_1_try7_files.txt', 'wt');

if (fid == -1)
error('cannot open file for writing');
end

for n = 1 : i-1
fprintf(fid,'%3d %3d %3d %3d\n', time_c(n,1), time_c(n,2),time_c(n,3), time_c(n,4) );
end

fclose (fid);
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```