Metal Process Simulation Laboratory
Department of Mechanical and Industrial Engineering
University of Illinois at Urbana-Champaign
Urbana, IL 61801

Master Degree Thesis

Tiebiao Shi

Continuous Casting Consortium

Report
2001

Submitted to

Accumold
Allegheny Ludlum
AK Steel
Columbus Stainless
Hatch Associates
LTV Steel
Stollberg, Inc

July 2, 2001

EFFECT OF ARGON INJECTION ON FLUID FLOW AND HEAT TRANSFER IN THE CONTINUOUS SLAB CASTING MOLD

BY

TIEBIAO SHI

B.ENGR. Dalian University of Technology, 1989

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Materials Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2001

Urbana, Illinois

Abstract

Three-dimensional computational models of turbulent multiphase flow and heat transfer in the molten steel in a continuous casting nozzle and mold region are developed and analyzed using the commercial package CFX. To increase the accuracy of the inlet conditions for the mold simulation, a nozzle simulation is first done and then its results at the port are input as the inlet conditions in the mold simulation. Experiments are conducted on water models to verify both the single phase and multiphase turbulent flow models both qualitatively and quantitatively. The simulations consistently match measurements in both the water model and steel caster, although flow in the 0.4-scale water model is sometimes different than in the steel caster when gas is present. Thermocouple measurements are conducted in the liquid pool of the steel caster to verify the heat transfer model. The computed temperatures match the measurements. Parametric studies are performed with the multiphase flow model to quantify the important effects of gas fraction, slab width, SEN submergence depth and bubble size on the flow pattern. It is shown that there is more single roll tendency with higher gas fraction, wider slab, smaller SEN submergence depth and smaller bubble size. Single roll flow pattern is more likely than double roll to cause defects such as slivers and pencil pipes due to higher predicted level fluctuation at the top surface and larger downward velocity along the solidified shell. Flow pattern transition might also be detrimental. For a given slab width and SEN submergence depth, there is a critical gas fraction where the flow pattern changes from double roll to single roll. The critical gas fraction appears not to change much with the casting speed. For a given casting speed, the critical gas fraction increases with decreasing the slab width, increasing the SEN submergence depth, and decreasing bubble size.

Acknowledgement

I would like to express my sincere gratitude to my advisor, Professor Brian G. Thomas for his guidance, support and encouragement. It is a grateful experience to work with him. What I have learned from him, including scientific knowledge and attitude to work and life will benefit me greatly in my future. I would also like to thank the National Science Foundation (NSF - Grant # DMI - 98-00274) and the Continuous Casting Consortium (CCC) at the University of Illinois at Urbana Champaign (UIUC) for their support of this research, the National Center for Supercomputing Applications (NCSA) at the UIUC for computing time, AEA technology for use of the CFX 4.2 and 4.3 program, and Accumold, AK Steel. Columbus Stainless Steel, Allegheny Ludlum, Hatch Associates, LTV Steel and Stollberg for their support. Additional thanks are extend to LTV Steel for use of the water model and PIV system and access to PIV and MFC measurements and Armco Inc. for use of temperature measurement equipment for continuous caster.

I would also like to thank all members in the Metals Processing Simulation Lab for their help to my research project.

Table of Contents

List of Tables

List of Figures

Nomenclature

A	Projected area of bubble
B	body force
B_b	birth rate due to break-up
B_c	birth rate due to coalescence
$c_{liq\ gas}^{(drag)}$	inter-phase drag terms
C_D	drag coefficient
C_p	specific heat
d_i	Diameter of bubble i
d_s	Sauter mean diameter of bubble
D_b	Death rate of bubbles in group i
E	log-layer constant for velocity
E_H	log-layer constant for enthalpy
f_{BV}	breakage volume fraction
f_{liq}	volume fraction of liquid phase
f_{gas}	volume fraction of gas phase
$g(v_j:v_I)$	Break-up rate of bubbles of volume v_j into bubbles of volume v_i
h	static enthalpy
H	total enthalpy
K	turbulent kinetic energy
k_{eff}	effective conductivity of molten steel
k_o	molecular conductivity
k_t	turbulent conductivity
p	pressure
Pr_t	turbulent Prandtl number for velocity
Pr_H	turbulent Prandtl number for enthalpy
Q_{ij}	coalescence rate of bubbles between size group i and j
Re_b	bubble Reynolds number
S_{ij}	collision cross-sectional area of bubbles
S_{masss}, $\mathbf{S}_{mom}$	source term in continuity and momentum equations
T	temperature

T_{liq}	liquidus temperature of steel
$\mathbf{V}$	fluid velocity
$\mathbf{V}_{liq}$	velocity of liquid phase
$\mathbf{V}_{gas}$	velocity of gas phase
x	distance in model domain from centerplane to narrow face
y	distance in model domain from centerplane to wide face
y^+	non-dimensional distance from wall
y_o^+	crossover point for velocity
y_H^+	crossover point for enthalpy
z	distance in model domain from top surface to mold exit
Φ	shear production in turbulence models
α	inlet jet angle
ε	turbulent energy dissipation rate
κ	Von-Karmen constant
μ_{eff}	effective viscosity
μ_o	molecular viscosity
μ_t	turbulent viscosity
ρ	fluid density
θ^T	collision frequency due to turbulence mechanism
θ^B	collision frequency due to buoyancy mechanism
θ^{LS}	collision frequency due to laminar shear mechanism
τ_{ij}	contact time of bubble I and bubble j
η	collision efficiency
χ_c	critical dimensionless energy for break-up
ξ	dimensionless size of eddies
λ	turbulence eddy size
σ_K	constant in turbulence models
σ_ε	constant in turbulence models

Section 1

Introduction

Most steel in the world is solidified by continuous casting. Although novel casting processes are being invented, continuous casting will remain as the major process for mass production of steel in the foreseeable future. A schematic of the continuous casting process is shown in Figure 1.1 [1]. Liquid steel is poured into a tundish from ladles. From the tundish, liquid steel flows through a ceramic submerged entry nozzle (SEN), where a slide gate controls the flow rate of the steel, then exits through bifurcated ports into the liquid pool in the mold. To prevent clogging in the nozzle, argon gas is often injected into the nozzle through ceramic pores in the nozzle wall. Steel in the mold solidifies against the water-cooled mold wall and forms a shell. The shell holds the liquid steel pool. Out of the mold, rollers roll on the shell to drag it downward at “casting speed”. The shell grows as the shell goes down and eventually solidifies through all thickness to form a solid slab.

The flow pattern of liquid steel in the liquid pool has a great influence on quality of the continuous cast steel slab. The flow pattern may affect solidified shell thickness by influencing heat transfer behavior in the mold. High temperature liquid steel flows into the mold from the SEN port with high momentum and impinges the narrow face. The high temperature flow has thinning effect on the shell. If the impingement area is too narrow, heat flow at impingement point will be high and the thinning of the shell will be large. Under extreme conditions, the thinner shell will not be strong enough to hold the liquid steel and a costly “breakout” happens [2]. Flow in the mold should be controlled to deliver heat against the narrow face relatively spreadly to reduce the shell thinning effect. The flow pattern may affect liquid flux filling into the gap between the steel shell and the

mold wall and then produce slab surface quality problems such as cracks and high vibration marks [3]. For a single roll flow pattern, the flow hits the top surface with high momentum. Vertical momentum of the flow will lift the level of the top surface, creating a variation in the interface level or “standing wave”. A high standing wave at the mold wall will prevent liquid flux filling the gap between the steel shell and the mold wall. Reducing the flux in the gap will increase the friction force between the mold wall and the solidified shell and cause the non-uniform heat flow which increases thermal stress in the shell. The friction force and the thermal stress together may cause cracks in the shell. The flow pattern also affects the trajectory of inclusions and argon bubbles and cause quality problems such as slivers and pencil-pipes. Small gas bubbles and inclusion particles can stay in the flow for a much long time. A bad flow pattern may cause these small inclusions and bubbles to stay in the liquid pool much longer and increase their chance to be trapped by the growing shell [4].

The flow pattern in the mold is associated with many factors, such as flow behavior at the SEN port, SEN port angle, SEN port geometry, submergence depth of the SEN, gas fraction and bubble size in the liquid flow, casting speed and geometry of the mold. The flow pattern can be changed by adjusting above parameters. Previous work studied the effect of gas flow rate, bubble size, SEN submergence depth, casting speed and mold width on the flow pattern [4-8, 17, 22, 29-31]. Although these studies produced positive results, they are also limited in following respects. First, testing data set is often small Most cases have only 3-4 testing cases in [5] and [7]. The small testing set can not give detailed information to support the conclusions. Second, for the bubble size effect, only single sized bubbles are considered [6, 17, 29]. In fact, gas in the flow is multiple

sized and bubbles interact with each other. Finally, validation of the flow pattern details with measurements is rare. Even where validation was done, it is not quantitative. The blue ink was used to qualitatively validate the flow pattern in the mold [4].

In this study, both single phase and multiphase flow and single phase heat transfer in the mold are developed, validated quantitatively and applied to investigate the flow pattern in the mold. In section 2, various models are developed to describe the single phase flow, the heat transfer in the single phase flow, the multi-phase flow and the multiple-sized bubbles in the multi-phase flow in the continuous casting mold. In section 3, Particle Image Velocimetry (PIV) is used to quantitatively analyze the flow field in a 0.4 scaled water model. The experimental flow field is compared with the numerically simulated flow field to verify the single phase flow model proposed in section 2. In section 4, temperature in the mold is measured by thermocouples. The single phase heat transfer in the mold is modeled by the heat transfer model developed in section 2. The experimental results are compared with the modeling results. In section 5, a 2-phase flow with uniform bubble size is simulated. A total of 8 different bubble sizes distributed in bubble size range, from 0.5 mm to 5 mm, under normal casting conditions [8] is considered. This section gives a clear understanding on how the flow pattern changes with increasing bubble size. In section 6, a 2-phase flow with multiple bubble sizes is simulated by considering possible coalescence and breakup of bubbles existing in the flow. This section studies how casting conditions influence bubble size distribution in the flow and the flow patterns in the mold.

This study is mainly concerned with flow behavior in the mold and how gas bubbles affect the flow patterns in the mold. It aims to provide perspective to quality

problems by analyzing the influence of bubbles on the flow patterns. The computational domain is limited to the mold region of the caster. However, other parts of the continuous casting process, such as the SEN, also influence the flow jet at the port and bubble size. The flow behavior in the SEN has also been studied by Continuous Casting Consortium(CCC) researchers [16]. To address other quality problems, such as cracks, heat conduction and stress in the solidified shell and water-cooled mold wall structure are investigated by CCC researchers [19,20]. All these researches make up a big project to solve the quality problems in the continuous casting. This study is part of this big project.

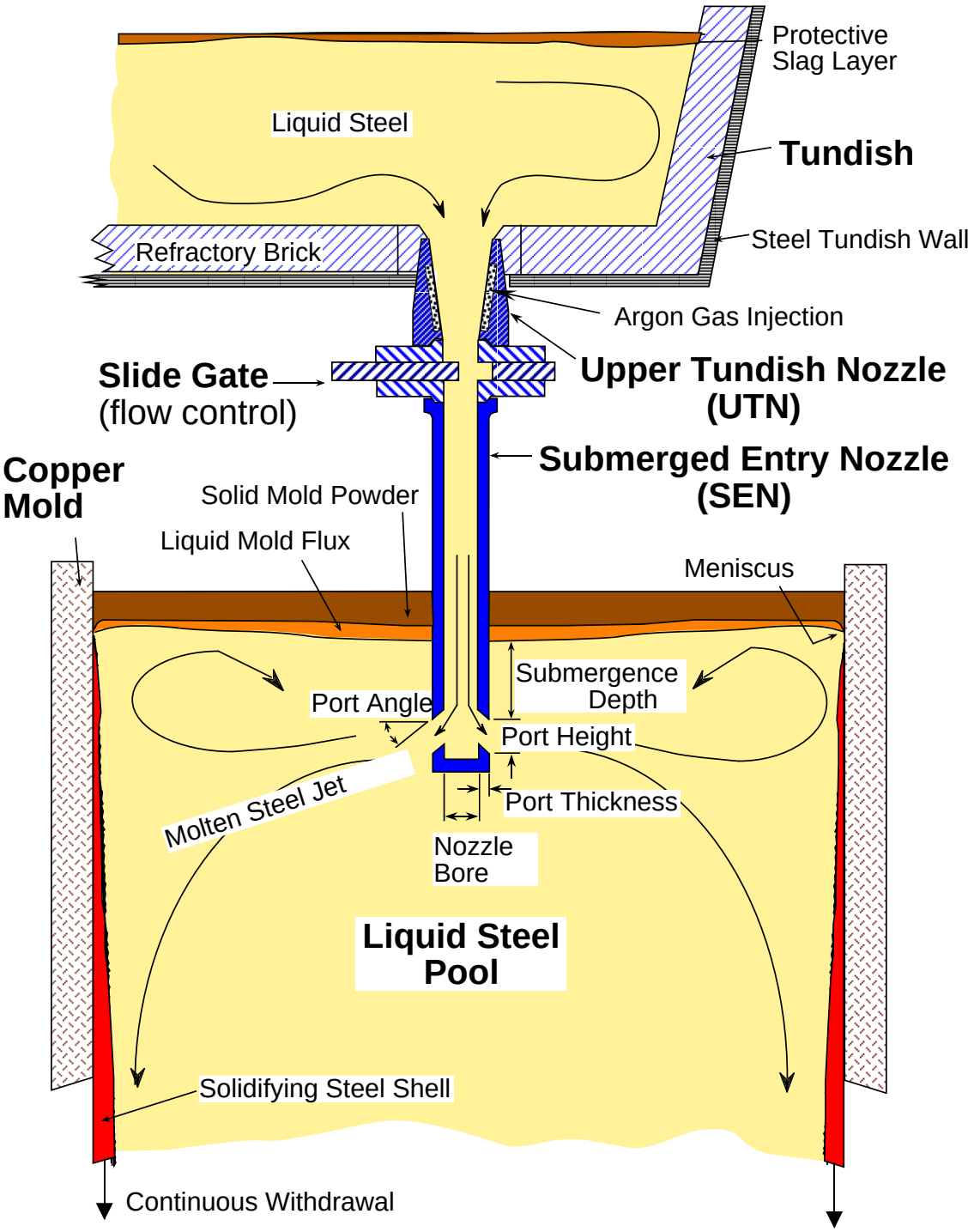


Figure 1.1 Schematic of tundish, nozzle and mold in continuous casting process [1]

Section 2

Model Formulation

2.1 Introduction

In this section, various models will be developed to simulate different fluid flow and heat transfer phenomena in the continuous casting process, including liquid steel (single phase) problems and argon-liquid steel (multi-phase) problems. Each model selects and solves specific governing equations to give an accurate description of the important physics of the specific problems. All these modeling methods and their governing equations are described. The following models are discussed in this section: single phase turbulent flow model, multiphase turbulent flow model, Multi-Size-Group (MUSIG) model and single phase heat transfer model. Boundary conditions in the modeling of continuous casting are also discussed.

Models of single phase turbulent flow, multi-phase turbulent flow with single-sized bubbles and single phase heat transfer are implemented using the commercial package CFX4.2 [9]. The Multi-Size-Group (MUSIG) model is implemented using CFX4.3 [10].

2.2 Single Phase Model

Single phase flow in the water mold and continuous casting mold is turbulent flow [5]. The standard K-ε model is used to simulate the turbulent flow with no sources. The continuity equation for non-source single phase standard K-ε model is:

$$\nabla \cdot V = 0 \qquad (2.1)$$

The momentum equation for non-source single phase standard K-ε model is:

$$\nabla \cdot (\rho V \otimes V) = \nabla\left(\mu_{eff}\left(\nabla V + (\nabla V)^T\right)\right) - \nabla p \tag{2.2}$$

where ρ is the fluid density, V is the mean velocity, p is the pressure, The effective viscosity, μ_{eff} is defined as

$$\mu_{eff} = \mu_o + \mu_t \tag{2.3}$$

where μ_o is the molecular viscosity whose value is given in following sections, μ_t is the turbulent viscosity. Its value is calculated from the K and ε turbulence parameters:

$$\mu_t = C_\mu \rho \frac{K^2}{\varepsilon} \tag{2.4}$$

The turbulent viscosity depends on the turbulent kinetic energy, K, and its rate of dissipation, ε. K and ε are found by solving two transport equations:

$$\nabla \cdot (\rho K \mathrm{V}) = \nabla \cdot \left(\left(\mu_0 + \frac{\mu_t}{\sigma_k}\right)\nabla K\right) + \Phi - \rho\varepsilon \tag{2.5}$$

$$\nabla \cdot (\rho\varepsilon \mathrm{V}) = \nabla \cdot \left(\left(\mu_0 + \frac{\mu_t}{\sigma_\varepsilon}\right)\nabla\varepsilon\right) + C_1 \frac{\varepsilon}{K}\Phi + C_2 \rho \frac{\varepsilon^2}{K} \tag{2.6}$$

where the shear production, Φ, is defined by

$$\Phi = \mu_{eff} \nabla V \cdot \left(\nabla V + (\nabla V)^T\right) - \frac{2}{3}\nabla \cdot V\left(\mu_{eff}\nabla \cdot V + \rho K\right) \tag{2.7}$$

The empirical constants C_1, C_2 ,C_μ,σ_k and σ_ε are given by [9]:

C_1=1.44, C_2=1.92, C_μ=0.09, σ_k=1.00, σ_ε=1.30

2.3 Multi-Phase Flow Model

2.3.1 Governing Equations in Multi-Phase Flow

In most continuous casting process, argon is injected into nozzle to prevent clogging of nozzle. Gas and liquid steel form a 2-phase flow. This 2-phase flow is also turbulent flow. If considering gas bubbles with different size as different phases, steel-argon system could be dealt as multi-phase flow. The standard K-ε model is also used for multi-phase turbulent flow. In the multi-phase model, velocity is solved for each phase separately. K and ε are solved only for the liquid phase. The pressure field is shared for the all phases. The continuity and momentum equations for each phase are different from Eqn.2.1 and Eqn.2.2 in three aspects. First, velocity of each phase is not independent from each other. The velocity fields are coupled by an empirical inter-phase drag model to handle the transfer of momentum between the phases (section 2.3.2). Second, the volume fraction of each phase must be introduced into the governing equations. Third, the buoyancy (section 2.3.3) must be considered.

The continuity equation for the gas phase is:

$$\nabla(f_{gas}V_{gas}) = 0 \tag{2.8}$$

The continuity equation for the liquid phase is:

$$\nabla(f_{liq}V_{liq}) = 0 \tag{2.9}$$

The momentum equation for the liquid phase is:

$$\begin{aligned}&\nabla\cdot(f_{liq}\rho_{liq}(V_{liq}\otimes V_{liq}))\\&=\nabla(f_{liq}\mu_{eff}(\nabla V_{liq}+(\nabla V_{liq})^T))-f_{liq}\nabla p+c_{liq-gas}^{(drag)}(V_{gas}-V_{liq})+B_{liq}\end{aligned} \tag{2.10}$$

The momentum equation for the gas phase is:

$$\nabla\cdot(f_{gas}\rho_{gas}(V_{gas}\otimes V_{gas}))$$
$$=\nabla(f_{gas}\mu_{eff}(\nabla V_{gas}+(\nabla V_{gas})^T))-f_{gas}\nabla p+c^{(drag)}_{gas-liq}(V_{liq}-V_{gas})+B_{gas} \quad (2.11)$$

where f is volume fraction, V is velocity. p is pressure, subscript liq and gas represent liquid and gas respectively. Inter-phase drag terms $c^{(drag)}$ are solved in section 2.3.2. B is the buoyancy.

Because relatively smaller density of gas phase, turbulence in gas phase is neglected. Gas phase is set as laminar flow which is solved by Navier-Stokes equations. Turbulent parameters are solved only for liquid phase:

$$\nabla\cdot(f_{liq}(\rho_{liq}K_{liq}V_{liq}))=\nabla\cdot(f_{liq}(\mu_{0liq}+\frac{\mu_{tliq}}{\sigma_K})\nabla K_{liq})+f_{liq}\Phi_{liq}-f_{liq}\rho_{liq}\varepsilon_{liq} \quad (2.12)$$

$$\nabla\cdot(f_{liq}(\rho_{liq}\varepsilon_{liq}V_{liq}))=$$
$$\nabla\cdot(f_{liq}(\mu_{0liq}+\frac{\mu_{tliq}}{\sigma_\varepsilon})\nabla\varepsilon_{liq})+C_1 f_{liq}\frac{\varepsilon_{liq}}{K_{liq}}\Phi_{liq}-C_2 f_{liq}\rho_{liq}\frac{\varepsilon^2{}_{liq}}{K_{liq}} \quad (2.13)$$

where Φ_{liq} is defined as

$$\Phi_{liq}=\mu_{liq_{eff}}\nabla V_{liq}\cdot(\nabla V_{liq}+(\nabla V_{liq})^T)-\frac{2}{3}\nabla\cdot V_{liq}(\mu_{liq_{eff}}\nabla V_{liq}+\rho_{liq}K_{liq}) \quad (2.14)$$

μ_t is calculated from Eqn.2.4. The values for the five constants C_1, C_2 ,C_μ,σ_k and σ_ε are the same as those given in section 2.2.

2.3.2 Inter-Phase Drag Model for Multi-Phase Flow

In argon-steel multi-phase system, argon bubbles disperse in continuous liquid steel flow. The drag exerted on an argon bubble by moving liquid steel is assumed to arise from two mechanisms only. The first is due to the viscous surface shear stress, and is called skin friction. The second is due to the pressure distribution around the bubble, and is called form drag. The total drag force can be expressed in terms of the non-

dimensional drag coefficient C_D. Using drag coefficient C_D, the inter-phase drag terms $c_{liq\ gas}^{(drag)}$ and $c_{gas\ liq}^{(drag)}$ in the momentum equations of above multi-phase model are defined as:

$$c_{liq\ gas}^{(drag)} = \frac{3}{4}\frac{C_D}{d} f_{gas}\rho_{liq}\left|V_{gas} - V_{liq}\right| \tag{2.15}$$

$$c_{gas\ liq}^{(drag)} = \frac{3}{4}\frac{C_D}{d} f_{liq}\rho_{gas}\left|V_{liq} - V_{gas}\right| \tag{2.16}$$

The drag coefficient C_D is defined as:

$$C_D = \frac{D}{\frac{1}{2}\rho V^2 A} \tag{2.17}$$

where D is the magnitude of the drag force, A is the projected area of the bubble in the direction of flow. For a particle of a given shape, undergoing motion in a Newtonian incompressible fluid, C_D depends only on Reynolds number:

$$\mathrm{Re}_b = \frac{\rho_{liq}\left|V_{liq} - V_{gas}\right| d}{\mu_{o\ liq}} \tag{2.18}$$

The function $C_D(\mathrm{Re}_b)$ is determined experimentally, and is known as the drag curve. In argon-steel system in continuous casting, the bubbles move closely with the liquid, and the bubble Reynolds number is about 125 for a 1 mm bubble. The drag curve for the Allen regime was adopted:

$$C_D = \frac{24}{\mathrm{Re}_b}\left(1 + 0.15\,\mathrm{Re}_b^{0.687}\right) \tag{2.19}$$

The drag coefficient for spherical particles continues to decrease monotonically with Reynolds number.

2.3.3 Buoyancy Term in Multi-Phase Momentum Equations

In the gas-liquid multi-phase system, there is a large difference in density between phases. Bubbles will reach a substantial upward velocity due to the great buoyancy force before they leave top surface. The drag force exerted by bubbles to liquid is substantial as well and will affect liquid flow greatly. Meanwhile, for steel continuous casting, it is hoped all gas float up out of bulk liquid to prevent gas related defects. So, it is important to calculate buoyancy force precisely. Buoyancy force introduced into the K-ε model momentum equations is calculated as:

$$\mathbf{B}_{gas} = \left(\rho_{liq} - \rho_{gas}\right)\mathbf{g} \tag{2.20}$$

where g is the gravity vector {0, 0, 9.8}. The Boussinesq approximation is employed, whereby gas compressibility and thermal effects on ρ are ignored [9].

2.3.4 Source Terms in Multi-Phase Model

In the 2-phase flow in continuous casting, liquid steel and argon gas flow into the mold. Liquid steel solidifies and exits the bottom of the mold. Argon gas escapes from the top surface. Different from previous work [5], in this study, the inlet boundary is set for both liquid and gas phases at the SEN port. No additional source term is set for the gas phase. Because the top surface is set as the wall boundary condition, a sink is set for the gas phase at the top boundary of domain to absorb escaping gas. This sink has no influence on the liquid phase. It adds a mass source and a momentum source to the gas equations at the top surface boundary cells. The equations of the gas phase in other cells except top surface boundary cells remain unchanged. After adding the source term, the

continuity equation (Eqn.2.8) for the gas phase at the top surface boundary cells becomes:

$$\nabla(f_{gas}\rho_{gas}V_{gas}) = S_{mass} \tag{2.21}$$

The mass source term S_{mass} is calculated as:

$$S_{mass} = f_{gas}\rho_{gas}W_{gas}A_w \tag{2.22}$$

where W_{gas} is the upward velocity of gas phase, A_w is the projection area of the cell at the top surface.

The momentum equation (Eqn.2.11) for the gas phase at the top surface boundary cells is:

$$\begin{aligned}&\nabla\cdot(f_{gas}\rho_{gas}(V_{gas}\otimes V_{gas}))\\&=\nabla(f_{gas}\mu_{eff}(\nabla V_{gas}+(\nabla V_{gas})^T))-f_{gas}\nabla p+c_{gas-liq}^{(drag)}(V_{liq}-V_{gas})+B_{gas}+S_{mom}\end{aligned} \tag{2.23}$$

The momentum source term S_{mom} is calculated as:

$$S_{mom} = S_{mass}V_{gas} \tag{2.24}$$

2.4 Multi-Size-Group Model

In the preceding multiphase model, the dispersed phase is assumed to have the same size and shape through the domain. In reality, the size and shape of the dispersed phase are in a range due to break-up and coalescence. Bubble size is decided by break-up rate and coalescence rate of bubbles. To precisely simulate gas bubble behavior in the liquid and their influence on liquid flow, different bubble sizes and break-up and coalescence between them should be incorporated into the model.

The Multi-Size-Group Model (MUSIG) provides an approximation to solve this problem. In multi-size-group model, bubbles are divided into finite size groups. Break-up

and coalescence occur among these groups. Various MUSIG models are developed. In this study, Luo's MUSIG model [11] as implemented into CFX4.3 [10] was used. In this model, the population balance method together with the break-up and coalescence models was incorporated into three-dimensional CFD calculations. All bubble velocities are related to the average value. Only one momentum equation is solved for all bubbles. For more simplicity, the MUSIG model assumes that all bubbles are moving at the same velocity. This makes the momentum equation for bubbles much easier to solve. However, the results are rougher, because collision between bubbles moving with different velocity is avoided. Only turbulent collision is considered. All gas bubbles are solved as gas phase in a whole using same equations presented in section 2.3. At certain intervals, such as 20 iterations, MUSIG model is invoked to change the volume distribution among size groups. The volume redistribution among size groups will change average size of the bubbles.

To calculate bubble size redistribution, mass transfer between size groups must be calculated first. The mass transfer equation for size group i:

$$\nabla \cdot (\rho V n_i f_{gas}) = S_i \qquad (2.25)$$

where S_i is the rate of mass transfer into the size group i due to break-up or coalescence. f_{gas} is volume fraction of gas phase. n_i is the volume fraction of size group i in total gas phase.

The mass transfer rate into size group i is:

$$S_i = \rho v_i (B_b - D_b + B_c - D_c) \qquad (2.26)$$

where B_b is birth rate of bubble i due to breakup, D_b is death rate of bubble i due to breakup, B_c is birth rate of bubble i due to coalescence, D_c is death rate of bubble i due to coalescence. v_i is the volume of a single bubble in group i.

Combining Eqn.2.25 and 2.26, the following equation is obtained:

$$\nabla \cdot (Vn_i f_{gas}) = v_i (B_b - D_b + B_c - D_c) \tag{2.27}$$

B_b, D_b, B_c and D_c are given in following breakup and coalescence models. Then, n_I is obtained.

For each update of size distribution by MUSIG, the average bubble size is changed. So, the average bubble size is calculated every time MUSIG is invoked. The Sauter mean diameter is used as the bubble average diameter. The Sauter mean diameter is defined as:

$$d_s = \frac{1}{\sum_i \frac{n_i}{d_i}} \tag{2.28}$$

d_s is used in Eqn.2.17 and 2.18 to calculate the drag coefficient C_D between the liquid and gas phases. MUSIG model affects momentum equation by changing average bubble diameter due to mass transfer between size groups. Changed average bubble diameter has changed drag force term in momentum equations. The projected area, A, of the bubbles in the direction of flow in Eqn.2.16 is calculated by:

$$A = \frac{6 f_{gas}}{d_s} \tag{2.29}$$

2.4.1 Break-up Model

In this study, the break-up model is based on Luo and Svendsen's break-up model of bubbles [11] as implemented into CFX4.3 [10]. This model is based on the theories of isotropic turbulence and probability and contains no unknown or adjustable parameters. It matches experimental measurements of the water-air system in a high-intensity pipeline flow[11]. In this model, only binary breakage is considered. The break-up rate of bubbles of volume v_j into bubbles of volume v_i $(=v_j f_{BV})$ is expressed as:

$$g(v_j : v_i) = f_B 0.923(1 - f_{gas}^{total})\left(\frac{\varepsilon}{d_j^2}\right)^{\frac{1}{3}} \int_{\xi_{\min}}^{1} \frac{(1+\xi)^2}{\xi^{\frac{11}{3}}} e^{-\chi_c} d\xi \qquad (2.30)$$

where the coefficient f_B is added for calibration of the model. In this study, f_B is chosen to be 1.0 according to Luo and Svendsen [11]. f_{gas}^{total} is total gas volume fraction. The critical dimensionless energy for break-up χ_c is calculated as:

$$\chi_c = \frac{12\left[f_{BV}^{\frac{2}{3}} + (1 - f_{BV})^{\frac{2}{3}} - 1\right]\sigma}{\beta \rho \varepsilon^{\frac{2}{3}} d_j^{\frac{5}{3}} \xi^{\frac{11}{3}}} \qquad (2.31)$$

where f_{BV} is the breakage volume fraction and is defined as:

$$f_{BV} = \frac{v_i}{v_j} \qquad (2.32)$$

where v_i and v_j are volume of bubble i and j. β is a constant (2.0) [11]. σ is surface tension between phases (1.192 for steel-argon system).

The dimensionless size of eddies in the inertial sub-range of isotropic turbulence ξ is calculated as:

$$\xi = \frac{\lambda}{d_j} \qquad (2.33)$$

where d_j is the size of bubble j, λ is the turbulence eddy size and is calculated as:

$$\lambda = (\frac{\mu_0^{\ 3}}{\varepsilon})^{\frac{1}{4}} \tag{2.34}$$

The birth rate of group i bubbles due to break-up of large bubbles is:

$$B_b = \sum_{j=i+1}^{N} g(v_j : v_i) n_j \tag{2.35}$$

The death rate of group i bubbles due to break-up to small bubbles is:

$$D_b = \sum_{j}^{i} g(v_i : v_j) n_i \tag{2.36}$$

where n_i is the volume fraction of bubble i in total gas phase.

2.4.2 Coalescence Model

Bubbles collide with each other due to their different moving velocities. Some of the collisions result in coalescence between them. Coalescence occurs in three steps. First, two bubbles collide. Second, liquid film becomes thinning. Third, if they stay together for an enough long time, the liquid film becomes thin enough to corrupt. Coalescence happens. Based on collision mechanism, the coalescence process could be considered associating to three different mechanism, turbulence, buoyancy and laminar shear. In this study, the coalescence model is based on Prince and Blanch's coalescence model [12] as implemented into CFX 4.3 [10]. The coalescence rate of bubbles between size group i and j is given by the total collision frequency multiplied by collision efficiency:

$$Q_{ij} = (\theta_{ij}^{T} + \theta_{ij}^{B} + \theta_{ij}^{LS}) \eta_{ij} \tag{2.37}$$

where collision frequency θ^T_{ij} , θ^B_{ij} and θ^{LS}_{ij} is corresponding to turbulence, buoyancy and laminar shear mechanism. All three collision frequencies form the total collision frequency. The collision efficiency η_{ij} is calculated as:

$$\eta_{ij} = e^{(-t_{ij}/\tau_{ij})} \tag{2.38}$$

t_{ij} is the required staying time of two bubbles for coalescence:

$$t_{ij} = \left(\frac{r_{ij}^3 \rho}{16\sigma}\right)^{\frac{1}{2}} \ln\frac{h_0}{h_f} \tag{2.39}$$

where h_0 is the initial film thickness and h_f is the critical film thickness where coalescence happens. Value of h_0 and h_f is given by experimental documents. For water-air system [13,14], h_0 is 10^{-4}m and h_f is 10^{-8}m. σ is the surface tension and ρ is density of the gas.

τ_{ij} is the contact time of two bubbles. For the large scale eddy, contact time is calculated :

$$\tau_{ij} = \frac{r_{ij}^{\frac{2}{3}}}{\varepsilon^{\frac{1}{3}}} \tag{2.40}$$

where the equivalent radius r_{ij} is given by:

$$r_{ij} = \frac{1}{2}\left(\frac{1}{r_i} + \frac{1}{r_j}\right)^{-1} \tag{2.41}$$

where r_i and r_j are radius of bubble i and j.

In this study, only one momentum equation is set for gas phase. All bubbles are assumed having same velocity. So collision due to buoyancy and laminar shear are not presented. Only turbulence collision is considered. Meantime, a calibration coefficient f_c, called coalescence coefficient, is added. Eqn.2.37 becomes:

$$Q_{ij} = f_c \theta^T_{ij} \eta_{ij} \tag{2.42}$$

where f_c is given by 0.05 in CFX4.3 [10]. The value may be changed to match different systems and set to 0 to ignore coalescence. The turbulence collision rate is given by:

$$\theta_{ij}^{T} = S_{ij}\left[u_{ti}^{2} + u_{tj}^{2}\right]^{\frac{1}{2}} \tag{2.43}$$

where collision cross-sectional area of bubbles S_{ij} is defined by:

$$S_{ij} = \frac{\pi}{4}\left[d_j + d_i\right]^2 \tag{2.44}$$

Turbulent fluctuating velocity u_t of bubble i and j is calculated as:

$$u_t = 1.4\varepsilon^{\frac{1}{3}} d^{\frac{1}{3}} \tag{2.45}$$

The birth rate of bubble group i due to coalescence of group-j and group-k bubbles is:

$$B_c = \frac{1}{2}\sum_{j}^{i}\sum_{k}^{i} Q_{jk} n_k n_j \tag{2.46}$$

The death rate of bubble group i due to coalescence is:

$$D_c = \sum_{j}^{N} Q_{ij} n_i n_j \tag{2.47}$$

2.5 Heat Transfer Model

Steel continuous casting is a heat processing. Heat transfer occurs continuously in a nearly steady uncompressible fluid flow. For the convenience of solution, the energy transport equation for that flow is solved for the enthalpy H instead of temperature T. Then temperature is derived from enthalpy solution. The energy equation takes the form:

$$\nabla \cdot \left(\rho H V\right) = \nabla \cdot \left(k_{eff} \nabla T\right) \tag{2.48}$$

where T is the temperature and where H is the total enthalpy. Because liquid flow is incompressible flow, velocity has little influence on H. H is equal to static enthalpy h :

$$H = h \tag{2.49}$$

Because overheat of steel in continuous casting is normally less than 100 °C, constant pressure specific heat C_p could be approximated as a constant, the C_p at solidus temperature of steel. Thus, static enthalpy *h* is given by :

$$h = C_p(T - T_{ref}) \tag{2.50}$$

where T_{ref} is the solidus temperature of steel. T is the temperature of liquid steel.

The effective conductivity k_{eff} in Eqn.2.48 is defined as

$$k_{eff} = k_o + k_t \tag{2.51}$$

where k_o is the molecular conductivity and k_t is the turbulent conductivity, which is defined as

$$k_t = \frac{C_p \mu_t}{\mathrm{Pr}_t} \tag{2.52}$$

where Pr_t is the turbulent Prandtl number constant. It is value is given as 1.0 [5]. μ_t is calculated from Eqn.2.4. The values of C_p, T_{ref} and k_o are given in specific cases in following sections.

2.6 Boundary Conditions

2.6.1 Inlet Boundary Condition

The inlet boundary condition is set as a multi-phase inlet. All phases are assumed to have the same velocity at the inlet. The volume fraction of each phase is set at the inlet. The total mass flowing into the mold should equal the total mass flowing out of mold.

The inlet velocity is calculated according to the mass balance of liquid phase. The normal inlet velocity is:

$$V_x = \frac{V_c WN}{A_{jet} f_{liq}} \tag{2.53}$$

where V_x is normal inlet velocity, V_c is the casting speed, W is width of the outlet of the mold, N is thickness of the outlet of the mold, A_{jet} is the effective area of the inlet, f_{liq} is volume fraction of the liquid phase.

Volume fraction f at inlet is calculated from liquid and gas flow rate. Gas is subjected to significant volume expansion during injection because it is heated from room temperature to liquid steel temperature. This expansion must be considered. According to B.G.Thomas and X.Huang [17], the volume fraction of gas f_{gas} is calculated as:

$$f_{gas} = \frac{Q_{gas}\beta}{Q_{liq} + Q_{gas}\beta} \tag{2.54}$$

where Q is the flow rate of liquid or gas. The factor β is defined as:

$$\beta = \frac{T_0 P_\infty}{T_\infty (P_\infty + \rho g L_n)}$$

where T_0 is liquid temperature at the inlet, T_∞ is room temperature, P_∞ is room pressure, ρ is liquid density, g is gravity, L_n is submergence depth of the nozzle. All the above values are given in the specific cases in following sections.

2.6.2 Outlet Boundary Condition

The outlet boundary conditions of both the nozzle and mold are set as pressure boundary condition. A reference pressure, which is set as 0 in this study, is set for

pressure. Neumann boundary conditions, i.e. zero normal gradients, are imposed on all other variables.

2.6.3 Symmetry Boundary Condition

For the symmetry boundary, the velocity component normal to the boundary is set to zero. Zero normal gradient boundary conditions are set to other variables.

2.6.4 Wall Boundary Condition

The wide face and narrow face walls are set to non-slip wall boundary conditions. For continuous caster, the top surface is also set to the non-slip wall boundary condition. In the multi-phase model, a sink is added to the top surface to absorb the escaped gas. The influence of the sink on the governing equations is shown in section 2.3.4. For the water model, the top surface is a free surface. So it is set to the slip wall boundary condition.

Velocity and temperature vary tremendously near the wall. To give accurate enough description of variables near the wall, very fine meshes have to be needed. This requires great memory and time to implement by computer. Wall laws provide an approximate method to avoid this. Wall laws calculate the high gradients of velocity in the boundary layer region using an empirical correlation based on the shear stress at the wall. This allows a coarse mesh to be used and still include the behavior in the boundary layer. The boundary condition for velocity is specified using wall laws in the standard K-ε model. In the following sections, the velocity field is solved using the standard K-ε model. The wall law is used to deal with wall boundary conditions.

In the standard K-ε model, tangential velocity to the wall is calculated as:

$$V_t = \begin{cases} -(C_\mu K)^{1/2} y^+ & y^+ < y_0^+ \\ \dfrac{-(C_\mu K)^{1/2}}{\kappa} \log(E y^+) & y^+ \geq y_0^+ \end{cases} \tag{2.55}$$

where V_t is the tangential velocity to the wall, C_μ is a turbulence model constant (0.09) [16]. K is turbulence kinetic energy. μ_0 is molecular viscosity. n is normal distance to the wall. E is log-layer constant (9.7930) [9]. κ is the Von-Karmen constant (0.419) [9]. The non-dimensional distance normal to the wall is:

$$y^+ = \frac{(\rho^2 C_\mu{}^{1/2} K)^{1/2}}{\mu} n \tag{2.56}$$

The cross over point between the viscous sub-layer and the logarithmic region is:

$$y_0^+ = \frac{1}{\kappa} \log(E y_0^+) \tag{2.57}$$

which obtains the value y^+_0=11.23. The turbulence kinetic energy K is solved in the control volume adjacent to the wall using transport equations for K-ε model. The production terms in the K equation are treated differently so that only quantities interior to the flow and the specified boundary conditions on velocity are used. To catch the profile near the wall, the non-dimensional distance of the first cell should be smaller than y^+_0 in Eqn.2.57 [9].

The turbulence dissipation at the wall is calculated from K using the relation:

$$\varepsilon_{(y^+=0)} = \frac{C_\mu^{3/4} K^{3/2}}{\kappa\, n} \tag{2.58}$$

The boundary conditions for heat transfer calculations are similar to the velocity boundary conditions.

The enthalpy in the wall layer is assumed to be:

$$H = \begin{cases} \dfrac{\mathrm{Pr}_H E_H y^+ \left(\dfrac{\partial H}{\partial n}\right)_{y^+=0}}{\rho C_\mu^{1/4} K^{1/2}}, & for\, y^+ \le y_H^+ \\ \dfrac{\dfrac{\mathrm{Pr}_t}{\kappa} \log\left(E_H y^+\right) \left(\dfrac{\partial H}{\partial n}\right)_{y^+=0}}{\rho C_\mu^{1/4} K^{1/2}}, & for\, y^+ > y_H^+ \end{cases} \tag{2.59}$$

where y_H^+ is the cross over point between the viscous sub-layer and the logarithmic region for enthalpy (31.77)[10], and E_H is the formula of Jayatilleke

$$E_H = E \exp\left[9.0\kappa\left(\left(\frac{\mathrm{Pr}}{\mathrm{Pr}_H}\right)^{0.75} - 1\right)\left(1 + 0.28 \exp\left(-0.007 \frac{\mathrm{Pr}}{\mathrm{Pr}_H}\right)\right)\right] \tag{2.60}$$

where Pr is the fluid Prandtl number ($C_p\mu_o/k_o$) and Pr_H is the Prandtl number for enthalpy(0.9) [10]. E has the same value as that in Eqn2.55.

2.7 Solution Method

A multi-block numerical grid with body-fitted coordinates is used to create the complex geometry of the nozzle and mold domain. The details of geometry of specific cases are discussed in following sections. The governing equations for above models are discretized using the finite difference method and solved using the commercial finite difference program CFX version 4.2 or 4.3 by AEA Technology [9, 10]. The discretized equations are solved using different iterative solution algorithms [9, 10]]. The velocity and volume fraction equations use full field Stone's algorithm. The K and ε equations use line relaxation algorithm. The pressure equation uses preconditioned conjugate gradients

algorithm. The solutions are considered converged when the scaled residuals of the most concerned variables are about 10^{-2}. The scaled residual is the ratio of the residual at the current iteration to that of the second iteration. For single phase simulations, the relaxation factors for velocity, mass, K and ε are normally set to 0.3 and that for pressure is set to 0.8. For multiphase simulations, relaxation factors for pressure and volume fraction are set to 0.8 and relaxation factors for velocity, mass, K and ε are set to 0.005-0.01 for the first 200 iterations and then increased to 0.3 for 1000 iterations or until it is converged. The program is run on the SGI Origin 2000 supercomputer at NCSA at university of Illinois. The CPU time varies with the number of cells and convergence criterion. For a single phase water model (section 3), 90 minutes may be needed to achieve a fully convergence. For a multi-phase steel caster with MUSIG model (section 6), 25 hours may be needed. The CPU time for specific cases are given in following sections when analyzing the cases.

Section 3

Validation of Fluid Flow Models

3.1 Introduction

In the preceding section, model formulation was discussed. This section will validate the single phase turbulent K-ε model by comparing modeling results with experimental measurements. Because velocity measurements are difficult in a real caster, to validate the model, experiments and simulation were conducted on scaled water models. Validation with a water model is reasonable because of the hydrodynamic similarity existing between water and liquid steel which is characterized by Reynolds and Froude numbers [4]. In this section, scaled simulation was also compared with full size simulation to evaluate proportional scaling. An alternative scaling method, Froude scaling, is also evaluated.

3.2 Comparison of Simulations and PIV Measurements

3.2.1 Scaled Water Model Experimental Description

Water model experiments are widely applied in steelmaking industry for two major reasons. First, there is dynamic similarity between the water and steel flow as characterized by the Reynolds and Froude numbers. These two dimensionless numbers account for the relative influences of inertial, gravitational and viscous forces so that the internal fluid flow between the water and steel can be comparable. Second, a water model is easy to operate and allows direct observation of the liquid flow so that simulated steel flow can be evaluated [18].

A full size water model with the same size mold as a real continuous caster is very large and inconvenient to be built in the lab. So scaled water models are often set up for experiments. This study uses a 0.4 scaled water model constructed by LTV steel. The mold dimensions and other details are given in Figure 3.1. There are three round 39-mm diameter outlet holes at the bottom of the mold. Tiny particles (50-100 μm) are added into the water to trace the flow. Particle Image Velocimetry (PIV) technology is used to trace these particles to give velocity field images of the flow. PIV provides practical quantitative turbulence information on a plane section. This makes it easy to compare simulation and water model quantitatively.

In some measurements, air was injected at a proportion same as that in practice to simulate argon injection through the submerged entry nozzle at the real continuous caster. PIV data were processed by computer and compared with simulation results.

3.2.2 Numerical Simulation of 0.4 Scaled Water Model

Numerical simulations of 0.4 proportionally scaled water model used the single phase standard K-ε model. The geometric, dynamic and other parameters in numerical simulations are given in Table 3.2 and Figure 3.1 and are similar to the PIV measurements except for approximating the 39-mm round outlet as 35×35 mm square holes which have the same area. This treatment is reasonable because it has little influence on flow behavior at the upper part of the mold where defects are believed to form. Because of the symmetry of the mold, only one quarter of the mold is analyzed. Figure 3.1 shows the quarter of 0.4 scaled water model. Water goes in from the SEN (Submergence Entry Nozzle) and goes out from the three holes at the bottom. In the

numerical simulation, total number of cells is 41,960. The boundary conditions are shown in Table 3.1. Table 3.2 shows the data used in simulation. The simulation runs 1200 iterations spending 1.56 CPU hours to reach the acceptable convergence. The convergence plot (Figure 3.2) shows that residuals go down to 1% of the initial level. This suggests a good convergence.

3.2.3 Proportional Scaling and Froude Scaling

The water model is set up according to similarity between water and steel flow which is characterized by the Reynolds number and Froude number. To decide the correct flow rates in the scale water model, a straightforward scaling method is proportional scaling which follows $V_{model} = V_{real}(\frac{L_{model}}{L_{real}})$. This is based on matching Reynolds number $\mathrm{Re} = \frac{\rho VL}{\mu}$. Another alternative scaling mechanism is Froude scaling which is based on Froude similarity. Froude number is defined as $Fr = \frac{V^2}{gL}$. According to Froude similarity, $\frac{V_1^2}{gL_1} = \frac{V_2^2}{gL_2}$. So Froude scaling produces $V_{model} = V_{real}\sqrt{\frac{L_{model}}{L_{real}}}$. In this section, Both of these two scaling methods are compared.

Simulations of a 0.4 proportionally scaled water model, a 0.4 Froude scaled water model and a full size water model are implemented in this section. The 0.4 proportionally scaled model and the 0.4 Froude scaled model have the same domain but inlet velocities. Figure 3.1 shows the geometry of the domain for these cases. The data for these three simulations are listed in Table 3.2. Boundary conditions are all same for three cases as listed in Table 3.1. The convergence history for three cases is shown in Figure 3.2. The

residuals for velocities in three cases all go down to less than 2% of original value. This indicates all velocities converged well.

Flow patterns of three cases are compared in Figure 3.3. Flow in Froude scaled water model is more similar with that in full size water model than that in proportionally scaled water model. The magnitude of velocities in Froude scaled water model is larger than that in the proportionally water model because the velocities are scaled by the square root of the ratio between geometries according to Froude method so that its inlet velocity is larger. The positions of the circulation eyes and impingement points are listed in Table 3.3. From the table, the geometric features of flow pattern in the Froude scaled water model are slightly closer to that in the full size water model than that in the proportionally scaled water model.

Figure 3.4 and Figure 3.5 compare the speed along several typical directions between the above three simulations. The speeds are scaled by their respective inlet speeds to provide comparability. The dimensionless speed is defined as $V^*_{dimensionless} = \frac{V}{V_{inlet}}$. The dimensionless distance below the meniscus is defined as $Z^*_{dimensionless} = \frac{Z}{Z_{domain}}$, where Z_{domain} is the height of the domain and Z is the distance below the meniscus. The dimensionless distance from SEN is defined as $X^* = \frac{X}{X_{domain}}$, where X_{domain} is half of the mold width and X is the horizontal distance from SEN. The quantitative comparisons did not show too much different between flows in Froude scaled water model and proportionally scaled water model. Although there is a difference between the flow patterns in the Froude scaled water model and the proportionally scaled water model, the dimensionless velocities remain almost identical at most areas in the

domain. These two different scaling methods do not bring in too much difference to the flow.

3.2.4 Comparison of Numerical Simulation Results and PIV Measurements on 0.4 Scaled Water Model

In the previous section, it shows that proportional scaling (based on Reynolds number) and Froude scaling (based on Froude number) have similar computed results after dimensionless treatment by inlet speed. This simulation uses the same inlet velocities and flow rate as those of LTV water model experiments which adopted Froude scaling. Model predictions are validated in several ways: comparison with flow pattern on the centerplane parallel to wide face, speed and velocities along jet and velocities along centerline between the SEN and narrow face. Figure 3.6 compares the computed flow pattern on the centerplane parallel to the wide face with corresponding PIV measurements. The flow pattern of numerical simulation is qualitatively very similar to that in the water model. The positions of the upper and lower eyes of flow and impingement points for both cases are very close (Table 3.3).

To quantitatively compare the computed and measured flow in the water model, velocities along two lines are compared. Figure 3.7 compares velocities along the center line of the jet. The speed, horizontal velocity and vertical velocity of numerical simulation match those of the water model within +/- 10%. The difference in vertical velocity at the end of the jet between numerical simulation and PIV measurements is because the simulated jet bends up a little bit and the straight line with initial jet angle can not catch the jet at the end. The data at the end does not reflect the feature of the jet. Figure 3.8 shows comparison of velocities along center line between SEN and narrow

face. In this case, only the region above the jet is processed because this region is most concerned. The horizontal velocity changes form negative to positive with increasing depth from the meniscus because of its double-roll flow characteristic. The vertical velocity remains low because the center line goes through the upper eye. Vertical velocity above and below the upper eye is very small (Figure 3.6), Both horizontal velocity and vertical velocity of numerical simulation matches PIV measurements well. The difference in vertical velocity between the numerical simulation and the PIV measurements is due to the offset of the upper eye in the simulation. In the numerical simulation, the center line does not go through exactly the upper eye. So the vertical velocity along the center line is a little higher than PIV measurements. The discrepancies are not larger than 5%.

3.3 Numerical Simulation of Flow in Real Continuous Caster

The water model is often scaled from the real continuous caster to avoid the inconvenience in constructing a full size mold. However, similarity between the flow in the water model and the real caster has not been verified. The fluid is different. The geometry is also different. Thus, the flow in the water model and real caster might be different in some respects. Because of the difficulty in measurements in continuous caster, it is hard to collect adequate accurate data of the flow in the caster. The similarity is hard to be verified directly. However, if the simulations of the water model and the real caster match, that would increase confidence in the scale water model. The relationship is shown in Figure 3.10. If relation 1 and 2 can be validated, then relation 3 and 4 could be inferred based on similarity.

Figure 3.1 shows the geometry of the full size mold of continuous caster. The simulation domain for the real full size mold measures 1825×200×3000 mm, which is 2.5 times larger than water model except for the longer length of 3000 mm. The boundary conditions are the same as those for 0.4 scaled water model except the top surface is set non-slip wall which considers the less mobile flux layer. Data used in simulation are listed in Table 3.2. The simulation runs in 1800 iterations spending 6.24 CPU hours to reach acceptable convergence. The scaled residual plot is shown in Figure 3.2. The convergence plots show that residuals of all variables go down to 3% of the initial level.

Figure 3.9 shows comparison of the flow patterns on the centerplane parallel to wide face of the 0.4 scaled water model simulation with that of the full size continuous casting simulation. The two cases have similar flow patterns. The positions of the upper eye and the impingement point match fairly well (Table 3.3). The position of the lower eye is different because the outlet condition is different. For the water model, the holes are used to drain the water. For the real caster, the whole bottom of the domain is outlet. The total area of the outlet in the water model is smaller than that in the real caster. Therefore, the outlet velocity of the water is relatively larger. This lowers the position of the lower eye in the real caster.

The quantitative comparison of full size continuous casting simulation with 0.4 scaled water model simulation is shown in Figure 3.7 and Figure 3.8. The velocities and speed are scaled by the corresponding inlet velocity to provide comparability. Along the jet direction, the two cases show almost identical results for velocity vectors. Along the center line between the narrow face and the SEN, velocities also match each other well. The largest difference is less than 5%.

From comparison of the full size continuous casting simulation with the 0.4 scaled water model simulation, the two simulations have very similar results. Because the simulation of the 0.4 scaled model matches the PIV measurements as well, the simulation of the full size continuous caster hopefully matches the flow in the real caster.

3.4 Conclusions

1. Numerical K-ε model simulations of flow in the 0.4 proportionally scaled water model match PIV measurements both qualitatively and quantitatively.
2. Dimensionless comparisons of the flow simulations in the 0.4 proportionally scaled water model, full size water model and liquid steel flow in the full size continuous caster show that these flows are very similar. The standard K-ε model which is validated in 0.4 scaled water model could be applied to model a full size continuous caster with reasonable accuracy.
3. Analysis of flow in Froude scaling and proportionally scaling show that these two scaling methods produce negligible differences at least for single phase flow.

Table 3.1 Boundary conditions in numerical simulation

Boundary	Boundary Condition		Boundary	Boundary Condition
Inlet Port	Inlet		Top Surface	Slip Wall
Narrow Face	Non-slipWall		Centerplane	Symmetry
Wide Face	Non-slip Wall		Outlet Holes	Pressure

Table 3.2 Data used in simulations

	0.4 Proportionally Scaled Water Model	0.4 Froude Scaled Water Model	1:1 Water Model	1:1 Liquid Steel Caster
Domain Width W *(mm)*	730	730	1825	1825
Domain Thickness D *(mm)*	80	80	200	200
Domain Height H *(mm)*	950	950	2375	3000
Nozzle Submergence Depth (top surface to top of port)	80	80	200	200
Jet Angle	30° down	30° down	30° down	30° down
Inlet Jet Spread Angle	0°	0°	0°	0°
Inlet Speed, V_c *(m/s)*	0.4239	0.6703	1.0597	1.0597
Inlet Velocity, V_x *(m/s)*	0.3671	0.5805	0.9178	0.9178
Inlet Velocity, V_z *(m/s)*	0.2119	0.3351	0.5298	0.5298
Inlet Turbulent Kinetic Energy, K_o (m^2/s^2)	0.0440	0.0440	0.0440	0.0502
Inlet Turbulence Dissipation Rate, ε_o (m^2/s^3)	0.999	0.999	0.999	0.457
Liquid Density, ρ (kg/m^3)	1000	1000	1000	7.02×10^3
Liquid Viscosity (m^2/s)	1×10^{-3}	1×10^{-3}	1×10^{-3}	5.55×10^{-3}

Table 3.3 Dimensionless position of circulation eyes and impingement point

Dimensionless position (x*, z*)	PIV measurements on 0.4 scale water model	0.4 Froude scaled water model	0.4 proportionally scaled water model	Full size water model	Real caster simulation
Upper eye					
x*	0.638	0.639	0.694	0.643	0.610
z*	0.118	0.113	0.114	0.116	0.095
Lower eye					
x*	0.552	0.634	0.634	0.571	0.559
z*	0.433	0.532	0.413	0.526	0.459
Impingement point					
x*	1.000	1.000	1.000	1.000	1.000
z*	0.231	0.231	0.234	0.229	0.210

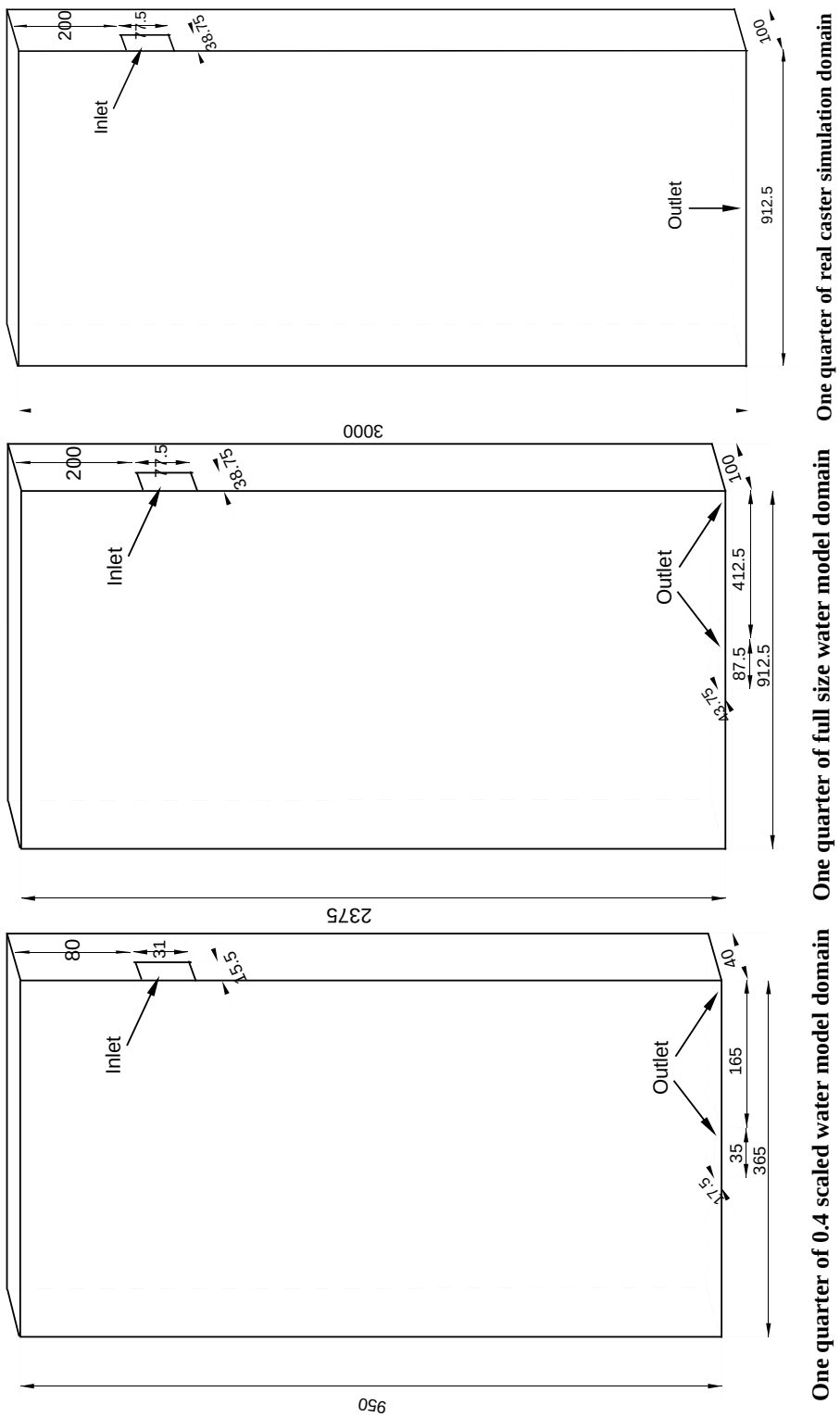


Figure 3.1 Simulation domain for three molds

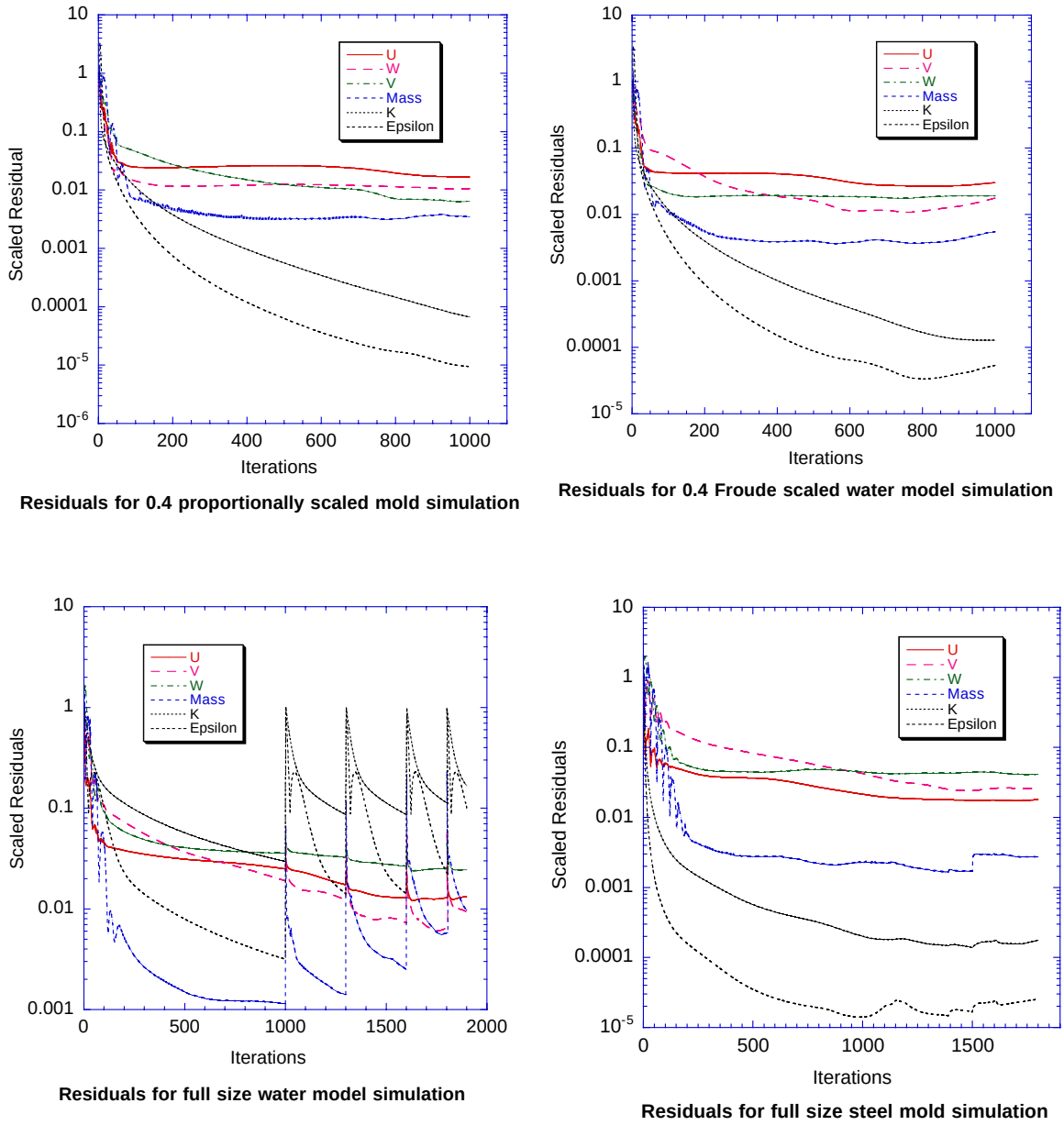


Figure 3.2 Convergence history of simulations of 0.4 proportionally scaled water model, 0.4 Froude scaled water model, full size water model and full size real continuous casting

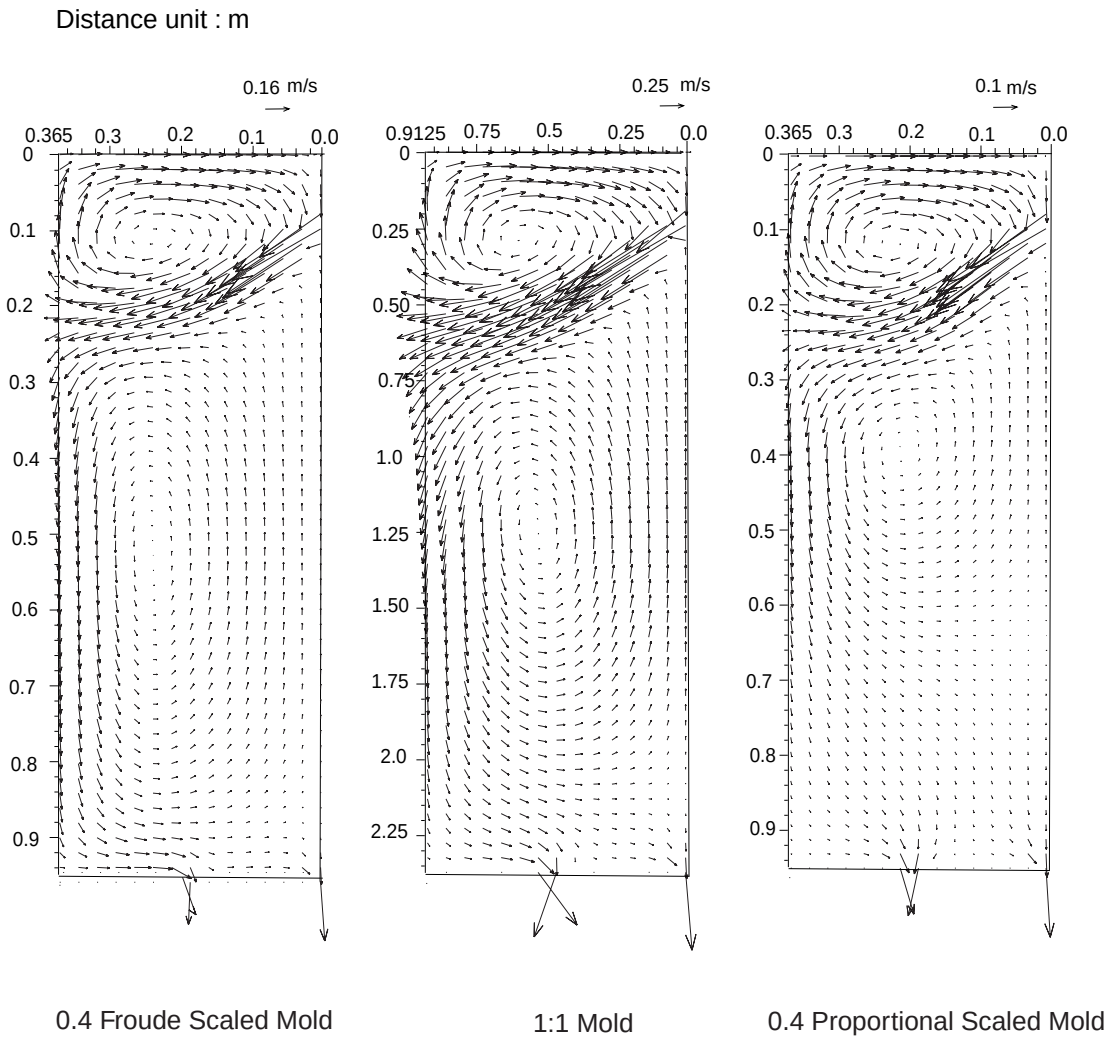
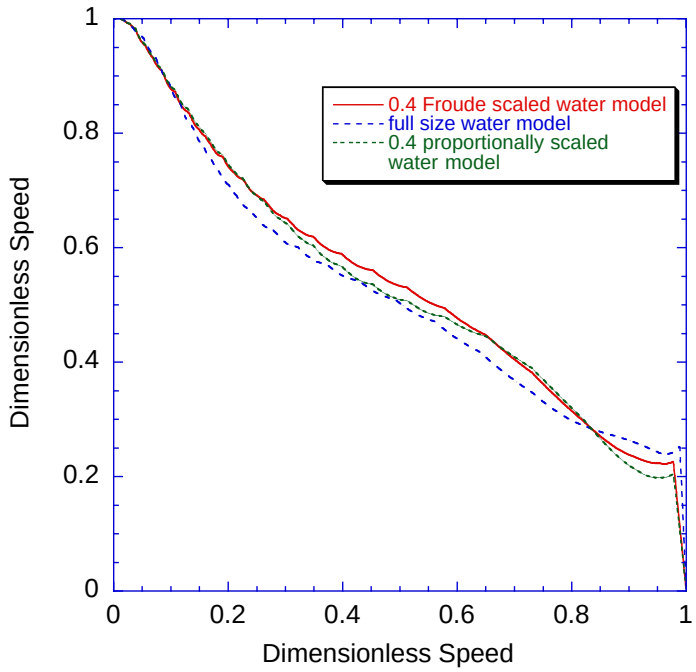
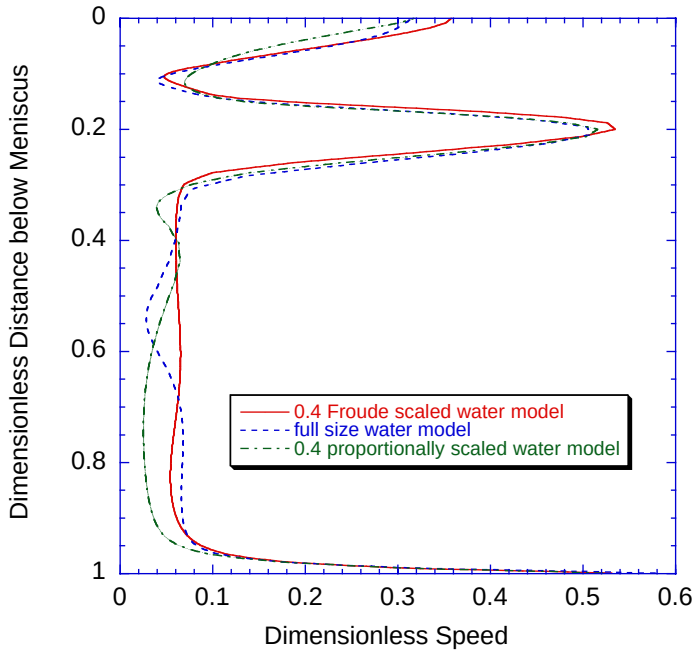


Figure 3.3 Comparison of computed flow patterns between 0.4 Froude scaled, 0.4 proportionally scaled and full size single phase water model



Speed along Jet Direction



Speed along Center Line between SEN and NF

Figure 3.4 Comparison of speed along jet and centerline between 0.4 Froude scaled, 0.4 proportionally scaled and full size water model

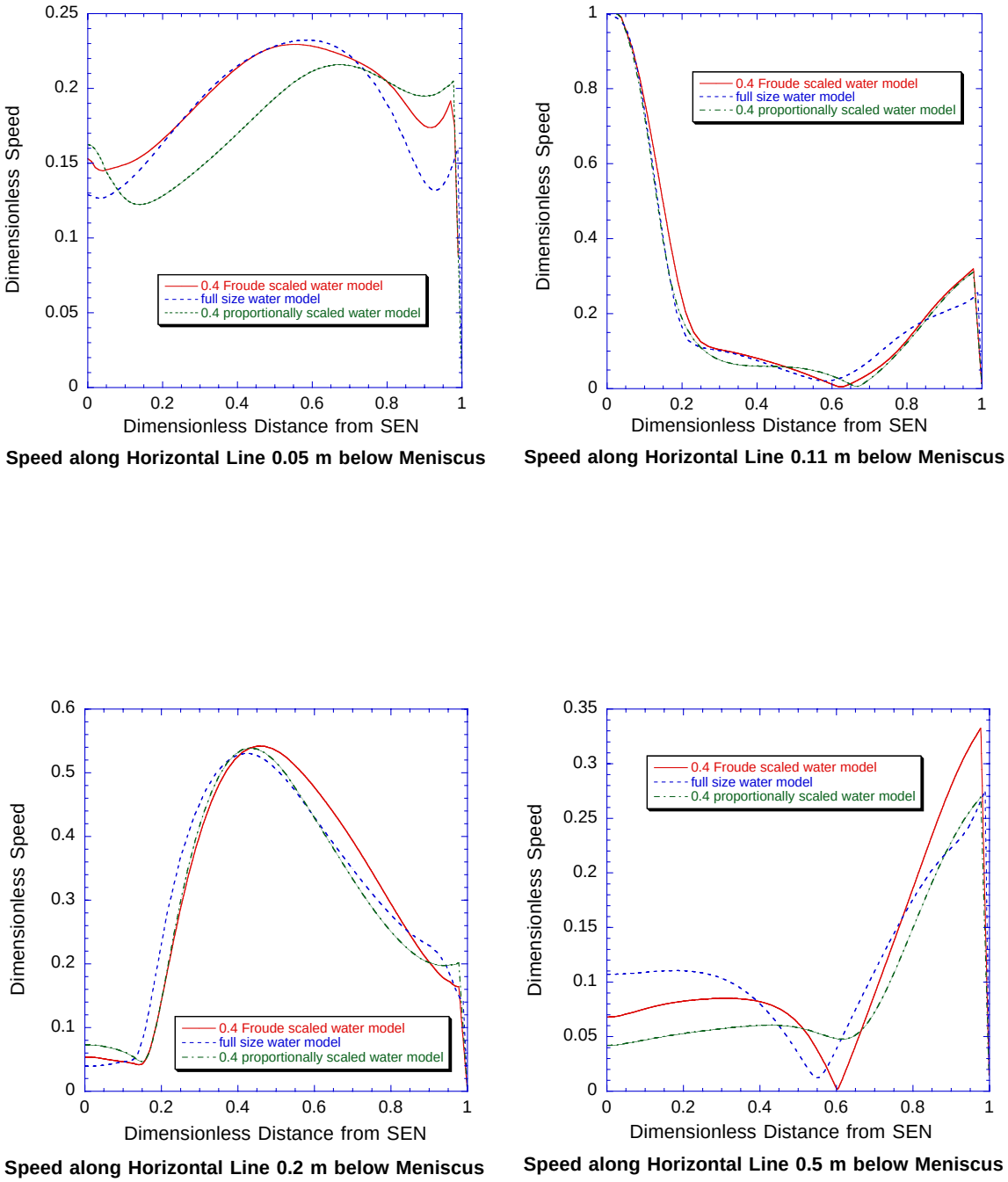


Figure 3.5 Comparison of speed along horizontal lines between 0.4 Froude scaled, 0.4 proportionally scaled and full size water model

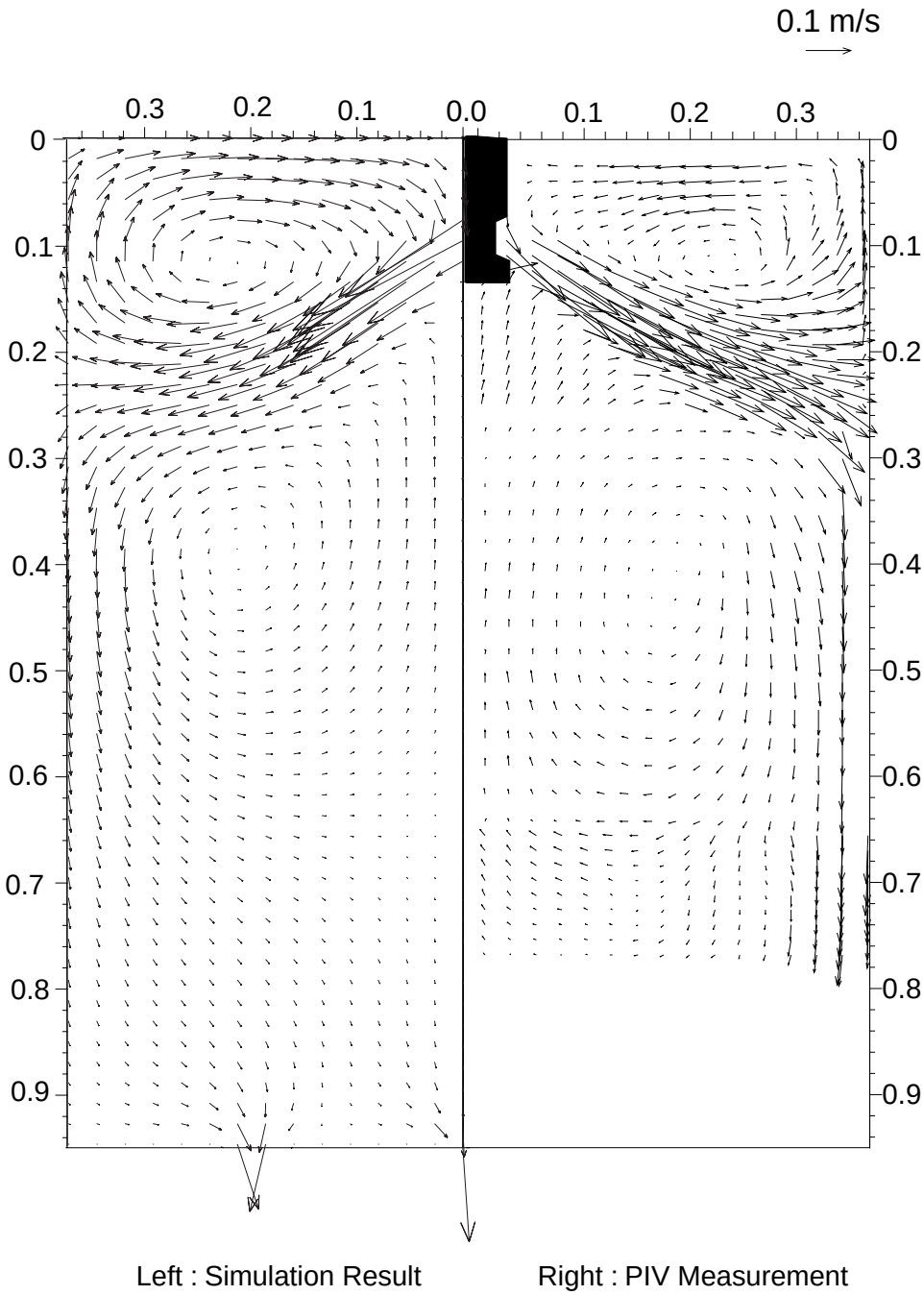
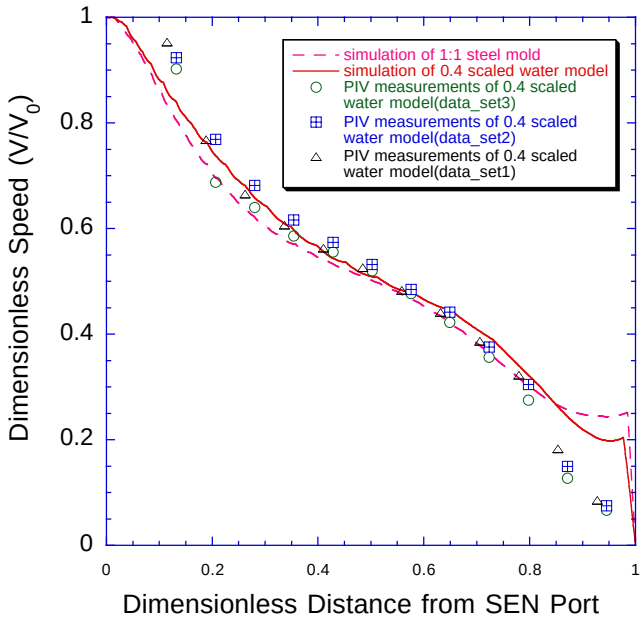
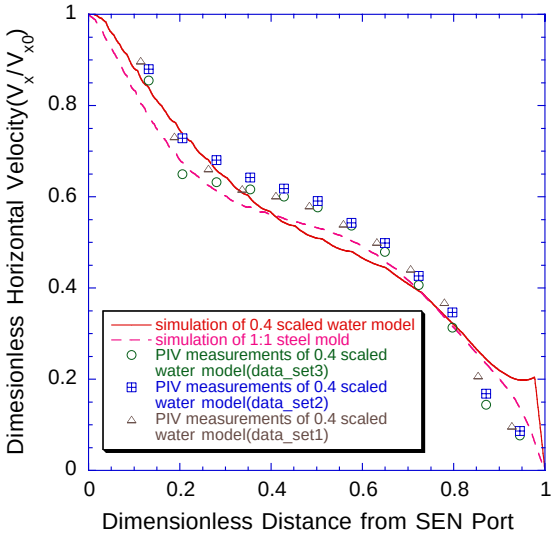


Figure 3.6 Comparison of flow patterns between
PIV measurements and simulation of 0.4 scaled water model



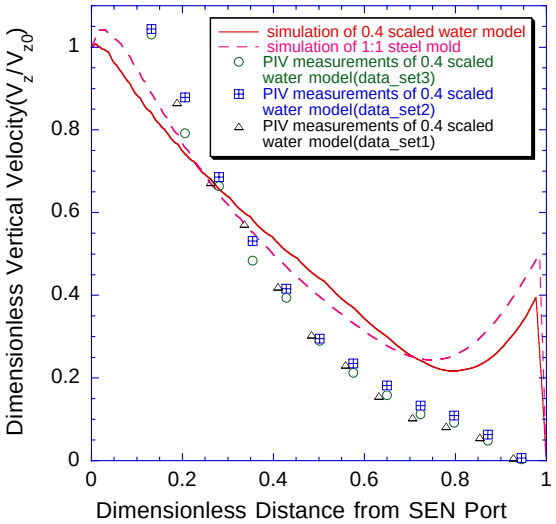
Speed along Jet Direction

(V_0=0.4239m/s for 0.4 scaled model,
V_0=1.0597m/s for real caster)



Horizontal Velocity along Jet Direction

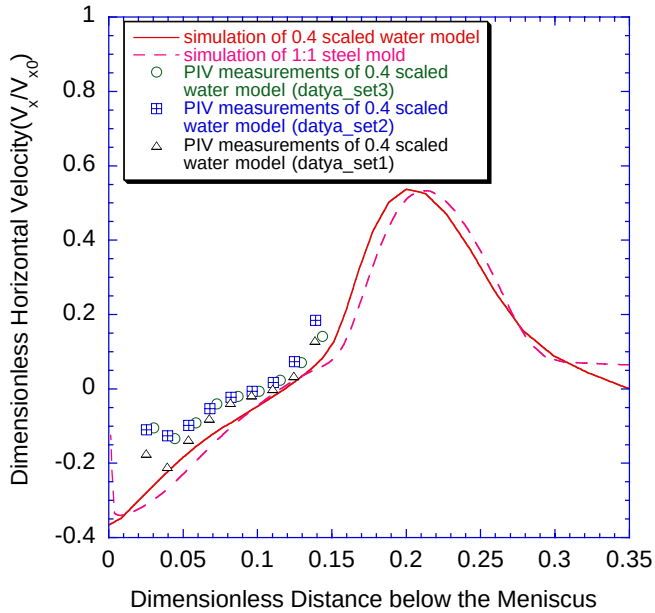
(V_{x0}=0.3671m/s for 0.4 scaled model,
V_{x0}=0.9178m/s for real caster)



Vertical Velocity along Jet Direction

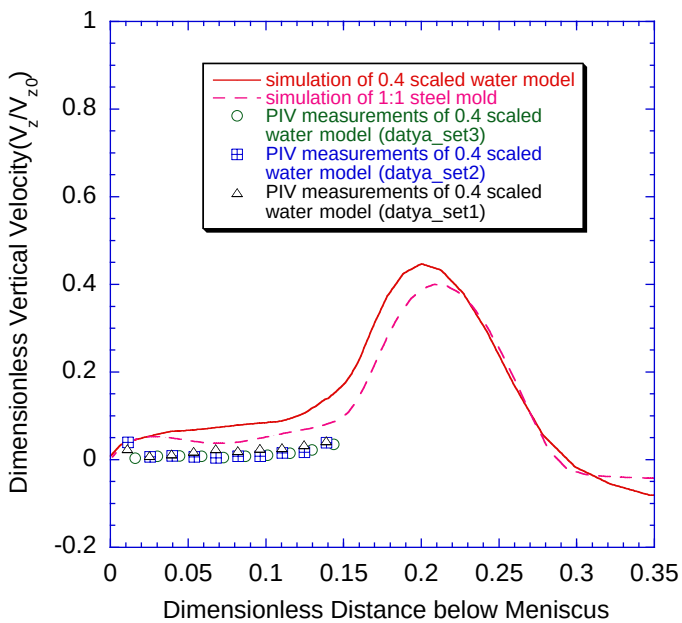
(V_{z0}=0.2119m/s for 0.4 scaled model,
V_{z0}=0.5298m/s for real caster)

Figure 3.7 Comparison of velocities along jet between simulation and PIV measurements of 0.4 scaled water model and real caster



Horizontal Velocity along Center Line between SEN and NF

(V_{x0}=0.3671m/s for 0.4 scaled model,
V_{x0}=0.9178m/s for real caster)



Vertical Velocity along Center Line between SEN and NF

(V_{z0}=0.2119m/s for 0.4 scaled model,
V_{z0}=0.5298m/s for real caster)

Figure 3.8 Comparison of velocities along centerline between simulation and PIV measurements of 0.4 scaled water model and real caster

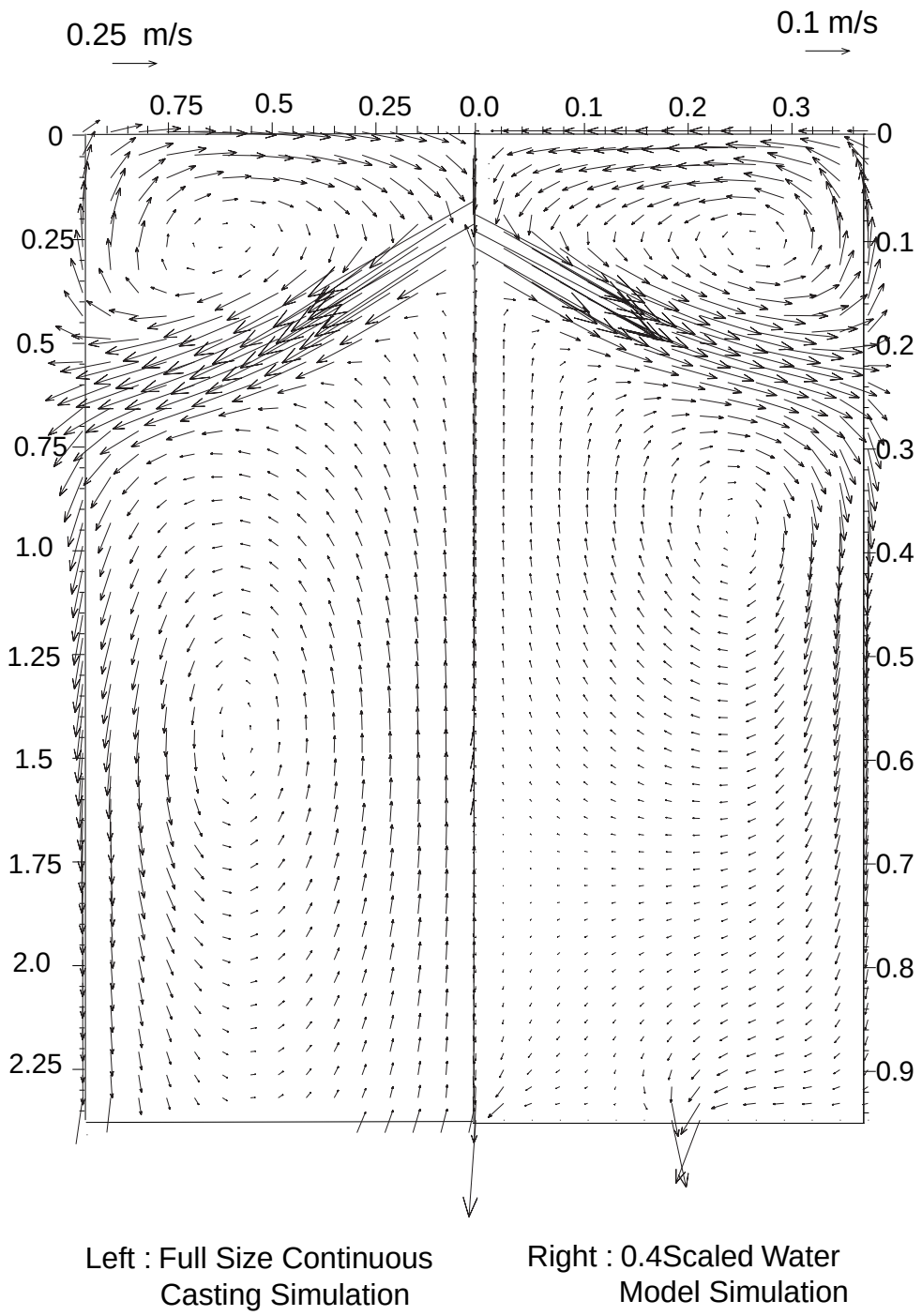


Figure 3.9 Comparison of flow patterns between simulations of continuous casting and 0.4 proportionally scaled water model

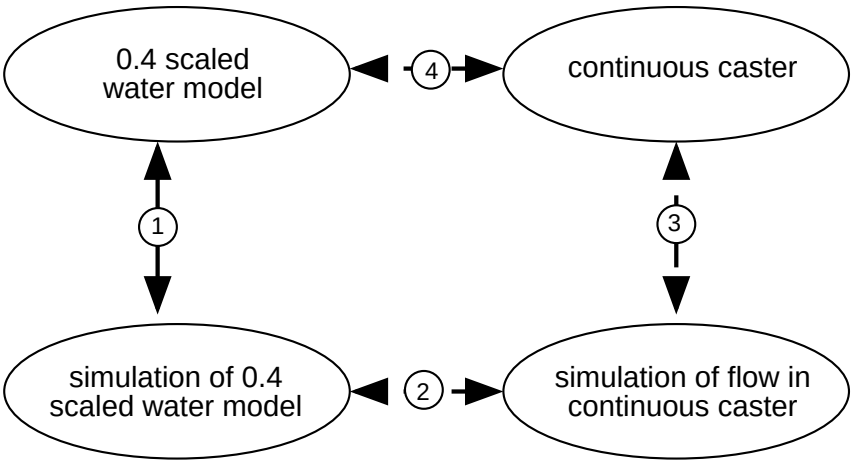


Figure 3.10 A schematic of relationship between numerical simulation, scaled water model and real caster

Section 4

Validation of Heat Transfer Models

4.1 Introduction

Liquid steel is poured into the mold at a temperature above the liquidus. The sensible heat contained in the liquid steel represented by this temperature difference is known as the superheat. The superheat must be transferred out of the steel mass before it can solidify. The heat transfer behavior in the mold affects solidified microstructure and formation of defects, such as meniscus hook formation, breakouts, centerline segregation etc. A good understanding of heat transfer in the mold may help to improve product quality.

Heat transfer in continuous casting is associated with flow in the mold. The complex nature of heat transfer in the continuous casting mold and the high temperature phenomena make it very difficult to analyze it by traditional methods such as water model and plant measurements. Numerical simulation provides a feasible method. In this work, assuming flow and heat transfer is steady in the liquid pool, temperature is computed by solving the single phase Navier-Stokes equations and corresponding Eulerian heat conduction equation (see section 2.4). The heat flux on the wide face and narrow face is also computed based on the temperature gradient. By adjusting factors such as nozzle geometry, thickness of the powder/flux layer or other parameters, the temperature distribution in the liquid pool and the corresponding heat flux to the shell may be changed as desired. These changes are easy to make and evaluated in the mathematical simulation before expensive experiments are conducted.

In this study, three-dimensional turbulent flow and the associated heat transfer is modeled with the finite difference code CFX4.2, using the standard K-ε model with a user defined wall law. To validate the results of the simulation of heat transfer, temperature at different positions in the upper liquid pool is measured in an operating caster and compared with the simulation.

4.2 Experiments

Temperature distribution in the liquid steel pool is difficult to measure. In the experiment in this work, an apparatus was constructed by AK Steel's Research Center in Middletown, OH to measure vertical temperature profiles in the liquid pool [21]. The apparatus allows a thermocouple to go down and up vertically at a slow speed (0.6mm/sec) so that thermal equilibrium could be approached at the thermocouple tip. The maximum insertion depth of the thermocouple is 180mm. Temperature signals are transmitted to a computer for data processing. The computer stored the temperature associated with insertion depth. Each cycle of insertion and withdraw makes a single data set. The instrument schematic is shown in Figure 4.1.

Figure 4.2 shows a typical set of the measured data. From the changes of the temperature gradient (slope of this plot), it is possible to identify the powder/flux and flux/steel boundary. This is because these different materials have different heat conduction coefficient. The temperature slope in different materials should have steepness inversely proportional to their different thermal conductivity. In Figure 4.2, slope 1, 2 and 3 should represent powder, flux and steel respectively. From Figure 4.2,

the position of top surface of steel could be determined. This method is used in data analysis in this section.

Table 4.1 lists the common parameters used in the experiments. All other experimental conditions are listed in Table 4.2. The first five measurements have the same submergence depth (130 mm) and represent different positions in the mold. The other two measurements repeat at two positions with different submergence depth (159 mm). To compare measurements under the same experimental conditions, only the first five measurements are considered. All five points are located along the center plane between the wide faces. The 5 locations are 50 mm from SEN, 125 mm from SEN, the middle point between SEN and the narrow face, 125 mm from the narrow face and 50 mm from the narrow face (Figure 4.3). The pouring temperature is about 1559°C (2838 F). The casting speed is 25.4mm/sec (60 inch/min).

4.3 Numerical Simulation of Heat Transfer

Modeling of heat transfer in the liquid pool is based on the results of modeling of flow. The commercial package CFX4.2 is used to solve the Navier-Stokes equations (Eq.2.10 in section 2) and corresponding Eulerian heat conduction equation (Eq 2.25). The model standard K-ε model with user defined wall law (described in section 2.5.4) is used. Considering the symmetry of the mold, one quarter of the mold is modeled and the solidified shell is not considered. The geometry was set up by D. Creech in previous work [5]. The boundary conditions along the narrow faces and wide faces are set as constant temperature, non- slip walls. The temperature of the wall is set at the liquidus temperature of the steel corresponding to the solidification interface between the solidified shell and the liquid. The top surface is set as the constant heat flux wall

corresponding to the assumption of constant thickness of powder layers. The inlet temperature is set to the pouring temperature. The outlet temperature is set at the liquidus temperature and Table.4.3 lists parameters used in heat transfer simulation. Figure 4.4 shows the convergence history of the heat transfer simulation. The residual of enthalpy reflects convergence level of temperature. In Figure 4.4, the residual of enthalpy is reduced by 4 orders of magnitude. This means temperature converged to an excellent level.

4.4 Modeling and Experimental Results

The velocity on the center plane between wide faces is shown in Figure 4.5. The predicted temperature distribution on the center plane based on the flow modeling is shown in Figure 4.6. Figure 4.7 to Figure 4.11 compares the experimental and predicted temperatures at those positions. Each figure shows the entire temperature signal recorded by the thermocouple from the beginning to the end of each experiment and the corresponding superheat. The measurement at 125 mm from the narrow face has a lower temperature than the other measurements (Figure 4.10) because the ladle was being changed during the measurement and temperature in the tundish is only 1551°C, 8°C lower than normal. The thermocouple hit the shell and broke while measuring at 50 mm from the narrow face so temperature was recorded in only one direction. Measurements were lost during other experiments also (Figure 4.7). From Figure 4.7 to Figure 4.11, the predicted temperature matches the measurements well. The largest difference is less than 4°C, which is within a 7% error with respect to superheat.

In Figure 4.6, two jets from the SEN create a high temperature zone respectively. The right jet impinges the narrow face and forms a peak of temperature and heat flux on

the narrow face. Then the jet splits into two flows. One goes down to form a relatively high temperature area along the narrow face. Another flow goes up to meet meniscus and heat up upper part of the liquid pool. The lowest temperatures appear at the upper and lower recirculation eyes. The corner of the meniscus and the narrow face is another low temperature zone. This low temperature zone is dangerous because hooks might form in this area if the liquid temperature drops below the solidus temperature of steel. Flow parameters should be adjusted to prevent this happening.

Heat flux along center line of the narrow face could be used to calculate shell growth by shell solidification model CON1D [15]. Comparison of measured and predicted shell growth can validate prediction of heat transfer [21]. Figure 4.12 shows the predicted heat flux along three vertical lines. From the plot, most of the superheat is conducted through narrow faces. The peak superheat flux is about 1500 kw/m^2.

4.5 Conclusions

1. In this study, an efficient model of heat transfer for 3-D turbulent flow in a thin slab caster has been developed. This model uses the standard K-ε model, the Eulerian heat conduction equation and a wall law to simulate fluid flow and heat transfer.
2. Temperature along 5 vertical lines on the center plane between wide faces of the liquid steel pool is measured with thermocouple probes.
3. Comparisons of measured and predicted temperature show that the modeling results match experimental results within 7%. The heat transfer model developed in this section can be applied to simulate temperature in the liquid pool and heat flux through solidified shell.

Table.4.1 Parameters of mold and nozzle

Parameters	**Regular Value**			
Mold Width (mm)		Total	Left	Right
	Top	970.25	485.11	485.14
	Bottom	958.85	479.44	479.41
Mold Thickness (mm)	132.08			
Casting Speed (mm/sec)	25.4			
Gas Inflow Rate	No Gas			
Nozzle Port Height (mm)	77			
Nozzle Port Width (mm)	32			
Nozzle Wall Thickness (mm)(near port)	17.5			
Nozzle Bore Inner Diameter (mm) (near port)	70			
Nozzle Submergence Depth (mm)	129.54			
Nominal Vertical Angle of Port Edges	15°C			
Inlet Jet Spread Angle	0°C			

Table.4.2 Experimental parameters

No.	Position of Measurement	Pouring Temperature	Submergence Depth	Thickness of Powder (mm)	Thickness of Flux (mm)	Casting Speed
1	Midway of NF vs SEN (295 mm from CL)	1832 K (2838 F)	129 mm (5.1 inch)	60	6	25.4 mm/s (60 IPM)
2	50mm from SEN (150mm from CL)	1831 K (2836 F)	129 mm (5.1 inch)	62	10	25.4 mm/s (60 IPM)
3	50mm from NF (440 mm from CL)	1831 K (2836 F)	129 mm (5.1 inch)	83	5	25.4 mm/s (60 IPM)
4	125mm from SEN (225mm from CL)	1831 K (2836 F)	129 mm (5.1 inch)	68	7	25.4 mm/s (60 IPM)
5	125mm from NF (365mm from CL)	1824 K (2824 F)	129 mm (5.1 inch)	68	5	25.4 mm/s (60 IPM)
6	50mm from NF (440mm from CL)	1826 K (2827 F)	159 mm (6.25 inch)	28	5	25.4 mm/s (60 IPM)
7	Midway of NF vs SEN (295 mm from CL)	1828 K (2830 F)	159 mm (6.25 inch)	34	7	25.4 mm/s (60 IPM)

Table.4.3 Parameters used in heat transfer simulation

Mold Width	984 *mm*
Mold Thickness	132 *mm*
Nozzle Submergence Depth (top surface to top of port)	127 *mm*
Nozzle port Angle	15° *down*
Inlet Jet Spread Angle	0°
Casting Speed, V_c	25.4*mm/s*
Inlet Velocity, V_x,V_y	0
Inlet Velocity, V_z	0.857 *m/s*
Inlet Turbulent Kinetic Energy, K_o	0.00425 m^2/s^2
Inlet Turbulence Dissipation Rate, ε_o	0.020 m^2/s^3
Liquid Steel Density, ρ	7020 kg/m^3
Laminar Conductivity, k_0	26 $kg\text{-}m/\ s^3\text{-}K$
Specific Heat, C_p	680 $m^2/\ s^2\text{-}K$
Laminar Viscosity, μ_0	0.0056 *kg/m s*
Pouring Temperature	1832 *K*
Liquidus Temperature	1775 *K*
Heat Flux from Top Surface	12,000 W/m^2

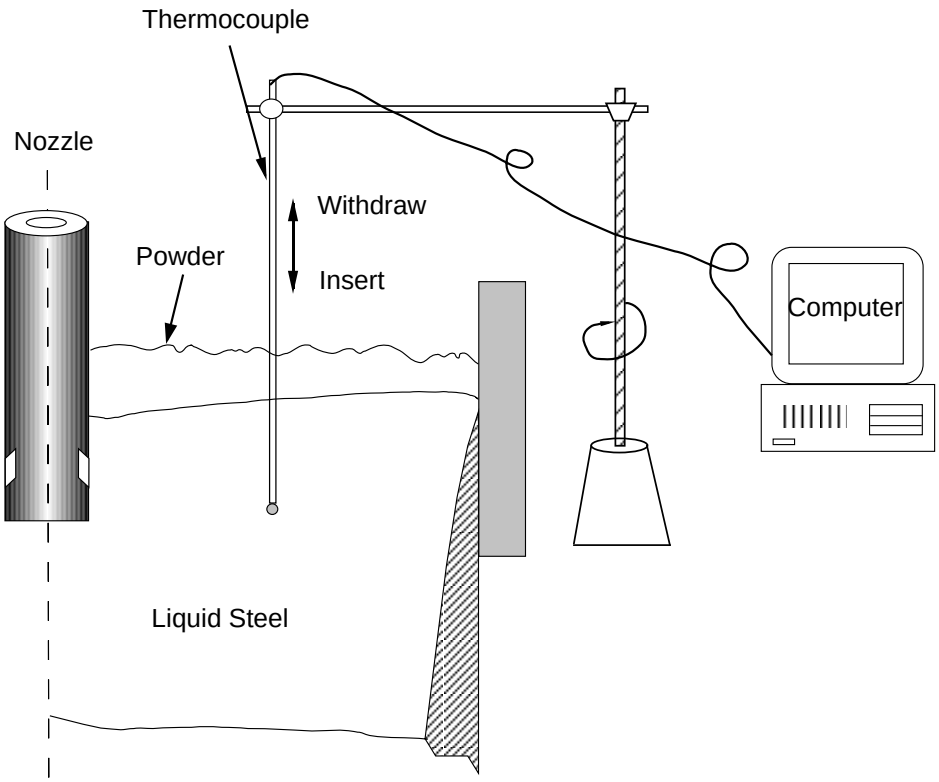


Figure 4.1 Equipment used in measuring temperature in liquid pool

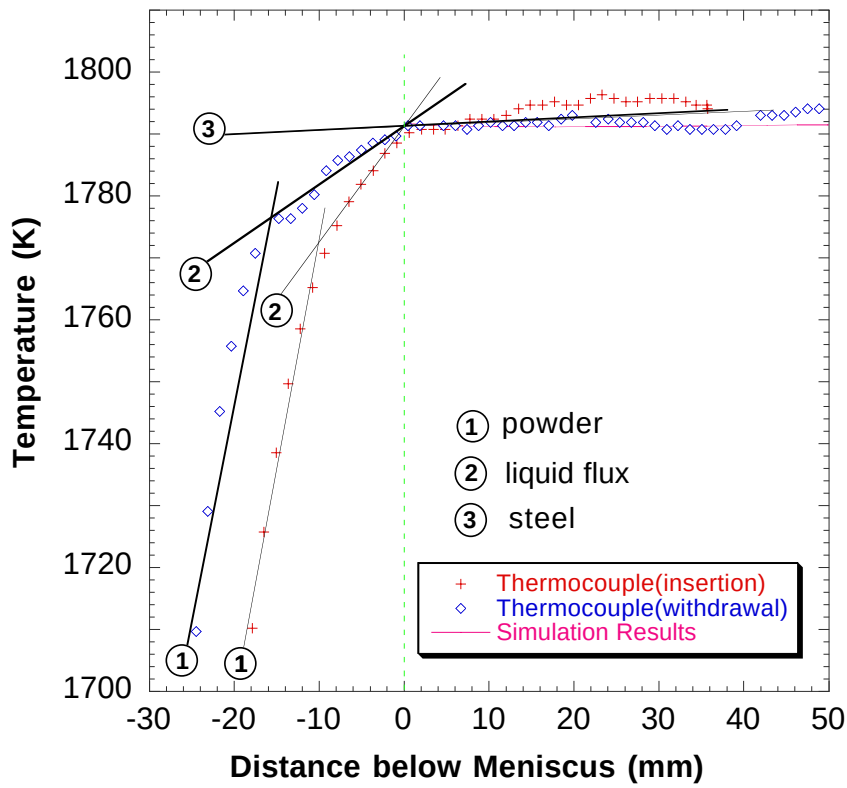


Figure 4.2 Temperature slope of three different material layers

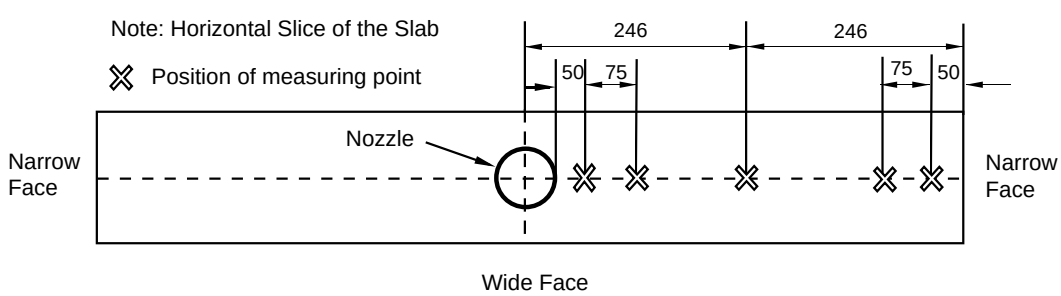


Figure 4.3 Positions of measuring points

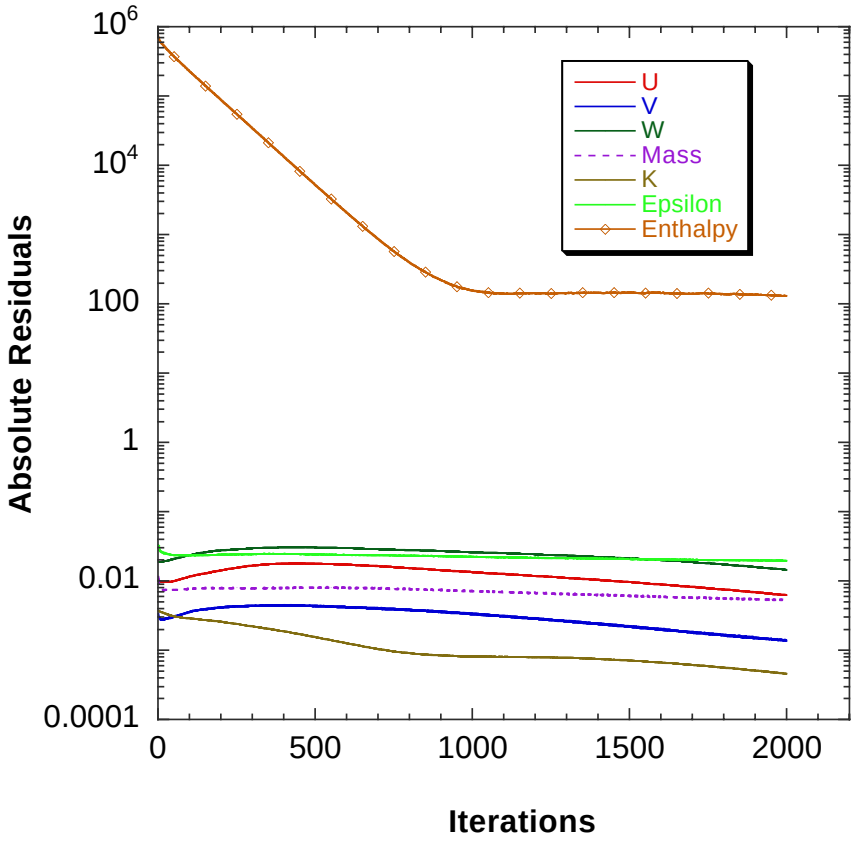


Figure 4.4 Residuals of heat transfer modeling

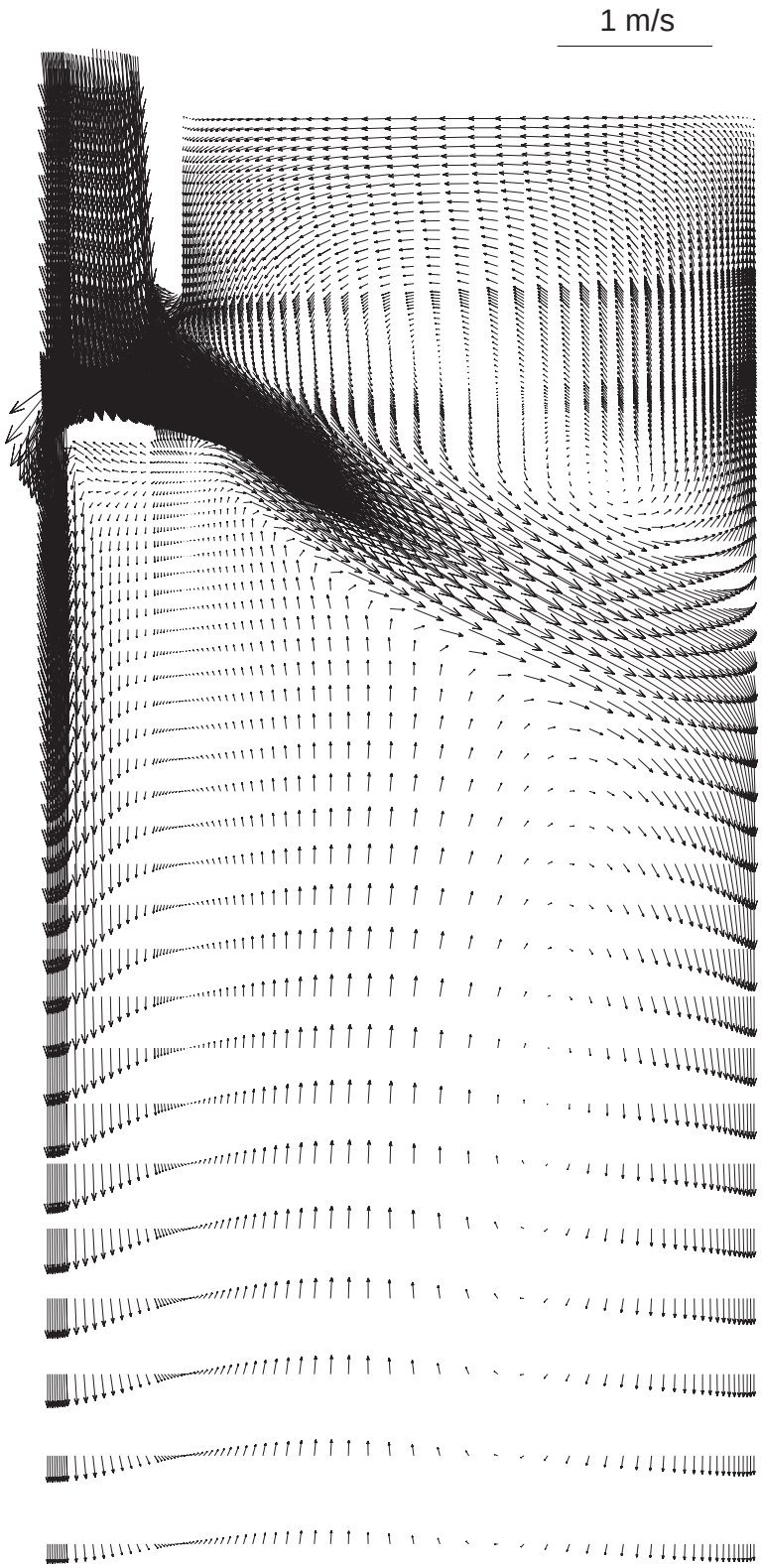


Figure 4.5 Predicted velocity on center plane between wide faces

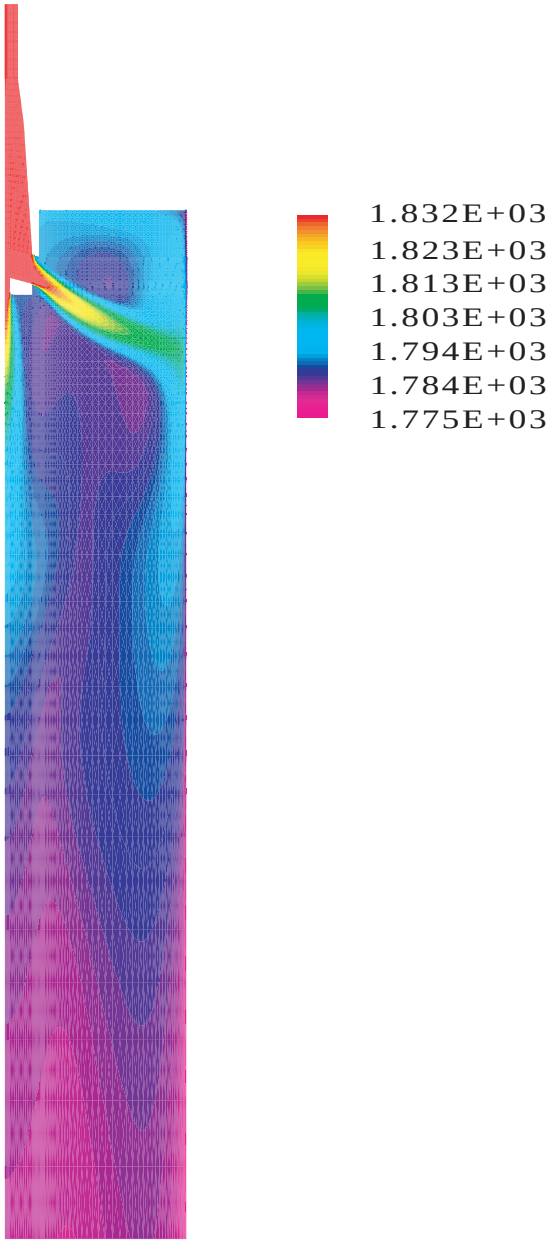
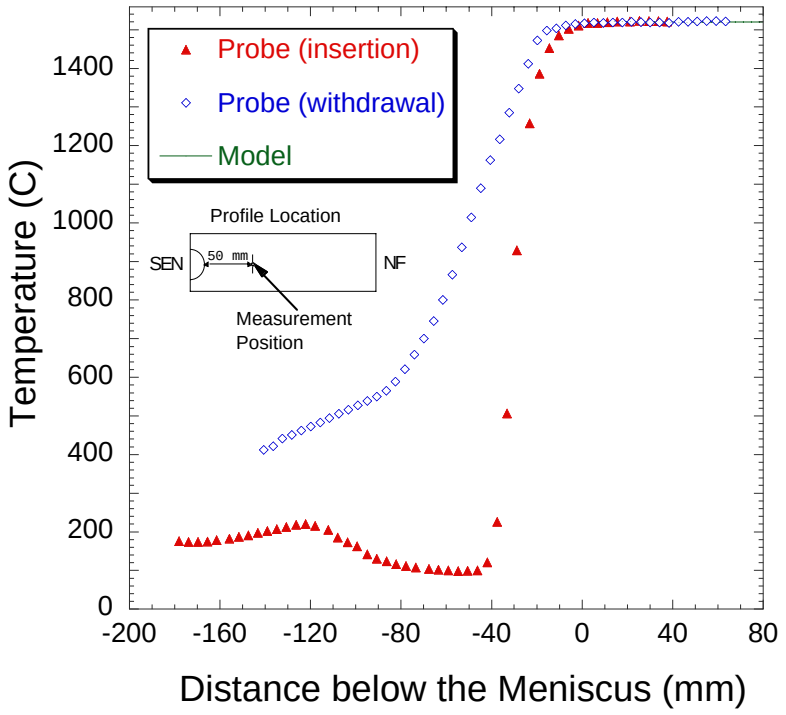


Figure 4.6 Predicted temperature on center plane between wide faces



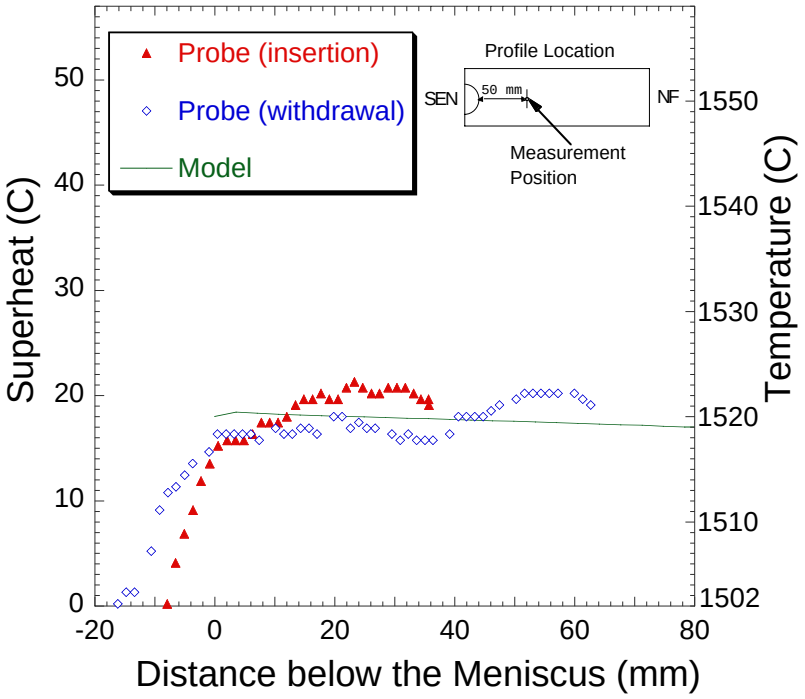


Figure 4.7 Measurement and modeling at 50 mm from SEN

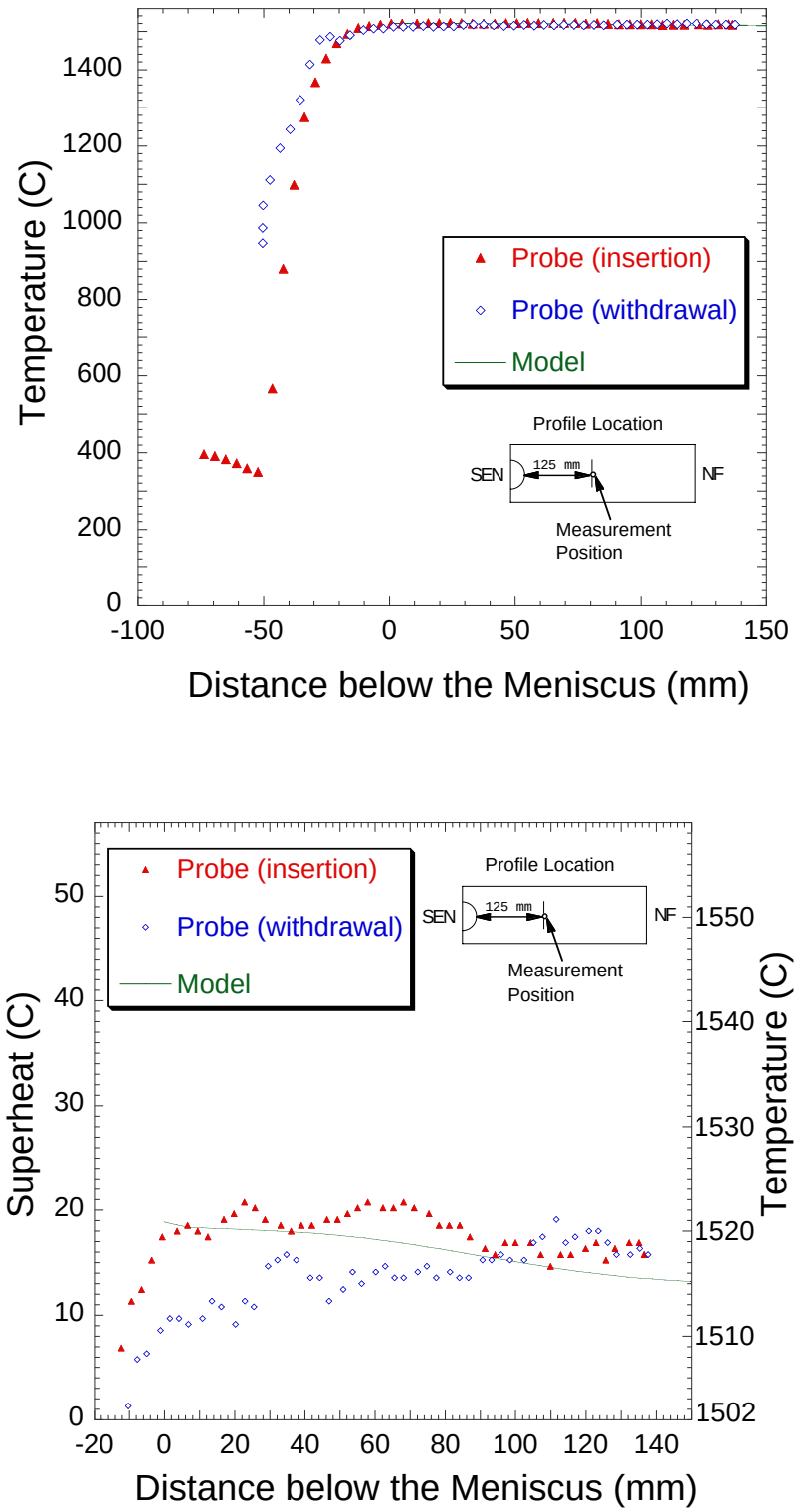
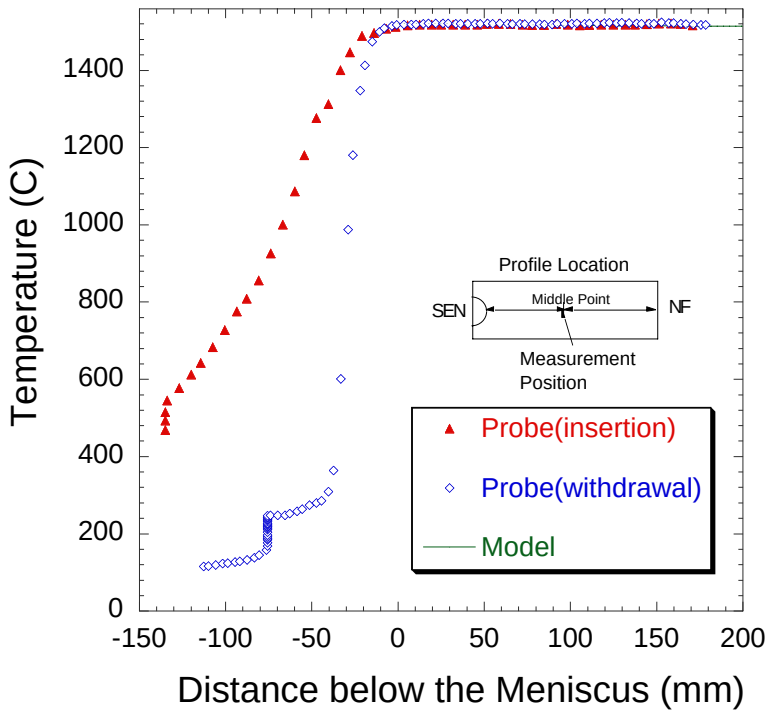


Figure 4.8 Measurement and modeling at 125 mm from SEN



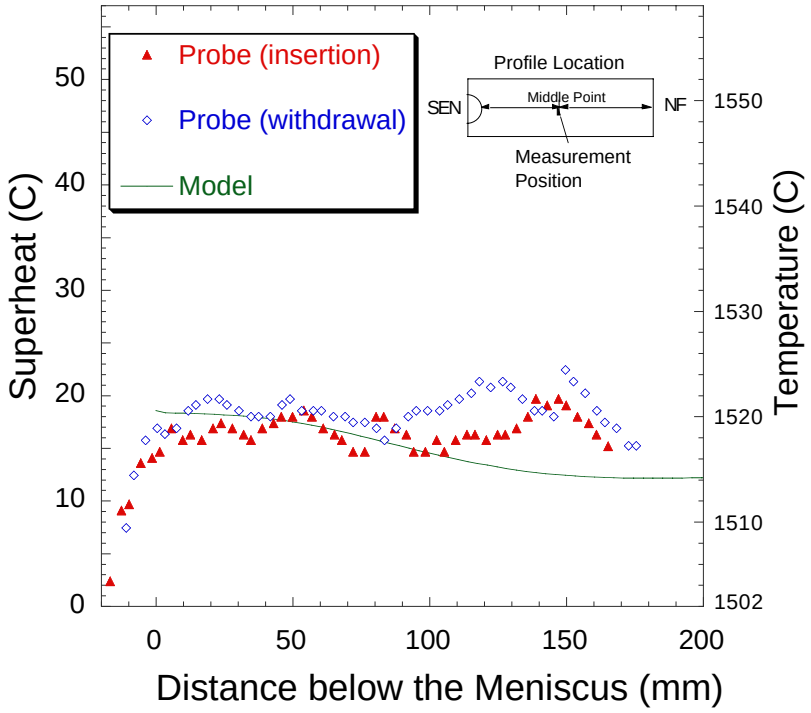
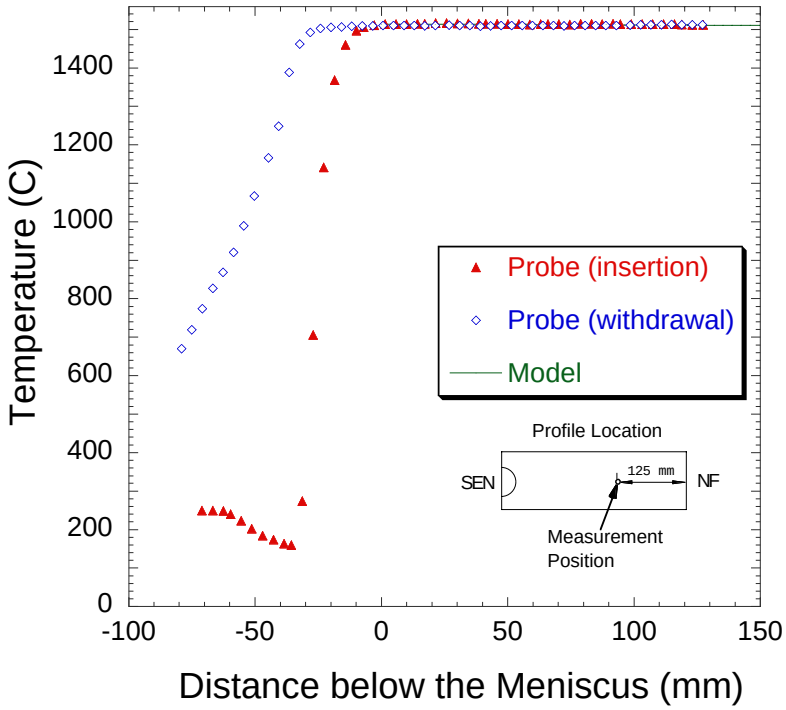


Figure 4.9 Measurement and modeling at midway between SEN and narrow face



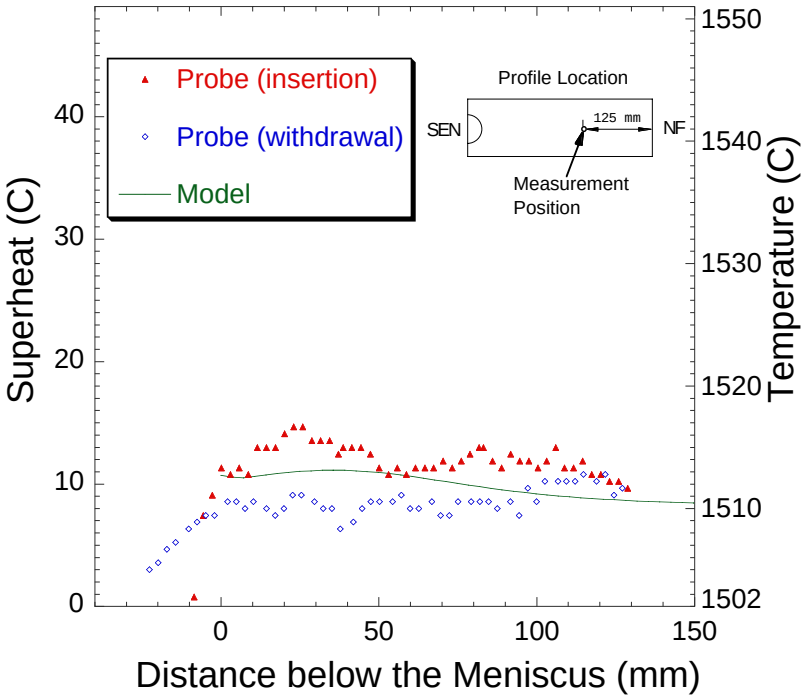
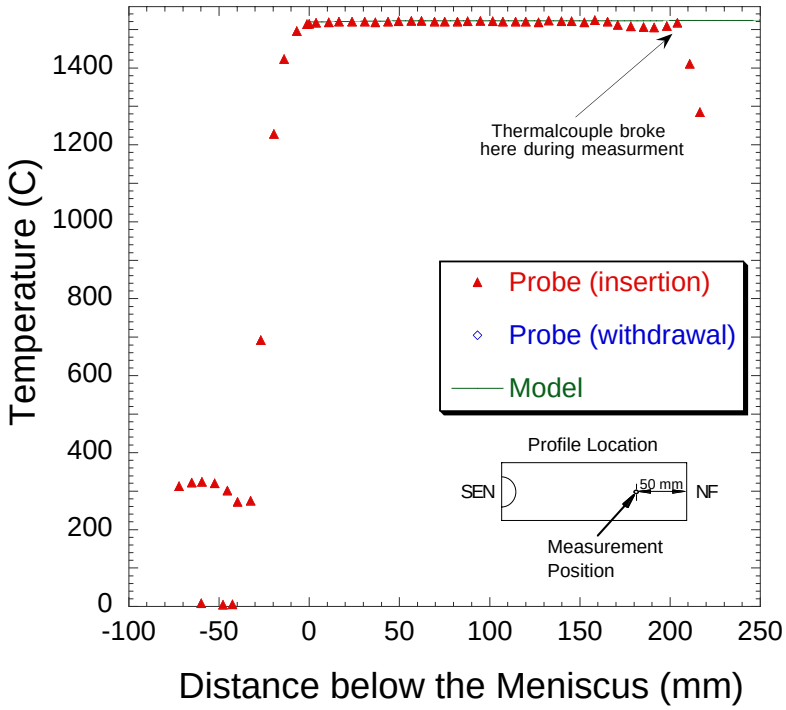


Figure 4.10 Measurement and modeling at 125 mm from narrow face



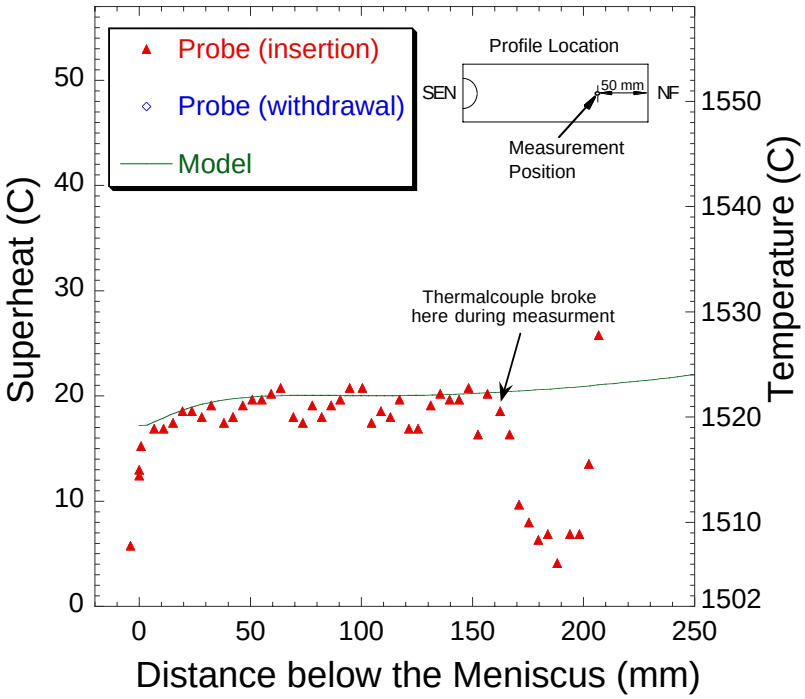


Figure 4.11 Measurement and modeling at 50 mm from narrow face

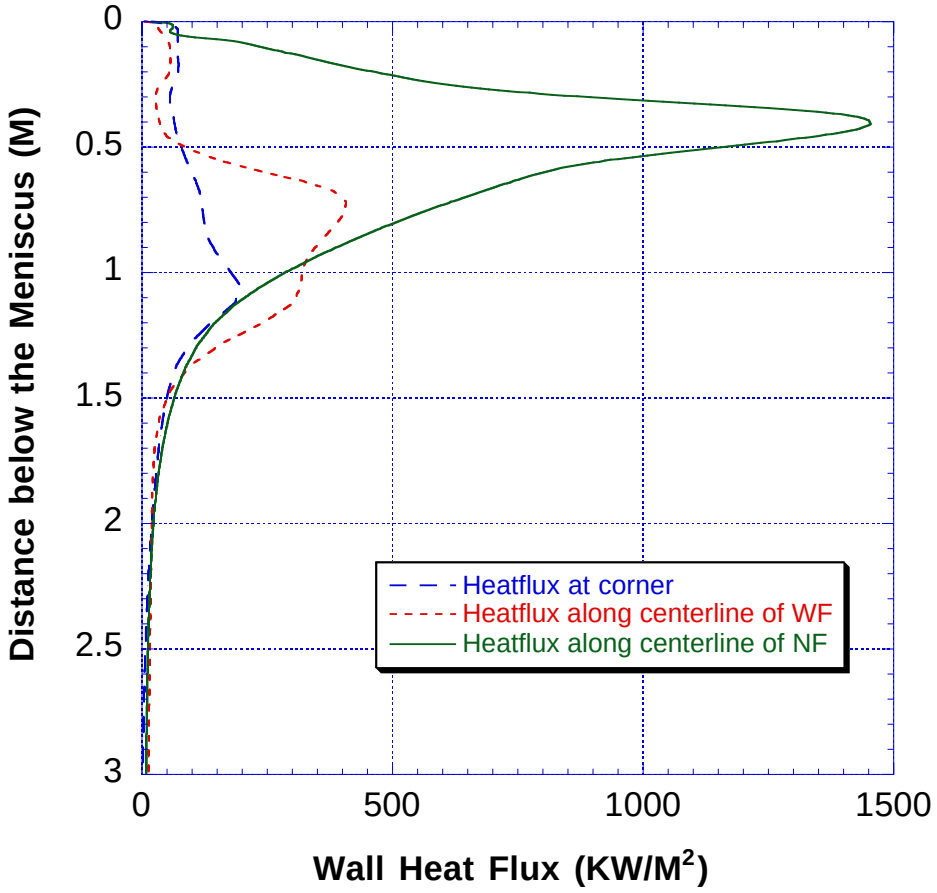


Figure 4.12 Heat flux along three typical lines

Section 5

Effect of Gas Bubble Size on Fluid Flow in Continuous Casting Mold

5.1 Introduction

Argon gas injection is a common practice in continuous casting. Argon gas is applied at several stages in the continuous casting process to encourage mixing, to help prevent nozzle clogging, and to promote the flotation of solid inclusion particles from liquid steel. However, gas bubbles change the flow pattern greatly [17, 22] and might cause some new problems.

Bubble size is one of the major factors which influence flow pattern. B.G.Thomas [17] et al investigated the effects of bubble size on flow pattern and heat transfer and gained important results. But only two different bubble sizes were present in this work. This limits the detailed understanding of effects of bubble size on flow pattern and other parameters. Another possible drawback of this work is that, to simplify the calculation, no momentum equation was solved for the gas phase. Instead, the bubbles were assumed to have a constant velocity in the vertical direction. This assumption might bring an inaccurate description of movement of bubbles and its influence on liquid flow because the distance might be too short to reach a terminal velocity.

In this study, a three-dimensional gas-liquid multiphase turbulent flow model is developed. Coupled liquid phase and gas phase equations are solved simultaneously so that movement of bubbles can be predicted more accurately.

With this model, the effects of bubble size on the flow pattern and other parameters are investigated. In this section, only one bubble size is set for gas phase. So

all gas bubbles are identical. To investigate the importance of difference in bubble size expected to exist in the liquid steel, seven cases are run, each with a constant bubble size ranging from 0.5mm to 4.0mm. This range covers most of the bubbles forming in the liquid steel according H. Bai's simulation [16] . For each case, the gas volume fraction is kept at 8.5%. To compare with single-phase flow, flow without gas is also modeled.

The flow pattern has a great influence on the formation of inclusion defects such as slivers and pencilpipes. Using these simulation results, the effects of bubble size on the formation of gas entrapment defects and inclusion particles are also investigated.

5.2 Model Description

In this study, a 3-D gas-liquid multiphase turbulent flow model is developed. Using CFX 4.2, the standard K-ε model is used to solve the turbulent liquid steel flow in the caster mold. Only large scale turbulent eddies could be simulated in this model. The gas phase is assumed to be laminar. An Eulerian multiphase multi-fluid model is used to simulate the motion of gas bubbles in the liquid steel. Each phase has its own velocity field. But only liquid phase has K and ε fields. The pressure field is shared by all phases. The velocity fields of the bubbles and liquid are coupled by an empirical inter-phase drag model. Details of this model are addressed in section 2.

The gas bubbles and particles concentrate mainly in the upper liquid pool in the caster. To save computing time, the length of the domain is set 3m from meniscus instead of the entire liquid pool. Again, considering the symmetry of the geometry, only a quarter of the strand is modeled.

At the nozzle port inlet boundary, the liquid and gas have the same velocity. The fraction of each phase is also specified. Inlet pressure values are extrapolated from downstream.

The meniscus is set as the non-slip wall boundary. Because almost all of the argon gas escapes from the meniscus (very little gas is entrapped in the slab), a gas sink is set at the top surface of the domain. A user subroutine is written to set up to absorb the escaping gas [subroutine degas in Appendix D].

The wide face and narrow face is set as non-slip wall boundary. According to the work of D. Creech [5], the solidified shell of wide face and narrow face can be neglected in the modeling without excessive error. The movement of the shell is also neglected. The velocity of the shell is set to zero.

The bottom of domain is set to a pressure boundary. Previous work showed that this condition can handle outlets where the flow is not fully developed better than the mass flow boundary [23]. Normal gradients for all variables are set to zero at the pressure boundary. In this study, the pressure is also specified to equal zero.

The two symmetry planes are set to symmetry boundary. The velocity normal to the plane is set to zero. For all other variables, including pressure, zero normal gradients are specified at symmetry boundary.

A schematic of the simulation domain with boundary conditions and a typical mesh are shown in Figure 5.1. The parameters used in the simulations are listed in Table 5.1. A single phase case with the same casting speed and the same geometry in Table 5.1 is computed first. All 2-phase flow simulations start from the results of the single-phase simulation as an initial guess. This method makes convergence of the 2-phase simulations

much easier and faster. The single-phase simulation converged in 1500-200 iterations, needing 5-8 hours on NCSA Origin 2000 supercomputer. The 2-phase simulations converged in 2000-3000 iterations, needing 20-30 hours on the same supercomputer.

5.3 Results and Analysis

5.3.1 Effect of Bubble Size on Flow Pattern

Gas injection affects the casting process through its influence on the liquid flow pattern. The extent of this effect depends on factors such as the gas injection rate, the bubble size, jet velocity etc. In this study, all variables are fixed except bubble size. Mean bubble diameter changes from 0.5mm to 4.0mm, which covers most of the bubble size range met in practice. The effect of bubble size on the flow pattern is shown in Figure 5.2. Small bubbles (d<1.5mm) have greater influence on the liquid flow than big bubbles (d>1.5mm). They tend to stay in the main jet of the flow longer and bend all the flow upward to impinge the top surface, thus tending to form a single roll flow. Big bubbles tend to float up faster and split the jet. In this case, the jet divides into two branches of flow. One branch floats up quickly when the jet enters the domain and impinges the top surface. Another flow, which has little gas left with it, moves on and impinges the narrow face. A generally double roll flow pattern is formed. For the non-gas case, jet goes straight to the narrow face and forms a classic double roll flow pattern after impinging the narrow face.

5.3.2 Effect of Bubble Size on Gas Entrapment

Most gas bubbles float upward to escape from the top surface due to their strong buoyancy force. Bubbles entering lower circulation zone have a larger chance to be

trapped. Although no more than 5% of the gas enter the lower circulation zone of the liquid flow [17], it is still dangerous because the volume fraction for serious gas defects is as low as 10^{-4}% [24]. Thus, it is important to know how much gas is left in lower region of the mold and how large the region where the gas fraction is over 10^{-4}% is.

A parameter "penetration depth" is used to tell how deep the dangerous gas fraction level can extend downward from meniscus. Penetration depth is defined as the deepest distance from the meniscus where the 10^{-5}% gas volume fraction contour reaches. It is assumed that the deeper the penetration depth, the more difficult it is for gas to float up to meniscus and the more possible it is for gas to be entrapped. This parameter gives a clue of the chance that gas might be trapped.

Figure 5.3 shows the penetration depth of different bubbles sizes. For small bubbles (0.5mm ~ 1.0mm in diameter), the penetration depth increases with bubble size. In this size range, bubbles stay with liquid steel flow. The smaller the bubble, the stronger tendency it has to stay with the flow. Most of the flow will meet the top surface and let gas escape. With increasing bubble size, the drag forces by the bubbles decrease. Some part of the flow will not bend upward to meet the top surface. This part will go down before it meets the top surface. Gas in this part of flow will also go down the domain and has a deeper penetration. The 1.0mm bubble has the deepest penetration. Large bubbles (1.5mm~4.0mm in diameter) have a very small penetration depth. Big bubbles have a smaller total surface area and thus smaller drag force to liquid steel flow. Most bubbles leave the jet quickly after entering the domain. Very few large bubbles go down the domain with flow. The larger the bubble, the less gas leaves in the flow, and the less the penetration depth.

For large bubbles, the gas percentage near the narrow face is much smaller than 10^{-4}% (see Figure 5.4). This indicates that there is very small chance for large bubbles (1.5mm~4.0mm in diameter) to be trapped and form pencil-pipe defects on the narrow face.

Another parameter that should be concerned is the area which is enclosed by the 10^{-5}% gas fraction contour on the solidifying shell. It is reasonable to assume that the larger this area is, the higher the possibility of gas entrapment. This is because entrapment can only occur when the gas bubble moves very slowly relative to the solidifying interface. Figure 5.5 shows the area enclosed by the 10^{-5}% gas fraction contours for different bubbles. Small bubbles (0.5mm ~ 1.0mm) have much more area enclosed by the 10^{-5}% gas fraction contour than large bubbles (1.5mm~4.0mm). The 1.0mm bubbles have largest area. This indicates that small bubbles (especially around 1.0mm) are more prone to gas entrapment. Large bubbles (2.0mm~4.0mm) have very small potential entrapment areas so might not have gas entrapment.

It is assumed that gas entrapment happens when the speed of gas flowing down the shell approaches the casting speed. It is much more possible for gas to be entrapped when the gas volume fraction is over 10^{-5}% and the gas downward velocity is close to casting speed simultaneously. Figure 5.6 shows the overlap of 10^{-5}% gas fraction contour and gas vertical velocity contour for four different bubbles. The red-shaded area is the overlap region. These regions might be the position where gas entrapment happens.

5.3.3 Effect of Bubble Size on Inclusion Formation

There are two major sources of inclusion particles: inclusion particles entering with the inlet flow and inclusion particles sheared off from flux layer and entrapped in

the flow. The first kind of inclusion particles comes from steel making and erosion of refractory materials. Most of these particles circulate in the liquid pool and become entrained harmlessly into the flux layer. Some of them are entrapped in the growing shell and initiate quality problems such as surface slivers. According to experiments by R.C.Sussman etc.[4] , 22.3% of particles will not float up and finally remain in the solidified shell when gas is not injected into inlet flow. With such a high percentage of entrapment of particles, inclusion defects should be very common in continuous casting. Fortunately, this is not the fact. Serious inclusion defects happen only occasionally. This indicates that steel inlet flow is rather clean and particles in inlet flow might not be the major cause of inclusion problem. Flux particles sheared away and entrapped by liquid steel flow might be the major source of inclusion particles. There are two possible mechanisms by which flux is sheared away and entrained in the steel. The first mechanism is that flux could be sheared away by liquid level fluctuations of the interface between the liquid steel and the flux layer (Figure 5.7). The second is that flux also could be sheared by downward steel flow as shown in Figure 5.8. This suggests that both the flow pattern (single vs double roll) and the velocity magnitude are important.

5.3.3.1. Effect of Bubble Size on Level Fluctuations

In this study, turbulence is assumed to be isotropic. The average velocity fluctuation can be obtained from the Kinetic Energy K [9]:

$$\overline{u'} = \sqrt{\frac{2}{3}K}$$

The level fluctuation height is reported to related to K also [17]:

$$h = \frac{\rho_{steel} * K}{0.5(\rho_{steel} - \rho_{flux})g}$$

Figure 5.9 shows the maximum kinetic energy and maximum level fluctuations on the top surface for different bubble sizes. Generally, small bubbles have higher level fluctuation than larger bubbles. Flow without gas has the smallest level fluctuation on the top surface. This simply shows that the stronger surface flow caused by the small bubbles leads to more level fluctuations and consequently more surface defects.

5.3.3.2 Effect of Bubble Size on Downward Velocity near Solidifying Shell

During operation of the continuous caster, the top liquid steel level is always moving up and down with minor fluctuations of the inlet flow rate and casting speed. This might sometimes lower the liquid steel level below the top of solidified shell sometimes. The top edge of the solidified shell then inserts into flux layer like a blade. When the flux moves to the gap, the lower part of the flux layer is peeled off and stays at the solidified shell. If liquid steel flow is upward, the peeled flux would be put back and has less chance to form inclusion particles. Otherwise, the peeled flux would be pushed down into the liquid pool and form large inclusion particles. Because these particles form just near the growing shell, they would be easily entrapped by the shell.

The maximum downward velocity can give clues on the tendency to entrap flux particles by this mechanism. Figure 5.10 shows the effect of the bubble size on vertical velocity around the top surface perimeter of the mold. The maximum downward vertical velocity appears on the wide face in all eight cases. This indicates that the wide face would likely suffer more inclusion defects than the narrow face. The inclusion particles concentrate more near the SEN on the wide face with a shallow depth. The maximum

downward vertical velocity increases with increasing bubble size and reaches a maximum at 1.5 mm bubble. The maximum downward vertical velocity decreases with further increasing the bubble size. Bubbles between 1.0mm and 2.0mm have the largest maximum downward vertical velocity so might have the largest tendency to suffer inclusion defects caused by the second mechanism. Combined with the first entrapment mechanism, it is believed that smaller bubbles (d<2.0mm) have more tendency to entrap flux into liquid flow.

What should be noticed is that flow without gas has the smallest vertical velocity around top surface perimeter of the mold. Considering the flow without gas also has the smallest level fluctuations, it might have the smallest tendency to suffer inclusion entrapment. Flow without gas should have less inclusion defects than flow with gas if the liquid steel is clean. If this is the case, we will find more flux inclusions in gas injection practice than in no gas practice. This suggests that gas injection is a dangerous practice that must be carefully optimized.

5.4 Conclusions

1. The jet without gas goes across the mold and impinges on the narrow face forming a classic double roll flow pattern. Jets with small gas bubbles (<1.5mm) bend upward and impinge first on the top surface to form a classic single roll flow pattern. Jets with large gas bubbles (>1.5mm) split into two branches. One, which is mostly gas, goes up and impinges on the top surface. The other, which is mostly liquid steel, impinges first on the narrow face. The flow with large bubbles (>1.5mm) shows a generally double roll flow pattern.

2. Medium size bubbles (around 1.0mm) appear to have the largest tendency to be entrapped deep below the meniscus as gas defects such as pencil pipes because they encourage a single roll flow pattern which imparts the largest downward velocity to the liquid jet and they are easily carried by the jet deep into the caster. With decreasing bubble size below 1 mm, the tendency of forming gas defects also decreases. Large bubbles (>1.5mm) have the least tendency to form gas defects because they float out the liquid quickly.
3. Smaller bubbles induce larger level fluctuations on the top surface and increase the possibility of entrapping flux particles into the liquid flow by liquid level fluctuation.
4. Downward velocity along the solidified shell at the meniscus reaches its maximum value at 1.5mm diameter. Bubbles with size between 1.0mm and 2.0mm have the largest possibility to cause flux to be trapped in the growing shell at meniscus due to downward movement of liquid steel and flux. The most possible position for inclusions happening by this mechanism is near the SEN at wide faces.
5. Compared with the flow with gas, the flow without gas has much less tendency to entrap flux particles and might have fewer inclusion problems in cases when the liquid steel is clean.
6. The study in this section is based on the assumption of uniform bubble size, which is not the case. The next section will explore the importance of bubble size distribution.

Table 5.1. Parameters used in the simulations

	Case B(6.3L/min,35″/min)
Mold Width	1854 *mm*
Mold Thickness	228 *mm*
Nozzle Submergence Depth (top surface to top of port)	120 *mm*
Nozzle Bore Inner Diameter	78 *mm*
Port Wall Thickness	27.5 *mm*
Nozzle Port Height	78 *mm*
Nozzle Port Width	78*mm*
Inlet Jet Height, L_h	50 *mm*
Inlet Jet Width, L_w	78 *mm*
Nominal Vertical Angle of Port Edges	15° down
Inlet Jet Spread Angle	0°
Casting Speed, V_c	14.8 *mm/s*
Inlet Velocity, V_x	0.8766 *m/s*
Inlet Velocity, V_z	0.3018 *m/s*
Inlet Turbulent Kinetic Energy, K_o	0.0502 m^2/s^2
Inlet Turbulence Dissipation Rate, ε_o	0.457 m^2/s^3
Liquid Steel Density, ρ	7020 kg/m^3
Steel Laminar (Molecular) Viscosity, μ_o	0.00560 *kg/m s*
Inlet Gas Flow Rate(whole slab)	6.3 *l/min*
Inlet Gas Volume Fraction, f_{gas}	8.5% or 0%(No gas)
Average Gas Bubble Diameter, D_o	0.5, 0.7,1.0,1.5, 2.0, 3.0, 4.0 *mm*
Gravitational Acceleration, g	9.8 m/s^2

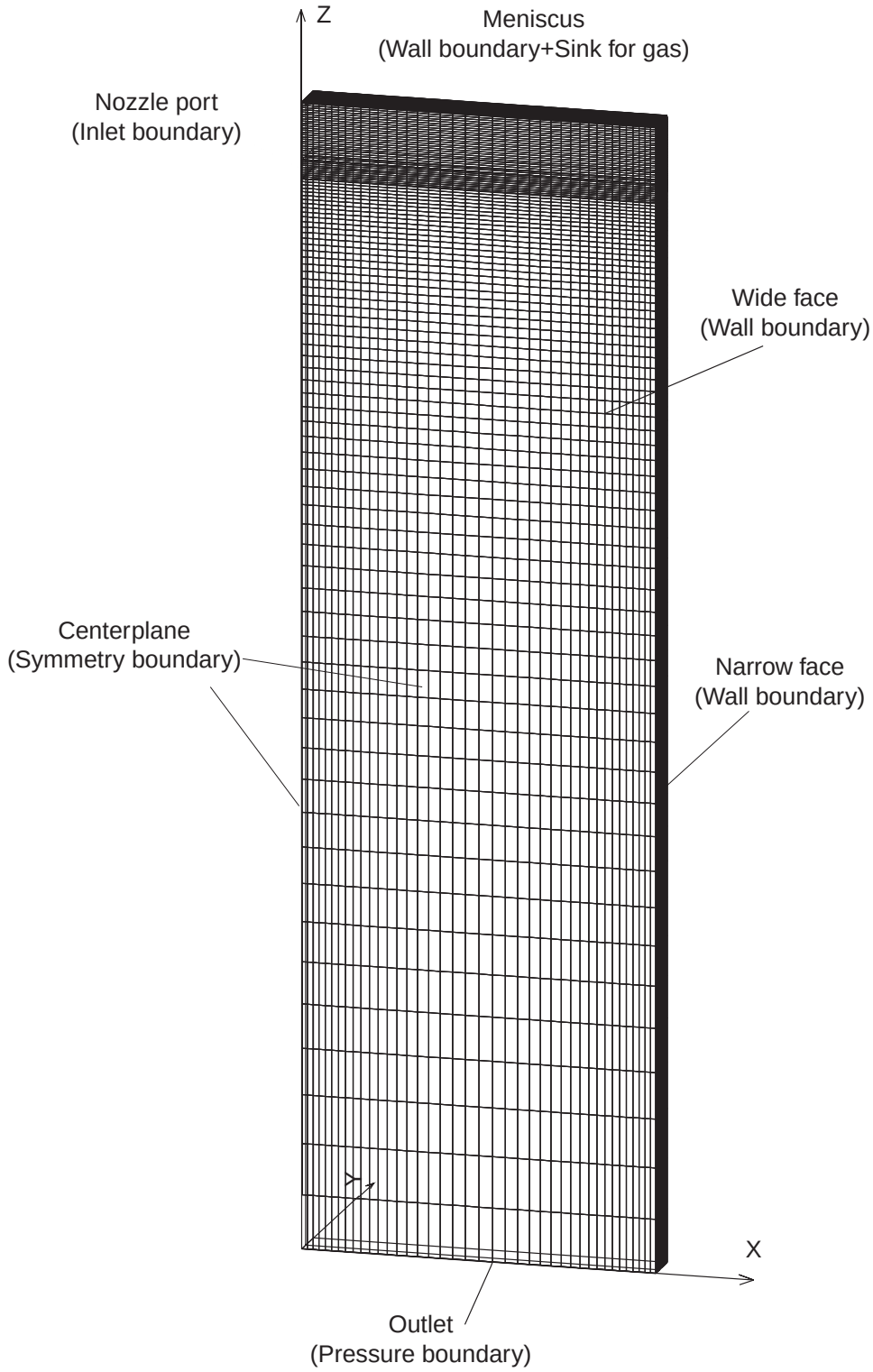


Figure 5.1 Schematic of simulation domain with boundary conditions and typical meshes

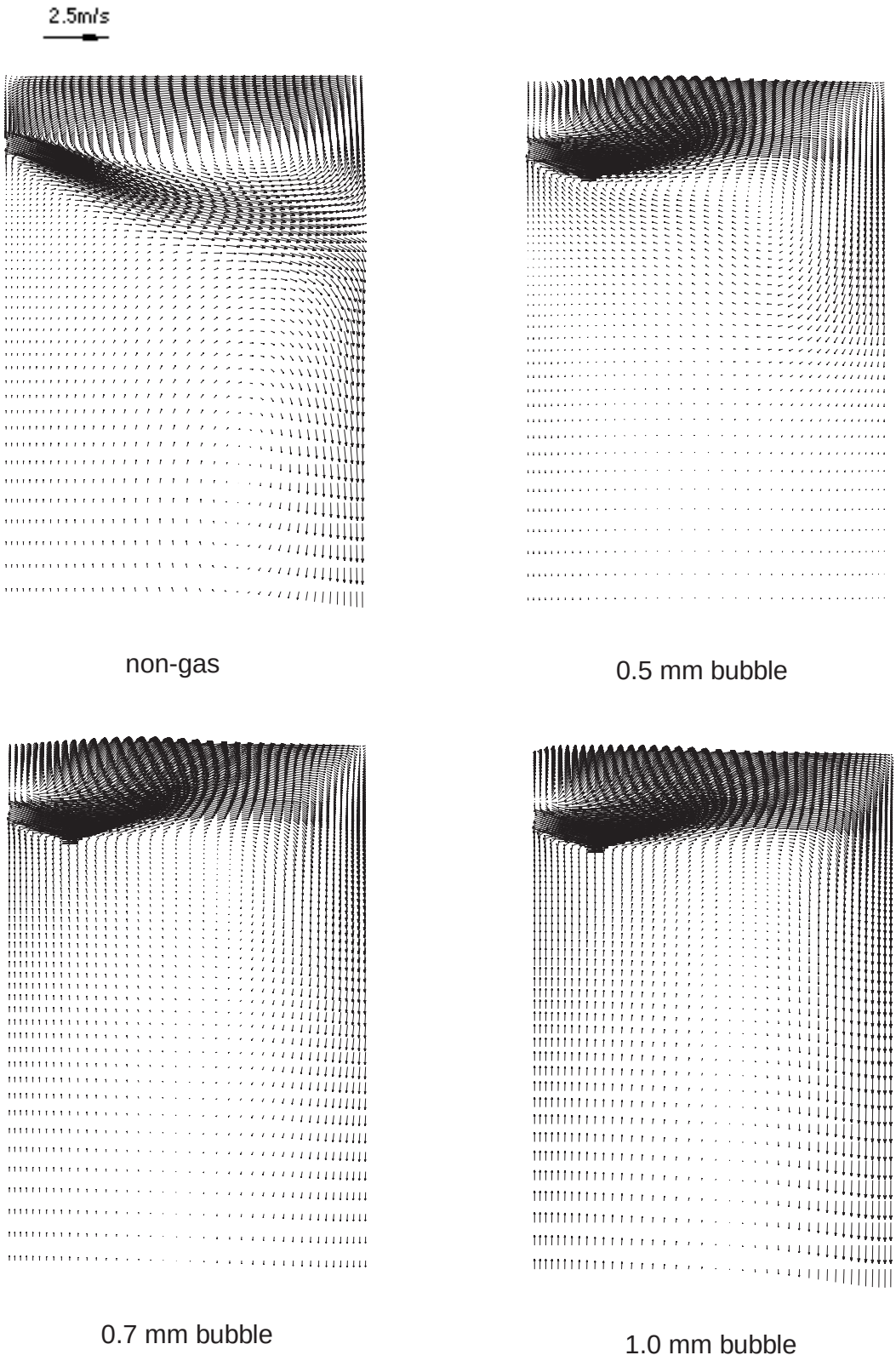


Figure 5.2 Effects of bubble size on fluid flow pattern
in continuous casting mold

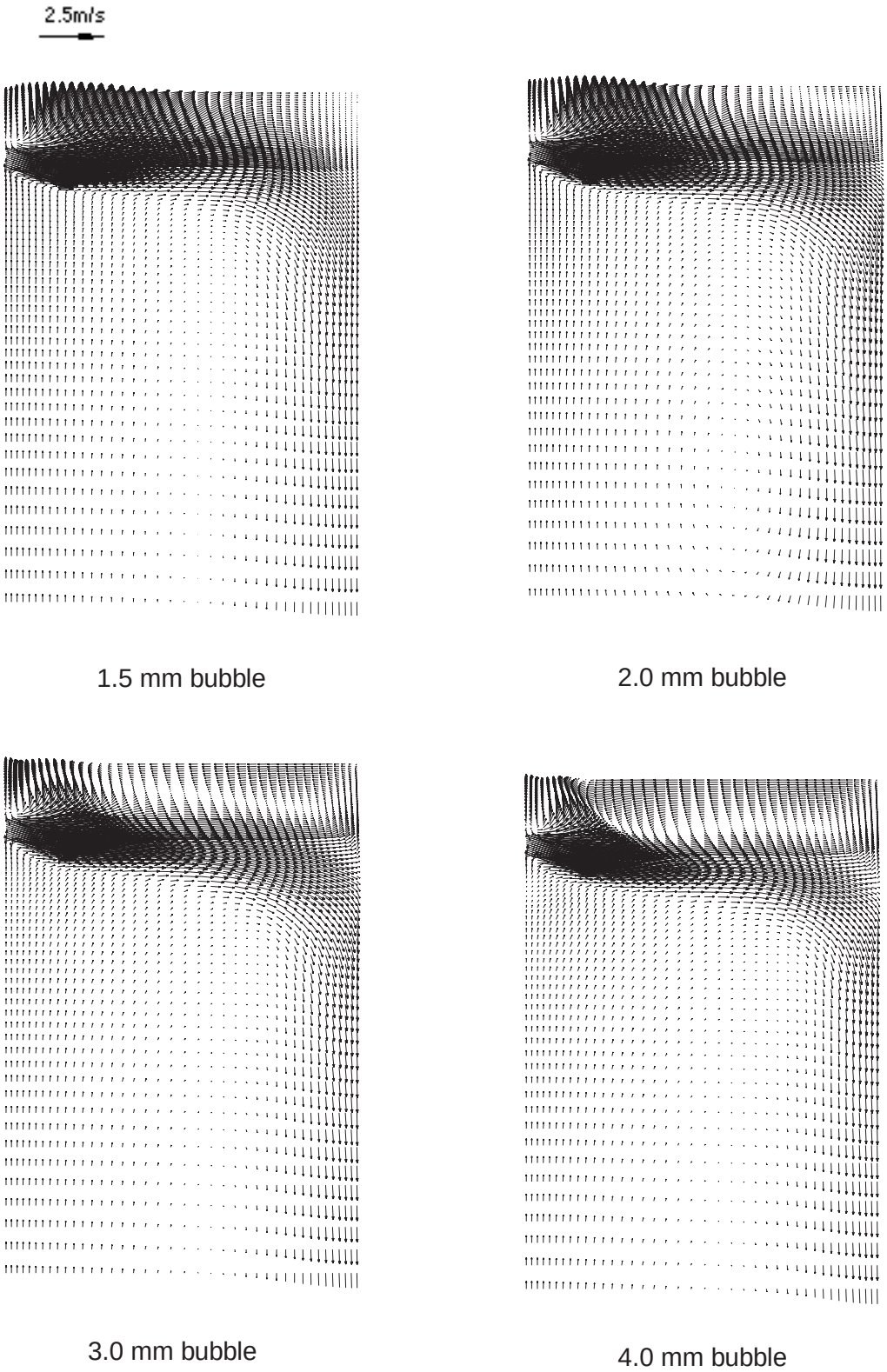


Figure 5.2 Effects of bubble size on fluid flow pattern
in continuous casting mold(continued)

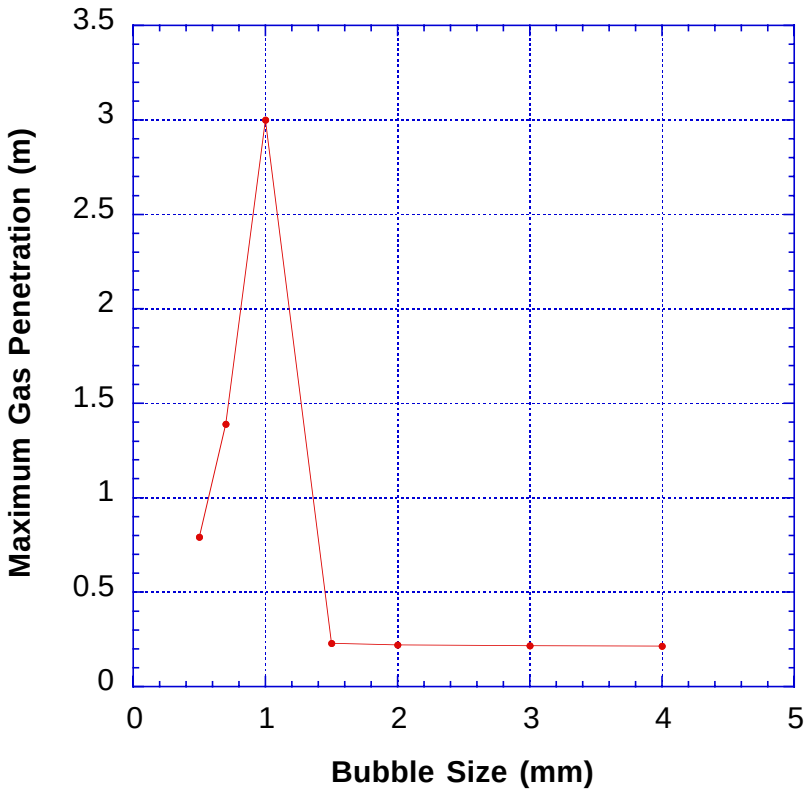


Figure 5.3 Effect of bubble size on gas penetration depth

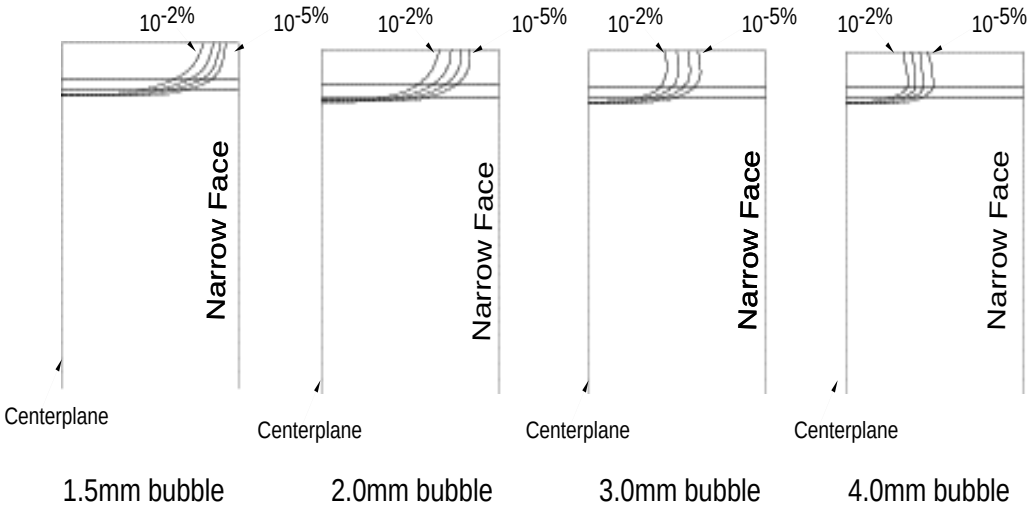


Figure 5.4 Gas fraction contour for different sized bubbles

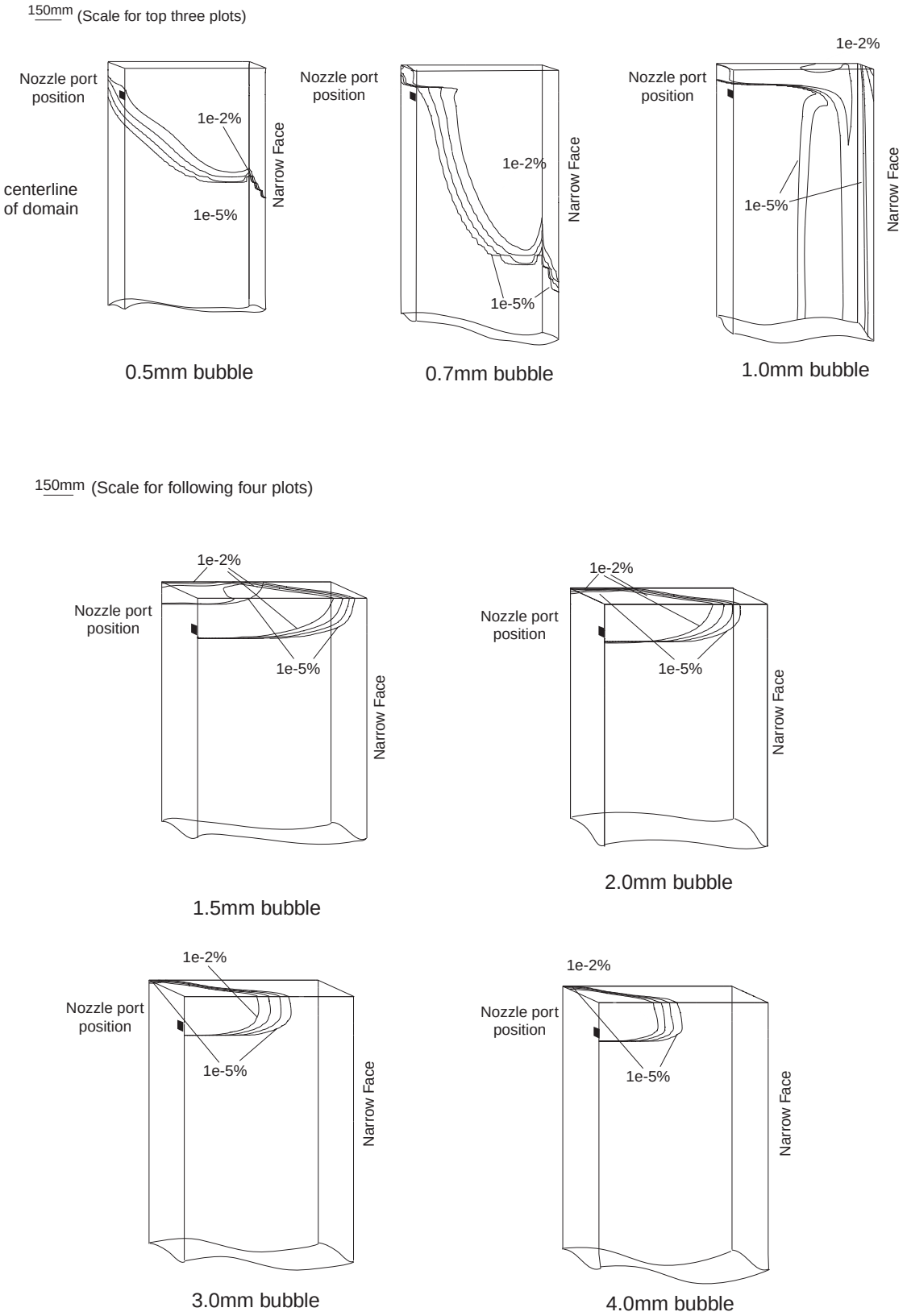
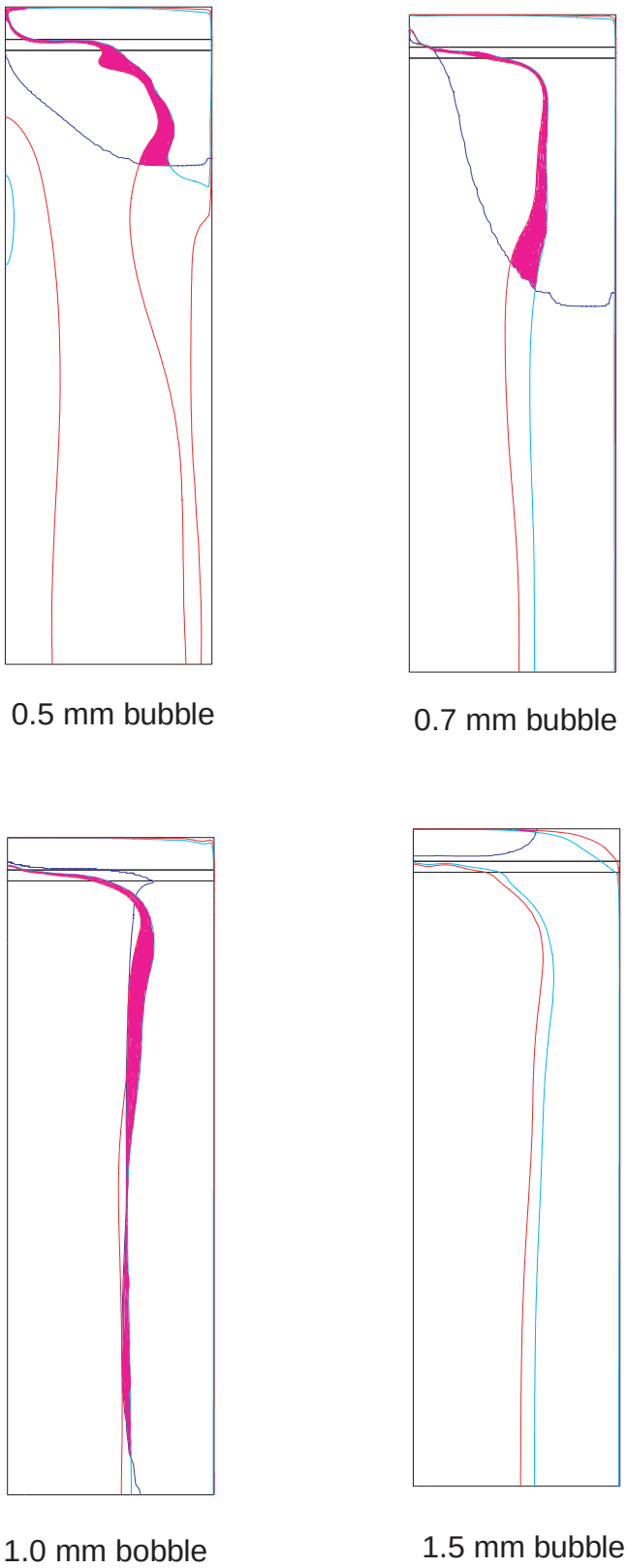


Figure 5.5 Gas percentage contours for different bubbles



Explanations:

red line: W=-24mm/s contour

cyan line: W=-8mm/s contour

casting Speed = -14.8 mm/s

blue line:
1e-05% gas fraction contour

pink area:
The region where inclusions might occur

* All 4 pots are the planes 3mm offset wide face wall

*The two horizontal line in the planes show the position of upper and lower boundary of the nozzle port

Figure 5.6 Overlap of the contours of gas fraction and vertical velocity

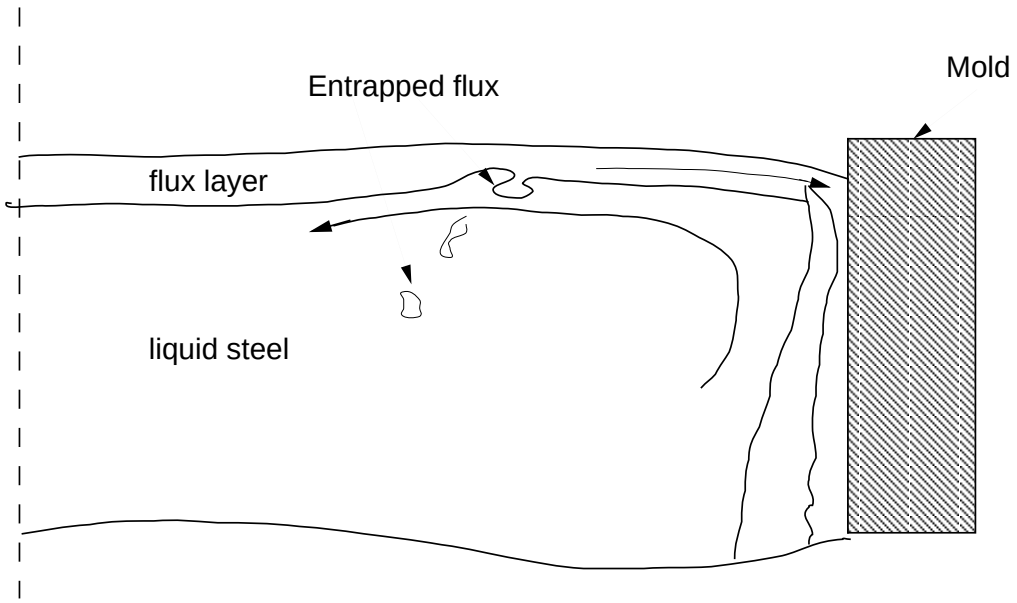


Figure 5.7 Schematic of the first mechanism of flux entrapment

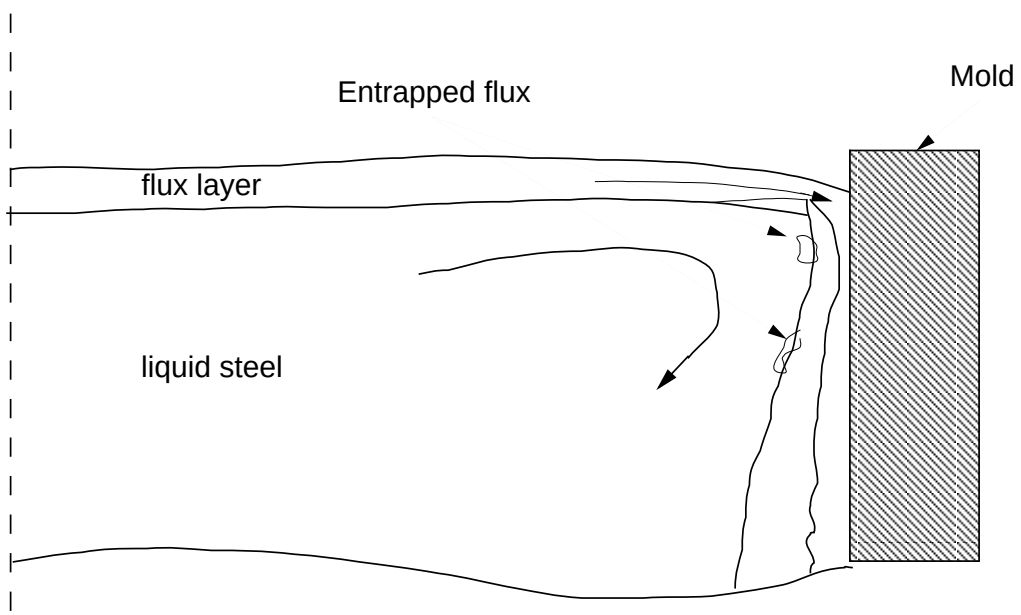


Figure 5.8 Schematic of the second mechanism of flux entrapment

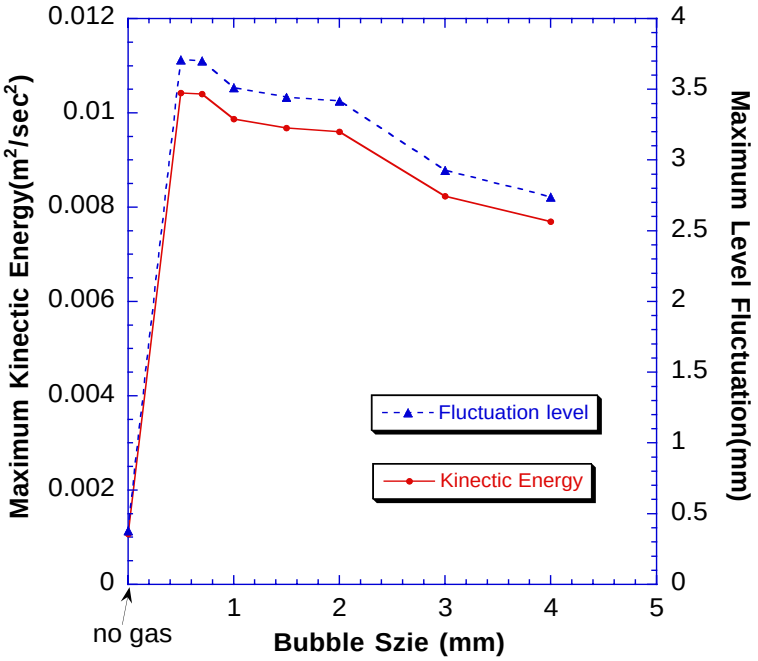


Figure 5.9 Effect of bubble size on the maximum kinetic energy and level fluctuation on the top surface

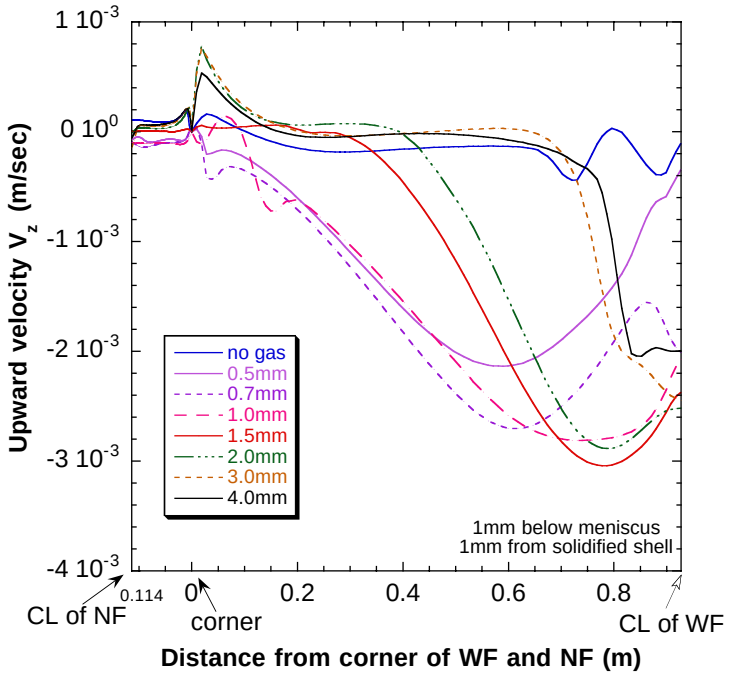


Figure 5.10 Effect of bubble size on vertical velocity around top surface perimeter

Section 6

Effect of Argon Flow Rate on Flow Pattern in Continuous Slab Casting

6.1 Introduction

The influence of argon on flow in continuous slab casting varies with factors such as the gas volume fraction in liquid steel flow, gas distribution in the jet, bubble size and jet direction etc. Increasing gas volume fraction gives more lift force to the liquid jet and tends to cause single roll. A jet with most gas concentrating in upper part of the jet tends to cause double roll due to less interaction with jet. Larger bubbles leave jet more quickly and tends to cause double roll. Those factors are influenced by liquid flow rate, argon flow rate, argon injection method and nozzle geometry. The mold flow is also influenced by submergence depth, mold geometry and nozzle orientation.

Flow in the nozzle and mold is difficult to measure due to the high temperature. Computational simulation is therefore an efficient way to know the flow behavior in the nozzle and mold. In this section, typical argon flow rates, casting speeds and slab widths used in practice are investigated. Details of the nozzle are given in Table 6.1. The computational simulation takes two steps. In the first step, flow in the nozzle is simulated. In the second step, flow in the mold is simulated using inlet conditions from the results recorded at the nozzle port in the first step. The inlet mesh is chosen exactly the same as the nozzle port in the first step. This integrates the nozzle and mold computational simulation into a complete computational simulation set. The validity of this procedure was demonstrated by Hershey [28].

In Section 5, it was shown that bubble size has a great influence on the flow pattern. Bubble size and its distribution in the nozzle varies greatly due to different bubble formation mechanisms which depend on liquid flow velocity and gas injection methods [16]. Gas bubbles will also breakup and coalesce after they enter the nozzle and mold. Treating gas bubbles as uniform-sized is too rough an approximation and is not able to capture the influence of bubbles on the liquid flow in detail.

In this section, the MUSIG model (described in Section 2) is used to model breakup and coalescence of bubbles. The initial bubble size and its distribution in the nozzle are measured from a two-needle water model experiment. Because the bubble size under the same casting conditions differs in water-air system and steel-argon system, Bai's model [16] is used to extrapolate the results from water-air system to the steel-argon system. For the 0.4-scale water model, the bubble size is measured from still photographs. The objective of this study is to validate the multiphase model with MUSIG model by the comparison of computed results with the water model measurements. The model is then used to identify the flow pattern in the real caster for different casting conditions.

6.2 Definition of the Mean Bubble Size in This Study

In this section, the average bubble diameter is calculated by the Average Volume Method. First, the total volume of all bubbles in the sample pool is calculated. Then, the average volume of a single bubble is calculated by dividing the total volume by the total number of bubbles. Finally, the average diameter is calculated by assuming spherical bubbles having the same volume as the average volume. The equation to calculate the

average bubble diameter is:

$$D_{avg} = 2 * \sqrt[3]{3 * (\sum_{i=1}^{n} (Ni * V_i) / \sum_{i=1}^{n} N_i) / 4 / \pi}$$

where V_I is the volume of a single bubble at size i. N_I is the number of bubbles at size I in a sample pool.

The volume fraction of bubbles at size i is straightforward:

$$VF_i = N_i * V_i / \sum_{i=1}^{n} N_i * V_i$$

where VF_I is the volume fraction of bubbles at size i.

6.3 Bubble Size and Its Distribution in the 0.4-scale Water Model

In this study, instantaneous photographs are used to measure the bubble size and its distribution in the water model for the conditions in Table 6.4 and 6.5. Two such photographs are shown in Figs. 6.1 and 6.2. The air injection method of the 0.4-scale water model at LTV steel is different from the steel casting nozzle. Instead of a porous ceramic filter, the 0.4-scale water model uses a slot ring to inject air. This method produces very large bubbles at low casting speed (> 10 mm). The results are shown in Table 6.2 and 6.3. Figure 6.3 shows the bubble size and its distribution in the water model. With slot ring injection, bubble size is relatively uniform at high casting speed (23.2 mm/s). Most bubbles have the size between 2.5 mm and 3.5 mm. Average bubble size is 2.59 mm. The bubble size is widely distributed at low casting speed (14.8 mm/s). Although few in numbers, huge bubbles exist and account for a large fraction of total gas volume. In this case, two huge bubbles (9.5 mm and 10.5mm) account for about 47% of total gas volume. Most bubbles are smaller than 5.0 mm. The number of bubbles smaller

than 1.0 mm is larger than that at high casting speed (case A). This causes the average bubble size (2.43 mm) to be smaller than that at high casting speed (2.59 mm).

6.4 Validation of MUSIG Model by 0.4-scale Water Model

The MUSIG model was validated by comparison with experiments in the 0.4-scale water model. Two water model cases (case A and B) were performed. These two cases are scaled from two typical plant cases at LTV Steel. Case A corresponds to 1.854 m slab with casting speed of 23.2 mm/s and 13SLPM argon flow rate. Case B corresponds to 1.854 m slab with casting speed of 14.8 mm/s and 6.3 SLPM argon flow rate. The conditions for these two water model cases are listed in Table 6.4 and 6.5. In the practice, case A tends to have more pencil pipe defects and case B tends to have more sliver defects. The computational simulation domain for the 0.4-scale water model is the same as that in section 3 (Figure 3.1). The bubble size and its distribution were obtained by still photographs as discussed in section 6.3. Figure 6.4 compares the still photograph, the result of computational simulation and that of PIV measurement on the case A. On the left is a still photograph of the flow for case A. In the middle is the computational simulation result of velocity on the center plane of the 0.4-scale water model. On the right is the PIV measurement of velocity on the center plane of the 0.4-scale water model. Figure 6.6 shows a similar comparison but for case B. From these comparisons, it is obvious that the computational simulation matches the observations and measurements qualitatively for both cases. Both case A and case B have a single roll flow under normal casting conditions.

An interesting phenomenon was observed in the experiments. The flow rate control was not so smooth so that flow rate changes sometimes occurred. When the water level is low, the flow rate was increased by up to 15% to restore the standard water level. In the steel continuous casting, the same phenomenon might exist when ladle is changed (As a practice in LTV steel, casting speed is decreased from 23.2 mm/s to 14.8 mm/s and then increased back to 23.2 mm/s with no gas flow rate changes when ladle is changed). The 15% increase of flow rate in the 0.4-scale water model changes the flow pattern. Figure 6.5 and Figure 6.7 show still photographs of the flow during the 15% increase in flow rate for the conditions the same as case A and B respectively. For case A, the flow changes to a slight double roll from the single roll. However, for case B, the flow is always single roll, even for the 15% increase in the flow rate. The computational simulation matches the measurements in both cases. Although the observations, measurements and computational simulations in the 0.4-scale water model match each other, they don't match the plant observations and measurements in steel continuous casting [26]. Further discussion is given in section 6.8.5.

6.5 Bubble Size and Its Distribution in the Steel Casting Nozzle

In the steel casting nozzle, the initial bubble size depends mainly on the vertical liquid velocity, the gas flow rates per orifice and the contact angle [16]. The steel throughput, nozzle bore diameter, argon flow rate and ceramic filter structure are the main factors that determine the bubble size and its distribution. Under different casting conditions, the bubble size and its distribution in the nozzle are different. The nozzle geometry parameters and casting conditions for both case A and B are listed in Table 6.1.

The average liquid velocity is calculated from liquid flow rate:

$$V = \frac{Q_{liq}}{\pi R^2}$$

where R is the radius of the nozzle. The liquid flow rate Q_{liq} is calculated according to:

$$Q_{liq} = V_c * W * T$$

where V_c is casting speed, W is mold width, T is mold thickness.

The vertical liquid velocity shearing gas at pores as it across the ceramic pores in the nozzle is 2.05 m/s and 1.31 m/s for case A and B respectively. The number of active pores in the ceramic filter is estimated based on Zhang's experiments [25] which suggests 60 mm^2 of surface area per active site. For the nozzle in case A and B, the area of the porous ceramic filter is 50 mm high which gives total inner surface area of ceramic filter of 12246 mm^2. The total active pores in the ceramic filter is then 204. The gas flow rate per pore is about 2.9 ml/s and 1.8 ml/s .

The bubble size for the above parameters is based on measurements in the two-needle water model experiments by Hua Bai [16]. The water model is composed of a 35×35 cm square section nozzle and two needles (Figure 6.8). The nozzle wall is made of transparent plastic for easy observation. Needles with 0.2, 0.3 and 0.4 mm in diameter were used. Water flow through the nozzle and air is injected into the nozzle through two needles. Adjusting the water flow velocity and air injection rate obtains different bubble sizes. The bubble size is measured from high speed video camera frames. Bai's experiments don't have the exact case A and B. This study chooses the closest cases to case A and B. Table 6.6 and 6.7 show the results. For the high liquid flow rate case (case A), bubbles distribute over a relatively wide size range. Small bubbles are much more numerous than larger bubbles. But large bubbles (>3.0 mm) take more than 76% volume.

The mean bubble size for case A is 1.94 mm. For the low liquid flow rate case (case B), bubbles concentrate around the mean bubble size. Bubbles show a relatively uniform size. The mean bubble size for case B is 2.12 mm. A statistical analysis of the bubble size distribution for the two cases is shown in Figure 6.9. The above bubble sizes and their distribution are used in this study.

The bubble in the real caster might be slightly larger for two reasons. Firstly, the mean bubble size in the steel argon system is larger than the corresponding bubble size in the water-air system according to Bai's bubble formation model (Figure 6.10) [16]. For case A (vertical liquid velocity = 1.9 m/s and gas flow rate per pore = 2.9 ml/s), the steel-argon system has a mean bubble size of 2.7 mm while water-air system has a mean bubble size of 2.6 mm. For case B (vertical liquid velocity = 1.4 m/s and gas flow rate per pore = 1.8 ml/s), the steel-argon system has a mean bubble size of 2.75 mm while water-air system has a mean bubble size of 2.38 mm. Secondly, Bai measured a larger mean bubble diameter than in the nozzle computational simulation in this study.

6.6 Computational simulation of Flow in the Nozzle

Before every mold computational simulation, a nozzle computational simulation is run for the particular throughput and gas injection rate. Figure 6.11 shows the geometry of the nozzle. Liquid flows in from the top surface where the nozzle connects to the tundish. Argon is injected into nozzle through porous ceramic in the gas injection zone. The top surface and gas injection zone are set as inlet boundaries. The nozzle wall is set as non-slip wall boundary. The nozzle port is set as pressure boundary. Linear pressure is applied at the nozzle port based on submergence depth. The geometry and

other parameters for nozzle computational simulation are listed in Table 6.7. About 2 CPU hours on NCSA SGI supercomputer are needed for 500 iterations which achieve a fully convergence.

Some typical results for case A and B are shown in Figure 6.12 and 6.13. There is a swirl on the center plane parallel to SEN port in case A while flow on that plane in case B is symmetric. The inflow area is larger in case B than that in case A. The spread angle of the jet in case A is also larger than that in case B.

6.7 Computational simulation of Flow in the Continuous Caster

The inlet condition of the mold computational simulation is from the nozzle computational simulation output. Variables are passed by a data file for initial conditions at inlet (see Appendix E and F). Figure 6.14 and 6.15 shows that these variables are passed consistently. Because the slide gate orientation is 90°, the flow at the two ports is assumed to be identical. Thus, flow in the caster is assumed to be symmetric on left and right sides. Only half of the mold is simulated. Figure 6.16 shows the geometry of mold domain. The top surface is set as a non-slip wall boundary considering the slow movement of the high viscosity slag. The wide face and narrow face are set as non-slip wall as well. The outlet is set as a pressure boundary. Other parameters are shown in Table 6.8. The standard K-ε model is used to simulate multiphase flow in the mold and the MUSIG model is used to model bubble breakup and coalescence. About 13 CPU hours are needed for 1000 iterations on NCSA SGI supercomputer which gives a fully convergence (decreasing final residuals below 0.1% of original residual).

6.8 Results and Analysis

6.8.1 Influence of Casting speed on Flow in the Continuous Caster

Casting speed has a great influence on flow in the nozzle. High casting speed needs a high liquid velocity and gate opening in nozzle. High liquid velocity tends to produce unbalanced flow and a large jet horizontal angle at nozzle port. For case A (high casting speed), the horizontal angle of the jet is 7°. For case B (low casting speed), the horizontal angle of the jet is 3°. The high horizontal angle directs more liquid toward the wide face and makes the flow more complex at high casting speed. Low casting speed makes liquid velocity in the nozzle small. Small liquid velocity produces larger bubbles according to Bai's model [16]. Large bubbles leave the jet more quickly after it enters the mold and imparts less buoyancy to the jet. The jet is bent up less and has more tendency to form a double roll flow pattern as explained in section 5. Figure 6.17 and 6.18 show that high casting speed has a complex flow pattern at the center plane while lower casting speed has a barely double roll flow pattern. The horizontal flow might help to prevent changing from single roll to double roll or vice versa by causing a relatively wide complex flow range. With gas percentage varying in a small range (11% for Figure 6.17 and 8.9% for Figure 6.18), increasing casting speed has no great influence to the flow. With gas flow rate constant (13 SLPM for both Figure 6.17 and Figure 6.19), increasing casting speed from 14.8 mm/s (Figure 6.19) to 23.2 mm/s (Figure 6.17) changes the flow from single roll to complex flow. Therefore, gas fraction is a much better indicator to identify the flow pattern in the mold than casting speed and gas flow rate.

6.8.2 Influence of Gas Fraction on the Flow Pattern.

Gas fraction is a good indicator to identify the flow pattern in the mold. High gas fraction increases buoyancy applied to the jet and bends up the jet more to produce a single roll flow pattern. Low gas fraction has more tendency to form a double roll. With increasing gas fraction, the flow pattern will evolve through three different flow patterns: double roll, complex and single roll. Changing the flow patterns brings liquid level fluctuation at meniscus and is very detrimental. Gas fraction should be controlled to avoid flow pattern switching. Plant measurements show that there is a critical gas fraction for flow switching. Below the gas fraction, the flow is double roll. Above the gas fraction, the flow is single roll. The critical gas fraction does not change with the casting speed or throughput. Figure 6.24 shows the computational simulation results of the relationship between the gas fraction and the flow patterns for four different slab widths. It gives the critical gas fractions where double roll changes to single roll for different slab widths. The critical gas fraction at which the flow switch from double roll to single roll or vice versa is not simply a point. There is a range of gas fraction which is called “complex zone” here for the flow pattern transition. Above this zone, the flow pattern is single roll. Below it, the flow pattern is double roll. In the complex zone, the flow pattern is complex.

Even the complex flow is relatively stable and is expected to have no harmful effect to slab quality such as slivers (case A). However, a flow pattern change which increases liquid level fluctuations is suspected to be more harmful in this aspect (case B) [26]. Plant measurements from LTV steel [26] show that, during ladle changing, the casting speed dropped from 23.2 mm/s (case A) to 14.8 mm/s (case B) without

decreasing the gas flow rate. The computational simulations show that this process experiences a flow pattern change from complex flow to single roll due to the increase in gas fraction from 11% to 16.4% [Figure 6.24]. Production records show that slabs produced in this period had more sliver problems [26]. The flow pattern switching happening in this period could explain the sliver defects. Although a stable complex flow pattern should have less tendency to cause high liquid level fluctuation and entrap flux, its complex feature is still harmful. Gas is less likely to escape the upper recirculation zone and is eventually entrapped in the slab to form pencil pipes. Case A (11% gas) experienced more pencil pipe defects than case B even though case B has a much higher gas fraction (16.4% gas). Gas fraction should be kept stable and away from complex zone to give a stable flow pattern. Figure 6.24 and Figure 6.25 show the relationship between gas fraction and flow pattern.

6.8.3 Influence of Slab Width on the Flow Pattern

Decreasing slab width decreases the distance between the nozzle port and the narrow face. The jet has less time to be lifted up to hit the top surface. Thus, the narrower slab is more likely to form a double roll flow pattern than a wider slab. Figure 6.25 shows this tendency. With a gas flow rate of 15 SLPM and throughput of 2 ton/min, 1.854 m and 1.600 m slab show single roll, 1.321 m slab shows complex flow and 1.016 m slab shows double roll. With a casting speed of 16.9 mm/s and gas percentage of 15%, 1.854 m slab shows a single roll, 1.600 m slab shows a complex flow, and 1.321 m and 1.016 m slabs show double roll (Figure 6.24). It is concluded that, keeping casting speed and gas fraction constant, decreasing slab width is likely to have double roll flow in the mold.

6.8.4 Influence of Submergence Depth on the Flow Pattern

Increasing submergence depth increases the vertical distance from the jet to the top surface. It is more difficult for the bent jet to hit the top surface for deep submergence depth. Thus, deeper submergence depth has more tendency to form a double roll. Figure 6.26 shows how the flow pattern changes with submergence depth for a given slab width of 1.854 m. Figure 6.27 and Figure 6.28 show velocity vector plots at center plane which show the relationship between SEN submergence depth and flow patterns. For a given slab width, submergence depth and gas fraction, throughput has little effect on the flow pattern.

6.8.5 Difference between flow in water model and steel caster

In Figure 6.4 through Figure 6.7, PIV measurements in water model verified the computational simulations. The model then is applied to the corresponding steel cases. The results of steel caster computational simulation are different from that of the computational simulation of water models. For case A, the steel caster computational simulation shows a complex flow which is more closer to a double roll than a single roll. The water model computational simulation and PIV measurements show that its water model counterpart is normally single roll. For case B, the steel caster computational simulation shows a slight double roll flow. The water model computational simulation and PIV measurements show its water model counterpart is always single roll. Plant measurements [26] showed that case A is normally double roll and case B is mostly double roll but experiences some flow pattern switching. The plant measurements also show that, keeping gas flow rate constant and decreasing casting speed from 23.2 mm/s

to 14.8 mm/s, the flow pattern changes from double roll to single roll. This phenomenon is matched by the computational simulations. Keeping gas flow rate at 13 SLPM and changing casting speed from 23.2 mm/s (Figure 6.17) to 14.8 mm/s (Figure 6.19), the flow pattern changes from double roll to single roll. The plant measurements are consistent with computational simulations of flow in the steel caster. Although both plant measurements in steel caster and 0.4-scale water model experiments match their computational simulations, they are not consistent with each other. Computational simulation and measurements show that the flow pattern in the steel caster is sometimes very different from that in a scale water model and the steady, multiphase K-ε computation can match both. The most important cause for the difference is the reduced-scale of the water model combined with the Froude-based velocity scaling criterion used to choose the water model flow rates.

6.9 Conclusions

1. Multiphase computational simulations of a 0.4-scale water model using the MUSIG model match experimental measurements. Plant measurements of the flow pattern by MFC sensors also verify multiphase computational simulation of the real caster. The 0.4-scale water model does not always match flow in the real caster and in some cases predicts the wrong flow pattern (e.g. 1.854 m slab, casting speed of 14.8 mm/s and gas fraction of 8.5%).
2. Transient increases in the liquid flow rate, such as required to accommodate level changes, may produce significant changes in the flow pattern, if the conditions are near critical.

3. Gas changes the flow pattern by applying buoyancy force to the liquid jet. Increasing gas fraction changes the flow from double roll to an intermediate state--complex flow, then to single roll. For a given slab width, there is a critical gas fraction range where the flow pattern is in transition from double roll to single roll. This range does not change too much with throughput for a given slab width and submergence depth. The critical gas fraction increases when decreasing slab width.
4. Slab width has influence on flow patterns. For a given casting speed and gas fraction, narrow slab tends to form double roll while wide slab tends to form single roll.
5. Nozzle submergence depth also affects the flow pattern in the mold. Deeper submergence depth has more tendency to form double roll.

Table 6.1 Nozzle parameters (steel caster computational simulation)

Dimension & Condition	Case A Nozzle	Case B Nozzle
UTN top diameter (mm)	100	
UTN length (mm)	310	
Gate thickness(mm)	45	
Gate diameter(mm)	70	
Shroud holder thickness (mm)	66	
SEN length (mm)	776	
SEN bore diameter (mm)	78	
SEN submerged depth (mm)	165	
Port width X height(mmXmm)	78X78	
Port thickness(mm)	28.5	
Port angle (down)	15°	
Recessed bottom well depth (mm)	12	
Slide gate orientation	90°	
Gate opening (F_L)	58.6%	52%
Casting speed (V_C) (m/min)	1.40	0.89
Liquid flow rate (l/min)	590.5	375.8
Tundish depth (H_T) (m)	0.927	
Argon injection flow rate (Q_G) (SLPM)	13	6.3
Argon injection (hot) volume fraction f_g	11%	8.9%

*Blank in second column is the same as the first column.

Table 6.2 Measured bubble size distribution in LTV 0.4-scale water model for case A (23.2 mm/s (55ipm) + 13 SLPM/11% hot gas)

Region	≤1mm	1-2mm	2-3mm	3-4mm	4-5mm	≥5mm
A	>10	16	20	17	2	0
B	>5	8	16	10		
C	>10	9	13	5		
Total	25	33	49	32	2	
Volume (ml)	13.08	58.29	400.68	718.01	95.38	
Volume percentage	1.02%	4.53%	31.15%	55.83%	7.42%	
Average	2.59mm					

Table 6.3 Measured bubble size distribution in LTV 0.4-scale water model for case B (14.8 mm/s (35ipm) + 6.3 SLPM/8.5% hot gas)

Region	≤1mm	1-2mm	2-3mm	3-4mm	4-5mm	5-6mm	6-7mm	7-8mm	8-9mm	9-10mm	≥10mm
11	5	2	1								
12	>11	3	1	1							
13	>14	6	4	2	1						
14	>16	4	2	1	1						1
21	9	3									
22	>13	4	1		1						
23	>20	3	2	1							
24	>20	4	2	2		2				1	
31	>10	4									
32	>10	3	2								
33	>17	4	3	1		1					
34	5	6	2		1						
35	6	1	3		1						
44	6	4	1								
45	>12	5	2								
Total	>174	57	26	8	5	3				1	1
Volume (ml)	91.05	100.68	212.60	179.50	238.44	261.20				448.69	523.69
Volume percent	4.43%	4.90%	10.34 %	8.73%	11.60 %	12.71 %				21.83%	25.48%
Average size	2.43mm										

Table 6.4 Parameters for scale water model for case A

	Normal Conditions
Mold Width W (mm) x Thickness H (mm)	730 x 80
Mold Height (mm)	950
Nozzle Submergence Depth (top surface to top of port)	80
Nozzle Inner Diameter (mm)	31
Nozzle Port Width (mm) x Height (mm)	31 x 31
Jet Angle	30° down
Inlet Jet Spread Angle	0°
Water Flow Rate Q_w (SLPM)	58.59 (15.5 GPM)
Equivalent Steel Casting Speed (mm/s) $V_c = \frac{Q_w}{0.4 \times W \times 0.4 \times H \times \sqrt{0.4}}$	22.9
Gas Flow Rate (SLPM, hot volume)	7.43 (15.8 SCFH)
Gas Volume Fraction (%)	11.3
Inlet Velocity, V_x (m/s)	0.571
Inlet Velocity, V_z (m/s)	0.330
Inlet Turbulent Kinetic Energy, K_o (m^2/s^2)	0.044
Inlet Turbulence Dissipation Rate, ε_o (m^2/s^3)	0.999
Water Density (kg/m^3)	1000
Water Viscosity (m^2/s)	1×10^{-3}
Gas Density (kg/m^3)	1.20
Gas Viscosity (m^2/s)	1.7×10^{-5}
Average Bubble Diameter (mm)	2.590
Breakup Coefficient	0.5
Coalescence Coefficient	0

Table 6.5 Parameters for scale water model for case B

	Normal Conditions
Mold Width W (mm) x Thickness H (mm)	730 x 80
Mold Height (mm)	950
Nozzle Submergence Depth (top surface to top of the port)	80
Nozzle Inner Diameter (mm)	31
Nozzle Port Width (mm) x Height (mm)	31 x 31
Jet Angle	30° down
Inlet Jet Spread Angle	0°
Water Flow Rate (SLPM)	37.80 (10.0 GPM)
Equivalent Steel Casting Speed (mm/s) $V_c = \frac{Q_w}{0.4 \times W \times 0.4 \times H \times \sqrt{0.4}}$	14.8
Gas Flow Rate (SLPM, hot volume)	3.71 (7.9 SCFH)
Gas Volume Fraction (%)	8.9
Inlet Velocity, V_x (m/s)	0.358
Inlet Velocity, V_z(m/s)	0.207
Inlet Turbulent Kinetic Energy (m^2/s^2)	0.044
Inlet Turbulence Dissipation Rate (m^2/s^3)	0.999
Water Density (kg/m^3)	1000
Water Viscosity (m^2/s)	1×10^{-3}
Gas Density (kg/m^3)	1.20
Gas Viscosity (m^2/s)	1.7×10^{-5}
Average Bubble Diameter (mm)	2.43
Breakup Coefficient	0.1
Coalescence Coefficient	0

Table 6.6 Measured bubble size distribution for case A in double-needle experiments

Diameter of bubbles(mm)	<1.0	1.0	1.5	2.0	2.5	3.0	3.0-5.0	>5.0	Photo Frame No.
Data Set #1	4	3	1	1	0	1	1	0	159
Data Set #2	2	0	0	0	0	0	1	0	219
Data Set #3	0	0	0	1	1	0	1	0	265
Data Set #4	2	1	1	0	1	0	1	0	285
Data Set #5	2	2	1	0	1	0	0	0	315
Data Set #6	3	2	0	0	0	1	2	0	368
Data Set #7	0	1	1	0	0	0	2	0	397
Data Set #8	3	2	0	1	0	0	1	0	532
Data Set #9	2	1	1	0	1	0	2	0	631
Data Set #10	1	2	0	1	0	0	2	0	668
Total Number	19	14	5	4	4	2	13	0	
Total Volume (mm^3)	3.73	11.00	8.84	12.57	19.63	14.14	163.3	0	
Volume %	1.6	4.72	3.79	5.39	8.42	6.06	70	0	
Average	1.94 mm								

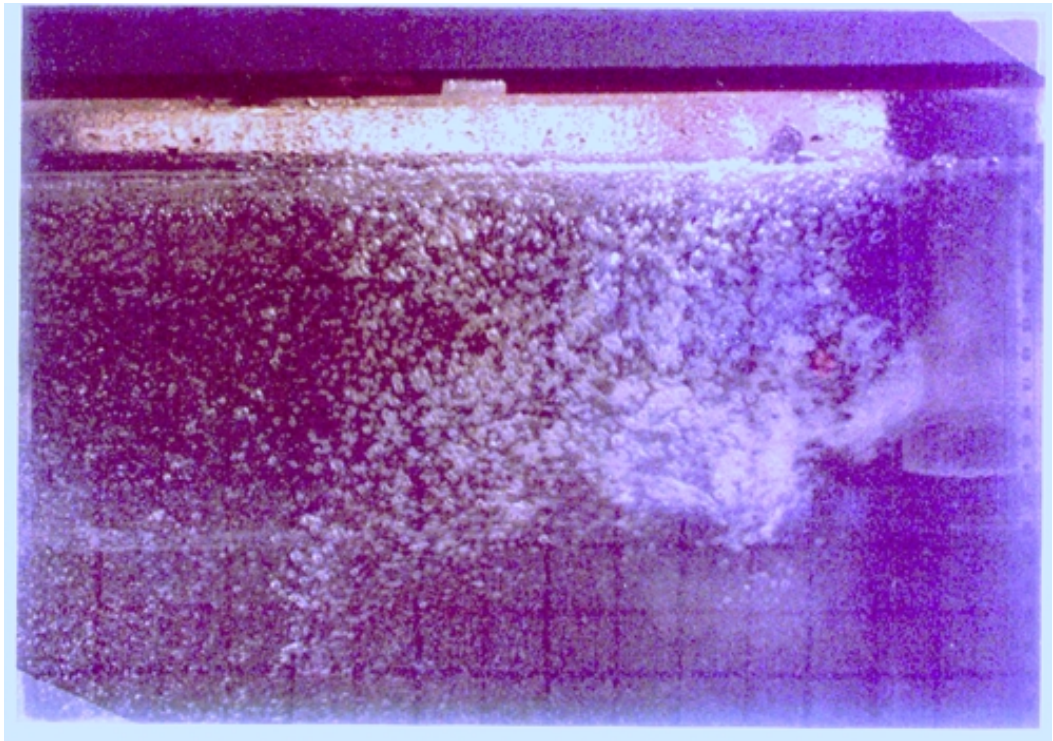
Table 6.7 Measured bubble size distribution for case B in double-needle experiments

Diameter of bubbles(mm)	<1.0	1.0	1.5	2.0	2.5	3.0	>3.0	Photo Frame No.
Data Set #1	0	0	2	5	4	2	0	141
Data Set #2	0	0	3	4	2	3	0	177
Data Set #3	0	0	1	7	3	0	0	211
Data Set #4	0	0	2	7	3	1	0	246
Data Set #5	0	1	6	4	4	0	0	286
Data Set #6	0	0	2	3	6	2	0	341
Data Set #7	0	1	3	7	3	0	0	380
Data Set #8	0	0	2	8	4	2	0	444
Data Set #9	0	0	1	6	5	0	0	481
Data Set #10	0	0	2	5	4	2	0	522
Total Number	0	2	24	56	38	12	0	
Total Volume (mm^3)	0	2.09	56.55	234.57	248.71	113.1	0	
Volume %	0	0.32	8.63	35.81	37.97	17.27	0	
Average	2.12 mm							

Table 6.8 Parameters in the steel caster computational simulation

	Case A	Case B
Mold Width (mm)	1854	
Mold Thickness (mm)	228	
Nozzle Submergence Depth (top surface to top of port) (mm)	165	
Nozzle Bore Inner Diameter (mm)	78	
Nozzle Port Height (mm)	78	
Nozzle Port Width (mm)	78	
Vertical Velocity in Nozzle (m/s)	2.05	1.31
Nominal Vertical Angle of Port Edges	15° down	
Inlet Jet Spread Angle	0°	
Casting Speed, V_c (mm/s)	23.2	14.8
Liquid Steel Density, ρ_l (kg/m^3)	7020	
Gas Density, ρ_{gas} (kg/m^3)	0.27	
Steel Laminar (Molecular) Viscosity, μ_o (kg/m/s)	0.00560	
Gas Viscosity, μ_{gas}	7.42E-5	
Surface Tension Coeff. (Steel-Argon) (N/m)	1.192	
Inlet steel flow rate (m^3/min)	0.584	0.376
Throughput (ton/min)	4.10	2.64
Inlet Gas Flow Rate (SLPM)	13	6.3 / 13
Inlet Gas Volume Fraction, f_{gas}	11%	8.5%/16.4%
Gravitational Acceleration, g (m/s^2)	9.8	
Breakup Coefficient	0.5	0.1
Coalescence Coefficient	0	0

*Blank in second column is the same as the first column.



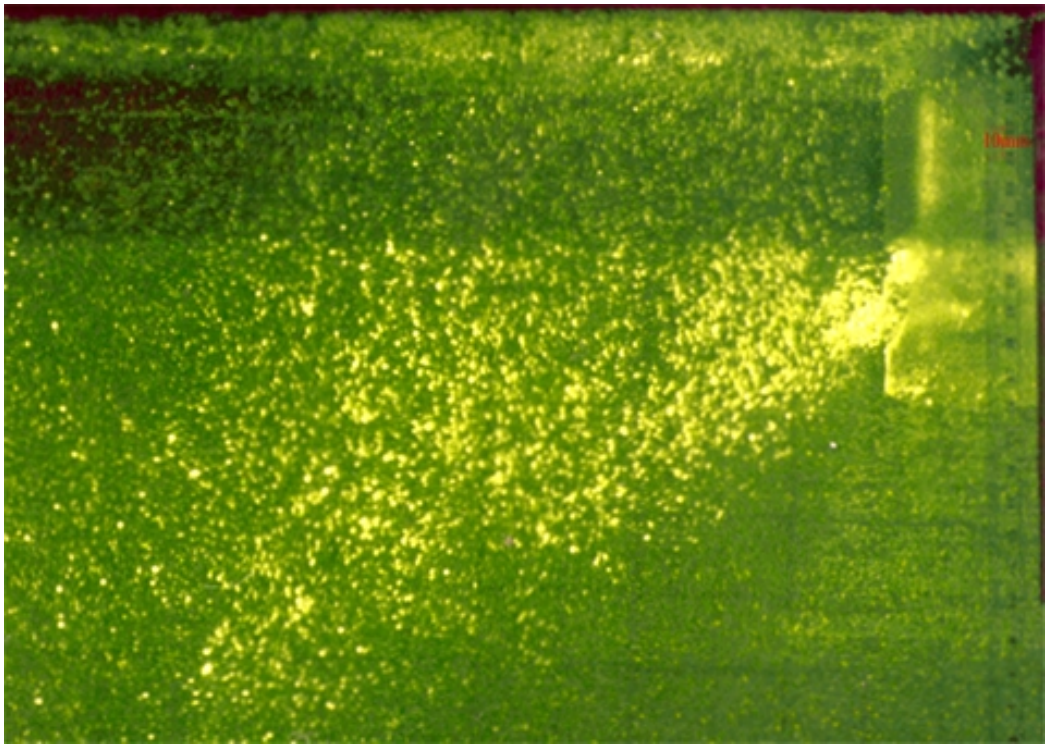
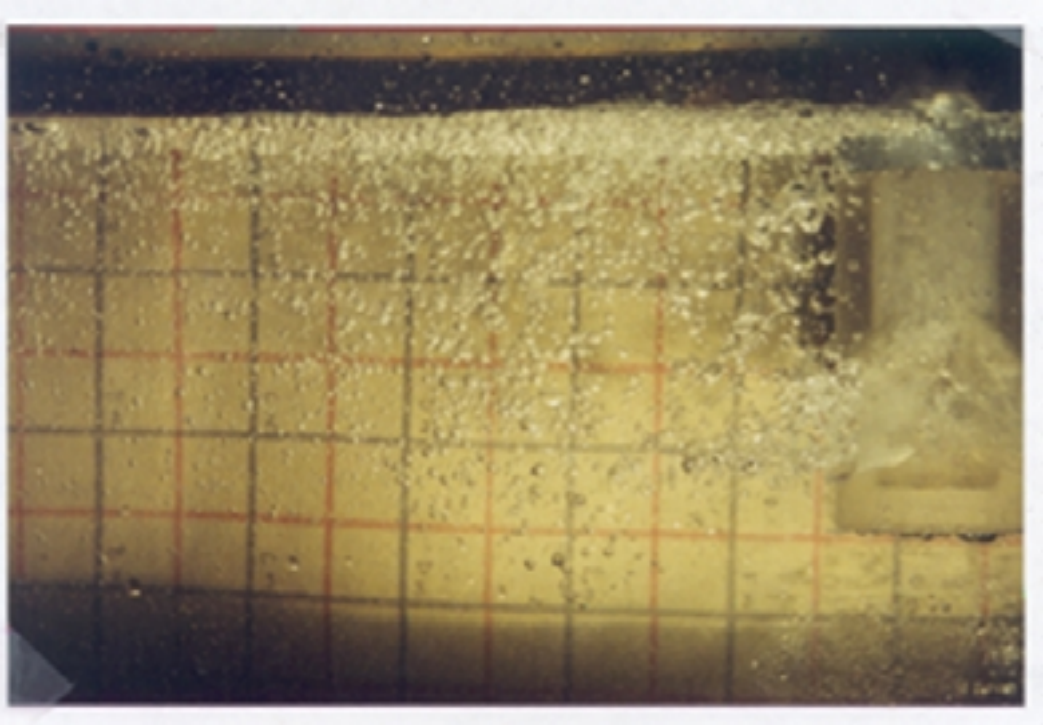


Figure 6.1 Bubble size in the water model
(case A, 23.2 mm/s + 13 SLPM/ 11% hot gas)



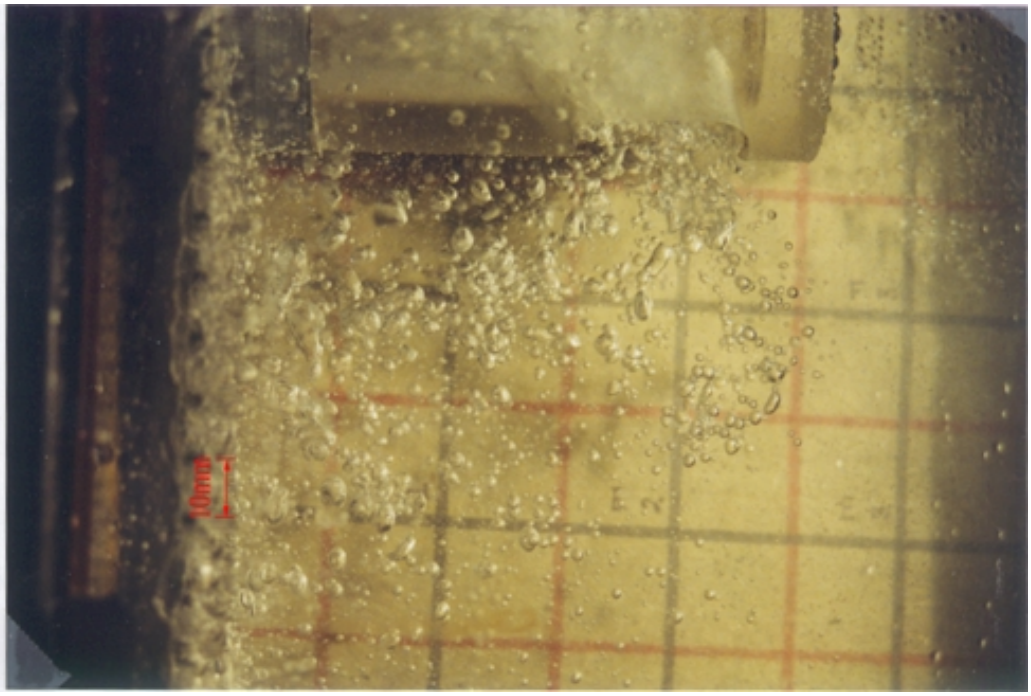


Figure 6.2 Bubble size in the water model
(case B, 14.8 mm/s + 6.3 SLPM/ 8.5% hot gas)

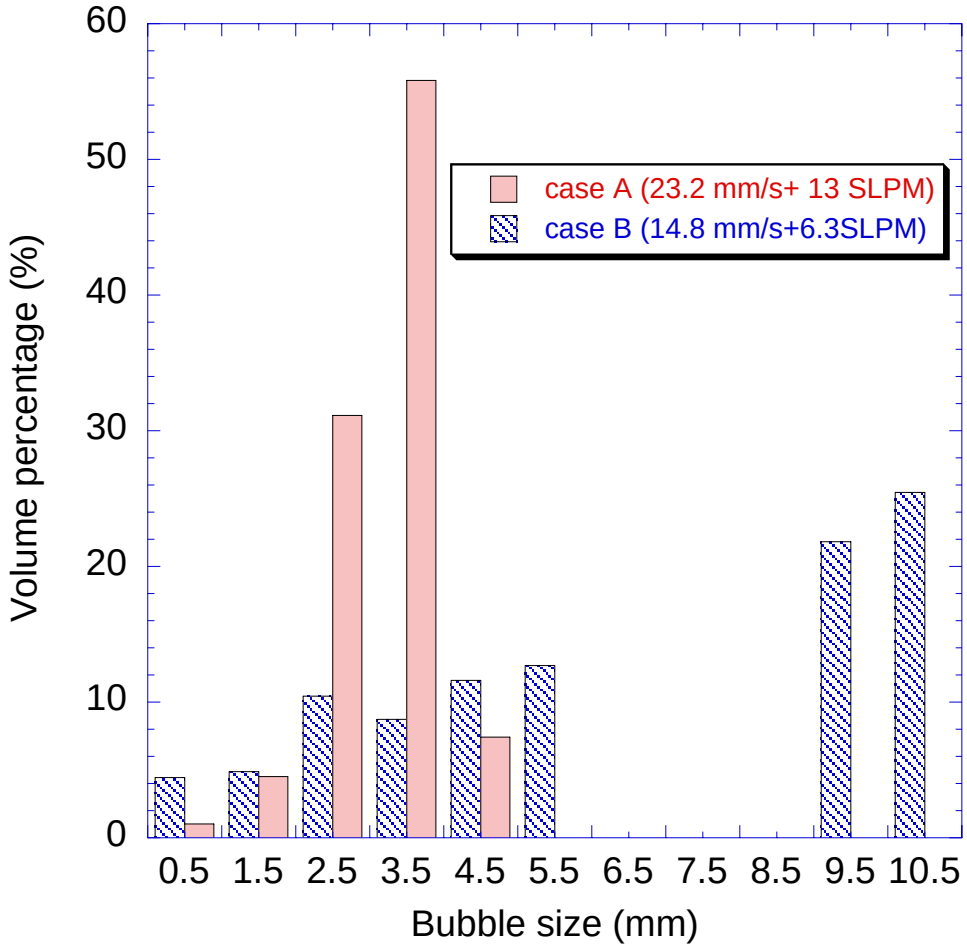


Figure 6.3 Bubble size distribution in the mold
(measurements in 0.4 scaled water model)

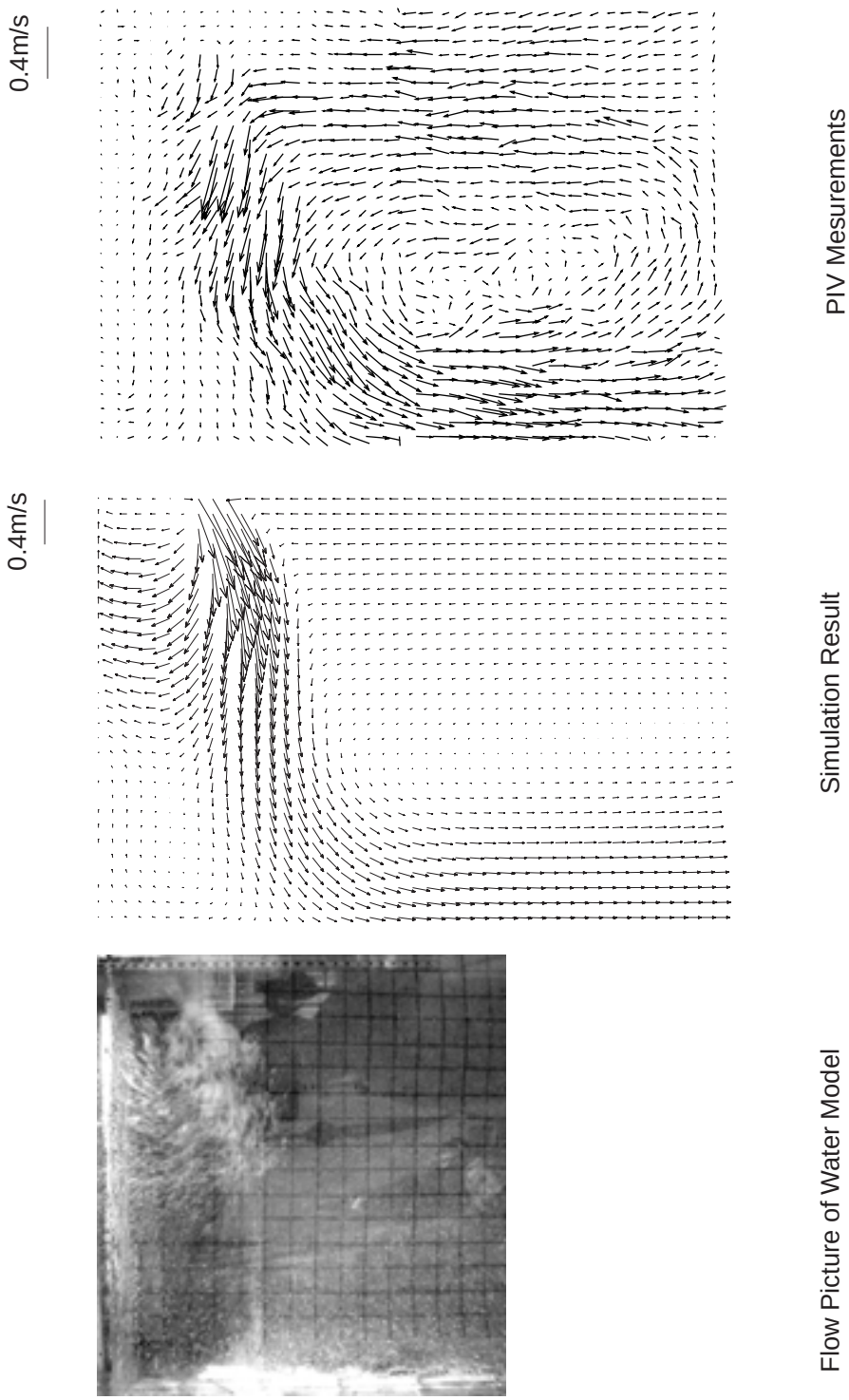
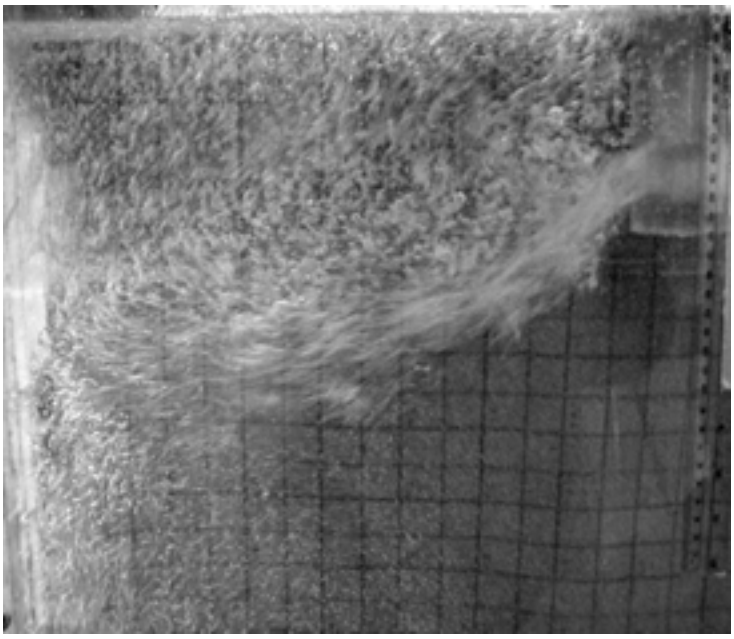
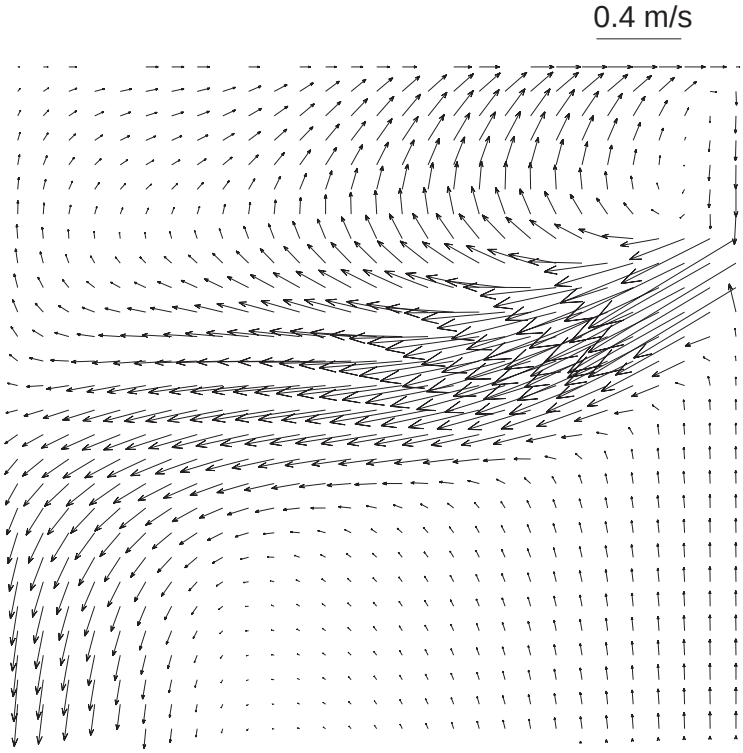


Figure 6.4 Comparison of velocity at centerplane between PIV measurements, computational simulation and eyeview for case A



Flow Picture of Water Model (55ipm +11% gas)



Liquid Velocity Vectors of Modeling (55ipm+11%gas)

Figure 6.5 Comparison of computational simulation and eyeviews while adjusting liquid level with 15% increase in liquid flow rate for case A

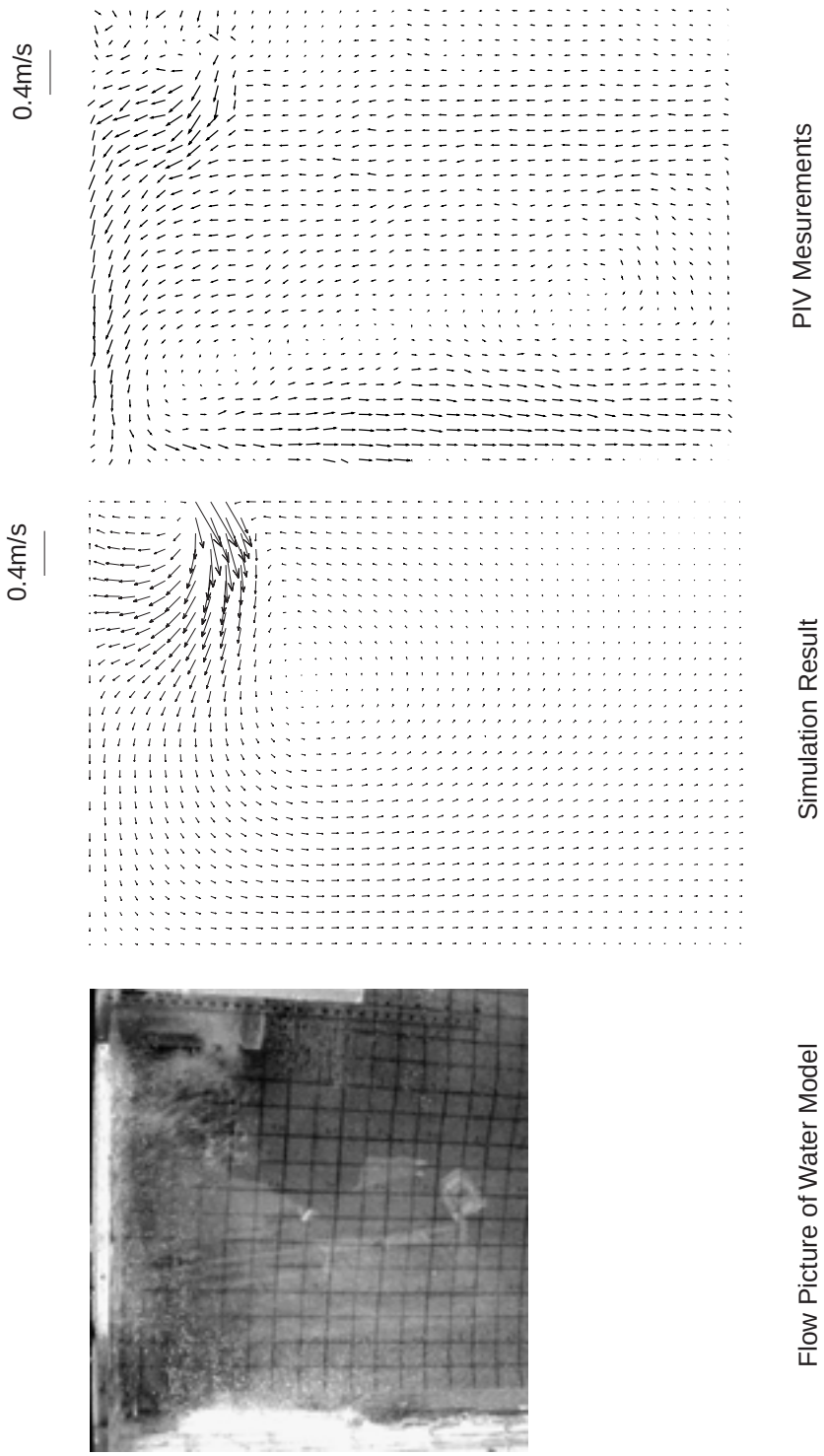
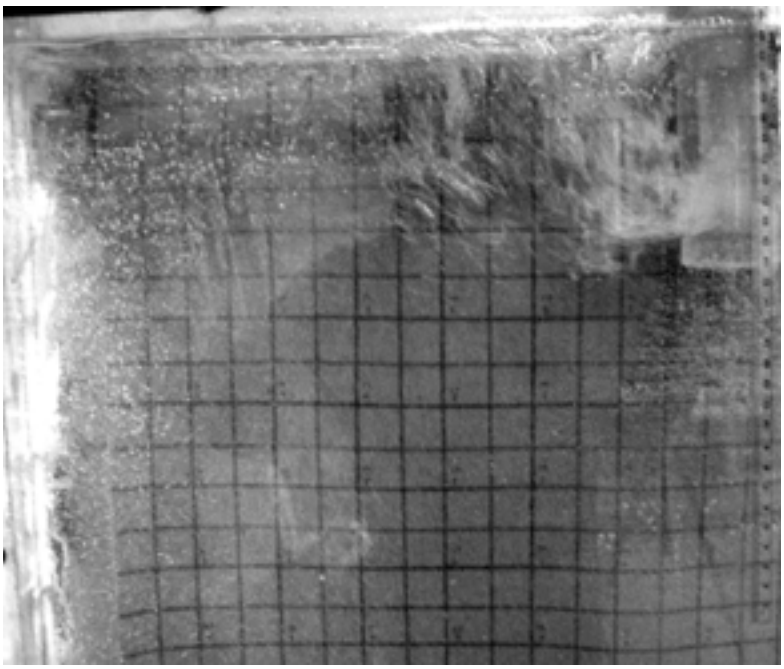
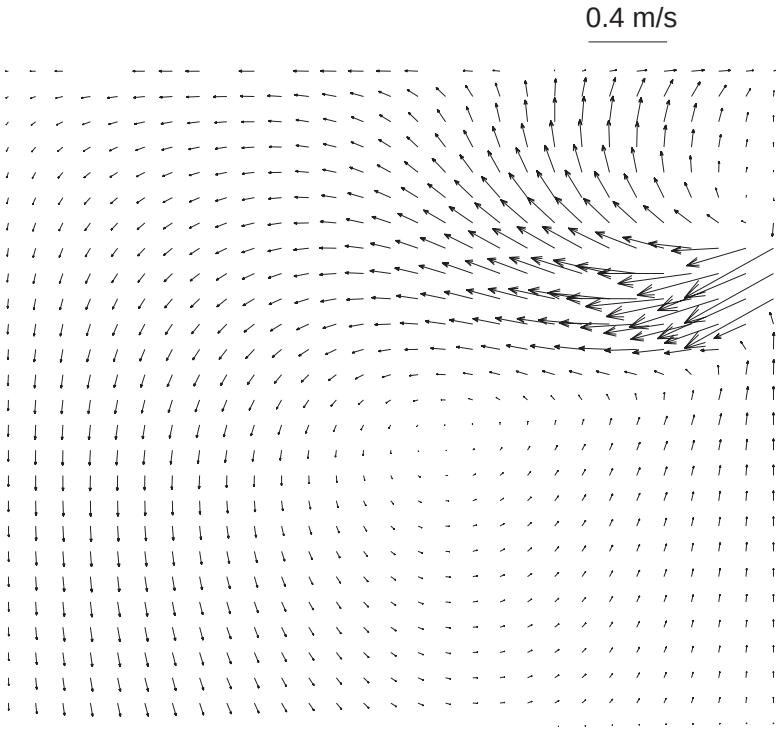


Figure 6.6 Comparison of velocity at centerplan between PIV measurements, computational and eyeview for case B



Flow Picture of Water Model (35ipm +8.5% gas)



Liquid Velocity Vectors of Modeling (35ipm+8.5%gas)

Figure 6.7 Comparison of computational simulation and eyeviews while adjusting liquid level with 15% increase in liquid flow rate for case B

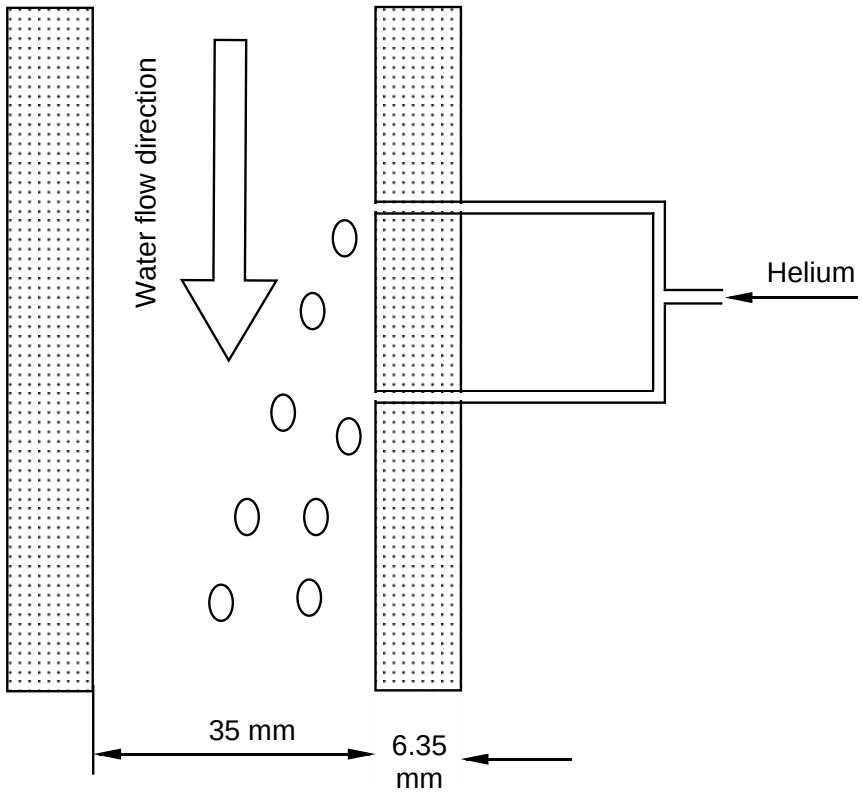


Figure 6.8 Schematic of water model of bubble formation

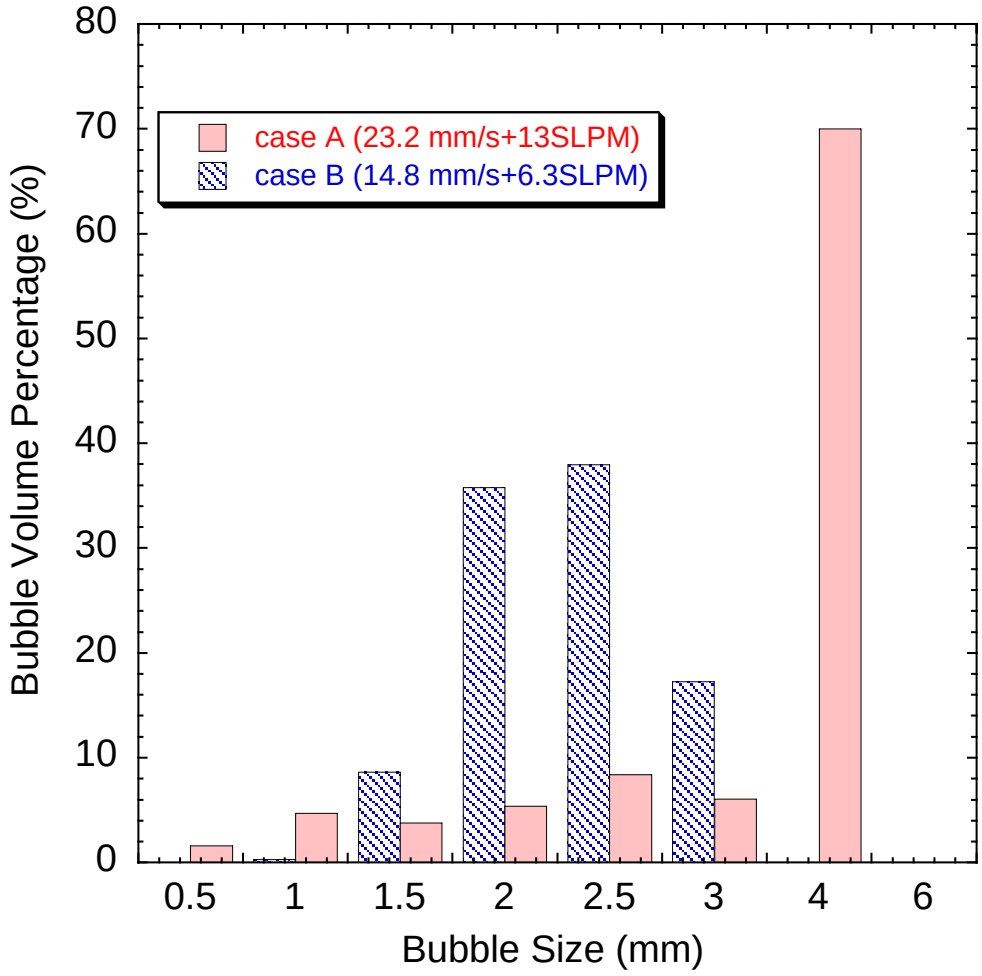


Figure 6.9 Assumed bubble size distribution in the steel caster nozzle (based on double-needle experimental measurements)

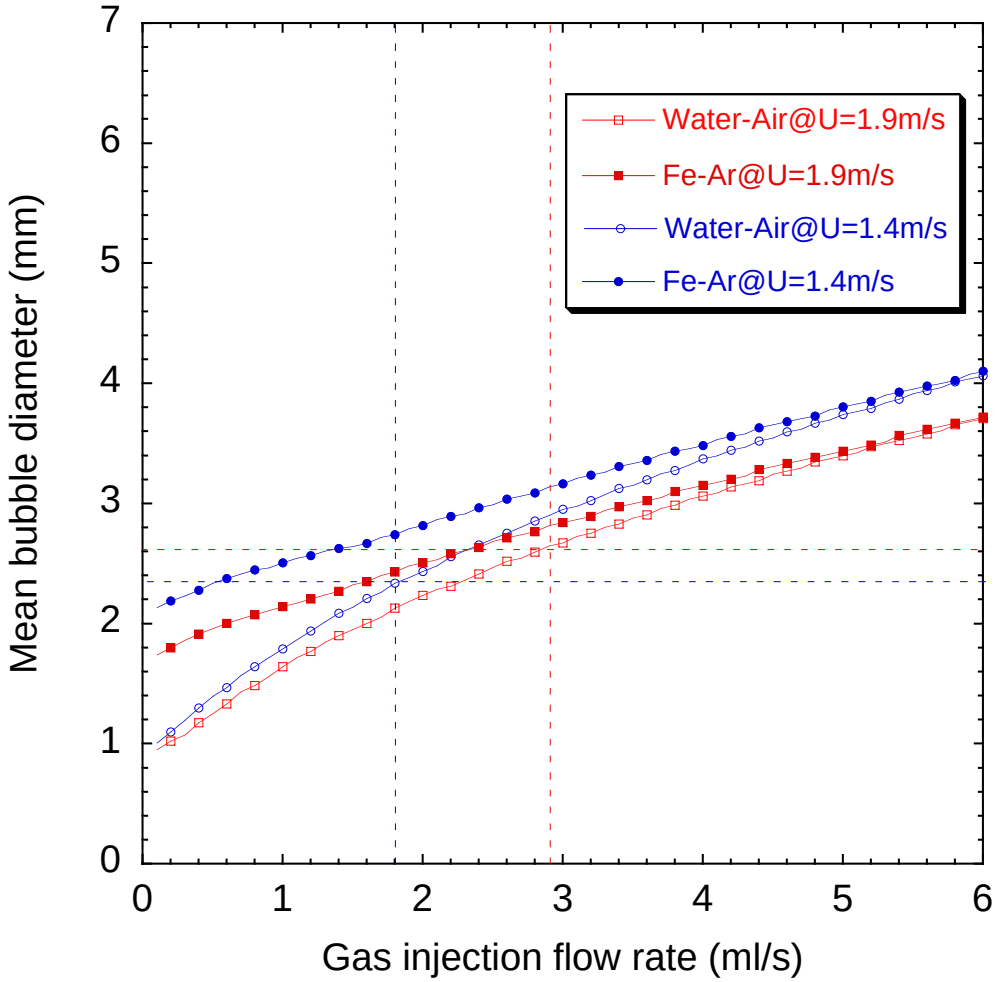


Figure 6.10 Average diameter of bubbles vs gas flow rate
(Rearranged from Figure 18 in [27]),

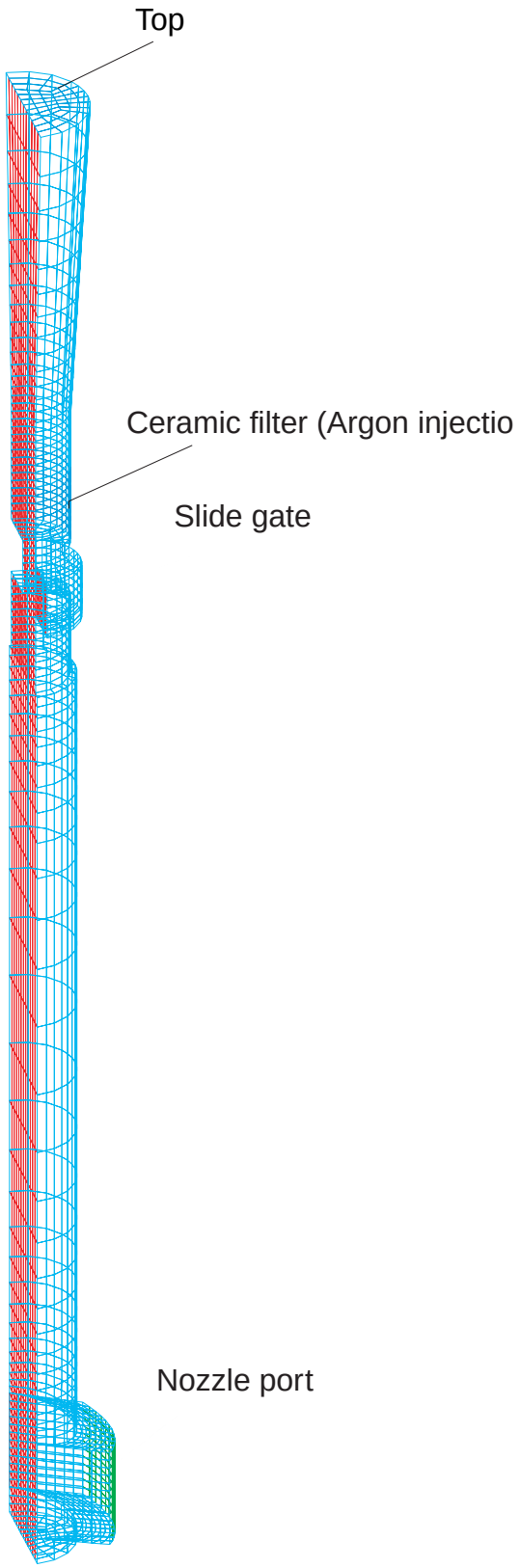


Figure 6.11 Geometry of the nozzle

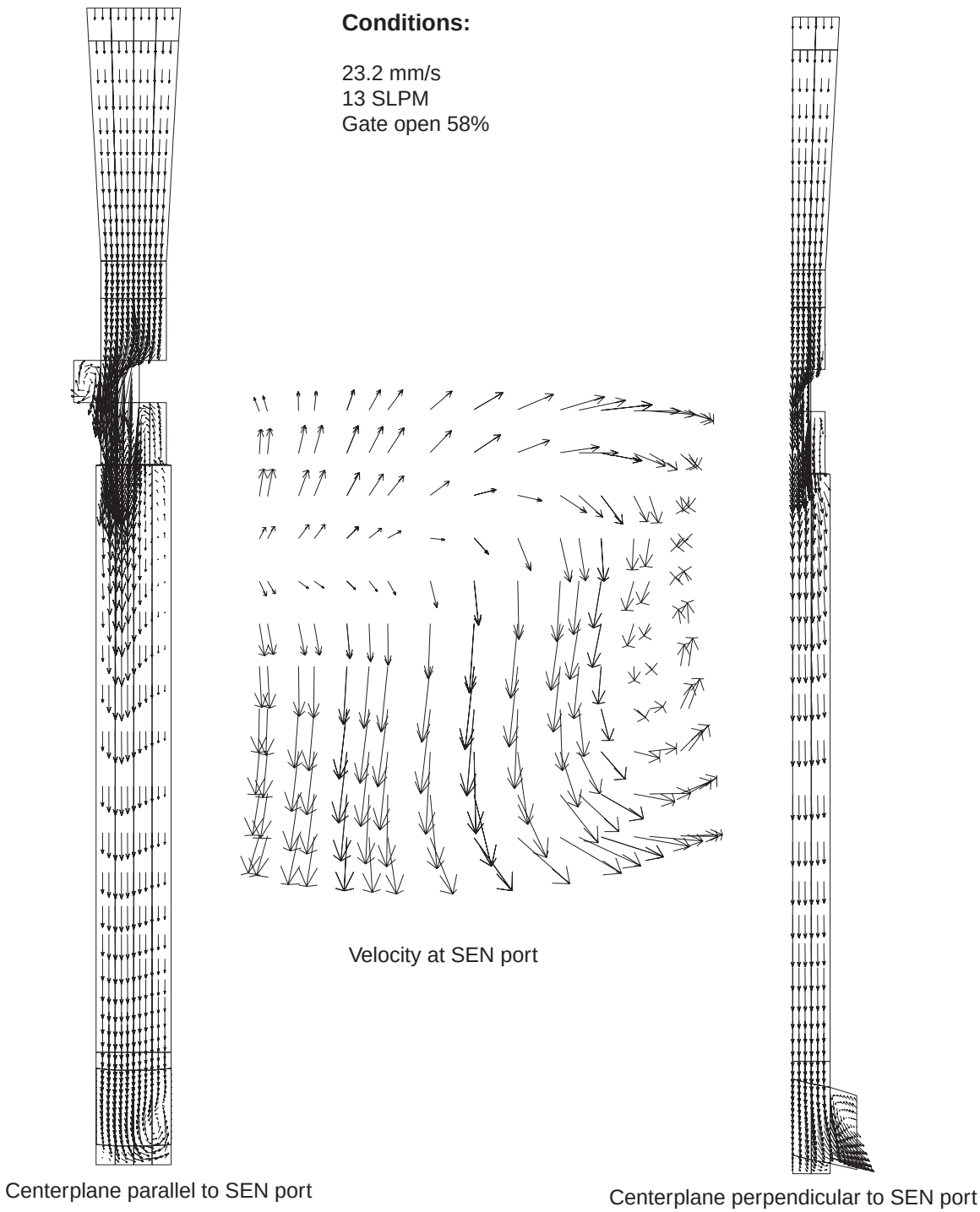


Figure 6.12 Liquid steel velocity in the nozzle for case A

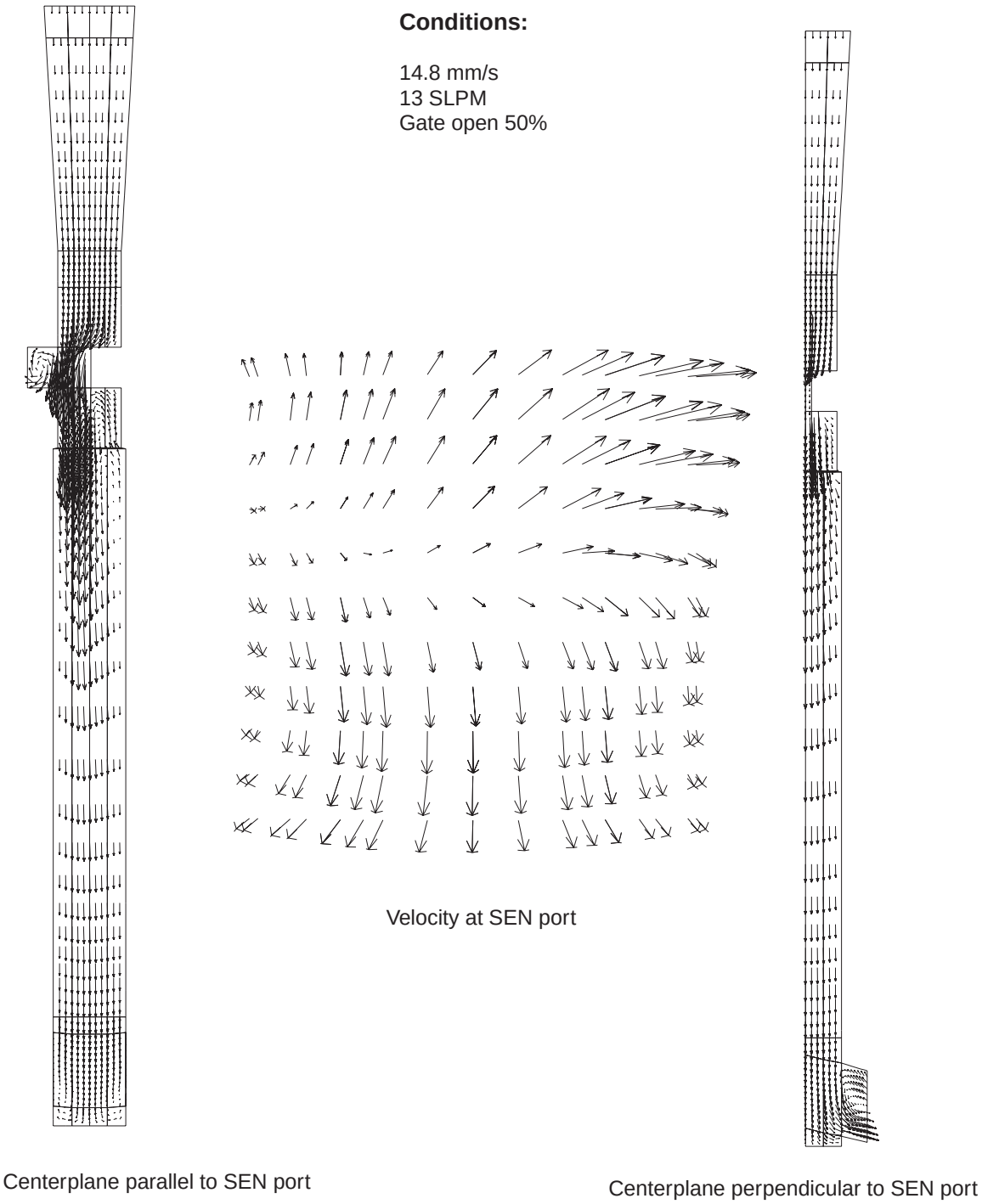


Figure 6.13 Liquid steel velocity in the nozzle for case B

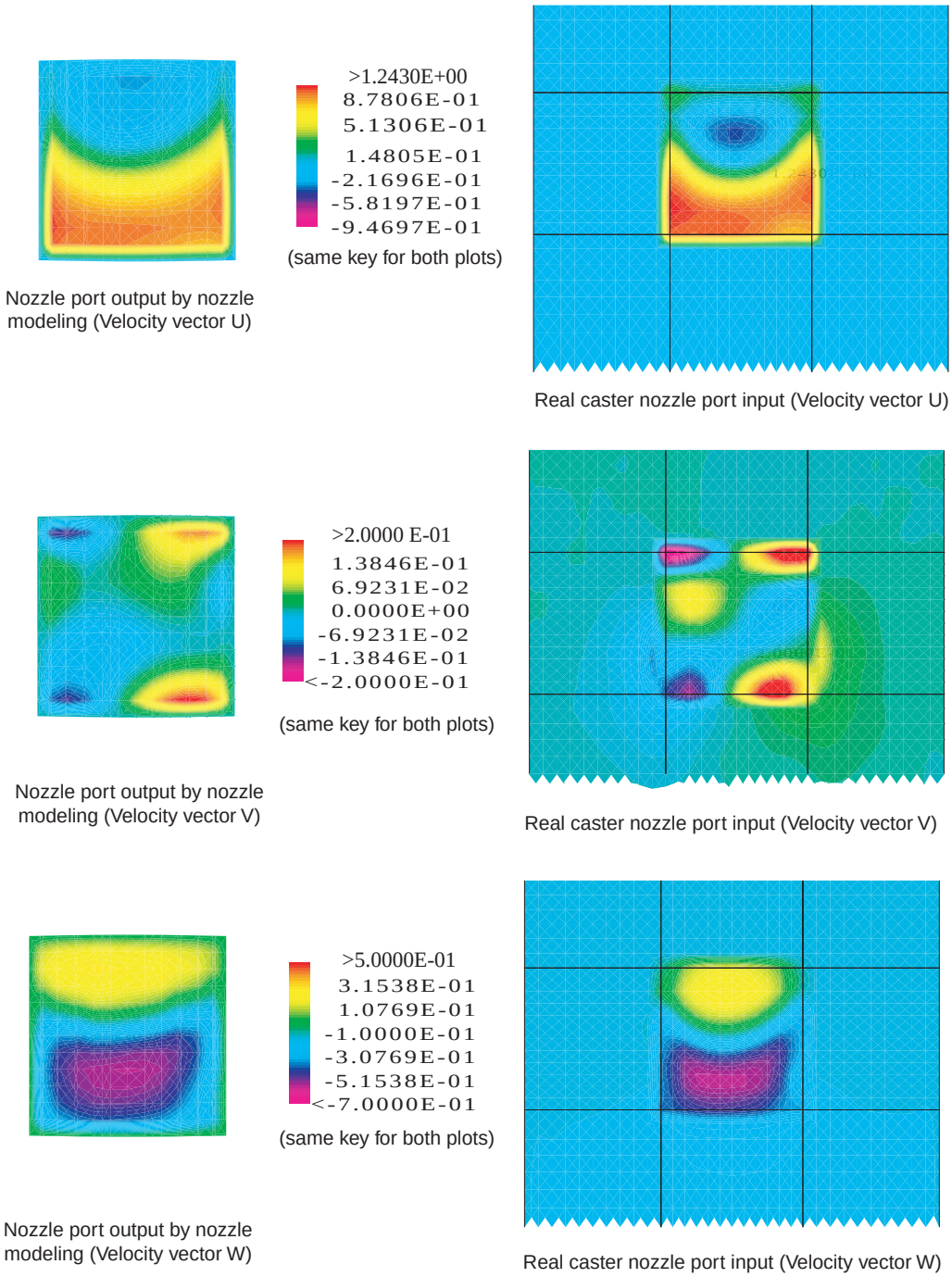


Figure 6.14 Correspondence of nozzle computational simulation output to mold computational simulation input

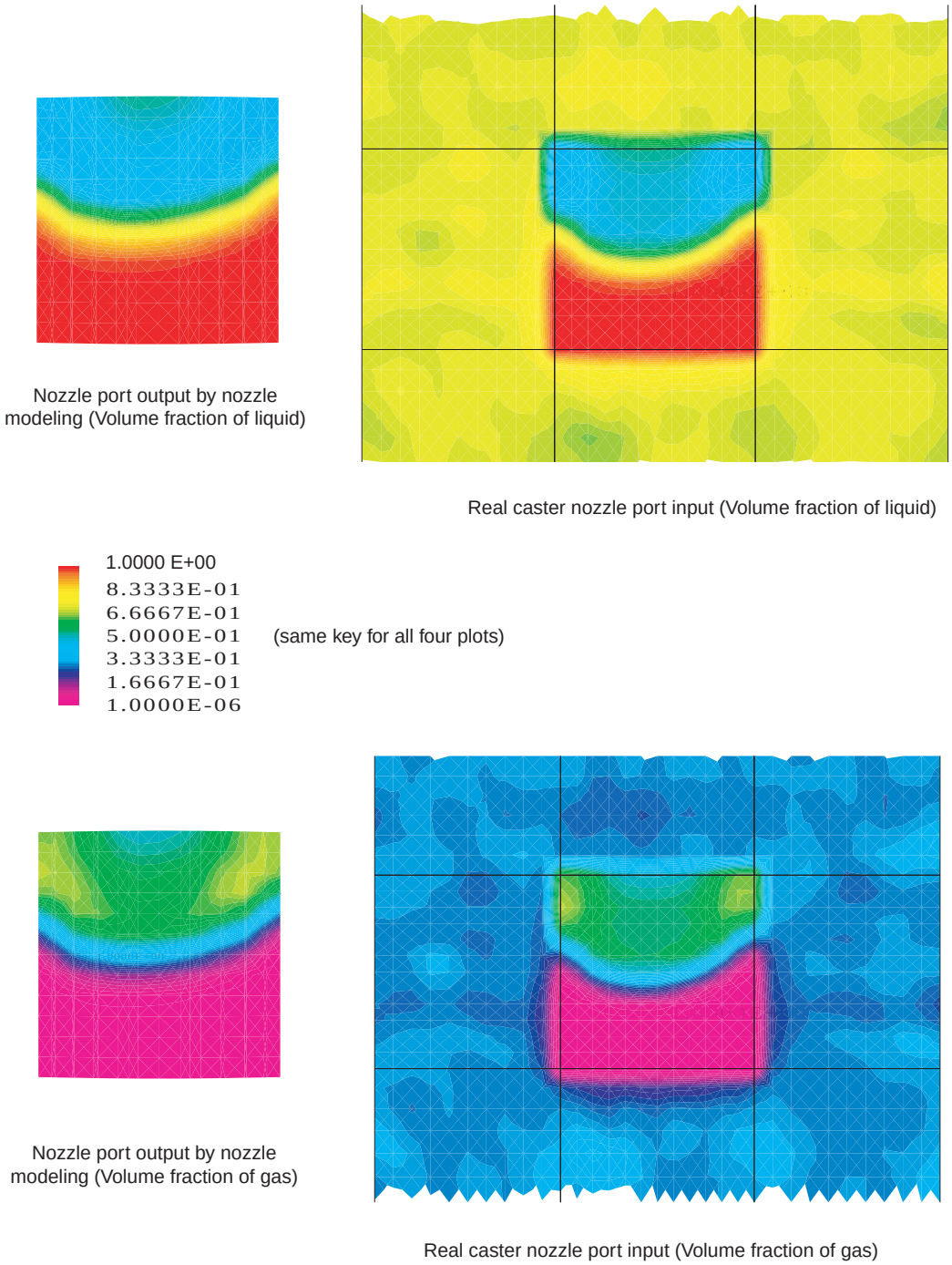


Figure 6.15 Correspondence of nozzle computational simulation output to mold computational simulation input

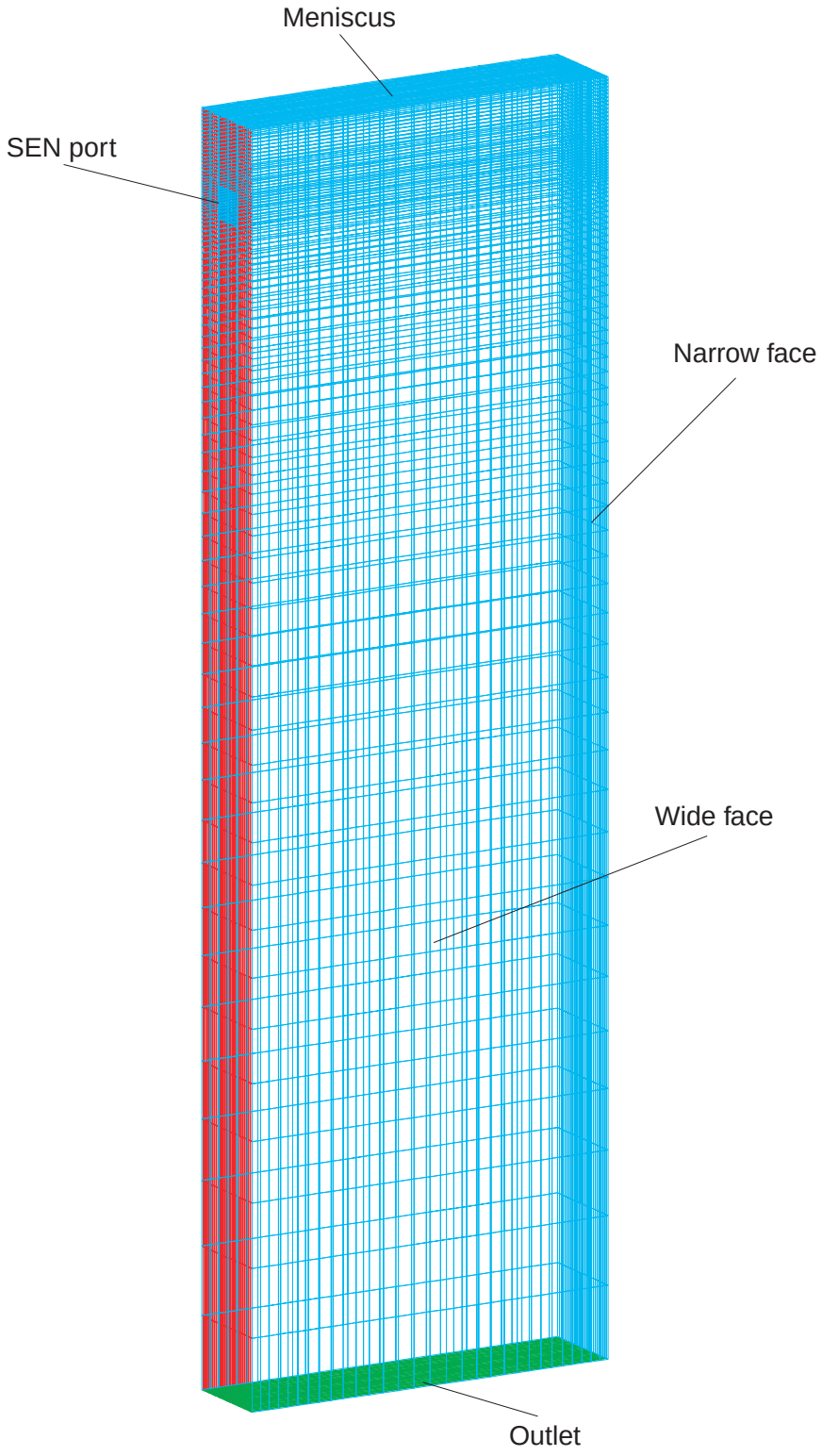


Figure 6.16 Computational simulation domain for mold with typical meshes

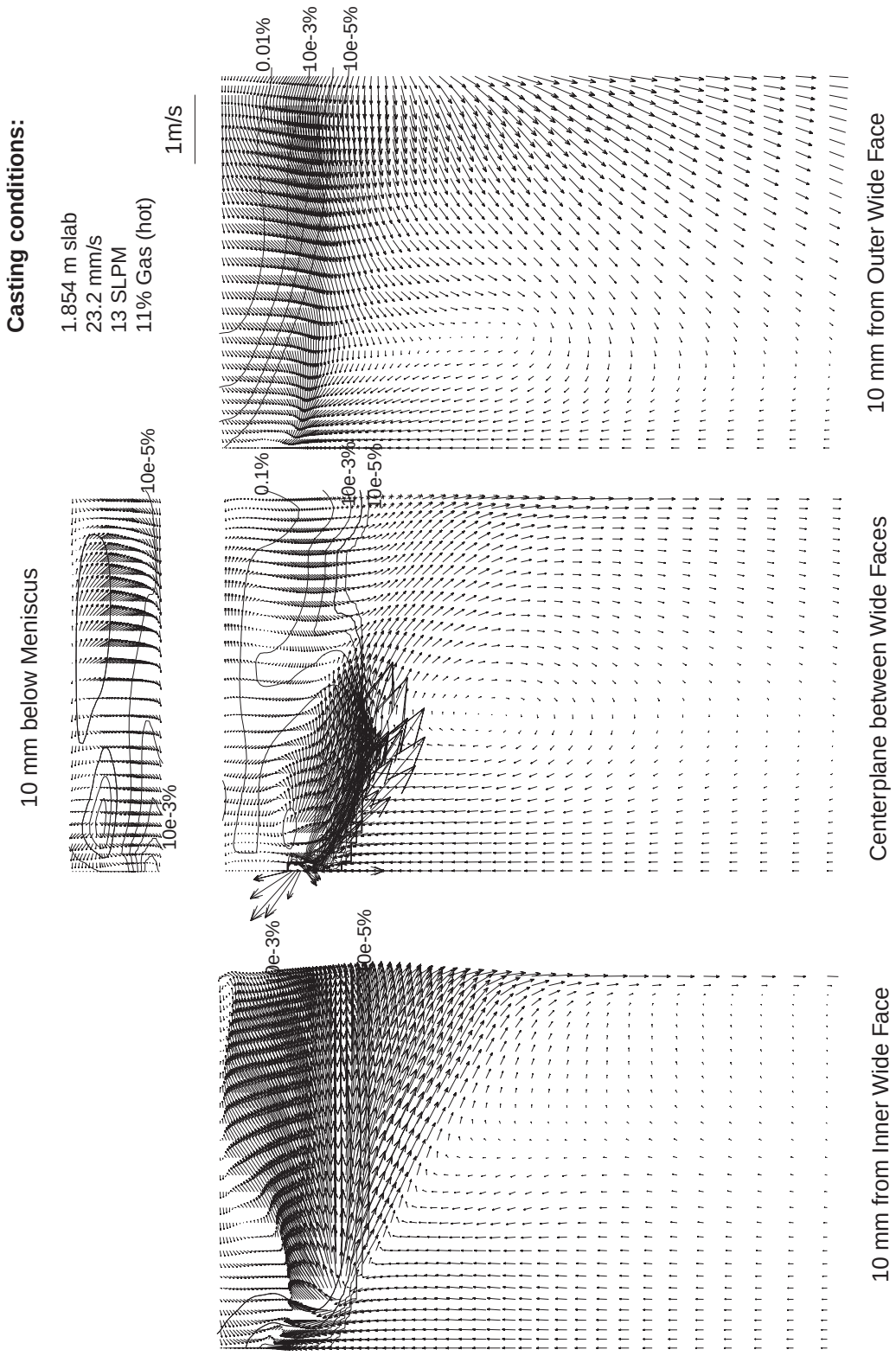


Figure 6.17 Computed steel flow pattern with distributed bubble size for case A (wide face)

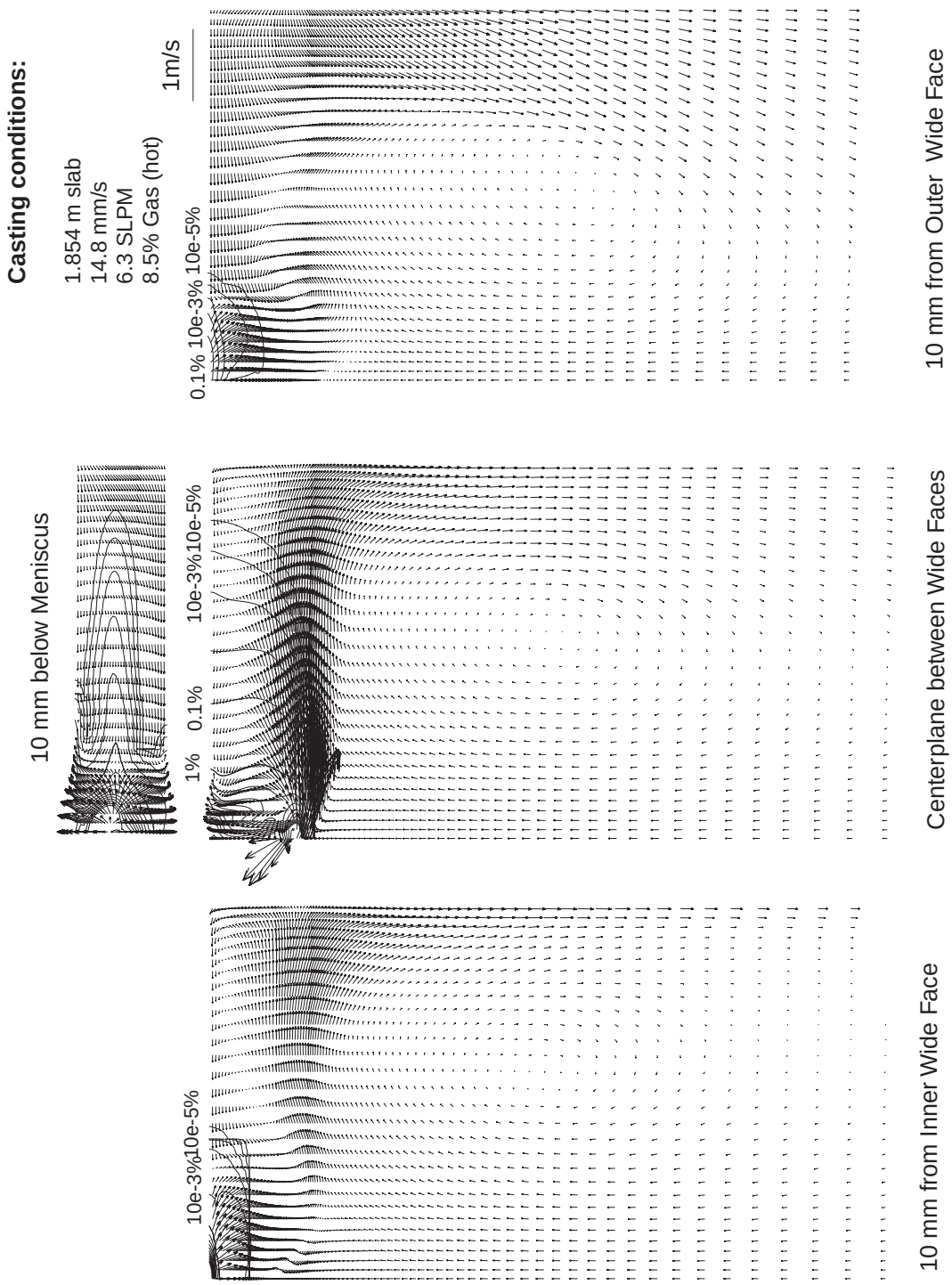


Figure 6.18 Computed steel flow pattern with distributed bubble size for case B (wide face)

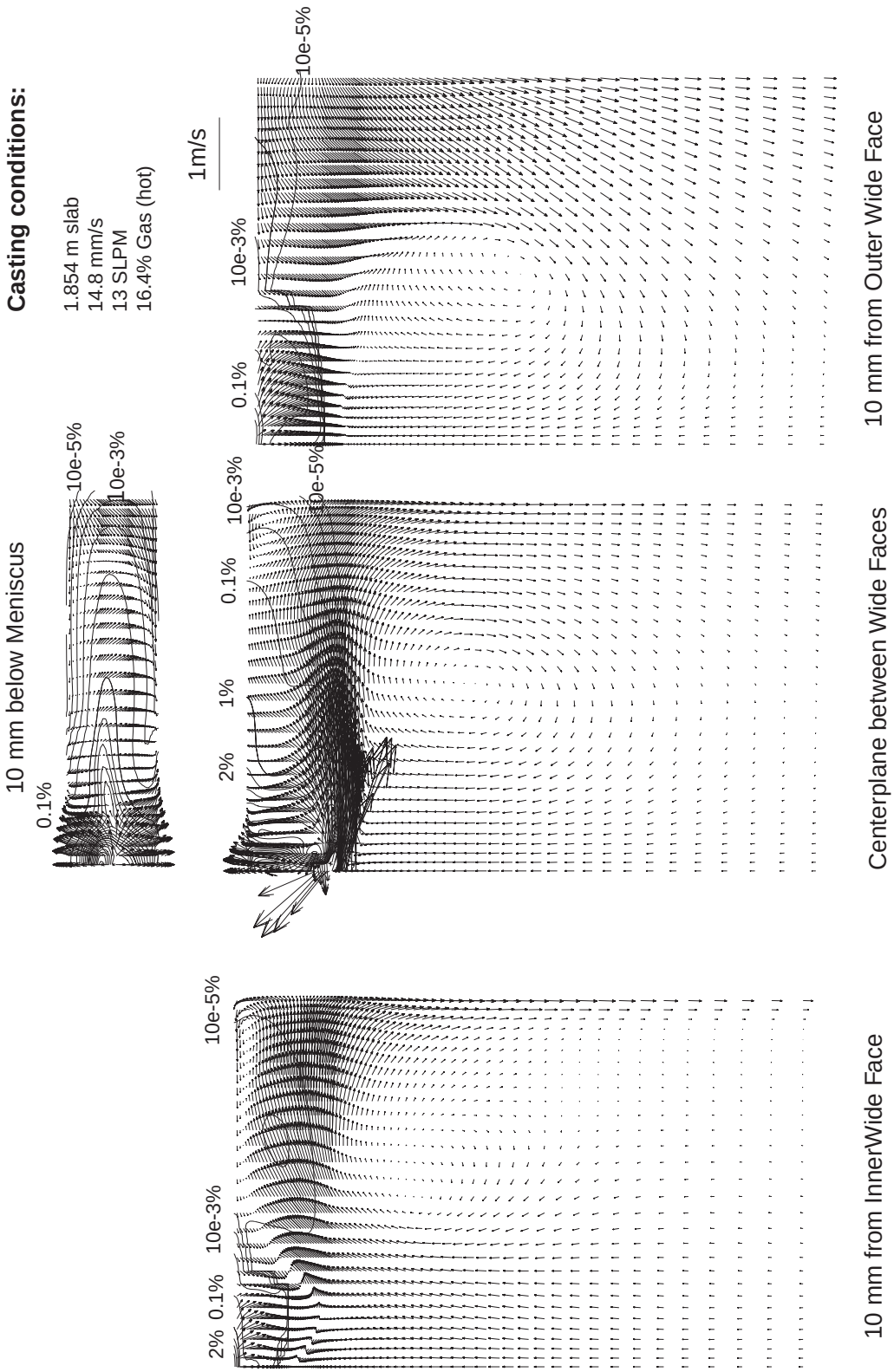


Figure 6.19 Computed steel flow pattern with distributed bubble size for case B with high gas fraction (wide face, 16.4% gas)

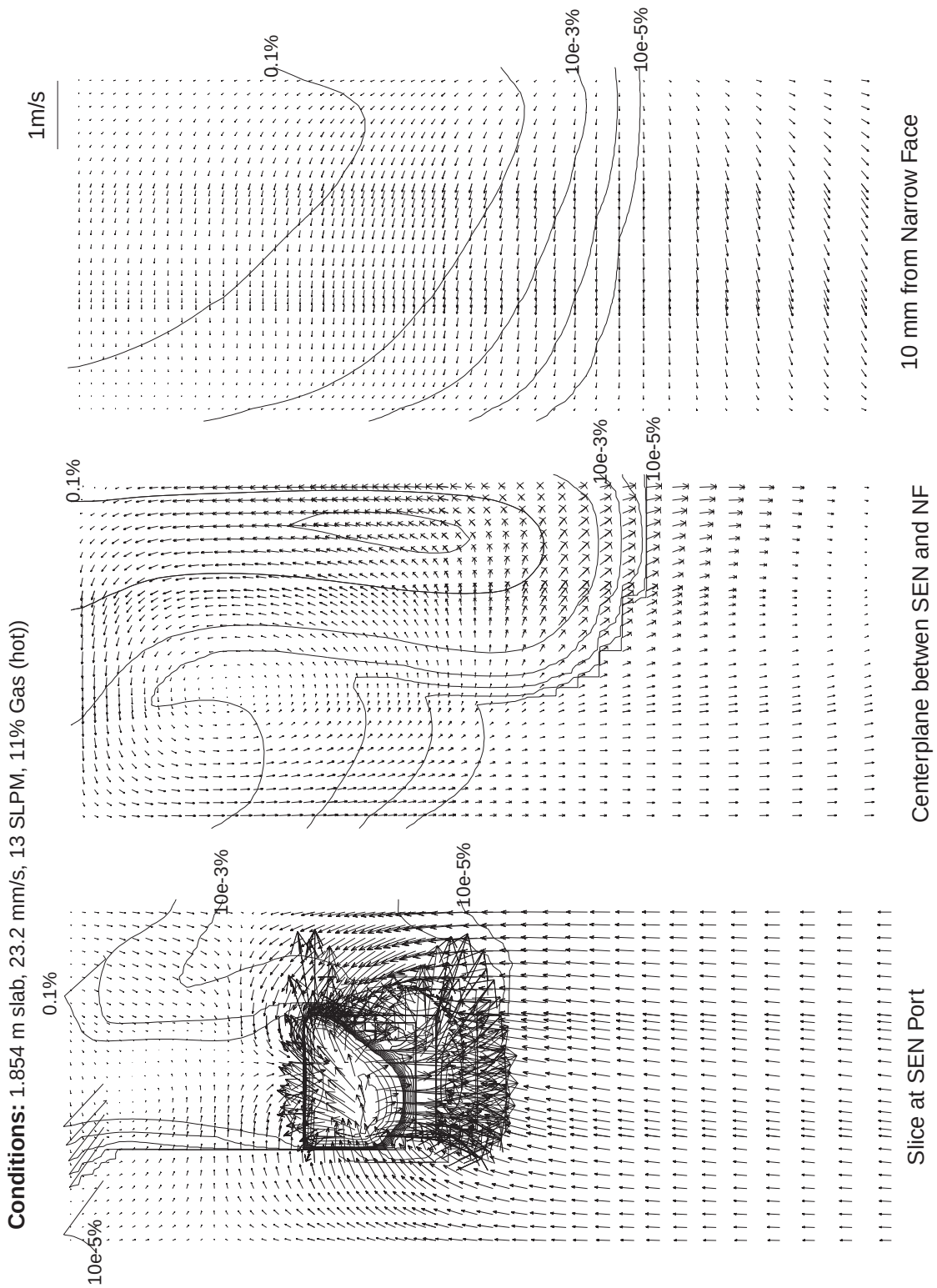


Figure 6.20 Computed steel flow pattern with distributed bubble size for case A (narrow face)

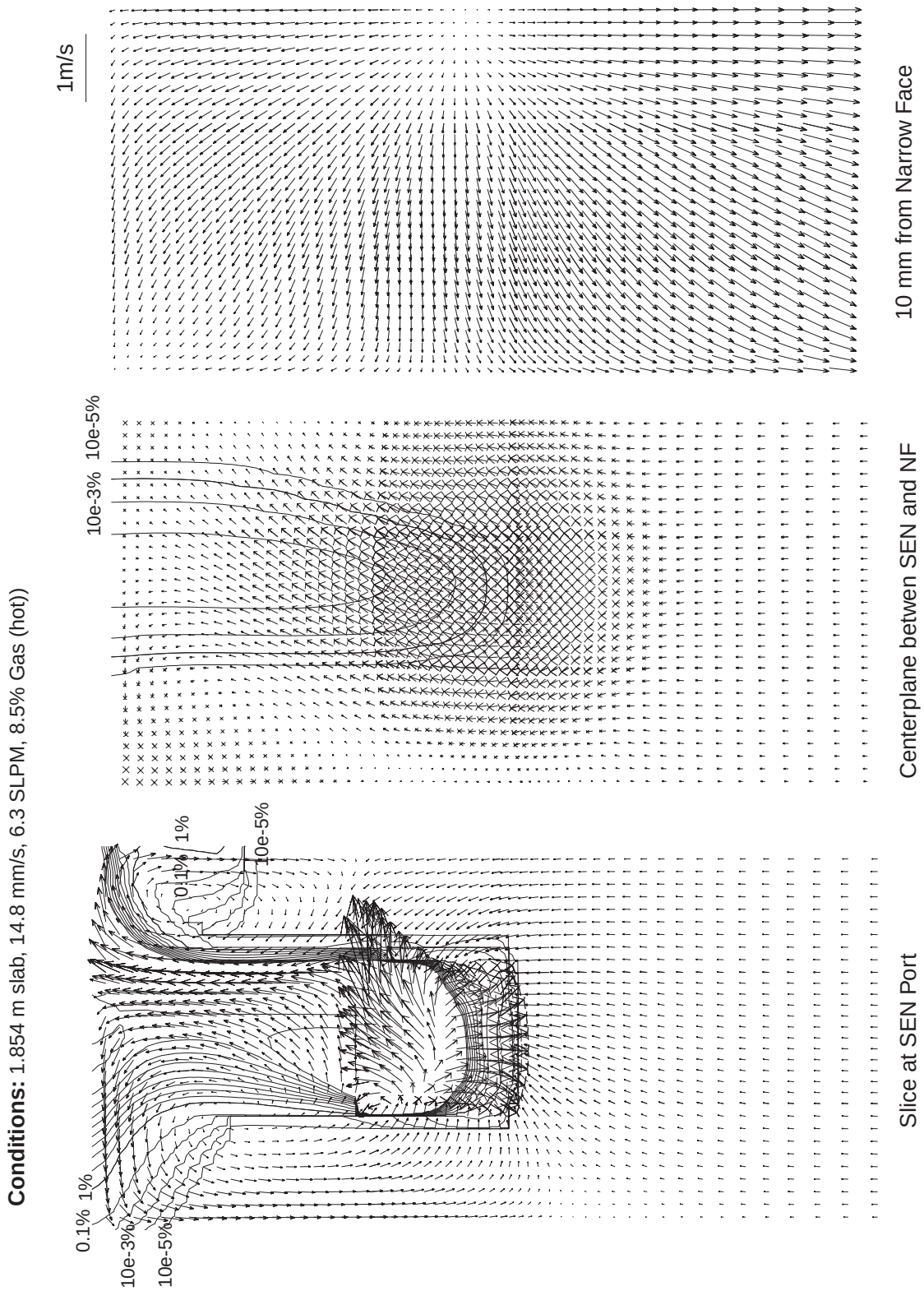


Figure 6.21 Computed steel flow pattern with distributed bubble size for case B (narrow face)

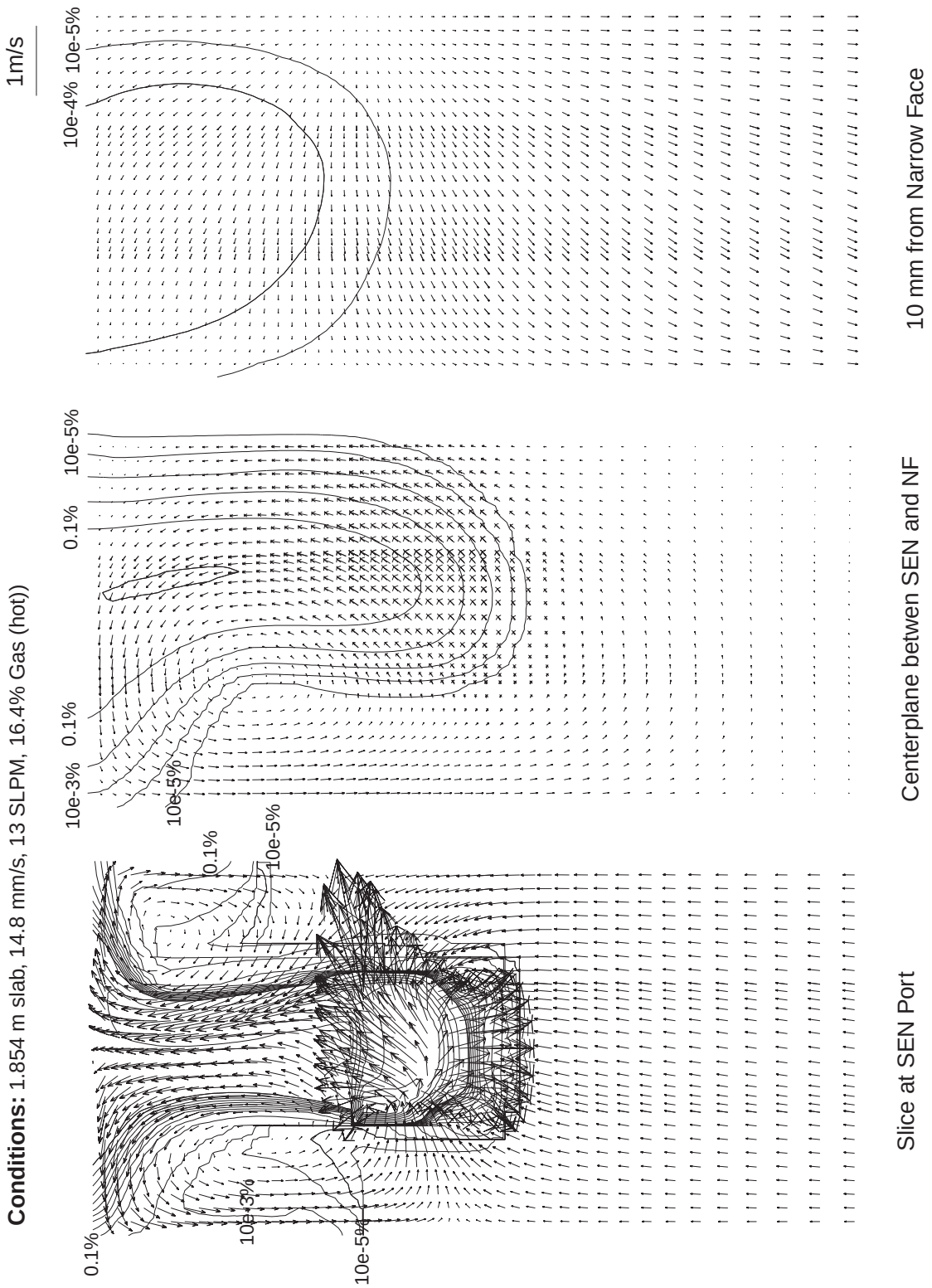


Figure 6.22 Computed steel flow pattern with distributed bubble size for case B with high gas fraction (narrow face, 16.4% gas)

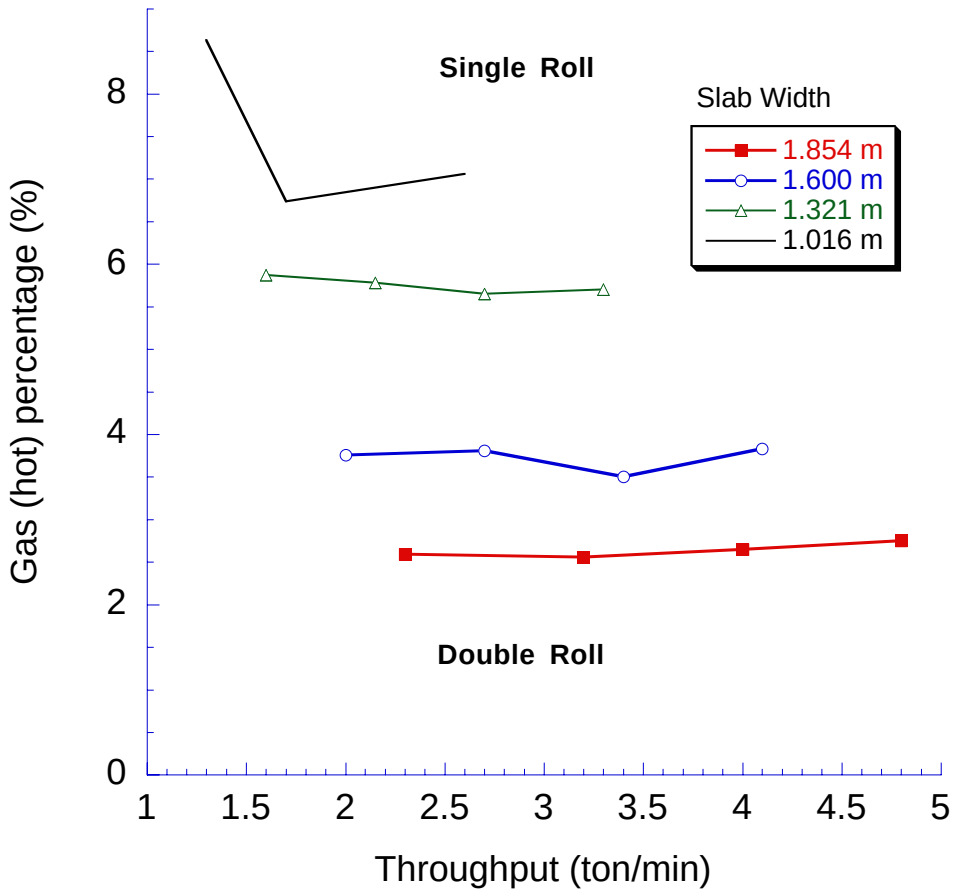


Figure 6.23 Flow pattern identification for 0.4-scale water model (modified from M. Assar, P. Dauby and G. Lawson [26])

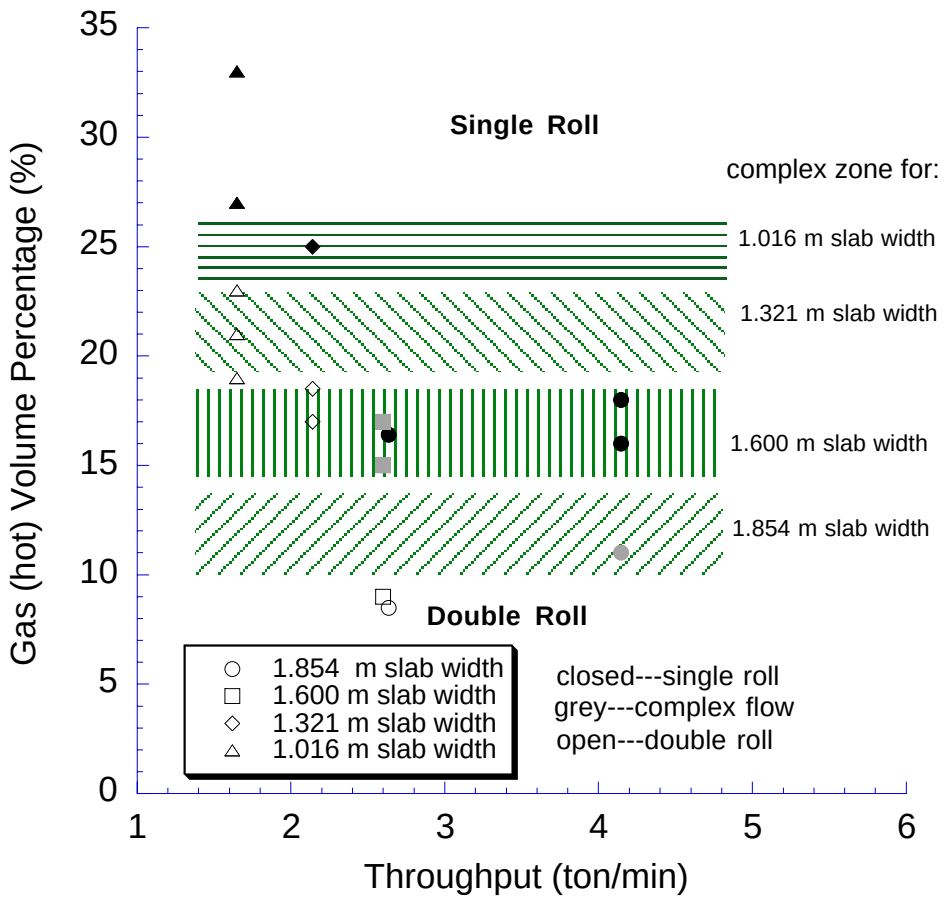


Figure 6.24 Flow pattern identification for steel caster
(from computational simulation of steel caster)

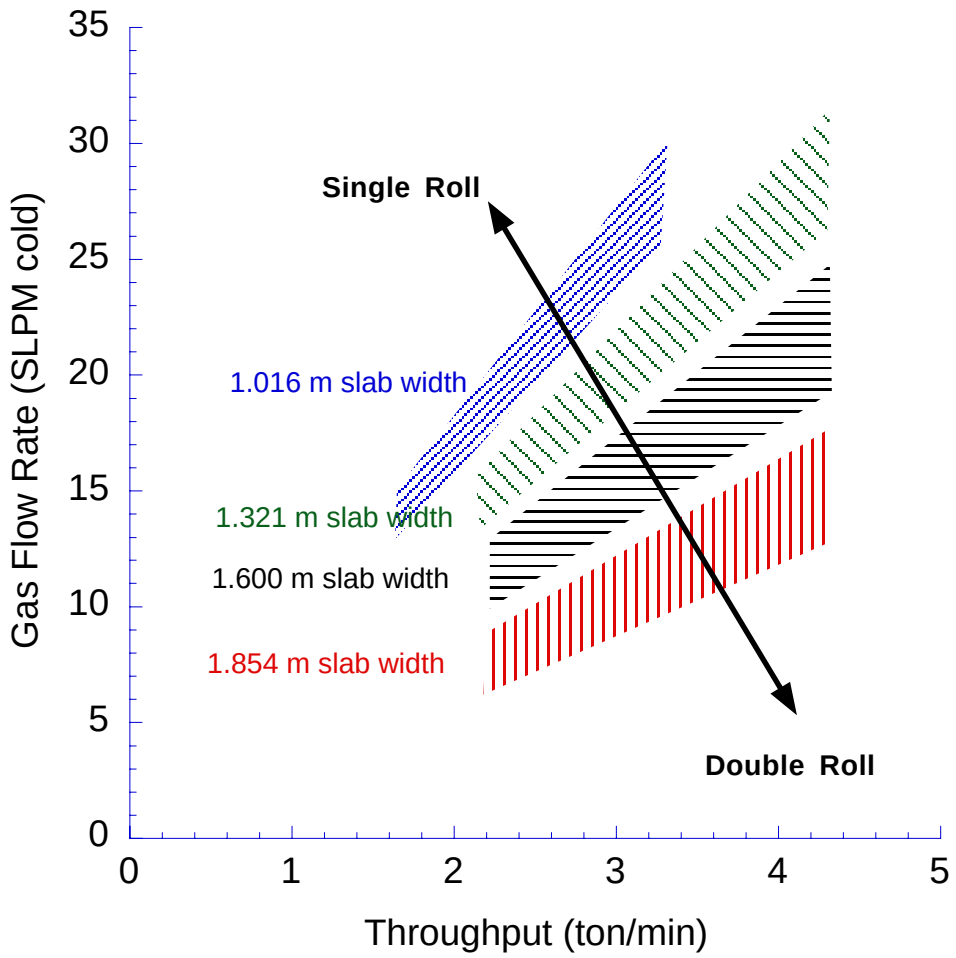


Figure 6.25 Flow pattern identification
(estimated from computational simulations of steel caster)

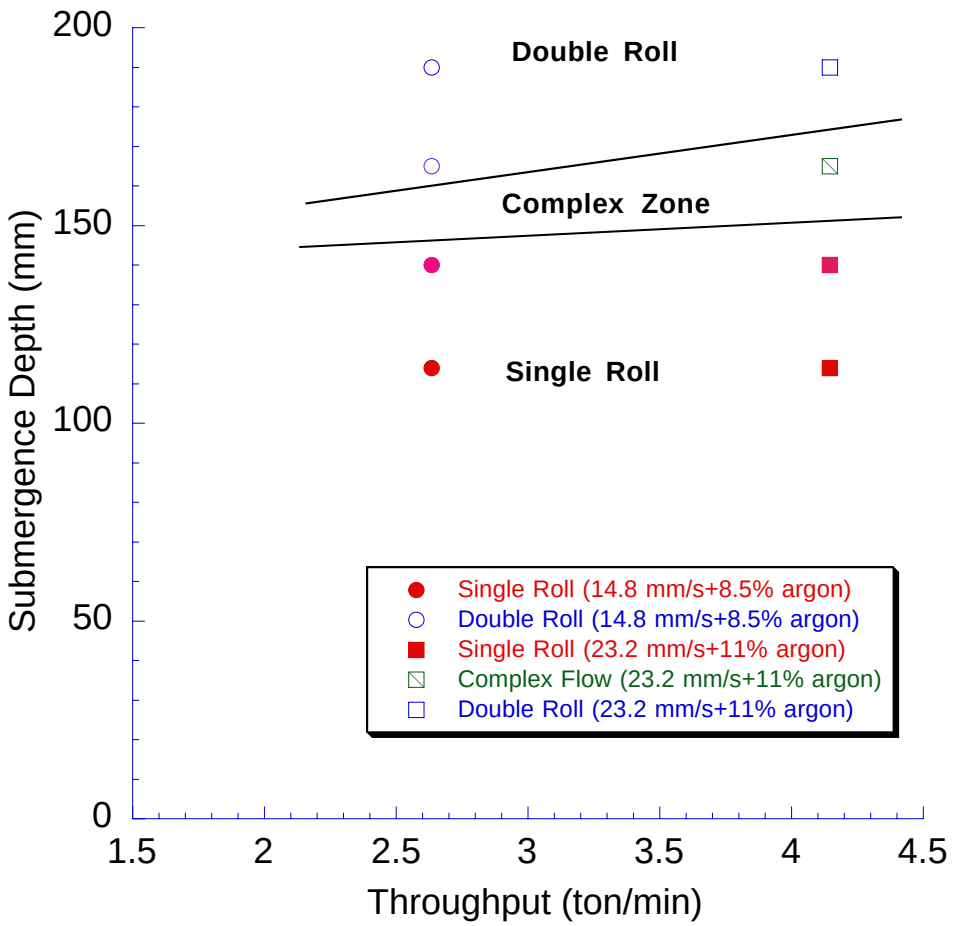


Figure 6.26 Relationship between throughput and submergence depth (1.854 m slab width)

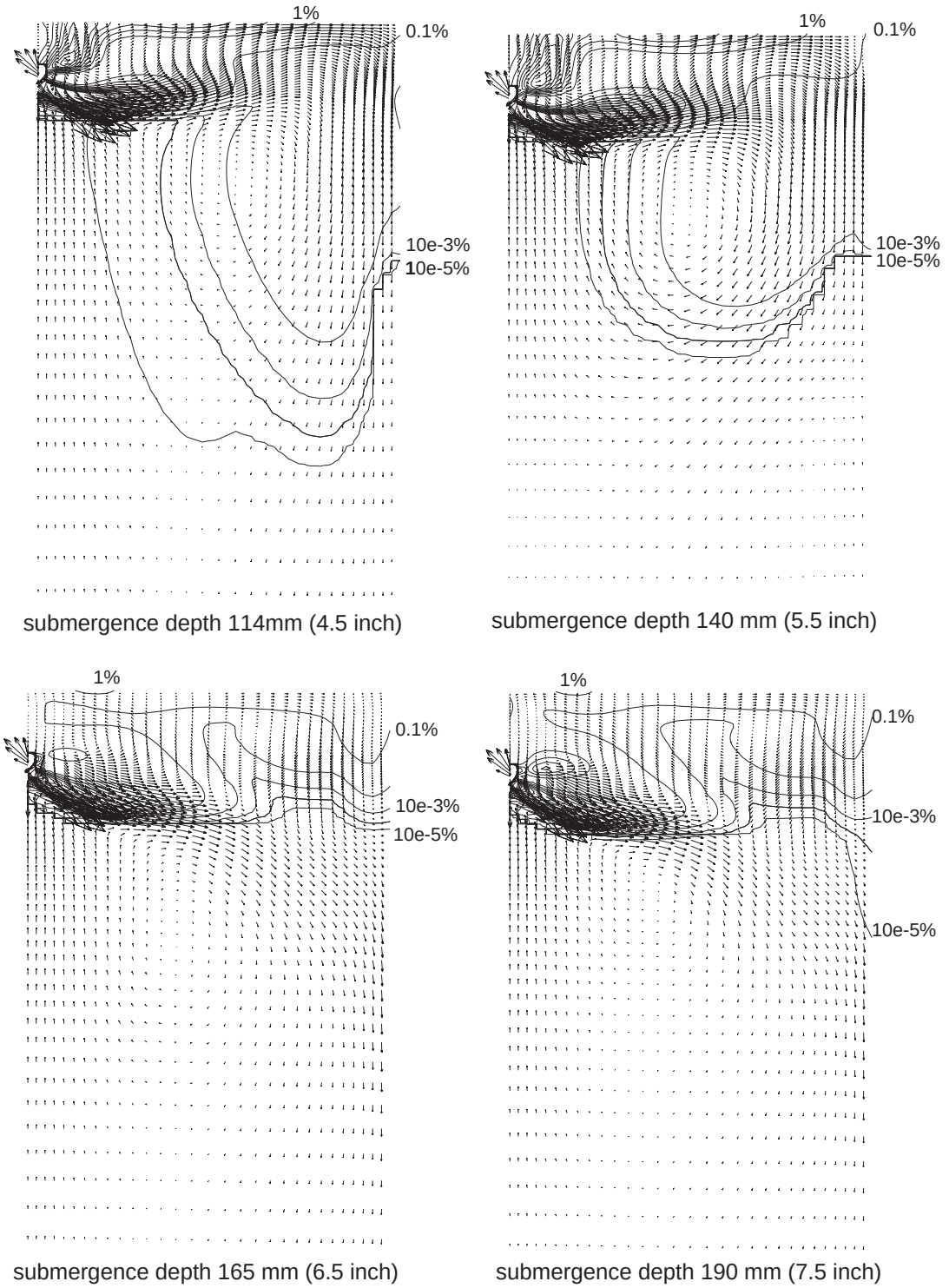


Figure 6.27 Computed velocity at centerplane for different SEN submergence depth (1.854 m slab, 23.2 mm/s castingspeed and 11% hot gas)

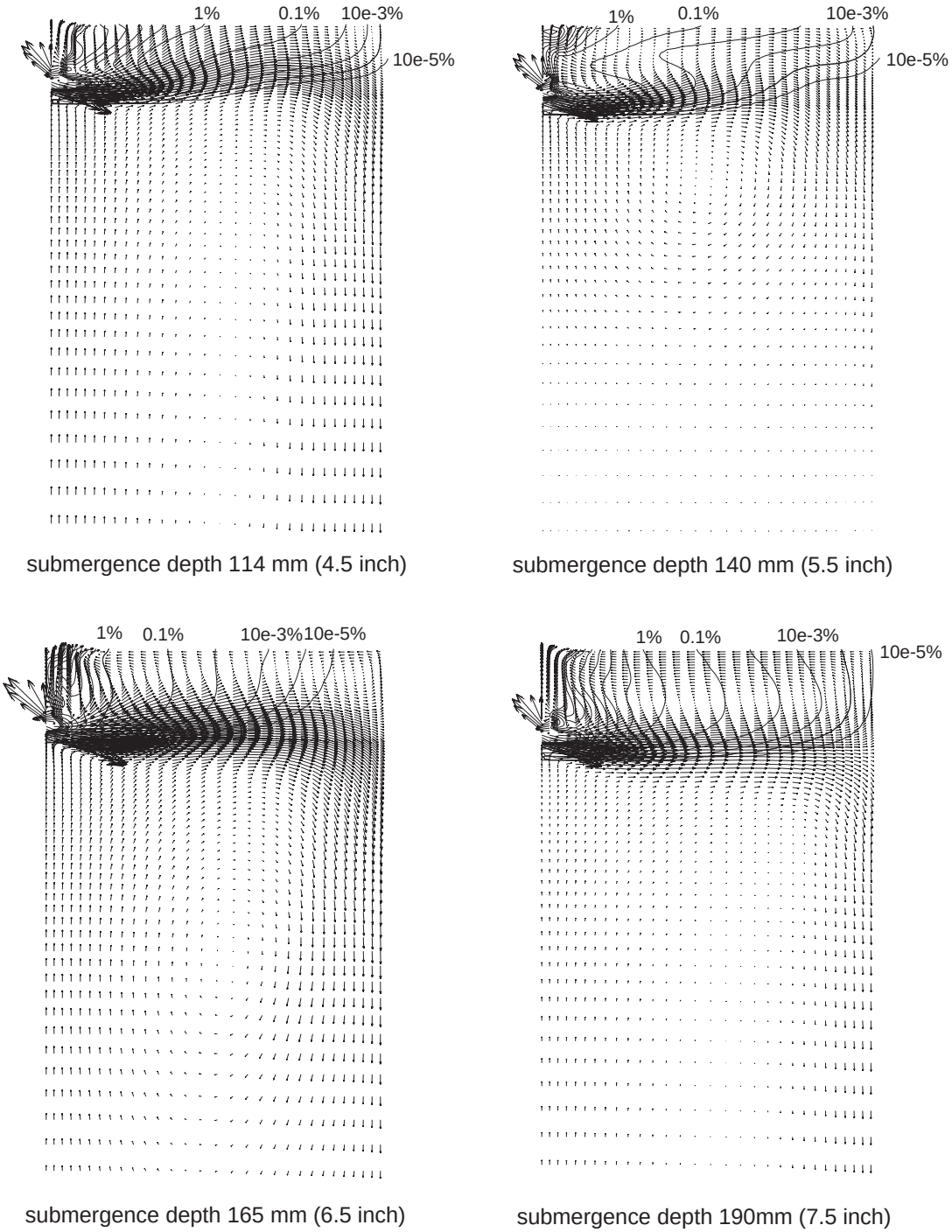


Figure 6.28 Computed velocity at centerplane for different SEN submergence depth (1.854 m slab, 14.8 mm/s castingspeed and 8.5% hot gas)

Section 7

Conclusions

In this study, computational models of turbulent flow in continuous casting of steel have been developed, including single phase flow, single phase heat transfer, multiphase flow with uniform size and multi-size bubbles. Experiments are performed on a 0.4-scale water model to verify the single phase and multiphase turbulent flow models by comparing their prediction with the velocity measurements using PIV (Particle Image Velocity) technology. The 0.4-scale water model measurements and computational results for multiphase with multi-size bubbles are then compared with plant measurements using MFC (Mold Flow Control) sensors to verify the 0.4-scale water model and the multiphase model. The validated models are then applied in a parametric study to investigate the effects of casting conditions on the defects such as pencil pipes and slivers. Plant measurements using thermocouples in the steel caster are performed to verify the single phase heat transfer model. The validated model was applied to predict the heat flux in the steel caster. The heat flux is used in the calculation of the solidification of the shell using CON1D program.

7.1 Validation of Fluid Flow Models in Continuous Casting

The velocities predicted by both the single phase and multiphase K-ε model developed in this study match PIV measurements in the 0.4-scale water model both qualitatively and quantitatively. The comparison between the simulations of 0.4-scale water model, full size water model and steel caster showed that the flows in these cases are similar with each other and the single phase K-ε model developed in this study can

be used to model a full size continuous caster with reasonable accuracy. Comparison between the simulations of the proportionally scaled and Froude scaled water model showed that the difference between these two scaling methods is negligible for single phase flow.

The multiphase flow model with the MUSIG model developed in this study matches the PIV measurements in the 0.4-scale water model. Different simulations using the same model also match plant measurements of the flow pattern using MFC sensors in the steel caster. However, the 0.4-scale water model does not always match flow in the real caster, due to the effects of gas injection, and in some cases produced very different flow patterns. This indicates that the 0.4-scale water model can not reliably reproduce multiphase flow in the steel caster at least when using Froude similarity.

7.2 Validation of Heat Transfer Model

The single phase heat transfer model of temperature in the molten pool developed in this study was verified by plant measurements using thermocouples in the steel caster. The thermocouple measurements show that the current single phase heat transfer model can predict temperature in the upper part of the mold well (with an error less than 7%). The heat transfer model developed in this study can be used to simulate the temperature in the liquid pool of the steel caster and the heat flux through the solidified shell.

7.3 Effects of Gas Injection on Fluid Flow in the Steel Caster

The effects of gas bubble size on the flow pattern in the steel caster are first investigated based on the oversimplified assumption of uniform size bubbles. Then, The

effects of multi-size bubbles on the liquid flow were investigated with the MUSIG model (multiphase flow with multi-size bubbles) developed in this study. Specific findings include:

- Gas changes the flow pattern by applying buoyancy force to the liquid jet. Increasing gas fraction changes the flow from double roll to an intermediate state--complex flow, then to single roll. For a given slab width, there is a critical gas fraction range where the flow pattern is in transition from double roll to single roll. This range does not change too much with steel throughput for a given slab width and submergence depth.
- The critical gas fraction range decreases with smaller bubbles. For a typical case, the jet without gas goes across the mold and impinges on the narrow face forming a classic double roll flow pattern. With 8.5% small uniform gas bubbles (<1.5mm), the jets bend upward and impinge first on the top surface to form a classic single roll flow pattern. With 8.5% large gas bubbles (>1.5mm), the jets split into two branches. One, which is mostly gas, goes up and impinges on the top surface. The other, which is mostly liquid steel, impinges first on the narrow face. The flow with 8.5% large bubbles (>1.5mm) shows a generally double roll flow pattern.
- For the same gas fraction, small bubbles appear to have more tendency to cause a single roll flow pattern. Increasing a uniform bubble size above1.5 mm, flow pattern changes from single roll to double roll. The tendency to form gas defects is also decreased because large bubbles float out of the liquid quickly.

- The Single roll flow pattern such as induced by smaller bubbles induces larger level fluctuations on the top surface and increases the possibility of entrapping flux particles into the liquid flow.
- The downward velocity along the solidified shell at the meniscus reaches its maximum value for the strong single roll flow pattern which accompanied 1.5 mm-diameter uniform bubbles. Bubbles with size between 1.0 mm and 2.0 mm have the largest possibility to cause flux to be trapped in the growing shell at meniscus due to downward movement of liquid steel and flux. The most likely position for inclusion entrapment by this mechanism is near the SEN at wide faces.
- The single roll flow pattern imparts larger downward velocity to the liquid jet than double roll and is likely to carry bubbles deep into the caster to form gas defects such as pencil pipes.
- Compared with the flow with gas, the flow without gas has much less tendency to entrap flux particles and might have fewer inclusion problems in cases when the liquid steel is clean.

7.4 Effects of Slab Width and Nozzle Submergence Depth on the Fluid Flow in the Steel Caster

The effects of slab width and nozzle submergence depth were investigated with the MUSIG model developed in this study. The main findings include:

- For a given casting speed, gas fraction and submergence depth, narrow slabs tend to form double roll while wide slab tends to form single roll. Thus, the critical gas fraction range for this flow transition increases when decreasing slab width.

- Nozzle submergence depth greatly affects the flow pattern in the mold. Deeper nozzle submergence depth has more tendency to form double roll, so thereby increases the critical gas fraction range.

Appendix A. A sample command file for heat transfer model

```
>>CFX4
 >>OPTIONS
  THREE DIMENSIONS
  BODY FITTED GRID
  CARTESIAN COORDINATES
  TURBULENT FLOW
  HEAT TRANSFER
  INCOMPRESSIBLE FLOW
  STEADY STATE
  USER SCALAR EQUATIONS 2
 >>USER FORTRAN
  USRWTM
 >>VARIABLE NAMES
  U VELOCITY 'U VELOCITY'
  V VELOCITY 'V VELOCITY'
  W VELOCITY 'W VELOCITY'
  PRESSURE 'PRESSURE'
  DENSITY 'DENSITY'
  VISCOSITY 'VISCOSITY'
   K 'K'
  EPSILON 'EPSILON'
  TEMPERATURE 'TEMPERATURE'
  ENTHALPY 'ENTHALPY'
  USER SCALAR1 'YPLUS'
  USER SCALAR2 'WALL HEAT FLUX'
>>MODEL DATA
 >>AMBIENT VARIABLES
   K 3.5000E-03
  TEMPERATURE 1.7750E+03
 >>TITLE
  PROBLEM TITLE 'ARMCO SINGLE-PHASE HEAT TRANSFER MODEL'
 >>PHYSICAL PROPERTIES
  >>FLUID PARAMETERS
   VISCOSITY 5.5500E-03
   DENSITY 7.0200E+03
  >>HEAT TRANSFER PARAMETERS
   THERMAL CONDUCTIVITY 2.6000E+01
   FLUID SPECIFIC HEAT 1*  6.800000E+02
   ENTHALPY REFERENCE TEMPERATURE 1.7980E+03
  >>TURBULENCE PARAMETERS
   >>TURBULENCE MODEL
    TURBULENCE MODEL 'K-EPSILON'
>>SOLVER DATA
 >>PROGRAM CONTROL
  MAXIMUM NUMBER OF ITERATIONS 1000
  MASS SOURCE TOLERANCE 1.0000E-06
  ITERATIONS OF TURBULENCE EQUATIONS 1
  ITERATIONS OF VELOCITY AND PRESSURE EQUATIONS 1
  ITERATIONS OF TEMPERATURE AND SCALAR EQUATIONS 1
  ITERATIONS OF HYDRODYNAMIC EQUATIONS 1
  SOLVER DEBUG PRINT STREAM 20
 >>DEFERRED CORRECTION
  K START 1900
  K END 2000
  EPSILON START 1900
  EPSILON END 2000
 >>REDUCTION FACTORS
```

```
    U VELOCITY 3.5000E-01
    V VELOCITY 3.5000E-01
    W VELOCITY 3.5000E-01
   PRESSURE 1.0000E-01
      K 2.5000E-01
   TEMPERATURE 2.5000E-01
  >>UNDER RELAXATION FACTORS
    U VELOCITY 4.5000E-01
    V VELOCITY 4.5000E-01
    W VELOCITY 4.5000E-01
   PRESSURE 1.0000E+00
   VISCOSITY 1.0000E+00
      K 7.000E-01
   EPSILON 7.000E-01
   TEMPERATURE 8.5000E-01
>>MODEL BOUNDARY CONDITIONS
 >>INLET BOUNDARIES
   PATCH NAME 'NOZZLE INLET'
   NORMAL VELOCITY 8.5700E-01
     K 4.2500E-03
   TEMPERATURE 1.8320E+03
 >>PRESSURE BOUNDARIES
   PATCH NAME 'MOLD OUTLET'
   PRESSURE 0.0000E+00
   STATIC PRESSURE SPECIFIED
 >>WALL BOUNDARIES
   PATCH NAME 'MOLD NARROW WALL'
   TEMPERATURE 1.7750E+03
 >>WALL BOUNDARIES
   PATCH NAME 'MOLD WIDE WALL'
   TEMPERATURE 1.7750E+03
 >>WALL BOUNDARIES
   PATCH NAME 'MOLD TOP SURFACE'
   TEMPERATURE ABC 0.0  0.025  300.0
>>OUTPUT OPTIONS
 >>FRONTEND PRINTING
   NO TOPOLOGY STRUCTURE
 >>PRINT OPTIONS
    >>WHAT
     NO VARIABLES
 >>LINE GRAPH DATA
   EACH ITERATION
   FILE NAME 'RESIDUALS'
   ALL VARIABLES
   RESIDUAL
>>STOP
```

Appendix B. A sample command file for multiphase MUSIG model using output of nozzle simulation as input conditions

```
>>CFX4
 >>SET LIMITS
  TOTAL INTEGER WORK SPACE 15000000
  TOTAL CHARACTER WORK SPACE 9000
  TOTAL REAL WORK SPACE 95000000
  MAXIMUM NUMBER OF BLOCKS 20
  MAXIMUM NUMBER OF PATCHES 100
  MAXIMUM NUMBER OF INTER BLOCK BOUNDARIES 40
 >>OPTIONS
  THREE DIMENSIONS
  BODY FITTED GRID
  CARTESIAN COORDINATES
  TURBULENT FLOW
  ISOTHERMAL FLOW
  INCOMPRESSIBLE FLOW
  BUOYANT FLOW
  STEADY STATE
  USER SCALAR EQUATIONS 1
  NUMBER OF PHASES 2
  NUMBER OF MUSIG SIZE GROUPS 11
 >>USER FORTRAN
  USRSRC
  USRTRN
  USRBCS
 >>VARIABLE NAMES
  U VELOCITY 'U VELOCITY'
  V VELOCITY 'V VELOCITY'
  W VELOCITY 'W VELOCITY'
  PRESSURE 'PRESSURE'
  VOLUME FRACTION 'VOLUME FRACTION'
  DENSITY 'DENSITY'
  VISCOSITY 'VISCOSITY'
   K 'K'
  EPSILON 'EPSILON'
  USER SCALAR1 'YPLUS'
>>MODEL DATA
 >>AMBIENT VARIABLES
   K 5.0200E-02
  EPSILON 4.5700E-01
 >>TITLE
  PROBLEM TITLE 'LTV MULTIPHASE MUSIG'
 >>PHYSICAL PROPERTIES
  >>BUOYANCY PARAMETERS
   ALL PHASES
   GRAVITY VECTOR 0.000000E+00  0.000000E+00  -9.800000E+00
  >>FLUID PARAMETERS
   PHASE NAME 'PHASE1'
   VISCOSITY 5.5500E-03
   DENSITY 7.0200E+03
  >>FLUID PARAMETERS
   PHASE NAME 'PHASE2'
   VISCOSITY 7.4200E-05
   DENSITY 5.5900E-01
  >>MULTIPHASE PARAMETERS
   >>PHASE DESCRIPTION
    PHASE NAME 'PHASE1'
```

```
      LIQUID
      CONTINUOUS
     >>PHASE DESCRIPTION
      PHASE NAME 'PHASE2'
      GAS
      DISPERSE
      MEAN DIAMETER 2.5900E-03
     >>MULTIPHASE MODELS
      >>MOMENTUM
        INTER PHASE TRANSFER
        SINCE
     >>INTER PHASE TRANSFER MODELS
      >>MOMENTUM
        FIRST PHASE NAME 'PHASE1'
        SECOND PHASE NAME 'PHASE2'
        SURFACE TENSION COEFFICIENT 1.1920E+00
        >>PARTICLE DRAG MODEL
          FLOW REGIME 'VISCOUS'
     >>MUSIG MODEL
      PHASE NAME 'PHASE2'
      MAXIMUM DIAMETER 1.0500E-02
      MINIMUM DIAMETER 5.0000E-04
      EQUAL DIAMETER DIVISION
      >>BREAKUP MODEL
       LUO SVENDSEN
       BREAKUP COEFFICIENT 4.0000E+00
      >>COALESCENCE MODEL
       PRINCE BLANCH
       COALESCENCE COEFFICIENT 5.0000E-02
        INITIAL FILM THICKNESS 1.0000E-04
       CRITICAL FILM THICKNESS 1.0000E-08
   >>TURBULENCE PARAMETERS
     >>TURBULENCE MODEL
      PHASE NAME 'PHASE1'
      TURBULENCE MODEL 'K-EPSILON'
      PARTICLE INDUCED TURBULENCE 'NONE'
     >>TURBULENCE MODEL
      PHASE NAME 'PHASE2'
      TURBULENCE MODEL 'LAMINAR'
      PARTICLE INDUCED TURBULENCE 'NONE'
>>SOLVER DATA
 >>PROGRAM CONTROL
  MAXIMUM NUMBER OF ITERATIONS 500
  MASS SOURCE TOLERANCE 1.0000E-08
  ITERATIONS OF TURBULENCE EQUATIONS 1
  ITERATIONS OF VELOCITY AND PRESSURE EQUATIONS 1
  ITERATIONS OF TEMPERATURE AND SCALAR EQUATIONS 1
  ITERATIONS OF HYDRODYNAMIC EQUATIONS 1
  SOLVER DEBUG PRINT STREAM 200
 >>DEFERRED CORRECTION
  K START 2000
  K END 2201
  EPSILON START 2000
  EPSILON END 2201
 >>UNDER RELAXATION FACTORS
  PHASE NAME 'PHASE1'
  U VELOCITY 5.5000E-01
  V VELOCITY 5.5000E-01
  W VELOCITY 5.5000E-01
  PRESSURE 9.0000E-01
```

```
  VOLUME FRACTION 9.000E-01
  VISCOSITY 1.0000E+00
    K 7.0000E-01
  EPSILON 6.0000E-01
 >>UNDER RELAXATION FACTORS
  PHASE NAME 'PHASE2'
  U VELOCITY 5.0000E-01
  V VELOCITY 5.0000E-01
  W VELOCITY 5.0000E-01
  PRESSURE 9.0000E-01
  VOLUME FRACTION 9.000E-01
  VISCOSITY 1.0000E+00
    K 5.0000E-01
  EPSILON 5.0000E-01
  ALL MUSIG VOL FRACTIONS 8.0000E-01
>>MODEL BOUNDARY CONDITIONS
 >>INLET BOUNDARIES
  PHASE NAME 'PHASE1'
  PATCH NAME 'NOZZLE INLET'
  U VELOCITY 1.41270E-01
  V VELOCITY 0.0000E+00
  W VELOCITY -4.05100E-01
  VOLUME FRACTION 8.900E-01
    K 5.0200E-02
  EPSILON 4.5700E-01
 >>INLET BOUNDARIES
  PHASE NAME 'PHASE2'
  PATCH NAME 'NOZZLE INLET'
  U VELOCITY 1.41270E-01
  V VELOCITY 0.0000E+00
  W VELOCITY -4.05100E-01
  VOLUME FRACTION 1.1000E-01
    K 1.0000E-04
  EPSILON 1.0000E-04
  MUSIG VOL FRACTION1 0.0107
  MUSIG VOL FRACTION2 0.0453
  MUSIG VOL FRACTION3 0.3115
  MUSIG VOL FRACTION4 0.5583
  MUSIG VOL FRACTION5 0.0742
  MUSIG VOL FRACTION6 0.0
  MUSIG VOL FRACTION7 0.0000
  MUSIG VOL FRACTION8 0.0000
  MUSIG VOL FRACTION9 0.0000
  MUSIG VOL FRACTION10 0.0
  MUSIG VOL FRACTION11 0.0
 >>PRESSURE BOUNDARIES
  PHASE NAME 'PHASE1'
  PATCH NAME 'OUTLET'
  PRESSURE 0.0000E+00
  VOLUME FRACTION 1.0000E+00
 >>PRESSURE BOUNDARIES
  PHASE NAME 'PHASE2'
  PATCH NAME 'OUTLET'
  PRESSURE 0.0000E+00
>>WALL BOUNDARIES
  PHASE NAME 'PHASE1'
  PATCH NAME 'TOP SURFACE'
  U VELOCITY 0.0000E+00
  V VELOCITY 0.0000E+00
  W VELOCITY 0.0000E+00
```

```
 >>WALL BOUNDARIES
   ALL PHASES
   PATCH NAME 'WIDE FACE'
   U VELOCITY 0.0000E+00
   V VELOCITY 0.0000E+00
   W VELOCITY 0.0000E+00
 >>WALL BOUNDARIES
   ALL PHASES
   PATCH NAME 'NARROW FACE'
   U VELOCITY 0.0000E+00
   V VELOCITY 0.0000E+00
   W VELOCITY 0.0000E+00
>>OUTPUT OPTIONS
 >>FRONTEND PRINTING
   NO TOPOLOGY STRUCTURE
 >>PRINT OPTIONS
    >>WHAT
     NO VARIABLES
 >>LINE GRAPH DATA
   EACH ITERATION
   FILE NAME 'RESIDUALS'
   ALL VARIABLES
   RESIDUAL
>>STOP
```

Appendix C. Subroutine for heat transfer model

```
SUBROUTINE USRWTM(CNAME,IEQN,IPHASE,CTROP,CWALL
     +              ,GAMMA,YWALL,TMULT,ISTART,NCV
     +              ,DIFLAM,XYPLUS,ELOG,PRT
     +              ,U,V,W,P,VFRAC,DEN,VIS,TE,ED,RS,T,H,RF,SCAL
     +              ,XP,YP,ZP,VOL,AREA,VPOR,ARPOR,WFACT,IPT
     +              ,IBLK,IPVERT,IPNODN,IPFACN,IPNODF,IPNODB,IPFACB
     +              ,WORK,IWORK,CWORK)
C
C*************************************************************
C
C  USER SUBROUTINE TO OVERWRITE DEFAULT WALL MULTIPLIERS (TMULT)
C   FLUX = TMULT * (VARIABLE AT WALL - VARIABLE AT NEAREST NODE)
C
C*************************************************************
C
C*************************************************************
C
C*************************************************************
C
     LOGICAL LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
     +       ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
C
     CHARACTER*(*) CWALL,CWORK
     CHARACTER     CNAME*6,CTROP*6
C
C+++++++++++++++++  USER AREA 1 +++++++++++++++++++++++++++++++++++
C---- AREA FOR USERS EXPLICITLY DECLARED VARIABLES
C
C+++++++++++++++++  END OF USER AREA 1  ++++++++++++++++++++++++++++
C
     COMMON
      + /ALL/    NBLOCK,NCELL,NBDRY,NNODE,NFACE,NVERT,NDIM
      + /ADDIMS/ NPHASE,NSCAL,NVAR,NPROP
      +          ,NDVAR,NDPROP,NDXNN,NDGEOM,NDCOEF,NILIST,NRLIST,NTOPOL
      + /CHKUSR/ IVERS,IUCALL,IUSED
      + /DEVICE/ NREAD,NWRITE,NRDISK,NWDISK
      + /IDUM/   ILEN,JLEN
      + /LOGIC/  LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
      +          ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
      + /MLTGRD/ MLEVEL,NLEVEL,ILEVEL
      + /SGLDBL/ IFLGPR,ICHKPR
      + /SPARM/  SMALL,SORMAX,NITER,INDPRI,MAXIT,NODREF,NODMON
      + /TRANSI/ NSTEP,KSTEP,MF,INCORE
      + /TRANSR/ TIME,DT,DTINVF,TPARM
      + /TURBKE/ CMU,C1,C2,C3,CAPPA,CMU14,CMU34
C
C+++++++++++++++++  USER AREA 2 +++++++++++++++++++++++++++++++++++
C---- AREA FOR USERS TO DECLARE THEIR OWN COMMON BLOCKS
C     THESE SHOULD START WITH THE CHARACTERS 'UC' TO ENSURE
C     NO CONFLICT WITH NON-USER COMMON BLOCKS
C
C+++++++++++++++++  END OF USER AREA 2  ++++++++++++++++++++++++++++
C
     DIMENSION CWALL(NPHASE),GAMMA(NCELL,NPHASE,*)
    +,YWALL(NCV),TMULT(NCV,NPHASE)
    +,DIFLAM(NVAR,NPHASE),XYPLUS(NVAR,NPHASE),ELOG(NVAR,NPHASE)
    +,PRT(NVAR,NPHASE)
```

```
      DIMENSION
     + U(NNODE,NPHASE),V(NNODE,NPHASE),W(NNODE,NPHASE),P(NNODE,NPHASE)
     +,VFRAC(NNODE,NPHASE),DEN(NNODE,NPHASE),VIS(NNODE,NPHASE)
     +,TE(NNODE,NPHASE),ED(NNODE,NPHASE),RS(NNODE,NPHASE,6)
     +,T(NNODE,NPHASE),H(NNODE,NPHASE),RF(NNODE,NPHASE,4)
     +,SCAL(NNODE,NPHASE,NSCAL)
      DIMENSION
     + XP(NNODE),YP(NNODE),ZP(NNODE)
     +,VOL(NCELL),AREA(NFACE,3),VPOR(NCELL),ARPOR(NFACE,3),WFACT(NFACE)
      +,IPT(*),IBLK(5,NBLOCK)
     +,IPVERT(NCELL,8),IPNODN(NCELL,6),IPFACN(NCELL,6),IPNODF(NFACE,4)
     +,IPNODB(NBDRY,4),IPFACB(NBDRY)
      +,IWORK(*),WORK(*),CWORK(*)
C
C+++++++++++++++++  USER AREA 3 +++++++++++++++++++++++++++++++++++
C---- AREA FOR USERS TO DIMENSION THEIR ARRAYS
C
C---- AREA FOR USERS TO DEFINE DATA STATEMENTS
C
C+++++++++++++++++  END OF USER AREA 3  ++++++++++++++++++++++++++
C
C---- STATEMENT FUNCTION FOR ADDRESSING
      IP(I,J,K)=IPT((K-1)*ILEN*JLEN+(J-1)*ILEN+I)
C
C----VERSION NUMBER OF USER ROUTINE AND PRECISION FLAG
C
      IVERS = 1
      ICHKPR = 1
C
C+++++++++++++++++  USER AREA 4 +++++++++++++++++++++++++++++++++++
C---- TO USE THIS USER ROUTINE FIRST SET IUSED=1
C
      IUSED = 1
C
C+++++++++++++++++  END OF USER AREA 4  ++++++++++++++++++++++++
C
      IF (IUSED.EQ.0) RETURN
C
C---- FRONTEND CHECKING OF USER ROUTINE
      IF (IUCALL.EQ.0) RETURN
C
C+++++++++++++++++  USER AREA 5 +++++++++++++++++++++++++++++++++++
C
C-----EXAMPLE - TO USE A MOLECULAR VISCOSITY DEPENDENT ON TEMPERATURE
C     IN A SMOOTH WALL CALCULATION.
C     THE EXAMPLE HAS  MOLECULAR VISCOSITY = ATEMP*TEMPERATURE + BTEMP.
C     THE USER WOULD IN GENERAL SUPPLY A FUNCTIONAL RELATIONSHIP FOR
C     MOLECULAR VISCOSITY IN TERMS OF OTHER VARIABLES.
      CALL IPALL('MOLD NARROW WALL','*','PATCH','CENTRES',IPT,NPT,
     +          CWORK,IWORK)
C
C  FIND VARIABLE NUMBER FOR VELOCITY
       MAX_NF=0.0
       WALLHF_NF=0.0
       DO 101  I=1,NPT
      INODE=IPT(I)
      IBDRY=INODE-NCELL
      IFACE=IPFACB(IBDRY)
      INODE2=IPNODF(IFACE,1)
      AREAM=SQRT(AREA(IFACE,1)*AREA(IFACE,1)
```

```
     +          +AREA(IFACE,2)*AREA(IFACE,2)
     +          +AREA(IFACE,3)*AREA(IFACE,3))
      IF (ABS(SCAL(INODE,1,2)).GT.MAX_NF)  THEN
         MAX_NF=ABS(SCAL(INODE,1,2))
      END IF
      WALLHF_NF=WALLHF_NF+SCAL(INODE,1,2)*AREAM
 101  CONTINUE

      CALL IPALL('MOLD WIDE WALL','*','PATCH','CENTRES',IPT,NPT,
     +          CWORK,IWORK)

C  FIND VARIABLE NUMBER FOR VELOCITY
       MAX_WF=0.0
       WALLHF_WF=0.0
      DO 102  J=1,NPT
      INODE=IPT(J)
      IBDRY=INODE-NCELL
      IFACE=IPFACB(IBDRY)
      AREAM=SQRT(AREA(IFACE,1)*AREA(IFACE,1)
     +          +AREA(IFACE,2)*AREA(IFACE,2)
     +          +AREA(IFACE,3)*AREA(IFACE,3))
      IF (ABS(SCAL(INODE,1,2)).GT.MAX_WF)  THEN
         MAX_WF=ABS(SCAL(INODE,1,2))
      END IF
      WALLHF_WF=WALLHF_WF+SCAL(INODE,1,2)*AREAM
 102  CONTINUE

C
       CALL IPALL('MOLD TOP SURFACE','*','PATCH','CENTRES',IPT,NPT,
     +          CWORK,IWORK)
C
C  FIND VARIABLE NUMBER FOR VELOCITY
       AREA_TOP=0.0
       WALLHF_TOP=0.0
      DO 103  I=1,NPT
      INODE=IPT(I)
      IBDRY=INODE-NCELL
      IFACE=IPFACB(IBDRY)
      INODE2=IPNODF(IFACE,1)
      AREAM=SQRT(AREA(IFACE,1)*AREA(IFACE,1)
     +          +AREA(IFACE,2)*AREA(IFACE,2)
     +          +AREA(IFACE,3)*AREA(IFACE,3))
      AREA_TOP=AREA_TOP+AREAM
      WALLHF_TOP=WALLHF_TOP-SCAL(INODE,1,2)*AREAM
 103  CONTINUE
       TOP_HF=-12000*AREA_TOP
C
       CALL IPALL('NOZZLE INLET','*','PATCH','CENTRES',IPT,NPT,
     +          CWORK,IWORK)
C
C  FIND VARIABLE NUMBER FOR VELOCITY
       SUM=0.0
       HF_IN=0.0
       TMASS_IN=0.0
      DO 104  I=1,NPT
      INODE=IPT(I)
      IBDRY=INODE-NCELL
      IFACE=IPFACB(IBDRY)
      AREAM=SQRT(AREA(IFACE,1)*AREA(IFACE,1)
     +          +AREA(IFACE,2)*AREA(IFACE,2)
```

```
     +          +AREA(IFACE,3)*AREA(IFACE,3))
      TMASS_IN=TMASS_IN+7020*W(INODE,1)*AREAM
       HF_IN=HF_IN+680*7020*W(INODE,1)*AREAM*(T(INODE,1)-1775)
  104  CONTINUE
C
      CALL IPALL('MOLD OUTLET','*','PATCH','CENTRES',IPT,NPT,
     +          CWORK,IWORK)
C
C  FIND VARIABLE NUMBER FOR VELOCITY
      HF_OUT=0.0
      AREA_OUT=0.0
      TEMP3=0.0
      TMASS_OUT=0.0
      DO 105  I=1,NPT
      INODE=IPT(I)
      IBDRY=INODE-NCELL
      IFACE=IPFACB(IBDRY)
      ISURF1=IPNODB(IBDRY,1)
      ISURF2=IPNODB(IBDRY,2)
      INODE2=IPNODF(IFACE,1)
      AREAM=SQRT(AREA(IFACE,1)*AREA(IFACE,1)
     +          +AREA(IFACE,2)*AREA(IFACE,2)
     +          +AREA(IFACE,3)*AREA(IFACE,3))
      TMASS_OUT=TMASS_OUT+7020*W(ISURF1,1)*AREAM
       HF_OUT=HF_OUT-680*7020*W(ISURF1,1)*AREAM*(T(ISURF1,1)-1775)
C      TEMP3=TEMP3+T(INODE,1)*AREAM
      AREA_OUT=AREA_OUT+AREAM
  105  CONTINUE
C
C      AVRG_TEMP=TEMP3/AREA_OUT
C        SUM2=VOLUME*(AVRG_TEMP-298)*680*7020
C
C
      OUT_HF=HF_OUT+WALLHF_NF+WALLHF_WF+TOP_HF
      HEAT_BALANCE=HF_IN+OUT_HF
      ERROR=100*HEAT_BALANCE/HF_IN
      TMASS_DIFF=ABS(TMASS_OUT-TMASS_IN)
      TMASS_ERR=TMASS_DIFF/ABS(TMASS_IN)
C
      WRITE(NWRITE,*) '   '
      WRITE(NWRITE,*) '****   INPUT RATE OF HEAT AND  MASS****'
      WRITE(NWRITE,*) 'INPUT HEAT RATE FROM INLET=',HF_IN
      WRITE(NWRITE,*) 'INPUT MASS RATE FROM INLET=',ABS(TMASS_IN)
      WRITE(NWRITE,*) '   '
      WRITE(NWRITE,*) '****   OUTGOING RATE OF HEAT AND MASS****'
      WRITE(NWRITE,*) 'HEAT RATE ON NARROW FACE=',WALLHF_NF
      WRITE(NWRITE,*) 'HEAT RATE ON WIDE FACE=',WALLHF_WF
      WRITE(NWRITE,*) 'HEAT RATE ON TOP SURFACE=',WALLHF_TOP
      WRITE(NWRITE,*) 'OUTGOING HEAT RATE FROM OUTLET=',HF_OUT

      WRITE(NWRITE,*) 'TOTAL OUTGOING HEAT RATE=',OUT_HF
      WRITE(NWRITE,*) 'TOTAL OUTGOING MASS RATE=',ABS(TMASS_OUT)
      WRITE(NWRITE,*) '   '
      WRITE(NWRITE,*) '****HEAT  BALANCE****'

      WRITE(NWRITE,*) 'HEAT BALANCE=',HEAT_BALANCE,"W"
      WRITE(NWRITE,*) 'ERROR of HEAT BALANCE=',ERROR,"%"
      WRITE(NWRITE,*) 'MASS BALANCE=',TMASS_DIFF,"KG"
      WRITE(NWRITE,*) 'ERROR of MASS BALANCE=',TMASS_ERR,"%"
      WRITE(NWRITE,*) '   '
```

```
      WRITE(NWRITE,*) '****MONITORING TERMS****'
      WRITE(NWRITE,*) 'HEAT RATE ON TOP SURFACE(1)=',WALLHF_TOP
      WRITE(NWRITE,*) 'HEAT RATE ON TOP SURFACE(2)=',TOP_HF
      WRITE(NWRITE,*) 'TOTAL AREA of OUTLET=',AREA_OUT
      WRITE(NWRITE,*) 'TOTAL NUMBER OF CELLS=',NCELL
      WRITE(NWRITE,*) 'MAX. WALL HEAT FLUX on NF=',MAX_NF
      WRITE(NWRITE,*) 'MAX. WALL HEAT FLUX on WF=',MAX_WF
C
      CALL IPALL('*','WALL','PATCH','CENTRES',IPT,NPT,CWORK,IWORK)
C
C  FIND VARIABLE NUMBER FOR ENTHALPY
      CALL GETVAR('USRWTM','H      ',IVAR)
C  IF ENTHALPY EQUATION SET MULTIPLIER
     IF (IVAR.EQ.IEQN) THEN
        PRANDT = PRT(IVAR,1)
      DO 120 I = 1, NPT
        INODE = IPT(I)
        IBDRY = INODE - NCELL
         LCV   = IBDRY - ISTART + 1
        INODE1 = IPNODB(IBDRY,1)
         CMU = 0.09
        DENS = DEN(INODE1,1)
        DENSQK = DEN(INODE1,1) * SQRT( TE(INODE1,1) )
        EPSILON = ED(INODE,1)
         AKE = TE(INODE1,1)
         AKK = AKE*AKE
C  NORMAL DISTANCE FROM NODE TO WALL
C
         DN = YWALL(LCV)
C         WRITE(NWRITE,*) YWALL(LCV)
C
C  COMPUTE MULTIPLIER
         TMULT(LCV,1) = (CMU*DENS*AKK)/(PRANDT*DN*EPSILON)
  120 CONTINUE
C
     END IF
C
C+++++++++++++++++  END OF USER AREA 5  ++++++++++++++++++++++++++++
C
     RETURN
      END
```

Appendix D. Subroutine for multiphase flow with MUSIG model

```
SUBROUTINE USRSRC(IEQN,ICALL,CNAME,CALIAS,AM,SP,SU,CONV
     +                ,U,V,W,P,VFRAC,DEN,VIS,TE,ED,RS,T,H,RF,SCAL
     +                ,XP,YP,ZP,VOL,AREA,VPOR,ARPOR,WFACT,IPT
     +                ,IBLK,IPVERT,IPNODN,IPFACN,IPNODF,IPNODB,IPFACB
     +                ,WORK,IWORK,CWORK)
C
     LOGICAL LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
     +       ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
C
     CHARACTER*(*) CWORK
     CHARACTER     CNAME*6,  CALIAS*24
C
C
     COMMON
      + /ALL/    NBLOCK,NCELL,NBDRY,NNODE,NFACE,NVERT,NDIM
      + /ALLWRK/ NRWS,NIWS,NCWS,IWRFRE,IWIFRE,IWCFRE
      + /ADDIMS/ NPHASE,NSCAL,NVAR,NPROP
      +         ,NDVAR,NDPROP,NDXNN,NDGEOM,NDCOEF,NILIST,NRLIST,NTOPOL
      + /CHKUSR/ IVERS,IUCALL,IUSED
      + /DEVICE/ NREAD,NWRITE,NRDISK,NWDISK
      + /IDUM/    ILEN,JLEN
      + /LOGIC/  LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
      +         ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
      + /MLTGRD/ MLEVEL,NLEVEL,ILEVEL
      + /SGLDBL/ IFLGPR,ICHKPR
      + /SPARM/  SMALL,SORMAX,NITER,INDPRI,MAXIT,NODREF,NODMON
      + /TRANSI/ NSTEP,KSTEP,MF,INCORE
      + /TRANSR/ TIME,DT,DTINVF,TPARM
C
C+++++++++++++++++  USER AREA 2  +++++++++++++++++++++++++++++++++++
C
     COMMON /UCSURF/ JISURF,NISURF,JDSURF,NDSURF
     COMMON /UCINJR/ FINJ1,FINJ2,UGASIN,VGASIN,WGASIN
     COMMON /UCFLUX/ GASIN,GASOUT
C
C+++++++++++++++++  END OF USER AREA 2  ++++++++++++++++++++++++++
C
     DIMENSION AM(NCELL,6,NPHASE),SP(NCELL,NPHASE),SU(NCELL,NPHASE)
    +,CONV(NFACE,NPHASE)
C
     DIMENSION
    + U(NNODE,NPHASE),V(NNODE,NPHASE),W(NNODE,NPHASE),P(NNODE,NPHASE)
    +,VFRAC(NNODE,NPHASE),DEN(NNODE,NPHASE),VIS(NNODE,NPHASE)
    +,TE(NNODE,NPHASE),ED(NNODE,NPHASE),RS(NNODE,NPHASE,6)
    +,T(NNODE,NPHASE),H(NNODE,NPHASE),RF(NNODE,NPHASE,4)
    +,SCAL(NNODE,NPHASE,NSCAL)
C
     DIMENSION
    + XP(NNODE),YP(NNODE),ZP(NNODE)
    +,VOL(NCELL),AREA(NFACE,3),VPOR(NCELL),ARPOR(NFACE,3)
    +,WFACT(NFACE)
     +,IPT(*),IBLK(5,NBLOCK)
    +,IPVERT(NCELL,8),IPNODN(NCELL,6),IPFACN(NCELL,6),IPNODF(NFACE,4)
    +,IPNODB(NBDRY,4),IPFACB(NBDRY)
     +,IWORK(*),WORK(*),CWORK(*)
C
C---- STATEMENT FUNCTION FOR ADRESSING
```

```
      IP(I,J,K)=IPT((K-1)*ILEN*JLEN+(J-1)*ILEN+I)
C
C----VERSION NUMBER OF USER ROUTINE AND PRECISION FLAG
C
     IVERS=4
     ICHKPR = 1
C
C
      IUSED=1
      IF (IUSED.EQ.0) RETURN
C
C---- FRONTEND CHECKING OF USER ROUTINE
      IF (IUCALL.EQ.0) RETURN
C
C---- ADD TO SOURCE TERMS
      IF (ICALL.EQ.1) THEN
C
C+++++++++++++++++  USER AREA 5  +++++++++++++++++++++++++++++++
C
C
C-----------------------------------------------------------------
C  VOLUME FRACTION [M]/[T]
C
     IF(CALIAS.EQ.'VOLUME FRACTION') THEN
C
C.......SINK AT TOP
        CALL USRDEGAS(CALIAS,SP,W,VFRAC,DEN,AREA,IWORK(JDSURF),NDSURF
     +            ,IPNODB,IPFACB,TOTAL)
       WRITE(NWRITE,*) ' VOLUME GASOUT=:',TOTAL
C       WRITE(NWRITE,*) ' VF :',TOTAL
C
     ENDIF
C
C-----------------------------------------------------------------
C PRESSURE [M]/[T]
C
     IF(CALIAS.EQ.'PRESSURE') THEN
C
       GASIN =  0.0
C
C.......SINK AT TOP
        CALL USRDEGAS(CALIAS,SU,W,VFRAC,DEN,AREA,IWORK(JDSURF),NDSURF
     +            ,IPNODB,IPFACB,TOTAL)
       GASOUT = TOTAL
       WRITE(NWRITE,*) ' PRESSURE GASOUT=:',TOTAL
C
     ENDIF
C
C-----------------------------------------------------------------
C  U VELOCITY [M][L]/[T]^2
C
     IF(CALIAS.EQ.'U VELOCITY') THEN
C
C
C.......SINK AT TOP
        CALL USRDEGAS(CALIAS,SP,W,VFRAC,DEN,AREA,IWORK(JDSURF),NDSURF
     +            ,IPNODB,IPFACB,TOTAL)
C       WRITE(NWRITE,*) ' U :',TOTAL
C       WRITE(NWRITE,*) ' U GASOUT=:',TOTAL
C
```

```
C.......SET MINIMUM VOID FRACTION
        VFMIN=1.0E-8
        DO 101  INODE=1,NNODE
         VFRAC(INODE,2)=MAX(VFRAC(INODE,2),VFMIN)
 101    CONTINUE
C
     ENDIF
C
C-----------------------------------------------------------------
C  V VELOCITY [M][L]/[T]^2
C
     IF(CALIAS.EQ.'V VELOCITY') THEN
C
C
C.......SINK AT TOP
        CALL USRDEGAS(CALIAS,SP,W,VFRAC,DEN,AREA,IWORK(JDSURF),NDSURF
     +            ,IPNODB,IPFACB,TOTAL)
C       WRITE(NWRITE,*) ' V :',TOTAL
C.......SET MINIMUM VOID FRACTION
        VFMIN=1.0E-8
        DO 201  INODE=1,NNODE
         VFRAC(INODE,2)=MAX(VFRAC(INODE,2),VFMIN)

 201    CONTINUE
C
     ENDIF
C
C-----------------------------------------------------------------
C  W VELOCITY [M][L]/[T]^2
C
     IF(CALIAS.EQ.'W VELOCITY') THEN
C
C.......SINK AT TOP
        CALL USRDEGAS(CALIAS,SP,W,VFRAC,DEN,AREA,IWORK(JDSURF),NDSURF
     +            ,IPNODB,IPFACB,TOTAL)
C       WRITE(NWRITE,*) ' W :',TOTAL
C
C.......SET MINIMUM VOID FRACTION
        VFMIN=1.0E-8
        DO 301  INODE=1,NNODE
         VFRAC(INODE,2)=MAX(VFRAC(INODE,2),VFMIN)

 301    CONTINUE
C
     ENDIF
C
C+++++++++++++++++  END OF USER AREA 5   ++++++++++++++++++++++++++++
     ENDIF
C
C---- OVERWRITE SOURCE TERMS
      IF (ICALL.EQ.2) THEN
C
     ENDIF
     RETURN
      END
C
C
     SUBROUTINE USRDEGAS(CALIAS,SUP,W,VFRAC,DEN,AREA,IPTSRF,NDSURF
     +                ,IPNODB,IPFACB,TOTAL)
C
```

```
C*************************************************************
C
C   UTILITY SUBROUTINE FOR USER-SUPPLIED FREE SURFACE DEGASSING
C
C*************************************************************
C
C   THIS SUBROUTINE IS CALLED BY THE FOLLOWING SUBROUTINES
C     USRSRC
C
C*************************************************************
      CHARACTER     CALIAS*24
C
C+++++++++++++++++  USER AREA 1  +++++++++++++++++++++++++++++++++++
C---- AREA FOR USERS EXPLICITLY DECLARED VARIABLES
C
C+++++++++++++++++  END OF USER AREA 1   ++++++++++++++++++++++++++++
C
      COMMON
       + /ALL/    NBLOCK,NCELL,NBDRY,NNODE,NFACE,NVERT,NDIM
       + /ALLWRK/ NRWS,NIWS,NCWS,IWRFRE,IWIFRE,IWCFRE
       + /ADDIMS/ NPHASE,NSCAL,NVAR,NPROP
       +         ,NDVAR,NDPROP,NDXNN,NDGEOM,NDCOEF,NILIST,NRLIST,NTOPOL
       + /DEVICE/ NREAD,NWRITE,NRDISK,NWDISK
C
C
      DIMENSION SUP(NCELL,NPHASE)
     +,W(NNODE,NPHASE),VFRAC(NNODE,NPHASE),DEN(NNODE,NPHASE)
     +,AREA(NFACE,3),IPTSRF(NDSURF),IPNODB(NBDRY,4),IPFACB(NBDRY)
C
C
C---- STATEMENT FUNCTION FOR ADRESSING
C        IP(I,J,K)=IPT((K-1)*ILEN*JLEN+(J-1)*ILEN+I)
C
C+++++++++++++++++  USER AREA 5  +++++++++++++++++++++++++++++++++++
C
C-----DEGASSING PHASE INDEX
       TOTAL=0.0
C
C-----------------------------------------------------------------
C  VOLUME FRACTION
C
      IF(CALIAS.EQ.'VOLUME FRACTION') THEN
C
             DO 101 I=1,NDSURF
        INODE = IPTSRF(I)
        IBDRY = INODE-NCELL
        INODE1= IPNODB(IBDRY,1)
        IFACE = IPFACB(IBDRY)
        AREAM = SQRT ( AREA(IFACE,1)*AREA(IFACE,1)
     +                +AREA(IFACE,2)*AREA(IFACE,2)
     +                +AREA(IFACE,3)*AREA(IFACE,3) )
        FLUX  = DEN(INODE1,2)*
     +            AREAM*MAX(W(INODE1,2),0.0)
        SUP(INODE1,2)=SUP(INODE1,2)-FLUX
         IF(I.EQ.5) THEN
             WRITE(NWRITE,*) ' AREA= ', AREAM
        WRITE(NWRITE,*) 'W= ',  W(INODE1,2)
        WRITE(NWRITE,*) 'FLUX= ', FLUX
        WRITE(NWRITE,*) 'TOTAL==  ', TOTAL
        WRITE(NWRITE,*) 'VOL.FRAC= ', VFRAC(INODE1,2)
```

```
            ENDIF
        TOTAL = TOTAL+FLUX
C       WRITE(NWRITE,*) ' FLUX=',FLUX
 101  CONTINUE
C
C-----------------------------------------------------------------
C  OTHER VARIABLES
C
     ELSE
C
      DO 201 I=1,NDSURF
        INODE = IPTSRF(I)
        IBDRY =  INODE-NCELL
        INODE1= IPNODB(IBDRY,1)
        IFACE = IPFACB(IBDRY)
        AREAM =  SQRT ( AREA(IFACE,1)*AREA(IFACE,1)
     +                +AREA(IFACE,2)*AREA(IFACE,2)
     +                +AREA(IFACE,3)*AREA(IFACE,3)  )
        FLUX  = VFRAC(INODE1,2)*DEN(INODE1,2)
     +            *AREAM*MAX(W(INODE1,2),0.0)
       SUP(INODE1,2)=SUP(INODE1,2)-FLUX
            TOTAL = TOTAL+FLUX
C       WRITE(NWRITE,*) ' FLUX=',FLUX
 201  CONTINUE
C
     ENDIF
 401  CONTINUE
C
     RETURN
      END
C
C
     SUBROUTINE USRTRN(U,V,W,P,VFRAC,DEN,VIS,TE,ED,RS,T,H,RF,SCAL,
     +                 XP,YP,ZP,VOL,AREA,VPOR,ARPOR,WFACT,CONV,IPT,
     +                 IBLK,IPVERT,IPNODN,IPFACN,IPNODF,IPNODB,IPFACB,
     +                 WORK,IWORK,CWORK)
C
C*************************************************************
C
C  USER SUBROUTINE TO ALLOW USERS TO MODIFY OR MONITOR THE SOLUTION AT
C   THE END OF EACH TIME STEP
C   THIS SUBROUTINE IS CALLED BEFORE THE START OF THE RUN AS WELL AS AT
C   THE END OF EACH TIME STEP
C
C*************************************************************
C
C   THIS SUBROUTINE IS CALLED BY THE FOLLOWING SUBROUTINES
C      CUSR  TRNMOD
C
C*************************************************************
C
C*************************************************************
C
     LOGICAL LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
     +       ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
C
      CHARACTER*(*) CWORK
C
C+++++++++++++++  USER AREA 1 +++++++++++++++++++++++++++++++++++
C---- AREA FOR USERS EXPLICITLY DECLARED VARIABLES
```

```
C
C+++++++++++++++  END OF USER AREA 1  ++++++++++++++++++++++++++
C
      COMMON
       + /ALL/    NBLOCK,NCELL,NBDRY,NNODE,NFACE,NVERT,NDIM
       + /ALLWRK/ NRWS,NIWS,NCWS,IWRFRE,IWIFRE,IWCFRE
       + /ADDIMS/ NPHASE,NSCAL,NVAR,NPROP
       +          ,NDVAR,NDPROP,NDXNN,NDGEOM,NDCOEF,NILIST,NRLIST,NTOPOL
       + /CHKUSR/ IVERS,IUCALL,IUSED
       + /CONC/   NCONC
       + /DEVICE/ NREAD,NWRITE,NRDISK,NWDISK
       + /IDUM/   ILEN,JLEN
       + /LOGIC/  LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
       +          ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
       + /MLTGRD/ MLEVEL,NLEVEL,ILEVEL
       + /SGLDBL/ IFLGPR,ICHKPR
       + /SPARM/  SMALL,SORMAX,NITER,INDPRI,MAXIT,NODREF,NODMON
       + /TIMUSR/ DTUSR
       + /TRANSI/ NSTEP,KSTEP,MF,INCORE
       + /TRANSR/ TIME,DT,DTINVF,TPARM
C
C+++++++++++++++  USER AREA 2 +++++++++++++++++++++++++++++++++
C---- AREA FOR USERS TO DECLARE THEIR OWN COMMON BLOCKS
C     THESE SHOULD START WITH THE CHARACTERS 'UC' TO ENSURE
C     NO CONFLICT WITH NON-USER COMMON BLOCKS
C
      COMMON / UCSURF/ JISURF,NISURF,JDSURF,NDSURF
      COMMON /UCINJR/ FINJ1,FINJ2,UGASIN,VGASIN,WGASIN
      COMMON /UCFLUX/ GASIN,GASOUT
C
C+++++++++++++++  END OF USER AREA 2  +++++++++++++++++++++++++
C
      DIMENSION
     + U(NNODE,NPHASE),V(NNODE,NPHASE),W(NNODE,NPHASE),P(NNODE,NPHASE)
     +,VFRAC(NNODE,NPHASE),DEN(NNODE,NPHASE),VIS(NNODE,NPHASE)
     +,TE(NNODE,NPHASE),ED(NNODE,NPHASE),RS(NNODE,NPHASE,6)
     +,T(NNODE,NPHASE),H(NNODE,NPHASE),RF(NNODE,NPHASE,4)
     +,SCAL(NNODE,NPHASE,NSCAL)
      DIMENSION
     + XP(NNODE),YP(NNODE),ZP(NNODE)
     +,VOL(NCELL),AREA(NFACE,3),VPOR(NCELL),ARPOR(NFACE,3)
     +,WFACT(NFACE),CONV(NFACE,NPHASE)
      +,IPT(*),IBLK(5,NBLOCK)
     +,IPVERT(NCELL,8),IPNODN(NCELL,6),IPFACN(NCELL,6),IPNODF(NFACE,4)
     +,IPNODB(NBDRY,4),IPFACB(NBDRY)
      +,IWORK(*),WORK(*),CWORK(*)
C
C
C---- STATEMENT FUNCTION FOR ADDRESSING
       IP(I,J,K)=IPT((K-1)*ILEN*JLEN+(J-1)*ILEN+I)
C
C----VERSION NUMBER OF USER ROUTINE AND PRECISION FLAG
C
      IVERS=3
      ICHKPR = 1
C
C+++++++++++++++  USER AREA 4 +++++++++++++++++++++++++++++++++
C---- TO USE THIS USER ROUTINE FIRST SET IUSED=1
C
       IUSED=1
```

```
      IF (IUSED.EQ.0) RETURN
C+++++++++++++++  END OF USER AREA 4  ++++++++++++++++++++++++
C
C---- FRONTEND CHECKING OF USER ROUTINE
      IF (IUCALL.EQ.0) RETURN
C
C+++++++++++++++  USER AREA 5 +++++++++++++++++++++++++++++++
C
     IF (KSTEP.EQ.0) THEN
C
C-----INJECTION BOUNDARY LOCATIONS
C
      CALL IPALL('Entrance','*','PATCH','CENTRES',IPT,NPT,
     +          CWORK,IWORK)
C
     NISURF=NPT
C    SET INJECTION BOUNDARY LIST INTO INTERGER WORK SPACE
      CALL SETPER('USRTRN','IWORK ','ISURF ',NISURF,JISURF)
C
      DO 101 I=1,NISURF
       IWORK(JISURF+I-1)=IPT(I)
 101  CONTINUE
     WRITE(NWRITE,*) '   '
     WRITE(NWRITE,*) '********   USRTRN - INLET **********'
     WRITE(NWRITE,*) '  NISURF=',NISURF,' JISURF=',JISURF
     WRITE(NWRITE,*) '  IWORK(JISURF)=',IWORK(JISURF)
     WRITE(NWRITE,*) '  IWORK(JISURF+NISURF-1)=',IWORK(JISURF+NISURF-1)
C
C-----INJECTION RATES: MASS FLUX IN kg/s,  VELOCITY IN  m/s
     FGASIN =  0
     UGASIN =  0
     VGASIN =  0
     WGASIN =  0
C
C-----DETERMINE TOTAL AREA OF INLET BOUNDARY AND MASS FLUX PER AREA
      TOTALA =  0.0
      DO 201 I=1,NISURF
       INODE = IWORK(JISURF+I-1)
       IBDRY =  INODE-NCELL
       IFACE = IPFACB(IBDRY)
       AREAM =  SQRT ( AREA(IFACE,1)*AREA(IFACE,1)
     +                +AREA(IFACE,2)*AREA(IFACE,2)
     +                +AREA(IFACE,3)*AREA(IFACE,3) )
        TOTALA = TOTALA+AREAM
 201  CONTINUE
C
     WRITE(NWRITE,*) '  TOTAL AREA OF INLET =',TOTALA
      FINJ2 = FGASIN/TOTALA
C
C
C
C-----DEGASSING BOUNDARY LOCATIONS
C
      CALL IPALL('Top','*','PATCH','CENTRES',IPT,NPT,
     +          CWORK,IWORK)
C
     NDSURF=NPT
C    SET DEGASSING BOUNDARY LIST INTO INTERGER WORK SPACE
      CALL SETPER('USRTRN','IWORK ','DSURF ',NDSURF,JDSURF)
C
```

```
      DO 401 I=1,NDSURF
       IWORK(JDSURF+I-1)=IPT(I)
 401  CONTINUE
      WRITE(NWRITE,*) '   '
      WRITE(NWRITE,*) '********   USRTRN - OUTLET *********'
      WRITE(NWRITE,*) ' NDSURF=',NDSURF,' JDSURF=',JDSURF
      WRITE(NWRITE,*) ' IWORK(JDSURF)=',IWORK(JDSURF)
      WRITE(NWRITE,*) ' IWORK(JDSURF+NDSURF-1)=',IWORK(JDSURF+NDSURF-1)
C
      ENDIF
C
C-----------------------------------------------------------------------
C     END OPERATION
C
      IF(KSTEP.EQ.NSTEP) THEN
C
C=====CHECK MASS BALANCE
C
C.....GAS FLOW RATE FROM SOURCE IS SET IN USRSRC

      INFLOE_SOURCE=GASIN
C.....INLET

      CALL IPALL('Entrance','*','PATCH','CENTRES',IPT,NPT,
     +           CWORK,IWORK)
C
      GASIN=0.0
      GASIN_2=0.0
      WATIN =  0.0
C.....GASIN INITIALISED IN USRSRC
      DO 501  I=1,NPT
       INODE = IPT(I)
       IBDRY = INODE-NCELL
       IFACE = IPFACB(IBDRY)
       WATIN = WATIN+CONV(IFACE,1)
       GASIN_2 = GASIN_2+CONV(IFACE,2)
 501  CONTINUE

      GASIN=GASIN_2
      SUM=0.0
      SUM=INFLOE_SOURCE+GASIN
      WRITE(NWRITE,*) '  '

      WRITE(NWRITE,*) '&&&&&&&  SHOWING OF OUTPUT OF INFLOW DATA &&&&&'
      WRITE(NWRITE,*) '  '
      WRITE(NWRITE,*) 'GAS INFLOW RATE FROM SOURCE=',INFLOE_SOURCE
      WRITE(NWRITE,*) 'GAS INFLOW RATE FROM INLET=',GASIN
      WRITE(NWRITE,*) 'THE SUM OF GAS INFLOW RATE FROM SOURCE AND
     +INLET=',SUM
      WRITE(NWRITE,*) '  '

      WRITE(NWRITE,*) '  '
C
C.....OUTLET
      GASOUT_SINK=GASOUT
      WRITE(NWRITE,*) '&&&&&&&  SHOWING OF OUTPUT OF OUTFLOW DATA &&&&&'
      WRITE(NWRITE,*) '  '
      WRITE(NWRITE,*) 'GAS OUTFLOW RATE FROM TOP SURFACE=', GASOUT_SINK

      WRITE(NWRITE,*) '  '
```

```
      CALL IPALL('OUTLET','*','PATCH','CENTRES',IPT,NPT,
     +          CWORK,IWORK)
C
     GASOUT =  0.0
     GASOUT_2 = 0.0
     WATOUT =  0.0
C.....GASOUT INITIALISED IN USRSRC
      DO 601 I=1,NPT
       INODE = IPT(I)
       IBDRY = INODE-NCELL
       IFACE = IPFACB(IBDRY)
       WATOUT = WATOUT+CONV(IFACE,1)
       GASOUT_2 = GASOUT_2+CONV(IFACE,2)
 601  CONTINUE
     GASOUT =GASOUT_2
      SUM1=0.0
      SUM1= GASOUT_SINK+GASOUT

     WRITE(NWRITE,*) 'GAS OUTFLOW RATE FROM BOTTUM=', GASOUT
     WRITE(NWRITE,*) 'THE SUM OF GAS OUTFLOW RATE FROM TOP SURFACE AND
     + BOTTOM=',SUM
     WRITE(NWRITE,*) ' '
     WRITE(NWRITE,*) ' '
C
     WRITE(NWRITE,*) '  ***  MASS BALANCE ***'
     WRITE(NWRITE,*) ' '
     WRITE(NWRITE,*) ' AIR'
     WRITE(NWRITE,*) ' TOTAL   INFLOW   (kg/s)  =',SUM
     WRITE(NWRITE,*) ' TOTAL   OUTFLOW (kg/s)  =',SUM1
     GASERR=(SUM1-SUM)*100.0/SUM
     WRITE(NWRITE,*) '   % ERROR        =',GASERR
     WRITE(NWRITE,*) ' '
     WRITE(NWRITE,*) ' WATER'
     WRITE(NWRITE,*) '   INFLOW   (kg/s)  =',WATIN
     WRITE(NWRITE,*) '   OUTFLOW (kg/s)  =',WATOUT
      WATERR=(WATOUT+WATIN)*100.0/WATIN
     WRITE(NWRITE,*) '   % ERROR        =',WATERR
     WRITE(NWRITE,*) ' '
C
C.....SET IPLT=1 TO CLEAN UP VELOCITY FOR PLOTTING
C*****NOTE  THAT THIS OPERATION WILL SLIGHTLY ALTER THE VELOCITY FIELD
C     HENCE THE DUMP FILE GENERATED SHOULD BE USED FOR PLOTTING ONLY
C     NOT FOR RESTART.
C
       IPLT=0
C
C.....FILTER OUT VELOCITY WITH SMALL VOLUME FRACTION
       IF(IPLT.EQ.1) THEN
          DO 701  INODE=1,NNODE
           FILTER=1.0-EXP(-3.0*VFRAC(INODE,2)/1.E-4)
          U(INODE,2)=U(INODE,2)*FILTER
          V(INODE,2)=V(INODE,2)*FILTER
          W(INODE,2)=W(INODE,2)*FILTER
 701      CONTINUE
C
      ENDIF
C
     ENDIF
C
C+++++++++++++++  END OF USER AREA 5   ++++++++++++++++++++++++++
```

```
C
      RETURN
      END
```

Appendix E. Subroutine for nozzle simulation

```
SUBROUTINE USRTRN(U,V,W,P,VFRAC,DEN,VIS,TE,ED,RS,T,H,RF,SCAL,
     +                XP,YP,ZP,VOL,AREA,VPOR,ARPOR,WFACT,CONV,IPT,
     +                IBLK,IPVERT,IPNODN,IPFACN,IPNODF,IPNODB,IPFACB,
     +                WORK,IWORK,CWORK)
C
C*********************************************************
C
C  USER SUBROUTINE TO ALLOW USERS TO MODIFY OR MONITOR THE SOLUTION AT
C   THE END OF EACH TIME STEP
C   THIS SUBROUTINE IS CALLED BEFORE THE START OF THE RUN AS WELL AS AT
C   THE END OF EACH TIME STEP
C
C*********************************************************
C
C   THIS SUBROUTINE IS CALLED BY THE FOLLOWING SUBROUTINES
C     CUSR  TRNMOD
C
C*********************************************************
C
C*********************************************************
C
     LOGICAL LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
     +       ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
C
     CHARACTER*(*) CWORK
C
C+++++++++++++++  USER AREA 1 +++++++++++++++++++++++++++++++++
C---- AREA FOR USERS EXPLICITLY DECLARED VARIABLES
C
C+++++++++++++++  END OF USER AREA 1  +++++++++++++++++++++++++
C
     COMMON
      + /ALL/    NBLOCK,NCELL,NBDRY,NNODE,NFACE,NVERT,NDIM
      + /ALLWRK/ NRWS,NIWS,NCWS,IWRFRE,IWIFRE,IWCFRE
      + /ADDIMS/ NPHASE,NSCAL,NVAR,NPROP
      +         ,NDVAR,NDPROP,NDXNN,NDGEOM,NDCOEF,NILIST,NRLIST,NTOPOL
      + /CHKUSR/ IVERS,IUCALL,IUSED
      + /CONC/   NCONC
      + /DEVICE/ NREAD,NWRITE,NRDISK,NWDISK
      + /IDUM/   ILEN,JLEN
      + /LOGIC/  LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
      +         ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
      + /MLTGRD/ MLEVEL,NLEVEL,ILEVEL
      + /SGLDBL/ IFLGPR,ICHKPR
      + /SPARM/  SMALL,SORMAX,NITER,INDPRI,MAXIT,NODREF,NODMON
      + /TIMUSR/ DTUSR
      + /TRANSI/ NSTEP,KSTEP,MF,INCORE
      + /TRANSR/ TIME,DT,DTINVF,TPARM
C
C+++++++++++++++  USER AREA 2 ++++++++++++++++++++++++++++++
C---- AREA FOR USERS TO DECLARE THEIR OWN COMMON BLOCKS
C    THESE SHOULD START WITH THE CHARACTERS 'UC' TO ENSURE
C     NO CONFLICT WITH NON-USER COMMON BLOCKS
C
     COMMON /UCSURF/ JISURF,NISURF,JDSURF,NDSURF
     COMMON /UCINJR/ FINJ1,FINJ2,UGASIN,VGASIN,WGASIN
     COMMON / UCFLUX/ GASIN,GASOUT
```

```
C
C+++++++++++++++  END OF USER AREA 2  +++++++++++++++++++++++++++++
C
      DIMENSION
     + U(NNODE,NPHASE),V(NNODE,NPHASE),W(NNODE,NPHASE),P(NNODE,NPHASE)
     +,VFRAC(NNODE,NPHASE),DEN(NNODE,NPHASE),VIS(NNODE,NPHASE)
     +,TE(NNODE,NPHASE),ED(NNODE,NPHASE),RS(NNODE,NPHASE,6)
     +,T(NNODE,NPHASE),H(NNODE,NPHASE),RF(NNODE,NPHASE,4)
     +,SCAL(NNODE,NPHASE,NSCAL)
      DIMENSION
     + XP(NNODE),YP(NNODE),ZP(NNODE)
     +,VOL(NCELL),AREA(NFACE,3),VPOR(NCELL),ARPOR(NFACE,3)
     +,WFACT(NFACE),CONV(NFACE,NPHASE)
      +,IPT(*),IBLK(5,NBLOCK)
     +,IPVERT(NCELL,8),IPNODN(NCELL,6),IPFACN(NCELL,6),IPNODF(NFACE,4)
     +,IPNODB(NBDRY,4),IPFACB(NBDRY)
      +,IWORK(*),WORK(*),CWORK(*), TMP(20,20,11)
C
C
C---- STATEMENT FUNCTION FOR ADDRESSING
       IP(I,J,K)=IPT((K-1)*ILEN*JLEN+(J-1)*ILEN+I)
C
C----VERSION NUMBER OF USER ROUTINE AND PRECISION FLAG
C
      IVERS=3
      ICHKPR = 1
C
C+++++++++++++++  USER AREA 4 +++++++++++++++++++++++++++++++++
C---- TO USE THIS USER ROUTINE FIRST SET IUSED=1
C
       IUSED=1
       IF (IUSED.EQ.0) RETURN
C+++++++++++++++  END OF USER AREA 4  ++++++++++++++++++++++++
C
C---- FRONTEND CHECKING OF USER ROUTINE
       IF (IUCALL.EQ.0) RETURN
C
C+++++++++++++++  USER AREA 5 ++++++++++++++++++++++++++++++++
C
      IF (KSTEP.EQ.NSTEP) THEN
C
C-----INJECTION BOUNDARY LOCATIONS
C
      OPEN(5,FILE='DATA1.DAT',STATUS='NEW',FORM='FORMATTED')

C
       VEL =  0.0
       TOTAL_BOOT = 0
       TOT_UP_U = 0
       TOT_UP_V = 0
       TOT_UP_W = 0
       TOT_UP_K = 0
       TOT_UP_E = 0
        TOT_AV_U = 0
        TOT_AV_V = 0
        TOT_AV_W = 0
        TOT_AV_K = 0
        TOT_AV_E = 0
       GAS_AV_U = 0
        GAS_AV_V = 0
```

```
      GAS_AV_W = 0
     GAS_FLOWRATE = 0
     LIQ_FLOWRATE = 0
     AREA_FRAC1 = 0
     AREA_FRAC2 = 0
      X_LIQ_OUT = 0
      Y_LIQ_OUT = 0
      Z_LIQ_OUT = 0
       TOTALA = 0

      IHEAD = 0

       CALL IPREC('BLOCK-NUMBER-19','BLOCK','CENTRES',IPT,ILEN,
     +          JLEN,KLEN,CWORK,IWORK)

C
C
      DO 203 K=1,KLEN
       DO 202 I=1,ILEN
              INODE = IP(I,8,K)
         IFACE = INODE+NCELL
C
C  Record the variables

          TMP(KLEN+1-K,IHEAD+I,1) = U(INODE,1)
              TMP(KLEN+1-K,IHEAD+I,2) = V(INODE,1)
              TMP(KLEN+1-K,IHEAD+I,3) = W(INODE,1)
          TMP(KLEN+1-K,IHEAD+I,4) = TE(INODE,1)
              TMP(KLEN+1-K,IHEAD+I,5) = ED(INODE,1)
              TMP(KLEN+1-K,IHEAD+I,6) = U(INODE,2)
              TMP(KLEN+1-K,IHEAD+I,7) = V(INODE,2)
              TMP(KLEN+1-K,IHEAD+I,8) = W(INODE,2)
              TMP(KLEN+1-K,IHEAD+I,9) = VFRAC(INODE,1)
              TMP(KLEN+1-K,IHEAD+I,10) = VFRAC(INODE,2)
C   End of recording variables

C  VEL is speed at current node

              VEL = SQRT(U(INODE,1)*U(INODE,1)+
     +          V(INODE,1)*V(INODE,1)+W(INODE,1)*W(INODE,1))
              AREAX = AREA(IFACE,1)
         AREAY = AREA(IFACE,2)
              AREAZ = AREA(IFACE,3)


C  Following average variable is calculated according to Hua's Formula

         TOTAL_BOOT=TOTAL_BOOT+VEL*AREAX*VFRAC(INODE,1)

         TOT_UP_U=TOT_UP_U+U(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
              TOT_UP_V=TOT_UP_V+V(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
              TOT_UP_W=TOT_UP_W+W(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
              TOT_UP_K=TOT_UP_K+TE(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
              TOT_UP_E=TOT_UP_E+ED(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
C   End of Hua's Average variables

C   Followinf variables are based on area weighted average
           TOT_AV_U = TOT_AV_U+AREAX*U(INODE,1)*VFRAC(INODE,1)
               TOT_AV_V = TOT_AV_V+AREAX*V(INODE,1)*VFRAC(INODE,1)
               TOT_AV_W = TOT_AV_W+AREAX*W(INODE,1)*VFRAC(INODE,1)
```

```
            TOT_AV_K = TOT_AV_K+AREAX*TE(INODE,1)*VFRAC(INODE,1)
            TOT_AV_E = TOT_AV_E+AREAX*ED(INODE,1)*VFRAC(INODE,1)
C   End of area weighted average

            GAS_AV_U = GAS_AV_U+AREAX*U(INODE,2)*VFRAC(INODE,2)
        GAS_AV_V = GAS_AV_V+AREAX*V(INODE,2)*VFRAC(INODE,2)
        GAS_AV_W = GAS_AV_W+AREAX*W(INODE,2)*VFRAC(INODE,2)

       AREA_FRAC1 = AREA_FRAC1 + AREAX*VFRAC(INODE,1)
            AREA_FRAC2 = AREA_FRAC2 + AREAX*VFRAC(INODE,2)

C Liquid flowrate is calculate in three faces respectively. They are
C added up in final stage

       X_LIQ_OUT=X_LIQ_OUT+U(INODE,1)*AREAX*VFRAC(INODE,1)
       Y_LIQ_OUT=Y_LIQ_OUT+V(INODE,1)*AREAY*VFRAC(INODE,1)
        Z_LIQ_OUT=Z_LIQ_OUT+W(INODE,1)*AREAZ*VFRAC(INODE,1)

       GAS_FLOWRATE=GAS_FLOWRATE+AREAX*U(INODE,2)*VFRAC(INODE,2)
     +                +AREAY*V(INODE,2)*VFRAC(INODE,2)
     +                +AREAZ*W(INODE,2)*VFRAC(INODE,2)

            TOTALA = TOTALA+AREAX
 202  CONTINUE
 203  CONTINUE

     IHEAD = IHEAD+ILEN

      CALL IPREC('BLOCK-NUMBER-20','BLOCK','CENTRES',IPT,ILEN,
     +          JLEN,KLEN,CWORK,IWORK)

     DO 303  K=1,KLEN
      DO 302  I=1,ILEN
        INODE = IP(I,8,K)
        IFACE = INODE+NCELL
C
C  Record the variables

        TMP(KLEN+1-K,IHEAD+I,1) = U(INODE,1)
        TMP(KLEN+1-K,IHEAD+I,2) = V(INODE,1)
        TMP(KLEN+1-K,IHEAD+I,3) = W(INODE,1)
        TMP(KLEN+1-K,IHEAD+I,4) = TE(INODE,1)
        TMP(KLEN+1-K,IHEAD+I,5) = ED(INODE,1)
        TMP(KLEN+1-K,IHEAD+I,6) = U(INODE,2)
        TMP(KLEN+1-K,IHEAD+I,7) = V(INODE,2)
        TMP(KLEN+1-K,IHEAD+I,8) = W(INODE,2)
        TMP(KLEN+1-K,IHEAD+I,9) = VFRAC(INODE,1)
        TMP(KLEN+1-K,IHEAD+I,10) = VFRAC(INODE,2)
C   End of recording variables

C  VEL is speed at current node

        VEL = SQRT(U(INODE,1)*U(INODE,1)+
     +          V(INODE,1)*V(INODE,1)+W(INODE,1)*W(INODE,1))
        AREAX = AREA(IFACE,1)
        AREAY = AREA(IFACE,2)
        AREAZ = AREA(IFACE,3)


C  Following average variable is calculated according to Hua's Formula
```

```
      TOTAL_BOOT=TOTAL_BOOT+VEL*AREAX*VFRAC(INODE,1)

      TOT_UP_U=TOT_UP_U+U(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
      TOT_UP_V=TOT_UP_V+V(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
      TOT_UP_W=TOT_UP_W+W(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
      TOT_UP_K=TOT_UP_K+TE(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
      TOT_UP_E=TOT_UP_E+ED(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
C   End of Hua's Average variables

C   Followinf variables are based on area weighted average
      TOT_AV_U = TOT_AV_U+AREAX*U(INODE,1)*VFRAC(INODE,1)
      TOT_AV_V = TOT_AV_V+AREAX*V(INODE,1)*VFRAC(INODE,1)
      TOT_AV_W = TOT_AV_W+AREAX*W(INODE,1)*VFRAC(INODE,1)
      TOT_AV_K = TOT_AV_K+AREAX*TE(INODE,1)*VFRAC(INODE,1)
      TOT_AV_E = TOT_AV_E+AREAX*ED(INODE,1)*VFRAC(INODE,1)
C   End of area weighted average

      GAS_AV_U = GAS_AV_U+AREAX*U(INODE,2)*VFRAC(INODE,2)
      GAS_AV_V = GAS_AV_V+AREAX*V(INODE,2)*VFRAC(INODE,2)
      GAS_AV_W = GAS_AV_W+AREAX*W(INODE,2)*VFRAC(INODE,2)

      AREA_FRAC1 = AREA_FRAC1 + AREAX*VFRAC(INODE,1)
      AREA_FRAC2 = AREA_FRAC2 + AREAX*VFRAC(INODE,2)

C Liquid flowrate is calculate in three faces respectively. They are
C added up in final stage

      X_LIQ_OUT=X_LIQ_OUT+U(INODE,1)*AREAX*VFRAC(INODE,1)
      Y_LIQ_OUT=Y_LIQ_OUT+V(INODE,1)*AREAY*VFRAC(INODE,1)
      Z_LIQ_OUT=Z_LIQ_OUT+W(INODE,1)*AREAZ*VFRAC(INODE,1)

      GAS_FLOWRATE=GAS_FLOWRATE+AREAX*U(INODE,2)*VFRAC(INODE,2)
     +              +AREAY*V(INODE,2)*VFRAC(INODE,2)
     +              +AREAZ*W(INODE,2)*VFRAC(INODE,2)

      TOTALA = TOTALA+AREAX
 302  CONTINUE
 303  CONTINUE

     IHEAD = IHEAD+ILEN

     CALL IPREC('BLOCK-NUMBER-21','BLOCK','CENTRES',IPT,ILEN,
     +         JLEN,KLEN,CWORK,IWORK)


     DO 403 K=1,KLEN
      DO 402 I=1,ILEN
        INODE = IP(I,8,K)
        IFACE = INODE+NCELL
C
C  Record the variables

        TMP(KLEN+1-K,IHEAD+I,1) = U(INODE,1)
        TMP(KLEN+1-K,IHEAD+I,2) = V(INODE,1)
        TMP(KLEN+1-K,IHEAD+I,3) = W(INODE,1)
        TMP(KLEN+1-K,IHEAD+I,4) = TE(INODE,1)
        TMP(KLEN+1-K,IHEAD+I,5) = ED(INODE,1)
        TMP(KLEN+1-K,IHEAD+I,6) = U(INODE,2)
        TMP(KLEN+1-K,IHEAD+I,7) = V(INODE,2)
```

```
      TMP(KLEN+1-K,IHEAD+I,8) = W(INODE,2)
      TMP(KLEN+1-K,IHEAD+I,9) = VFRAC(INODE,1)
      TMP(KLEN+1-K,IHEAD+I,10) = VFRAC(INODE,2)
C   End of recording variables

C  VEL is speed at current node

      VEL = SQRT(U(INODE,1)*U(INODE,1)+
     +        V(INODE,1)*V(INODE,1)+W(INODE,1)*W(INODE,1))
      AREAX = AREA(IFACE,1)
      AREAY = AREA(IFACE,2)
      AREAZ = AREA(IFACE,3)


C  Following average variable is calculated according to Hua's Formula

      TOTAL_BOOT=TOTAL_BOOT+VEL*AREAX*VFRAC(INODE,1)

      TOT_UP_U=TOT_UP_U+U(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
      TOT_UP_V=TOT_UP_V+V(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
      TOT_UP_W=TOT_UP_W+W(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
      TOT_UP_K=TOT_UP_K+TE(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
      TOT_UP_E=TOT_UP_E+ED(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
C   End of Hua's Average variables

C   Followinf variables are based on area weighted average
      TOT_AV_U = TOT_AV_U+AREAX*U(INODE,1)*VFRAC(INODE,1)
      TOT_AV_V = TOT_AV_V+AREAX*V(INODE,1)*VFRAC(INODE,1)
      TOT_AV_W = TOT_AV_W+AREAX*W(INODE,1)*VFRAC(INODE,1)
      TOT_AV_K = TOT_AV_K+AREAX*TE(INODE,1)*VFRAC(INODE,1)
      TOT_AV_E = TOT_AV_E+AREAX*ED(INODE,1)*VFRAC(INODE,1)
C   End of area weighted average

      GAS_AV_U = GAS_AV_U+AREAX*U(INODE,2)*VFRAC(INODE,2)
      GAS_AV_V = GAS_AV_V+AREAX*V(INODE,2)*VFRAC(INODE,2)
      GAS_AV_W = GAS_AV_W+AREAX*W(INODE,2)*VFRAC(INODE,2)

      AREA_FRAC1 = AREA_FRAC1 + AREAX*VFRAC(INODE,1)
      AREA_FRAC2 = AREA_FRAC2 + AREAX*VFRAC(INODE,2)

C Liquid flowrate is calculate in three faces respectively. They are
C added up in final stage

      X_LIQ_OUT=X_LIQ_OUT+U(INODE,1)*AREAX*VFRAC(INODE,1)
      Y_LIQ_OUT=Y_LIQ_OUT+V(INODE,1)*AREAY*VFRAC(INODE,1)
      Z_LIQ_OUT=Z_LIQ_OUT+W(INODE,1)*AREAZ*VFRAC(INODE,1)

      GAS_FLOWRATE=GAS_FLOWRATE+AREAX*U(INODE,2)*VFRAC(INODE,2)
     +              +AREAY*V(INODE,2)*VFRAC(INODE,2)
     +              +AREAZ*W(INODE,2)*VFRAC(INODE,2)

      TOTALA = TOTALA+AREAX
 402  CONTINUE
 403  CONTINUE

     IHEAD =IHEAD+ILEN

     CALL IPREC('BLOCK-NUMBER-22','BLOCK','CENTRES',IPT,ILEN,
     +        JLEN,KLEN,CWORK,IWORK)
```

```
      DO 503 K=1,KLEN
       DO 502 I=1,ILEN
         INODE = IP(I,8,K)
         IFACE = INODE+NCELL
C
C  Record the variables

         TMP(KLEN+1-K,IHEAD+I,1) = U(INODE,1)
         TMP(KLEN+1-K,IHEAD+I,2) = V(INODE,1)
         TMP(KLEN+1-K,IHEAD+I,3) = W(INODE,1)
         TMP(KLEN+1-K,IHEAD+I,4) = TE(INODE,1)
         TMP(KLEN+1-K,IHEAD+I,5) = ED(INODE,1)
         TMP(KLEN+1-K,IHEAD+I,6) = U(INODE,2)
         TMP(KLEN+1-K,IHEAD+I,7) = V(INODE,2)
         TMP(KLEN+1-K,IHEAD+I,8) = W(INODE,2)
         TMP(KLEN+1-K,IHEAD+I,9) = VFRAC(INODE,1)
         TMP(KLEN+1-K,IHEAD+I,10) = VFRAC(INODE,2)
C   End of recording variables

C  VEL is speed at current node

         VEL = SQRT(U(INODE,1)*U(INODE,1)+
     +          V(INODE,1)*V(INODE,1)+W(INODE,1)*W(INODE,1))
         AREAX = AREA(IFACE,1)
         AREAY = AREA(IFACE,2)
         AREAZ = AREA(IFACE,3)


C  Following average variable is calculated according to Hua's Formula

         TOTAL_BOOT=TOTAL_BOOT+VEL*AREAX*VFRAC(INODE,1)

         TOT_UP_U=TOT_UP_U+U(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
         TOT_UP_V=TOT_UP_V+V(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
         TOT_UP_W=TOT_UP_W+W(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
         TOT_UP_K=TOT_UP_K+TE(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
         TOT_UP_E=TOT_UP_E+ED(INODE,1)*VEL*AREAX*VFRAC(INODE,1)
C   End of Hua's Average variables

C   Followinf variables are based on area weighted average
          TOT_AV_U = TOT_AV_U+AREAX*U(INODE,1)*VFRAC(INODE,1)
          TOT_AV_V = TOT_AV_V+AREAX*V(INODE,1)*VFRAC(INODE,1)
          TOT_AV_W = TOT_AV_W+AREAX*W(INODE,1)*VFRAC(INODE,1)
          TOT_AV_K = TOT_AV_K+AREAX*TE(INODE,1)*VFRAC(INODE,1)
          TOT_AV_E = TOT_AV_E+AREAX*ED(INODE,1)*VFRAC(INODE,1)
C   End of area weighted average

          GAS_AV_U = GAS_AV_U+AREAX*U(INODE,2)*VFRAC(INODE,2)
          GAS_AV_V = GAS_AV_V+AREAX*V(INODE,2)*VFRAC(INODE,2)
          GAS_AV_W = GAS_AV_W+AREAX*W(INODE,2)*VFRAC(INODE,2)

         AREA_FRAC1 = AREA_FRAC1 + AREAX*VFRAC(INODE,1)
         AREA_FRAC2 = AREA_FRAC2 + AREAX*VFRAC(INODE,2)

C Liquid flowrate is calculate in three faces respectively. They are
C added up in final stage

         X_LIQ_OUT=X_LIQ_OUT+U(INODE,1)*AREAX*VFRAC(INODE,1)
         Y_LIQ_OUT=Y_LIQ_OUT+V(INODE,1)*AREAY*VFRAC(INODE,1)
```

```
      Z_LIQ_OUT=Z_LIQ_OUT+W(INODE,1)*AREAZ*VFRAC(INODE,1)

     GAS_FLOWRATE=GAS_FLOWRATE+AREAX*U(INODE,2)*VFRAC(INODE,2)
    +              +AREAY*V(INODE,2)*VFRAC(INODE,2)
    +              +AREAZ*W(INODE,2)*VFRAC(INODE,2)

      TOTALA = TOTALA+AREAX
 502  CONTINUE
 503  CONTINUE

      IHEAD = IHEAD+ILEN
          KHEAD = KLEN
C
    AVERAGE_U=TOT_UP_U/TOTAL_BOOT
    AVERAGE_V=TOT_UP_V/TOTAL_BOOT
    AVERAGE_W=TOT_UP_W/TOTAL_BOOT
    AVERAGE_K=TOT_UP_K/TOTAL_BOOT
    AVERAGE_E=TOT_UP_E/TOTAL_BOOT

    AREA_AV_FRAC1 = AREA_FRAC1/TOTALA
    AREA_AV_FRAC2 = AREA_FRAC2/TOTALA
    AREA_AV_U=TOT_AV_U/AREA_FRAC1
    AREA_AV_V=TOT_AV_V/AREA_FRAC1
    AREA_AV_W=TOT_AV_W/AREA_FRAC1
    AREA_AV_K=TOT_AV_K/AREA_FRAC1
    AREA_AV_E=TOT_AV_E/AREA_FRAC1

    AREA_GAS_U=GAS_AV_U/AREA_FRAC2
    AREA_GAS_V=GAS_AV_V/AREA_FRAC2
    AREA_GAS_W=GAS_AV_W/AREA_FRAC2

    FLOWRATE_LIQ = 2*(X_LIQ_OUT+Y_LIQ_OUT+Z_LIQ_OUT)
    GAS_FLOWRATE = 2*GAS_FLOWRATE
    GAS_FRAC=100*GAS_FLOWRATE/(GAS_FLOWRATE+FLOWRATE_LIQ)

    Ver = AVERAGE_W/AVERAGE_U
    Hor = AVERAGE_V/AVERAGE_U

    V_JET_ANGLE = ATAN(Ver)*180/3.14159
    H_JET_ANGLE =  ATAN(Hor)*180/3.14159

    OPEN(6,FILE='DATA2.DAT',STATUS='NEW')
    WRITE(6,*) 'AVERAGE LIQUID U ON THE PORT',AVERAGE_U
    WRITE(6,*) ' '
    WRITE(6,*) 'AVERAGE LIQUID V ON THE PORT',AVERAGE_V
    WRITE(6,*) ' '
    WRITE(6,*) 'AVERAGE LIQUID W ON THE PORT',AVERAGE_W
    WRITE(6,*) ' '
    WRITE(6,*) 'AVERAGE K ON THE PORT       ',AVERAGE_K
    WRITE(6,*) ' '
    WRITE(6,*) 'AVERAGE EPSILON ON THE PORT ',AVERAGE_E

    WRITE(6,*) ' '
    WRITE(6,*) 'VERTICAL JET ANGLE=         ',V_JET_ANGLE
    WRITE(6,*) ' '
    WRITE(6,*) 'HORIZONTAL JET ANGLE=       ',H_JET_ANGLE
    WRITE(6,*) ' '
    WRITE(6,*) '##############################################'
    WRITE(6,*) 'TOTAL AREA OF PORT =',TOTALA
    WRITE(6,*) 'LIQUID FLOWRATE AT X AXIS=',X_LIQ_OUT
```

```
      WRITE(6,*) 'LIQUID FLOWRATE AT Y AXIS=',Y_LIQ_OUT
      WRITE(6,*) 'LIQUID FLOWRATE AT Z AXIS=',Z_LIQ_OUT
      WRITE(6,*) 'LIQUID FLOWRATE AT PORT =',FLOWRATE_LIQ
      WRITE(6,*) 'GAS FLOWRATE AT PORT =',GAS_FLOWRATE
      WRITE(6,*) 'GAS VOLUME FRACTION IN OUTFLOW AT PORT(%)=',GAS_FRAC
      WRITE(6,*) 'LIQUID AREA WEIGHTED AVERAGE U', AREA_AV_U
      WRITE(6,*) 'LIQUID AREA WEIGHTED AVERAGE V', AREA_AV_V
      WRITE(6,*) 'LIQUID AREA WEIGHTED AVERAGE W', AREA_AV_W
      WRITE(6,*) 'LIQUID AREA WEIGHTED AVERAGE VOL.FRAC',AREA_AV_FRAC1
      WRITE(6,*) 'LIQUID AREA WEIGHTED AVERAGE K', AREA_AV_K
      WRITE(6,*) 'LIQUID AREA WEIGHTED AVERAGE EPSILON',AREA_AV_E
      WRITE(6,*) 'GAS AREA WEIGHTED AVERAGE U', AREA_GAS_U
      WRITE(6,*) 'GAS AREA WEIGHTED AVERAGE V', AREA_GAS_V
      WRITE(6,*) 'GAS AREA WEIGHTED AVERAGE W', AREA_GAS_W
      WRITE(6,*) 'GAS AREA WEIGHTED AVERAGE VOL.FRAC',AREA_AV_FRAC2

      WRITE(6,*) '##############################################'
      WRITE(6,*) ' '
      WRITE(6,*) 'K J            U            V            W'
       DO 703  K=1,KHEAD
        DO 702  J=1,IHEAD
         WRITE(6,*) K,J,TMP(K,J,1),TMP(K,J,2),TMP(K,J,3)
 702    CONTINUE
 703   CONTINUE

      CLOSE(6)

            WRITE(5,*) KHEAD,IHEAD

      DO 603 K=1,KHEAD
       DO 602  J=1,IHEAD
        DO 601  I=1,10
              WRITE(5,*) TMP(K,J,I)
 601  CONTINUE
 602  CONTINUE
 603  CONTINUE

      CLOSE(5)


C
C
C
      ENDIF
C
C+++++++++++++++  END OF USER AREA 5  ++++++++++++++++++++++++++++
C
      RETURN
      END

      SUBROUTINE USRBCS(VARBCS,VARAMB,A,B,C,ACND,BCND,CCND
     +                  ,IWGVEL,NDVWAL
     +                  ,FLOUT,NLABEL,NSTART,NEND,NCST,NCEN
     +                  ,U,V,W,P,VFRAC,DEN,VIS,TE,ED,RS,T,H,RF,SCAL
     +                  ,XP,YP,ZP,VOL,AREA,VPOR,ARPOR,WFACT,IPT
     +                  ,IBLK,IPVERT,IPNODN,IPFACN,IPNODF,IPNODB,IPFACB
     +                  ,WORK,IWORK,CWORK)
C
C*************************************************************
C
```

```
C USER ROUTINE TO SET REALS AT BOUNDARIES.
C
C  >>> IMPORTANT                                        <<<
C   >>>                                              <<<
C  >>> USERS MAY ONLY ADD OR ALTER PARTS OF THE SUBROUTINE WITHIN  <<<
C   >>> THE DESIGNATED USER AREAS                              <<<
C
C*************************************************************
C
C THIS SUBROUTINE IS CALLED BY THE FOLLOWING SUBROUTINE
C   CUSR  SRLIST
C
C*************************************************************
C  CREATED
C       30/11/88  ADB
C  MODIFIED
C       08/09/90  ADB  RESTRUCTURED FOR USER-FRIENDLINESS.
C       10/08/91  IRH FURTHER RESTRUCTURING ADD ACND BCND CCND
C       22/09/91  IRH CHANGE ICALL TO IUCALL + ADD /SPARM/
C       10/03/92  PHA  UPDATE CALLED BY COMMENT, ADD RF ARGUMENT,
C                 CHANGE LAST DIMENSION OF RS TO 6  AND IVERS TO 2
C       03/06/92  PHA  ADD PRECISION FLAG AND CHANGE IVERS TO 3
C       30/06/92  NSW  INCLUDE FLAG FOR CALLING BY ITERATION
C                 INSERT EXTRA COMMENTS
C       03/08/92  NSW  MODIFY DIMENSION STATEMENTS FOR VAX
C       21/12/92  CSH  INCREASE IVERS TO 4
C       02/08/93  NSW  INCORRECT AND MISLEADING COMMENT REMOVED
C       05/11/93  NSW  INDICATE USE OF FLOUT IN MULTIPHASE FLOWS
C       23/11/93  CSH  EXPLICITLY DIMENSION IPVERT ETC.
C       01/02/94  NSW  SET VARIABLE POINTERS IN WALL EXAMPLE.
C                 CHANGE FLOW3D TO CFDS-FLOW3D.
C                 MODIFY MULTIPHASE MASS FLOW BOUNDARY TREATMENT.
C       03/03/94  FHW  CORRECTION OF SPELLING MISTAKE
C       02/07/94  BAS  SLIDING GRIDS - ADD NEW ARGUMENT IWGVEL
C                  TO ALLOW VARIANTS OF TRANSIENT-GRID WALL BC
C                 CHANGE VERSION NUMBER TO 5
C       09/08/94  NSW  CORRECT SPELLING
C                 MOVE 'IF(IUSED.EQ.0) RETURN' OUT OF USER AREA
C       19/12/94  NSW  CHANGE FOR CFX-F3D
C       02/02/95  NSW  CHANGE COMMON /IMFBMP/
C       02/06/97  NSW  MAKE EXAMPLE MORE LOGICAL
C       02/07/97  NSW  UPDATE FOR CFX-4
C       08/09/98  NSW  CORRECT SIZE OF WALL ARRAY IN COMMENT
C
C*************************************************************
C
C  SUBROUTINE ARGUMENTS
C
C    VARBCS - REAL BOUNDARY CONDITIONS
C    VARAMB - AMBIENT VALUE OF VARIABLES
C     A      - COEFFICIENT IN WALL BOUNDARY CONDITION
C     B      - COEFFICIENT IN WALL BOUNDARY CONDITION
C     C     - COEFFICIENT IN WALL BOUNDARY CONDITION
C    ACND   - COEFFICIENT IN CONDUCTING WALL BOUNDARY CONDITION
C    BCND   - COEFFICIENT IN CONDUCTING WALL BOUNDARY CONDITION
C    CCND   - COEFFICIENT IN CONDUCTING WALL BOUNDARY CONDITION
C    IWGVEL - USAGE OF INPUT VELOCITIES (0 = AS IS,1 = ADD GRID MOTION)
C    NDVWAL - FIRST DIMENSION OF ARRAY IWGVEL
C    FLOUT  - MASS FLOW/FRACTIONAL MASS FLOW
C    NLABEL - NUMBER OF DISTINCT OUTLETS
```

```
C     NSTART - ARRAY POINTER
C     NEND   - ARRAY POINTER
C     NCST   - ARRAY POINTER
C     NCEN   - ARRAY POINTER
C      U      - U COMPONENT OF VELOCITY
C      V      - V COMPONENT OF VELOCITY
C      W      - W COMPONENT OF VELOCITY
C      P      - PRESSURE
C     VFRAC  - VOLUME FRACTION
C     DEN    - DENSITY OF FLUID
C      VIS    - VISCOSITY OF FLUID
C      TE     - TURBULENT KINETIC ENERGY
C      ED     - EPSILON
C      RS     - REYNOLD STRESSES
C      T      - TEMPERATURE
C      H      - ENTHALPY
C      RF     - REYNOLD FLUXES
C      SCAL   - SCALARS (THE FIRST 'NCONC' OF THESE ARE MASS FRACTIONS)
C      XP     - X COORDINATES OF CELL CENTRES
C      YP     - Y COORDINATES OF CELL CENTRES
C      ZP     - Z COORDINATES OF CELL CENTRES
C      VOL    - VOLUME OF CELLS
C      AREA   - AREA OF CELLS
C      VPOR   - POROUS VOLUME
C      ARPOR  - POROUS AREA
C      WFACT  - WEIGHT FACTORS
C
C      IPT    - 1D POINTER ARRAY
C      IBLK   - BLOCK SIZE INFORMATION
C      IPVERT - POINTER FROM CELL CENTERS TO 8 NEIGHBOURING VERTICES
C      IPNODN - POINTER FROM CELL CENTERS TO 6  NEIGHBOURING CELLS
C      IPFACN - POINTER FROM CELL CENTERS TO 6 NEIGHBOURING FACES
C      IPNODF - POINTER FROM CELL FACES TO 2  NEIGHBOURING CELL CENTERS
C      IPNODB - POINTER FROM BOUNDARY CENTERS TO CELL CENTERS
C      IPFACB - POINTER TO NODES FROM BOUNDARY FACES
C
C      WORK   - REAL WORKSPACE ARRAY
C      IWORK  - INTEGER WORKSPACE ARRAY
C      CWORK  - CHARACTER WORKSPACE ARRAY
C
C  SUBROUTINE ARGUMENTS PRECEDED WITH A '*' ARE ARGUMENTS THAT MUST
C  BE SET  BY THE USER IN THIS ROUTINE.
C
C  NOTE THAT OTHER DATA MAY BE OBTAINED FROM CFX-4 USING THE
C  ROUTINE GETADD, FOR FURTHER DETAILS SEE THE VERSION 4
C  USER MANUAL.
C
C*************************************************************
      LOGICAL LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
     +        ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
C
      CHARACTER*(*) CWORK
C
C+++++++++++++++++  USER AREA 1  +++++++++++++++++++++++++++++++
C---- AREA FOR USERS EXPLICITLY DECLARED VARIABLES
C
C+++++++++++++++++  END OF USER AREA 1  +++++++++++++++++++++++++
C
      COMMON
     + /ALL/    NBLOCK,NCELL,NBDRY,NNODE,NFACE,NVERT,NDIM
```

```
     + /ALLWRK/ NRWS,NIWS,NCWS,IWRFRE,IWIFRE,IWCFRE
     + /ADDIMS/ NPHASE,NSCAL,NVAR,NPROP
     +         ,NDVAR,NDPROP,NDXNN,NDGEOM,NDCOEF,NILIST,NRLIST,NTOPOL
     + /BCSOUT/ IFLOUT
     + /CHKUSR/ IVERS,IUCALL,IUSED
     + /DEVICE/ NREAD,NWRITE,NRDISK,NWDISK
     + /IDUM/   ILEN,JLEN
     + /IMFBMP/ IMFBMP,JMFBMP
     + /LOGIC/  LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
     +         ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
     + /MLTGRD/ MLEVEL,NLEVEL,ILEVEL
     + /SGLDBL/ IFLGPR,ICHKPR
     + /SPARM/  SMALL,SORMAX,NITER,INDPRI,MAXIT,NODREF,NODMON
     + /TRANSI/ NSTEP,KSTEP,MF,INCORE
     + /TRANSR/ TIME,DT,DTINVF,TPARM
     + /UBCSFL/ IUBCSF
C
C+++++++++++++++++  USER AREA 2 ++++++++++++++++++++++++++++++
C---- AREA FOR USERS TO DECLARE THEIR OWN COMMON BLOCKS
C     THESE SHOULD START WITH THE CHARACTERS 'UC' TO ENSURE
C     NO CONFLICT WITH NON-USER COMMON BLOCKS
C
C+++++++++++++++++  END OF USER AREA 2  ++++++++++++++++++++++++++
C
      DIMENSION
     + VARBCS(NVAR,NPHASE,NCELL+1:NNODE),VARAMB(NVAR,NPHASE)
     +,A(4+NSCAL,NPHASE,NSTART:*)
     +,B(4+NSCAL,NPHASE,NSTART:*),C(4+NSCAL,NPHASE,NSTART:*)
     +,FLOUT(*),ACND(NCST:*),BCND(NCST:*),CCND(NCST:*)
     +,IWGVEL(NDVWAL,NPHASE)
C
      DIMENSION
     + U(NNODE,NPHASE),V(NNODE,NPHASE),W(NNODE,NPHASE),P(NNODE,NPHASE)
     +,VFRAC(NNODE,NPHASE),DEN(NNODE,NPHASE),VIS(NNODE,NPHASE)
     +,TE(NNODE,NPHASE),ED(NNODE,NPHASE),RS(NNODE,NPHASE,6)
     +,T(NNODE,NPHASE),H(NNODE,NPHASE),RF(NNODE,NPHASE,4)
     +,SCAL(NNODE,NPHASE,NSCAL)
C
      DIMENSION
     + XP(NNODE),YP(NNODE),ZP(NNODE)
     +,VOL(NCELL),AREA(NFACE,3),VPOR(NCELL),ARPOR(NFACE,3),WFACT(NFACE)
      +,IPT(*),IBLK(5,NBLOCK)
     +,IPVERT(NCELL,8),IPNODN(NCELL,6),IPFACN(NCELL,6),IPNODF(NFACE,4)
     +,IPNODB(NBDRY,4),IPFACB(NBDRY)
      +,IWORK(*),WORK(*),CWORK(*)
C
C+++++++++++++++++  USER AREA 3 +++++++++++++++++++++++++++++++++
C---- AREA FOR USERS TO DIMENSION THEIR ARRAYS
C
C---- AREA FOR USERS TO DEFINE DATA STATEMENTS
C
C+++++++++++++++++  END OF USER AREA 3  ++++++++++++++++++++++++
C
C---- STATEMENT FUNCTION FOR ADDRESSING
       IP(I,J,K)=IPT((K-1)*ILEN*JLEN+(J-1)*ILEN+I)
C
C----VERSION NUMBER OF USER ROUTINE AND PRECISION FLAG
C
      IVERS=5
      ICHKPR = 1
```

```
C
C+++++++++++++++++  USER AREA 4 ++++++++++++++++++++++++++++++
C---- TO USE THIS USER ROUTINE FIRST SET IUSED=1
C     AND SET IUBCSF FLAG:
C    BOUNDARY CONDITIONS NOT CHANGING                        IUBCSF=0
C    BOUNDARY CONDITIONS CHANGING WITH ITERATION             IUBCSF=1
C    BOUNDARY CONDITIONS CHANGING WITH TIME                  IUBCSF=2
C    BOUNDARY CONDITIONS CHANGING WITH TIME AND ITERATION    IUBCSF=3
C
      IUSED=1
      IUBCSF=0

C
C+++++++++++++++++  END OF USER AREA 4  ++++++++++++++++++++++++
C
      IF (IUSED.EQ.0) RETURN
C
C---- FRONTEND CHECKING OF USER ROUTINE
      IF (IUCALL.EQ.0) RETURN
C
C+++++++++++++++++  USER AREA 5 +++++++++++++++++++++++++++++++
C
C---- AREA FOR SETTING VALUES AT INLETS, PRESSURE BOUNDARIES
C     AND OUTLETS. (NOTE THAT THE MASS FLOW AT OUTLETS IS
C    SPECIFIED IN USER AREA 7 )
C
      CALL GETVAR('USRBCS','P      ',KP)
C
      CALL IPREC('RIGHTPORT','PATCH','CENTRES',IPT,ILEN,JLEN,KLEN,
     +          CWORK,IWORK)
C
       ZMAX=0.0828
       ZMIN=0.0048
       PMAX=16717.43
       PMIN=11351.34
C  LOOP OVER PATCH
      DO 103  K  = 1, KLEN
         DO 102  J = 1, JLEN
            DO 101  I = 1, ILEN
C  USE STATEMENT FUNCTION IP TO GET ADDRESSES
            INODE = IP(I,J,K)
C  SET VARBCS
             F=(ZMAX-ZP(INODE))/(ZMAX-ZMIN)
            VARBCS(KP,1,INODE) = F*PMAX + (1.0-F)*PMIN
            VARBCS(KP,2,INODE) = F*PMAX + (1.0-F)*PMIN
  101       CONTINUE
  102    CONTINUE
  103  CONTINUE
C
C----END OF EXAMPLE
C
C+++++++++++++++++  END OF USER AREA 5  ++++++++++++++++++++++++++
C
     RETURN
      END
```

Appendix F. Subroutine for multiphase MUSIG model using output of nozzle simulation as input conditions

```
SUBROUTINE USRSRC(IEQN,ICALL,CNAME,CALIAS,AM,SP,SU,CONV
     +                ,U,V,W,P,VFRAC,DEN,VIS,TE,ED,RS,T,H,RF,SCAL
     +                ,XP,YP,ZP,VOL,AREA,VPOR,ARPOR,WFACT,IPT
     +                ,IBLK,IPVERT,IPNODN,IPFACN,IPNODF,IPNODB,IPFACB
     +                ,WORK,IWORK,CWORK)
C
     LOGICAL LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
     +       ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
C
      CHARACTER*(*) CWORK
     CHARACTER     CNAME*6,  CALIAS*24
C
C
     COMMON
      + /ALL/    NBLOCK,NCELL,NBDRY,NNODE,NFACE,NVERT,NDIM
      + /ALLWRK/ NRWS,NIWS,NCWS,IWRFRE,IWIFRE,IWCFRE
      + /ADDIMS/ NPHASE,NSCAL,NVAR,NPROP
      +         ,NDVAR,NDPROP,NDXNN,NDGEOM,NDCOEF,NILIST,NRLIST,NTOPOL
      + /CHKUSR/ IVERS,IUCALL,IUSED
      + /DEVICE/ NREAD,NWRITE,NRDISK,NWDISK
      + /IDUM/    ILEN,JLEN
      + /LOGIC/  LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
      +         ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
      + /MLTGRD/ MLEVEL,NLEVEL,ILEVEL
      + /SGLDBL/ IFLGPR,ICHKPR
      + /SPARM/  SMALL,SORMAX,NITER,INDPRI,MAXIT,NODREF,NODMON
      + /TRANSI/ NSTEP,KSTEP,MF,INCORE
      + /TRANSR/ TIME,DT,DTINVF,TPARM
C
C+++++++++++++++++  USER AREA 2 ++++++++++++++++++++++++++++
C
     COMMON /UCSURF/ JISURF,NISURF,JDSURF,NDSURF
     COMMON /UCINJR/ FINJ1,FINJ2,UGASIN,VGASIN,WGASIN
     COMMON /UCFLUX/ GASIN,GASOUT
C
C+++++++++++++++++  END OF USER AREA 2  +++++++++++++++++++++++
C
     DIMENSION AM(NCELL,6,NPHASE),SP(NCELL,NPHASE),SU(NCELL,NPHASE)
    +,CONV(NFACE,NPHASE)
C
     DIMENSION
    + U(NNODE,NPHASE),V(NNODE,NPHASE),W(NNODE,NPHASE),P(NNODE,NPHASE)
    +,VFRAC(NNODE,NPHASE),DEN(NNODE,NPHASE),VIS(NNODE,NPHASE)
    +,TE(NNODE,NPHASE),ED(NNODE,NPHASE),RS(NNODE,NPHASE,6)
    +,T(NNODE,NPHASE),H(NNODE,NPHASE),RF(NNODE,NPHASE,4)
    +,SCAL(NNODE,NPHASE,NSCAL)
C
     DIMENSION
    + XP(NNODE),YP(NNODE),ZP(NNODE)
    +,VOL(NCELL),AREA(NFACE,3),VPOR(NCELL),ARPOR(NFACE,3)
    +,WFACT(NFACE)
     +,IPT(*),IBLK(5,NBLOCK)
    +,IPVERT(NCELL,8),IPNODN(NCELL,6),IPFACN(NCELL,6),IPNODF(NFACE,4)
    +,IPNODB(NBDRY,4),IPFACB(NBDRY)
     +,IWORK(*),WORK(*),CWORK(*)
C
```

```
C---- STATEMENT FUNCTION FOR ADRESSING
      IP(I,J,K)=IPT((K-1)*ILEN*JLEN+(J-1)*ILEN+I)
C
C----VERSION NUMBER OF USER ROUTINE AND PRECISION FLAG
C
     IVERS=4
     ICHKPR = 1
     IUSED=1
      IF (IUSED.EQ.0) RETURN
C
C---- FRONTEND CHECKING OF USER ROUTINE
      IF (IUCALL.EQ.0) RETURN
C
C---- ADD TO SOURCE TERMS
      IF (ICALL.EQ.1) THEN
C
C+++++++++++++++++  USER AREA 5  +++++++++++++++++++++++++++++++++
C
C-----------------------------------------------------------------
C  VOLUME FRACTION [M]/[T]
C
     IF(CALIAS.EQ.'VOLUME FRACTION') THEN
C
C.......SINK AT TOP
        CALL USRDEGAS(CALIAS,SP,W,VFRAC,DEN,AREA,IWORK(JDSURF),NDSURF
     +            ,IPNODB,IPFACB,TOTAL)
       WRITE(NWRITE,*) ' VOLUME GASOUT=:',TOTAL
C       WRITE(NWRITE,*) ' VF :',TOTAL
C
     ENDIF
C
C-----------------------------------------------------------------
C PRESSURE [M]/[T]
C
     IF(CALIAS.EQ.'PRESSURE') THEN
C
       GASIN =  0.0
C
C.......SINK AT TOP
        CALL USRDEGAS(CALIAS,SU,W,VFRAC,DEN,AREA,IWORK(JDSURF),NDSURF
     +            ,IPNODB,IPFACB,TOTAL)
       GASOUT = TOTAL
       WRITE(NWRITE,*) ' PRESSURE GASOUT=:',TOTAL
C
     ENDIF
C
C-----------------------------------------------------------------
C  U VELOCITY [M][L]/[T]^2
C
     IF(CALIAS.EQ.'U VELOCITY') THEN
C
C
C.......SINK AT TOP
        CALL USRDEGAS(CALIAS,SP,W,VFRAC,DEN,AREA,IWORK(JDSURF),NDSURF
     +            ,IPNODB,IPFACB,TOTAL)

C.......SET MINIMUM VOID FRACTION
        VFMIN=1.0E-8
        DO 101  INODE=1,NNODE
         VFRAC(INODE,2)=MAX(VFRAC(INODE,2),VFMIN)
```

```
 101    CONTINUE
C
      ENDIF
C
C-----------------------------------------------------------------
C  V VELOCITY [M][L]/[T]^2
C
      IF(CALIAS.EQ.'V VELOCITY') THEN
C
C
C.......SINK AT TOP
         CALL USRDEGAS(CALIAS,SP,W,VFRAC,DEN,AREA,IWORK(JDSURF),NDSURF
     +            ,IPNODB,IPFACB,TOTAL)
C        WRITE(NWRITE,*) ' V :',TOTAL
C.......SET MINIMUM VOID FRACTION
         VFMIN=1.0E-8
         DO 201  INODE=1,NNODE
          VFRAC(INODE,2)=MAX(VFRAC(INODE,2),VFMIN)

 201    CONTINUE
C
      ENDIF
C
C-----------------------------------------------------------------
C  W VELOCITY [M][L]/[T]^2
C
      IF(CALIAS.EQ.'W VELOCITY') THEN
C
C.......SINK AT TOP
         CALL USRDEGAS(CALIAS,SP,W,VFRAC,DEN,AREA,IWORK(JDSURF),NDSURF
     +            ,IPNODB,IPFACB,TOTAL)
C        WRITE(NWRITE,*) ' W :',TOTAL
C
C.......SET MINIMUM VOID FRACTION
         VFMIN=1.0E-8
         DO 301  INODE=1,NNODE
          VFRAC(INODE,2)=MAX(VFRAC(INODE,2),VFMIN)

 301    CONTINUE
C
      ENDIF
C
C+++++++++++++++++  END OF USER AREA 5   ++++++++++++++++++++++++++
      ENDIF
C---- OVERWRITE SOURCE TERMS
       IF (ICALL.EQ.2) THEN
C
      ENDIF
      RETURN
       END
C
C
      SUBROUTINE USRDEGAS(CALIAS,SUP,W,VFRAC,DEN,AREA,IPTSRF,NDSURF
     +                  ,IPNODB,IPFACB,TOTAL)
C
C*************************************************************
C
C   UTILITY SUBROUTINE FOR USER-SUPPLIED FREE SURFACE DEGASSING
C
C*************************************************************
```

```
C
C   THIS SUBROUTINE IS CALLED BY THE FOLLOWING SUBROUTINES
C     USRSRC
C
C*************************************************************
      CHARACTER     CALIAS*24
C
C+++++++++++++++++  USER AREA 1 +++++++++++++++++++++++++++++++++++
C---- AREA FOR USERS EXPLICITLY DECLARED VARIABLES
C
C+++++++++++++++++  END OF USER AREA 1  ++++++++++++++++++++++++++++
C
      COMMON
       + /ALL/    NBLOCK,NCELL,NBDRY,NNODE,NFACE,NVERT,NDIM
       + /ALLWRK/ NRWS,NIWS,NCWS,IWRFRE,IWIFRE,IWCFRE
       + /ADDIMS/ NPHASE,NSCAL,NVAR,NPROP
      +         ,NDVAR,NDPROP,NDXNN,NDGEOM,NDCOEF,NILIST,NRLIST,NTOPOL
       + /DEVICE/ NREAD,NWRITE,NRDISK,NWDISK
C
C
      DIMENSION SUP(NCELL,NPHASE)
     +,W(NNODE,NPHASE),VFRAC(NNODE,NPHASE),DEN(NNODE,NPHASE)
     +,AREA(NFACE,3),IPTSRF(NDSURF),IPNODB(NBDRY,4),IPFACB(NBDRY)
C
C
C---- STATEMENT FUNCTION FOR ADRESSING
C+++++++++++++++++  USER AREA 5 +++++++++++++++++++++++++++++++++
C
C-----DEGASSING PHASE INDEX
       TOTAL=0.0
C-------------------------------------------------------------
C  VOLUME FRACTION
C
      IF(CALIAS.EQ.'VOLUME FRACTION') THEN
C
              DO 101 I=1,NDSURF
        INODE = IPTSRF(I)
        IBDRY = INODE-NCELL
        INODE1= IPNODB(IBDRY,1)
        IFACE = IPFACB(IBDRY)
        AREAM = SQRT ( AREA(IFACE,1)*AREA(IFACE,1)
      +                +AREA(IFACE,2)*AREA(IFACE,2)
      +                +AREA(IFACE,3)*AREA(IFACE,3) )
        FLUX  = DEN(INODE1,2)*
      +            AREAM*MAX(W(INODE1,2),0.0)
        SUP(INODE1,2)=SUP(INODE1,2)-FLUX
C             ENDIF
        TOTAL = TOTAL+FLUX
C        WRITE(NWRITE,*) ' FLUX=',FLUX
 101  CONTINUE
C
C-------------------------------------------------------------
C  OTHER VARIABLES
C
      ELSE
C
       DO 201 I=1,NDSURF
        INODE = IPTSRF(I)
        IBDRY = INODE-NCELL
        INODE1= IPNODB(IBDRY,1)
```

```
      IFACE = IPFACB(IBDRY)
      AREAM =  SQRT ( AREA(IFACE,1)*AREA(IFACE,1)
     +                +AREA(IFACE,2)*AREA(IFACE,2)
     +                +AREA(IFACE,3)*AREA(IFACE,3) )
      FLUX  = VFRAC(INODE1,2)*DEN(INODE1,2)
     +            *AREAM*MAX(W(INODE1,2),0.0)
      SUP(INODE1,2)=SUP(INODE1,2)-FLUX
            TOTAL = TOTAL+FLUX
C       WRITE(NWRITE,*) ' FLUX=',FLUX
 201  CONTINUE
C
     ENDIF
 401  CONTINUE
C
     RETURN
      END
C
C
     SUBROUTINE USRTRN(U,V,W,P,VFRAC,DEN,VIS,TE,ED,RS,T,H,RF,SCAL,
     +                 XP,YP,ZP,VOL,AREA,VPOR,ARPOR,WFACT,CONV,IPT,
     +                 IBLK,IPVERT,IPNODN,IPFACN,IPNODF,IPNODB,IPFACB,
     +                 WORK,IWORK,CWORK)
C
C*************************************************************
C
C  USER SUBROUTINE TO ALLOW USERS TO MODIFY OR MONITOR THE SOLUTION AT
C   THE END OF EACH TIME STEP
C   THIS SUBROUTINE IS CALLED BEFORE THE START OF THE RUN AS WELL AS AT
C   THE END OF EACH TIME STEP
C
C*************************************************************
C
C   THIS SUBROUTINE IS CALLED BY THE FOLLOWING SUBROUTINES
C      CUSR  TRNMOD
C
C*************************************************************
C
C*************************************************************
C
     LOGICAL LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
     +       ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
C
     CHARACTER*(*) CWORK
C
C+++++++++++++++  USER AREA 1 +++++++++++++++++++++++++++++++
C---- AREA FOR USERS EXPLICITLY DECLARED VARIABLES
C
C+++++++++++++++  END OF USER AREA 1  +++++++++++++++++++++++
C
     COMMON
      + /ALL/    NBLOCK,NCELL,NBDRY,NNODE,NFACE,NVERT,NDIM
      + /ALLWRK/ NRWS,NIWS,NCWS,IWRFRE,IWIFRE,IWCFRE
      + /ADDIMS/ NPHASE,NSCAL,NVAR,NPROP
     +         ,NDVAR,NDPROP,NDXNN,NDGEOM,NDCOEF,NILIST,NRLIST,NTOPOL
     + /CHKUSR/ IVERS,IUCALL,IUSED
      + /CONC/   NCONC
     + /DEVICE/ NREAD,NWRITE,NRDISK,NWDISK
     + /IDUM/    ILEN,JLEN
      + /LOGIC/  LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
     +         ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
```

```
     + /MLTGRD/ MLEVEL,NLEVEL,ILEVEL
     + /SGLDBL/ IFLGPR,ICHKPR
     + /SPARM/  SMALL,SORMAX,NITER,INDPRI,MAXIT,NODREF,NODMON
     + /TIMUSR/ DTUSR
     + /TRANSI/ NSTEP,KSTEP,MF,INCORE
     + /TRANSR/ TIME,DT,DTINVF,TPARM
C
C+++++++++++++++  USER AREA 2 +++++++++++++++++++++++++++++++++
C---- AREA FOR USERS TO DECLARE THEIR OWN COMMON BLOCKS
C     THESE SHOULD START WITH THE CHARACTERS 'UC' TO ENSURE
C     NO CONFLICT WITH NON-USER COMMON BLOCKS
C
     COMMON /UCSURF/ JISURF,NISURF,JDSURF,NDSURF
     COMMON /UCINJR/ FINJ1,FINJ2,UGASIN,VGASIN,WGASIN
     COMMON /UCFLUX/ GASIN,GASOUT
C
C+++++++++++++++  END OF USER AREA 2  ++++++++++++++++++++++++++
C
     DIMENSION
    + U(NNODE,NPHASE),V(NNODE,NPHASE),W(NNODE,NPHASE),P(NNODE,NPHASE)
    +,VFRAC(NNODE,NPHASE),DEN(NNODE,NPHASE),VIS(NNODE,NPHASE)
    +,TE(NNODE,NPHASE),ED(NNODE,NPHASE),RS(NNODE,NPHASE,6)
    +,T(NNODE,NPHASE),H(NNODE,NPHASE),RF(NNODE,NPHASE,4)
    +,SCAL(NNODE,NPHASE,NSCAL)
     DIMENSION
    + XP(NNODE),YP(NNODE),ZP(NNODE)
    +,VOL(NCELL),AREA(NFACE,3),VPOR(NCELL),ARPOR(NFACE,3)
    +,WFACT(NFACE),CONV(NFACE,NPHASE)
     +,IPT(*),IBLK(5,NBLOCK)
    +,IPVERT(NCELL,8),IPNODN(NCELL,6),IPFACN(NCELL,6),IPNODF(NFACE,4)
    +,IPNODB(NBDRY,4),IPFACB(NBDRY)
     +,IWORK(*),WORK(*),CWORK(*)
C
C
C---- STATEMENT FUNCTION FOR ADDRESSING
      IP(I,J,K)=IPT((K-1)*ILEN*JLEN+(J-1)*ILEN+I)
C
C----VERSION NUMBER OF USER ROUTINE AND PRECISION FLAG
C
     IVERS=3
     ICHKPR = 1
C
C+++++++++++++++  USER AREA 4 ++++++++++++++++++++++++++++
C---- TO USE THIS USER ROUTINE FIRST SET IUSED=1
C
      IUSED=1
      IF (IUSED.EQ.0) RETURN
C+++++++++++++++  END OF USER AREA 4  +++++++++++++++++++++
C
C---- FRONTEND CHECKING OF USER ROUTINE
      IF (IUCALL.EQ.0) RETURN
C
C+++++++++++++++  USER AREA 5 ++++++++++++++++++++++++++++
C
     IF (KSTEP.EQ.0) THEN
C
C-----INJECTION BOUNDARY LOCATIONS
C
      CALL IPALL('NOZZLE INLET','*','PATCH','CENTRES',IPT,NPT,
     +          CWORK,IWORK)
```

```
C
      NISURF=NPT
C     SET INJECTION BOUNDARY LIST INTO INTERGER WORK SPACE
      CALL SETPER('USRTRN','IWORK ','ISURF ',NISURF,JISURF)
C
      DO 101 I=1,NISURF
       IWORK(JISURF+I-1)=IPT(I)
 101  CONTINUE
      WRITE(NWRITE,*) '   '
      WRITE(NWRITE,*) '********   USRTRN - INLET **********'
      WRITE(NWRITE,*) '  NISURF=',NISURF,' JISURF=',JISURF
      WRITE(NWRITE,*) '  IWORK(JISURF)=',IWORK(JISURF)
      WRITE(NWRITE,*) '  IWORK(JISURF+NISURF-1)=',IWORK(JISURF+NISURF-1)
C
C-----INJECTION RATES: MASS FLUX IN kg/s,  VELOCITY IN  m/s
      FGASIN =  0
      UGASIN =  0
      VGASIN =  0
      WGASIN =  0
C
C-----DETERMINE TOTAL AREA OF INLET BOUNDARY AND MASS FLUX PER AREA
      TOTALA =  0.0
      DO 201 I=1,NISURF
       INODE = IWORK(JISURF+I-1)
       IBDRY =  INODE-NCELL
       IFACE = IPFACB(IBDRY)
       AREAM =  SQRT ( AREA(IFACE,1)*AREA(IFACE,1)
     +                +AREA(IFACE,2)*AREA(IFACE,2)
     +                +AREA(IFACE,3)*AREA(IFACE,3) )
        TOTALA = TOTALA+AREAM
 201  CONTINUE
C
      WRITE(NWRITE,*) '  TOTAL AREA OF INLET =',TOTALA
      FINJ2 = FGASIN/TOTALA
C
C-----DEGASSING BOUNDARY LOCATIONS
C
      CALL IPALL('TOP SURFACE','*','PATCH','CENTRES',IPT,NPT,
     +           CWORK,IWORK)
C
      NDSURF=NPT
C     SET DEGASSING BOUNDARY LIST INTO INTERGER WORK SPACE
      CALL SETPER('USRTRN','IWORK ','DSURF ',NDSURF,JDSURF)
C
      DO 401 I=1,NDSURF
       IWORK(JDSURF+I-1)=IPT(I)
 401  CONTINUE
      WRITE(NWRITE,*) '   '
      WRITE(NWRITE,*) '********   USRTRN - OUTLET *********'
      WRITE(NWRITE,*) ' NDSURF=',NDSURF,' JDSURF=',JDSURF
      WRITE(NWRITE,*) ' IWORK(JDSURF)=',IWORK(JDSURF)
      WRITE(NWRITE,*) ' IWORK(JDSURF+NDSURF-1)=',IWORK(JDSURF+NDSURF-1)
C
      ENDIF
C
C-----------------------------------------------------------------
C     END OPERATION
C
      IF(KSTEP.EQ.NSTEP) THEN
C
```

```
C=====CHECK MASS BALANCE
C
C.....GAS FLOW RATE FROM SOURCE IS SET IN USRSRC

     INFLOE_SOURCE=GASIN

C.....INLET

      CALL IPALL('NOZZLE INLET','*','PATCH','CENTRES',IPT,NPT,
     +          CWORK,IWORK)
C
      GASIN=0.0
      GASIN_2=0.0
      WATIN =  0.0
C.....GASIN INITIALISED IN USRSRC
      DO 501 I=1,NPT
        INODE = IPT(I)
        IBDRY = INODE-NCELL
        IFACE = IPFACB(IBDRY)
        WATIN = WATIN+CONV(IFACE,1)
        GASIN_2 = GASIN_2+CONV(IFACE,2)
 501  CONTINUE

      GASIN=GASIN_2
      SUM=0.0
     SUM=INFLOE_SOURCE+GASIN
     WRITE(NWRITE,*) ' '

     WRITE(NWRITE,*) '&&&&&&&  SHOWING OF OUTPUT OF INFLOW DATA &&&&&'
     WRITE(NWRITE,*) ' '
     WRITE(NWRITE,*) 'GAS INFLOW RATE FROM SOURCE=',INFLOE_SOURCE
     WRITE(NWRITE,*) 'GAS INFLOW RATE FROM INLET=',GASIN
     WRITE(NWRITE,*) 'THE SUM OF GAS INFLOW RATE FROM SOURCE AND
    +INLET=',SUM
     WRITE(NWRITE,*) ' '

     WRITE(NWRITE,*) ' '
C
C.....OUTLET
     GASOUT_SINK=GASOUT
     WRITE(NWRITE,*) '&&&&&&&  SHOWING OF OUTPUT OF OUTFLOW DATA &&&&&'
     WRITE(NWRITE,*) ' '
     WRITE(NWRITE,*) 'GAS OUTFLOW RATE FROM TOP SURFACE=', GASOUT_SINK

     WRITE(NWRITE,*) ' '
      CALL IPALL('OUTLET','*','PATCH','CENTRES',IPT,NPT,
     +          CWORK,IWORK)
C
     GASOUT =  0.0
     GASOUT_2 = 0.0
     WATOUT =  0.0
C.....GASOUT INITIALISED IN USRSRC
      DO 601 I=1,NPT
        INODE = IPT(I)
        IBDRY = INODE-NCELL
        IFACE = IPFACB(IBDRY)
        WATOUT = WATOUT+CONV(IFACE,1)
        GASOUT_2 = GASOUT_2+CONV(IFACE,2)
 601  CONTINUE
     GASOUT =GASOUT_2
```

```
      SUM1=0.0
      SUM1= GASOUT_SINK+GASOUT

      GFLOWRATE_IN=SUM/0.559
      GFLOWRATE_OUT=SUM1/0.559
      WFLOWRATE_IN=WATIN/7020
      WFLOWRATE_OUT=WATOUT/7020

      VOLFRAC = GFLOWRATE_IN/(GFLOWRATE_IN+WFLOWRATE_IN)

      WRITE(NWRITE,*) 'GAS OUTFLOW RATE FROM BOTTUM=', GASOUT
      WRITE(NWRITE,*) 'THE SUM OF GAS OUTFLOW RATE FROM TOP SURFACE AND
     + BOTTOM=',SUM1
      WRITE(NWRITE,*) ' '
      WRITE(NWRITE,*) ' '
C
      WRITE(NWRITE,*) '  ***  MASS BALANCE ***'
      WRITE(NWRITE,*) ' '
      WRITE(NWRITE,*) ' AIR'
      WRITE(NWRITE,*) ' TOTAL   INFLOW   (m^3/s)  =',GFLOWRATE_IN
      WRITE(NWRITE,*) ' TOTAL   OUTFLOW (m^3/s)   =',GFLOWRATE_OUT
      GASERR=(SUM1-SUM)*100.0/SUM
      WRITE(NWRITE,*) '    % ERROR         =',GASERR
      WRITE(NWRITE,*) ' '
      WRITE(NWRITE,*) ' WATER'
      WRITE(NWRITE,*) '   INFLOW   (m^3/s)  =',WFLOWRATE_IN
      WRITE(NWRITE,*) '   OUTFLOW (m^3/s)   =',WFLOWRATE_OUT
       WATERR=(WATOUT+WATIN)*100.0/WATIN
      WRITE(NWRITE,*) '    % ERROR         =',WATERR
      WRITE(NWRITE,*) ' '
C
      WRITE(NWRITE,*) 'Volume  Fraction  of  gas =',VOLFRAC
C
C.....SET IPLT=1 TO CLEAN UP VELOCITY FOR PLOTTING
C*****NOTE  THAT THIS OPERATION WILL SLIGHTLY ALTER THE VELOCITY FIELD
C     HENCE THE DUMP FILE GENERATED SHOULD BE USED FOR PLOTTING ONLY
C     NOT FOR RESTART.
C
        IPLT=0
C
C.....FILTER OUT VELOCITY WITH SMALL VOLUME FRACTION
        IF(IPLT.EQ.1) THEN
           DO 701  INODE=1,NNODE
            FILTER=1.0-EXP(-3.0*VFRAC(INODE,2)/1.E-4)
           U(INODE,2)=U(INODE,2)*FILTER
           V(INODE,2)=V(INODE,2)*FILTER
           W(INODE,2)=W(INODE,2)*FILTER
 701       CONTINUE
C
        ENDIF
C
      ENDIF
C
C+++++++++++++++  END OF USER AREA 5   ++++++++++++++++++++++++++++
C
      RETURN
      END

      SUBROUTINE USRBCS(VARBCS,VARAMB,A,B,C,ACND,BCND,CCND
     +                  ,IWGVEL,NDVWAL
```

```
     +                ,FLOUT,NLABEL,NSTART,NEND,NCST,NCEN
     +                ,U,V,W,P,VFRAC,DEN,VIS,TE,ED,RS,T,H,RF,SCAL
     +                ,XP,YP,ZP,VOL,AREA,VPOR,ARPOR,WFACT,IPT
     +                ,IBLK,IPVERT,IPNODN,IPFACN,IPNODF,IPNODB,IPFACB
     +                ,WORK,IWORK,CWORK)
C
C*************************************************************
C
C USER ROUTINE TO SET REALS AT BOUNDARIES.
C
C   >>> IMPORTANT                                    <<<
C    >>>                                          <<<
C   >>> USERS MAY ONLY ADD OR ALTER PARTS OF THE SUBROUTINE WITHIN   <<<
C   >>> THE DESIGNATED USER AREAS                              <<<
C
C*************************************************************
C
C THIS SUBROUTINE IS CALLED BY THE FOLLOWING SUBROUTINE
C   CUSR  SRLIST
C
C*************************************************************
C  CREATED
C        30/11/88  ADB
C  MODIFIED
C        08/09/90  ADB  RESTRUCTURED FOR USER-FRIENDLINESS.
C        10/08/91  IRH  FURTHER RESTRUCTURING ADD ACND BCND CCND
C        22/09/91  IRH  CHANGE ICALL TO IUCALL + ADD /SPARM/
C        10/03/92  PHA  UPDATE CALLED BY COMMENT, ADD RF ARGUMENT,
C                  CHANGE LAST DIMENSION OF RS TO 6  AND IVERS TO 2
C        03/06/92  PHA  ADD PRECISION FLAG AND CHANGE IVERS TO 3
C        30/06/92  NSW  INCLUDE FLAG FOR CALLING BY ITERATION
C                  INSERT EXTRA COMMENTS
C        03/08/92  NSW  MODIFY DIMENSION STATEMENTS FOR VAX
C        21/12/92  CSH  INCREASE IVERS TO 4
C        02/08/93  NSW  INCORRECT AND MISLEADING COMMENT REMOVED
C        05/11/93  NSW  INDICATE USE OF FLOUT IN MULTIPHASE FLOWS
C        23/11/93  CSH  EXPLICITLY DIMENSION IPVERT ETC.
C        01/02/94  NSW  SET VARIABLE POINTERS IN WALL EXAMPLE.
C                  CHANGE FLOW3D TO CFDS-FLOW3D.
C                  MODIFY MULTIPHASE MASS FLOW BOUNDARY TREATMENT.
C        03/03/94  FHW  CORRECTION OF SPELLING MISTAKE
C        02/07/94  BAS  SLIDING GRIDS - ADD NEW ARGUMENT IWGVEL
C                   TO ALLOW VARIANTS OF TRANSIENT-GRID WALL BC
C                  CHANGE VERSION NUMBER TO 5
C        09/08/94  NSW  CORRECT SPELLING
C                  MOVE 'IF(IUSED.EQ.0) RETURN' OUT OF USER AREA
C        19/12/94  NSW  CHANGE FOR CFX-F3D
C        02/02/95  NSW  CHANGE COMMON /IMFBMP/
C        02/06/97  NSW  MAKE EXAMPLE MORE LOGICAL
C        02/07/97  NSW  UPDATE FOR CFX-4
C        08/09/98  NSW  CORRECT SIZE OF WALL ARRAY IN COMMENT
C
C*************************************************************
C
C  SUBROUTINE ARGUMENTS
C
C    VARBCS - REAL BOUNDARY CONDITIONS
C    VARAMB - AMBIENT VALUE OF VARIABLES
C     A      - COEFFICIENT IN WALL BOUNDARY CONDITION
C     B      - COEFFICIENT IN WALL BOUNDARY CONDITION
```

```
C     C     - COEFFICIENT IN WALL BOUNDARY CONDITION
C     ACND   - COEFFICIENT IN CONDUCTING WALL BOUNDARY CONDITION
C     BCND   - COEFFICIENT IN CONDUCTING WALL BOUNDARY CONDITION
C     CCND   - COEFFICIENT IN CONDUCTING WALL BOUNDARY CONDITION
C     IWGVEL - USAGE OF INPUT VELOCITIES (0 = AS IS,1 = ADD GRID MOTION)
C     NDVWAL - FIRST DIMENSION OF ARRAY IWGVEL
C     FLOUT  - MASS FLOW/FRACTIONAL MASS FLOW
C     NLABEL - NUMBER OF DISTINCT OUTLETS
C     NSTART - ARRAY POINTER
C     NEND   - ARRAY POINTER
C     NCST   - ARRAY POINTER
C     NCEN   - ARRAY POINTER
C     U      - U COMPONENT OF VELOCITY
C     V      - V COMPONENT OF VELOCITY
C     W      - W COMPONENT OF VELOCITY
C     P      - PRESSURE
C     VFRAC  - VOLUME FRACTION
C     DEN    - DENSITY OF FLUID
C     VIS    - VISCOSITY OF FLUID
C     TE     - TURBULENT KINETIC ENERGY
C     ED     - EPSILON
C     RS     - REYNOLD STRESSES
C     T      - TEMPERATURE
C     H      - ENTHALPY
C     RF     - REYNOLD FLUXES
C     SCAL   - SCALARS (THE FIRST 'NCONC' OF THESE ARE MASS FRACTIONS)
C     XP     - X COORDINATES OF CELL CENTRES
C     YP     - Y COORDINATES OF CELL CENTRES
C     ZP     - Z COORDINATES OF CELL CENTRES
C     VOL    - VOLUME OF CELLS
C     AREA   - AREA OF CELLS
C     VPOR   - POROUS VOLUME
C     ARPOR  - POROUS AREA
C     WFACT  - WEIGHT FACTORS
C
C     IPT    - 1D POINTER ARRAY
C     IBLK   - BLOCK SIZE INFORMATION
C     IPVERT - POINTER FROM CELL CENTERS TO 8 NEIGHBOURING VERTICES
C     IPNODN - POINTER FROM CELL CENTERS TO 6 NEIGHBOURING CELLS
C     IPFACN - POINTER FROM CELL CENTERS TO 6 NEIGHBOURING FACES
C     IPNODF - POINTER FROM CELL FACES TO 2 NEIGHBOURING CELL CENTERS
C     IPNODB - POINTER FROM BOUNDARY CENTERS TO CELL CENTERS
C     IPFACB - POINTER TO NODES FROM BOUNDARY FACES
C
C     WORK   - REAL WORKSPACE ARRAY
C     IWORK  - INTEGER WORKSPACE ARRAY
C     CWORK  - CHARACTER WORKSPACE ARRAY
C
C  SUBROUTINE ARGUMENTS PRECEDED WITH A '*' ARE ARGUMENTS THAT MUST
C  BE SET BY THE USER IN THIS ROUTINE.
C
C  NOTE THAT OTHER DATA MAY BE OBTAINED FROM CFX-4 USING THE
C  ROUTINE GETADD, FOR FURTHER DETAILS SEE THE VERSION 4
C  USER MANUAL.
C
C*************************************************************
      LOGICAL LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
     +        ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
C
      CHARACTER*(*) CWORK
```

```
C
C+++++++++++++++++  USER AREA 1 +++++++++++++++++++++++++++++++++++
C---- AREA FOR USERS EXPLICITLY DECLARED VARIABLES
C
C+++++++++++++++++  END OF USER AREA 1  ++++++++++++++++++++++++++
C
      COMMON
       + /ALL/    NBLOCK,NCELL,NBDRY,NNODE,NFACE,NVERT,NDIM
       + /ALLWRK/ NRWS,NIWS,NCWS,IWRFRE,IWIFRE,IWCFRE
       + /ADDIMS/ NPHASE,NSCAL,NVAR,NPROP
       +          ,NDVAR,NDPROP,NDXNN,NDGEOM,NDCOEF,NILIST,NRLIST,NTOPOL
       + /BCSOUT/ IFLOUT
       + /CHKUSR/ IVERS,IUCALL,IUSED
       + /DEVICE/ NREAD,NWRITE,NRDISK,NWDISK
       + /IDUM/   ILEN,JLEN
       + /IMFBMP/ IMFBMP,JMFBMP
       + /LOGIC/  LDEN,LVIS,LTURB,LTEMP,LBUOY,LSCAL,LCOMP
       +          ,LRECT,LCYN,LAXIS,LPOROS,LTRANS
       + /MLTGRD/ MLEVEL,NLEVEL,ILEVEL
       + /SGLDBL/ IFLGPR,ICHKPR
       + /SPARM/  SMALL,SORMAX,NITER,INDPRI,MAXIT,NODREF,NODMON
       + /TRANSI/ NSTEP,KSTEP,MF,INCORE
       + /TRANSR/ TIME,DT,DTINVF,TPARM
       + /UBCSFL/ IUBCSF
C
C+++++++++++++++++  USER AREA 2 +++++++++++++++++++++++++++++++++++
C---- AREA FOR USERS TO DECLARE THEIR OWN COMMON BLOCKS
C     THESE SHOULD START WITH THE CHARACTERS 'UC' TO ENSURE
C     NO CONFLICT WITH NON-USER COMMON BLOCKS
C
C+++++++++++++++++  END OF USER AREA 2  ++++++++++++++++++++++++++
C
      DIMENSION
     + VARBCS(NVAR,NPHASE,NCELL+1:NNODE),VARAMB(NVAR,NPHASE)
     +,A(4+NSCAL,NPHASE,NSTART:*)
     +,B(4+NSCAL,NPHASE,NSTART:*),C(4+NSCAL,NPHASE,NSTART:*)
     +,FLOUT(*),ACND(NCST:*),BCND(NCST:*),CCND(NCST:*)
     +,IWGVEL(NDVWAL,NPHASE)
C
      DIMENSION
     + U(NNODE,NPHASE),V(NNODE,NPHASE),W(NNODE,NPHASE),P(NNODE,NPHASE)
     +,VFRAC(NNODE,NPHASE),DEN(NNODE,NPHASE),VIS(NNODE,NPHASE)
     +,TE(NNODE,NPHASE),ED(NNODE,NPHASE),RS(NNODE,NPHASE,6)
     +,T(NNODE,NPHASE),H(NNODE,NPHASE),RF(NNODE,NPHASE,4)
     +,SCAL(NNODE,NPHASE,NSCAL)
C
      DIMENSION
     + XP(NNODE),YP(NNODE),ZP(NNODE)
     +,VOL(NCELL),AREA(NFACE,3),VPOR(NCELL),ARPOR(NFACE,3),WFACT(NFACE)
      +,IPT(*),IBLK(5,NBLOCK)
     +,IPVERT(NCELL,8),IPNODN(NCELL,6),IPFACN(NCELL,6),IPNODF(NFACE,4)
     +,IPNODB(NBDRY,4),IPFACB(NBDRY)
       +,IWORK(*),WORK(*),CWORK(*),TMP(20,20,11)
C
C+++++++++++++++++  USER AREA 3 +++++++++++++++++++++++++++++++++++
C---- AREA FOR USERS TO DIMENSION THEIR ARRAYS
C
C---- AREA FOR USERS TO DEFINE DATA STATEMENTS
C
C+++++++++++++++++  END OF USER AREA 3  ++++++++++++++++++++++++++
```

```
C
C---- STATEMENT FUNCTION FOR ADDRESSING
      IP(I,J,K)=IPT((K-1)*ILEN*JLEN+(J-1)*ILEN+I)
C
C----VERSION NUMBER OF USER ROUTINE AND PRECISION FLAG
C
     IVERS=5
     ICHKPR = 1
C
C+++++++++++++++++  USER AREA 4 ++++++++++++++++++++++++++++++++++++++++++
C---- TO USE THIS USER ROUTINE FIRST SET IUSED=1
C     AND SET IUBCSF FLAG:
C    BOUNDARY CONDITIONS NOT CHANGING                        IUBCSF=0
C    BOUNDARY CONDITIONS CHANGING WITH ITERATION             IUBCSF=1
C    BOUNDARY CONDITIONS CHANGING WITH TIME                  IUBCSF=2
C    BOUNDARY CONDITIONS CHANGING WITH TIME AND ITERATION    IUBCSF=3
C
      IUBCSF=0
      IUSED=1
C
C+++++++++++++++++  END OF USER AREA 4  +++++++++++++++++++++++++++++++++++
C
      IF (IUSED.EQ.0) RETURN
C
C---- FRONTEND CHECKING OF USER ROUTINE
     IF (IUCALL.EQ.0) RETURN
C
C+++++++++++++++++  USER AREA 5 +++++++++++++++++++++++++++++++++++
C
C---- AREA FOR SETTING VALUES AT INLETS, PRESSURE BOUNDARIES
C     AND OUTLETS. (NOTE THAT THE MASS FLOW AT OUTLETS IS
C    SPECIFIED IN USER AREA 7 )

C
      OPEN(5,FILE='DATA1.DAT',STATUS='OLD',FORM='FORMATTED')
       READ(5,*) KHEAD,JHEAD

      DO 13 K = 1, KHEAD
        DO 12 J = 1, JHEAD
          DO 11 I = 1, 1 0

                 READ(5,*) TMP(K,J,I)
 11        CONTINUE
 12       CONTINUE
 13     CONTINUE

      CALL GETVAR('USRBCS','U      ',IU)
      CALL GETVAR('USRBCS','V      ',IV)
      CALL GETVAR('USRBCS','W      ',IW)
      CALL GETVAR('USRBCS','VFRAC      ',IVFRAC)
      CALL GETVAR('USRBCS','TE      ',ITE)
      CALL GETVAR('USRBCS','ED      ',IED)
C
C USE IPREC TO FIND ADDRESSES

      CALL IPREC('NOZZLE INLET','PATCH','CENTRES',IPT,ILEN,JLEN,KLEN,
     +           CWORK,IWORK)
C
C LOOP OVER PATCH
      DO 103 K = 1, KLEN
```

```
      DO 102  J = 1, JLEN
        DO 101  I = 1, ILEN
       INODE = IP(I,J,K)
         JSCALE = JHEAD*J/JLEN
              KSCALE = KHEAD*K/KLEN
             VARBCS(IU,1,INODE) = TMP(KSCALE,JSCALE,1)
         VARBCS(IV,1,INODE) = TMP(KSCALE,JSCALE,2)
         VARBCS(IW,1,INODE) = TMP(KSCALE,JSCALE,3)
         VARBCS(ITE,1,INODE) = TMP(KSCALE,JSCALE,4)
         VARBCS(IED,1,INODE) = TMP(KSCALE,JSCALE,5)
         VARBCS(IU,2,INODE) = TMP(KSCALE,JSCALE,6)
         VARBCS(IV,2,INODE) = TMP(KSCALE,JSCALE,7)
              VARBCS(IW,2,INODE) = TMP(KSCALE,JSCALE,8)

              XFRAC=TMP(KSCALE,JSCALE,9)
         IF(XFRAC.GE.1.0) THEN
           VARBCS(IVFRAC,1,INODE) = 1.0
         ENDIF
         IF(XFRAC.LT.1.0) THEN
           VARBCS(IVFRAC,1,INODE) = XFRAC
         ENDIF

              VARBCS(IVFRAC,2,INODE) = 1-VARBCS(IVFRAC,1,INODE)



  101        CONTINUE
  102    CONTINUE
  103  CONTINUE

C+++++++++++++++++   END OF USER AREA 5
C
RETURN
      END
```

Reference

1. F. M. Najjar, B. G. Thomas and D. E. Hershey, “Turbulent flow simulations in bifurcated nozzles: effects of design and casting operation”, Metallurgical Transactions B, 26B(4), 1995, pp749-765.
2. H. Nakato, M. Ozawa, K. Habu and T. Emi, “Factors affecting the formation of shell and longitudinal cracks in mold during high speed continuous casting of slabs”, Trans.Iron&Steel Inst. Japan, vol.24(11), 1984, pp957-965.
3. R. McDavid and B. G. Thomas, “Flow and thermal behavior of the top-surface flux/powder layer in continuous casting molds”, Metallurgical Transactions B, 27B(4), 1996, pp672-685.
4. R. Sussman, M. Burns, X. Huang and B.G. Thomas, “Inclusion particle behavior in a continuous slab casting mold”, 10th process Technology Conference Proceedings, Toronto, Ontario, April 5-8, 1992, Iron and Steel Society, Warrendale, PA, pp291-304.
5. D. Creech, M.S. Thesis, University of Illinois at Urbana-Champaign, 1999.
6. B. G. Thomas, A. Dennisov and H. Bai, “Behavior of argon bubbles during continuous casting of steel”, ISS 80th Steelmaking Conference, Chicago, IL, ISS, April 13-16, 1997.
7. X. Huang, B. G. Thomas and F. M. Najjar, “Modeling superheat removal during continuous casting of steel slabs”, Metallurgical Trans.B, June 1992, Vol.23B, pp339-356.

8. M. Iguchi and N. Kasai, “Water model study of Horizontal molten steel-Ar two-phase jet in a continuously casting mold”, Met&Mat Trans.B, June 2000, Vol.31B, p453-460.
9. CFX-4.2: Solver Manual, AEA Technology plc, 1997.
10. CFX-4.3: Solver Manual, AEA Technology plc, 1999.
11. H. Luo and H. Svendsen, “Experimental determination and modeling of bubble size distributions in bubble columns”, AICHE Journal, 1996, May, Vol.42, pp1225-1233.
12. M. Prince and H. Blanch, AICHE Journal, “Bubble coalescence and break-up in air-sparged bubble columns”, 1990, October, Vol. 36, pp1485-1499.
13. R. Kirkparick and M. Lockett, “The influence of approach velocity on bubble coalescence”, Chem. Eng. Sci., 1974, Vol. 29, pp2363-2371.
14. W. Kim and K. Lee, “Coalescence behavior of two bubbles in stagnant liquids”, J. of Chem. Eng. Of Japan, 1987, Vol.20, pp448-453.
15. D. Stone, M.S. Thesis, University of Illinois at Urbana-Champaign, 2000.
16. H. Bai, PhD. Thesis, University of Illinois at Urbana-Champaign, 2000.
17. B. G. Thomas and X. Huang, “Effect of argon gas on fluid flow in a continuous slab casting mold”, 76th Steelmaking Conference, Dallas, TX, March 28-31, 1993, Iron and Steel Society, Warrendale, PA, Vol.76, 1993, pp. 273-289.
18. J. Herbertson, Q.L.He, P. J. Flint and R. B. Mahapatra, “Modeling of metal delivery to continuous casting moulds”, Steelmaking Conference Proceedings, 1991, pp171-185.
19. L. Yu, Thesis, University of Illinois at Urbana-Champaign, 2000.
20. M. Langeneckert, Thesis, University of Illinois at Urbana-Champaign, 2000.

21. B. G. Thomas, R. O'Malley, T. Shi, Y. Meng, D. Creech and D. Stone, "Modeling of casting, welding and advanced solidification process IX", Archen, Germany, August, 2000, pp769-776.

22. N. Bessho, R. Yoda and H. Yamasaki, "Numerical analysis of fluid flow in the continuous casting mold by a bubble dispersion model", Iron steelmaker, 1991, Vol.18(4), pp39-44.

23. H. Bai and B.G. Thomas, Annual Report, Continuous Casting Consortium, 1998.

24. Y. Shang, Annual Report, Continuous Casting Consortium, 1996.

25. L. Zhang and S. Taniguchi, "Water model study on inclusion removal from liquid steel by bubble flotation", ISIJ International, 2001, (in press).

26. M. B. Assar, P. H. Dauby and G. D. Lawson, "Opening the black box: PIV and MFC measurements in a continuous caster mold", Steelmaking Conference Proceeding, 83, ISS, Warrendale, PA, 2000, pp397-411.

27. H. Bai and B. G. Thomas, "Bubble Formation During Horizontal Gas Injection into Downward Flowing Liquid", Metallurgical Trans.B, 2001, (in press).

28. D. E. Hershey, B. G. Thomas and F. M. Najjar, "Turbulent Flow through Bifurated Nozzles", Int. J. Num. Meth. In Fluids, 1993, vol.17 (1), pp23-47.

29. B. G. Thomas, X. Huang and R. C. Sussman, "Simulation of argon gas flow effects in a continuous slab caster", Met. Trans.B, 1994, vol.25B (4), pp527-547.

30. X. K. Lan, J. M. Khodadadi and F. Shen, "Evaluation of six k-ε turbulence model predictions of flow in a continuous casting billet-mold water model using laser doppler velocity measurements", Met. & Mat. Trans. B, 1997, vol.28B (2), pp321-332.

31. P. Andrzejewski, K. U. Kohler and W. Pluschkell, “Model investigations on the fluid flow in continuous casting moulds of wide dimensions”, Steel Res., 1992, vol.63 (6), pp242-246.