Finite element modelling of turbulent fluid flow and heat transfer in continuous casting

Brian G. Thomas and Fady M. Najjar

Department of Mechanical and Industrial Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, USA

A two-dimensional finite element model has been developed to analyze turbulent, steady-state fluid flow and heat transfer within the liquid pool of a continuous steel-slab-casting machine, using the CFD code FIDAP. This high Reynolds number problem is often prone to instability in solving the finite element equations. To help provide guidelines for achieving convergence to a good solution for problems of this type, various solution strategies, relaxation factors, and meshes have been investigated. The effect of various numerical modelling parameters on the flow and temperature solutions are also investigated. These include, in particular, the boundary conditions for K and ϵ inlet conditions and wall laws and the turbulent Prandtl number. Finally, the predicted flow patterns and velocity fields show reasonable agreement with experimental observations and measurements conducted by using a Plexiglas water model, and the predicted heat flux profiles closely match previous measurements.

Introduction

Previous contributions to the understanding of turbulent fluid flow in metallurgical processes, such as continuous casting, have come about mainly through industrial experiments and full-scale physical water models.[1-4] In recent years the decrease in computational costs and the increasing capability of commercial modelling packages are making it possible to apply mathematical models as an additional tool to understanding these processes. These models have the advantage of easy extension to heat transfer, which is difficult with isothermal water models.

Almost all of the recent work that has been done to model these liquid metal processes has used finite domain or finite difference programs.[4-11] Applying the finite element method to solve these problems has the important additional advantage of easy adaptation to arbitrary complex geometries. However, experience in using the finite element method for these problems is very limited. Thus relatively little is known about the solution strategies, modelling parameters, boundary conditions, and convergence criteria required to obtain an accurate, economic solution to this type of problem using this method.

The present study was undertaken to investigate the effects of various numerical modelling parameters on the finite element solution of a particular turbulent fluid flow and heat transfer problem of some industrial importance: the flow of molten steel in the mold of a continuous casting machine. This example is intended to suggest guidelines for modelling metallurgical processes of this type, which are not widely known to the process engineers using the programs.

This chosen example problem is a simplified part of a larger system of models that are being developed to calculate heat transfer, shrinkage, and stress development in the solidifying shell and mold of a continuous slab caster, shown in *Figure 1*. The flow of liquid steel contained within the solidifying shell is very influential on the distribution of inclusion particles, which is important to the internal cleanliness and quality of the steel. In addition, turbulent heat transfer to the shell during the critical early stages of solidification greatly affects steel surface quality, breakouts, internal structure, and crack formation. This process, and related near-net-shape casting processes, involve complex nozzle and mold cavity geometries, which the finite element model is ideally suited to handle. The insights provided from an accurate model would be valuable in preventing defects and improving caster design and operation.

The purpose of this study is to provide some insight into the numerical aspects of the mathematical modelling of turbulent flow problems such as this, using the finite element method. The effects of various solution strategies, streamline upwinding and initial conditions, and mesh refinement on the convergence
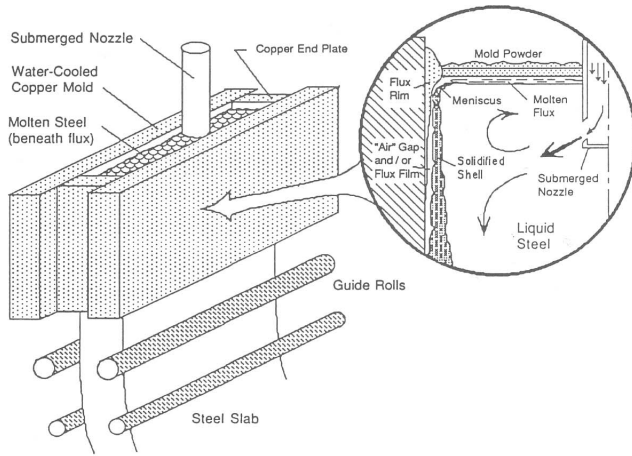
Figure 1. Schematic of continuous steel slab caster showing mold, fluid flow, and shell growth

Figure 2. Simulation domain and typical mesh used in the model

behavior, stability, and computational cost are investigated. A study was then conducted to determine the model sensitivity to boundary conditions including inlet, wall laws, K and ϵ turbulence parameters, and turbulent Prandtl number. Finally, the accuracy of the model predictions of both fluid velocity and heat transfer was validated through comparison with experimental and empirical results.

Model formulation

A finite element model was developed to simulate fluid flow and heat transfer within the domain illustrated in *Figure 2*. This represents a two-dimensional, vertical section parallel to the wide face through the center of a continuous steel-slab-casting machine. This section was chosen because the bifurcated nozzles used in slab casters, combined with the high aspect ratio of the slab mold, produce flow patterns whose major characteristics are exhibited in these two dimensions.

Fluid enters the model domain through an inlet surface that represents a nozzle port whose center is submerged 0.265 m below the meniscus and 0.365 m below the top of the 0.700-m-long mold. As the jet of liquid steel leaving the nozzle impinges upon the cooled "narrow-face" wall of the mold, a solid shell forms, which contains the liquid as it is withdrawn continuously from the bottom of the mold. Thus the position of the shell remains constant with time, and a steady-state analysis can be used to calculate the flow and the heat transfer that develop. The right side of the model domain is the inside of the solidifying steel shell, which is adjacent to the mold for the top 0.600 m. In this study, the top 3.00 m of a 1.32-m (52-inch)-wide slab caster was simulated, exploiting symmetry about the centerline, which forms the left side of the model domain.

The flow is fully turbulent, even far away from the nozzle, as is indicated by the calculated Reynolds number at mold exit of 10,700 (which is based on the data in *Table 1* and treating the rectangular mold as an equivalent diameter pipe, D). Thus the two-equation
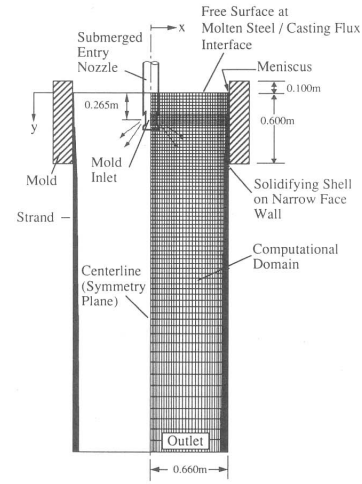
K-ϵ model,[12,13] which has been used previously to simulate a variety of turbulent flows,[6–10] was chosen to incorporate the effects of turbulence on the momentum and heat transfer. This model is far from perfect but appears to be the most accurate and robust model currently available.

This two-dimensional, steady-state, incompressible, fluid flow problem must solve for two unknown velocity distributions, v_x, and v_y, and the pressure distribution, p, which are governed by the continuity (volume conservation) equation and two momentum equations:

$$\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} = 0 \tag{1}$$

$$\rho\left(v_x\frac{\partial v_x}{\partial x} + v_y\frac{\partial v_x}{\partial y}\right) = -\frac{\partial p}{\partial x} + \rho f_x + 2\frac{\partial}{\partial x}\left(\mu_{\text{eff}}\frac{\partial v_x}{\partial x}\right)$$
$$+ \frac{\partial}{\partial y}\left(\mu_{\text{eff}}\left(\frac{\partial v_x}{\partial y} + \frac{\partial v_y}{\partial x}\right)\right) \tag{2}$$

$$\rho\left(v_x\frac{\partial v_y}{\partial x} + v_y\frac{\partial v_y}{\partial y}\right) = -\frac{\partial p}{\partial y} + \rho f_y + 2\frac{\partial}{\partial y}\left(\mu_{\text{eff}}\frac{\partial v_y}{\partial y}\right)$$
$$+ \frac{\partial}{\partial x}\left(\mu_{\text{eff}}\left(\frac{\partial v_x}{\partial y} + \frac{\partial v_y}{\partial x}\right)\right) \tag{3}$$

The effective viscosity, μ_{eff}, needed in equations (2) and (3) is greatly enhanced by the turbulent motion and is defined as the sum of molecular and turbulent (or eddy) viscosity components:

$$\mu_{\text{eff}} = \mu_0 + \mu_t \tag{4}$$

where μ_t is found from the K and ϵ model parameters by

$$\mu_t = C_\mu \rho \frac{K^2}{\epsilon} \tag{5}$$

These are in turn found by simultaneously solving two additional transport equations for the turbulent kinetic

Table 1. Standard simulation conditions

Half caster width simulated	0.660 m (52-in.-wide mold)
Caster length simulated	3.00 m
Nominal nozzle angle (at inlet)	15° downward
Nozzle submergence depth	0.265 m
Casting speed	0.0167 m/s
Inlet casting temperature	1550°C
Fixed wall temperature (liquidus solidification temperature)	1525°C
Surface heat transfer coefficient	40 W/(m²K)
Ambient temperature (above top surface of powder layer)	27°C
Convergence strategy	16 Suc. Sub. Iters. with RF = 0.4
Mesh size (node grid density)	40 × 75
Number of nodes across inlet	8
Inlet jet width, w	0.0148 m
Inlet and initial K (kinetic energy)	0.0502 m²/s²
Inlet and initial ϵ (dissipation)	0.457 m²/s³
Inlet peak velocity, v_x and v_y	1.062 m/s and 0.471 m/s
Initial velocity guess	0.0
Boundary conditions at wall	Wall law 2
Thickness of laminar boundary, x_n	0.01 m
Molecular viscosity, μ_0	0.00385 kg/(ms)
Density, ρ	7020 kg/m³
Molecular thermal conductivity, k_0	26 W/(mK)
Specific heat, C_p	680 J/(kg K)
Turbulent Prandtl Number, Pr_t	0.9
Dimensionless parameters:	
Reynolds number (inlet)	121,300
Reynolds number (outlet)	10,700
Grid molecular Reynolds number (inlet)	3,560
Grid turbulent Reynolds number (inlet)	2.5
Grid molecular Reynolds number (outlet)	490
Grid turbulent Reynolds number (outlet)	6
Grid Peclet number (inlet)	356
Grid Peclet number (outlet)	49
Laminar Prandtl number ($\mu_0 C_p/k_0$)	0.1
Brinkman number ($\mu_0 V^2)/(k_0 \Delta T)$	2.07e-08

energy, K, and its rate of dissipation, ϵ:

$$\rho\left(v_x\frac{\partial K}{\partial x} + v_y\frac{\partial K}{\partial y}\right) = \frac{\partial}{\partial x}\left(\frac{\mu_t}{\sigma_K}\frac{\partial K}{\partial x}\right) + \frac{\partial}{\partial y}\left(\frac{\mu_t}{\sigma_K}\frac{\partial K}{\partial y}\right)$$
$$- \rho\epsilon + \mu_t\Phi \quad (6)$$

$$\rho\left(v_x\frac{\partial \epsilon}{\partial x} + v_y\frac{\partial \epsilon}{\partial y}\right) = \frac{\partial}{\partial x}\left(\frac{\mu_t}{\sigma_\epsilon}\frac{\partial \epsilon}{\partial x}\right) + \frac{\partial}{\partial y}\left(\frac{\mu_t}{\sigma_\epsilon}\frac{\partial \epsilon}{\partial y}\right)$$
$$+ C_1\frac{\epsilon}{K}\mu_t\Phi - C_2\rho\frac{\epsilon^2}{K} \quad (7)$$

where

$$\Phi = 2\left(\frac{\partial v_x}{\partial x}\right)^2 + 2\left(\frac{\partial v_y}{\partial y}\right)^2 + \left(\frac{\partial v_y}{\partial x} + \frac{\partial v_x}{\partial y}\right)^2$$

These equations involve five empirical constants, which produce reasonable results for a wide range of flows when given standard values[12-14] as follows:

$$C_1 = 1.44, \qquad C_2 = 1.92, \qquad C_\mu = 0.09,$$
$$\sigma_K = 1.0, \qquad \sigma_\epsilon = 1.3$$

The temperature distribution, T, is found by solving the steady-state energy equation

$$\rho C_p\left(v_x\frac{\partial T}{\partial x} + v_y\frac{\partial T}{\partial y}\right) = \frac{\partial}{\partial x}\left(k_{\mathrm{eff}}\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(k_{\mathrm{eff}}\frac{\partial T}{\partial y}\right)$$
$$+ \mu_{\mathrm{eff}}\Phi \quad (8)$$

K and ϵ have an important effect on the heat flow solution owing to their influence on μ_t. In addition to controlling the generation of kinetic energy (the last term in equation (8)), μ_t greatly enhances the effective thermal conductivity, k_{eff}:

$$k_{\mathrm{eff}} = k_0 + k_t \quad (9)$$

where

$$k_t = \frac{C_p\mu_t}{\mathrm{Pr}_t} \quad (10)$$

and the turbulent Prandtl number, Pr_t, is an additional empirical constant that controls the turbulent thermal conductivity, k_t.

Buoyancy effects were neglected in the present analysis. Thus the energy equation was uncoupled from the other equations, and temperatures were calculated on the basis of the previous velocity solution. This

assumption is valid in the region of interest near the inlet, where the dimensionless parameter, Gr/Re^2, is 0.0037. Lower in the caster, however, natural convection will become important. Single-phase flow was also assumed, so effects such as those from argon gas bubble injection are not considered.

Boundary conditions

The governing equations are subject to boundary conditions along every edge of the computational domain (the mesh), illustrated in *Figure 2*.

Narrow face wall. To avoid the computational difficulties associated with modelling latent heat evolution during solidification below the liquidus temperature, fluid flow and heat flow were modelled up to, but not including, the mushy zone. Thus the edge of the mesh along the narrow face wall was defined to correspond to the outer extent of the laminar boundary layer found adjacent to the solid shell against the narrow face wall. Here, "wall function" boundary conditions were imposed to account for large property variations within this laminar mushy zone close to the solid shell wall, where the K-ϵ model equations are no longer valid. Along this vertical boundary the velocity normal to the wall, v_x, is set to zero, and the tangential velocity gradient is imposed by using an empirical function:

$$v_x = 0 \qquad (11)$$

$$\frac{\partial v_y}{\partial x} = \frac{\rho}{\mu_t} v_*^2 \frac{v_y}{|v_y|} \qquad (12)$$

where v_* is the empirically calculated friction velocity, which depends on the values of K and ϵ at the wall.

Boundary conditions must also be imposed on the K and ϵ fields at the wall. Two different sets of conditions were investigated, referred to as wall law 1:

$$K = C_\mu^{-0.5} v_*^2 \qquad (13)$$

$$\epsilon = \frac{|v_*|^3}{K x_n} \qquad (14)$$

$$v_* = \frac{\kappa v_y}{\ln\left(E\dfrac{\rho x_n v_*}{\mu_0}\right)} \qquad (15)$$

which is the most commonly used,[6,9,11] and wall law 2:

$$\frac{\partial K}{\partial x} = 0 \qquad (16)$$

$$\epsilon = \frac{C_\mu^{0.75} K^{1.5}}{\kappa x_n} \qquad (17)$$

$$v^* = \frac{\kappa v_y}{\ln\left(E\dfrac{\rho x_n C_\mu^{0.25} K^{0.5}}{\mu_0}\right)} \qquad (18)$$

The additional constants needed in the wall law equa-

tions are defined in the nomenclature. Reducing the wall roughness constant to simulate the rough surface of the dendrite wall was found to increase the sharpness of the heat flux peak, with few other effects.

No wall law was used for temperature. Instead, it was assumed that the laminar zone consisted solely of the interdendritic, or "mushy region" between the tips of the dendrites, at the liquidus temperature, and the fully solid shell, as pictured in *Figure 3*. Consequently, the temperature along the vertical narrow face edge of the model, corresponding to the dendrite tips, was set to the liquidus temperature. This assumption has been found to be reasonable for natural convection heat transfer during metal alloy solidification.[14] It has the great computational advantage of forcing latent heat evolution to occur entirely outside the simulation domain. The average thickness of the mushy region, which is known from calculations using solidification models, was thus used to define the boundary layer thickness, x_n.

To account for solidification of the solid shell against the mold wall, the narrow face computational boundary was tapered inward with distance down the mold wall for some runs. This required a nonzero normal velocity through the narrow face wall to account for steel leaving the computational domain by moving across the solidifying shell interface:

$$v_N = \frac{\text{casting speed * final shell thickness}}{\text{model domain length}} \qquad (19)$$

As was expected, the model results were insensitive to the exact value of this small change in the domain, which varied from 0 to 0.04 m at exit from the domain, 3 m below the meniscus.

Inlet. To derive the inlet velocity boundary conditions at the nozzle ports, a mass balance was performed in two dimensions, setting nozzle port length * nozzle outlet velocity equal to mold outlet width * casting speed. An eight-node, parabolic input velocity profile was employed both to satisfy this constraint and to achieve the same average and peak velocities into the mold as are found through the actual casting nozzle port. To achieve this, the nozzle port length had to be adjusted by using the above mass balance equation.
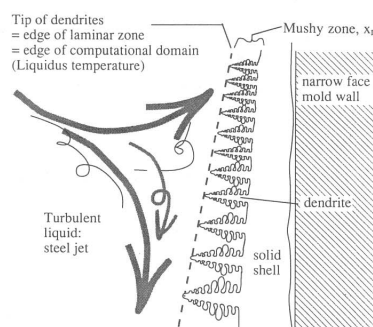
Components of the nozzle port velocity and values

Figure 3. Schematic of model treatment of near-wall region

for K and ϵ were fixed across the inlet boundary surface of the model. The accuracy of this method hinges on finding reasonable values for v_x, v_y, K, and ϵ at the inlet. These values were therefore calculated by using a separate fluid flow model of the nozzle itself, which we discuss later. Temperature across the inlet was fixed at the casting temperature of 1540°C.

Bottom outlet. Normal gradients of all variables (including v_x, v_y, K, ϵ, and T) were left at zero along the bottom outlet surface of the computational model domain. Constraining the outlet flow in any other way, such as fixing the outlet vertical velocity to the casting speed, was found to produce unrealistic results.

Top surface and centerline. Along the centerline symmetry plane (the left side) the normal velocities were constrained to zero, and gradients of other variables were left at zero. The same constraints were imposed along the top surface of the model, except for temperature. Here, heat is lost via conduction through three powder layers and radiation and convection to the ambient temperature above the top powder layer surface. This was represented by an equivalent convective heat transfer coefficient of 40 W/m² and an ambient temperature of 27°C. Observations of casters and water models indicate that the surface is relatively quiescent, so no free surface representation was required or attempted.

Solution methodology

The equations described above were converted into simultaneous algebraic equations by using the finite element method, assembled and solved by the commercial program FIDAP.[15] A penalty function approach, using a penalty parameter of 10^{-8}, was used to satisfy continuity without solving an additional PDE (see Ref. 15 for more detail). The standard conditions assumed in the model runs are given in *Table 1*. The standard mesh, shown in *Figure 2*, consists of a 39 × 74 grid of four-node linear, quadrilateral elements and 74 two-node linear wall elements along the right side boundary. All meshes were graded to provide smaller elements near the walls and inlet area. Although various solution strategies were investigated, most runs employed 15–30 successive substitution iterations, using an underrelaxation factor, RF, of 0.3 or 0.4, until the relative error in the residual force vector was stably reduced to less than 1%. Experience is required to control the solution procedure to achieve convergence for each given simulation, which will be discussed later. Numerical stability was improved by activating a Petrov-Galerkin formulation or streamline upwinding.[16] Computation times required for different computers are summarized in *Table 2*.

Nozzle model

The characteristics of the fluid leaving the submerged entry nozzle have a great influence on both fluid flow in the mold and heat transfer to the growing shell.[17] To determine the proper boundary conditions for the inlet of the caster model (v_x, v_y, K, and ϵ), a separate model for fluid flow within this nozzle has been developed. The model is currently based on a 76-mm diameter bifurcated nozzle with square 65 × 90 mm outlet ports angled at 15° down from the horizontal. The vertical velocity into the top of the nozzle was set to the profile characteristic of fully developed turbulent flow, known as the power-law or seventh-root profile[18,19] whose average was calculated through a two-dimensional mass balance as described earlier. K and ϵ were computed as functions of distance across the inlet using a mixing-length approximation[20] whose values average about 0.0045 and 0.02, respectively. Boundary conditions and solution strategy were similar to that employed in the full caster model, and convergence was rapid and computationally inexpensive. Further details are given elsewhere.[21]

A typical two-dimensional flow pattern in the nozzle is shown in *Figure 4*. Calculations confirm observations of physical water models that the flow leaves the

Table 2. Computational requirements (40 × 75 mesh)

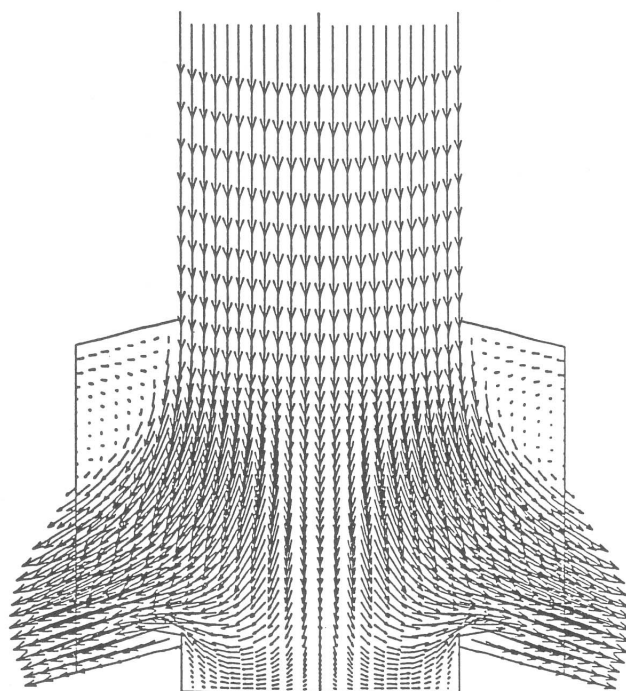| Computer	CPU time (s/iter) (direct solver)	CPU time (s/iter) (indirect solver)
Ridge 32 S	2400	
Iris 4D/20 (8 MB RAM)	1200	180
Iris 4D/20 (16 MB RAM)	600	120
Convex C1-XP	150	
Cray-2	50	15
Cray X-MP 48	40	

Figure 4. Flow from nozzle used to determine inlet boundary conditions

nozzle at a steeper downward angle than the nominal angle built into the nozzle exit ports.[21] In addition, the velocity distribution over the nozzle outlet is skewed, most of the fluid leaving the lower half of the nozzle port openings and some recirculating fluid actually entering the upper part. This distribution is reasonably approximated by a truncated parabolic velocity profile, which was used as the inlet condition for the mold model. Weighted average K and ϵ values were calculated across the outlet ports and used as inlet conditions for most mold simulations. These inlet conditions were based only on the magnitude of the positive outwardly flowing velocities leaving the bottom of the nozzle.

Convergence strategies

To understand the convergence behavior of this problem, it is helpful to examine the equations involved in the finite-element method. Assembly of the governing equations applied to each element produces a global system of nonlinear algebraic equations that can be expressed as

$$[K(U)]\{U\} = \{F\} \tag{20}$$

where the solution vector $\{U\} = \{v_x, v_y, p, T, K, \epsilon\}$, $\{F\}$ is the force vector, and the global stiffness matrix, $[K]$, can be decomposed into

$$[K(U)] = [K_d] + [K_c(U)] \tag{21}$$

where $[K_d]$ represents the contribution of the diffusion, pressure, and continuity terms and $[K_c(U)]$ consists of convective terms and thus depends on the solution vector, $\{U\}$.

The highly nonlinear aspects of the governing equations for this turbulent, recirculate flow field made the solution to the present problem prone to instability. This is typical for high Reynolds number problems and arises because the low viscosity reduces the importance of $[K_d]$, making the nonlinear $[K_c]$ dominate the solution. Use of the K-ϵ turbulence model actually contributes to improving stability. It does this by increasing the effective viscosity by several orders of magnitude, thereby increasing the importance of the linear $[K_d]$ terms and substantially reducing the grid Reynolds number (from 3560 based on molecular viscosity to 21 based on effective viscosity).

It is difficult to generate an iteration scheme for reaching convergence in problems of this type and to decide when convergence to the "proper solution" has been achieved. To help provide guidelines for these decisions, an investigation of the effect of various iteration strategies on the convergence of this problem was conducted.

The best, most efficient method investigated was found to be simple, relaxed successive substitution, in which the linearized equations are solved implicitly or "directly" in each of 15–30 iterations. This strategy uses the solution calculated at the previous iteration to help evaluate the nonlinear terms in $[K]$ for the next iteration:

$$[K(U_{i-1})]\{U^*\} = \{F\} \tag{22}$$

To suppress oscillatory behavior, the next guess of the solution vector is "relaxed," using a linear combination of the new solution vector, $\{U^*\}$, and the one obtained at the previous iteration:

$$\{U_i\} = RF\{U^*\} + (1 - RF)\{U_{i-1}\} \tag{23}$$

The relaxation factor, RF, is the fraction of the new solution that is used to calculate nonlinear terms for the next iteration. The error in the solution is given by

$$\{R_i\} = [K(U_{i-1})]\{U_i\} - \{F\} \tag{24}$$

Convergence is attained only when this error is small. The percent error in the residual (relative to the magnitude of the initial residual vector), $|R_i|/|R_0| \times 100\%$, is required to decrease from its initial defined value of 100% to below some chosen tolerance. At the same time the solution vector should change very little between successive iterations, so that the percent difference between successive solution vectors, $\|U_i - U_{i-1}\|/\|U_i\| \times 100\%$, is less than another tolerance. These two criteria together are reported to "measure" the solution accuracy.[15,22] Choosing these tolerances is a difficult task, however.

The behavior of these two convergence criteria during the numerical simulation is shown in *Figures 5* and *6*, which also illustrate the great importance of the relaxation factor. Initially, the relative error in the residual decreases most rapidly with a high RF, but it eventually becomes oscillatory, as is indicated by following the history of percent residual error in *Figure 5*. The onset of oscillation is delayed by lowering the relaxation factor, to use more of the old solution in the next guess, thereby increasing stability of the solution.
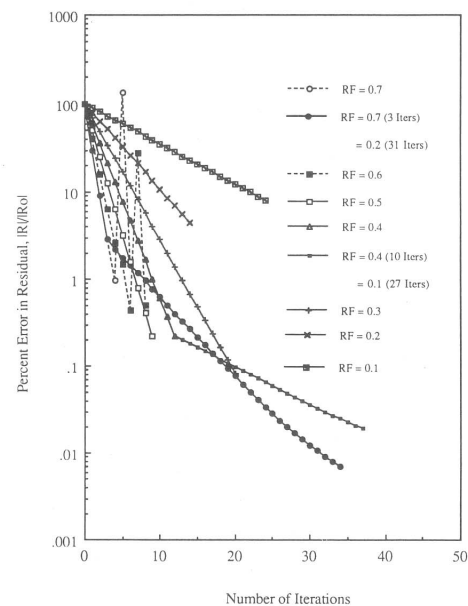

Figure 5. Effect of relaxation factor on convergence behavior: Relative error in residual (%)
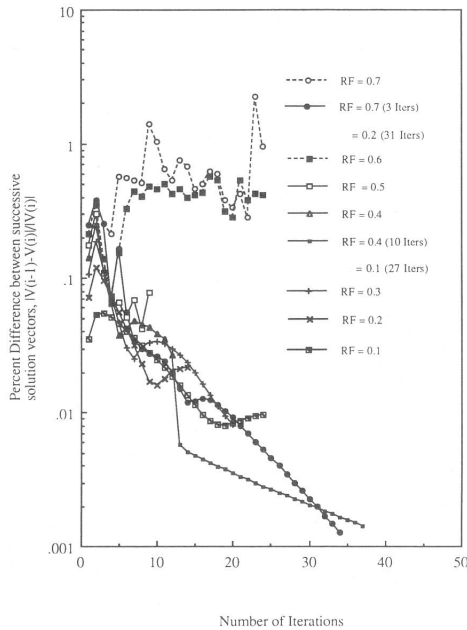
Figure 6. Effect of relaxation factor on convergence behavior: Difference between successive solution vectors (%)

A striking feature of all of the stable runs in *Figure 5* is the monotonic decrease in percent residual error by the same ratio in each iteration. This behavior can be examined theoretically by forming the ratio between residual vector at two consecutive iterations:

$$\frac{\{R_i\}}{\{R_{i-1}\}} = \frac{[K(U_{i-1})]\{U_i\} - \{F\}}{[K(U_{i-2})]\{U_{i-1}\} - \{F\}} \quad (25)$$

Assuming that $[K]$ does not vary much between iterations, then substitution of equation (23) into (25) yields

$$\frac{\{R_i\}}{\{R_{i-1}\}} = \frac{RF[K]\{U^*\} + (1 - RF)[K]\{U_{i-1}\} - \{F\}}{[K]\{U_{i-1}\} - \{F\}} \quad (26)$$

Further substitution of equation (22) into (26) yields

$$\frac{\{R_i\}}{\{R_{i-1}\}} = \frac{(1 - RF)[K]\{U_{i-1}\} - (1 - RF)\{F\}}{[K]\{U_{i-1}\} - \{F\}}$$

$$= 1 - RF \quad (27)$$

Thus the percent residual error should decrease monotonically by the ratio $(1 - RF)$ in each iteration. This convergence rate is manifested by the straight downward slopes of the residual error curves in *Figure 5*. It is achieved when $[K]$ does not vary much between successive iterations, which occurs when $[K_d]$ dominates the stiffness matrix. Naturally, this is true initially, since the initial velocities were zero, and it is also true during the early iterations of the simulation when $[K_c]$ is small and the residual error is high. Surprisingly, this condition also held very close even during later iterations for all *RF* tested for this problem, as long as the solution did not go unstable.

The exponential decrease in residual error implied

by this factor of $(1 - RF)$ therefore represents the maximum attainable convergence rate for a given relaxation factor. Faster convergence is obviously achieved when the substitution is relaxed as little as possible, by using large values of *RF*, as long as unstable oscillation can be avoided.

This knowledge can be used to develop a better strategy to reach stable convergence by systematically changing *RF* as the solution proceeds. The convergence rate is accelerated early by using a large *RF* to greatly reduce the large initial error while $[K]$ is linear. Later, a smaller *RF* is used to increase stability and suppress the tendency toward oscillatory behavior. This is needed for convergence during these latter iterations, since the velocities increase and the equations become more nonlinear.

Figure 5 illustrates the benefit of using this combined strategy, starting with a high relaxation factor (0.7) for a few iterations (three) and then switching to a smaller one (for example, 0.2) for the remaining 31 iterations. This produces the same downward slope after restarting as obtained using $RF = 0.2$ from the initial 100% error. Thus a significant savings in computation is possible.

Combining two lower relaxation factors, $RF = 0.4$ then $RF = 0.1$, can increase stability for a more difficult run. However, it is less efficient, since the residual error does not reach 0.020% until the 37th iteration, compared to only 27 iterations required for the $RF = 0.7/0.2$ strategy.

Other convergence criteria

The previous discussion has focused on the residual error as a reliable indicator of convergence. In contrast, *Figure 6* shows how the percent difference between successive solution vectors is not as reliable. Sometimes it increases while the solution is still stably converging; other times it decreases while the solution has become oscillatory and poor (for example, $RF = 0.7$ from 10 to 20 iterations). Another example is the factor of 5 reduction in percent difference that sharply occurred upon changing *RF* from 0.4 to 0.1. This might be misleading, since it merely reflects the smaller change in solution vector that accompanies the *slower* rate of convergence that occurred after changing *RF* (which is clearly seen in *Figure 5*). Thus, following the history of the residual error gives a clearer understanding of the convergence.

Figure 7(a) shows the streamlines for a smoothly converged solution obtained after 12 iterations using $RF = 0.3$ and standard conditions. The solution (which had a 0.7% error in residual, 0.03% difference between successive solution vectors) was almost identical to that of a highly converged solution (0.007% error in residual, 0.0015% difference between successive solution vectors).

The residual error alone is an insufficient indicator of convergence to a reasonable solution. Evidence for this is provided in *Figure 7(b)*, where a solution was obtained (after 25 iterations using a relaxation factor
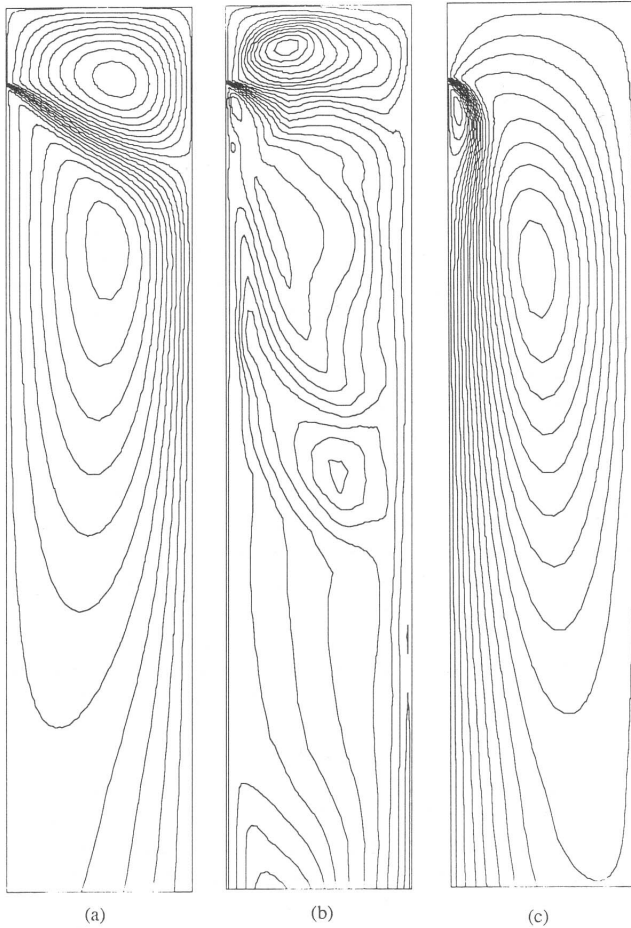
(a)   (b)   (c)

Figure 7. Velocity vector profile in the mold for three convergence schemes: (a) smooth converged solution (12 iterations with $RF = 0.3$) (0.7% error in residual, 0.03% difference between successive solution vectors). (b) oscillatory (25 iterations with $RF = 0.6$) (0.5% error in residual, 0.4% difference between successive solution vectors). (c) same as part (a) with high initial guess of velocity (0.08% error in residual, 0.01% difference between successive solution vectors)

of 0.6) that had a residual error of 0.5%, which is better than the good solution in *Figure 7(a)*. However, the residual error had been oscillating significantly between successive iterations, and the difference between successive solution vectors was 0.4%. The flow field computed can be seen to be qualitatively unreasonable, even though its mass balance was good. Since it is not obvious that 0.4% is too high a difference, close attention to the history of the residual error might be a better criterion for deciding when a solution is oscillatory and not to be trusted.

Initial conditions

The effect of the initial guess of the velocities on the convergence behavior is shown in *Figure 7(c)*. The model was run from a high uniform initial velocity, v_y, of 2 m/s. Even though the numerical simulation was stable and converged rapidly, the velocity field is qualitatively unreasonable. However, this "solution" appears to represent a "local minimum," where mass conservation is satisfied and error parameters are very low, even though the velocities are far away from the true solution. (If the fluid properties are altered greatly in this problem, the flow pattern in *Figure 7(c)* could become the correct solution.) Thus, choosing good initial conditions is a very important factor in attaining a fast converging solution. In the absence of good information for this guess, setting initial velocities to zero appears to be the best choice, since this helps to linearize the equations for the early iterations.

Upwinding

The primary cause of instability in the numerical simulation appears to be the large grid Reynolds number. The stability of the turbulent problem has been enhanced by adopting the Petrov-Galerkin formulation or streamline upwinding.[16,23–25] By preparing the downstream elements for the flow before it arrives, this method greatly improves stability of the solution. However, upwinding also falsely increases diffusion by artificially enhancing the viscosity in the momentum equation.[24] This in turn increases turbulent conductivity for the heat transfer problem, even though it has no effect on solution of the linear energy equation in the present problem.

A solution made to converge at great difficulty without upwinding revealed that this method has minimal effect on the calculated velocities and only a small effect on the heat transfer. Increasing the upwinding factor from 0.5 to 1.5 in the velocity solution was found to increase the heat flux by a maximum of 13% at the peak. There was almost no difference in heat flux beyond the impingement region, since the upwinding scheme avoids "cross-wind" false diffusion.[15]

Mesh refinement

Mesh design is a very critical aspect of numerical modelling, particularly for the finite element method. The present study investigated convergence behavior and computational time for a variety of meshes, using the same simulation conditions as are outlined in *Table 1*.

All meshes studied were found to produce similar velocity predictions when converged. However, a finer mesh near the wall boundaries was required for accurate temperature predictions in order to resolve the steep velocity and temperature gradients there. Coarser meshes produced lower heat flux to the wall, particularly at higher casting speeds, where the increase in Reynolds number, caused by the higher average velocities, increased convergence difficulty for a given mesh. Thus the 40×75 node mesh was chosen as optimum for the present problem.

Numerical stability was found to improve as the mesh was refined. Thus fewer iterations were required for convergence on a finer mesh, since a larger relaxation factor could be used. For example, the 40×75 mesh converged to 0.05% residual error in 23 iterations, while a 20×40 mesh required 40 iterations. The increase in stability can be understood by considering that the grid Reynolds number at the outlet (based on
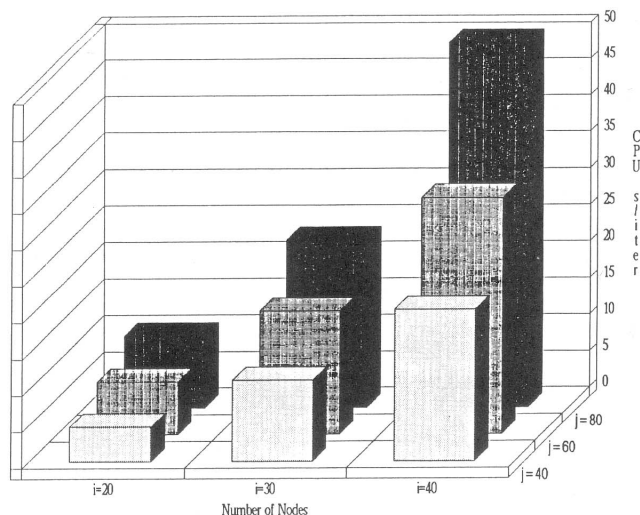
Figure 8. Effect of mesh refinement on computation time ($i \times j$ node grid)

μ_0) decreased from 1100 for the 20 × 40 mesh to only 490 for the 40 × 75 mesh.

The CPU time required per iteration is summarized for different meshes in *Figure 8*. The observed trends are roughly consistent with the correlation[26] that the computational cost for a direct solver is proportional to the square of the bandwidth multiplied by the total number of nodes in the mesh. Thus for a constant number of nodes per row the bandwidth is constant, so the CPU time per iteration rose linearly. When the mesh density was doubled in both directions simultaneously, this cost increased by a factor of 13, which is close to the 16-fold increase predicted by the correlation. This high cost escalation for mesh refinement more than offsets the stability advantages of finer meshes. It provides a large incentive to find more computationally efficient solution methods for large problems.

Other iteration strategies

Prior to the choice of the direct solution method (relaxed successive substitution), which was discussed in depth in the previous sections, several other iteration methods were investigated to obtain the solution. The first such strategy attempted to reach the steady-state solution by time stepping through the initial transient. Although this method appeared likely to succeed, far more than the 15–30 steps needed for a reliable steady-state solution were required, so this strategy is prohibitively costly. This finding indicates that direct solvers are much better suited to iterative schemes to reach the steady-state solution. This might not be the case if indirect solution methods or explicit integration schemes had been tested with this method.

The Newton-Raphson iteration approach, which has proven to be very successful for solving nonlinear finite element stress problems, was found to produce extremely unstable behavior in the present flow problem,

regardless of how close the initial guess was to the final solution. For example, restarting a run that had already converged to within 0.2% by using Newton-Raphson iteration diverged to a 1000% error in only two iterations.

Another solution strategy, designed to enhance the importance of $[K_d]$ and thereby improve stability, was to slowly increase the Reynold's number in stages from an artificially low initial value. This strategy involved iteration within each of about six steps that composed a single run of the model. The first step used a viscosity 100 times larger than the true μ_0 and started with one tenth the proper density. After a reasonably converged solution for this artificially viscous fluid was obtained, by incrementing the density over ten iterations, the results were used as the initial guess for the next step with a higher viscosity. Over successive steps the fluid properties were gradually altered to approach their proper values. This method was extremely time consuming and computationally expensive. However, it was successfully employed to solve for the flow field for the computationally difficult boundary condition of zero gradients for K and ϵ at the inlet. Combining this stepwise increase in the Reynold's number with successive substitution is a very robust method to achieve a solution for extremely unstable problems.

Finally, an indirect solution method was tested. This method divides up $\{U\}$ in equation (20) and solves for v_x, v_y, K, ϵ, T, p, and Δp as seven uncoupled systems of equations, using a pressure projection technique.[15] This ''segregated solver'' needs 15 to 20 times less CPU storage and significantly less computation time at each iteration. However, many more iterations are needed to reach convergence, since the system is not completely self-consistent at the end of each iteration. This method requires a relaxation factor for each unknown degree of freedom, and higher values can be used before stability problems are encountered. *Figure 9* shows that convergence to 0.03% on the present problem required 325 iterations, using relaxation factors (*RF*) of 0.85 for v_x, v_y, K, and ϵ. The convergence
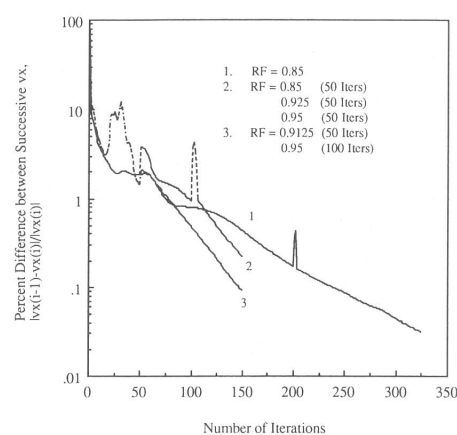

Figure 9. Convergence history of *x*-velocity component using the segregated solver: Difference between successive v_x solution vectors (%)

behavior of this method is similar to that of the direct methods. As *Figure 9* shows, increasing the relaxation factors after 50 iterations reduced the number of iterations required for 0.1% convergence from 250 to only 150. Unrecoverable oscillation and a poor solution resulted when the relaxation factors were increased further.

The computational cost per iteration for this method is about five times less than that of the direct solvers, as is shown in *Table 2*. The speedup is most pronounced on workstations with limited CPU RAM storage. Moreover, this cost increases only linearly with the number of nodes in the mesh, compared with quadratic or more increases for the direct solvers. However, for the present problem and mesh the increased number of iterations required (ten times more than the direct solver) overcomes this improvement, so this method was less economical. As the mesh is refined, more iterations appear to be required for convergence, which is contrary to experience with direct solvers. These findings suggest that the segregated solver might be useful only for very large problems, such as are encountered in three-dimensional simulations, in which reduced RAM storage becomes an important advantage as well.

K and ϵ boundary conditions

Having established a reliable procedure to obtain convergence, the effect of various numerical parameters on the solution were investigated. The boundary conditions on the K and ϵ parameters were found to be very influential on both the results and numerical convergence.

Effect of inlet conditions

The importance of the K and ϵ values used across the inlet plane of the caster model is shown in *Figure 10*. Significant differences in the angle of the jet leaving the inlet can be seen with different boundary conditions. Increasing K appears to steepen jet angle while increasing ϵ decreases it. These turbulence parameters also have an important influence on the temperature field. Thus it is crucial to find acceptable values for these parameters at the inlet.

The first method employed to get around this problem was to set the gradients of all variables, including K and ϵ, to zero across the inlet. Unfortunately, this proved to be very computationally expensive, since the solution had poor stability and required incremental adjustment of fluid properties and about 100 iterations before convergence was reached. This required roughly five times the execution time of the method that is currently used, although it produced similar flow results. Fixing K and ϵ inlet values results in much more stable convergence behavior.

The present procedure prescribes values for K, ϵ, v_x, and v_y across the inlet plane to the mold (see *Table 1*) using the model of the nozzle introduced earlier.
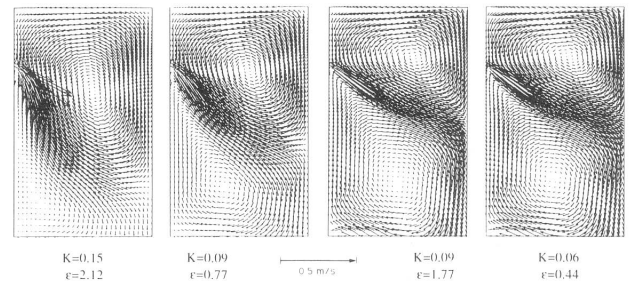

Figure 10. Effect of K and ϵ inlet values on calculated velocities

Nozzle model runs using a variety of inlet boundary conditions, including the variable profiles based on fully developed pipe flow, determined that the velocities, K, and ϵ at the nozzle outlet were insensitive to the values of K and ϵ used at the nozzle inlet. This finding was expected, since the turbulence levels leaving the nozzle are determined mainly by the major flow changes occurring near the bottom of the nozzle. Thus by moving the less certain K and ϵ boundary conditions farther upstream, flow in the area of interest in the caster interior was made insensitive to the exact values of K and ϵ employed as mold inlet boundary conditions.

Effect of the wall law

In addition to being important at the inlet, the boundary conditions for K and ϵ are very influential at the other edges of the computational domain. They were next investigated by varying the wall law along the narrow face wall where the shell grows.

Figures 11 and *12* show that neither the streamlines nor the velocity profile along the jet axis are much affected by the wall function set employed. However, the wall law has a great effect on the turbulent kinetic energy and turbulent viscosity distribution. *Figure 13* shows that differences between the K profiles increase with distance from the fixed inlet value. Along the edge of the computational domain, wall law 2 produces K values that are more than ten times greater than wall law 1 (see *Figure 14*) and 1000 times greater at the jet impingement point. Wall law 1 produces an unrealistic sharp minimum in kinetic energy at the impingement point instead of the expected increase. This problem arises because equations (13)–(18) for wall law 1 were derived by assuming that the flow pattern was remote from recirculating or detachment regions and stagnation points. Near such points, v^* vanishes, leading to low turbulence values. Wall law set 2 does not have this problem, since the K and ϵ fields are not as closely linked to the velocities at the wall. It therefore produces more appropriate flow characteristics near the stagnation region at jet impingement.

Figure 15 shows that the turbulent viscosity behaves in the same manner as K, with lower overall values and a sharp decrease at the jet impingement point for wall law 1, reflecting the domination of K (as compared to ϵ) in determining μ_t. This effect of wall law on the turbulent viscosity has a corresponding high influence
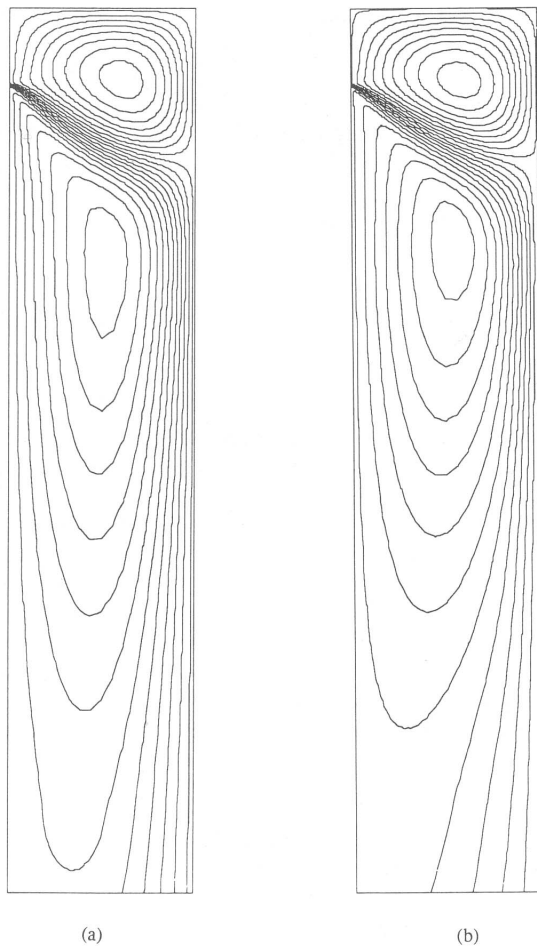
(a)   (b)

Figure 11. Effect of wall law boundary conditions on calculated streamlines. (a) Wall law set 1. (b) Wall law set 2

Figure 12. Effect of wall law on velocities calculated within the jet
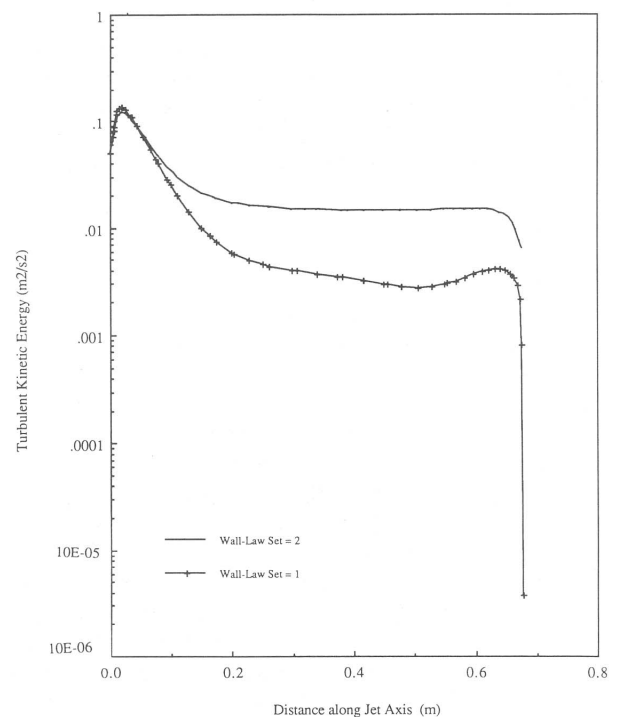
Figure 13. Effect of wall law on turbulent kinetic energy distribution within the jet

on the temperature distribution inside the caster, through its effect on k_t.

The important effect of wall law on the temperature field and heat flux profile down the narrow face are illustrated in *Figures 16* and *17*. The lower μ_t of wall law 1 greatly reduces k_{eff} at the wall, which reduces heat flux out of the wall and increases temperature in the liquid. The temperature contours in *Figure 16* for wall law 1 are indicative of the lower thermal diffusion, the heat following the path of the jet much further than is encountered using wall law 2. *Figure 17* shows the significant overall decline in heat flux produced using wall law 1. Wall law 1 also produces a sharp drop in heat flux at the impingement point, corresponding to the lower turbulence levels there. In contrast, results using wall law 2 are physically reasonable, with the maximum heat flux at the point of impingement where expected.

The effect of altering the wall boundary conditions was investigated further by modifying the boundary conditions on K in wall law 2. Instead of applying a zero gradient for the turbulent kinetic energy, the K at the wall was fixed to low and then high values on successive runs. This boundary condition had the ad-
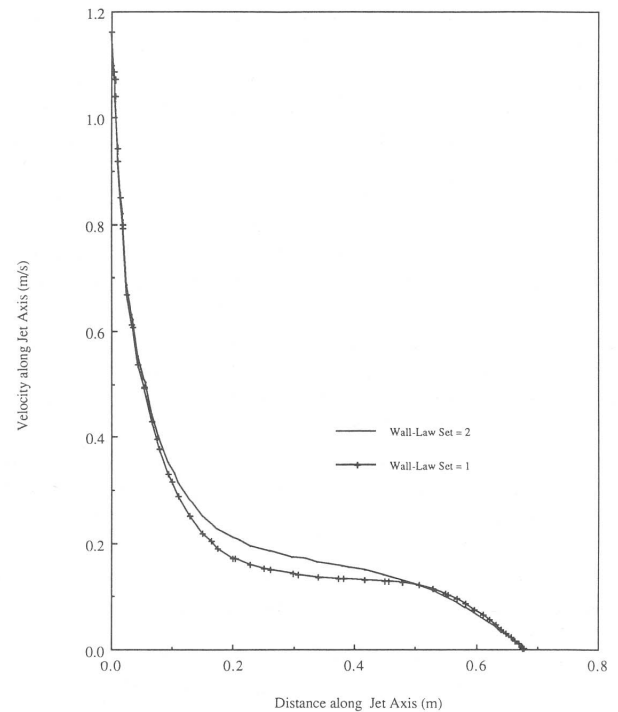
vantage of greatly accelerating the convergence rate of the solution, requiring only 15 iterations to converge, compared to 30 iterations required for the zero-gradient condition. This improvement is probably due to the inherent increase in stability of fixed wall con-
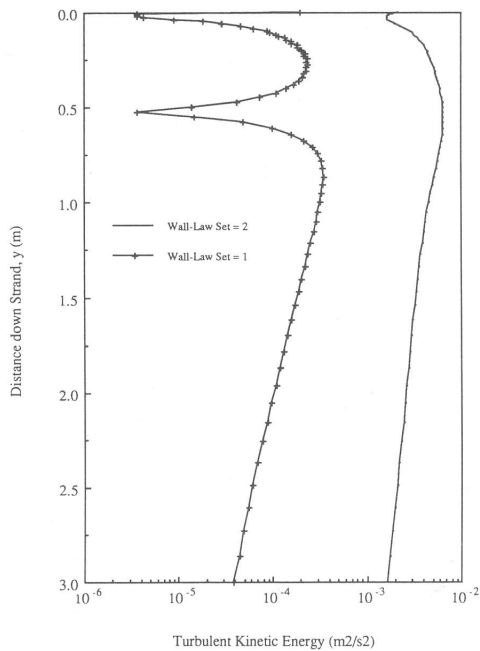
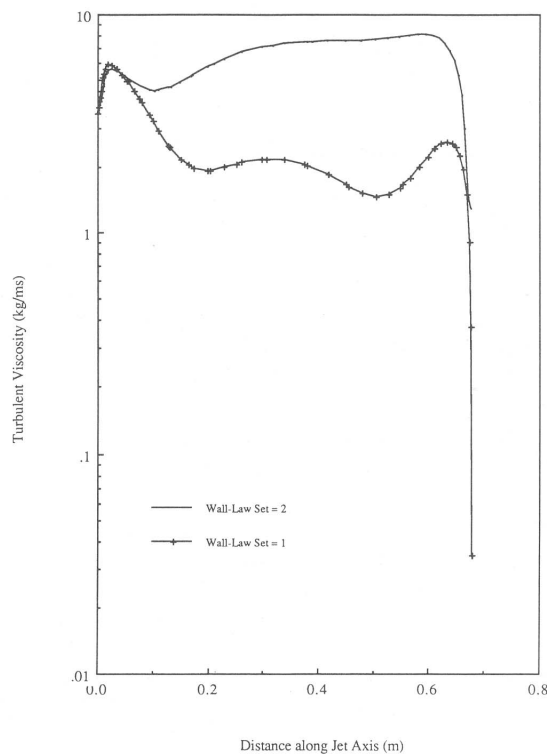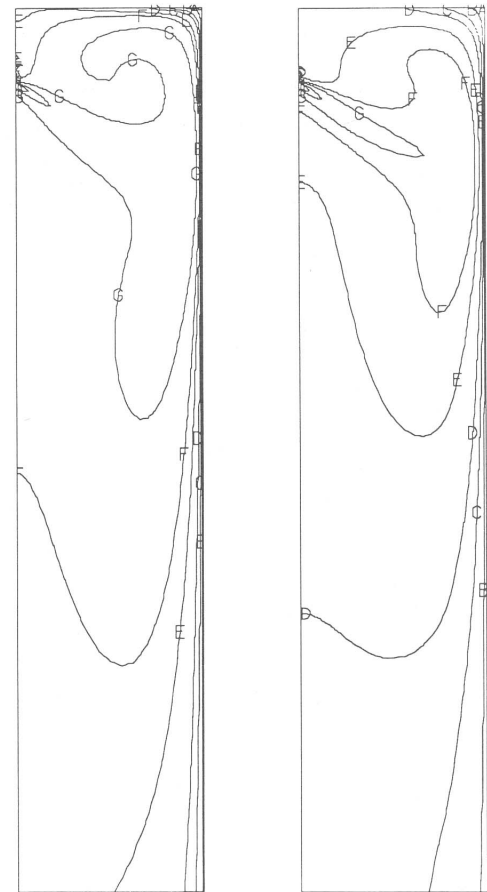**Figure 14.** Effect of wall law on turbulent kinetic energy profile down the narrow face wall

Figure 15. Effect of wall law on turbulent viscosity distribution within the jet

A = 1525	F = 1537
B = 1527	G = 1540
C = 1530	H = 1545
D = 1532	I = 1547
E = 1535	J = 1550

Figure 16. Effect of wall law on temperature distribution. (a) Wall law set 1. (b) Wall law set 2

the low value yields a larger recirculating region, comparable to the predictions of wall law set 1 (*Figure 11*). The streamlines computed with wall law set 2 fall between those two limiting cases. The corresponding effect on heat flux profiles is even more pronounced. *Figure 19* shows the expected great reduction in heat flux calculated for the low fixed value of K and excessively high heat flux for the high fixed value of K.

In conclusion, the boundary conditions on K and ϵ have an important influence on the simulation. Wall law 1, which was derived by assuming parallel flow along the wall, breaks down in regions of recirculation, attachment, detachment, and impingement, where this assumption is not valid. Wall law 2, with zero gradient on K at the wall, appears to be the most accurate boundary condition to use for the present problem. It predicts reasonable lower temperatures in the liquid and matches the heat flux. Although not greatly affecting the velocity prediction, the commonly used wall law 1 will lead to problems in predicting heat transfer in most practical systems, which involve recirculation and stagnation points.
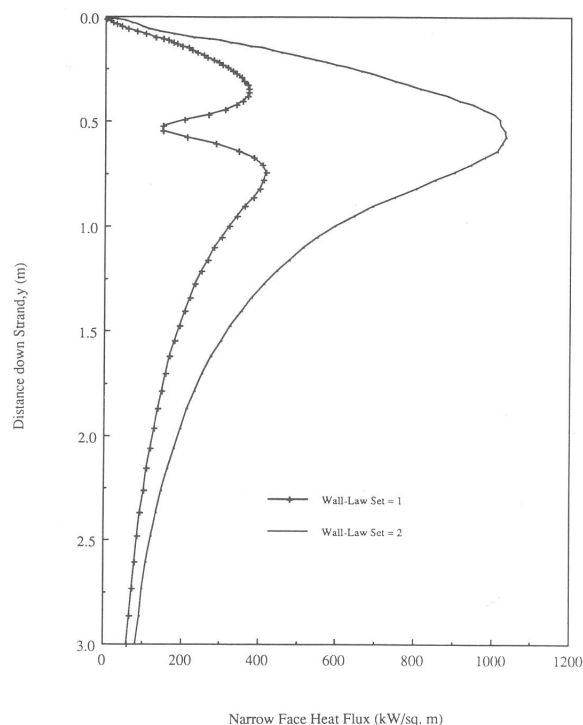
ditions over fixed gradients. Since the values of K are less able to change, the equations become less nonlinear.

The streamlines in *Figure 18* show that the flow pattern is quite sensitive to these wall conditions. The high value leads to a small recirculating zone, while

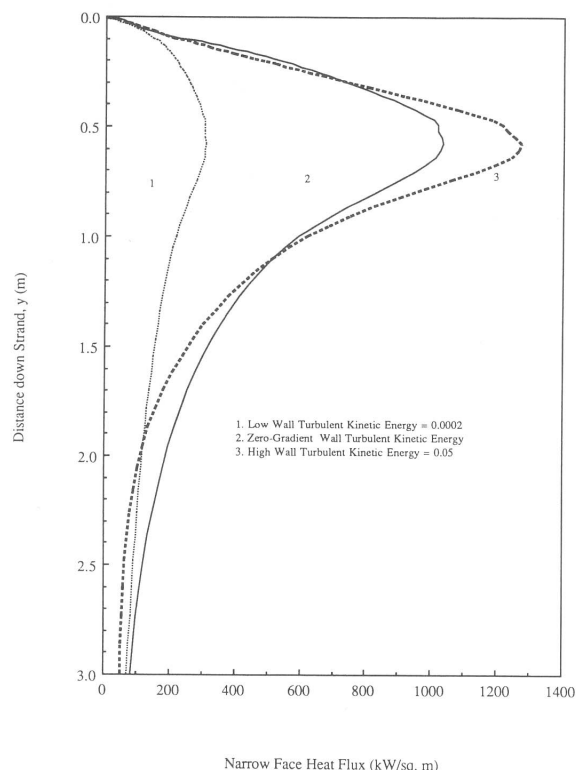**Figure 17.** Effect of wall law on heat flux profile down the narrow face wall

Figure 19. Effect of altering K boundary condition along internal walls on the heat flux profile
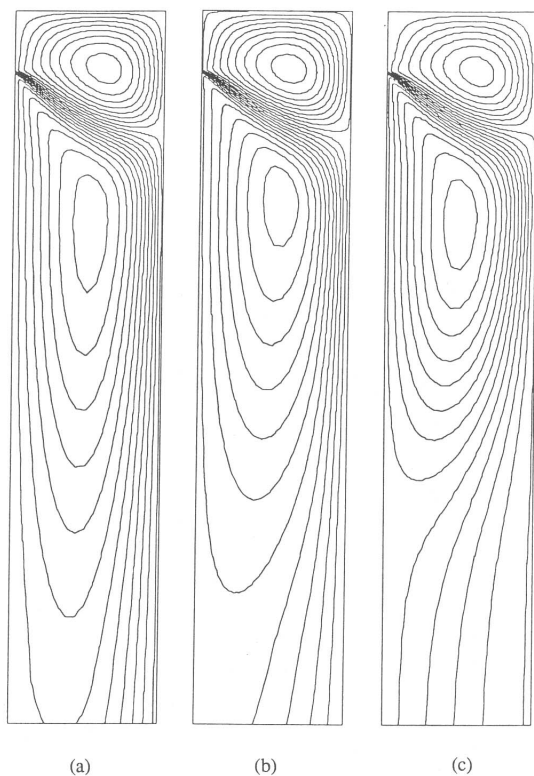

Figure 18. Effect of altering K boundary condition along internal wall(s) on streamlines. (a) Low K at wall. (b) Zero K-gradient at wall. (c) High K at wall

Turbulent Prandtl number

Conductive heat transfer in the flowing liquid is greatly enhanced by turbulent eddy transport, which is accounted for in the heat flow model by the turbulent thermal conductivity, k_t, in equations (8)–(10). Since μ_t is fixed by the velocity calculations, k_t is governed by the choice of the turbulent Prandtl number, Pr_t, used in equation (10). Thus, a good value for Pr_t must be found.

In the turbulent bulk of the flow, heat and momentum are both transported through the same process involving the turbulent eddies. Thus similarity has been found to exist between the eddy momentum and thermal diffusivities, so their ratio, Pr_t, is close to 1. This is Reynold's analogy and compares favorably with experimental values between 0.7 and 0.9 for many fluids.[27,28]

Near the wall, however, the relation is not as simple, and large scatter and sometimes contradictory values of Pr_t have been reported.[29] By applying a modified mixing-length theory to the transport of heat, Cebeci[30] obtained the following distribution for Pr_t as a function of distance from the wall, x_2:

$$\mathrm{Pr}_t = \frac{\kappa_m \left(1 - \exp\left(-\dfrac{x_2}{A_m} \right) \right)}{\kappa_\gamma \left(1 - \exp\left(-\dfrac{x_2}{A_\gamma} \right) \right)} \tag{28}$$

At large distances from the wall this yields $\mathrm{Pr}_t =$

$\kappa_m/\kappa_\gamma = 0.9$. As the wall is approached, equation (28) predicts that Pr_t increases for liquid metals, which have low molecular Prandtl numbers, reaching a maximum value greater than 10.

In view of its important effect on heat flux to the wall, the effect of varying Pr_t on model predictions was investigated in two ways: (1) by changing Pr_t equally throughout the domain and (2) by increasing Pr_t only near the wall. Using Boussinesq's approximation, the total heat flux from both laminar conduction and turbulent convection is calculated by

$$q_{\text{total}} = k_{\text{eff}} \frac{\partial T}{\partial n} = \left(k_0 + \frac{C_p \mu_t}{Pr_t} \right) \frac{\partial T}{\partial n} \qquad (29)$$

Figure 20 illustrates the effect of turbulent Prandtl number on the turbulent thermal conductivity near the wall. Usually, k_t is extremely high, exceeding the laminar value of 26 W/mK by more than one order of magnitude. As was expected, k_t drops greatly as Pr_t near the wall is increased, eventually making k_t negligible in relation to k_0, which then dominates the heat flux.

The temperature field is greatly affected by Pr_t, as shown in *Figures 21* and *22*. As Pr_t is increased, thermal diffusion within the liquid is reduced, so heat travels farther with the jet. *Figure 21* shows how this steepens the temperature gradients near the wall. The net result is higher temperature liquid, as is indicated in *Figure 22*, which shows the location of the calculated 1537.5°C isotherm for different choices of Pr_t. This
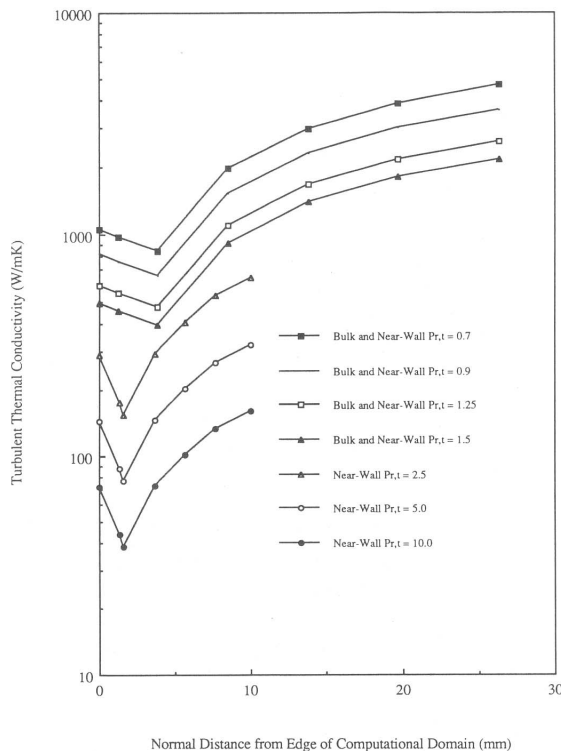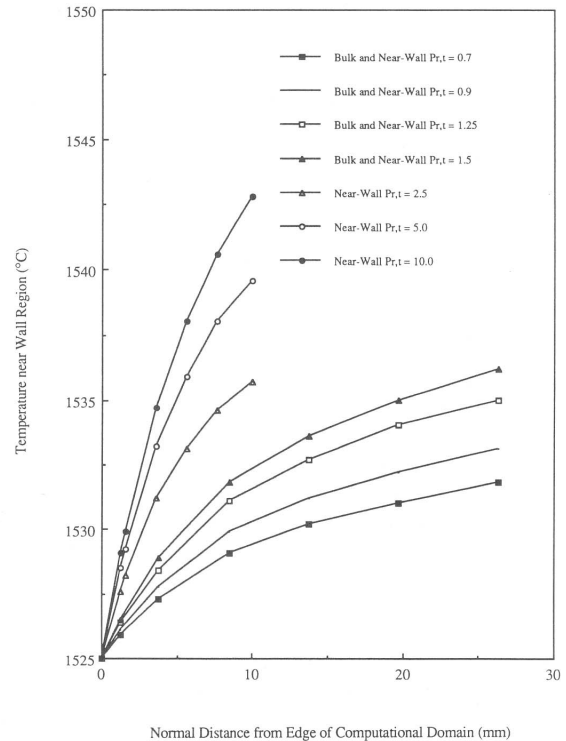
Figure 21. Effect of turbulent Prandtl number on the temperature profile near the narrow face wall (1 m below meniscus)

contour represents the location in the right half-mold, where the steel jet has lost 50% of its "superheat," which corresponds to the difference between its initial and liquidus solidification temperatures. *Figure 22(b)* shows that the increase in liquid temperature is most extreme when a thin 0.01-m strip near the wall has a large Pr_t while elsewhere Pr_t remains at 0.9. In these cases, unrealistically high temperatures are predicted. This result simply indicates that the thermal boundary layer, where Pr_t is large, is part of laminar region, which is *not* simulated in the present model. The value of Pr_t near the wall therefore should not be independently altered.

Uneven dissipation of superheat to the shell will produce a maximum heat input near the point of jet impingement. This can produce local "hot spot(s)" on the shell, where growth is slow, and may cause shell thinning and erosion and even lead to breakouts, particularly at higher casting speeds.[17] *Figure 23* shows that Pr_t has its most pronounced effect on heat flux near this critical impingement point.

Increasing Pr_t from 0.9 to 1.5 decreases heat flux at the impingement point by 20% while heat flux low in the mold remains unaffected. This is because the increase in temperature gradient low in the mold compensates for the reduction in effective thermal conductivity. Near the impingement point the temperature gradient is always high and so is less affected by the overall increase in temperature of the liquid caused by increasing Pr_t. The accompanying reduction in conductivity thus has a greater effect, and lower heat flux
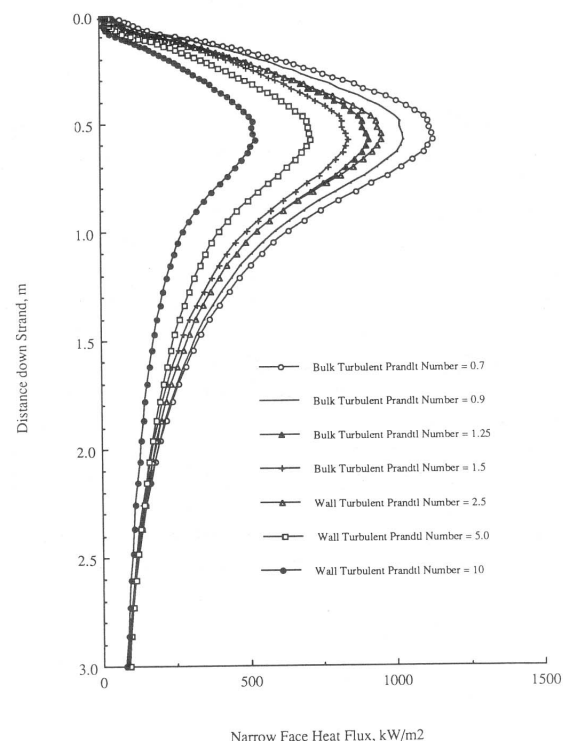
Figure 20. Effect of turbulent Prandtl number on the turbulent thermal conductivity profile near the narrow face wall (1 m below meniscus)

Table 3. Variation of heat balance with Pr$_t$ (All values are in MW/m)

	Pr$_t$ = 0.7	Pr$_t$ = 0.9	Pr$_t$ = 1.25	Pr$_t$ = 1.50
Heat in	−1.269	−1.265	−1.261	−1.260
Heat out, wall	1.221	1.153	1.048	0.984
Heat out, top	0.043	0.043	0.044	0.044
Heat out, centerline	−0.009	−0.007	−0.005	−0.004
Heat out, bottom	0.175	0.239	0.335	0.394
Balance	0.161	0.164	0.161	0.158
Deviation, %	−12.7	−13.0	−12.8	−12.5

(a)   (b)
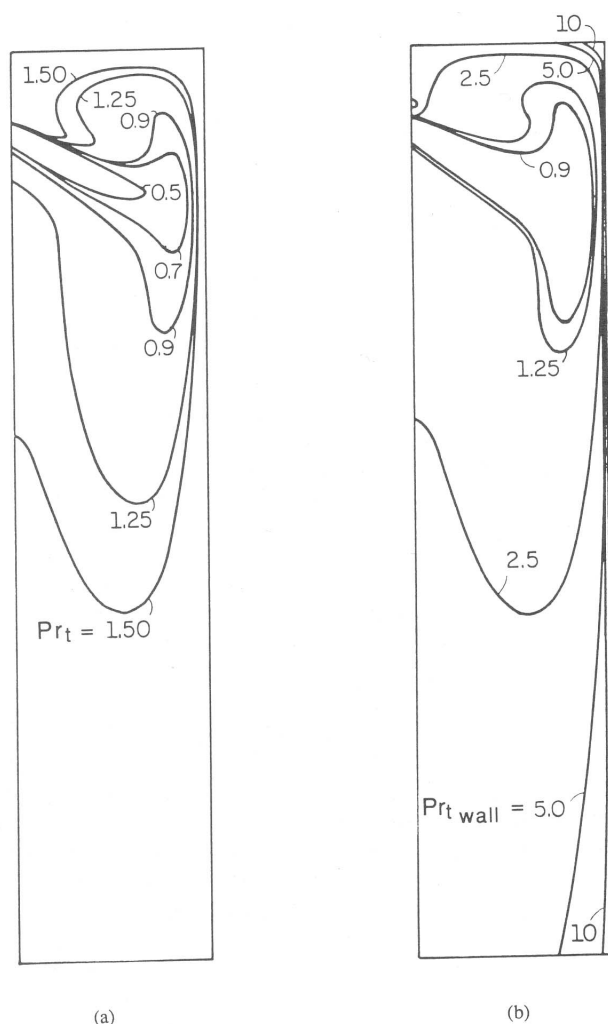
Figure 22. Effect of turbulent Prandtl number on the location of the 1537.5°C isotherm (representing 50% heat loss). (a) Equal variation in Pr$_t$ throughout domain. (b) Pr$_t$ increased only at wall (0.9 elsewhere)

Figure 23. Effect of turbulent Prandtl number on the heat flux distribution to the narrow face wall

is produced. It is interesting to observe in this figure that increasing Pr$_t$ at the wall to 1.5 reduces the heat flux for a 0.9 Pr$_t$ bulk value to almost the same level as is produced with an overall Pr$_t$ of 1.25.

The superheat in the fluid steel can be convected to and conducted through the solidifying steel shell to the copper mold walls, or it can be swept out of the mold region to be dissipated much lower in the caster. *Table 3* shows that most of the superheat is dissipated high

in the caster and that Pr$_t$ has a great influence on the extent to which this happens.

As is seen in *Table 3*, the heat leaving the bottom of the domain increases from 0.175 MW/m (14%) to 0.394 MW/m (31%) as Pr$_t$ increases from 0.7 to 1.5. The lower Pr$_t$ allows thermal diffusion to take heat out of the liquid, resulting in more heat leaving the narrow face, while higher Pr$_t$ keeps heat with the jet longer, decreasing heat flux through the wall and increasing the average temperature of the steel leaving to lower regions in the caster. It is also interesting that the heat balance in *Table 3* always calculates over 12% too much heat flowing out of the system. This difference is larger than expected from discretization errors alone.

Model verification

Having explored the effect of various modeling parameters on the solution, we compared the model predictions with experimental observations to evaluate the modeling assumptions.

Velocity predictions

To test the accuracy of the fluid velocity predictions, flow simulations were performed to model the physical water model at Inland Steel. The results were then compared with the experimental findings using the water model. *Figure 24* shows a photograph of the flow pattern in the water model resulting from a nominal 15° downward angled nozzle compared with the calculated velocities for the same conditions given in *Table 1*. The angle and shape of the jet streaming in from the submerged nozzle, the location of the impingement point on the narrow face wall, and the overall flow characteristics observed in this figure show a general similarity with those observed in the physical water model. Further details and results are given elsewhere.[21]

Heat transfer calculations

Measurements of heat transfer in turbulent metal flow systems are rare. Experimental correlations were derived for heat flux due to water jet impingement on a heated flat plate by Kumada and Mabuchi.[31] These correlations were adapted by Nakato[17] and validated to predict heat transfer due to steel jet impingement within the continuous casting mold. The same procedure was used in the present study to determine heat flux as a function of position down the narrow face shell wall:

$$q(y) = 0.0682 \, (1 + \sin \beta)^{0.8} \, \mathrm{Re}_j^{0.5} \, \mathrm{Pr}_0^{0.4}$$

$$\times \left(\frac{Z_w}{d}\right)^{-0.6} \left(\frac{|y - y_w|}{Z_w}\right)^{-0.37} \frac{k}{d} \Delta T_y \quad (30)$$

where

$$\Delta T_y = \Delta T_{y-\Delta y} - q(y - \Delta y)\frac{\Delta y}{C_p Q_j} \quad (31)$$
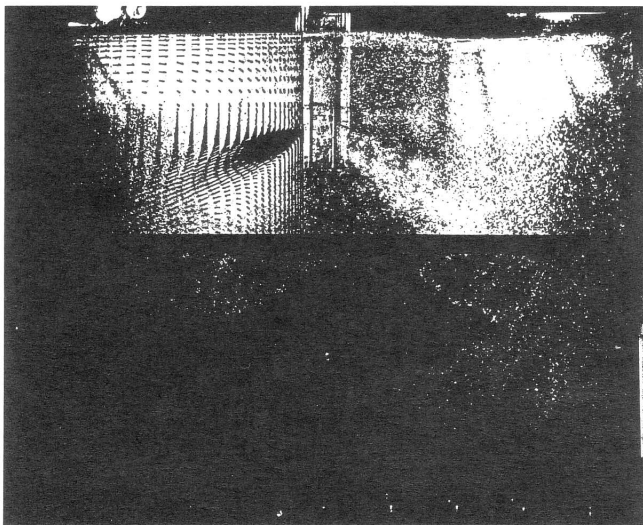
This recursive relation is started by calculating heat

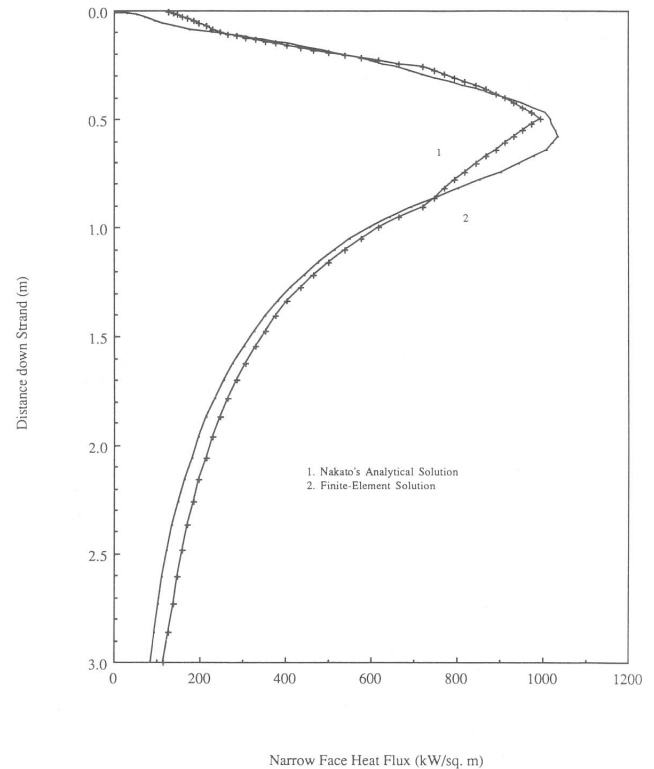**Figure 24.** Comparison of measured and calculated velocity distribution

Narrow Face Heat Flux (kW/sq. m)

Figure 25. Comparison of predicted heat flux distribution with previous estimates

flux at the jet impingement point from

$$q_w = 1.42 \, \mathrm{Re}_j^{0.58} \, \mathrm{Pr}_0^{0.43} \left(\frac{Z_w}{d}\right)^{-0.62} \frac{k}{d} \Delta T_{yw} \quad (32)$$

where

$$\Delta T_{yw} = T_{\mathrm{inlet}} - T_{\mathrm{liq}} \quad (33)$$

The maximum heat flux at the jet impingement point on the narrow face predicted by this correlation is almost $1.0 \, \mathrm{MW/m^2}$. This represents a significant fraction of the total heat flux withdrawn from the other side of the shell during solidification in the mold. *Figure 25* shows that a close match exists all the way down the mold between the heat flux predicted by this correlation and the finite element simulation, using standard conditions from *Table 1*.

This agreement implies that a turbulent Prandtl number of 0.9 combined with wall law 2 and the other assumptions made in the present model appear to be reasonable. The temperatures predicted by the finite element simulation appear to be slightly high in comparison with plant experience, a finding that might be due to the two-dimensional nature of the calculation.

Conclusions

Finite element simulation of turbulent fluid flow and heat transfer is a viable tool for modelling and understanding flow of liquid metal in metallurgical processes. The present study has presented the back-

ground for development of a turbulent fluid flow and heat transfer model of liquid steel flow in the mold region of a continuous slab caster. The model velocity predictions compare reasonably with visual observations of flow inside a physical water model, and the heat transfer results agree closely with calculations based on experimental correlations. Specific conclusions regarding the modelling procedure are as follows:

- Relaxed successive substitution with a direct solver is the best convergence strategy for this type of problem in two dimensions.
- Starting from an initial guess of zero velocity, a high relaxation factor (0.5–0.7) should be used for the first few iterations, to quickly reduce the initial error, followed by further iterations at a lower *RF* for increased stability. Following the *history* of the residual error is the best indicator of convergence. When large fluctuations appear in the residual error history, this indicates that the percent difference between successive solution vectors is too high and that the solution is bad.
- The model results are sensitive to values of velocity, K, and ϵ at the inlet, so the simulation should extend as far upstream as necessary to simulate the proper flow conditions entering the domain of interest.
- The error introduced by using upwinding to improve computational stability is small.
- Significant discrepancy in the flow simulation can be obtained by improper use of K and ϵ boundary conditions. For the present problem, involving flow with recirculation zones and reattachment and stagnation regions, wall law 2 produces more realistic behavior, particularly for heat transfer computation.
- Better computational efficiency (by a wide margin) is achieved by fixing K and ϵ values at the inlet (rather than fixing their gradients to zero), although both methods produced similar results.
- Increasing turbulent Prandtl number significantly increases overall temperatures and decreases heat flux through its effect of decreasing the turbulent conductivity. Good results were obtained by using the standard value of 0.9 throughout the computational domain.

If sufficient care is taken, finite element turbulence models are capable of reproducing the flow phenomena observed in a metallurgical system, such as a continuous slab-casting mold. Furthermore, a two-dimensional model can produce adequate velocity results if the essential flow characteristics of the physical system are found in the simulated plane.

Acknowledgments

Nomenclature

A_m, A_γ	Damping factor functions[30]
C_p	Specific heat (J/(kg K))
d	Diameter of fluid jet, 0.086 m
D	Equivalent pipe diameter of caster, 0.352 m
E	Wall roughness constant, 9.0
$\{F\}$	Force vector
Gr	Grashoff number
k	Thermal conductivity (W/(mK))
K	Turbulent kinetic energy (m^2/s^2)
$[K]$	Global stiffness matrix
Pe	Peclet number, Pe = Re Pr
Pr	Prandtl number = $\mu\,C_p/k$
Q_j	Jet flow rate = $\rho\pi d^2/4V_{\text{jet}}$
q_w	Heat flux at impingement point on narrow face wall (kW/m^2)
$q(y)$	Heat flux along narrow face wall (kW/m^2)
Re	Reynolds number = $VD\rho/\mu_0$
Re$_j$	Jet Reynolds number = $V_{\text{jet}}\,d\rho/\mu_0$
Re$_g$	Grid Reynolds number = $V\,\xi\,\rho/\mu_0$
RF	Relaxation factor, $0 < RF < 1$
$\{R\}$	Residual vector
T_{inlet}	Inlet temperature = 1550°C
T_{liq}	Liquidus temperature = 1525°C
$\{U\}$	Solution vector, $\{v_x, v_y, K, \epsilon, T\}$
V	Speed (m/s)
v_i	Velocity component (m/s)
v^*	Friction velocity (m/s)
x	Cartesian coordinate distance from centerline of caster (m)
x_2	Normal distance from edge of computational domain (m)
x_n	Offset distance of computational domain from wall, 0.01 m
y	Cartesian coordinate distance below meniscus (m)
y_w	y-coordinate of impingement point, 0.55 m
Z_w	Distance between nozzle and impingement point, 0.67 m

Greek symbols

β	Jet angle = 15°
Δy	Grid size in y-direction (m)
ϵ	Viscous dissipation (m^2/s^3)
κ	von Karman constant, 0.41
κ_m	Mixing-length constant for momentum, 0.40
κ_γ	Mixing-length constant for heat, 0.44
μ	Viscosity (kg/ms)
ρ	Density (kg/m^3)
σ_K, σ_ϵ	Turbulent "Prandtl numbers" for diffusion of K and ϵ

Φ   Source term in equations (6)–(8) proportional to generation of K

ξ   Element width (m)

Subscripts

0   molecular (laminar) value

t turbulent value

eff   effective value, (sum of laminar and turbulent components)

w impingement point

References

1 Sobolewski, R. and Hurtuk, D. J. Water modelling of slab caster flow conditions. *Second Process Technology Conference Proceedings*, Vol. 2, 1982, pp. 160–165

2 Ferretti, A., Podrini, M. and Si Schino, G. Submerged nozzle optimization to improve stainless steel surface quality at Terni steelworks. *Steelmaking Proc.* 1985, **68**, 49–57

3 McPherson, N. A. Continuous cast clean steel. *Steelmaking Proc.* 1985, **68**, 13–25

4 Guthrie, R. I. L. Physical and mathematical models for ladle metallurgy operations. *Mathematical Modelling of Materials Processing Operations*, ed. J. Szekely, L. B. Hales, H. Henein, N. Jarrett, K. Rajamani and I. Samarasekera. TMS Conference Proceedings, Palm Springs, Calif., Nov. 29, 1987, pp. 447–482

5 Robertson, T., Moore, P. and Hawkins, J. F. Computational flow model as aid to solution of fluid flow problems in the steel industry. *Ironmaking and Steelmaking* 1986, **13**(4), 195–203

6 Sahai, Y. Computer simulation of melt flow control due to baffles with holes in continuous casting tundishes. *Mathematical Modelling of Materials Processing Operations*, ed. J. Szekely, L. B. Hales, H. Henein, N. Jarrett, K. Rajamani and I. Samarasekera. TMS Conference Proceedings, Palm Springs, Calif., Nov. 29, 1987, pp. 431–445

7 Yao, M., Ichimiya, M., Syozo, K., Suzuki, K., Sugiyama, K. and Mesaki, R. Three dimensional analysis of molten metal flow in continuous casting mould. *Steelmaking Proc.* 1985, **68**, 27–34

8 Spitzer, K., Dubke, M. and Schwertfeger, K. Rotational electromagnetic stirring in continuous casting of round strands. *Metallurg. Trans. B*, 1986, **17B**, 119

9 Scharwz, M. P. and Turner, W. J. Applicability of the standard K-ϵ turbulence model to gas stirred baths. *Appl. Math. Modelling* 1988, **12**, 273–279

10 Robertson, T. and Perkins, A. Physical and mathematical modelling of liquid steel temperature in continuous casting. *Ironmaking and Steelmaking* 1986, **13**(6), 301–310

11 Ilegbusi, O. J. and Szekely, J. The modelling of fluid flow, tracer dispersion and inclusion behavior in tundishes. *Mathematical Modelling of Materials Processing Operations*, ed. J. Szekely, L. B. Hales, H. Henein, N. Jarrett, K. Rajamani

and I. Samarasekera. TMS Conference Proceedings, Palm Springs, Calif., Nov. 29, 1987, pp. 409–429

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

13 Rodi, W. *Turbulence Models and Their Application in Hydraulics—A State of the Art Review*. University of Karlsruhe, Karlsruhe, Germany, 1980

14 Chiesa, F. M. and Guthrie, R. I. L. Natural convective heat transfer rates during the solidification and melting of metals and alloy systems. *J. Heat Transfer* 1974, **96**(3), 377–384

15 Engelman, M. S. *FIDAP Theoretical Manual—Revision 4.5*. Fluid Dynamics International, Inc., Evanston, Ill., 1989

16 Brooks, A. N. and Thomas, T. J. R. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Comput. Methods Appl. Mechanics Engrg.* 1982, **32**, 199–259

17 Nakato, H., Ozawa, M., Kinoshita, K., Habu, Y. and Emi, T. Factors affecting the formation of shell and longitudinal cracks in mold during high speed continuous casting of slabs. *Trans. Iron Steel Inst. Japan* 1984, **24**(11), 957–965

18 Prandtl, L. and Tietjens, O. *Applied Hydro- and Aeromechanics*. McGraw-Hill, New York, 1934

19 Schlichting, H. *Boundary-Layer Theory*. McGraw-Hill, New York, 1955

20 Haroutunian, V. Turbulent flows with FIDAP. Seminar Notes, Fluid Dynamics International, 1988

21 Thomas, B. G., Mika, L. J. and Najjar, F. M. Simulation of fluid flow inside a continuous slab casting machine. *Metallurg. Trans. B*, 1990, **21B**, 387–400

22 Ortega, J. M. and Rheinboldt, W. C. *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, New York, 1980

23 Smith, S. L. and Brebbia, C. A. Improved stability techniques for the solution of Navier-Stokes equations. *Appl. Math. Modelling* 1977, **1**, 226–234

24 Carey, G. F. and Plover, T. Variable upwinding and adaptive mesh refinement in convection-diffusion. *Internat. J. Numer. Methods Engrg.* 1983, **19**, 341–353

25 Huyakorn, P. S. Solution of steady-state, convective transport equation using an upwind finite element scheme. *Appl. Math. Modelling* 1977, **1**, 187–195

26 Cook, R. D. *Concepts and Applications of Finite Element Analysis*. John Wiley, New York, 1981

27 Hinze, J. O. *Turbulence*. McGraw-Hill, New York, 1959

28 Yakhot, V., Orszag, S. A. and Yakhot, A. Heat transfer in turbulent fluids. I: Pipe Flow. *Internat. J. Heat Mass Transfer* 1987, **30**(1), 15–22

29 Blom, J. Experimental determination of the turbulent Prandtl number in a developing temperature boundary layer. *Proceedings of the Fourth International Heat Transfer Conference Paris-Versailles 1970*, Vol. 2, Paper FC 2.2, Elsevier, Amsterdam, 1970

30 Cebeci, T. A model for eddy conductivity and turbulent Prandtl number. *J. Heat Transfer, Trans. ASME* 1973, **95**, 227–234

31 Kumada, M. and Mabuchi, I. *Trans. Japan. Soc. Mech. Engrg.* 1969, **35**, 1053–1061