INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2001; **37**: 279–295 (DOI: 10.1002/fld.171)

Practical techniques for a three-dimensional FEM analysis of incompressible fluid flow contained with slip walls and a downstream tube bundle

Yuzuru Eguchi*[,1]

Hydraulics Department, Central Research Institute of Electric Power Industry, Abiko-shi, Chiba, Japan

SUMMARY

Two practical techniques are proposed in this paper to simulate a flow contained in a plenum with a downstream tube bundle under a PC environment. First, a technique to impose slip wall conditions on smooth-faced planes and sharp edges is proposed to compensate for the mesh coarseness relative to boundary layer thickness. In particular, a new type of Poisson equation is formulated to simultaneously satisfy both such velocity boundary conditions on walls and the incompressibility constraint. Second, a numerical model for a downstream tube bundle is proposed, where hydraulic resistance in a tube is imposed as a traction boundary condition on a fluid surface contacting the tube bundle end. The effectiveness of the techniques is numerically demonstrated in the application to a flow in a condenser water box. Copyright © 2001 John Wiley & Sons, Ltd.

KEY WORDS: boundary condition; finite element method; Poisson equation; slip edge; slip wall; tube bundle

1. INTRODUCTION

Since the capabilities of recent PCs come to exceed those of supercomputers used a decade ago, it is now possible to solve practical problems of complicated three-dimensional geometries at hand with a PC. In the present paper, the author proposes two finite element techniques suitable for a practical flow simulation under a PC environment, with special emphasis on an application to a flow in a condenser water box.

In a PC computation, mesh subdivision must be still coarse in comparison with high-performance computers due to the relatively small main memory capacity and relatively slow CPU speed. In such a situation, the physical boundary layer thickness is much smaller than the mesh size, especially in a high-*Re* flow, and imposition of the slip wall condition can be more

* Correspondence to: Hydraulics Department, Central Research Institute of Electric Power Industry, 1646 Abiko, Abiko-shi, Chiba 270-1194, Japan. Tel.: +81 471 821181; fax: +81 471 847142.
[1] E-mail: eguchi@criepi.denken.or.jp

Received 26 May 2000
Revised 12 December 2000

appropriate than that of the non-slip wall condition. Engelman *et al.* [1] proposed an implementation of a slip wall condition, where velocity vectors and momentum equations on a boundary are transformed into those of the other co-ordinates aligned to the boundary normal direction. However, in the literature there are on specific remarks on the treatment for an external sharp edge in three-dimensional configuration, where unrealistic velocity would be produced with straightforward application of the slip wall condition, as shown in Figure 1. In the present paper, the author proposes a technique for imposition of the slip condition, which is applicable not only to flat walls but also to sharp edges.

The other unique technique proposed here is a numerical model for a tube bundle downstream of a condenser water box. In the model, the traction, which is estimated with the pressure loss due to the flow resistance in a tube, is given on fluid surfaces facing the upstream end of the tube bundle. A numerical example is presented to show the effectiveness of the formulation.

In Section 2, theoretical aspects are presented, including treatments of the slip conditions, a time integration scheme, a new type of Poisson equation and a tube bundle model. In Section 3, some numerical results obtained on a PC are shown, putting stresses on the algorithm/code verification, effect of the wall boundary condition (BC) and an application to a flow in a condenser water box. Conclusions are drawn in Section 4.

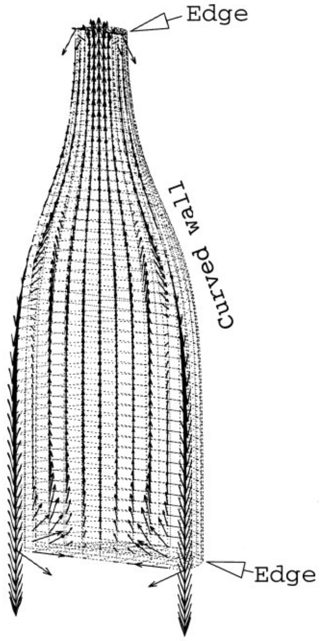

Figure 1. Natural convection in a bottle with sharp edges.

2. THEORETICAL BACKGROUND

2.1. Basic equations

The incompressible transient Navier–Stokes equations are written as follows:

$$\dot{u}_i + u_j u_{i,j} + p_{,i} - \nu_{\text{eff}} u_{i,j,j} = f_i \tag{1}$$

$$u_{i,i} = 0 \tag{2}$$

where u_i is the ith ($i = 1, 2, 3$) velocity component; p is the kinematic pressure (pressure divided by fluid density); f_i is the body force per unit mass; the superposed dot denotes the temporal derivative; $(\ )_{,j}$ denotes the spatial derivative in the jth direction. Summation convention is used for the repeated subscripts. An effective kinematic viscosity, ν_{eff}, is used instead of a molecular kinematic viscosity, ν, to formally represent turbulence, though no turbulent model such as the $k - \varepsilon$ model is introduced so far. The following BCs are assumed to supplement Equations (1) and (2):

$$u_i = u_i^{\text{BC}} \quad \text{on } \Gamma_u \tag{3a}$$

$$u_i n_i = 0, \qquad \chi_i t_i^1 = 0, \qquad \chi_i t_i^2 = 0 \quad \text{on } \Gamma_{\text{s-w}} \tag{3b-i}$$

$$u_i n_i^A = 0, \qquad u_i n_i^B = 0, \qquad \chi_i t_i^{AB} = 0 \quad \text{on } \Gamma_{\text{s-e}} \tag{3b-ii}$$

$$\chi_j = \chi_i^{\text{BC}} \quad \text{on } \Gamma_\chi \tag{3c}$$

where traction divided by fluid density, χ_i, is defined as follows:

$$\chi_i = (-p\delta_{ij} + \nu_{\text{eff}} u_{i,j}) n_j \tag{4}$$

In the above, Γ_u denotes the velocity-prescribed boundary including non-slip walls; $\Gamma_{\text{s-w}}$ denotes the slip boundary except 90°-type slip edge (see Section 2.3.2); $\Gamma_{\text{s-e}}$ denotes the 90°-type slip edge; Γ_χ denotes the traction-prescribed boundary; n_j stands for the jth component of an outward unit normal vector on boundary; t_i^1 and t_i^2 are the unit tangential vectors on a boundary surface; n_i^A and n_i^B stand for outward unit normal vectors on two planes A and B, forming a 90°-type edge; t_j^{AB} is a unit vector parallel to the ridge of the 90°-type edge or normal to n_i^A and n_i^B; and δ_{ij} is the Kronecker delta. It is assumed the following relations hold for the above three types of boundaries:

$$\Gamma_u \cup \Gamma_{\text{s}} \ (\equiv \Gamma_{\text{s-w}} \cup \Gamma_{\text{s-e}}) \cup \Gamma_\chi = \Gamma_{\text{all}} \quad \text{(whole boundary)} \tag{5a}$$

$$\Gamma_u \cap \Gamma_{\text{s}} = \varnothing \ \text{(null)}, \quad \Gamma_{\text{s}} \cap \Gamma_\chi = \varnothing, \quad \Gamma_u \cap \Gamma_\chi = \varnothing \tag{5b}$$

It is also assumed that the prescribed velocity, u_j^{BC}, is independent of time t, i.e. $\dot{u}_i^{\mathrm{BC}} = 0$, which means all boundaries are fixed in space.

2.2. Spatial discretization

The streamline upwind/Petrov–Galerkin (SUPG) method [2] is used to formulate the weighted residual forms of Equations (1) and (2), though all the computations presented in the paper were performed with the Galerkin formulation by reducing the discontinuous weighting function of the SUPG method to zero. For the spatial discretization, the trial solution for velocity, u_i, and the continuous weighting function for the momentum equations, w_k, are approximated as $\Phi_\alpha u_{i\alpha}$ and $\Phi_\alpha w_{k\alpha}$ with a tri-linear interpolation function Φ_α (α is node number). The trial solution for pressure, p, and the weighting function for the continuity equation, q, are approximated as $\Psi_\gamma p_\gamma$ and $\Psi_\gamma q_\gamma$ with a piecewise constant function, Ψ_γ (γ is element number). The spatially discretized equations are finally expressed in the following matrix form

$$\boldsymbol{M}\dot{\boldsymbol{v}} + \boldsymbol{B}\boldsymbol{v} + \boldsymbol{K}\boldsymbol{v} - \boldsymbol{C}\boldsymbol{p} = \boldsymbol{f}_{\mathrm{b}} + \boldsymbol{f}_{\mathrm{t}} \tag{6}$$

$$\boldsymbol{C}^{\boldsymbol{T}}\boldsymbol{v} = \boldsymbol{0} \tag{7}$$

where $\boldsymbol{M}$ is the mass matrix; $\boldsymbol{B}$ is the convection matrix; $\boldsymbol{K}$ is the diffusion matrix; $\boldsymbol{C}$ is the gradient matrix; $\boldsymbol{f}_{\mathrm{b}}$ is the body force vector; $\boldsymbol{f}_{\mathrm{t}}$ is the traction vector; $\boldsymbol{C}^{\boldsymbol{T}}$ is the divergence matrix; $\boldsymbol{p}$ is a vector composed of unknown pressures of all elements, p_γ. Velocity vector $\boldsymbol{v}$ is composed of all nodal velocities $u_{i\alpha}$, both specified and unspecified. The velocity BCs are introduced in a time integration process, which is explained in the subsequent two sections.

2.3. Imposition of slip and non-slip wall conditions

2.3.1. Slip wall treatment. A slip velocity, $\boldsymbol{V}_{\mathrm{s}}$, can be calculated as follows by subtracting the normal component from an arbitrary velocity, $\boldsymbol{V}^*$:

$$\boldsymbol{V}_{\mathrm{s}} = \boldsymbol{V}^* - (\boldsymbol{V}^* \cdot \boldsymbol{n})\boldsymbol{n} \tag{8}$$

where $\boldsymbol{n} = (n_x, n_y, n_z)$ is a unit normal vector on a slip wall; $(\boldsymbol{a} \cdot \boldsymbol{b})$ denotes inner product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$. The above equation indicates that the slip wall condition can be fulfilled by pre-multiplying a 3 × 1 velocity vector at node α by the following 3 × 3 projection matrix, $\boldsymbol{H}_\alpha$:

$$\boldsymbol{H}_\alpha = \begin{bmatrix} 1 - n_x^2 & -n_y n_x & -n_z n_x \\ -n_x n_y & 1 - n_y^2 & -n_z n_y \\ -n_x n_z & -n_y n_z & 1 - n_z^2 \end{bmatrix} \tag{9}$$

2.3.2. Edge treatment. Special care has to be taken to singular configurations such as edges where one cannot straightforwardly define the normal vector. In the paper, singular configurations are classified into two types, i.e. 270°-type internal edge and 90°-type external edge as shown in Figure 2(a) and (b) respectively. Velocities on a 270°-type edge are treated in the same manner as the slip wall explained in the above, being of 2 df. Attention has to be paid to the selection of a normal vector, because an inappropriate choice of the normal vector leads to net inflow or net outflow of mass through the edge walls. Engelman *et al.* [1] proposed a method to define the normal vector, which is consistent with mass balance. In the present study, the unit normal vector $\boldsymbol{n}_\alpha$ at node α on a 270°-type edge is similarly calculated as follows:

$$\boldsymbol{n}_\alpha = \sum_m \boldsymbol{S}_m \boldsymbol{n}_m \Bigg/ \left| \sum_m \boldsymbol{S}_m \boldsymbol{n}_m \right| \tag{10}$$

where $\boldsymbol{S}_m$ and $\boldsymbol{n}_m$ are the area of mth element surface sharing a velocity node α and its outward unit normal vector respectively. $|\boldsymbol{a}|$ denotes the length of vector $\boldsymbol{a}$.

On the other hand, imposition of the slip wall condition on a 90°-type edge would produce unrealistic velocity vectors, even though mass is conserved. An example is shown in Figure 1, where we can see the fluid penetrates the fluid boundary at the bottom of the bottle and at the top free surface. The author proposes a unique technique to circumvent such unrealistic flow. That is, velocities on the 90°-type edges are assumed to slip only along the edge line (ridge) in one-dimensional manner. This condition can be fulfilled by doubly pre-multiplying a velocity at node α by the projection matrices, $\boldsymbol{H}_\alpha^A$ and $\boldsymbol{H}_\alpha^B$. The matrices $\boldsymbol{H}_\alpha^A$ and $\boldsymbol{H}_\alpha^B$ denote the slip wall matrices defined by Equation (9), where the unit normal vectors standing on face A and B are used respectively, assuming faces A and B form a 90°-type edge at node α. Then the projection matrix for the 90°-type edges, $\boldsymbol{W}_\alpha$ is defined as follows:

$$\boldsymbol{W}_\alpha = \boldsymbol{H}_\alpha^A \boldsymbol{H}_\alpha^B \tag{11}$$

2.3.3. BC imposition matrix. The velocity-prescribed condition such as non-slip wall condition can be imposed by pre-multiplying an acceleration increment on such boundaries by the zero projection matrix. Then, a global matrix $\boldsymbol{G}$ to impose various velocity BCs can be defined as follows, and is used in a time integration process:

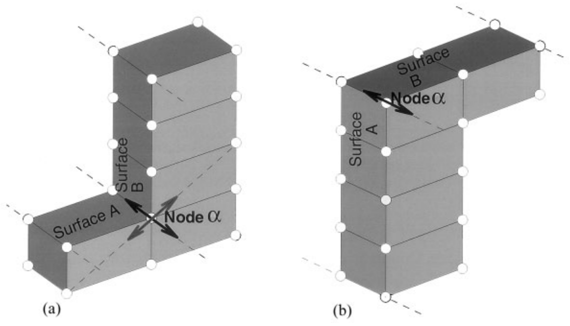

Figure 2. Permitted df of velocity at a node on an edge (an arrow represents a free velocity component). (a) Slice of 270°-type edge. (b) Slice of 90°-type edge.

$$\boldsymbol{G}=\begin{bmatrix}
\boldsymbol{O}_1 & \boldsymbol{O} & & & & \cdots & & & & & \boldsymbol{O}\\
& \cdots & & & & & & & & & \\
& & \boldsymbol{O}_1 & \boldsymbol{O} & & \cdots & & & & & \boldsymbol{O}\\
& & & \boldsymbol{W}_1 & \boldsymbol{O} & & \cdots & & & & \boldsymbol{O}\\
& & & & \cdots & & & & & & \\
& & & & & \boldsymbol{W}_j & \boldsymbol{O} & \cdots & & & \boldsymbol{O}\\
& & & & & & \boldsymbol{H}_1 & \boldsymbol{O} & \cdots & & \boldsymbol{O}\\
& & & & & & & \cdots & & & \\
\text{Sym.} & & & & & & & & \boldsymbol{H}_K & \boldsymbol{O} \cdots & \boldsymbol{O}\\
& & & & & & & & & \boldsymbol{E}_1\ \boldsymbol{O}\cdots & \boldsymbol{O}\\
& & & & & & & & & \cdots & \\
& & & & & & & & & & \boldsymbol{E}_N
\end{bmatrix} \tag{12}$$

In the above matrix, $\boldsymbol{O}_\alpha$ is the 3 × 3 zero matrix; $\boldsymbol{E}_\alpha$ is the 3 × 3 identity matrix; I is the number of velocity nodes on velocity-prescribed boundaries, Γ_u; J is the number of the velocity nodes on the 90°-type slip edge, $\Gamma_{\text{s-e}}$; K is the number of velocity nodes on slip walls, $\Gamma_{\text{s-w}}$, including the 270°-type edge; and N is the number of the other unconstrained velocity nodes. In an actual computation, the order of the sub-matrices $\boldsymbol{O}_\alpha$, $\boldsymbol{W}_\alpha$, $\boldsymbol{H}_\alpha$ and $\boldsymbol{E}_\alpha$ must be shuffled depending on the velocity type at node α, i.e. whether node α is located on Γ_u, or on $\Gamma_{\text{s-e}}$, or on $\Gamma_{\text{s-w}}$ or otherwise respectively.

2.4. Time integration scheme

2.4.1. Fully explicit time integration. The ordinary differential equation (ODE) system (6) must be integrated in time with satisfying the incompressibility constraint (7). Usually the projection method (whose family is named as the fractional method, the splitting method, the pressure correction method, etc.) is used to uncouple the incompressibility constraint instead of solving fully coupled large system, i.e. equation system (6) is implicitly or explicitly integrated in time to yield intermediate velocity, and then the intermediate velocity is projected into a divergence-free subspace by solving a pressure Poisson equation.

In an implicit method, the computational cost for matrix inversion must be suppressed for a practical computation, keeping its inherent phase accuracy. For example, Gresho and Chan [3] proposed semi-implicit (explicit for convection terms, implicit for viscous terms) schemes to realise computational efficiency by strategically utilizing a lumped mass matrix in a part of discretized Navier–Stokes equations. In an explicit method, phase errors associated with a lumped mass matrix must be suppressed. An example is the predictor/multi-corrector scheme proposed by Brooks and Hughes [2], where the consistent mass is partially used to reduce the phase errors.

In the present study, the predictor/multi-corrector scheme is used for the time integration because the use of the BC matrix (12) is compatible with a fully explicit scheme. Starting with a velocity, $\boldsymbol{v}_0$, satisfying the velocity BCs, and an initial pressure, $\boldsymbol{p}_0$, the time integration scheme combined with the BC imposition matrix $\boldsymbol{G}$ is summarized as follows:

[O] Set-up of initial condition
1. initialize time and time step counter: $t = 0$, $n = 0$

```
2. set an initial velocity and an initial pressure: $\boldsymbol{v}(n) = \boldsymbol{v}_0$, $\boldsymbol{p}(n) = \boldsymbol{p}_0$
3. compute the initial acceleration: $\boldsymbol{a}(n) = \underline{\boldsymbol{M}}^{-1}(\boldsymbol{f}_\mathrm{b} + \boldsymbol{f}_\mathrm{t} - \boldsymbol{B}\boldsymbol{v}_0 - \boldsymbol{K}\boldsymbol{v}_0 + \boldsymbol{C}\boldsymbol{p}_0)$
4. impose the velocity BC: $\boldsymbol{a}(n) = \boldsymbol{G}\boldsymbol{a}(n)$

[i] Predictor phase
1. reset inner-iteration counter: $k = 0$
2. set initial guesses of velocity, acceleration and pressure at the next time step, $n + 1$

       $\boldsymbol{v}^k = \boldsymbol{v}(n) + \Delta\boldsymbol{t}(1-\gamma)\boldsymbol{a}(n), \quad \boldsymbol{a}^k = 0, \quad \boldsymbol{p}^k = \boldsymbol{p}(n)$

[ii] Corrector phase
1. compute temporary acceleration increment

       $\Delta\boldsymbol{a}^{*k} = \underline{\boldsymbol{M}}^{-1}(\boldsymbol{f}_\mathrm{b} + \boldsymbol{f}_\mathrm{t} - \boldsymbol{M}\boldsymbol{a}^k - \boldsymbol{K}\boldsymbol{v}^k - \boldsymbol{B}\boldsymbol{v}^k + \boldsymbol{C}\boldsymbol{p}^k)$

2. impose velocity BC; $\Delta\boldsymbol{a}^{*k} = \boldsymbol{G}\Delta\boldsymbol{a}^{*k}$
3. compute temporary velocity; $\boldsymbol{v}^{*k} = \boldsymbol{v}^k + \gamma\Delta t\Delta\boldsymbol{a}^{*k}$
4. solve pressure Poisson equation; $\boldsymbol{C}^{\boldsymbol{T}}\boldsymbol{G}\underline{\boldsymbol{M}}^{-1}\boldsymbol{C}\Delta\boldsymbol{p}^k = -\boldsymbol{C}^{\boldsymbol{T}}\boldsymbol{v}^{*k}/\gamma\Delta t$
5. correct acceleration increment; $\Delta\boldsymbol{a}^k = \Delta\boldsymbol{a}^{*k} + \boldsymbol{G}\underline{\boldsymbol{M}}^{-1}\boldsymbol{C}\Delta\boldsymbol{p}^k$
6. correct velocity; $\boldsymbol{v}^{k+1} = \boldsymbol{v}^k + \gamma\Delta t\Delta\boldsymbol{a}^k$
7. correct acceleration; $\boldsymbol{a}^{k+1} = \boldsymbol{a}^k + \Delta\boldsymbol{a}^k$
8. correct pressure; $\boldsymbol{p}^{k+1} = \boldsymbol{p}^k + \Delta\boldsymbol{p}^k$
9. IF $|\Delta\boldsymbol{a}^k| > \varepsilon$ (not converged) AND $k + 1 < K_\mathrm{max}$,
     THEN: bump inner iteration step, $k$, and iterate [ii]
     ELSE: advance time; $t = t + \Delta t$, $\boldsymbol{v}(n+1) = \boldsymbol{v}^{k+1}$, $\boldsymbol{a}(n+1) = \boldsymbol{a}^{k+1}$, $\boldsymbol{p}(n+1) = \boldsymbol{p}^{k+1}$,
     bump time step, $n$, and return to [i] (go to the next time step)
```

In the above, $\underline{\boldsymbol{M}}$ is a lumped mass matrix; $\boldsymbol{M}$ is a consistent mass matrix; γ is a time integration parameter; ε is a convergence criterion; K_max is a prescribed upper limit of the multi-corrector iteration.

The consistent mass matrix $\boldsymbol{M}$ appearing in the right-hand side of the equation in step 1 in the corrector phase contributes toward reducing the phase error as numerically shown in the literature [2]. The time integration parameter, γ, affects the accuracy and stability of the integration. If the parameter is set at 1/2 and if the multi-corrector iteration converges, the solution should be almost the same as the one obtained by the trapezoidal rule for velocities and fully implicit rule for pressure with the consistent mass matrix.

2.4.2. A new type of pressure Poisson equation. The unique Poisson equation appearing at step 4 in the corrector phase is rewritten as follows

$$[\boldsymbol{C}^{\boldsymbol{T}}\boldsymbol{G}\underline{\boldsymbol{M}}^{-1}\boldsymbol{C}]\{\Delta\boldsymbol{p}^k\} = \{-\boldsymbol{C}^{\boldsymbol{T}}\boldsymbol{v}^{*k}/\gamma\Delta t\} \tag{13}$$

The above equation is formed by the requirement of the velocity BCs for the corrected acceleration increment, $\Delta\boldsymbol{a}^k$, and the requirement of the divergence-free condition for the corrected velocity, $\boldsymbol{v}^{k+1}$. The Poisson equation is iteratively solved by the preconditioned conjugate gradient (PCG) method with the diagonal scaling. The solution, $\boldsymbol{v}^{k+1}$, naturally satisfies both the velocity BCs specified and the incompressibility constraint.

The Dirichlet BCs for velocity are usually imposed by discarding momentum equations corresponding to the specified velocities and by transposing matrix elements relating to the velocity-prescribed BCs into the right-hand side (e.g. Gresho and Chan [3]). The counterparts of Equations (6) and (7) processed with these treatments are expressed as follows

$$M'\dot{u} + \boldsymbol{B}'\boldsymbol{u} + \boldsymbol{K}'\boldsymbol{u} - \boldsymbol{C}'\boldsymbol{p} = \boldsymbol{f} \tag{6$'$}$$

$$C'^{\boldsymbol{T}}\boldsymbol{u} = \boldsymbol{g} \tag{7$'$}$$

where $\boldsymbol{u}$ denotes a vector containing only unspecified velocities, and prime attached to a matrix denotes condensation by eliminating rows and/or columns corresponding to velocity BCs. The right-hand side vectors, $\boldsymbol{f}$ and $\boldsymbol{g}$, contain the transposed portions corresponding to the BCs. Combining Equations (6′) and (7′) yields the conventional pressure Poisson equation as follows:

$$[\boldsymbol{C}'^{\boldsymbol{T}}\boldsymbol{M}'^{-1}\boldsymbol{C}']\{\boldsymbol{p}\} = \{\dot{\boldsymbol{g}} - \boldsymbol{C}'^{\boldsymbol{T}}\boldsymbol{M}'^{-1}(\boldsymbol{f} - \boldsymbol{B}'\boldsymbol{u} - \boldsymbol{K}'\boldsymbol{u})\} \tag{14}$$

In contrast with the conventional pressure Poisson equation, Equation (14), no matrix condensation is needed in Equation (13) in the present method. Furthermore, though the matrix $[\boldsymbol{C}^{\boldsymbol{T}}\boldsymbol{G}\underline{\boldsymbol{M}}^{-1}\boldsymbol{C}]$ seems non-symmetric at first glance, it can be shown that matrix $[\boldsymbol{C}^{\boldsymbol{T}}\boldsymbol{G}\underline{\boldsymbol{M}}^{-1}\boldsymbol{C}]$ is symmetric and positive-semi-definite as well as the conventional ones. These properties allow to use most iterative solvers (e.g. Gresho *et al.* [4]) and to implement slip/non-slip conditions in an existing code without major modification such as matrix condensation and local co-ordinates transformation.

2.5. Model of a downstream tube bundle

A tube bundle downstream of a water box is modeled in this paper by imposing the pressure of flow resistance in tubes, onto fluid surfaces contacting the tube bundle end. Since one cannot specify pressures as the Dirichlet BC in a pressure Poisson equation (e.g. Gresho *et al.* [5]), the traction, instead of the pressure, is imposed on the fluid surfaces through the traction vector, $\boldsymbol{f}_t$. Through a numerical result imposed with the traction is not completely identical to that with the pressure for low *Re* flow, as shown by Yagawa and Eguchi [6], the difference between both is negligibly small in most of the flows arising in a heat exchanger because the inertial term usually dominates over the viscous term. Then, the traction (divided by density, ρ) on the fluid surface contacting the tube bundle end, χ_i^{BC}, is approximated by the following:

$$\chi_i^{\mathrm{BC}} = -(\Delta P/\rho)n_i \tag{15}$$

The pressure loss due to the flow resistance, ΔP, is calculated as follows with the aid of an empirical formula for the pressure loss in a tube:

$$\Delta P = 0.5(\zeta_{\text{in}} + \zeta_{\text{out}} + \zeta_{\text{tube}})\rho\,|U_{\text{tube}}|U_{\text{tube}} \tag{16a}$$

$$\zeta_{\text{tube}} = L/d(0.79 \ln Re_{\text{t}} - 1.64)^2 \tag{16b}$$

where ζ_{in} is an inlet loss coefficient (e.g. 0.5); ζ_{out} is an outlet loss coefficient (e.g. 1.0); ζ_{tube} is a friction loss coefficient; L is a tube length; d is a tube diameter; Re_t is the tube Reynolds number defined as $U_{\text{tube}}d/v$. The velocity in a tube, U_{tube}, is calculated by $U_{\text{tube}} = U_{\text{n}}/\beta$, where U_{n} is a normal velocity averaged over an element surface contacting a tube bundle end, and β is the ratio of the total tube hole areas against the fluid surface area. Then, the traction vector due to the tube resistance is expressed by the following:

$$\boldsymbol{f}_{\text{t}} = -\int_{\Gamma\chi} \Phi_\alpha(\Delta P/\rho)n_i \,\mathrm{d}\Gamma = -\int_{\Gamma\chi} \Phi_\alpha 0.5(\zeta_{\text{in}} + \zeta_{\text{out}} + \zeta_{\text{tube}})|U_{\text{n}}|U_{\text{n}}n_i/\beta^2 \,\mathrm{d}\Gamma \tag{17}$$

The traction force (17) is treated in a semi-implicit manner in step 1 in the corrector phase, intending to improve the numerical stability. The first part of the normal velocity, $|U_{\text{n}}|$, is computed explicitly with the known velocity at the former inner-iteration step, $\boldsymbol{v}^k$, while the rest, U_{n}, is assumed unknown and integrated implicitly with the unknown temporary velocity, $\boldsymbol{v}^{*k}$, as shown below

$$\boldsymbol{f}_{\text{t}} = -\int_{\Gamma\chi} \Phi_\alpha 0.5(\zeta_{\text{in}} + \zeta_{\text{out}} + \zeta_{\text{tube}})|n_j v^k_{j\alpha}|(n_l v^{*k}_{l\alpha})n_i/\beta^2 \,\mathrm{d}\Gamma \tag{18}$$

With Equation (18) and the equation in step 3 in the corrector phase, the equation in step 1 can be replaced by the following two equations:

$$(\boldsymbol{E} - \gamma\Delta t\underline{\boldsymbol{M}}^{-1}\boldsymbol{F})\boldsymbol{v}^{*k} = \boldsymbol{v}^k + \gamma\Delta t\underline{\boldsymbol{M}}^{-1}(\boldsymbol{f}_{\text{b}} - \boldsymbol{M}\boldsymbol{a}^k - \boldsymbol{K}\boldsymbol{v}^k - \boldsymbol{B}\boldsymbol{v}^k + \boldsymbol{C}\boldsymbol{p}^k) \tag{19}$$

$$\Delta\boldsymbol{a}^{*k} = (\boldsymbol{v}^{*k} - \boldsymbol{v}^k)/\gamma\Delta t \tag{20}$$

where $\boldsymbol{E}$ is the identity matrix, and $\boldsymbol{F}$ is a matrix composed of the following 3×3 sub-matrix $\boldsymbol{\vartheta}_\alpha$ corresponding to a velocity at node α

$$\boldsymbol{\vartheta}_\alpha = -\int_{\Gamma\chi} \Phi_\alpha 0.5(\zeta_{\text{in}} + \zeta_{\text{out}} + \zeta_{\text{tube}})|n_j v^k_{j\alpha}|/\beta^2 \,\mathrm{d}\Gamma \begin{bmatrix} n_x^2 & n_y n_x & n_z n_x \\ n_x n_y & n_y^2 & n_z n_y \\ n_x n_z & n_y n_z & n_z^2 \end{bmatrix} \tag{21}$$

3. COMPUTATIONAL RESULTS

The numerical algorithm explained in the above was coded in FORTRAN90 and all computations were performed on a PC (CPU = 266 MHz). The standard Galerkin formulation was used for all the computations because the SUPG formulation turned out to produce weaker circulation in the cavity flow problem shown in the subsequent section when *Re* number exceeds, say, 1000.

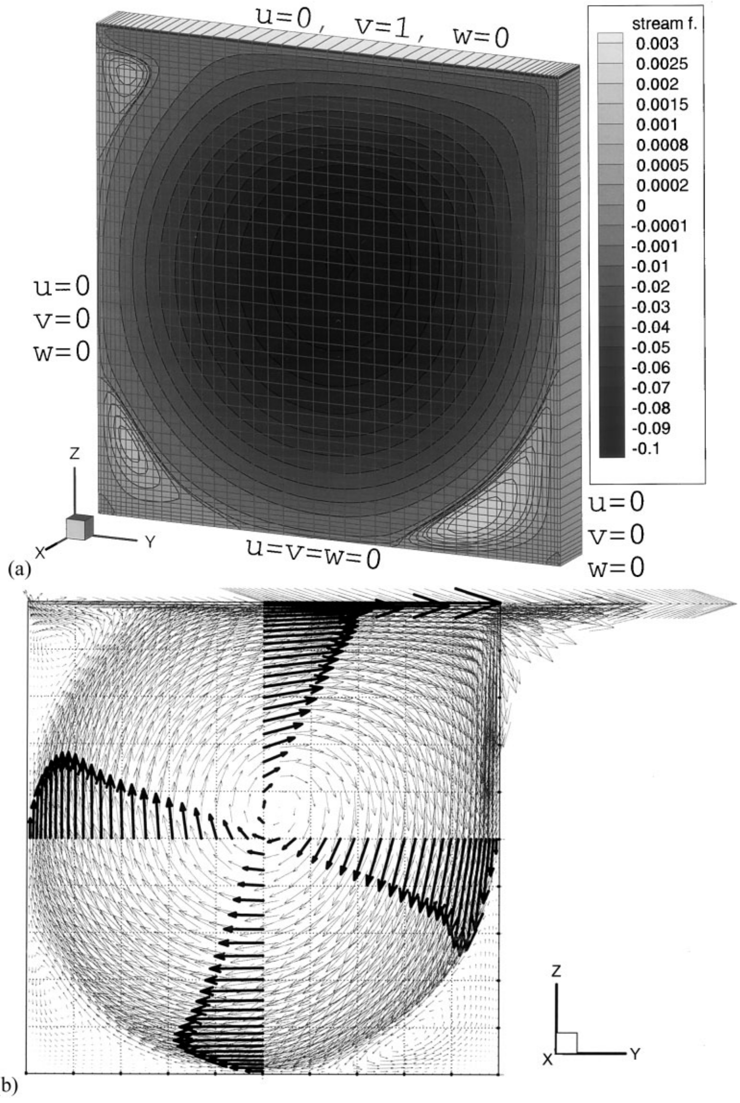

Figure 3. (a) Mesh subdivisions and streamfunction for a driven cavity flow. (b) Velocity vectors at nodes of a driven cavity flow (non-slip wall, $Re = 5000$).

3.1. Code verification

To confirm the validity of the algorithm and the code implementation, a classical problem of a two-dimensional driven cavity flow was computed with the three-dimensional code. Figure 3(a) shows the three-dimensional slab model of 0.1 × 1 × 1 m divided into 1 × 50 × 50 meshes, together with streamfunction obtained at the quasi-steady state. The upper lid is driven at velocity of 1 m s^{-1}, including the corners (i.e. 'leaky' lid). The slip wall conditions are applied to the planes of $x=0$ and $x=0.1$ to realize a pseudo two-dimensional flow, while all the other three walls are assumed as non-slip boundaries. The kinematic viscosity of the fluid is 0.0002 m^2 s^{-1} to set the *Re* number at 5000. Time integration was performed with a time increment of 0.0005 s until 30 s using a result of $Re=1000$ as the initial condition. Figure 3(b) shows the velocity vectors, emphasizing the velocity distributions along vertical and horizontal centerlines. Quantitative comparison with other results [3,7] is made in Table I, which shows that the present result is in good agreement with the others.

3.2. Slip wall computation for cavity flow

To see the effect of the wall condition, the driven cavity flow ($Re=5000$) was computed with the same mesh, using the slip wall and the slip edge conditions for all the boundaries except the upper driven lid. The quasi-steady state was reached with the final time increment of 0.0005 s. Figure 4(a) and (b) show the streamfunction and the velocity vectors respectively. It is seen that there is neither a boundary layer near the walls nor anti-clockwise small vortices in the cavity corners. The maximum and minimum velocities on the centerlines are considerably exaggerated as shown in Table II. The absolute value of the minimum streamfunction is about 1.5 times larger than that of the results with the non-slip case.

Table I. Quantitative comparison of a driven cavity flow ($Re=5000$).

Quantity	Authors		
	Ghia *et al.* [7]	Gresho and Chan [3][a]	Present study
Number of mesh	256 × 256	50 × 50	1 × 50 × 50
Minimum streamfunction	−0.119 (0.512, 0.535)[b]	−0.088 (not available)	−0.101 (0.533, 0.567)
Minimum v on vertical centerline (CL)	−0.436 (0.5, 0.070)	−0.379 (0.5, 0.074)	−0.376 (0.5, 0.073)
Minimum w on horizontal CL	−0.554 (0.953, 0.5)	−0.507 (0.953, 0.5)	−0.494 (0.951, 0.5)
Maximum w on horizontal CL	0.436 (0.078, 0.5)	0.360 (0.074, 0.5)	0.364 (0.087, 0.5)

[a] Only the result obtained with 'Projection 2' method is shown here.

[b] A pair of numbers in parenthesis in Tables I and II denotes the two-dimensional location.

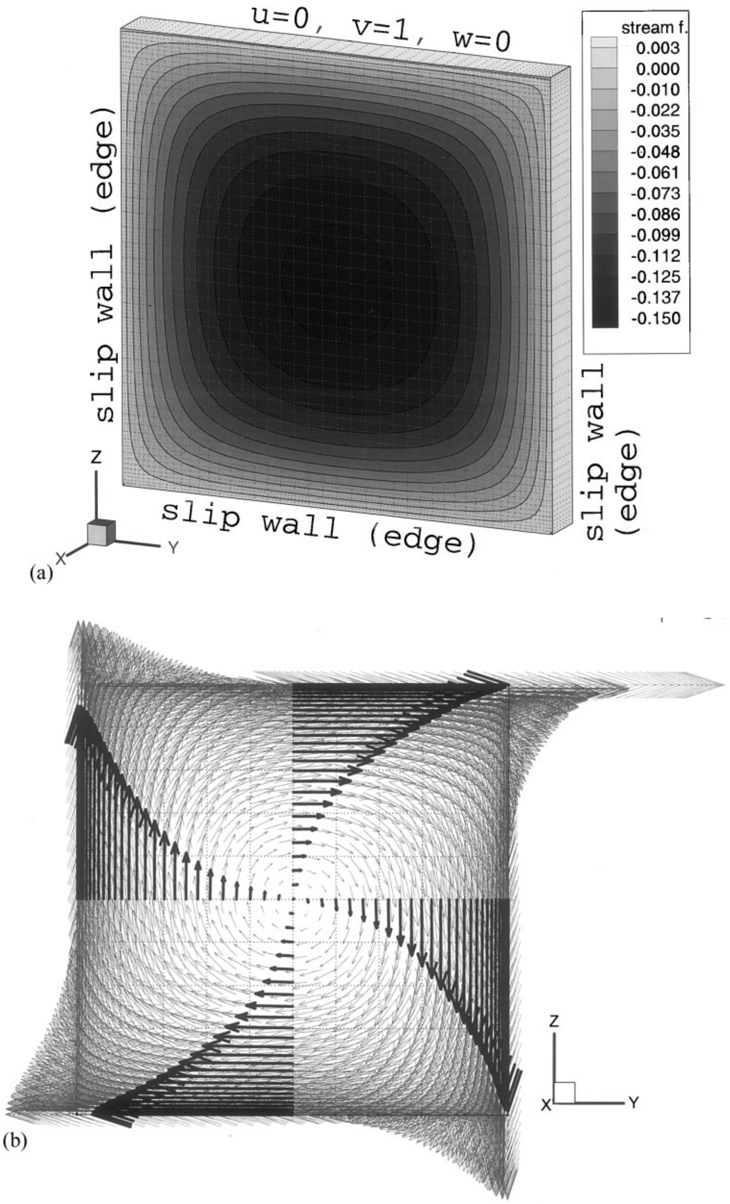

Figure 4. (a) Mesh subdivisions and streamfunction. (b) Velocity vectors of a driven cavity flow (slip wall condition, $Re = 5000$).

Table II. Computational result of driven cavity flow with slip wall and slip edge conditions ($Re = 5000$).

Quantity	Results under slip conditions	Ratio against non-slip result
Minimium streamfunction	-0.155 (0.5, 0.5)	1.53
Minimum v on vertical CL	-0.929 (0.5, 0.0)	2.47
Minimum w on horizontal CL	-0.985 (1.0, 0.5)	1.99
Maximum w on horizontal CL	0.898 (0.0, 0.5)	2.47

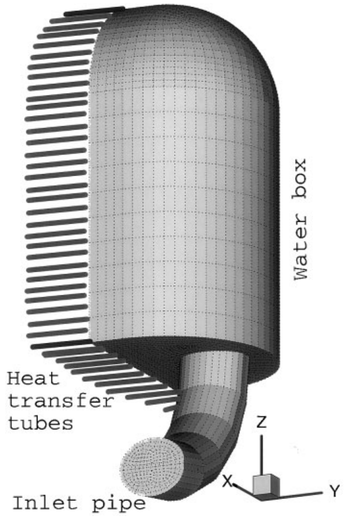

Figure 5. Finite element model for flow in a condenser water box.

3.3. Application to a condenser water box

The code is applied to simulation of a flow in a condenser water box. The water first enters a condenser water box through an inlet pipe and exits it to run inside tens of thousands of heat transfer tubes. The geometry of the condenser water box is idealized in the present study as shown in Figure 5. The flow region is subdivided with 11080 eight-node iso-parametric brick-like elements, sharing 12431 velocity nodes. A constant velocity is specified on the inlet plane, while the slip wall and slip edge conditions are used for all the walls. A tube bundle specified in Table III is assumed downstream of the whole outlet plane. The other computational conditions are tabulated in Table IV.

Figure 6 shows the velocity vectors on the walls and on the outlet plane ($y = -1$) in a quasi-steady state. It is seen that the slip velocity on the walls is plausible, including the 90° edge along the rim of the water box bottom and the 270° edge along the joint of the inlet pipe and the water box bottom. Figure 7 shows the mass flow rate (unit: ton s^{-1} m^{-2})

Table III. Specification of the downstream tube bundle.

Tube length, L (m)	Tube diameter, d (m)	Kinematic viscosity for Re_t, ν ($m^2 s^{-1}$)	Tube hole area ratio, β	Inlet plus outlet loss coefficients, $\zeta_{in}+\zeta_{out}$
17.8	0.035	1×10^{-6}	0.506	1.5

Table IV. Computational conditions for water box flow

Inlet velocity ($m s^{-1}$)	Effective kinematic viscosity, ν_{eff} ($m^2 s^{-1}$)	Time increment, Δt (s)	Time integration parameter, γ	Number of time steps (terminal time)	Iteration limit in corrector phase, K_{max}
1	0.01	0.02	0.6	5000 (100 s)	3

through the outlet plane or the tube plate, assuming the density of fluid is 1 ton m^{-3}. Figure 8 shows the kinematic pressure distribution on the solid walls. The pressure distribution indicates that high pressure appears not only in the jet impingement area, but also in the upper part of the outlet due to the larger pressure loss generated by the flow resistance in the tube bundle.

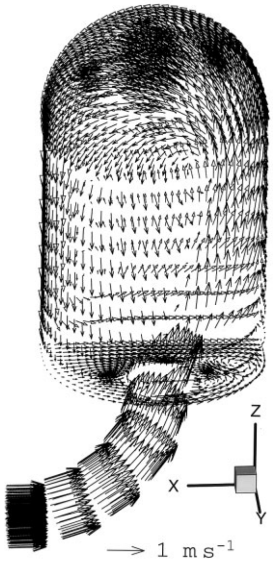

Figure 6. Velocity vectors on walls (black arrows) and on outlet plane (gray arrows with hollow head).

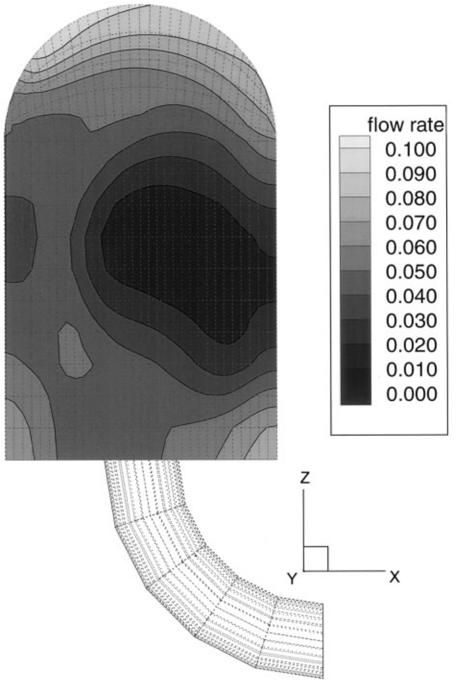

Figure 7. Flow rate distribution on the outlet plane (tube plate).

4. CONCLUSIONS

A technique to impose the slip wall and slip edge conditions has been proposed for a practical three-dimensional flow simulation under a PC environment. A special matrix to fulfill the velocity BCs is derived and is used in the fully explicit time integration process with a new type of pressure Poisson equation. Though the slip wall condition tends to yield exaggerated wall velocities, the proposed technique may be useful to figure out primary flow pattern on a crude mesh for screening several design options or to develop turbulence models for a solid wall.

Another unique technique has also been proposed to represent a tube bundle downstream of a contained flow. A simulation of a flow in the condenser water box has demonstrated that the technique enables to solve the interaction between a plenum flow and a downstream tube bundle, and to predict a flow rate distribution in a tube bundle of a heat exchanger. Experimental verification of the present model is our future work.

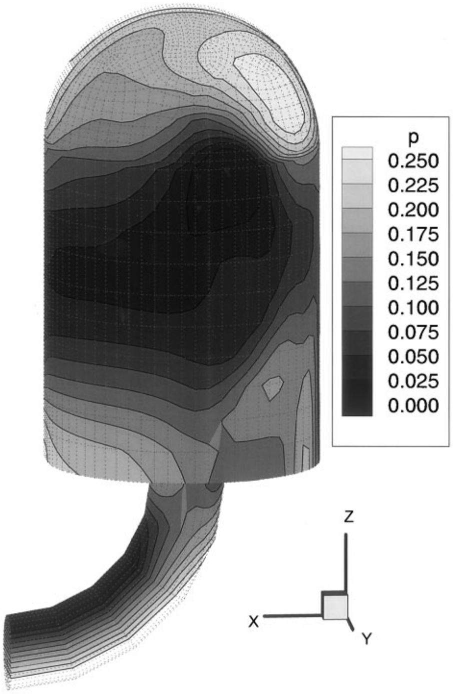

Figure 8. Kinematic pressure on walls of inlet pipe and water box.

ACKNOWLEDGMENTS

The author would like to express his gratitude to Professor G. Yagawa, University of Tokyo, for his encouragement to the present study. Thanks also go to Ms M. Maki, Denryoku Computing Center (DCC) for preparing the finite element geometry data.

REFERENCES

1. Engelman MS, Sani RL, Gresho PM. The implementation of normal and/or tangential boundary conditions in finite element codes for incompressible fluid flow. *International Journal for Numerical Methods in Fluids* 1982; **2**: 225–238.
2. Brooks AN, Hughes TJR. Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1982; **32**: 199–259.
3. Gresho PM, Chan ST. On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. Part 2: implementation. *International Journal for Numerical Methods in Fluids* 1990; **11**: 621–659.
4. Gresho PM. Time integration and conjugate gradient methods for the incompressible Navier–Stokes equations. UCRL-94000, Lawrence Livermore National Laboratory, University of California, 1986.

5. Gresho PM, Lee RL, Sani RL. On the time-dependent solution of the Navier–Stokes equations in two and three dimensions. In *Recent Advances in Numerical Methods in Fluids*, vol. 1, Taylor C, Morgan K (eds). Pineridge: Swansea, 1980; 27–81.
6. Yagawa G, Eguchi Y. Comparison between the traction and pressure-imposed boundary conditions in finite element flow analyses. *Internationl Journal for Numerical Methods in Fluids* 1987; **7**: 521–532.
7. Ghia U, Ghia N, Shin CT. High-Re solutions for incompressible flow using the Navier–Stokes equations and a multi-grid method. *Journal of Computational Physics* 1982; **48**: 387–411.