INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2008; **56**:823–843
Published online 4 July 2007 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1555

Parallel large eddy simulation of turbulent flow around MIRA model using linear equal-order finite element method

Hyounggwon Choi[1], Sungwoo Kang[2] and Jung Yul Yoo[2,*,†,‡]

[1]*Department of Mechanical Engineering, Seoul National University of Technology, 172, Gongreung-2-dong, Nowon-gu, Seoul 139-743, Korea*
[2]*School of Mechanical and Aerospace Engineering, Seoul National University, San 56-1, Shilim-dong, Kwanak-gu, Seoul 151-742, Korea*

SUMMARY

A parallel large eddy simulation code that adopts domain decomposition method has been developed for large-scale computation of turbulent flows around an arbitrarily shaped body. For the temporal integration of the unsteady incompressible Navier–Stokes equation, fractional 4-step splitting algorithm is adopted, and for the modelling of small eddies in turbulent flows, the Smagorinsky model is used. For the parallelization of the code, METIS and Message Passing Interface Libraries are used, respectively, to partition the computational domain and to communicate data between processors. To validate the parallel architecture and to estimate its performance, a three-dimensional laminar driven cavity flow inside a cubical enclosure has been solved. To validate the turbulence calculation, the turbulent channel flows at $Re_\tau = 180$ and 1050 are simulated and compared with previous results. Then, a backward facing step flow is solved and compared with a DNS result for overall code validation. Finally, the turbulent flow around MIRA model at $Re = 2.6 \times 10^6$ is simulated by using approximately 6.7 million nodes. Scalability curve obtained from this simulation shows that scalable results are obtained. The calculated drag coefficient agrees better with the experimental result than those previously obtained by using two-equation turbulence models. Copyright © 2007 John Wiley & Sons, Ltd.

Received 13 November 2006; Revised 22 March 2007; Accepted 22 May 2007

KEY WORDS: Navier–Stokes equation; splitting finite element method; parallel computing; domain decomposition method; large eddy simulation

*Correspondence to: Jung Yul Yoo, School of Mechanical and Aerospace Engineering/IAMD, Seoul National University, San 56-1, Shilim-dong, Kwanak-gu, Seoul 151-742, Korea.

†E-mail: jyyoo@snu.ac.kr

‡Professor.

Contract/grant sponsor: Ministry of Commerce, Industry and Energy, Republic of Korea
Contract/grant sponsor: Ministry of Education and Human Resources Development, Republic of Korea

1. INTRODUCTION

Finite element methods (FEMs) are frequently adopted to discretize the incompressible Navier–Stokes equations for its versatility in treating complex geometries. Due to the rapid growth of computing power and the development of cost-effective algorithms such as matrix-free iterative method [1, 2] and segregated algorithms [3, 4], FEM has been successfully applied to practical fluid flow problems such as those related to road vehicles. Johnson and Tezduyar [5] and Tezduyar *et al.* [6] performed large-scale parallel computations for the analyses of practical problems with complex geometries by utilizing an integrated stabilized finite element formulation of the Navier–Stokes equations and mesh partitioning method, while Rifai *et al.* [7] conducted large-scale parallel simulations using both the Galerkin least-squares (GLS) method and domain decomposition method. They used an equal-order integrated P1/P1 finite element formulation that works well with diagonal preconditioning of which parallel algorithm can be developed with ease.

On the other hand, Choi *et al.* [8] used a splitting 4-step formulation with P2/P1 finite element for the serial computation of the Navier–Stokes equations. The method does not require a special treatment of pressure boundary condition which is typical of the splitting algorithm [3] since both pressure and velocity are obtained simultaneously at the last (divergence-free) stage of the splitting method. However, it has a difficulty in constructing scalable parallel preconditioner of a saddle-point type global matrix given at the last stage. Therefore, in the present study, a 4-step splitting formulation with P1/P1 finite element [3] is to be adopted for a scalable parallel computation of a large-scale problem. Elliptic type pressure equation, which is easy to parallelize, is to be solved instead of a saddle-point type global matrix. Moreover, the present formulation solves the equations corresponding to three velocity and pressure variables separately. Hence, it is intrinsically suitable for a scalable computation and reduces memory requirement compared to a velocity–pressure integrated formulation.

For simulating the turbulent flow around a vehicle model, many efforts have been made. Carr [9], Gaylard *et al.* [10], Hur *et al.* [11] and Myoung and Jin [12] performed numerical simulations of the flow around MIRA (Motor Industry Research Association, Nuneaton, Warwickshire, United Kingdom, http://www.mira.co.uk) model using two-equation turbulence models. Carr compared the numerical solutions obtained from various commercial codes with the experimental results. Gaylard *et al.* compared their numerical results for three different MIRA models, obtained by finite volume method (FVM) based on SIMPLE algorithm, with experimental results. However, both the numerical results were obtained without wheels. The previous finite volume results based on conventional two-equation turbulence models reveal that the drag coefficient of the flow around MIRA model is very hard to predict. Hur *et al.* [11] and Myoung and Jin [12] simulated the flow with wheels using FVM based on SIMPLE algorithm in conjunction with k–ε and RNG k–ε turbulence models, where the predicted drag coefficients were about 60 and 30% higher than the experimental result, respectively. Hajiloo *et al.* [13], Johnson and Tezduyar [5] and Rifai *et al.* [7] calculated an external turbulent flow around a car model with FEM. Hajiloo *et al.* [13] used a two-equation turbulence model and streamline upwind Petrov–Galerkin (SUPG) method with segregated formulation. Although they obtained quite accurate results for pressure profiles on the surface of the car, the typical vortex behind the model was not predicted as it should be. Johnson and Tezduyar [5] used the Smagorinsky model as a zero-equation turbulence model and predicted drag coefficients that agree well with the experimental result. Rifai *et al.* [7] used Spalart–Allmaras turbulence model and obtained pressure profile at the symmetry plane that agrees well with an existing experimental result. However, it has to be noted that almost all results from FEM or FVM

DOI: 10.1002/fld

are time-averaged results from steady flow simulation despite the fact that the actual flow field is unsteady.

On the other hand, there is still few reported works on large eddy simulation (LES) of the turbulent flow around a car model. Choi *et al*. [8] conducted a LES of the flow around the MIRA model and obtained quite an accurate drag coefficient with approximately 0.7 million nodes. However, as their calculations were carried out with a single processor, they spent about 1 month of CPU time for the 20-s simulation of the flow using a Compaq GS320 Machine, which is untenable for practical use.

The objective of the present study is to parallelize a fractional 4-step splitting algorithm with P1/P1 finite element [3], which is cost effective and easy to parallelize due to the splitting of velocity and pressure variables, for carrying out LES calculations of turbulent flows around an arbitrarily shaped body. Domain decomposition method with Message Passing Interface (MPI) Library [14] is used and parallel block ILU preconditioner [15, 16] is adopted to accelerate the convergence of conjugate-gradient-type iterative solvers. Driven cavity flow in a cubical enclosure is solved to verify the developed parallel code and to estimate the parallel performance of the code. Both constant-coefficient and dynamic-Smagorinsky models are adopted as turbulence models. To validate the turbulence calculation we solve the turbulent channel flows at two different Reynolds numbers, and for overall code validation we solve a backward facing step flow. Lastly, the flow around the MIRA model is calculated using the present parallel code with 32 IBM SP processors, by comparing with both the experiment [9] and the serial result based on a splitting P2/P1 finite element formulation [8].

2. NUMERICAL METHODS

The governing equations are the continuity equation and the incompressible Navier–Stokes equation in conjunction with Smagorinsky LES models:

$$\nabla \cdot U = 0 \tag{1}$$

$$\frac{\partial U}{\partial t} + (U \cdot \nabla) U = \nabla \cdot \widetilde{\sigma} \tag{2}$$

where $\widetilde{\sigma} = -(p/\rho)\widetilde{I} + \widetilde{\tau}$, $\widetilde{\tau} = 2\nu_{\mathrm{e}}\widetilde{S}$, $\nu_{\mathrm{e}} = \nu + \nu_t$, $\nu_t = C\Delta^2\overline{S}$, $\overline{S} = (2\widetilde{S} : \widetilde{S})^{1/2}$, $\widetilde{S} = \frac{1}{2}(\nabla U + \nabla U^{\mathrm{T}})$, $\Delta = (\text{volume})^{1/3}$ and $\nu, \nu_{\mathrm{e}}, \nu_t, C, \widetilde{S}$ and Δ denote the kinematic viscosity, effective viscosity, eddy viscosity, Smagorinsky model constant, strain-rate tensor and the characteristic length of a finite element, respectively. The boundary and initial conditions for Equation (2) is given as follows:

$$\begin{aligned} U|_{\Gamma_1} &= \mathbf{b}, \quad \widetilde{\sigma} \cdot \mathbf{n}|_{\Gamma_2} = \mathbf{t} \\ U(\mathbf{x}, 0) &= U_0(\mathbf{x}) \end{aligned} \tag{3}$$

where Γ_1 and Γ_2 are two non-overlapping subsets of the piecewise smooth domain boundary, $\mathbf{b}$ is the velocity given on Γ_1, $\mathbf{t}$ is the traction vector given on Γ_2, and $\mathbf{n}$ is the outward unit normal vector to Γ_2.

Fractional 4-step splitting method is adopted to solve the above equations, which was devised by Choi and Moin [17] and implemented in the context of equal-order finite element (P1/P1) method by Choi *et al*. [3]. In order to avoid the blow-up of the numerical solution at very high Reynolds numbers, SUPG method [18] has been adopted only for the flow around MIRA model.

DOI: 10.1002/fld

2.1. P1/P1 fractional 4-step method and finite element formulation

1. Given the pressure field of previous time step p^m, we need to obtain an intermediate velocity field from the non-linear incompressible momentum equation, which is temporally discretized by Crank–Nicolson method, at the first fractional step, as follows:

Find $\hat{U} \in H^1$ such that

$$\int_\Omega \left[W \cdot \left(\frac{\hat{U}}{\Delta t} + \frac{1}{2}(\hat{U} \cdot \nabla)\hat{U} \right) + \nabla W : \frac{\hat{\tau}}{2} \right] \mathrm{d}\Omega$$

$$= \int_\Omega \left[W \cdot \left(\frac{U^m}{\Delta t} - \frac{1}{2}(U^m \cdot \nabla)U^m \right) - \nabla W : \left(\frac{\hat{\tau}}{2} - \frac{p^m}{\rho}\widetilde{I} \right) \right] \mathrm{d}\Omega + \int_{\Gamma_2} W \cdot \mathbf{t}\, \mathrm{d}\Omega \quad \forall W \in H^1 \tag{4}$$

$$W|_{\Gamma_1} = 0, \quad U|_{\Gamma_1} = \mathbf{b}, \quad U(\mathbf{x}, 0) = U_0(\mathbf{x})$$

In the above equation, H^1 denotes a space where the first derivative of the basis function is square integrable. The Dirichlet boundary condition for intermediate velocity is $\hat{U} = U^{m+1}$, as was discussed in Reference [3].

2. At the second fractional step, the following equation for another intermediate velocity U^*, whose finite element formulation is trivial, is solved with the given pressure field of the previous time step:

$$\frac{U^* - \hat{U}}{\Delta t} = \frac{1}{\rho} \nabla p^m \tag{5}$$

This fractional step is necessary for the second-order accurate boundary condition of the intermediate velocity, which was discussed in [3, 16].

3 and 4. At the next fractional steps, the following Poisson-type pressure equation derived from divergence-free constraint is solved and then the updated pressure field corrects the velocity field:

$$\nabla^2 p^{m+1} = \frac{\rho}{\Delta t} \nabla \cdot U^* \tag{6}$$

$$\frac{\hat{U}^{m+1} - U^*}{\Delta t} = -\frac{1}{\rho} \nabla p^{m+1} \tag{7}$$

The finite element formulation of Equation (6) can be found in Reference [3].

2.2. Finite element, numerical integration and element matrices

Since only tetrahedral element is used in this study, all variables are interpolated using a linear tetrahedral shape function within an element:

$$\begin{aligned} u^{(e)} &= \sum_{i=1}^{4} u_i N_i \\ N_i &= (a_i + b_i x + c_i y + d_i z)/6V \end{aligned} \tag{8}$$

DOI: 10.1002/fld

Formulas for V, a_i, b_i, c_i, d_i can be found in Reference [19] and derivatives of the shape function can be easily evaluated. The following well-known formula is used for integration:

$$\int_{\Omega} N_i^{\alpha} N_j^{\beta} N_k^{\gamma} N_l^{\delta} \, \mathrm{d}\Omega = \frac{\alpha!\beta!\gamma!\delta!6V}{(\alpha+\beta+\gamma+\delta+3)!} \tag{9}$$

On the other hand, the element matrix generated from the discretization of Equation (4) is given as follows:

$$\begin{aligned} (\mathbf{M}+\mathbf{C}+\mathbf{K})\hat{U} &= \mathbf{F} \\ \mathbf{F} &= \int_{\Omega} \left[W \cdot \left(\frac{U^m}{\Delta t} - \frac{1}{2}(U^m \cdot \nabla)U^m \right) \right. \\ &\quad \left. - \nabla W : \left(\frac{\tau^m}{2} - \frac{p^m}{\rho}\tilde{I} \right) \right] \mathrm{d}\Omega + \int_{\Gamma_2} W \cdot \mathbf{t} \, \mathrm{d}\Omega \end{aligned} \tag{10}$$

where **M**, **C**, **K** are mass matrix, convective matrix and diffusion matrix, respectively. The non-linearity of convection term is linearized using the successive substitution method. For a detailed expression of these matrices, see Reference [20].

2.3. Pressure boundary conditions

As the fractional 4-step method solves the pressure equation, the boundary conditions for the pressure are required. By assuming traction-free condition at the exit and considering Equation (3), the pressure boundary condition at the exit is derived as follows:

$$p^{m+1}\mathbf{n} = \rho\tau^{m+1} \cdot \mathbf{n} = \rho\hat{\tau} \cdot \mathbf{n} \tag{11}$$

where τ^{m+1} can be replaced by $\hat{\tau}$ since $\hat{U} = U^{m+1}$ [3].

Utilizing the no-slip condition, the pressure boundary condition at the wall is given as follows:

$$\frac{\partial p^{m+1}}{\partial n} = 0 \tag{12}$$

2.4. SUPG method

We need to use at most 6.7 million nodes distributed among 32 processors for the simulation of the flow around the MIRA model at $Re = 2.6 \times 10^6$. However, the number of nodes is not sufficient enough to obtain a stable solution with Galerkin method at such a high Reynolds number. At present, we are not certain how many nodes are necessary to solve this problem with Galerkin LES. Therefore, the SUPG method is adopted to stabilize the formulation in the simulation of very high Reynolds number flows, so that the Galerkin weight in Equation (4) is modified as follows:

$$\tilde{W} = W + sU \cdot \nabla W, \quad s = \frac{\alpha h_{\mathrm{e}}}{2|u|_{\mathrm{e}}} \tag{13}$$

where α is a function of the cell Peclet number and h_{e} is the element characteristic length. It is noted that the results obtained will still be close to those obtained from Galerkin LES since the numerical diffusion introduced by applying SUPG to the discretization of the incompressible Navier–Stokes equations decreases as element size decreases.

DOI: 10.1002/fld

2.5. LES models

LES is adopted to describe the turbulent flow. Both the constant-coefficient and the dynamic-Smagorinsky models are implemented in the present code. The model constant of LES model may depend on the Reynolds number, the characteristics of the flow field, etc. While a fixed coefficient is adopted in the conventional-Smagorinsky model regardless of the flow field, the coefficient of dynamic LES model is a function of filtered stress tensors of the flow field. In the constant-coefficient model, eddy viscosity v_t is given as in Equation (2), where $C = 0.01$ regardless of the flow field. To account for the wall effect in the constant-coefficient model, the following damping function is used to modify the characteristic length of a finite element adjacent to the wall:

$$\overline{\Delta} = \{1 - \exp(-y^+/A^+)\}\Delta \tag{14}$$

where $\overline{\Delta}$ is the modified length scale of the element and empirical constant $A^+ = 25$ is used in this study.

In the dynamic model, generalized top hat filter [21] is adopted for the unstructured mesh around the MIRA model. The filtering process is carried out through the convolution of any function f with filtering kernel G as follows:

$$\widetilde{f}(x) = \frac{\int_\Omega G(x, x') f(x')\,\mathrm{d}x'}{\int_\Omega G(x, x')\,\mathrm{d}x'} \tag{15}$$

In this model, the model constant C is not fixed but is a function of resolved stress tensors L_{ij} and M_{ij}:

$$L_{ij} = \widetilde{\overline{u}_i\overline{u}_j} - \widetilde{\overline{u}}_i\widetilde{\overline{u}}_j, \quad M_{ij} = (\widetilde{\Delta}/\overline{\Delta})^2 \widetilde{\overline{S}}\widetilde{\overline{S}}_{ij} - \widetilde{\overline{S}\overline{S}}_{ij}, \quad C\overline{\Delta}^2 = \frac{1}{2}\frac{L_{ij}M_{ij}}{M_{pq}M_{pq}} \tag{16}$$

where the overbar denotes a variable filtered at an element length scale $\overline{\Delta}$ while the tilde denotes a variable filtered at a larger scale, which is twice the element length scale in the case of a uniform grid. A kind of averaging process in the dynamic model is not considered for the simulation of the MIRA model since a fully unstructured mesh is employed for the simulation. For the case of the channel flow where averaging process is involved, it has been found that the model constant is positive over nearly the whole computational domain. We have clipped out the constant in some regions, wherever it becomes negative.

2.6. Parallelization

Domain decomposition method is used to parallelize the fractional 4-step method. After the mesh generation, manual partitioning is used for structured mesh and METIS Library [22] is used for unstructured mesh to divide the computational domain into many subdomains. Message passing between processors is performed by MPI Library. Conjugate gradient (CG) method and Bi-CGSTAB [23] are, respectively, used to solve symmetric and unsymmetric matrices that result from fractional 4-step method. All matrices are preconditioned with block ILU parallel preconditioner [15, 16]. A computational domain is decomposed by vertex-oriented domain decomposition method on the basis of the fact that each subdomain has the same number of vertices to satisfy the load balance for all processors. Each decomposed subdomain consists of internal nodes, interior and exterior boundary nodes, as illustrated in Figure 1 [24]. After domain decomposition, the nodes are locally renumbered in the order of internal nodes, interior boundary nodes and exterior boundary nodes.

DOI: 10.1002/fld

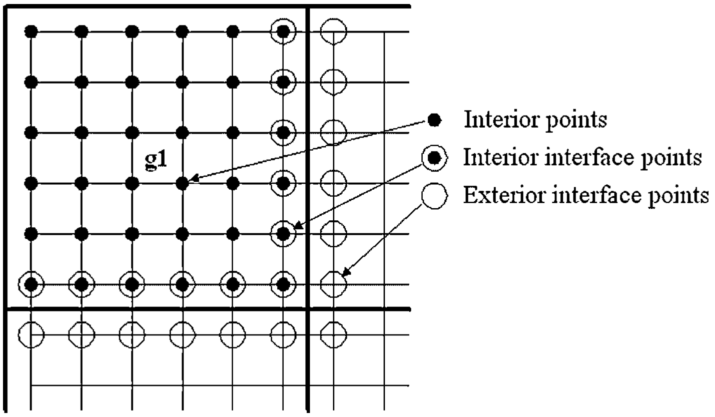

Figure 1. Classification of nodes in a subdomain.

Each processor calculates the values of flow variables at internal nodes independently. However, for those at interior and exterior boundary nodes, communications with adjacent processors are needed before the calculation. In each processor, only local values belonging to each subdomain are computed and stored. Therefore, rows for exterior boundary nodes are neglected in constructing the local matrix for each processor. The local matrix $[A_i]$ for solving values at internal and interior boundary nodes in decomposed ith subdomain can be typically constructed as follows:

$$[A_i]\{x_i\}=\{f_i\}$$

$$[A_i]=\begin{pmatrix} B_i & E_i & 0 \\ F_i & C_i & I_i^j \end{pmatrix}, \quad \{x_i\}=\begin{pmatrix} u_i \\ v_i \\ w_i^j \end{pmatrix}, \quad \{f_i\}=\begin{pmatrix} s_i \\ t_i \end{pmatrix} \tag{17}$$

It is noted that $\{u_i\}$ and $\{s_i\}$ are components from internal node values, $\{v_i\}$ and $\{t_i\}$ from interior boundary node values and $\{w_i^j\}$ from exterior boundary node values, where the superscript j indicates that ith subdomain is linked with jth subdomain. At each iteration step of CG solver, ith subdomain sends $\{v_i\}$ to jth subdomain and receives $\{w_i^j\}$ from jth subdomain to complete matvec (matrix vector multiplication) operation. As for parallel preconditioner, block ILU decomposition is applied to all subdomains simultaneously for the following square block matrix:

$$[\widetilde{A}_i]=\begin{pmatrix} B_i & E_i \\ F_i & C_i \end{pmatrix} \tag{18}$$

3. NUMERICAL RESULTS

3.1. Driven cavity flow in a cubical enclosure

In order to verify the parallel code and to check the parallel performance, we consider the driven cavity flow in a cubical enclosure, as shown in Figure 2, with the following boundary

DOI: 10.1002/fld

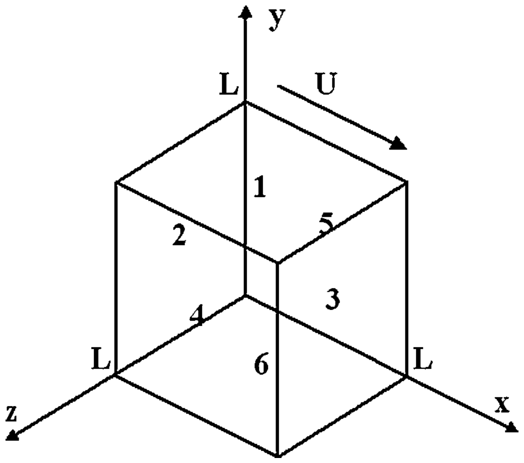

Figure 2. A schematic of computational domain for solving the driven cavity flow in a cubical enclosure.

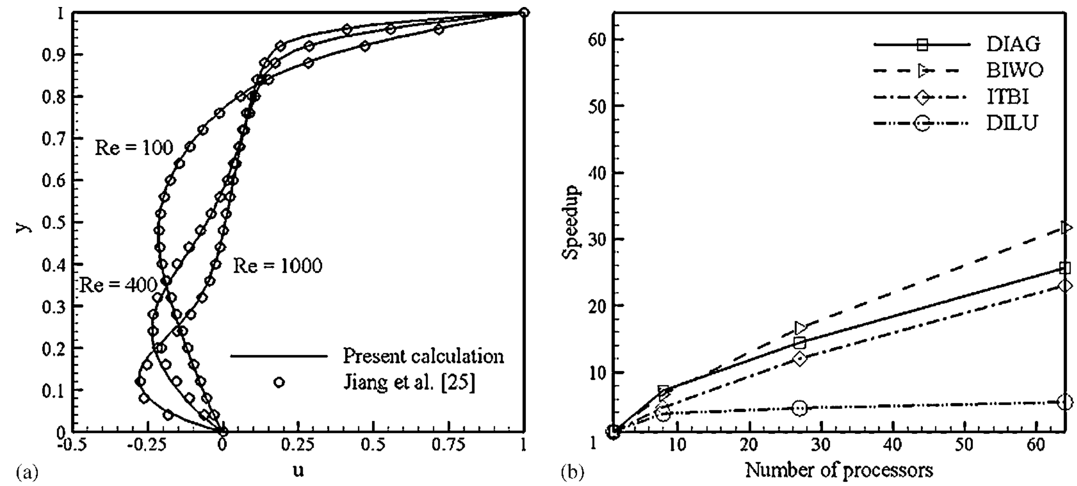

Figure 3. (a) Velocity profiles at $(x, z) = (0.5, 0.5)$ for three-dimensional driven cavity flow in a cubical enclosure at $Re = 100, 400$ and 1000, and (b) speedup of parallel computation for solving the driven cavity flow.

conditions:

$$\begin{aligned} &\text{face 1:} && u_i = U \\ &\text{faces 2–6:} && u_i = 0 \end{aligned} \tag{19}$$

The problem has been solved for $Re = 100$, 400 and 1000 using a $32 \times 32 \times 32$ non-uniform structured mesh. In Figure 3(a), the streamwise velocity distributions are compared at $(x, z) = (0.5, 0.5)$ for the three Reynold numbers. The results are in excellent agreement with previous results obtained by Jiang *et al.* [25] who used a $50 \times 52 \times 25$ mesh with velocity–pressure–vorticity formulation and least-squares FEM.

As a measure of parallelization efficiency, we use speedup which is defined as follows:

$$\text{speedup} = \frac{T_1}{T_N} \tag{20}$$

where T_1 is the calculation time using one processor and T_N is the calculation time using N processors. Figure 3(b) compares the speedups obtained with up to 64 processors when four different preconditioners are used. Namely, they are diagonal, block ILU, iterative block ILU and distributed ILU preconditioners [15, 16, 26]. As is well known, the speedup tends to be saturated as the number of processors increases. Note that the speedups gained from parallel preconditioners in Figure 3 are relatively small because the problem size is not big enough. It is to be shown in Section 3.4 that the present parallel code proves to be super-linear scalable for a very large-scale problem. Since the block ILU gives the best performance among the preconditioners tested in the present problem, we are to use it for the following problems.

3.2. Turbulent channel flow

In order to confirm the validity of the present algorithm in connection with the LES models, we solve turbulent channel flows at $Re_\tau = 180$ and 1050, based on channel half-height, fluid viscosity and friction velocity. We use a $51 \times 71 \times 71$ mesh of $3\pi \times 2 \times \pi$ periodic domain for the simulation at $Re_\tau = 180$, which is quite coarse compared to that used by Kim *et al.* [27], and a $64 \times 97 \times 128$ mesh of $2\pi \times 2 \times 2\pi$ periodic domain for the simulation at $Re_\tau = 1050$. Figures 4 and 5 show the mean velocity profiles and velocity fluctuations normalized by friction velocities at $Re_\tau = 180$ and 1050, respectively, when the constant-coefficient model is used. Mean velocity profiles agree satisfactorily with the results of the existing DNS [27] and dynamic LES [28], respectively, while turbulence intensity profiles differ from the previous numerical results, particularly at the higher Reynolds number. For the constant-coefficient model, damping function in Equation (14) is used to account for the wall effect. From Figures 4 and 5, it seems that the present FEM based on P1/P1 formulation works quite satisfactorily for obtaining the mean velocity profiles of the channel flow. However, a more sophisticated LES model [28] needs to be incorporated for more precise estimation of the turbulence intensity profiles. Figure 6 shows an instantaneous velocity field at $y^+ \sim 2$, where the streaky pattern is clearly revealed at $Re_\tau = 180$.

3.3. Turbulent flow over a backward-facing step

For overall code validation, turbulent flow over a backward-facing step is solved at $Re = 5100$ based on the step height h and inlet free-stream velocity, and an expansion ratio of 1.20, which coincides to the same flow domain as adopted in the DNS study of Le *et al.* [29]. In the present study, a $250 \times 120 \times 32$ grid system as shown in Figure 7 is used for LES, while Le *et al.* conducted DNS with a $768 \times 192 \times 64$ grid system. In the vertical direction, 40 out of 120 nodes are placed within the step $(y<h)$, in comparison with 70 out of 192 cells for the case of Le *et al.* As shown in Figure 8, the pressure coefficient obtained from the present LES agrees reasonably well with both the experiment of Jovic and Driver [30] and DNS of Le *et al.* from the step corner to the reattachment point, where the grid resolution in the streamwise direction is finer than the other regions. However, it deviates a little from the experimental value downstream of the reattachment point. This seems to be due to the fact that the streamwise grid resolution of the present LES is not fine enough. According to the grid resolution test of Le *et al.* [29] for the DNS of a backward-facing step flow, the streamwise grid resolution is very important in order to correctly simulate

DOI: 10.1002/fld

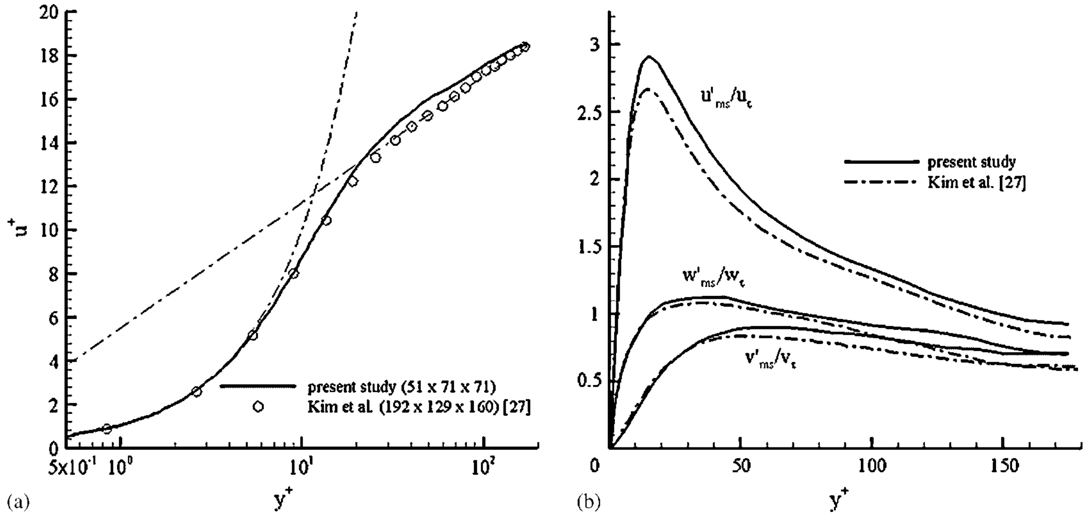

Figure 4. (a) Mean velocity profile and (b) velocity fluctuations normalized by wall velocity in a turbulent channel flow at $Re_\tau = 180$.

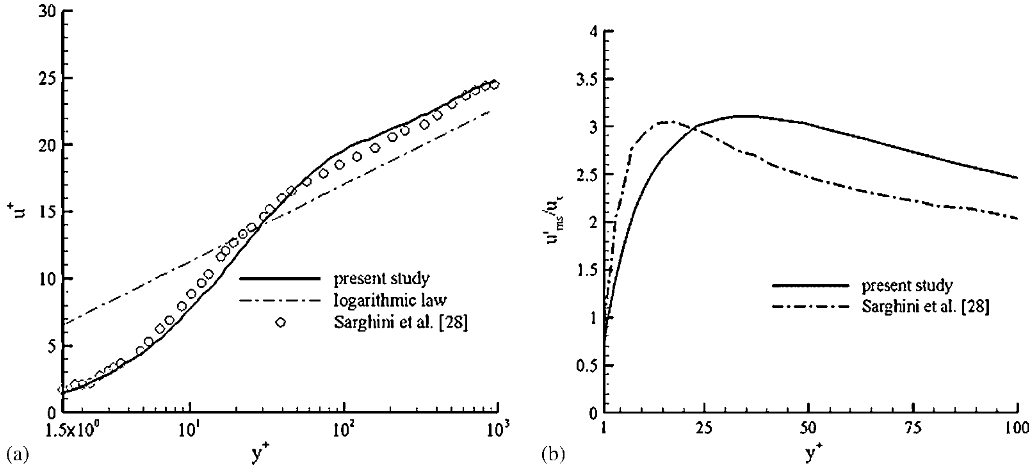

Figure 5. (a) Mean velocity profile and (b) velocity fluctuations normalized by wall velocity in a turbulent channel flow at $Re_\tau = 1050$.

the unsteady flow field. Figure 9 compares the wall skin friction coefficient distribution with the experimental [30] and DNS results [29]. As in Figure 8, the friction coefficient deviates from the experiment downstream of the reattachment point while it agrees well with both experiment and DNS in the recirculation region. It should be noted that in Figure 9 positive value of the friction coefficient appears in the reverse flow region for both the present simulation and DNS of Le *et al.* A similar result was reported in the LES study of Friedrich and Arnal [31].

DOI: 10.1002/fld

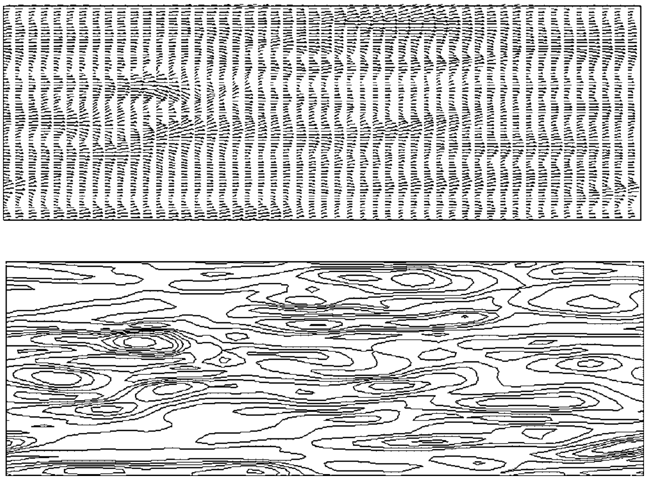

Figure 6. Instantaneous velocity field at $y^+ \sim 2$ in a turbulent channel flow at $Re_\tau = 180$.

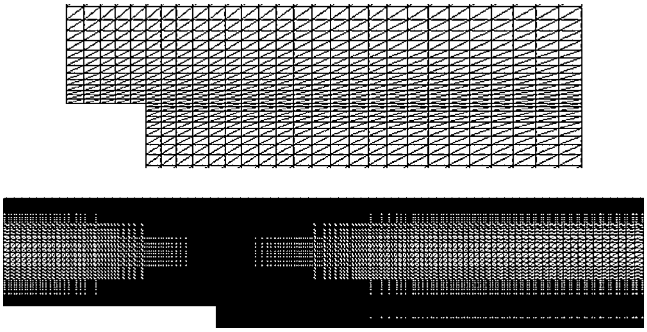

Figure 7. 250 × 120 × 32 mesh used for calculating the backward-facing step flow (x–y plane).

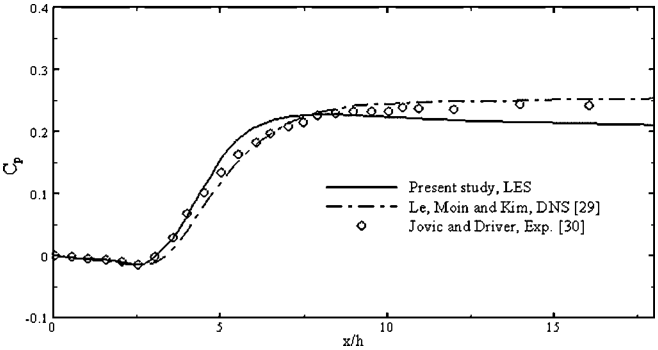

Figure 8. Comparison of pressure coefficient with experimental and DNS results.

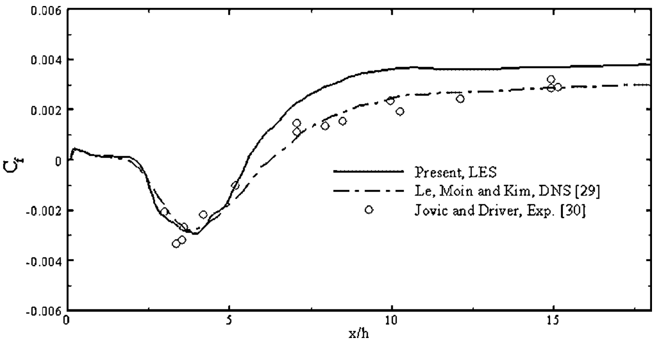

Figure 9. Comparison of step-wall skin friction coefficient with experimental and DNS results.

3.4. *Turbulent flow around the MIRA model*

Turbulent flow around the MIRA model at $Re = 2.6 \times 10^6$, which is based on inlet velocity, model height and fluid viscosity, is calculated with two Smagorinsky LES models. The unsteady solution which uses the constant-coefficient model is obtained first and then the solution which uses the dynamic model is sought by taking the former as the initial guess. Time step size is 0.003 s. To generate an unstructured mesh for the simulation, ICEM-CFD, a commercial mesh generation program, is used. Tetrahedral elements are used to fill the three-dimensional computational domain and then several layers of prism elements are stacked on the surface of the MIRA model in order to resolve the boundary layer, as was done by Johnson and Tezduyar [5]. Each prism element is later subdivided into three tetrahedrons to make the present code simpler. We have generated three meshes. Coarse one (Mesh I) consists of approximately 0.6 million nodes, intermediate one (Mesh II) about 2.7 million nodes, and the fine one (Mesh III) about 6.7 million nodes. Figure 10 shows some cross-sectional slices of unstructured tetrahedral element mesh around the MIRA model on some planes and indicates the boundary conditions corresponding to wind tunnel condition. Relatively denser nodes are used near the model surface and the wake region.

Figure 11 shows partitioned unstructured mesh around the model when 32 processors are used for parallel computation, based on domain decomposition method. METIS Library is used to partition the unstructured mesh. Figures 12 and 13 show the velocity fields on some planes at a given instant when the constant-coefficient LES model is employed, which are very complex. There is a recirculation zone in front of the windshield due to the local adverse pressure gradient and vortices are shed at both edges of the roof also due to strong adverse pressure gradients. We can also find that there is a reverse flow behind the wheels and strong reverse flow alongside the rear window. In Figure 14, some snapshots of the unsteady motion of the flow around the model are shown. The velocity and pressure fields change with time due to vortices that are shed from the front edge of the roof.

In Figure 15, we compared our time-averaged pressure distribution along the upper surface of the symmetric plane (z–x plane) of the MIRA model with the experimental result and the numerical result obtained by using the k–ε turbulence model. The results are obtained with Mesh III using two Smagorinsky models. The distance shown in the figure is dimensional and is measured from the reference point, where C_p is about 0.6. Although there are some discrepancies in the magnitudes of pressure peaks, the calculated results predict the peak positions exactly and agree quite well

DOI: 10.1002/fld

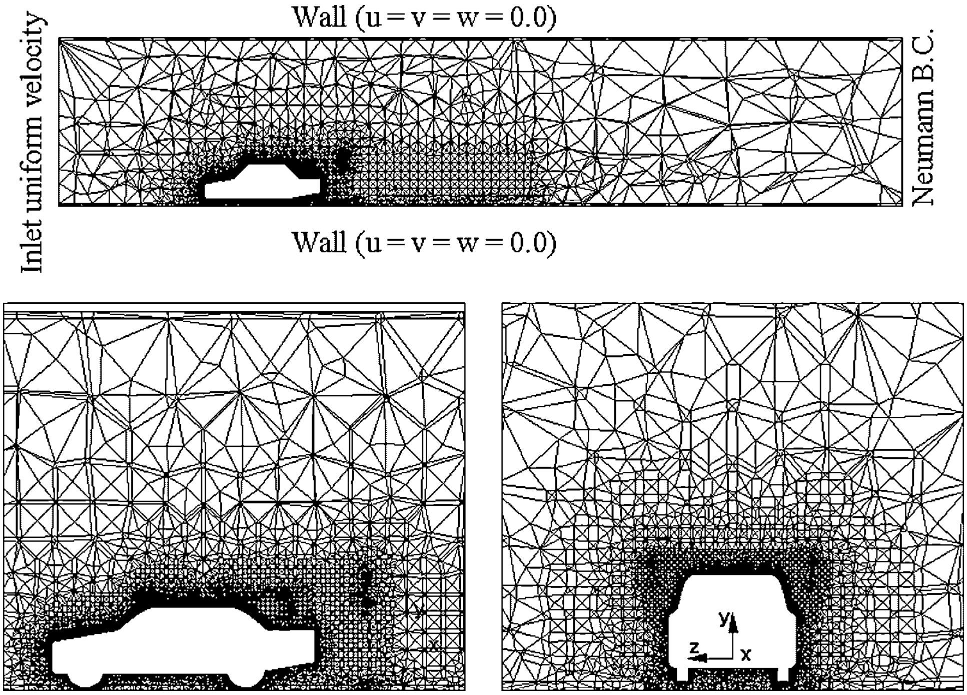

Figure 10. Some slices of unstructured mesh around the MIRA model.

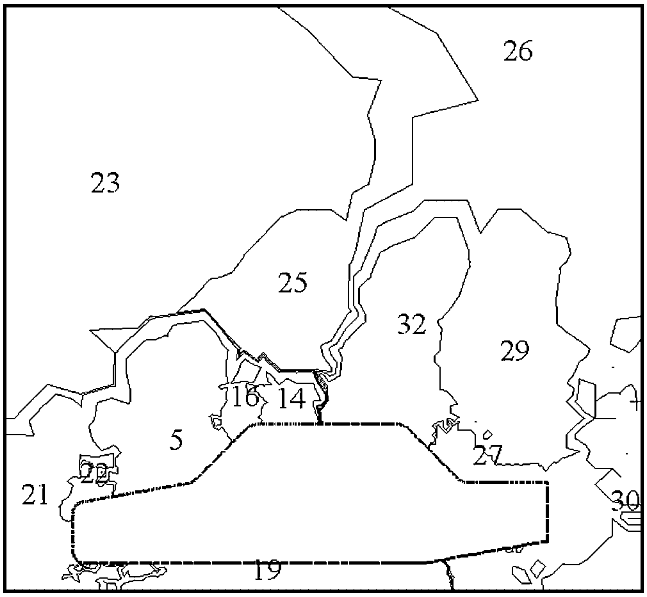

Figure 11. A slice of partitioned mesh in the centre plane around the MIRA model.

DOI: 10.1002/fld

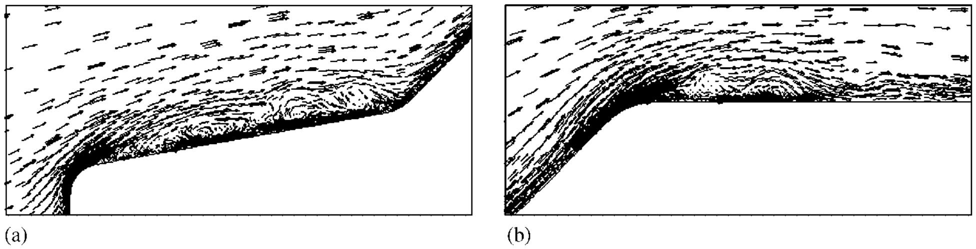

Figure 12. Instantaneous velocity fields over the front part of the MIRA model, obtained by the constant-coefficient LES model at $t \sim 17\,\mathrm{s}$ (Mesh III): (a) bonnet and (b) roof.

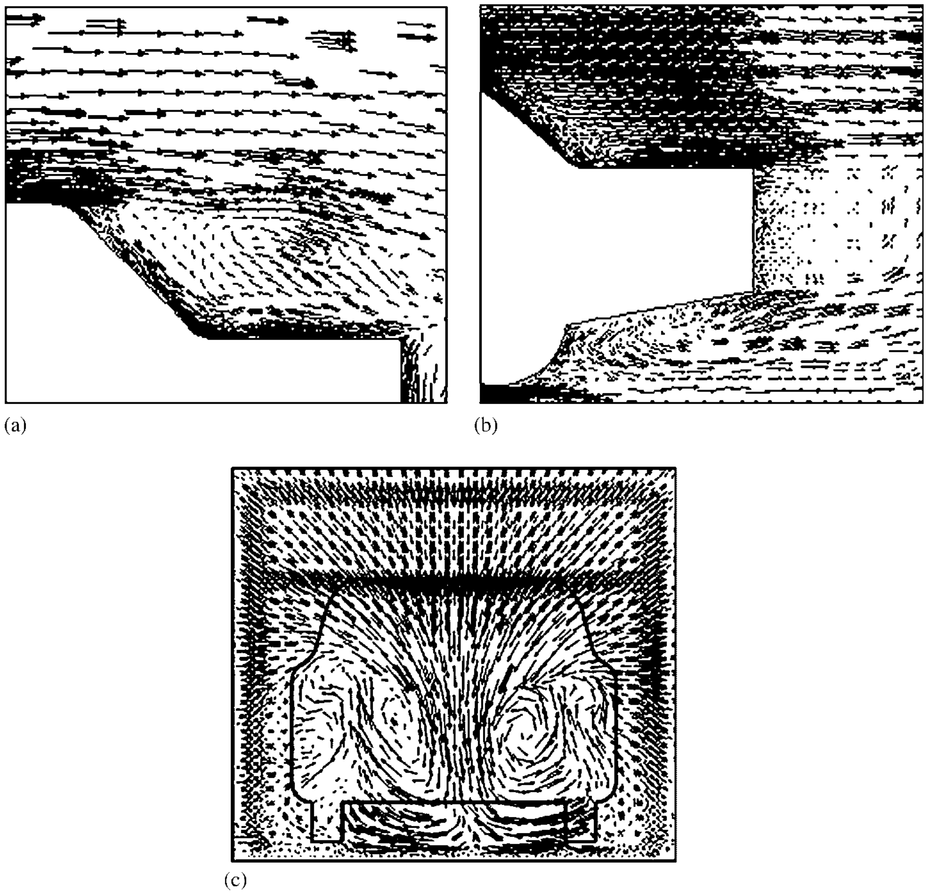

Figure 13. Instantaneous velocity fields over and past the rear part of the MIRA model, obtained by the constant-coefficient LES model at $t \sim 17\,\mathrm{s}$ (Mesh III): (a) backlight; (b) rear wheel; and (c) y–z plane at $0.2L$ behind the MIRA model.

DOI: 10.1002/fld

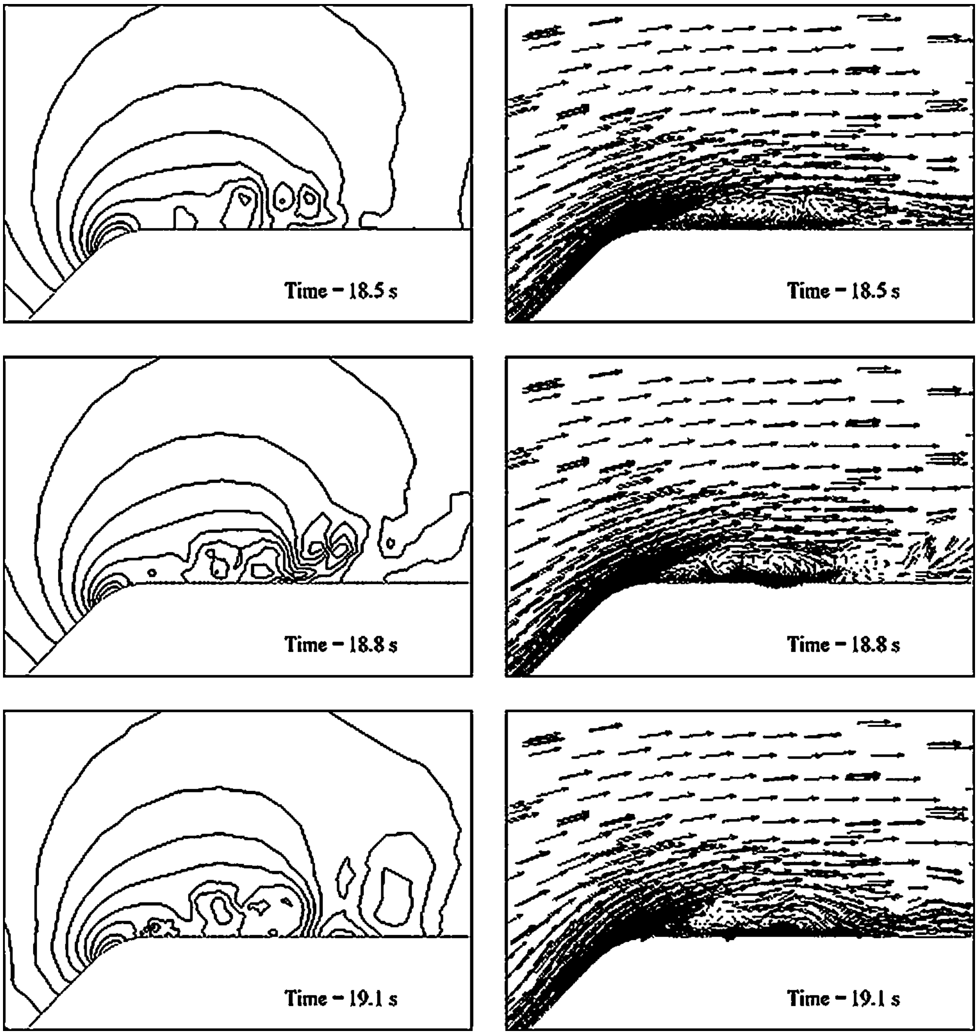

Figure 14. Pressure and velocity fields of unsteady flow around the MIRA model (Mesh III).

with the experiment overall. The results predicted by two Smagorinsky models are similar except that the predicted magnitude of the second pressure peak is smaller for the dynamic model. In Figure 16, the pressure distributions are compared, which are obtained by using the constant-coefficient model for two different formulations. Compared to the result obtained by Choi *et al.* [8] through a serial calculation using P2/P1 formulation with about 0.7 million nodes, the present C_p curve calculated by P1/P1 formulation with about 6.7 million nodes (Mesh III) deviates a little more from the experiment at the first pressure peak, but at the second peak it gives closer result to the experiment. This is not odd at all considering that P2/P1 formulation is of third-order accuracy in terms of the spatial discretization of the velocity field while P1/P1 is second-order accurate and that the present P1/P1 formulation has only 2.2 times as many nodes as P2/P1 formulation in each

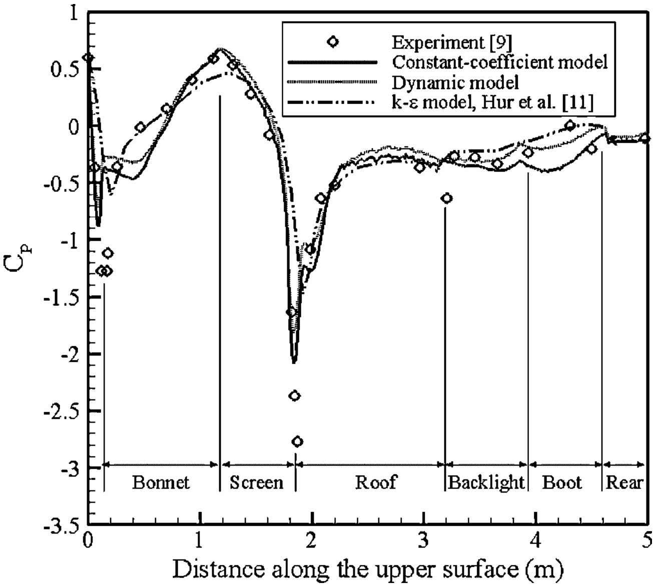

Figure 15. Comparison of C_p curves along the upper surface of the MIRA model obtained by LES models (Mesh III) with those obtained previously by experiment and two-equation turbulence model.

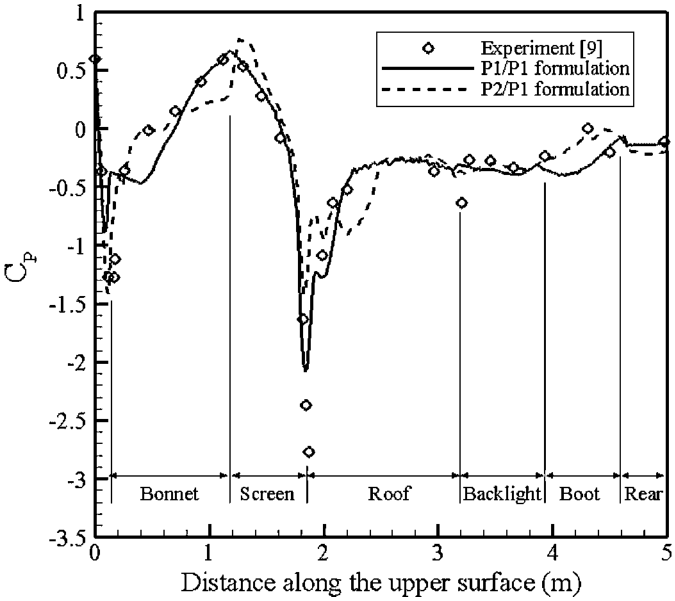

Figure 16. Comparison of C_p curves along the upper surface of the MIRA model obtained by using the constant-coefficient LES model for two different formulations.

DOI: 10.1002/fld

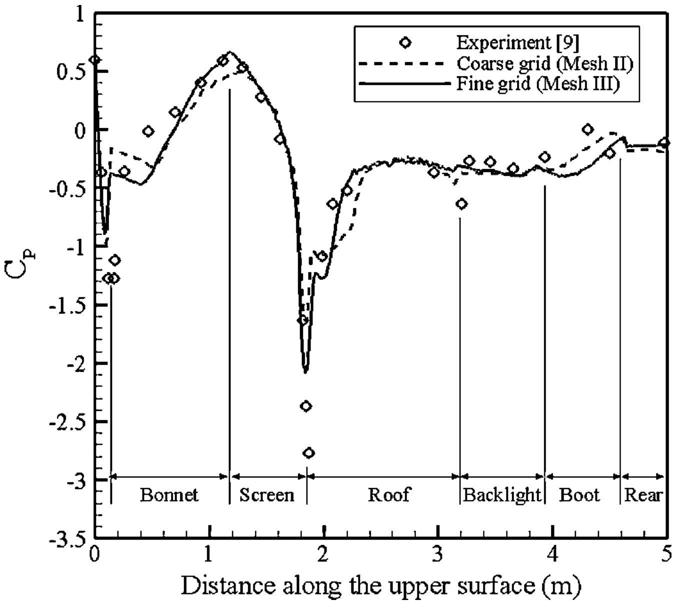

Figure 17. Comparison of C_{p} curves along the upper surface of the MIRA model obtained by using the constant-coefficient LES model for two different meshes.

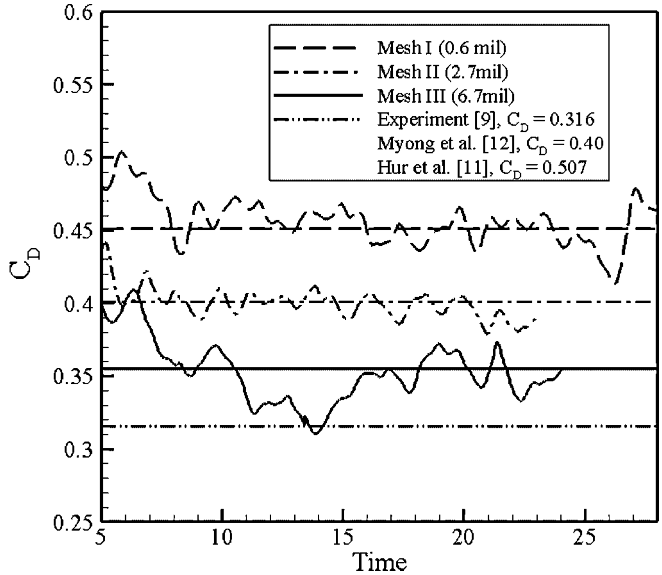

Figure 18. Comparison of drag histories of the MIRA model, obtained by using the constant-coefficient LES model for three different meshes at $Re = 2.6 \times 10^6$.

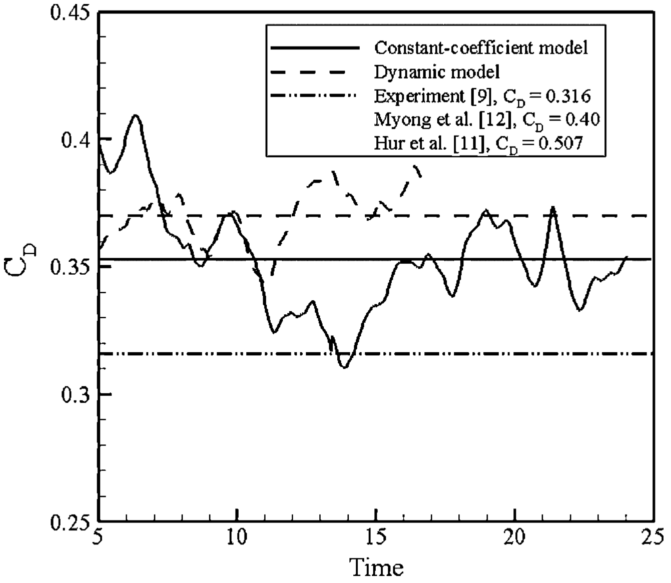

Figure 19. Comparison of drag histories of the MIRA model, obtained by using two Smagorinsky models at $Re = 2.6 \times 10^6$ (Mesh III).

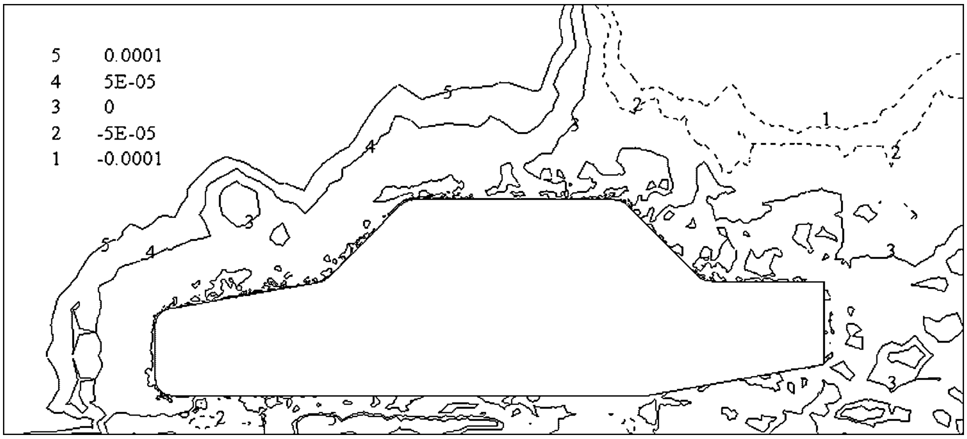

Figure 20. Instantaneous distribution of the dynamic LES model constant near the surface of the MIRA model.

direction of three-dimensional calculation. Thus, we can conclude that the mesh resolutions where the pressure peaks occur were not fine enough to predict exact results with second-order accurate discretization method. This can also be seen in Figure 17 where the pressure distributions obtained from two meshes, Mesh II and Mesh III, are compared. The distributions are almost similar to

DOI: 10.1002/fld

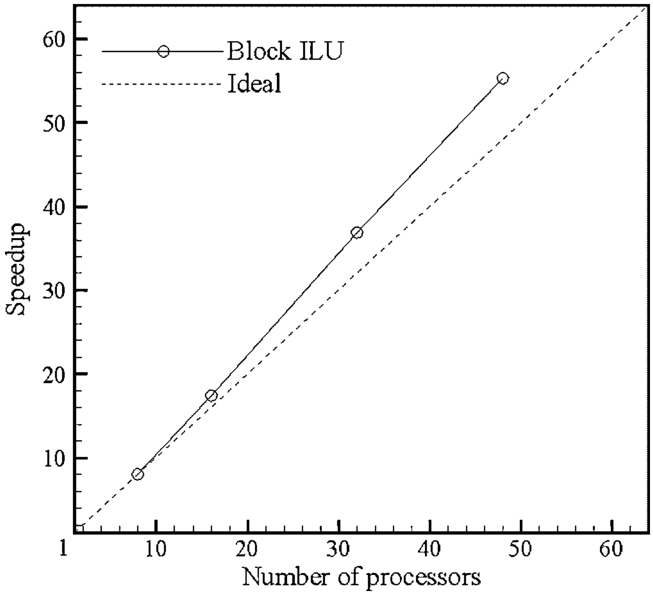

Figure 21. Speedup of parallel computation for solving the flow around MIRA (Mesh II).

each other except for the magnitude of the second peak. The finer mesh shows better accuracy at the peak, due to finer resolution.

Figure 18 shows the histories of drag coefficients predicted by the constant-coefficient LES model for three different meshes at $Re = 2.6 \times 10^6$. The drag coefficient approaches the experimental result as the number of nodes increases. In particular, the drag coefficient obtained from Mesh III (6.7 million nodes) is about 0.355, which is only 10% larger than the experimental one [9], 0.316, and gives quite a satisfactory result. In Figure 19, drag coefficients predicted with Mesh III using two Smagorinsky models are compared. The result predicted by the constant-coefficient model gives slightly better result than that predicted by the dynamic model. The reason for the higher drag prediction of the dynamic model is thought to be from the model constant calculation. In Figure 20, an instantaneous model constant distribution of the dynamic model near the surface of the MIRA model is shown. Unlike the turbulent channel flow, where the averaging process is performed on each constant y plane, model constants are negative in many regions near the surface. In the calculation, we have clipped out these negative model constants. Thus, the turbulent flow calculated in these regions is practically the same as those calculated without an LES model. As a result, the calculated drag coefficients are higher than that of the constant-coefficient model. It is found that mesh resolution of the present study is not fine enough near the solid boundary, so that the values of y^+ at the first nodes from the wall is around 50. Due to the above-mentioned reasons, more works need to be done in the future to apply the dynamic LES model with finer meshes or the detached eddy simulation model to the calculation of the turbulent flow around the MIRA model.

Lastly, a scalability curve obtained from the simulation of this large-scale problem is shown in Figure 21 when JS 20 at Seoul National University, whose CPU is PowerPC970 2.2 GHz 2-way with 1 Myrinet/1 Gigabyte Ethernet, is used. It is revealed that the present parallel code based on block ILU preconditioned CG algorithm is super-linearly scalable up to 48 processors due to cache memory effect. Therefore, it can be said that the present parallel algorithm is quite scalable for a large-scale computation.

4. CONCLUSIONS

In this work, 4-step splitting method is parallelized to solve the turbulent flow around the MIRA model at $Re = 2.6 \times 10^6$. Constant-coefficient and dynamic-Smagorinsky models are adopted for the modelling of small eddies in turbulent flow and the resulting solutions are compared.

In spite of good agreement in calculating turbulent channel flow and backward-facing step flow, the dynamic-Smagorinsky model caused the occurrence of negative model constants in calculating the turbulent flow around the MIRA model. As a result, the dynamic model predicted higher drag coefficient than the constant-coefficient model with the same mesh.

In contrast, the result obtained by the constant-coefficient model is in good agreement. Overall, the time-averaged pressure distribution agrees well with the experiment except for the two pressure peaks. There are some discrepancies in magnitude of the two peaks, but these are mainly attributed to insufficient mesh resolution near the edges of the upper surface. As the number of nodes increases, the predicted pressure distribution approaches the experimental result. In the viewpoint of the drag coefficient prediction, the present simulation gives quite a satisfactory result. The predicted drag coefficient is about 10% larger compared with the experimental result and agrees better than those obtained using two-equation turbulence models.

ACKNOWLEDGEMENTS

This work is supported by Next Generation Vehicle Technology Development Project, Ministry of Commerce, Industry and Energy, Republic of Korea and partially supported by Brain Korea 21 Project, Ministry of Education and Human Resources Development, Republic of Korea.

REFERENCES

1. Aliabadi S, Johnson A, Abedi J. Comparison of finite element and pendulum models for simulation of sloshing. *Computers and Fluids* 2003; **32**:535–545.
2. Aliabadi S, Tu S, Watts M, Ji A, Johnson A. Integrated high performance computational tools for simulations of transport and diffusion of contaminants in urban areas. *International Journal of Computational Fluid Dynamics* 2006; **20**:253–267.
3. Choi HG, Choi H, Yoo JY. A fractional four-step finite element formulation of the unsteady incompressible Navier–Stokes equations using SUPG and linear equal-order element methods. *Computer Methods in Applied Mechanics and Engineering* 1997; **143**:333–348.
4. Choi HG, Yoo JY. Streamline upwind scheme for the segregated formulation of the Navier–Stokes equation. *Numerical Heat Transfer Part B*: *Fundamentals* 1994; **25**:145–161.
5. Johnson A, Tezduyar TE. Parallel computation of incompressible flows with complex geometries. *International Journal for Numerical Methods in Fluids* 1997; **24**:1321–1340.
6. Tezduyar TE, Behr M, Mittal S, Johnson AA. Computation of unsteady incompressible flows with the stabilized finite element methods: space–time formulations, iterative strategies and massively parallel implementations. In *New Methods in Transient Analysis*, Smolinsky P, Liu WK, Hulbert G, Tamma K (eds). AMD, ASME: New York, vol. 143. 1992; 7–24.

DOI: 10.1002/fld

7. Rifai SM, Buell JC, Johan Z, Hughes TJR. Automotive design applications of fluid flow simulation on parallel computing platforms. *Computer Methods in Applied Mechanics and Engineering* 2000; **184**:449–466.
8. Choi HG, Nam YS, Yoo JY. Splitting finite element method for large eddy simulation. *ASME Fluids Engineering Division Summer Meeting*, *FEDSM2001-18122*, New Orleans, Louisiana, 2001.
9. Carr G. Validation of CFD codes for predicting aerodynamic performance. *Automotive Engineer* 1992; **17**:46–49.
10. Gaylard AP, Baxendale AJ, Howell JP. The use of CFD to predict the aerodynamic characteristics of simple automotive shapes. *SAE Paper 980036*.
11. Hur N, Ahn IK, Kim SR, Lee KH, Lee SC. A study on vehicle aerodynamics by using CFD: simulation of 3-D flow fields around MIRA reference vehicle model. *1994 Annual Report of AFERC*, *AFR-94-B04*. Advanced Fluids Engineering Research Center, POSTECH, Pohang, Korea, 1995; 105–143.
12. Myong HK, Jin E. Numerical simulation of three-dimensional turbulent flows around a MIRA vehicle model. *Proceedings of the KSME 1997 Spring Annual Meeting B*, Chonnam National University, Gwangju, Korea, 1997; 210–215.
13. Hajiloo A, Williams J, Hackett JE, Thompson SA. Limited mesh refinement study of the aerodynamic flow field around a car-like shape: computational versus experimental fluid dynamics. *SAE Paper 960677*.
14. Snir M, Otto S, Huss-Lederman S, Walker D, Dongarra J. *MPI*: *The Complete Reference*. The MIT Press: Cambridge, Massachusetts, 1996.
15. Radicati di Brozolo G, Robert Y. Parallel conjugate gradient-like algorithms for solving sparse nonsymmetric linear systems on a vector multiprocessor. *Parallel Computing* 1989; **11**:223–239.
16. Hysom D, Pothen A. A scalable parallel algorithm for incomplete factor preconditioning. *SIAM Journal on Scientific Computing* 2001; **22**:2194–2215.
17. Choi H, Moin P. Effects of the computational time step on numerical solutions of turbulent flow. *Journal of Computational Physics* 1994; **113**:1–4.
18. 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.
19. Allaire PE. *Basics of the Finite Element Method*: *Solid Mechanics*, *Heat Transfer*, *and Fluid Mechanics*. WCB Publishers: Dubuque, 1985; 49–51.
20. Reddy JN, Gartling DK. *The Finite Element Method in Heat Transfer and Fluid Dynamics*. CRC Press: Boca Raton, FL, 1994; 142–145.
21. Jansen K. A stabilized finite element method for computing turbulence. *Computer Methods in Applied Mechanics and Engineering* 1999; **174**:299–317.
22. http://www-users.cs.umn.edu/∼karypis/metis
23. Van der Vorst HA. BI-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing* 1992; **13**:631–644.
24. Carey GF, Shen Y, McLay RT. Parallel conjugate gradient performance for least-squares finite elements and transport problems. *International Journal for Numerical Methods in Fluids* 1998; **28**:1421–1440.
25. Jiang BN, Lin TL, Povinelli LA. Large-scale computation of incompressible viscous flow by least-squares finite element method. *Computer Methods in Applied Mechanics and Engineering* 1994; **114**:213–231.
26. Saad Y. *Iterative Methods for Sparse Linear Systems*. PWS Publishing Company: Boston, 1996.
27. Kim J, Moin P, Moser R. Turbulence statistics in fully developed channel flow at low Reynolds number. *Journal of Fluid Mechanics* 1987; **177**:133–166.
28. Sarghini F, Piomelli U, Balaras E. Scale-similar models for large-eddy simulations. *Physics of Fluids* 1999; **11**:1596–1607.
29. Le H, Moin P, Kim J. Direct numerical simulation of turbulent flow over a backward-facing step. *Journal of Fluid Mechanics* 1997; **330**:349–374.
30. Jovic S, Driver DM. Backward-facing step measurements at low Reynolds number, $Re_h = 5000$. *NASA Technical Memorandum 108807*, 1994.
31. Friedrich R, Arnal M. Analysing turbulent backward-facing step flow with the lowpass-filtered Navier–Stokes equations. *Journal of Wind Engineering and Industrial Aerodynamics* 1990; **35**:101–128.

DOI: 10.1002/fld