



Journal of Applied Sciences

ISSN 1812-5654

science
alert

ANSI*net*
an open access publisher
<http://ansinet.com>

Large Eddy Simulations of Turbulent Flow Across Tube Bundle Using Parallel Coupled Multiblock Navier-Stokes Solver

M. Talebi, E. Shirani and M. Ashrafizaadeh

Department of Mechanical Engineering, Isfahan University of Technology, Isfahan, Iran

Abstract: Turbulent flow around a tube bundle is simulated. Fully coupled Navier-Stokes equations in non-orthogonal coordinate system are solved employing a new multiblock parallel solver. For turbulent modeling, the Large-Eddy Simulation (LES) technique with the Smagorinsky subgrid eddy viscosity model is used. The discretized governing equations are second order in time and space and to prevent the pressure checkerboard problem, a Momentum Interpolation Method (MIM) is used. Parallelization of the computer code was done by a load balanced domain decomposition technique which results in a high computational efficiency and speed up. Flow pattern and characteristics are obtained for turbulent flow over five parallel tubes rows. Numerical results show a very good agreement with experimental results and reveal interesting flow behavior for such complicated industrial applications.

Key words: Large eddy simulation, tube bundle, parallel coupled Navier-Stokes solver

INTRODUCTION

Tube bundles are used in many engineering structures, such as heat exchangers, boilers and steam generators, fuel assembly in nuclear power plants, underwater piping and naval systems. Each of these applications involves a fluid flowing across array of tubes. The majority of the flows over tube bundle are turbulent, that is, unsteady, three-dimensional and highly vortical. The hydrodynamic forces applied on the tubes in such flows are oscillating forces. These forces cause sever oscillations with large amplitudes and may damage the tubes and their connections. The Fluid Induced Vibrations (FIV) imposed on the tubes is one of the important problems in analyzing the tube bundles and causes several problems. Specially in large systems, they may cause major problems and even shut down the whole system. The FIV effect is more important in nuclear system. Any leakage of nuclear fluid from primary or secondary loop in steam generators or other systems that contain tube array may cause major human damages. Thus it is important to consider this problem with special care.

Turbulent flows around tube bundles are very complicated. The most common method used to simulated turbulent flows is the use of Reynolds Averaged Navier-Stokes (RANS) equations. Although most of the practical problems are solved using RANS methods, there are a number of flows where these

methods cannot be used and they are not able to resolve the detail of the flow structures. It is because too much information is filtered out in the averaging process. Also, in some unsteady flows, interfacial flows and separated flows, it is not appropriate to use the concept of mean and fluctuating velocity as they are both of the same magnitude^[1]. Flow around and behind bluff bodies is another example where the RANS methods are not the appreciate way of simulating this problem when it is meant to obtain FIV effect.

Numerical simulation using LES technique is a good candidate for such flows. It can be used to calculate hydrodynamic forces and the related oscillations^[2]. With such technique and a suitable numerical method, non-orthogonal coordinate system and parallel computing, one can simulate the flow and obtain the major features the flow.

Freddie *et al.*^[3] simulated two dimensional flows over a steam generator tube bundle using the finite element method, Smagorinsky model and segregated solution technique. They obtained stream lines and velocity vectors for flow over seven rows tube configuration in a channel. They did not use fine mesh due to the very large number of tubes, so their solution resolution was not good enough and they only showed the general flow configurations. Barsamian and Hassan^[4] simulated two-dimensional flow over a tube bundle in a channel using the GUST computer code and supercomputer. They used four rows (each row contains seven tubes). The

Smagorinsky and Bardina's model were used in their work. They presented time variations of hydrodynamic forces on the tubes. The Power Spectral Density (PSD) of lift and drag forces which were obtained by their modified version of the turbulence model gave similar results compared to the Smagorinsky model. Yassin *et al.*^[5] also considered two-dimensional flow over 12 parallel tubes numerically. They arranged the tubes in four rows. The computational grid used in their work was rather coarse. They used the GUST code and a supercomputer. They presented the velocity vector and streamlines. Chen and Jendrzejczyk^[6] considered such flows experimentally. They used 49 tubes arranged in seven rows in a channel with square cross-section. They tested flows with Reynolds numbers 10^4 to 10^6 . They showed that the flow characteristics such as lift and drag coefficients and the flow patterns are similar for the third row and the rows after it. Weaver and Abd-Rabbo^[7] also considered the flow around tube bundles experimentally. They examined both laminar and turbulent flows. They showed that the flow becomes turbulent as the Reynolds number exceeds 400. They also showed that the flow patterns and characteristics are only different around the first three rows and are similar for the other rows. Beale and Spalding^[8] simulated laminar flow over two parallel tubes. They showed that since the flow is unsteady and laminar and the oscillations are periodic, with suitable initial and boundary conditions one can obtain reasonable results.

Almost in all of the numerical simulations performed by others for flow over tube bundles, rather large number of tube rows was used. Thus they had to either choose rather coarse grid with very low resolution of the solution or to use supercomputer. In the present work, we avoid such problem by using only five rows and also we use parallel computing technique to increase the computational speed without the need for supercomputers. As mentioned above, according to the experimental results obtained by Chen and Jendrzejczyk^[6], Weaver and Abd-Rabbo^[7], only the flow over the first three rows are different and for the rest of the rows the flow is nearly periodic. So the use of five rows with appreciated boundary conditions is good enough to simulate such flows.

Other researchers have so far used either serial (unparallel) codes or supercomputers. The serial codes are not suitable for simulations of such problem when LES technique is used. Because, in LES techniques, one need to have rather small grid size (the large number of grids) and it is very time consuming. In this study, in order to increase the computational efficiency, we parallelized the computer code and used 24 parallel PC's. Although

there have been several works for parallelization of Navier-Stokes and Euler equations solvers, but such computer codes are based on either explicit or segregated methods. Bui^[9] used parallel, explicit, three-dimensional algorithm based on finite volume approach for the solution of turbulent flows using LES technique. He used second order MacCormack time marching algorithm. Karlo and Tezduyar^[10] developed a parallel three-dimensional finite element implicit Navier-Stokes solver for unsteady incompressible flows and used it to solve unsteady flows around circular cylinders. Strietzel^[11] implemented a parallel three-dimensional incompressible Navier-Stokes solver by using LES and Direct Numerical Simulation (DNS) techniques.

In this study, a multiblock parallel implicit fully coupled Navier-stokes solver is developed and used for the first time. The ability and efficiency of the code in simulating the flows over a tube bundle is examined. The flow is unsteady and the code is able to handle complex geometry because of the use of the non-orthogonal coordinate systems. The parallelization process is based on a domain decomposition method such that the whole computational domain is divided into several sub-domains. The number of sub-domains depends on the number of processors and the configuration of the domain geometry. The flow inside of each sub-domain is solved separately by a single processor. The required information of each sub-domain is transferred from its neighboring sub-domains along their attached boundaries. To remove the checkerboard problem for the pressure calculation in the collocated grid, the momentum interpolation method^[12] is used and the mass and momentum equations are solved simultaneously using a coupled solver. In the present work a 24-node PC cluster was used for computations. The computational nodes are Intel P4 CPU's with 1GB of RAM and their network is a fast Ethernet. The operating system of the cluster is Linux. Also the implementation of the Message Passing Interface (MPI) which is giving MPICH^[13] and is publicly available is used.

To model the turbulence, we use the LES technique. In this method, the grid size must be small enough to resolve large structure of turbulent flow. Thus, it is difficult to simulated the flows with the complicated geometries. In order to minimize the number of grid points and at the same time, to simulate the turbulent flow around the tube bundles, we follow the conclusion made by experimental results^[6,7], that only the first three rows of the tube bundle have different characteristics and the rest of the rows have similar behavior, e.g. flow becomes fully developed hydrodynamically.

MATERIALS AND METHODS

Governing equations: The governing equations are continuity and momentum equations for unsteady and incompressible flows.

Momentum equations:

$$\frac{\partial u_i}{\partial t} + \frac{\partial u_i u_j}{\partial x_j} = -\frac{1}{\rho} \frac{\partial p}{\partial x_i} + \nu \frac{\partial^2 u_i}{\partial x_j \partial x_j} \tag{1}$$

Continuity equation:

$$\frac{\partial u_i}{\partial x_i} = 0 \tag{2}$$

In the LES method, where the large eddies are simulated and only the small eddies are modeled, each of the variables is separated into mean and fluctuating components:

$$f(x,t) = \bar{f}(x,t) + f'(x,t) \tag{3}$$

where, \bar{f} is the filtered or resolvable part and is related to the grid-scale. f' is the non-resolvable part and is related to the Subgrid-scale (SGS). In the space-filtering method, the instantaneous flow variables are integrated over a given volume to generate the space-filtered variable. In order to clearly define terms in LES technique, Leonard^[14] proposed a filtering method for defining the large scale velocity field, \bar{u}_i , as:

$$\bar{u}_i(\bar{x},t) = \int_V G(\bar{x}-\bar{x}') u_i(\bar{x}',t) d\bar{x}' \tag{4}$$

where, $G(\bar{x}-\bar{x}')$ is the filter function that is utilized in the integration. It may be expressed as:

$$G(\bar{x}-\bar{x}') = \prod_i G_i(x_i-x'_i) \tag{5}$$

where, $G_i(x_i-x'_i)$ is a one-dimensional filter. The top-hat filter (box filter), cut off filter and Gaussian filter are common types of filters. The box filter uses the mesh as its boundaries and is the most widely used filter function. Applying this filter, the filtered Navier-Stokes equations become:

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial \bar{u}_i \bar{u}_j}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_i} + \nu \frac{\partial^2 \bar{u}_i}{\partial x_j \partial x_j} - \frac{\partial \tau_{ij}}{\partial x_j} \tag{6}$$

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0 \tag{7}$$

where, the subgrid stresses, τ_{ij} , are given by:

$$\tau_{ij} = \overline{u_i u_j} - \bar{u}_i \bar{u}_j = (L_{ij} + C_{ij} + R_{ij}) \tag{8}$$

The subgrid stresses includes Leonard stresses term, L_{ij} cross stresses term, and C_{ij} Reynolds stresses term, R_{ij} . These terms should be modeled.

Subgrid scale modeling: For modeling the subgrid scale stresses, the Smagorinsky^[15] eddy viscosity model is used. In this model, the cross and Leonard terms are zero due to the top-hat filtering used in the formulation. The proportionality between the anisotropic part of the SGS Reynolds stresses and the large scale strain rate tensors is assumed as:

$$\tau_{ij} = -2\nu_t \bar{S}_{ij} \tag{9}$$

where, ν_t is the SGS eddy viscosity and is generally assumed to be a scalar quantity and \bar{S}_{ij} is the large scale strain tensor. The following definitions are applied for the SGS eddy viscosity:

$$\nu_t = l^2 |\bar{S}| \tag{10}$$

$$|\bar{S}| = \sqrt{2\bar{S}_{ij}\bar{S}_{ij}} \tag{11}$$

and

$$\bar{S}_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \tag{12}$$

where, l is the subgrid length scale. In the vicinity of wall, the Van Driest damping function^[16,17], is used:

$$l = C_s \Delta \left[1 - \exp\left(-\frac{y^*}{25}\right) \right]^{1/2} \tag{13}$$

where:

$$y^* = \frac{yu^*}{\nu} \tag{14}$$

$$u^* = \sqrt{\tau_w/\rho} \tag{15}$$

C_s is the model parameter ranging from 0.1 to 0.2^[18] and Δ is the length scale and usually determined by the size of the control volume of the grid. Finally the integral form of the governing equations are:

$$\frac{\partial}{\partial t} \int_V \rho u_i dV + \int_S \rho u_i \vec{V} \cdot \vec{n} ds - \int_S \frac{\partial p}{\partial x_i} dV - \int_S (\mu + \rho \nu_t) \chi \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \cdot \vec{n} ds - \int_V Q_i dV = 0 \tag{16}$$

and

$$\frac{\partial}{\partial t} \int_V \rho dV + \int_S \rho \vec{V} \cdot \vec{n} ds = 0 \tag{17}$$

Numerical method: To discretize the governing equations, we use control volume approach with collocated grid. Figure 1 shows a computational cell. In this figure, P is the cell center and e, w, n and s are points at the cell faces. The collocated grid makes the parallel programming much easier, specially for non-orthogonal grid that is used in this study. For the collocated grid, all boundary conditions, including the attached boundary conditions, can be easily defined. The transient terms of the governing equations are discretized using either the first order Euler or the second order Crank-Nicolson method. For the diffusion term, the second order central differencing scheme and for the convective term, the second order differed correction method^[19] are used. The resultant linearized equations are:

u-momentum :

$$Cu_p^u * u_p + Cu_N^u * u_N + Cu_S^u * u_S + Cu_E^u * u_E + Cu_W^u * u_W + Cp_P^u * p_P + Cp_N^u * p_N + Cp_S^u * p_S + Cp_E^u * p_E + Cp_W^u * p_W = b_u \quad (18)$$

v-momentum:

$$Cv_P^v * v_P + Cv_N^v * v_N + Cv_S^v * v_S + Cv_E^v * v_E + Cv_W^v * v_W + Cp_P^v * p_P + Cp_N^v * p_N + Cp_S^v * p_S + Cp_E^v * p_E + Cp_W^v * p_W = b_v \quad (19)$$

continuity:

$$Cp_P^p * p_P + Cp_N^p * p_N + Cp_S^p * p_S + Cp_E^p * p_E + Cp_W^p * p_W + Cu_P^p * u_P + Cu_N^p * u_N + Cu_S^p * u_S + Cu_E^p * u_E + Cu_W^p * u_W + Cv_P^p * v_P + Cv_N^p * v_N + Cv_S^p * v_S + Cv_E^p * v_E + Cv_W^p * v_W = b_u \quad (20)$$

where, the subscript P indicates the point at the cell center and subscripts N, S, E and W refer to north, south, east and west points of P, respectively. Superscripts u, v and p correspond to the u, v and p equations, respectively. For example Cp_W^v is the pressure coefficient corresponding to the west point in the v-equation.

To obtain the pressure equation, the x-momentum is first written for three control volumes corresponding the E, P and e points. Then we combine these three equations and calculate the advective velocity, \hat{u}_e . The same procedures are followed to obtain, \hat{u}_w , \hat{u}_n and \hat{u}_s . Finally the advective velocity at points e, w, n and s are substituted into the continuity equations and the pressure is calculated from the resulting equation^[12].

Parallelization method: The parallelization of the Navier-Stokes solver is based on a domain decomposition

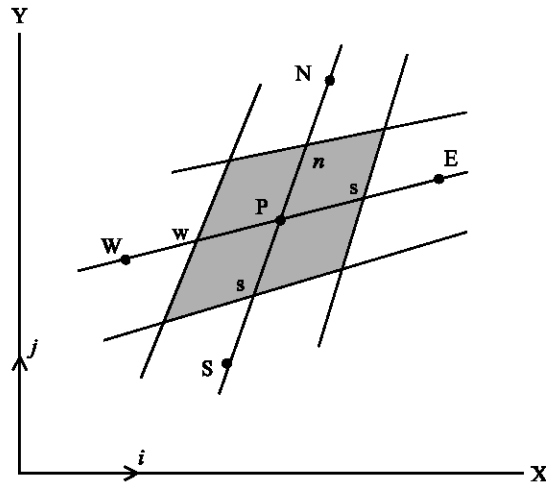


Fig. 1: A typical non-orthogonal grid control volume and the notation used

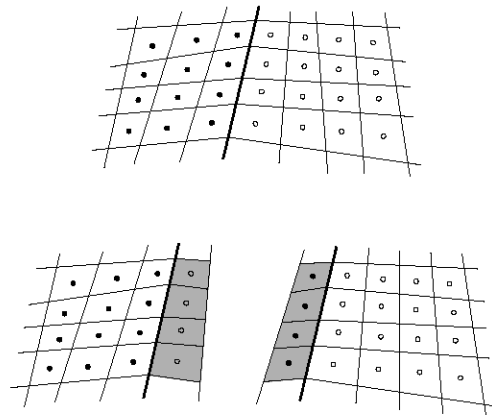


Fig. 2: Attached boundary and its shadow row neighbor

method. In this method, the whole computational domain is divided into several sub-domains and each processor in the parallel job execution performs all the computations within one or more sub-domains. The number of sub-domains depend on the problem size and the desired size of the sub-domain tasks. If the number of sub-domains is higher than the number of processors, more than one sub-domain maybe associated with a processor to obtain the optimum efficiency.

The sub-domains do not overlap and each processor computes its own nodal values. However, since the boundary nodes along the attached boundaries require some information of the neighboring nodes which reside on a neighboring sub-domain, it is necessary to have an extra shadow row along these attached boundaries^[20]. These shadow rows are used to store the required information from the neighboring blocks (Fig. 2). The

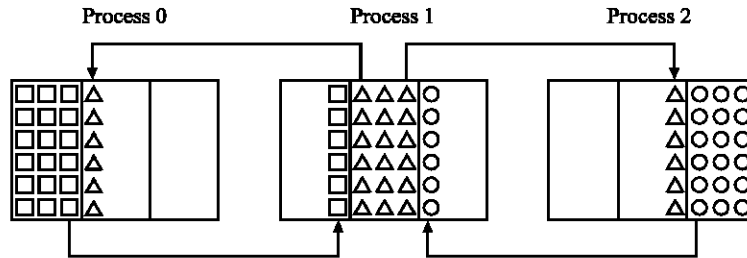


Fig. 3: Communication between three processors

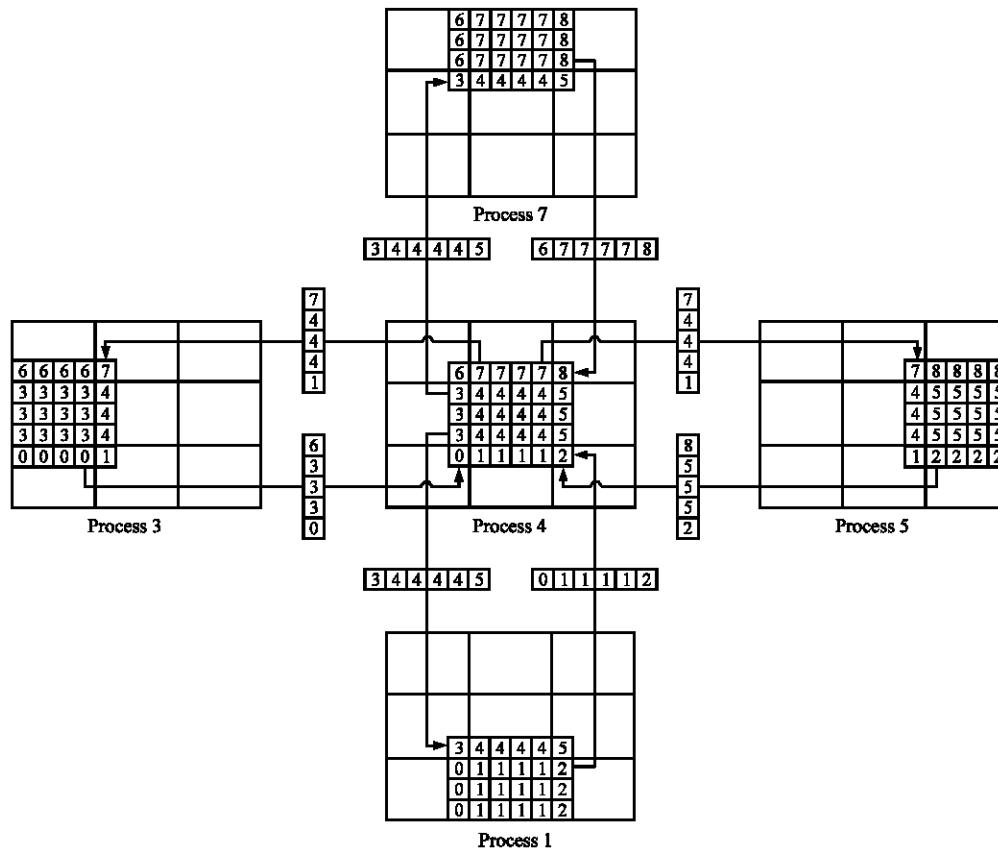


Fig. 4: Communication between one processor and four neighbor processors

values stored in these shadow cells have to be updated and exchanged after each step (overlapping of storage).

One should note that the attached boundaries are quite different compared to other types of boundaries in the sense that all the calculated coefficients are done exactly like those of interior nodes. However, the equation for each interior node uses the information of the nodes around it (Fig. 3 and 4), whereas the nodes along an attached boundary, have at least one neighbor along a shadow row. This requires that the nodal values at any shadow rows comes from the corresponding values of the previous iteration at the neighboring sub-domain. Therefore, the convergence rate might become slower than that of a single block serial code. Our numerical

experiences show that this problem can be alleviated by performing few interior iterations and updating all required nodal values and coefficients along the shadow rows.

Algorithm of main calculations specially for the parallelization contains the following steps:

- Initialization (initialization of boundaries for each sub-domain).
- Grid generation (generation the shadow row grid property).
- Calculation of old quantities.
- Determination coefficients of u, v and p equations (calculation and transfer of shadow row coefficient).

- Determination of boundary conditions (improve, transfer and calculation of boundary conditions for attachment boundary).
- Calculation of residual (in each sub-domain we calculate its maximum residual, then the main Processor finds the maximum of these residuals and based on this value, the calculations will be continued).
- Solution of the equations (each processor solves equations for a sub-domain).
- Updating values.
- Checking the time (return to step three if needed)

RESULTS AND DISCUSSION

To validate the computer code, the accuracy of the results and to demonstrate the efficiency of parallelization process, laminar flow inside Non-orthogonal Lid Driven Cavity (NLDCF) is simulated. The results obtained for both serial and parallel codes and compared with each other. Then turbulent flow over the tubes bundle is simulated.

Non-orthogonal Lid Driven Cavity Flow (NLDCF): The skewed cavity with angle 45 degree at Reynolds number 1000 has been solved. Figure 5 shows a typical domain which is decomposed into 12 sub-domains. Figure 6 shows the streamlines for this case. As it is shown, the continuity and smoothness of the streamlines at the attached boundaries are very good and there is no wiggle. The velocity profiles along the vertical and horizontal cavity centerlines are shown in Fig. 7 and 8. The results obtained by Demirozic *et al.*^[21] shown, very good agreements between the two results. In addition, Fig. 9 and 10 show the velocity components along the centerline for parallel and serial (calculation with one processor) cases. The results demonstrate the accuracy of the parallel simulations. This indicates that the parallelization of the code works very well and produces results identical to the serial results. To compute the efficiency of the parallelization, the speed up and efficiency of the parallelization are defined as:

$$Su = \frac{\text{CPU time for the serial code execution}}{\text{CPU time for the parallel code execution}} \quad (21)$$

and

$$\eta = \frac{1}{n} Su \quad (22)$$

respectively, where η is the number of processors. Table 1 shows the obtained values for Su and η for the NLDCF problem with different numbers of processors and the configurations of sub- domains.

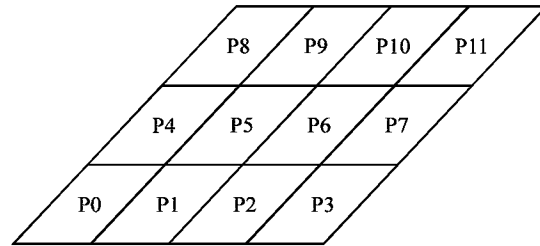


Fig. 5: Sample domain and its division to sub-domains for NOLDCF problem at alfa

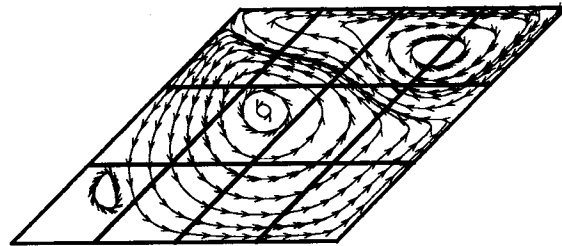


Fig. 6: Streamlines for NOLDCF at Re=1000 calculated with 12 processors

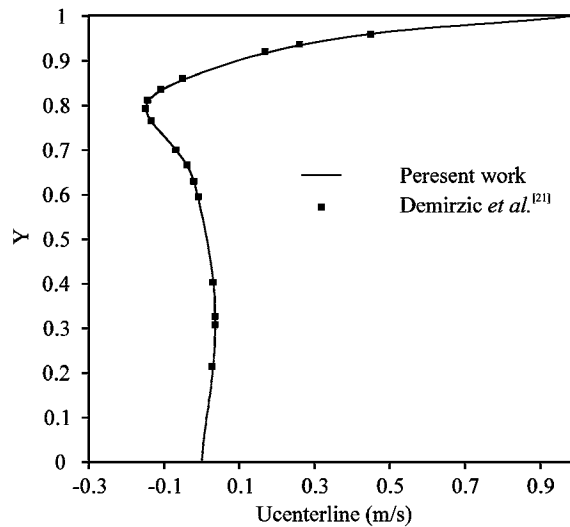


Fig. 7: x-component of the velocity profile along vertical centerline of non-orthogonal cavity

As can be seen, the CPU-time for the parallel code is reduced by a factor 3.56 to 20.49, depending on the numbers of the processors (2 to 12), when compared to that of the serial code. The first two rows in the Table 1 are for the cases where two sub-domains are used. The first row is for when the boundary between the two sub-domains is vertical and the second row is for the case where it is horizontal. Comparing the first two rows of the table, we see that when the domain is decomposed

Table 1: The speed up and efficiency for NLDCE problem

No. of processors	Sub-domain configuration	Su	η
2	Splitting vertically	3.56	1.78
2	Splitting horizontally	4.00	2.00
4	Splitting symmetrically	8.06	2.02
9	Splitting symmetrically	17.84	1.98
12	3(horizontally) and 4(vertically)	20.49	1.70

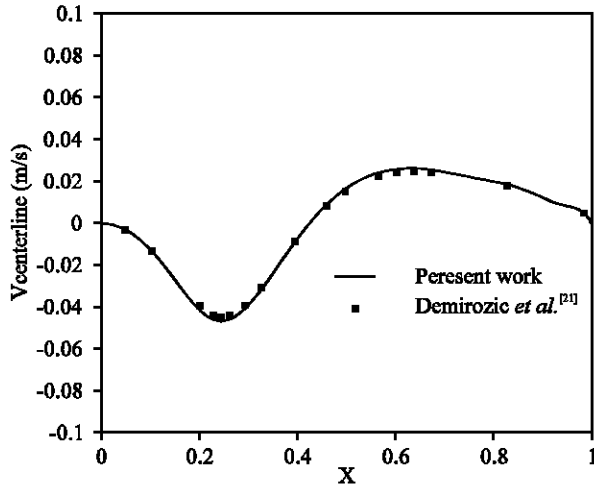


Fig. 8: y component of the velocity profile along horizontal centerline of non-orthogonal cavity

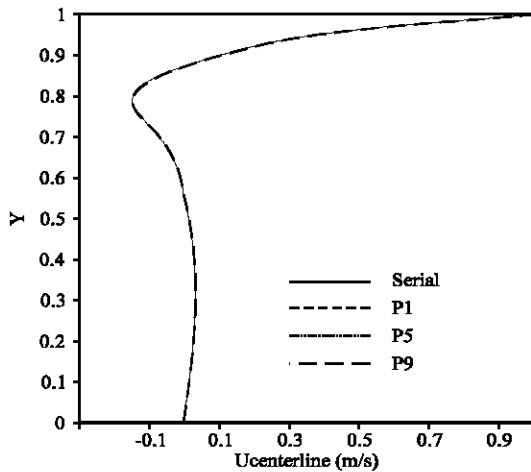


Fig. 9: x component of the velocity profile along vertical centerline of non-orthogonal cavity

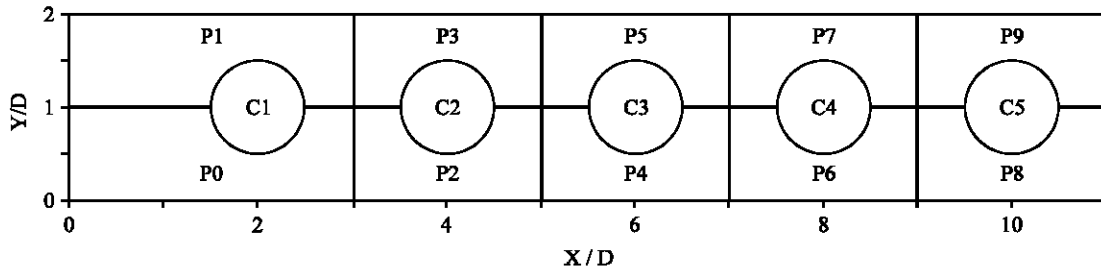


Fig. 11: Sample domain and its division to ten sub-domains

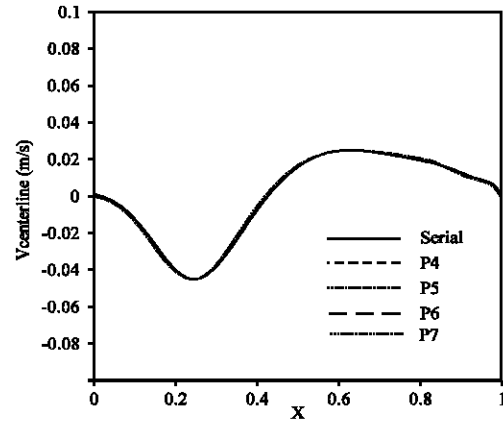


Fig. 10: y component of the velocity profile along vertical centerline of non-orthogonal cavity

horizontally, the CPU time is less and the efficiency of the parallelization is better. The reason is that the flow characteristics in the cavity, as show in Fig. 6, is more correlated horizontally than vertically. Thus it is more reasonable to divide the domain by a horizontal line than a vertical line and the calculations become more efficient when the domain is divided horizontally.

Another interesting problem is that, as it is shown in the Table 1, the efficiencies are bigger than one. The reason is due to the properties of the direct sparse solver (SPARSPAK), used in the code. This solver solves the coupled linear algebraic equations by a direct method, that is no iterative process is used^[22]. The time is required to solve directly a system of the equations increases non-linearly with the numbers of the equations. So as the number of the equations decreases as it is the case for the sub-domains, the relative CPU time required to solve the equations is decreased. Thus despite of the fact that in parallel processing, one expects to see slight increase in the CPU time due the transfer of data between the sub-domains and the extra algebraic operations needed for the interpolation of the data at the sub-domain boundaries, the CPU time required to solve the system of algebraic equations in each sub-domain multiply by

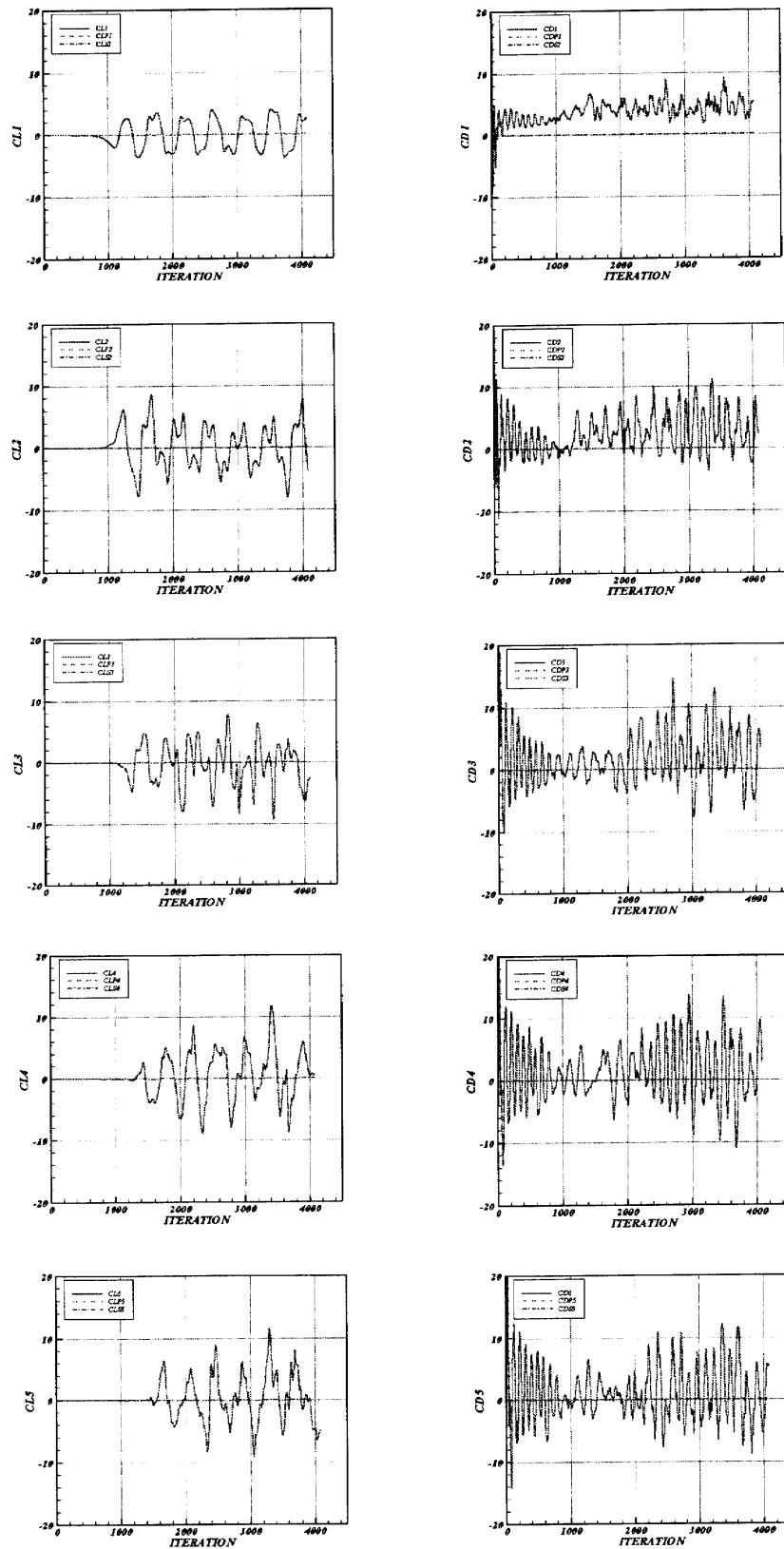


Fig. 12: Time variation of the lift (left) and Drag (right) coefficient for first to five tubes

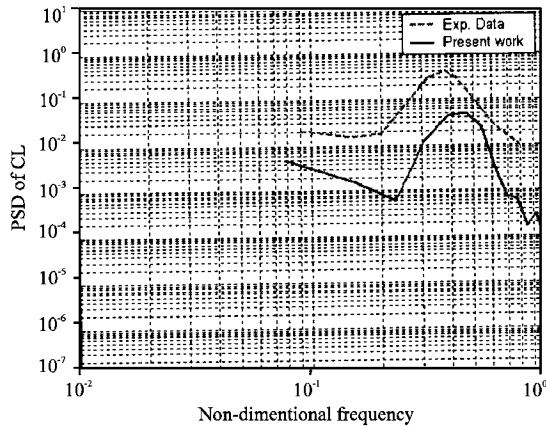


Fig. 13: PSD of Lift force via Non_dimensional frequency

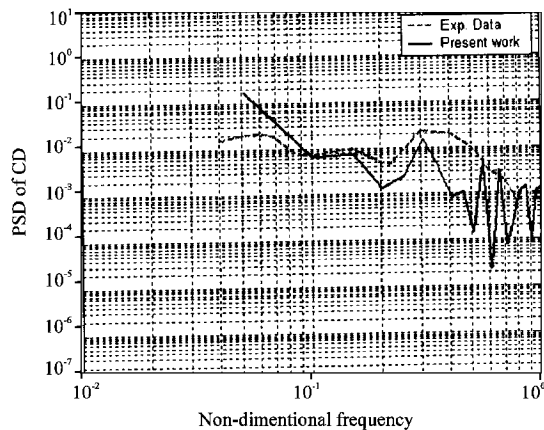


Fig. 14: PSD of Drag force via Non_dimensional frequency

numbers of sub-domains is much less than to CPU time for a serial calculation. That is, the sum of the CPU times used by processors is less than the total time needed by one processor to solve the undivided whole set of system of equations (serial code). This is called super-linear Speed Up, As an example, where 242*242 grid points are used, the time which is required by serial code for one time step is 182 sec, but when 9 processors are used, the time which is required for each processors is 6.65 sec. For this case, the number of grid points in each sub-domain is one-ninth of the whole grid points plus the extra points at the sub-domain boundaries, that is 82*82. So the total time used by nine processors is 6.65*9=58.85 sec which is much less than 182 sec. If we solve the same problem with only one processor, but with coarser grid, 82*82, then the CPU time is 6.16 sec. So the time for data transfer between the sub-domains and the interpolation is about 6.65-6.15=0.5 sec. It indicates that the parallel code has high efficiency. The reason is not only because the time required for data transfer between sub-domains is negligibly small, but also due to the super-linear effects.

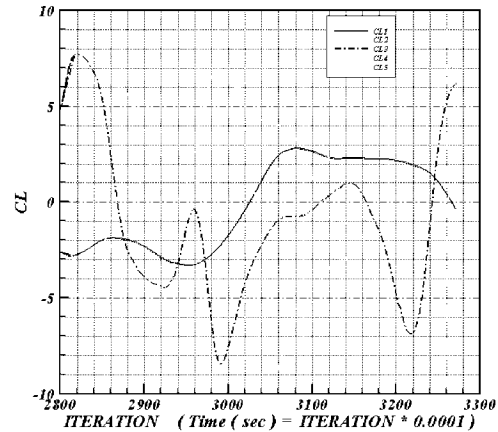


Fig. 15: Lift coefficient variation for five tubes (along 0.2820 to 0.3340 sec)

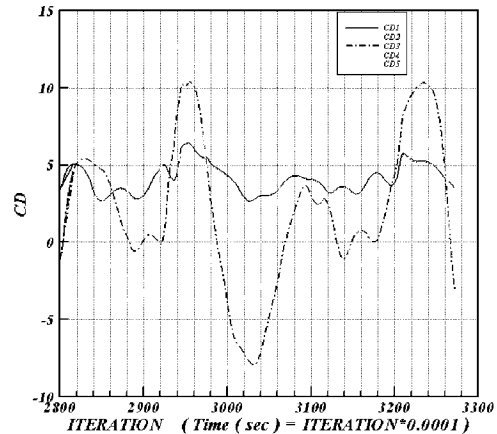


Fig. 16: Drag coefficient variation for five tubes (along 0.2820 to 0.3340 sec)

Turbulent flow around tube bundles: As mentioned before, we use five tube rows, in our simulations. For the last row, the pressure drop from the forth row and the velocity profile of the flow behind of the fourth row is used. These conditions are used by assuming the fully developed flow around the fourth and fifth rows. We also use periodic boundary conditions for the top and bottom boundaries. Figure 11 shows the problem domain and its decomposition to ten sub-domains for parallelization. The tube diameter is 20 mm. The pitch to diameter ratio of the tubes is 2. The Reynolds number based on the gap velocity and diameter of the tubes is 53300. These figures correspond to the values used in an experiment done by Oengoeren and Ziada^[23].

Figures 12 show the time variations of lift and drag coefficients for each tube row. The lift and drag coefficients are normalized lift and drag forces enforced on the tube due to the motion of fluid. The normalization

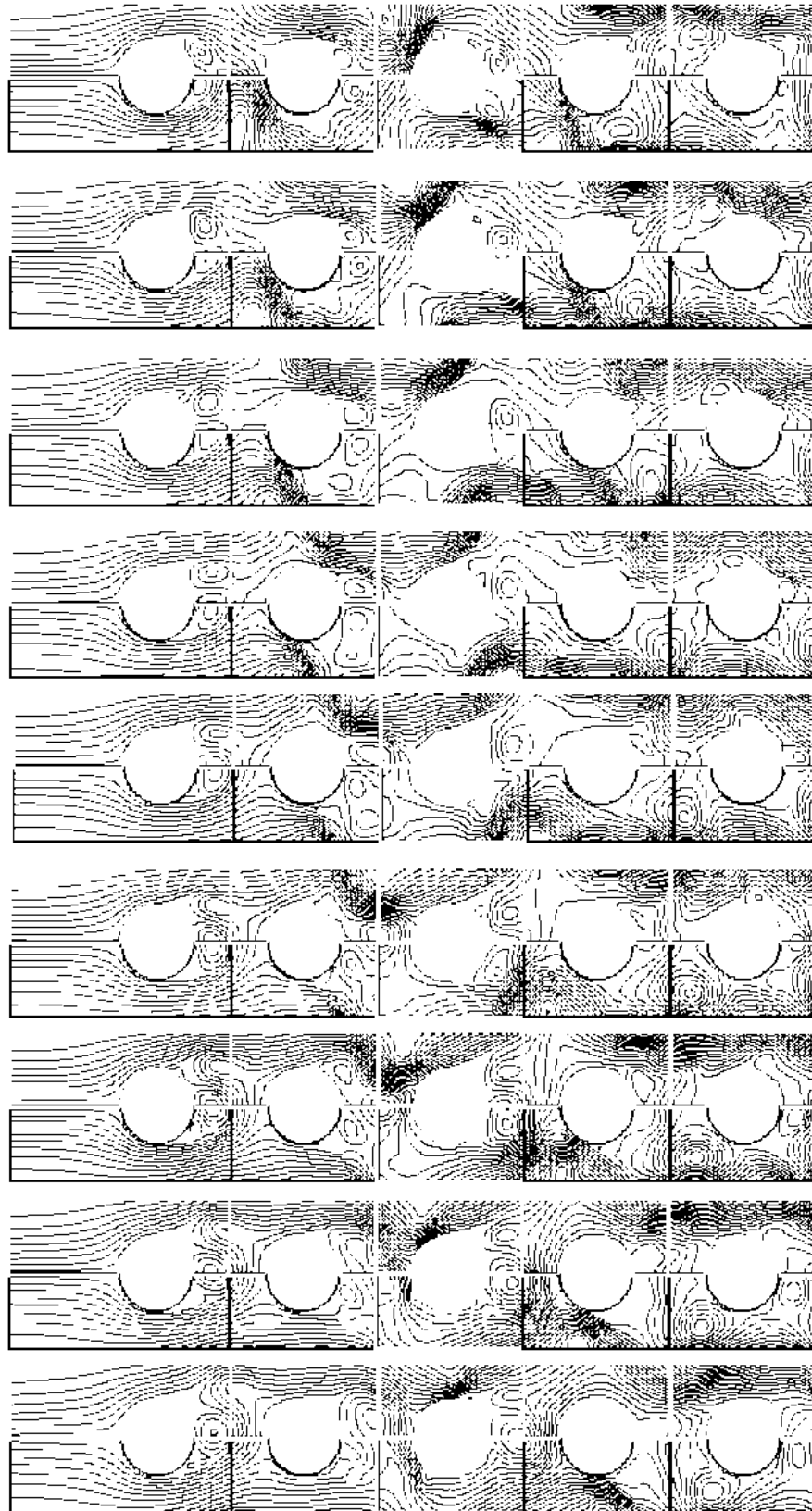


Fig. 17: Stream lines from time 0.2820 to 0.3340 (sec) (iteration is from 2820 to 3340 with interval=20)

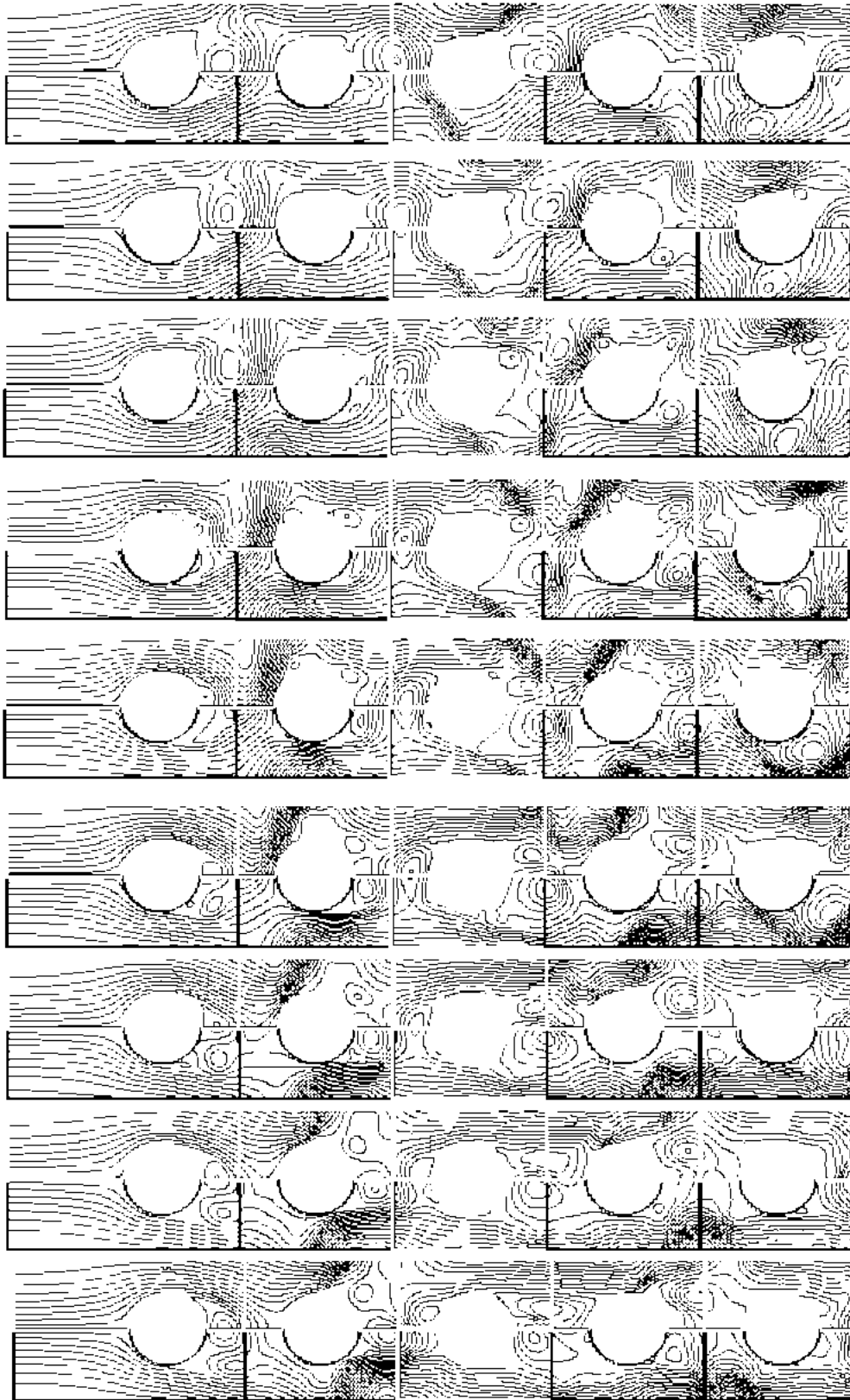


Fig. 17: Continue

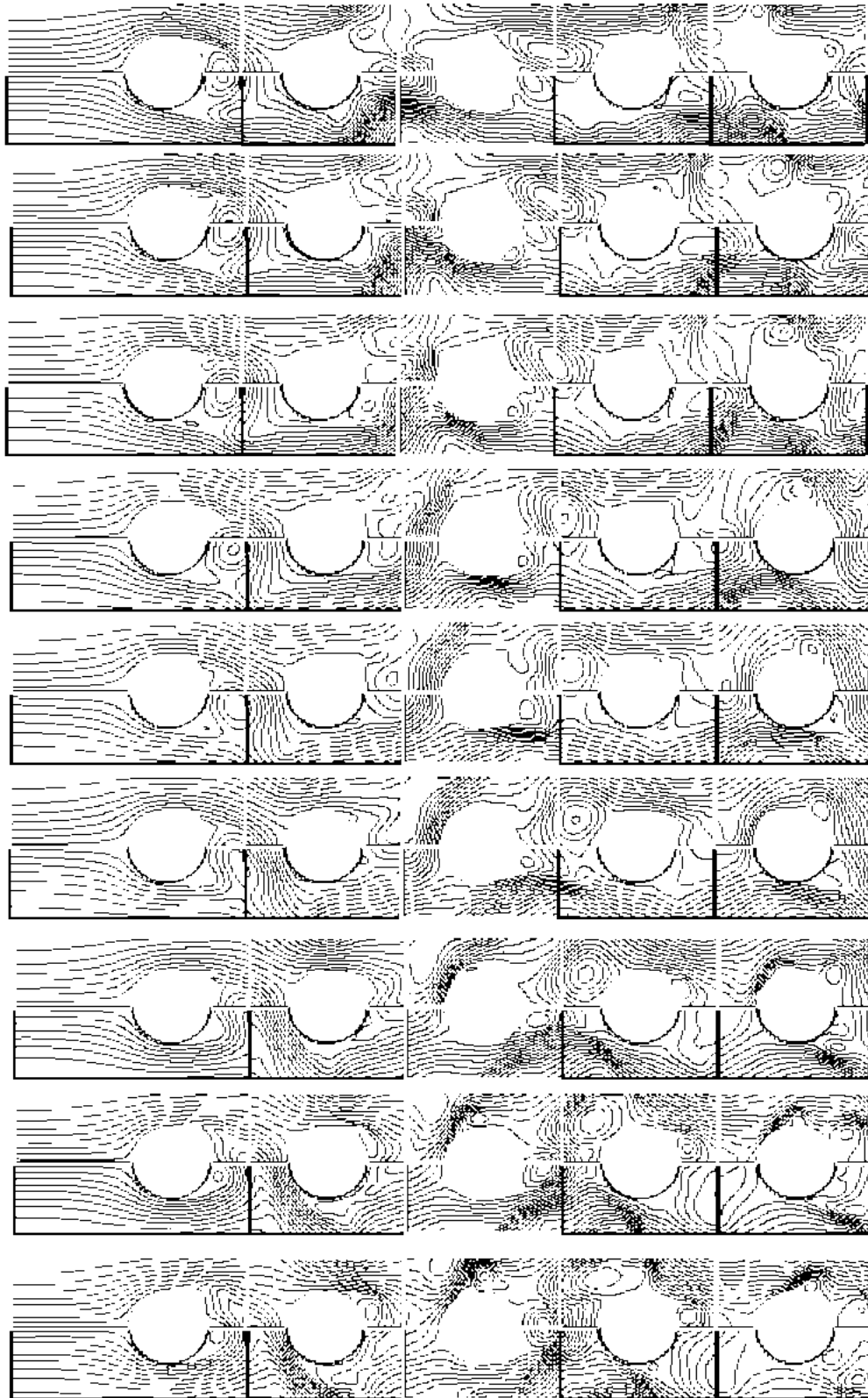


Fig. 17: Continue

is based on the density and the mean velocity of the flow and the tube diameter. In these figures $Re=53300$ and time step is 0.1 millisecond. CL, CLP and CLS are total, pressure and shear lift coefficient, respectively and CD, CDP and CDS are total, pressure and shear drag coefficient respectively. The pressure lift (or drag) coefficient is part of the total lift (or drag) coefficient that corresponds to the pressure force acting on the tubes and the shear lift (or drag) coefficient is the part of the total lift (or drag) coefficient that corresponds to the shear force acting on the tubes. As can be seen from Fig. 12, the variations of lift and drag coefficients do not have simple frequency as it is for single tube. For flow around a single tube, the flow is periodic and structured, whereas for flow around a tube bundle, due to the effects of tubes on each other, the flow is irregular and non-periodic. Due to the flow structure and the way that vortices are formed and separated, the oscillation frequency of drag coefficient is higher than that of the lift coefficient. The main part of the force exerted on the tubes are due to the pressures. This is also indicated by Yasin *et al.*^[5]. Finally the lift and drag coefficients are different for the first three rows, but are similar for the forth and fifth rows.

To examine more precisely, the Power Spectral Density (PSD) for CD and CL as functions of frequency on the fourth tube row are shown in Fig. 13 and 14. In these figures, the experimental results obtained by Ziada and Ongoeren^[23] are also presented. Relatively good results are obtained as it is shown. The maximum frequency is about 0.4. This frequency corresponds to the frequency of separation of vortices from the back of the tubes at the fourth row. The calculated frequency, 0.4, is the same as the value obtained by Yassin *et al.*^[5] and is closed to the measured frequency, 0.37^[24]. Also the simulation results agree well with the experimental data in capturing the vortex shedding peaks.

Flow patterns: The streamlines around the tube bundle at different times are given in Fig. 17. At these times, the drag and lift coefficients are shown in Fig. 15 and 16. As is shown, the flow patterns around the first row is very different from the other rows and as we move toward the last row, the flow patterns around each row become more similar to its proceeding row. Vortex generation and separation from the back of each row can also be seen from these figures. From Fig. 15 to 17, it can be seen that;

- 1 For all tubes, the frequency of CD is about twice the frequency of CL. The reason is that for each cycle of CL, two vortices are generated from the back of the tubes, one from the top and another from the bellow.

Then there are two oscillations for CD and one for CL. This phenomenon is more visible for a single tube than a tube bundle, because the tube rows affect each other.

- 2 The average of CL for all tube rows are zero while it is positive for CD.
- 3 Except for the first row, CD becomes negative at some portion of time in each period, but its average is positive. This is due to the existence of vortices being generated in front of each row due to the wake of the previous row. For the second to fifth rows large reduction of CD at 0.3 sec occurs as shown in Fig. 16. At this time, rather large vortices exist between the rows and the produce large pressure difference (see eleventh picture of Fig. 17).
- 4 As shown in Fig. 17, due to the high fluctuation of the turbulent flow, wide range of small and large scale vortices exist in the flow.
- 5 Except for the first row, CD and CL, have rather similar behavior for different rows. It is more obvious for the third and fourth rows. These results are also confirmed by experimental data presented by other researchers^[6,7].
- 6 The processes of generation and dissipation of vortices do not have a specific frequency. They behave randomly as it is shown in Fig. 12.

CONCLUSIONS

To simulated turbulent flows around tube bundles, the fully coupled Navier-Stokes equations in a non-orthogonal coordinate system are solved using parallelized method. The parallelized procedure is based on domain decomposition method and the use of the (MPI) standard. The large eddy simulation technique is used to model the turbulence structure of the flow. The computer code developed for this problem is used to simulated turbulent flows around tube bundle. The results were compared with corresponding experimental data and very good results are obtained. The efficiency and effectiveness of the parallelization code were examined and it was shown that the parallelization efficiency is very high. The flow patterns and its structure as well as the behavior of CL and CD were studied for different rows and compared with each other.

ACKNOWLEDGMENT

This study was financially supported by AEOI, Isfahan, Iran. We would like hereby to appreciate the support of this organization.

REFERENCES

1. Thompson B.E., 1985. Characteristic of a trailing edge flow with turbulent boundary layer separation. *J. Fluid Mechanics*, 157: 305-326.
2. Davidson, L., 1997, Implementation of a Large-Eddy Simulation Method Application to Recirculating Flow in a Ventilated Room, Gothenburg, Sweden.
3. Freddie, J., Davis Jr. and Y.A. Hassan, 1994. A two-dimensional finite element method large eddy simulation for application to turbulent steam generator flow. *J. Nuclear Technol.*, 106: 83-99.
4. Barsamian, H.R. and Y.H. Hassan, 1997. Large eddy simulation of turbulent cross flow in tube bundles. *Nuclear Engineering and Design*, 172: 103-122.
5. Yassin, A. Hassan and W.A. Ibrahim, 1997. Turbulent prediction in two-dimensional bundle flows using large-eddy simulation. *J. Nucl. Technol.*, 119: 11-28.
6. Chen, S.S. and J.A. Jendrzeczyk, 1987. Fluid excitation forces acting on a square tube array. *J. Fluid Engineering*, 109: 415-423.
7. Weaver, D.S. and A. Abd-Rabbo, 1985. A flow visualization study of a square array of tubes in water cross flow. *Trans. ASME.*, 107: 354-363.
8. Beale, S.B. and D.B. Spalding, 1999. A numerical study of unsteady fluid flow in in-line and staggered tube bank. *J. Fluid and Structures*, 13: 723-754.
9. Bui, T.T., 1999. A parallel, finite-volume algorithm for large-eddy simulation of turbulent flows. *J. Computers and Fluids*, 29: 877-915.
10. Karlo, V. and T. Tezduyar, 1997. Parallel Three Dimensional Computation of Unsteady Flows Around Circular Cylinders. Department of Aerospace Engineering and Mechanics Army High Performance, 1100 Washington Avenue South Minneapolis, MN 55451.
11. Strietzel, M., 1996. Parallel Turbulence Simulation Based on MPI, A Report from University of Koln, Report No., 96-236.
12. TASCflow, Theory Documentation, Version 2.4, March 1995.
13. Gropp, W.E. Lusk, MPICH Web Page, <http://www.mcs.anl.gov/gov/MPI/MPICH>.
14. Leonard, A., 1974. Energy cascade in large-eddy simulations of turbulent fluids. *Adv. Geophys.*, A18: 237.
15. Smagorinsky, J., 1963. General circulation experiments with the primitive Eq. 1. The basic experiment. *Mon. Weather Rev.*, 91: 99-165.
16. Piomelli, U., 1993. High reynolds number calculations using the dynamic subgrid scale stress model. *Phys. Fluid*, A5: 1484-1490.
17. Van Driest, E.R., 1956. On the turbulent flow near wall. *J. Aero. Sci.*, 23: 1007-1011.
18. Lee, S.Y., 1992. A Study and improvement of Large-Eddy Simulation (LES) for practical application Ph.D Thesis, Texas A&M University.
19. Ferziger, G.H. and M. Peric, 1999. *Computational Method for Fluid Dynamics*. 2nd Edn., Springer, pp: 182.
20. Peric, M., M. Schafer and E. Schreck, 1992. Numerical simulation of complex fluid flow on MIMD computers. *Parallel Computational Fluid Dynamics*, 92: 311-324.
21. Demirozic, I., Z. Lilek and M. Peric, 1992. Fluid flow and heat transfer test problems for non-orthogonal grid: Bench-Mark solutions. *Intl. J. Numerical Methods in Fluid*, 15: 329-354.
22. SPARSPAK: The Waterloo Sparse Matrix Package, <http://sparse.uwaterloo.ca/~jageore/sparspak.html> Scientific Computation Group, University of Waterloo, Ontario, Canada.
23. Oengoeren, A. and S. Ziada, 1992. Unsteady fluid forces acting on a square tube bundle in air cross-flow. *Proc. Sympo., Flow Induced Vibration and Noise*, 230: 55-74.
24. Taylor, C., M. Pettigrew, F. Axisa and B. Villard, 1988. Experimental determination of single and two phase cross flow induced forces on tube rows. *Flow Induced Vibration*, 104: 31.