41st Aerospace Sciences Meeting and Exhibit
6-9 January 2003, Reno, Nevada

AIAA 2003-84

Two Level Simulation of High-Reynolds Number Non-Homogeneous Turbulent Flows

K. Kemenov*and S. Menon†
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332

A new two-level simulation (TLS) has been developed based on the decomposition of velocity into resolved and small-scale components. A coupled system of the resolved and small-scale equation that is not based on an eddy-viscosity type of assumption has been derived and implemented to simulate very high-Re wall-bounded turbulent flows as in channel. Results suggest that the baseline TLS model which requires no adjustable parameters has the potential for capturing turbulent flow behavior in high Re channel flows using very coarse grids.

Introduction

Simulation of very high Reynolds (Re) number wall-bounded flows is computationally very expensive because the near wall region has to be properly resolved in order to achieve accurate prediction. Past affordable studies using direct numerical simulations (DNS) have been limited to relatively low Re (500-2000 based on wall units).[1] However, the typical Re associated with wall-bounded flows of practical interest can be an order (or more) of magnitude higher. DNS is obviously impossible for such flows even with the projected speedup of the next generation computers and large-eddy simulations (LES) may be the only viable approach to study these flows.

The present effort is focused on validating of new approach for LES of high Re flows that departs significantly from conventional methods used in LES. The present approach is similar to several alternative approaches (referred loosely here as “decomposition” approaches) that have emerged in literature recently. In contrast to LES, where decomposition of turbulent velocity into two components, resolved and small-scale (unresolved), is introduced through spatial filtering and the major effort is concentrated on SGS modelling, in ”decomposition” approaches considerable attention is devoted to modeling of small-scale velocity itself. This usually involves a derivation of governing equation for small-scale velocity with its subsequent simplification based on some physical arguments.[2–8]

Regardless of how scales are decomposed it is apparent that if both the resolved and small scales are simulated in three dimensions then this approach is no different than a DNS, and therefore, unviable. In order to implement this approach within the context of LES, we have developed a new framework denoted TLS in which the resolved scale motion are simulated using ’large-scale’ equations that is forced by the small-scale, unresolved motion. In TLS approach, the small-scale velocity field is explicitly reconstructed by solving an appropriate small-scale equation on a 3D family of 1D grid lines embedded inside the 3D resolved grid. Constructed in such a way the 3D small-scale velocity field serves as a closure for the 3D resolved scale equations. The reduction in dimensionality for the small-scale equations allows the coupled TLS approach to be computationally feasible (and efficient on massively parallel machines) and applicable to high-Re flow simulations.

In this paper, the mathematical formulation of the TLS approach is first highlighted and used to simulate three-dimensional channel flow.

*Graduate Research Assistant
†Professor, AIAA Associate Fellow

Formulation

Two-scale decomposition

We consider the incompressible Navier-Stokes equations:

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

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

and split the velocity field into resolved and small-scale components into resolved and small-scale components:

$$u_i(\bar{x}, t) = u_i^r(\bar{x}, t) + u_i^s(\bar{x}, t). \tag{3}$$

The meaning of the resolved velocity $u_i^r(\bar{x}, t)$ is quite general – it can represent not only the filtered quantity with respect to some spatial filter but also any resolved quantity known on the given resolved grid such as volume averaged velocity, as in finite volume formulation. The resolved quantity known on the “large” scale grid is denoted by $\Big[\cdot\Big]_r$ or superscript r. The resolved scale equation has following form:

$$\frac{\partial u_i^r}{\partial t} + \Big[\frac{\partial}{\partial x_j}(u_i^r + u_i^s)(u_j^r + u_j^s)\Big]_r = -\frac{\partial p^r}{\partial x_i} + \nu\frac{\partial^2 u_i^r}{\partial x_j^2} \quad (4)$$

$$\frac{\partial u_i^r}{\partial x_i} = 0 \quad (5)$$

If the small-scale velocity is known the resolved scale can be found by integrating Eq. (4). The small-scale equations are obtained by subtracting the resolved equations from the original Navier-Stokes equations. This gives:

$$\frac{\partial u_i^s}{\partial t} + \frac{\partial}{\partial x_j}(u_i^r + u_i^s)(u_j^r + u_j^s) = -\frac{\partial p^s}{\partial x_i} + +\nu\frac{\partial^2 u_i^s}{\partial x_j^2} \quad (6)$$

$$-\Big[\frac{\partial}{\partial x_j}(u_i^r + u_i^s)(u_j^r + u_j^s)\Big]_r$$

$$\frac{\partial u_i^s}{\partial x_i} = 0$$

A numerical simulation of the above small-scale equations is quite challenging and will require computational effort similar to that for a DNS. Therefore, to model the small-scale turbulent velocity field efficiently and with a high physical fidelity, the small-scale equations must be simplified while retaining the underlying picture of scale interaction. This is described in the next section.

TLS small-scale model

As mentioned earlier, the full two scale simulation model is not viable due to DNS comparable resource requirement. In the light of this, we propose a new simplified model for reconstructing the small-scale velocity field. In some respect this is consistent with the framework of the One-Dimensional Turbulence (ODT) approach originally developed by Kerstein and coworkers[9, 10]

In ODT approach, turbulent velocity and other properties are simulated along one-dimensional line of sight through a 3D turbulent flow. The reduction to 1D domain makes the model computationally efficient. However, a correct formulation and implementation of ODT model, as a closure for subgrid scale fluxes, is not straightforward and is still an area of active research.[11] The baseline ODT model has a disadvantage since it is not able to take into account the effects of resolved velocity on the modeled small-scale velocity field, thus somewhat hindering a forward energy cascade picture. Here, the proposed model automatically addresses this issue.

To make the small-scale equation more suitable for an efficient numerical simulations it is natural to introduce two time coordinates in velocity decomposition equation such that:

$$u_i(\bar{x}, t) = u_i(\bar{x}, t^r; t^s) = u_i^r(\bar{x}, t^r) + u_i^s(\bar{x}, t^r; t^s). \quad (7)$$

Here, we assumed that the resolved velocity does not depend on the small-scale time coordinate t^s thus it is treated as a "slow" variable, i.e., it is set to be a constant when it is viewed at the small-scale time scale. Thus the time derivative in Eq. (6) is assumed to be with respect t^s (with a little abuse of notation) and all resolved quantities depend on spatial coordinates only. Furthermore, to model the small-scale turbulent velocity field in a 3D domain Ω, we consider a family of 1D lines arranged as 3D lattice embedded in Ω. The family consists of three types of lines $\{l_1, l_2, l_3\}$ orthogonal each other and parallel to corresponding Cartesian coordinates x_i. The lines of each type intersects each other at a center of a cell which is defined by the resolved grid in the domain Ω. The line arrangement is shown in Fig. 1.

We model the 3D small-scale velocity field as a family of 1D small-scale velocity vector fields defined on the underlying family of lines $\{l_1, l_2, l_3\}$

$$u_i^s(\bar{x}, t^r; t^s) \longrightarrow u_{i,l_j}^s(l_j, t^s), \quad \bar{x} \in \Omega, \quad l_j \in R^1$$

The small-scale velocity field $u_{i,l_j}^s(l_j, t^r; t^s)$ can be viewed as a snapshot of the small-scale 3D turbulent field along the line $\{l_j\}$ somehow oriented in computational domain Ω (in principle, the orientation of this 1D line can be arbitrary but in the present effort is along the resolved grid lines).

The small-scale velocity fields evolve according to simplified 1D governing equations that can be obtained from the basic small-scale Eq. (6) by utilizing assumptions made in the definition of TLS model. Namely, since the velocities are defined on 1D lines only, all small-scale velocity and pressure derivatives with respect to directions orthogonal to the line remain unknown and have to be modelled. In the present effort they assumed to be equal to corresponding derivatives along the line. Though more sophisticated stochastic models of the cross-line small-scale derivatives can be introduced under this TLS approach. Thus, for example:

$$\frac{\partial u_i^s}{\partial l_1} \approx \frac{\partial u_i^s}{\partial l_2} \approx \frac{\partial u_i^s}{\partial l_3}$$

This gives a following system of 1D equations for small-scale velocities defined on the family of lines $\{l_1^{k_1}, l_2^{k_2}, l_3^{k_3}; k_j = 1, \ldots, N_j; j = 1, 2, 3\}$, where N_j is the number of lines of jth type,

$$\frac{\partial u_{ij}^s}{\partial t} + NL(u_{ij}^s, u_{ij}^r, l_j) = 3\nu\frac{\partial^2 u_{ij}^s}{\partial l_j^2} - \frac{\partial p^s}{\partial l_j} + \Big[NL(u_{ij}^s, u_{ij}^r, l_j)\Big]_r \quad (8)$$

$$\frac{\partial}{\partial l_j}(u_{1j}^s + u_{2j}^s + u_{3j}^s) = 0$$

Here, the following notation is used: $u^s_{ij}(l_j, t^s) \equiv u^s_{i,l_j}(l_j, t^s)$ represents the i-th component of the small-scale velocity belonging to j-th type of line, the upper index corresponding to the line number in $l^{k_j}_j$ is also dropped for notation simplicity and interaction terms are defined on the j-th line. The small and resolved scales are coupled through a non-linear convective term $NL(u^s_{ij}, u^r_{ij}, l_j)$ which subsumes all terms of the small-scale equation except diffusion, pressure gradient and non-stationary terms.

For example, the TLS small-scale equations along l_1 lines (which are parallel to x-coordinate of the resolved grid) gives the following equations (denoting for simplicity, $u^s_{i1}(l_1, t) \equiv (u^s, v^s, w^s)$) for u^s- component:

$$\frac{\partial u^s}{\partial t} + \frac{\partial}{\partial l_1}\Big[uu + u^s(v+w) + u^r(v^s + w^s)\Big] + \quad (9)$$

$$u^s\Big[\frac{\partial v^r}{\partial y} - \frac{\partial v^r}{\partial x} + \frac{\partial w^r}{\partial z} - \frac{\partial w^r}{\partial x}\Big] + v^s\Big[\frac{\partial u^r}{\partial y} - \frac{\partial u^r}{\partial x}\Big] +$$

$$w^s\Big[\frac{\partial u^r}{\partial z} - \frac{\partial u^r}{\partial x}\Big] + \frac{\partial u^r v^r}{\partial y} + \frac{\partial u^r w^r}{\partial z} = 3\nu\frac{\partial^2 u^s}{\partial l_1^2} - \frac{\partial p^s}{\partial l_1} + NL_r$$

for v^s-component:

$$\frac{\partial v^s}{\partial t} + \frac{\partial}{\partial l_1}\Big[uv + v^s(v+w) + v^r(v^s + w^s)\Big] + \quad (10)$$

$$v^s\Big[2\frac{\partial v^r}{\partial y} - 2\frac{\partial v^r}{\partial x} + \frac{\partial w^r}{\partial z} - \frac{\partial w^r}{\partial x}\Big] +$$

$$w^s\Big[\frac{\partial v^r}{\partial z} - \frac{\partial v^r}{\partial x}\Big] + \frac{\partial v^r v^r}{\partial y} + \frac{\partial v^r w^r}{\partial z} = 3\nu\frac{\partial^2 v^s}{\partial l_1^2} - \frac{\partial p^s}{\partial l_1} + NL_r$$

and for w^s-component:

$$\frac{\partial w^s}{\partial t} + \frac{\partial}{\partial l_1}\Big[uw + w^s(v+w) + w^r(v^s + w^s)\Big] + \quad (11)$$

$$w^s\Big[2\frac{\partial w^r}{\partial z} - 2\frac{\partial w^r}{\partial x} + \frac{\partial v^r}{\partial y} - \frac{\partial v^r}{\partial x}\Big] +$$

$$v^s\Big[\frac{\partial w^r}{\partial y} - \frac{\partial w^r}{\partial x}\Big] + \frac{\partial w^r w^r}{\partial z} + \frac{\partial v^r w^r}{\partial y} = 3\nu\frac{\partial^2 w^s}{\partial l_1^2} - \frac{\partial p^s}{\partial l_1} + NL_r$$

The resolved velocity is assumed to be known on each line so the small-scale velocity field can be reconstructed on lines by solving these equations. Note that, on different line the system will generate different small-scale fields since the resolved velocity is changing from line to line.

Numerical Implementation

Numerical algorithm

We apply our TLS approach to simulate 3D turbulent channel flow. The computational domain is parallelepiped $\Omega = 2\pi \times 2 \times \frac{2}{3}\pi$ which discretized by a very coarse resolved grid with a little stretching in wall normal directions. The numerical algorithm is

- At time $t = t^r$, having known resolved velocity $u^r_i(\bar{x}, t^r)$ on the coarse grid (obtained using a conventional staggered grid finite-volume scheme[12]), interpolate it on each 1D line $\{l_j\}$

$$u^r_i(\bar{x}, t^r) \longrightarrow u^r_{ij}(l_j)$$

- Solve the 1D small-scale system (8) on each line with corresponding boundary condition to find $u^s_{ij}(l_j, t^s)$.
- Compute the small-scale velocity on the resolved grid by averaging $u^s_{ij}(l_j, t^s)$ over the three lines intersecting at the resolved grid point

$$\Big[u^s_i(\bar{x}, t^r; t^s)\Big]_r \longleftarrow u^s_{ij}(l_j, t^s)$$

- Advance the resolved velocity $u^r_i(\bar{x}, t^r)$ at time $t^r + \Delta t^r$ by solving the resolved equations on the resolved grid.

In the above algorithm, the time scale t^r represents the time-scale on which all the resolved variables evolves. The time step Δt^r is therefore, the time step for the resolved-scale evolution. The small-scale field evolves on an faster time scale t^s within every large-scale time step. Thus, $N_s = \Delta t^r / \Delta t^s$ is the total number time steps the small-scale field evolves between large-scale events.

For each type of line, the small-scale equation can be written by equations in vector form:

$$\frac{\partial U^s_i}{\partial t} + \frac{\partial}{\partial l_j}F_j(U^s_i, U^r_i) + D_j(U^s_i, U^r_i) + R_j(U^r_i) = \quad (12)$$

$$= 3\nu\frac{\partial^2 U^s_i}{\partial l_j^2} - \frac{\partial P^s}{\partial l_j} + \Big[NL_j(U^s_i, U^r_i)\Big]_r,$$

where

$$U^s_i = (u^s_i, v^s_i, w^s_i)^\top,$$

and, each vector-valued function $F_j(U^s_i, U^r_i)$, $D_j(U^s_i, U^r_i)$ and $R_j(U^r_i)$ depends on both the small-scale and the resolved-scale velocities, and take a particular form on each line l_j. The second function D_j represents a product of the small-scale velocities with corresponding derivatives of the resolved velocity U^r_i and R_j, both depending only on the derivatives of the resolved velocity.

The small-scale velocity field can have a very intermittent, high-gradient velocity profile especially in flow regions with high turbulent intensity. Therefore, it is important to use a numerical scheme which does not introduce the spurious oscillations in the presence high-gradients in the field. Also, for a given fixed resolved velocity field one may expect that the solution of the small-scale equation should take more or less stationary form ensuring a uniqueness of the small-scale field for a given resolved field. Therefore, a suitable numerical scheme should be able to recognize stationary solutions. A two-step, component-wise TVD scheme[13] has been used to solve the small-scale equations. This TVD scheme does not use the characteristic decomposition used in usual TVD schemes and this makes it particularly easy to implement. A local Lax-Freidrichs flux splitting has been used for the scheme upwinding procedure. The set of eigenvalues of the Jacobian matrix of the flux function $F_j(U_i^s, U_i^r)$ has been found independent on orientation of line l_j. The matrix is hyperbolic, and has three eigenvalues: $u+v+w$, $u+v+w$ and $2u+2v+2w$. Here, the largest one has been used for the flux splitting procedure.

Small-scale pressure is handled as in a standard fractional step method where it plays the role of enforcing small-scale continuity. At the first step, the small-scale equations are integrated without the pressure gradient term, then the small-scale field is corrected in such a way that the continuity is enforced. Under the assumptions adopted in TLS small-scale model, the pressure equation collapses into a second-order non-homogeneous equation which can be easily integrated analytically. As can be seen from the continuity equation on the line, such a role of pressure is equivalent to having a following constraint on the small-scale velocity field:

$$u^s + v^s + w^s = C(l_j)$$

where $C(l_j)$ is a line-dependent constant which can be found by using an appropriate boundary condition.

Note that, since the 1D lines extend through the entire computational domain and terminate only at the physical or computational boundaries, physically consistent boundary conditions becomes particularly easy to implement. For example, at the wall no-slip condition can be enforced since the small-scale resolution is considered sufficient to resolve the near-wall region as in a DNS. Thus, the 3-component velocity solution evolving on the matrix of 1D lines employ the same forms of the governing equations but differ primarily in the specification of the boundary conditions.

Coupling issues

Coupling between scales is achieved by averaging the small-scale small-scale velocity field ${u_{ij}}^s(l_j)$ on the three orthogonal lines at an intersection point, which in the current case, is the cell center of the finite-volume resolved-scale grid. This averaging can be both in space within the large cell, as well as in time of evolution (note, small-scale time $\Delta t^s < \Delta t^r$. Such an averaged value of the small-scale velocity is treated as the resolved part of the small-scale velocity that is evolving at the slow time t^r on the large-scale grid. Coupling is completed by modifying the resolved velocity field by this resolved part of the small-scale velocity at each resolved cell in computational domain according to the resolved Eq. (4)

Parallel performance and efficiency

A critical part of TLS calculation is reconstruction of small-scale velocity field on lines. Thus, small-scale equations of a type, Eq. (8) have to be solved on each line. Master-slave algorithmic model provides a relatively straightforward way to parallelize small-scale velocity calculation. To compute small-scale velocity field the master process sends all relevant "line" information to each slave process which then performs the numerical integration of Eq. (8). The master process keeps sending new lines to slaves until no lines are left, at the same time, the master receives the computed small-scale line velocity profiles from slaves. Integration of the small-scale equations may take different times since the 1D resolution in each direction can be independently specified (as in the large-scale grid) and therefore, the time step for integration can be different. Time step can be restricted when there are large and rapid variation in velocity field as well. For example, due to high axial velocity fluctuations in the wall normal directions, a finer grid and a small time-step is necessary to resolve the flow features. Master-slave model, due to its self-scheduling feature provides a necessary load balancing for slave processes.

The parallel algorithm have been tested on several high performance hardware platforms including IBM SP3 "Habu" at NAVO MSRC. The IBM SP is a massively parallel scalable machine which contains 334 SMP nodes with 4 CPUs and 4 GB of memory per node. The parallel model demonstrated a good scalability (80%) for up to 70-80 processors with a superlinear speedup for a smaller number of processors.

Results and Discussion

3D TLS of high-Re channel flows have been conducted using a staggered grid technique.[12] A third-order, low storage Runge-Kutta scheme is employed for temporal discretization. This scheme is second-order accurate and the original solver employs the standard Germano's dynamics algebraic subgrid model. For the new TLS application, the SGS model is removed and replaced by the terms described in the TLS formulation. The small-scale field is also simulated using the TLS 1D model described earlier. No special treatment of the near-wall cell is used in the present effort since the primary goal of this study is to evaluate the baseline TLS approach.

Simulations for a Re_τ=590 (based on wall units) channel flow is performed in this initial effort since DNS data[1] is available for comparison. A resolution of 32x32x32 is employed for the resolved field with no stretching in the streamwise and spanwise direction and only a nominal stretching in the wall normal direction. This resolution is considered very coarse even for a LES (the equivalent DNS resolution has been

$384 \times 257 \times 384$), however, this grid is chosen in order to challenge the ability of the TLS approach to deal with high-Re flows using very coarse grid. With this coarse grid, the minimum wall normal spacing is $\Delta y^+ = 30$ and therefore, the near-wall turbulent field is not expected to be captured directly in the resolved field. The burden for this resolution is on the small-scale field.

The 1D lines, as noted above, can employ variable grid in each co-ordinate direction. Here, we employ a uniform grid of 8 1D cells per LES cell in all periodic directions and a variable grid in the wall normal direction with a number of 1D cells varying from 12 near the wall to 4 in the centerline region. The effect of varying the 1D grid resolution in each direction has not yet been studied in detail but will be revisited in the near future. For this resolution, a complete simulation of the channel flow requires around 380 single-processor hours on IBM SP3. We plan to continue to improve the computational speed since many elements of the current code can be improved upon.

Typical profiles of the streamwise small-scale and resolved velocities for two different wall normal lines are shown in Figs. 2 and 3. It can be seen that fluctuations of the small-scale velocity is of high intensity in the near wall region, as expected.

The mean profile of the streamwise velocity $u^r(y^+)$ along the wall-normal coordinate is shown in Fig. 4. Note that, since the first resolved cell location is at $y^+ = 30$, all data below this location is given entirely by the "average" (over the small-scale time) solution from the small-scale field. In general, the TLS profile is very similar to the DNS results especially in the near wall viscous sublayer region. However, away from the wall, the TLS profile shows a little higher resolved velocity magnitude. A possible reason for this can be attributed to insufficient (i.e., coarse) resolution of the resolved grid chosen for this test.

The rms velocity profiles are plotted in wall units in Fig. 5. Streamwise velocity rms demonstrates the correct location of its maximum although in the near-wall region, it is a little overestimated. However, since the peak occurs inside the small-scale field, the reasonable accuracy in predicting the peak location and its magnitude by this TLS approach suggests that this approach has some physically consistent features.

Conclusions

A new TLS approach alternative to LES has been developed based on the decomposition of velocity into resolved and small-scale components. A coupled system of the resolved and small-scale equation that is not based on an eddy-viscosity type of assumption has been derived and implemented to simulate very high-Re wall-bounded turbulent flows as in a channel. Results suggest that the baseline TLS model which requires no adjustable parameters has the potential for capturing turbulent flow behavior in high Re channel flows using very coarse grids (e.g., the resolved grid used for TLS is only 32x32x32 for a Re_τ=590 (whereas, a 384x257x384 grid resolution is needed for an equivalent DNS).

Further study is still needed to understand all the nuances of the TLS approach. Issues related to optimal resolved grid resolution for a given Re needs to be addressed in order to minimize the computational cost. Additional simulations of much higher Re channel flows, high-Re flow past bumps, and high-Re isotropic turbulence are currently being implemented. Finally, the methodology is independent of the flow speed and therefore, extension to compressible flow is also being pursued.

Acknowledgement

This work is supported by Office of Naval Research. Computational time has been provided at DOD High Performance Computing Centers at Maui, HI and NAVOCEANO, AL.

References

[1] Moser, R., Kim, J., and Mansour, N., "Direct numerical simulation of turbulent channel flow up to $Re_\tau = 590$," *Physics of Fluids*, Vol. 11, 1999, pp. 943–945.

[2] Laval, J., Dubrule, B., and Nazarenko, S., "Dynamical modeling of sub-grid scales in 2D turbulence," *Physica D*, Vol. 142, 2000, pp. 231–253.

[3] Laval, J., Dubrule, B., and Nazarenko, S., "Nonlocality and intermittency in three-dimensional turbulence," *Physics of Fluids*, Vol. 13, 2001, pp. 1995–2012.

[4] Dubois, T., Jauberteau, F., and Temam, R., *Dynamic multilevel methods and the numerical simulation of turbulence*, Cambridge University Press, 1999.

[5] Hughes, T., Feijo, G., L.Mazzei, L., and Quincy, J.-B., "The variational multiscale method- a paradigm for computational mechanics," *Computer Methods in Applied Mechanics and Engineering*, Vol. 166, 1998, pp. 3–24.

[6] Hughes, T., Mazzei, L., and Jansen, K., "The Large Eddy Simulation and variational multiscale method," *Comput. Vis. Sci.*, Vol. 3, 2000, pp. 47–59.

[7] Hylin, E. and McDonough, J., "Chaotic small-scale velocity fields as prospective models for unresolved turbulence in an additive decomposition of the Navier-Stokes equation," *International Journal of Fluid Mechanics Research*, Vol. 26, 1999, pp. 539–567.

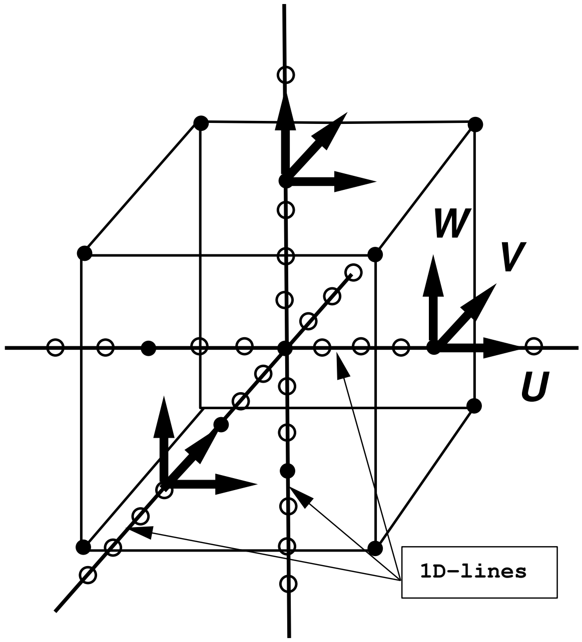

Fig. 1 The 1D line arrangement in TLS Model.

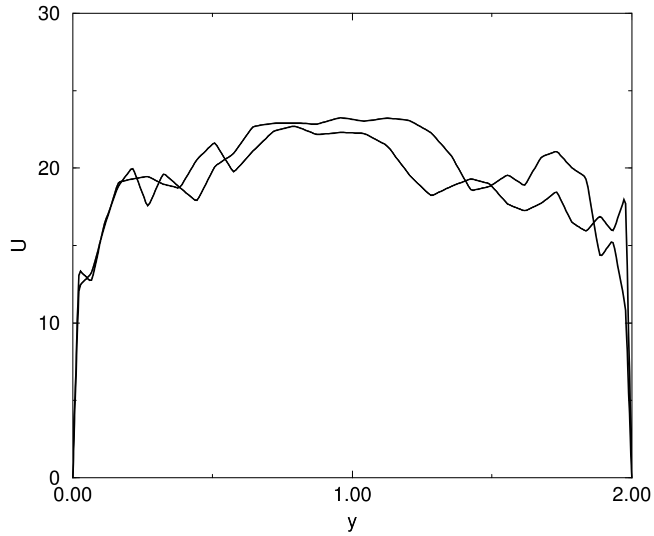

Fig. 2 The snapshot of the resolved streamwise velocity on two different wall-normal lines.

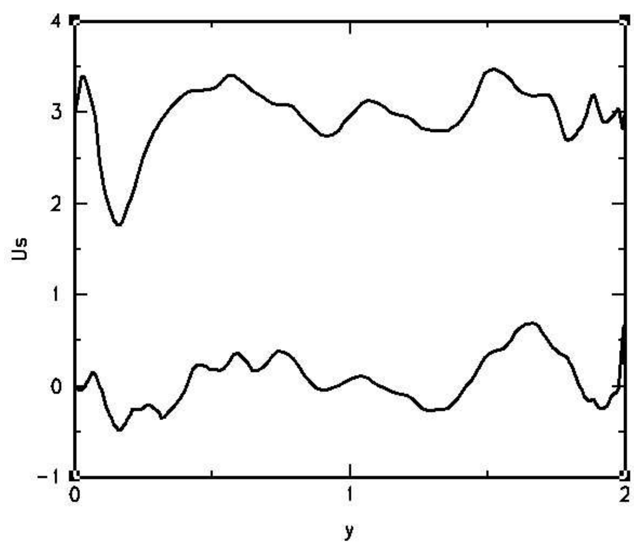

Fig. 3 The snapshot of the small-scale streamwise velocity on two different wall-normal lines. Note that one small-scale profile is shifted vertically from zero line for visualization.

[8] Domaradzki, J. and Saiki, E., "A subgrid-scale model based on the estimation of unresolved scales of turbulence," *Physics of Fluids*, Vol. 9, 1997, pp. 2148–2164.

[9] Kerstein, A., "One-dimensional turbulence: model formulation and application to homogeneous turbulence, shear flows and buyoant stratified flows," *Journal of Fluid Mechanics*, Vol. 392, 1999, pp. 277–334.

[10] Kerstein, A., Ashurst, W., Wunsch, S., and Nilsen, V., "One-dimensional turbulence: vector formulation and application to free shear flows," *Journal of Fluid Mechanics*, Vol. 447, 2001, pp. 85–109.

[11] Schmidt, R., Kerstein, A., Wunsch, S., and Nilsen, V., "Near-wall LES cLosure based on One-Dimensional Turbulence modeling," *Journal of Computational Physics*, 2000, pp. submitted.

[12] van Kan, J., "A second-order accurate pressure-correction scheme for viscous incompressible flow," *SIAM Journal of on Scientific and Statistical Computing*, Vol. 7, No. 3, 1986, pp. 870–891.

[13] Yu, H. and Liu, Y., "A Second Order Accurate, Component-Wise TVD Scheme for Nonlinear, Hyperbolic Conservation Laws," *Journal of Computational Physics*, Vol. 173, 2001, pp. 1–16.

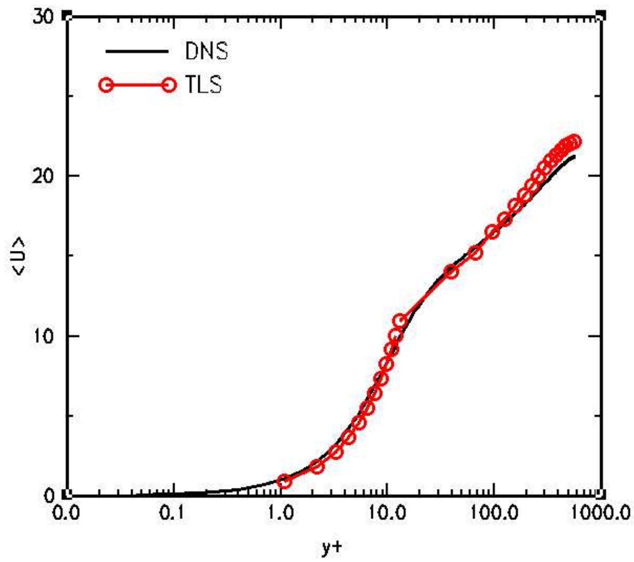

Fig. 4 The mean total streamwise velocity profile in wall units.

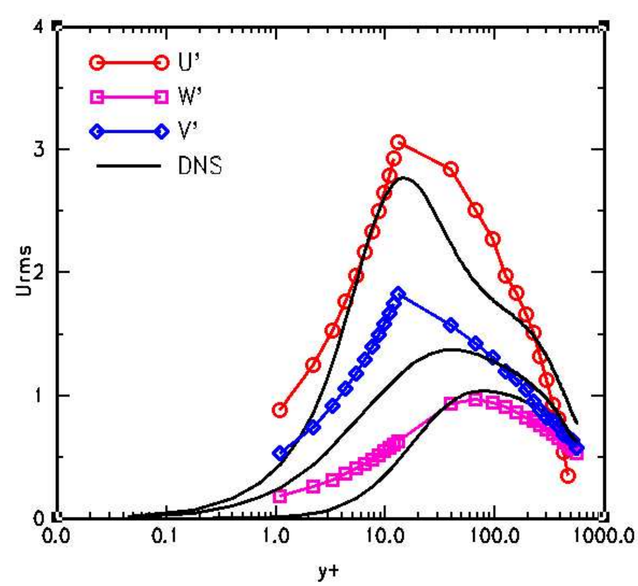

Fig. 5 Rms velocity profiles in wall units.