National Renewable Energy Laboratory
A national laboratory of the U.S. Department of Energy
Office of Energy Efficiency & Renewable Energy
Innovation for Our Energy Future

Wind Energy-Related Atmospheric Boundary Layer Large-Eddy Simulation Using OpenFOAM

Preprint

M.J. Churchfield and P.J. Moriarty
National Renewable Energy Laboratory

G. Vijayakumar and J.G. Brasseur
The Pennsylvania State University

Presented at 19th Symposium on Boundary Layers and Turbulence
Keystone, Colorado
August 2-6, 2010

Conference Paper
NREL/CP-500-48905
August 2010

NREL is operated for DOE by the Alliance for Sustainable Energy, LLC Contract No. DE-AC36-08-GO28308

Available electronically at http://www.osti.gov/bridge

Available for a processing fee to U.S. Department of Energy and its contractors, in paper, from:

U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831-0062
phone: 865.576.8401
fax: 865.576.5728
email: mailto:reports@adonis.osti.gov

Available for sale to the public, in paper, from:

U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
phone: 800.553.6847
fax: 703.605.6900
email: orders@ntis.fedworld.gov
online ordering: http://www.ntis.gov/ordering.htm

Printed on paper containing at least 50% wastepaper, including 20% postconsumer waste

1B.6 WIND ENERGY-RELATED ATMOSPHERIC BOUNDARY LAYER LARGE-EDDY SIMULATION USING OpenFOAM

Matthew J. Churchfield*
National Wind Technology Center, National Renewable Energy Laboratory, Golden, Colorado

Ganesh Vijayakumar
Department of Mechanical and Nuclear Engineering, The Pennsylvania State University, University Park, Pennsylvania

James G. Brasseur
Department of Mechanical and Nuclear Engineering, The Pennsylvania State University, University Park, Pennsylvania

Patrick J. Moriarty
National Wind Technology Center, National Renewable Energy Laboratory, Golden, Colorado

1. INTRODUCTION

The purpose of the current work is to develop and evaluate the performance of a large-eddy simulation (LES) solver in computing the atmospheric boundary layer (ABL) over flat terrain under a variety of stability conditions, ranging from shear driven (neutral stratification) to moderately convective (unstable stratification). This colocated, unstructured, finite-volume solver is developed using the OpenFOAM framework. Although, to our knowledge, OpenFOAM has never been used to perform LESs, its flexible, open source, and parallel nature combined with its availability at no cost, make it an ideal choice for this application. Data from our ABL simulations, like those presented in this work, will be used in our future work as precursors for LESs of flow through wind farms consisting of multi-megawatt wind turbines.

Over the past four decades, the atmospheric community has done much work using LES (in many cases with a pseudo-spectral solver) to accurately simulate the ABL. A few examples include the work of Deardorff (1972), Moeng and Sullivan (1994), Mason and Brown (1999), Khanna and Brasseur (1998), and Cuijpers and Duynkerke (1993). An objective of our work is to use that knowledge to create an OpenFOAM-based solver that can perform simulations with the same accuracy as that achieved by the atmospheric community. To assess our solvers' performance, we mainly compared our results to the LESs performed by Moeng and Sullivan (1994) because their LES code, which is outlined by Moeng (1984), is well proven, their simulations are of a smaller grid size that is ideal for this initial validation study, and the simulations are all windy while ranging from neutral to unstable, which is of interest in wind-farm modeling. We also compare energy spectra and isosurfaces depicting ABL structure to the LESs of Khanna and Brasseur (1998), which were also computed using the algorithm of Moeng (1984). In developing and testing our LES code, we will make use of a recent analysis (Brasseur and Wei 2010) that explains the well-known inability for LES to properly predict law-of-the-wall scaling and provides a design methodology to avoid this pathology.

Only a few researchers have performed LESs of wind farm flows. Examples include the work of Ivanell (2009), Calaf et al. (2010), and Stovall et al. (2010) (Stovall used an OpenFOAM-based solver), but all of these works are limited to neutrally-stratified conditions. The wind farm LES framework of Porté-Angel et al. (Porté-Agel et al. 2010; Conzemius et al. 2010) is clearly capable of simulation in a variety of stability conditions, though.

The eventual goal of our work is to perform wind-farm LESs that will provide a better understanding of both the interactions of wind turbine wakes with one another and with the surrounding ABL. Current utility-scale wind turbines have tower heights of 80 m to 100 m and rotor diameters of 70 m to 120 m. At these tower heights and with such large rotor diameters, utility-scale wind turbine blades can often cover the entire atmospheric surface layer, which contains the highest levels of shear in the mean wind. This means that turbine rotor blades experience substantial cyclic loading as they rotate. Additionally, each turbine in a wind farm creates a wake that can affect the performance and mechanical loads of the wind turbines downstream. Turbine wakes interact with each other and are significantly affected by the stability condition of the surrounding ABL. For example, according to Jensen (2007), power output data from the Horns Rev Offshore Wind Farm, which is an 8 by 10 array of 2-MW turbines in the North Sea 14 km off the Danish coast, shows that in a stable ABL, the farm's efficiency is 61%; in an unstable ABL, the farm's efficiency dramatically increases to 74%. This is because an un-

* *Corresponding author address:* Matthew J. Churchfield, National Renewable Energy Laboratory, 1617 Cole Boulevard, Golden, Colorado 80401; e-mail: matt.churchfield@nrel.gov.

stable ABL contains more turbulent kinetic energy that has a greater potential to replace the low-momentum air within a turbine's wake with higher-momentum surrounding air through turbulent mixing. Consequently, in the unstable condition, the downstream rows of turbines are fed higher-momentum air than in the stable condition, resulting in increased wind farm performance. However, according to Wharton and Lundquist (2010), in the absence of wakes, turbines perform more efficiently in stable conditions.

Many aerodynamic and atmospheric phenomena occur within large wind farms that are not well understood. This lack of understanding is reflected by the fact that wind farm developers consistently overpredict the performance of large modern wind farms by 5% or more (Johnson 2008). This lower performance can result in millions of dollars less revenue generated than expected over the farm's lifetime. Poor wake and atmospheric modeling is cited as one of the reasons for this overprediction. Developers currently use tools developed in the 1980s and early 1990s, like that of Katic et al. (1986) when wind turbines were much smaller, that are simpler and less computationally-expensive than computational fluid dynamics (CFD). A clearer understanding of the aerodynamics and atmospheric physics within large wind farms would allow for the creation of more accurate, lower-order wind farm planning and design tools. LES has the potential to provide an accurate, highly-resolved description of the flow field through a wind farm. However, much work remains to be done to establish a set of "best practices" for performing wind-farm-scale LES. This involves the ability to perform accurate LESs of canonical ABL flows, the subject of this paper.

2. FLOW PHYSICS EQUATIONS AND MODELING

2.1 *FILTERED GOVERNING EQUATIONS*

The filtered incompressible Navier-Stokes equations within a rotating frame of reference are used to perform these simulations. The momentum equation is

$$\frac{\partial \bar{u}_i}{\partial t} = -\frac{\partial}{\partial x_j}\left(\bar{u}_j \bar{u}_i\right) - \frac{\partial R_{ij}^D}{\partial x_j} - \frac{\partial \tilde{p}}{\partial x_i} - \left(\frac{\partial \tilde{p}}{\partial x_i}\right)^d + \left(1 - \frac{\bar{\theta}}{\bar{\theta}^0}\right) g_i + \epsilon_{ij3} f^c \bar{u}_j, \tag{1}$$

where, $\bar{u}_i$ is the resolved Cartesian velocity field, $R_{ij}^D = R_{ij} - R_{kk}\delta_{ij}/3$ is the deviatoric part of the sub-grid-scale (SGS) stress tensor and R_{ij} is the SGS stress tensor, $\tilde{p} = \bar{p}/\rho^0 + R_{kk}/3 - g_i h$ is the deviation of the resolved pressure from hydrostatic plus the contribution from the SGS energy all normalized by the density, ρ^0 is the density, g_i is the gravity vector, h is height above the surface, $(\partial\tilde{p}/\partial x_i)^d$ is a spatially constant driving pressure gradient term used to achieve a specified mean geostrophic wind, $\bar{\theta}$ is the resolved virtual potential temperature, $\bar{\theta}^0$ is the reference virtual potential temperature, ϵ_{ijk} is the alternating symbol, f^c is the Coriolis parameter, and the subscripts 1, 2, and 3 refer to the x-, y-, and z-directions, respectively. $\bar{\theta}^0$ is set to the initial virtual potential temperature below the capping inversion of 300 K, gravity is set to $g_i = (-9.81, 0, 0)$ m/s^2, and the Coriolis parameter is set to a typical mid-latitude value of 1.028×10^{-4} s^{-1}. The Boussinesq approximation for buoyancy is included through the fifth term on the right-hand side of Equation 1.

The filtered continuity equation is

$$\frac{\partial \bar{u}_j}{\partial x_j} = 0, \tag{2}$$

and the filtered transport equation for virtual potential temperature is

$$\frac{\partial \bar{\theta}}{\partial t} = -\frac{\partial}{\partial x_j}\left(\bar{u}_j \bar{\theta}\right) - \frac{\partial R_{\theta j}}{\partial x_j}, \tag{3}$$

where $R_{\theta j}$ is the SGS potential temperature flux vector.

2.2 *SUB-GRID-SCALE MODELING*

The deviatoric SGS stress tensor found in the momentum equation (Equation 1) is modeled using

$$R_{ij}^D = -\nu^{SGS} \bar{S}_{ij}, \tag{4}$$

where ν^{SGS} is the SGS viscosity, and

$$\bar{S}_{ij} = \frac{1}{2}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right) \tag{5}$$

is the resolved strain-rate tensor. The SGS temperature flux vector found in the temperature equation (Equation 3) is modeled using

$$R_{\theta j} = -\frac{\nu^{SGS}}{\mathrm{Pr}^t}\frac{\partial \bar{\theta}}{\partial x_j}, \tag{6}$$

where Pr^t is the turbulent Prandtl number. In this case, Pr^t = 1/3, although this should be modified in the future to follow the approach of Moeng (1984) in which Pr^t is generally 1/3 but adjusted based on the surrounding temperature stratification.

SGS viscosity is modeled using the Smagorinsky (1963) model

$$\nu^{SGS} = \left(C_s \Delta\right)^2 \left(2\bar{S}_{ij}\bar{S}_{ij}\right)^{\frac{1}{2}}, \tag{7}$$

where $\Delta = (\Delta x \Delta y \Delta z)^{\frac{1}{3}}$ is the grid scale and C_s is the model constant, which is set to OpenFOAM's default value of 0.168.

2.3 *BOUNDARY CONDITIONS*

The conditions on the upper boundary are as follows: the temperature gradient is specified to be that of the initial capping inversion profile; the velocity normal to the boundary and the gradient of the velocity parallel to the boundary are zero; the pressure gradient is obtained from the momentum equation normal to the boundary; and the gradient of SGS viscosity is zero (a boundary condition on SGS viscosity is used to indirectly specify the boundary condition of stress for the reasons discussed in Section 3.3).

At the lower boundary, the temperature flux is prescribed, the pressure gradient is again obtained from the momentum equation normal to the boundary, and the boundary normal velocity is zero. In the more commonly seen ABL codes in which the variables are staggered, there is no need to specify velocity parallel to the lower boundary. However, in this colocated code, a horizontal velocity at the lower boundary is necessary solely to compute velocity gradients at the cell centers immediately adjacent to this boundary. This velocity gradient is necessary for computing the SGS stress tensor and SGS viscosity. The Monin-Obukhov similarity law (Etling 1996, 400–402) involving the mean velocity gradient is applied locally at each of these cell centers to calculate local values of desired velocity gradient. Then, a velocity is applied at the lower boundary face of each of these cells so that the numerical gradient operation recovers the desired cell center velocity gradient. In this study, friction velocity, u_*, is specified. It is much more common, though, to specify surface roughness, z_0, as in Paulson (1970). We specified u_* rather than z_0 since Moeng and Sullivan (1994) did not list the value of z_0 specified in their simulations to which we compared our results, but they did give the value of u_* characteristic of each of their simulations at the final time. In this way, we achieve the same values of u_* as Moeng and Sullivan (1994) without iterating on a value of z_0 that recovers the desired u_*. (In future simulations, though, we desire to specify z_0 as did Moeng and Sullivan (1994)).

At the surface, $z = 0$, we apply the model of Schumann (1975) to compute the total wall surface shear stress vector

$$R^D_{13_w} = -u_*^2 \frac{\bar{u}_1(x, y, z_1)}{S(z_1)} \tag{8}$$

$$R^D_{23_w} = -u_*^2 \frac{\bar{u}_2(x, y, z_1)}{S(z_1)}, \tag{9}$$

where the subscript w refers to quantities at the lower boundary and $S(z_1)$ is the magnitude of the horizontally-averaged velocity vector (mean wind) at the first grid level. Section 3.3 explains how this surface stress is applied within our solver.

The boundary conditions for all quantities are periodic in the horizontal directions.

3. NUMERICAL METHOD

3.1 *THE OpenFOAM FRAMEWORK*

OpenFOAM (Ope 2009b,a), an acronym for *Open*-source *F*ield *O*perations *A*nd *M*anipulations, is a collection of C++ libraries for solving complex physics problems using the unstructured, colocated, finite-volume formulation. It is not a dedicated CFD solver, but rather a finite-volume toolbox or framework from which one can more easily create a solver and boundary conditions than if starting from scratch. It is this versatility that led us to use OpenFOAM. We have used version 1.6 in this work.

OpenFOAM volume integrates the equations it solves and makes use of Gauss' theorem wherever possible to convert volume integrals to surface integrals. It therefore requires not only cell-centered values of quantities, but also face-centered values. The values of quantities on faces are approximated as constant over the face, which limits the solver to second-order spatial accuracy. OpenFOAM included many methods for interpolation of quantities from cell centers to cell faces, ranging from simple upwind and linear interpolation to more sophisticated total variation diminishing (TVD) schemes. In this work, all interpolation to cell faces is linear, which is equivalent to second-order central differencing on uniform meshes. Gradients of quantities, though, should be directly computed on cell faces, which is crucial in implementing the method of Rhie and Chow (1983) used to compute velocity fluxes through cell faces to avoid the pressure-velocity decoupling that occurs with incompressible flow solvers using colocated variables.

Although OpenFOAM is a toolbox, its source distribution includes a variety of solvers for various types of flow, including incompressible flow. OpenFOAM runs in a Linux environment and is parallelized using the Message Passing Interface (MPI). Because it is an unstructured solver, the solution matrices resulting from implicit discretization are sparse, but not necessarily organized, so conjugate-gradient Krylov-subspace or multigrid solvers with the option for preconditioning are included in OpenFOAM.

3.2 *THE PISO SOLUTION ALGORITHM*

Many of the incompressible solvers included with the OpenFOAM distribution are based on the PISO (*P*ressure-*I*mplicit with *S*plitting of *O*perations) algorithm (Issa 1985). We used these solvers as a guide to create our LES solver for ABL flows. The basic idea of this algorithm is to solve the momentum equation (Equation 1) such that the velocity vector and pressure are treated implicitly in time to make the numerical scheme more stable. However, in our solver, the buoyant term, the SGS viscosity, and the Coriolis terms are treated explicitly. Following the other OpenFOAM PISO solvers, we numerically formulated the convec-

tive term of the momentum equation (the first term on the right-hand side of Equation 1) by making the velocity flux, $\phi_f = S_{j_f}\, \bar{u}_j|_f$, explicit in time, where the subscript f denotes linear interpolation of a quantity to cell face f from the cell center, and S_{j_f} is cell face f's surface normal vector. In the finite-volume formulation, the convective term is numerically integrated as follows,

$$\int_V \frac{\partial}{\partial x_j}\left(\bar{u}_j \bar{u}_i\right) \mathrm{d}V \approx \sum_f S_{j_f}\, \bar{u}_j|_f\, \bar{u}_i|_f = \sum_f \phi_f \bar{u}_i|_f, \tag{10}$$

and is treated in time as

$$\sum_f \phi_f^n \bar{u}_i|_f^{n+1}, \tag{11}$$

where n and $n+1$ are the old and new time levels, respectively. The PISO algorithm then attempts to solve the linear system

$$C\bar{u}_i^{n+1} + \frac{\delta}{\delta x_i}\tilde{p}^{n+1} = r_i, \tag{12}$$

where C is the resulting left-hand side matrix resulting from implicit time treatment, the $\delta/\delta x_i$ operator denotes the numerical approximation for the gradient, and r_i are any explicit right-hand side terms. Of course, the solution of Equation 12 would require the simultaneous solution of the momentum equation and pressure equation, which enforces continuity, presumably by some iterative method. The PISO algorithm removes the need for iteration by first predicting the momentum equation with velocity treated implicitly in time but with pressure treated explicitly. Then, a series of corrector steps occur in which the pressure equation is solved and the velocity is corrected explicitly. The original implicit momentum equation is then satisfied to second-order accuracy after two corrector steps.

The predictor step solves the following implicit linear system

$$C\bar{u}_i^* = A\bar{u}_i^* + H'\bar{u}_i^* = -\frac{\delta}{\delta x_i}\tilde{p}^n + r_i, \tag{13}$$

where A is a matrix containing the diagonal terms of C, H' is the matrix containing the off-diagonal terms of C, and $*$ denotes the predicted quantity.

The first corrector step solves the explicit linear system

$$A\bar{u}_i^{**} = -\frac{\delta}{\delta x_i}\tilde{p}^{**} + r_i - H'\bar{u}_i^* = -\frac{\delta}{\delta x_i}\tilde{p}^{**} + H_i^*, \tag{14}$$

where $H_i^* = r_i - H'\bar{u}_i^*$ and $**$ denotes the first corrected quantities. To correct the velocity field, the first corrected pressure field must be solved first using the discrete continuity equation, which states that $\delta/\delta x_i\left(\bar{u}_i^{**}\right) = 0$. Applying the same numerical divergence operator used in the continuity equation to Equation 14 results in a Poisson equation for the first corrected pressure

$$\frac{\delta}{\delta x_i}\left(\frac{\delta}{\delta x_i}A^{-1}\tilde{p}^{**}\right) = \frac{\delta}{\delta x_i}\left(A^{-1}H_i^*\right). \tag{15}$$

The second velocity corrector is

$$A\bar{u}_i^{***} = -\frac{\delta}{\delta x_i}\tilde{p}^{***} + H_i^{**}, \tag{16}$$

and the second pressure corrector is

$$\frac{\delta}{\delta x_i}\left(\frac{\delta}{\delta x_i}A^{-1}\tilde{p}^{***}\right) = \frac{\delta}{\delta x_i}\left(A^{-1}H_i^{**}\right). \tag{17}$$

Crank-Nicholson time advancement is used for the implicit velocity predictor, shown in Equation 13, which is solved using a diagonal incomplete LU-preconditioned bi-conjugate gradient solver.

The pressure Poisson corrector shown in Equation 15 is then solved. Here, a method like that used by Rhie and Chow (1983) to compute velocity fluxes at cell faces (as opposed to simply interpolating cell centered values of velocity to the faces) and a compact Laplacian stencil are necessary to avoid pressure-velocity decoupling as seen with pressure-bases solvers in which the variables are colocated on the mesh. Rhie-Chow interpolation is implemented by first computing the quantity $A^{-1}H_i^*$, and then linearly interpolating it to the cell faces. In effect, the velocity flux without the influence of the pressure gradient has been computed, which we will call $\phi_{-\tilde{p}}|_f$.

The discrete divergence operator of the continuity equation applied on the right-hand side of Equation 15 uses the face values of $A^{-1}H_i^*$ which is the quantity $\phi_{-\tilde{p}}|_f$ that has just been computed. The discrete divergence operator of the continuity equation is also applied to the discrete gradient operator acting on the pressure found in left-hand side of Equation 15. This successive application of discrete operators would first compute the pressure gradient at the cell centers and interpolate them to the cell faces, and then difference those cell face gradients. As a result, the stencil only includes every other cell center value of pressure, which aides in the pressure-velocity decoupling process.

To circumvent this problem, the left-hand side term of Equation 15 is evaluated by finding the pressure gradient at the cell faces directly using the cell center values of pressure, rather than interpolating the cell center values of gradient of pressure. This introduces an inconsistency in the treatment of the gradient operators in the momentum and pressure equations, but according to Ferziger and Perić (2002, p. 200), the error introduced is of the same magnitude of the basic discretization and does not add significantly to the discretization error. However, the scheme is not able to

conserve energy that may cause instabilities, though we have not observed instabilities of this sort.

Once Equation 15 for the pressure corrector is iteratively solved using a diagonal incomplete Cholesky-preconditioned conjugate gradient solver, the contribution to the velocity flux by the surface normal pressure gradient, $\phi_{+\tilde{p}}|_f = (\delta\tilde{p}/\delta x_i)\, S_{i_f}$, which is directly computed at the cell faces, is subtracted from $\phi_{-\tilde{p}}|_f$ resulting in the actual velocity flux at the face, $\phi_f = \phi_{-\tilde{p}}|_f - \phi_{+\tilde{p}}|_f$. This completes the Rhie-Chow interpolation.

The cell-centered $\delta\tilde{p}/\delta x_i$ is then reconstructed from the surface normal pressure gradient using

$$\frac{\delta}{\delta x_i}\tilde{p} = \sum_f \phi_{+\tilde{p}}|_f \frac{S_{i_f}}{|S_{i_f}|} \cdot \left(\sum_f \frac{S_{i_f}^2}{|S_{i_f}|} \right)^{-1}, \qquad (18)$$

where $S_{i_f}^2$ is an element-wise squaring of the cell face normal vectors.

Last, the cell-centered velocity is computed using $\bar{u}_i^* * = A^{-1}H_i^* - A^{-1}\delta\tilde{p}/\delta x_i$. Issa (1985) states that two corrector steps are necessary to approximate the solution of Equation 12 to second-order accuracy in time, which is the same as the order of accuracy of the Crank-Nicholson time advancement used in that equation.

At this point, Equations 3 and 7 for resolved potential temperature and SGS viscosity, respectively, are solved. The convective term of the temperature equation is now linear because the velocity at time step $n+1$ is already known, and the equation is solved implicitly with Crank-Nicholson time advancement. The resulting system of equations is iteratively solved using a diagonal incomplete LU-preconditioned bi-conjugate gradient solver.

3.3 *IMPLEMENTATION OF THE STRESS TERM*

The SGS stress term in Equation 1 (the second term on right-hand side) should not be differenced using the numerical operation $\delta/\delta x_j \left(R_{ij}^D\right)$. The resulting stencil uses the velocity at every other cell on a uniform mesh, which causes an undesirable "checker-board" velocity instability to arise. Instead, the SGS stress term can be recast by inserting the linear relationship between the SGS viscosity and the resolved strain-rate of Equation 4 resulting in

$$\frac{\partial}{\partial x_j} R_{ij}^D = -\frac{\partial}{\partial x_j}\left(\nu^{SGS}\frac{\partial \bar{u}_i}{\partial x_j}\right) - \frac{\partial}{\partial x_j}\left[\nu^{SGS}\left(\frac{\partial \bar{u}_j}{\partial x_i} - \frac{2}{3}\frac{\partial \bar{u}_k}{\partial x_k}\delta_{ij}\right)\right], \qquad (19)$$

where δ_{ij} is the Kronecker delta. The dominant term of this formulation is the first one on the right-hand side. It is differenced much like the Laplacian in the pressure equation by first evaluating the velocity gradient at cell faces directly, rather than evaluating them at cell centers and interpolating to the faces, and then applying the divergence operator directly to them. This discretization method results in a stencil of Δx rather than $2\Delta x$ and eliminates the checker-board instabilities; however, it limits the solver to SGS stress models based on the relationship in Equation 4.

As discussed in Section 2.3, we use the model of Schumann (1975) to compute surface shear stress vector, but as Equation 19 shows, stress does not directly enter the equations that are solved numerically. By transforming (at each cell adjacent to the lower boundary) into a coordinate system that lies in the bottom boundary plane and that is aligned with the local resolved velocity at the cell-center height, z_1, and then equating the right-hand sides of Equations 4 and 9, the following results:

$$\left(\nu^{SGS}\frac{\partial \bar{U}}{\partial z}\right)\Bigg|_w = u_*^2 \frac{\bar{U}(x,y,z_1)}{S(z_1)}, \qquad (20)$$

where $\bar{U} = |\bar{u}_i|$ is the magnitude of the resolved velocity vector. Because we have to apply a boundary condition on velocity at the lower surface as discussed in Section 2.3, a one-sided difference can be used to compute a value for $\partial\bar{U}/\partial z|_w(x,y)$ at each cell face adjacent to the lower boundary. Then a value of $\nu_w^{SGS}(x,y)$ can be chosen such that the product of $\nu_w^{SGS}(x,y)$ and $\partial\bar{U}/\partial z(x,y)$ is as specified by Schumann's surface stress model as formulated in Equation 20. It is not important that the values of ν_w^{SGS} and $\partial\bar{U}/\partial z|_w(x,y)$ be accurate on their own, but their product must equal the surface stress of Schumann's model. In this way, $\nu_w^{SGS}(x,y)$ is used as a lower boundary condition for ν^{SGS} such that, indirectly, our stress term, given on the right-hand side of Equation 19, recovers the desired surface stress. Consistent with staggered grid boundary layer methods (Moeng 1984), we do not apply no slip to $\bar{u}_i$ at the surface. Instead, we have derived an equation for $(\bar{u}_i)_w$ consistent with the staggered grid formulation.

4. CASES SIMULATED

The four cases simulated range from completely shear driven to moderately convective, and correspond to those simulated by Moeng and Sullivan (1994) (which we will refer to as MS94 from this point forward). All cases are windy (no completely buoyant cases), which are of interest in wind-farm modeling. Differences between our simulations and those of MS94 include the use of a finite-volume solver versus their pseudo-spectral one, use of the Smagorinsky model instead of their one-equation SGS model, use of the surface stress model of Schumann (1975) versus that of Moeng (1984), a less sophisticated formulation of Pr^t, and a different number of computational

Table 1. A description of the cases simulated.

Name	$D_x(D_y)$ (km)	D_z (km)	Q_s (K· m/s)	u_* (m/s)	U_g (m/s)	T_{nom} (s)
B	5	2	0.24	0.56	10.0	9000
S	3	1	0.00	0.50	15.0	15 225
SB1	3	1	0.05	0.59	15.0	9300
SB2	3	1	0.03	0.56	15.0	10 000

mesh volumes. We also do not include dealiasing by truncation as done in pseudo-spectral codes. Eventually, changes can be made to our code to make a more direct comparison.

Table 1 outlines the four cases run. D_x, D_y, and D_z are the streamwise, spanwise, and vertical dimensions of the domains, Q_s is the specified surface potential temperature flux, u_* is the friction velocity, U_g is the geostrophic wind at the top of the domain, and T_{nom} is the time at which we begin to save data for our analysis. The geostrophic wind is maintained through the adjustment of the driving pressure gradient of the momentum equation (Equation 1). In all cases, we use a mesh of $N_x \times N_y \times N_z = 160 \times 160 \times 128$, whereas MS94 used 96 grid points in all directions for all cases. As discussed in Section 2.3, because MS94 did not list the roughness height, z_0, prescribed in their simulations but do give the resulting values of u_* at the end of their simulations, we directly prescribe these values of u_* listed for each of MS94's corresponding cases.

All cases have an initial temperature field containing a capping inversion. The virtual potential temperature is 300 K up to an initial capping inversion height, z_{i_0}, and then it increases linearly by 8 K over the next 83 m (or 167 m in case B) followed by an increase of 0.003 K/m to the top of the domain. The initial capping inversion height is 468 m (937 m in case B). The initial velocity field is perturbed in a divergence-free way (de Villiers 2006, p. 163) only near the lower boundary in order to initiate turbulence in the solution. In all cases, the time step is 0.25 s, which corresponds to a maximum Courant-Friedrichs-Lewy (CFL) number of roughly 0.23 except for case B in which the maximum CFL number is 0.10.

5. RESULTS

To compare our work with that of MS94, we saved data beginning at MS94's stated simulation end times for the next 1000 s every 200 s. These end times are 9300 s, 15 225 s, 9000 s, and 10 000 s for cases B, S, SB1, and SB2, respectively. For example, in case SB2, data were saved at 10 000 s, 10 200 s, 10 400 s, 10 600 s, 10 800 s, and 11 000 s. Using these data, we created vertical profiles from both horizontal averages and time averages. We also created two-dimensional variance spectra at various vertical levels and averaged in time. Instantaneous slices or isocontours, however, were taken only at the stated end time.

5.1 *STATISTICS*

Various statistics for each of our simulations are presented in Table 2 along with the statistics of MS94's simulations. The percent difference between our statistics and those of MS94 are shown at the bottom of the table. The statistics shown are the convective velocity, w_*; the ABL height defined as the height of the maximum negative temperature flux due to the capping inversion, z_i; the stability parameter, $-z_i/L$, where L is the Obukov length; the large-eddy turnover time, τ_*; the virtual potential temperature flux at the top of the ABL, $\langle w'\theta' \rangle_{z_i}$; the horizontally-averaged wind speed at the height of MS94's first grid level, $S(z_1)$; the horizontally-averaged difference in temperature between the surface and the height of MS94's first grid level, $\Delta\theta(z_1)$; and the turbulent kinetic energy at a standard turbine hub height of z = 80 m, $k(z_{80})$. The convective velocity is defined as

$$w_* = \left(\frac{g}{\bar{\theta}^0} Q_s z_i\right)^{1/3}, \tag{21}$$

where g is the magnitude of the gravity vector. The Obuhkov length is

$$L = -\frac{\bar{\theta}^0 u_*^3}{g Q_s \kappa}, \tag{22}$$

where κ is the von Kármán constant, which is taken as 0.4. For the shear-driven case, $\tau_* = z_i/u_*$, but for all other cases in which convection is present, $\tau_* = z_i/w_*$.

Table 2 shows that our shear-driven simulation S has the largest percent difference between that of MS94. Interestingly, the ABL height predicted in this work is much lower than that predicted by MS94 for simulation S. This occurs because, in our simulation S, the fairly sharp transition in the initial temperature profile at the start of the capping inversion is smoothed with time as shown in Figure 1. In effect, the height of the start of the capping inversion is decreased with time, which lowers the top of the ABL with time and causes a 33.8% difference between our simulation's height and that of MS94. This, in effect, causes a large difference in predicted large-eddy turnover time.

Table 2. Internal parameters of the current simulations and those of Moeng and Sullivan (1994) at the final simulation times stated by Moeng and Sullivan. Also included are the percent differences between our data and Moeng and Sullivan's data.

	Name	w_* (m/s)	z_i (m)	$-z_i/L$ (m)	τ_* (s)	$\langle w'\theta' \rangle_{z_i}$ (K - m/s)	$S(z_1)$ (m/s)	$\Delta\bar{\theta}(z_1)$ (K)	$k(z_{80})$ $(m/s)^2$
	B	1.99	1008	18	506	-0.036	6.2	0.30	2.99
Current	S	0	324.2	0	633	-0.014	4.5	0	0.77
Work	SB1	0.88	425.8	1.3	475	-0.020	5.7	0.06	1.07
	SB2	0.73	371.1	0.9	541	-0.014	5.5	0.04	0.94
	B	2.02	1030	18	510	-0.040	5.1	2.8	
Moeng and	S	0	478	0	956	-0.007	8.7	0	
Sullivan (1994)	SB1	0.94	498	1.6	530	-0.020	8.1	0.5	
	SB2	0.79	493	1.4	624	-0.016	8.2	0.3	
	B	1.36	2.1	0.1	1	8.400	21.7	89.2	
Percent	S	0	33.8	0	34	93.714	48.0	0	
Difference	SB1	6.29	16.1	16.8	10	0.950	29.4	87.8	
	SB2	7.75	20.0	37.0	13	12.313	33.1	87.0	

This significant difference between our simulation and theirs may be linked to the fact that we set the turbulent Prandtl number, Pr^t, to 1/3, whereas MS94 use a Pr^t that is sensitized to positive temperature stratification to account for the small mixing length in stable regions, as occurs in the capping inversion. Our value of the turbulent Prandtl number, which is part of the potential temperature equation's diffusion term coefficient (see Equations 3 and 6), seems to cause too much sub-grid-scale mixing of temperature in the inversion possibly causing the strong smoothing of the temperature profile. One would then suspect that MS94's LES code maintains the capping inversion profile better than our current code. Possibly related is the fact that the potential temperature flux predicted at the inversion height in simulation S is about twice as great as that predicted by MS94.

The statistics predicted for the buoyant simulation B compare best overall to the corresponding statistics predicted by MS94. The degree to which the combined shear- and buoyancy-driven simulations' statistics compare to MS94's statistics lie somewhere in between those for simulations S and B. In fact, simulation SB2, which is more similar to the poorly performing simulation S, has a larger percent difference from MS94 than does simulation SB1.

Interestingly, our predicted values of $S(z_1)$ do not compare well to those of MS94. $S(z_1)$ for simulation B is higher than that predicted by MS94, and $S(z_1)$ for the other cases was predicted to be lower than those of MS94. This discrepancy may be caused by the fact that we used the stress model of Schumann (1975) at the lower boundary, but MS94 used the nonlinear model of Moeng (1984). Also of concern is that our predicted differences in temperature between the surface and the height of MS94's first grid level are about an order of magnitude smaller than those predicted by MS94 for an unknown reason.

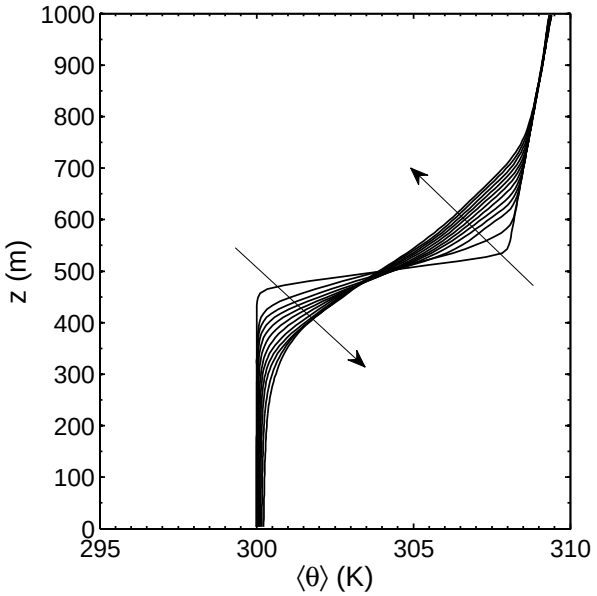


Fig. 1. Vertical profiles of horizontally-averaged potential temperature, $\langle\bar{\theta}\rangle$, at various times. The arrows indicate the direction of evolution of the profile with time.

One can see that as the amount of surface temperature flux is increased, the amount of turbulent kinetic energy that a turbine will encounter also increases. Furthermore, the increased turbulent kinetic energy will greatly affect the recovery length of the wind turbines' wakes and, therefore, the overall wind farm performance.

5.2 *VERTICAL PROFILES*

Figure 2 compares horizontally-averaged velocity profiles normalized by the geostrophic wind speed for all simulations. These plots clearly show that the shear-driven simulation S exhibits the horizontal velocity, $\sqrt{\langle U\rangle^2 + \langle V\rangle^2}$, with greatest shear. The buoyancy-driven simulation B has a velocity profile with much less shear as a result of the enhanced vertical mixing. The velocity profile predicted for simulation B compares well to that predicted by MS94 (denoted with circle symbols). The profile from simulation S does not compare as well, and we attribute the difference to the much reduced ABL height experienced in our simulation. The profiles from simulations SB1 and SB2 are similar, but, as expected, the increased amount of surface temperature flux in case SB1 leads to more vertical mixing and a profile with less shear. From a wind-energy perspective, velocity profiles with greater shear across the rotor disk, as seen in simulation S, cause cyclic loading on the turbine blades that periodically rotate through regions of slower and faster moving air.

Next, resolved velocity variances normalized by u_*^2 in the shear-driven case or w_*^2 in all other cases are shown in Figure 3. Comparison to the total (resolved plus sub-filter-scale) variances of MS94 are shown for simulations S and B. It is important to note that a direct comparison between our resolved variance profiles and the total variance profiles of MS94 cannot be made because their profiles include the sub-filter-scale contribution; however, some general observations can be made. Although the predicted variances share the same general trends as those of MS94 for cases S and B, there are slight differences in magnitude. As in MS94, our simulations predicted that for case S, the profile of $\langle u'u'\rangle$ is greater than that of $\langle v'v'\rangle$, which is greater than that of $\langle w'w'\rangle$. All of the profiles have greatest magnitude near the lower surface around $z/z_i = 0.1$. In contrast, the profiles for simulation B show that $\langle w'w'\rangle$ peaks around $z/z_i = 0.3$ and those of $\langle u'u'\rangle$ and $\langle v'v'\rangle$ peak both very near the lower surface and at the base of the capping inversion. In simulation B, $\langle u'u'\rangle$ and $\langle v'v'\rangle$ are similar in magnitude. The profiles for simulations SB1 and SB2 are a blend between those of simulations S and B. The profiles of the more convective case SB1 are more similar to those of simulation B, whereas the profiles for the more shear-driven simulation SB2 are more similar to those of simulation S.

Figure 4 shows plots of resolved vertical momentum fluxes and comparison to MS94 is provided for simulations S and B. The MS94 data are of total vertical momentum flux; therefore, their profiles do not approach zero near the ground, whereas our profiles do because we do not include the sub-grid-scale contribution to the momentum flux. Again, both cases agree qualitatively well, but there are some discrepancies in magnitude, which are more prevalent in case S. This may be because we are comparing resolved and total fluxes. The momentum flux profiles from simulation B best match those of MS94.

Figure 5 shows plots of resolved vertical flux of velocity variance. Comparisons to MS94's data are provided for simulations S and B. In this case, MS94 did not include the sub-grid-scale component because LES does not provide this, so equal comparisons between our data and theirs may be made. The profile for simulation B better matches MS94's profile than does that of simulation S. In simulation B, we predicted that $\langle w'^3\rangle$ is greater than the other variance fluxes, and that the $\langle w'u'^2\rangle$ and $\langle w'v'^2\rangle$ fluxes are nearly identical, similar to what MS94 predicted. In simulation S, though, we predicted that $\langle w'u'^2\rangle$ and $\langle w'v'^2\rangle$ are nearly identical, which does not agree with the predictions by MS94. Our simulation shows that $\langle w'^3\rangle$ has the smallest values, which agrees with MS94. The profiles for simulations SB1 and SB2 are similar to one another, those of simulation SB1 being smaller in magnitude, and resemble the profiles for MS94's simulation S.

5.3 *INSTANTANEOUS x-y CONTOURS*

Figures 6 and 7 show x-y contours of u' for simulations S and SB1, respectively, and Figure 8 shows the contours of w' for simulation B. These contours are shown at z/z_i of 0.1, 0.2, 0.3, 0.5, and 0.8.

In the shear-driven simulation S, shown in Figure 6, streak-like structures in the horizontal velocity fluctuations are observed near the surface and are aligned with the mean flow at that height. Because of the Coriolis force, the mean flow is directed at a roughly 45° angle to the x-direction. At $z/z_i = 0.5$, however, the structures are no longer apparent. This agrees well with MS94's simulation S, both in the characteristics of the structures and the magnitude of the fluctuations. Our grid is finer than that used by MS94, therefore it captured smaller-scale structures.

In simulation SB1, a moderate level of surface temperature flux is present, which gives rise to flow-oriented roll cells. Evidence of these cells is present in Figure 7. In the range $0.1 \leq z/z_i \leq 0.3$, low-speed streaks of horizontal velocity are spaced by $2z_i$-$3z_i$, which is the experimentally observed spacing of the roll cells (LeMone 1973). Although not shown here, these low-speed streaks of horizontal velocity correlate well with streaks of positive vertical fluctuations, providing more evidence that the roll cells are present.

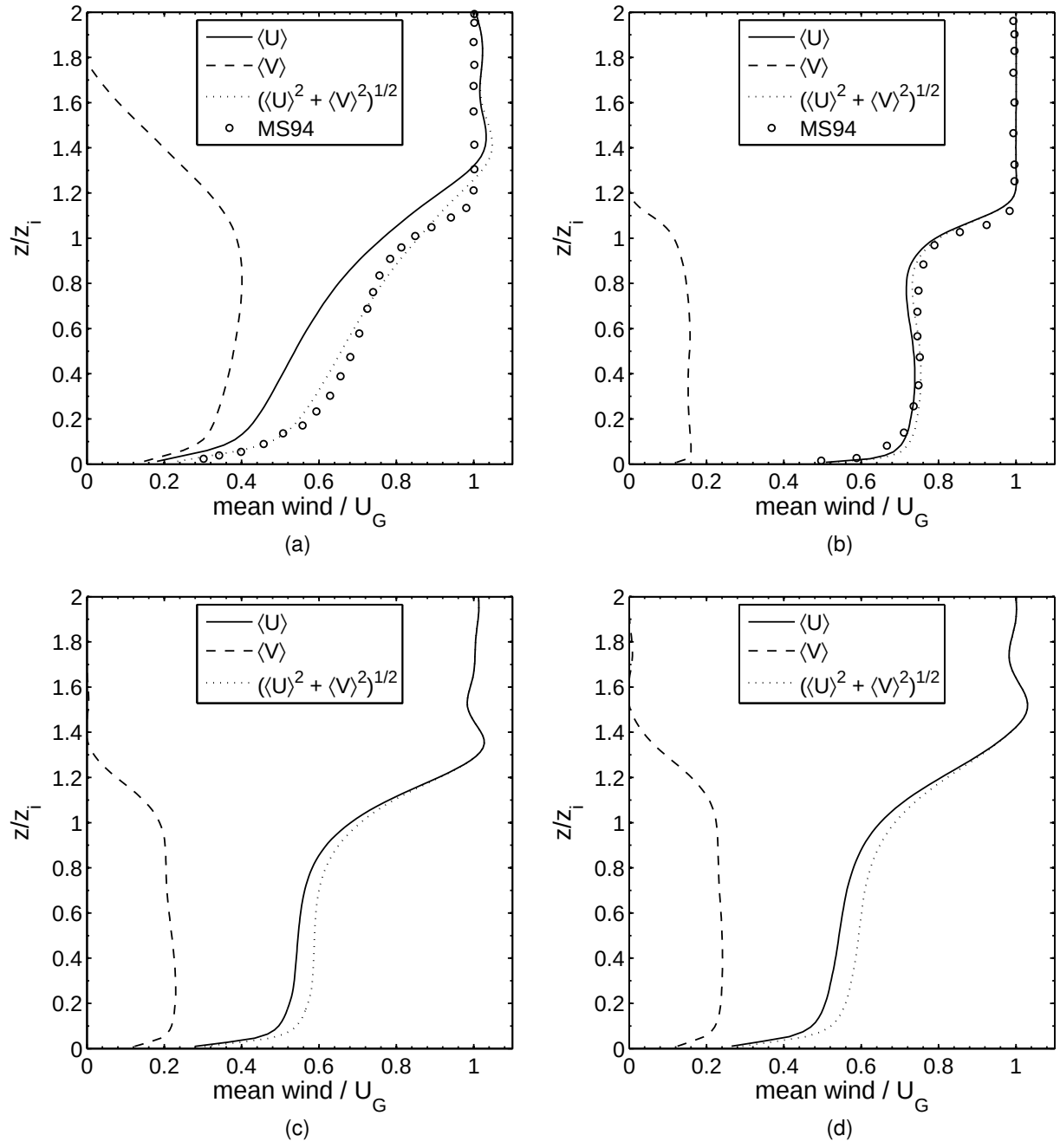


Fig. 2. Vertical profiles of horizontally-averaged resolved velocity for (a) simulation S, (b) simulation B, (c) simulation SB1, and (d) simulation SB2.

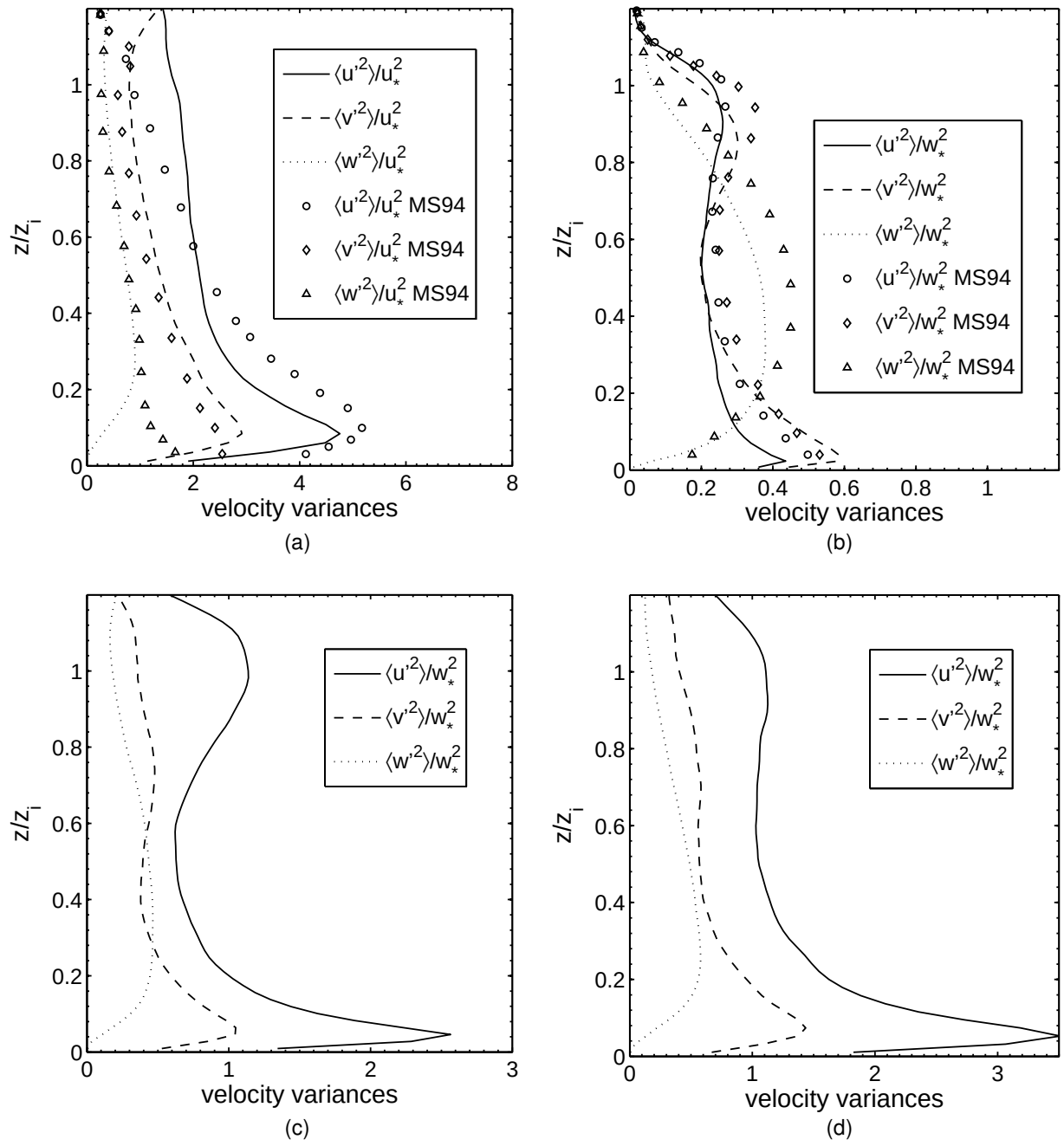


Fig. 3. Vertical profiles of horizontally-averaged resolved velocity variance for (a) simulation S, (b) simulation B, (c) simulation SB1, and (d) simulation SB2.

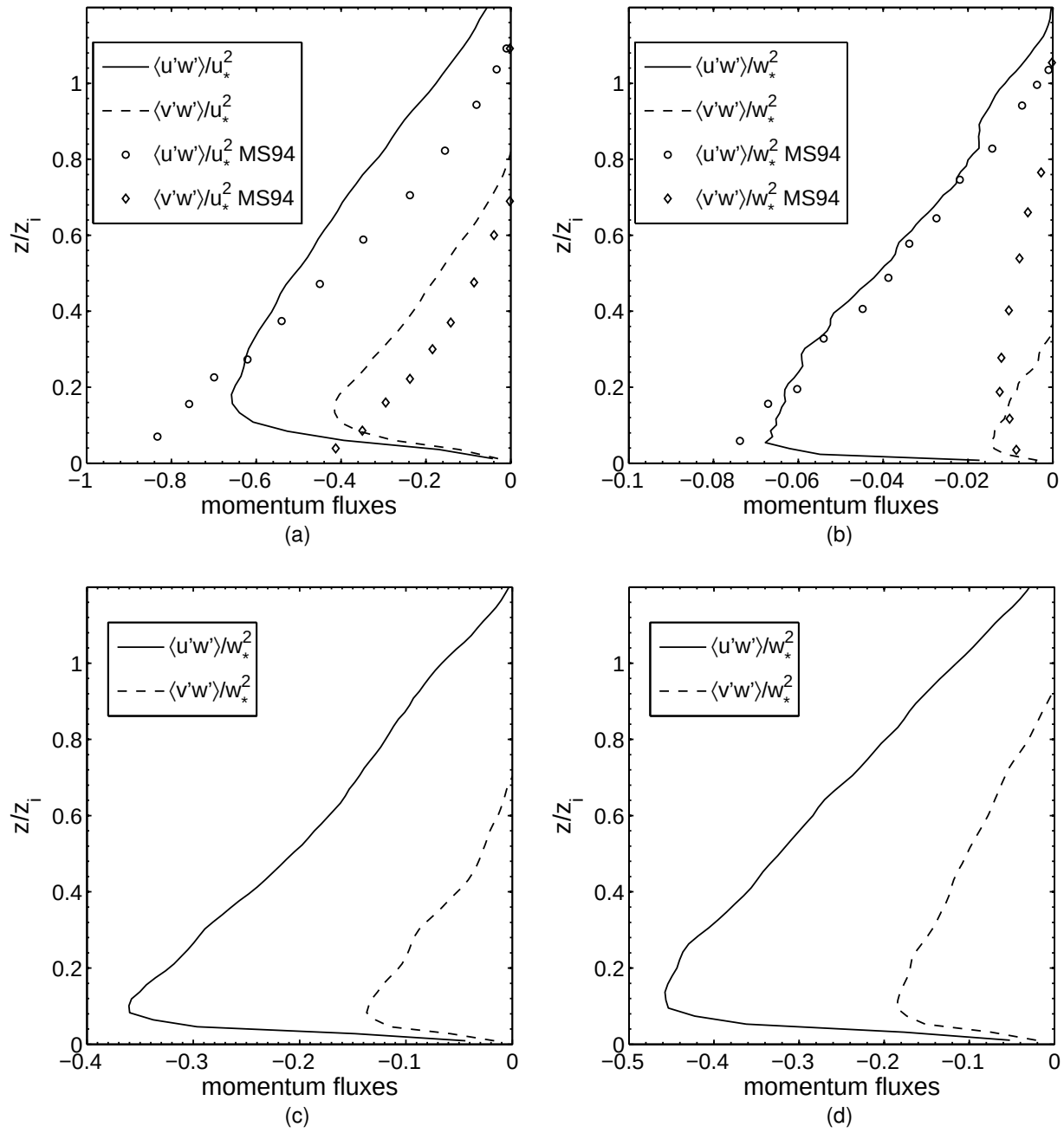


Fig. 4. Vertical profiles of horizontally-averaged resolved momentum fluxes for (a) simulation S, (b) simulation B, (c) simulation SB1, and (d) simulation SB2.

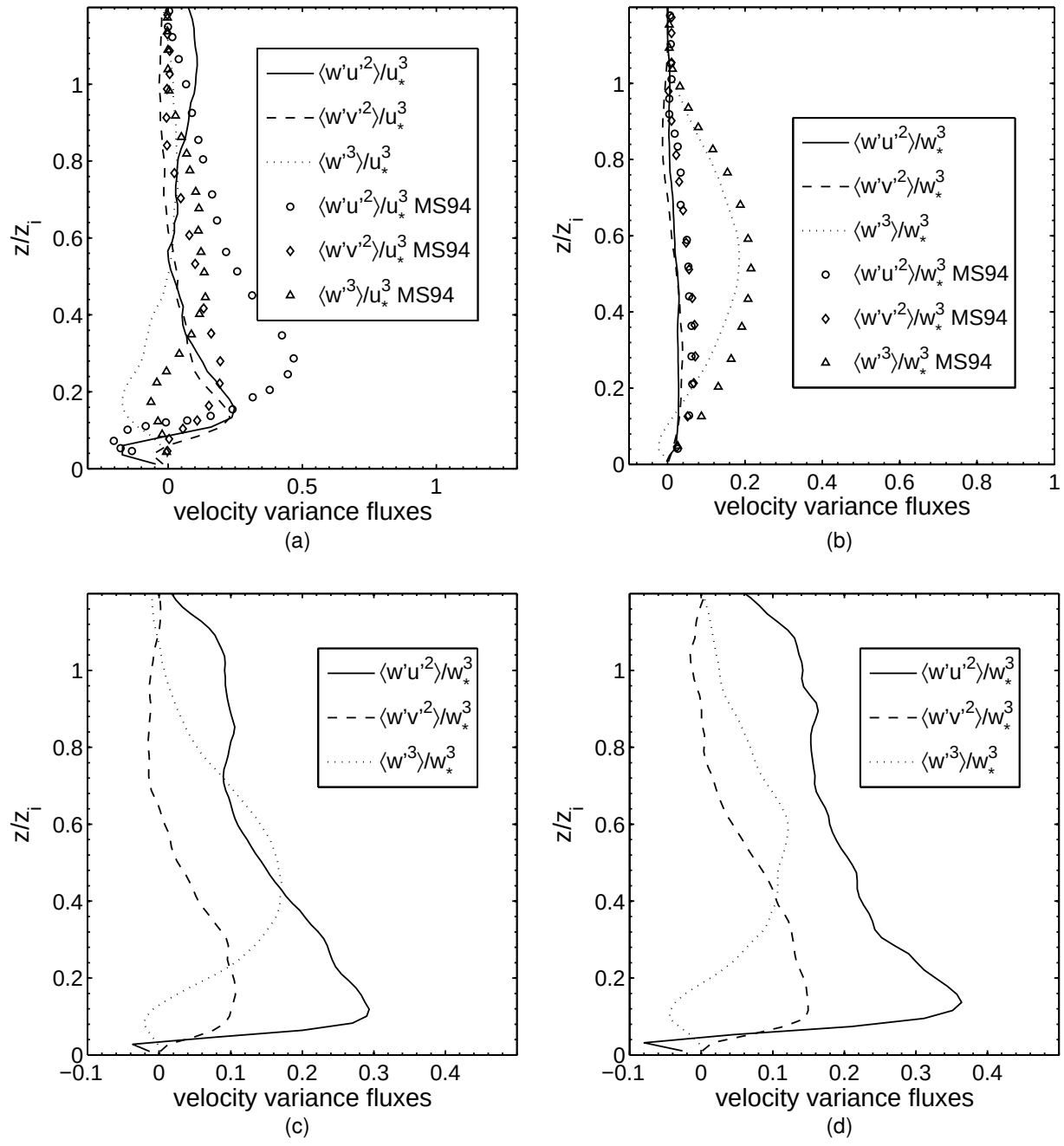


Fig. 5. Vertical profiles of horizontally-averaged resolved velocity variance fluxes for (a) simulation S, (b) simulation B, (c) simulation SB1, and (d) simulation SB2.

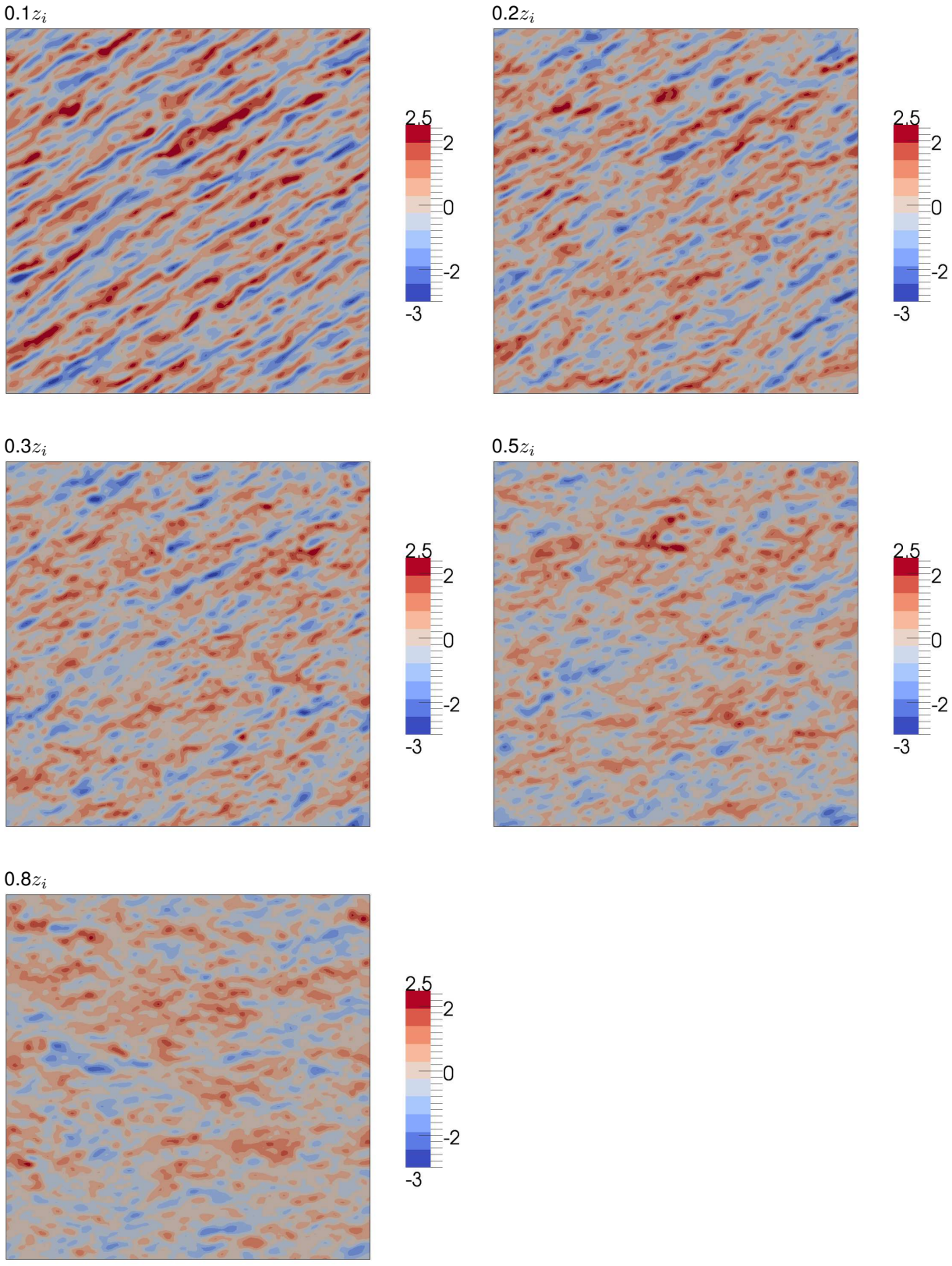


Fig. 6. Contours of x-directed horizontal velocity fluctuation, u' (m/s), at various heights in x-y planes from simulation S. The entire x-y domain extents of 3 km $\times$ 3 km are shown.

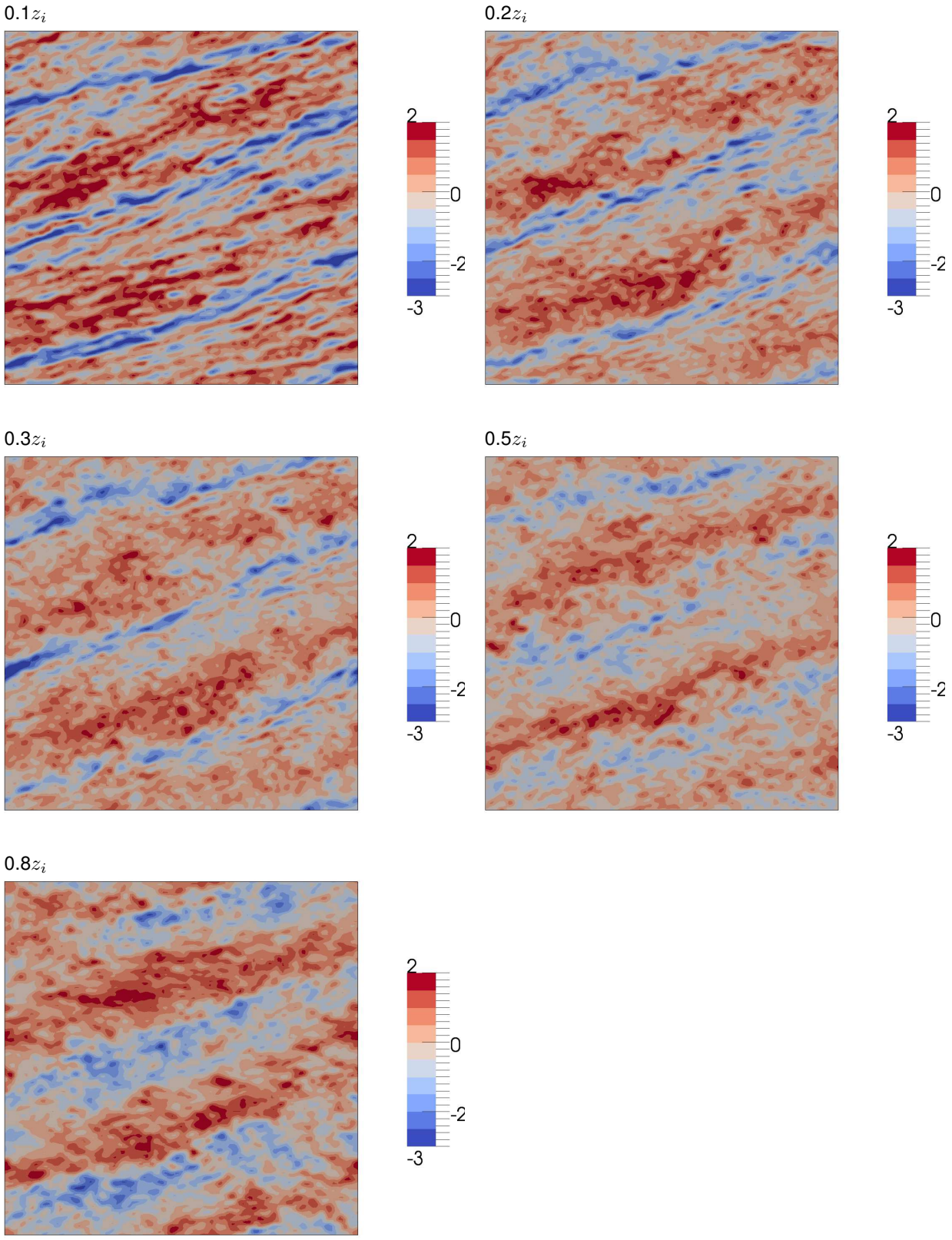


Fig. 7. Contours of x-directed horizontal velocity fluctuation, u' (m/s), at various heights in x-y planes from simulation SB1. The entire x-y domain extents of 3 km $\times$ 3 km are shown.

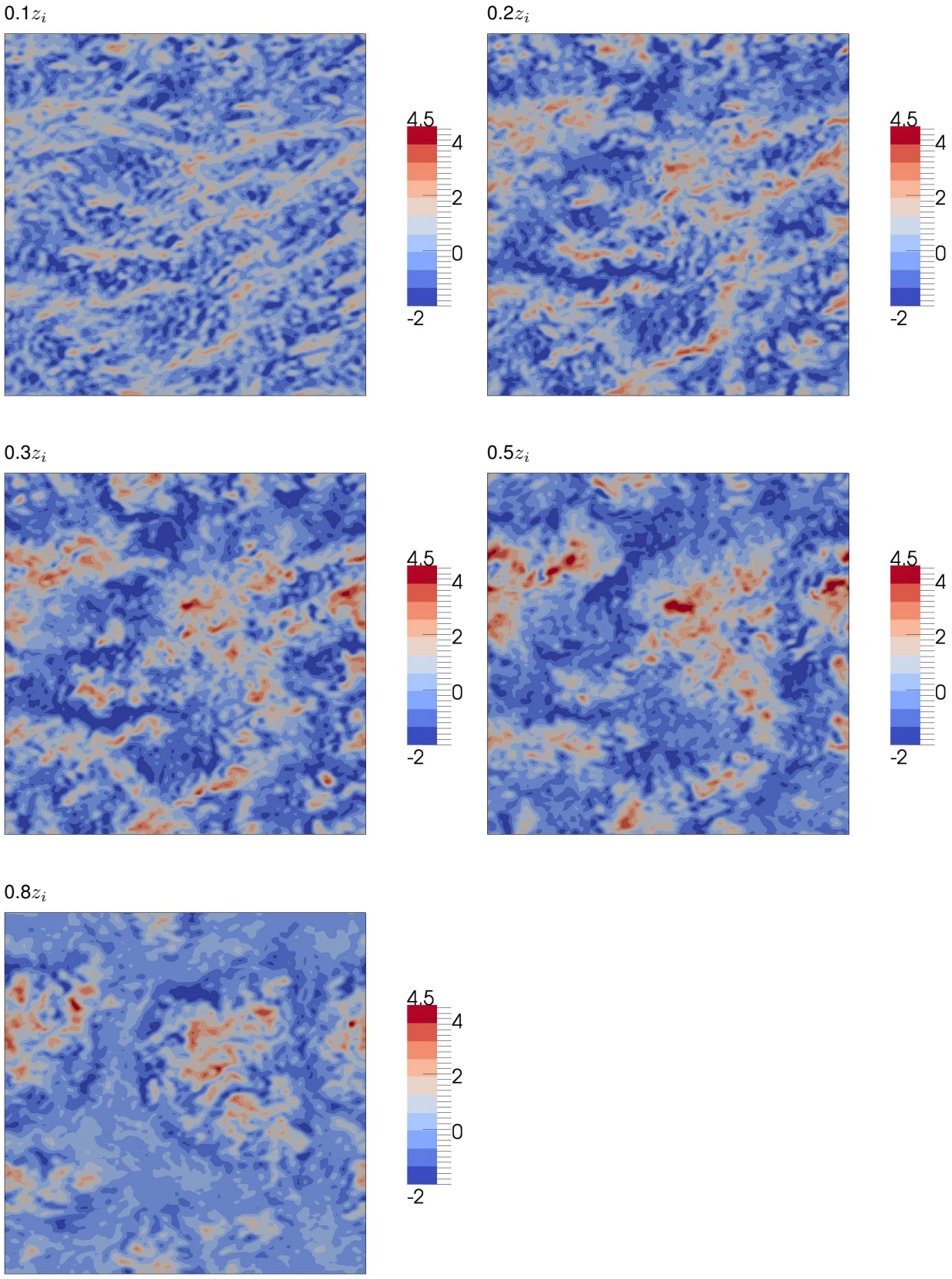


Fig. 8. Contours of vertical velocity fluctuation, w' (m/s), at various heights in x-y planes from simulation B. The entire x-y domain extents of 5 km $\times$ 5 km are shown.

At greater heights within the ABL, the low-speed regions of the horizontal velocity become fairly equal in width to the high-speed regions.

Figure 8 shows contours of vertical velocity fluctuations for the buoyancy-driven simulation B. Here, there are concentrated regions of positive vertical velocity fluctuations and less dense regions of negative vertical fluctuations. The positive fluctuations are larger in magnitude than the negative fluctuations, which agrees well with the simulations of MS94. With increasing height, the concentrated positive regions change in character from long structures to clump-like structures, similar to that observed in the corresponding simulation of MS94.

5.4 *SPECTRA*

Figures 9 and 10 show spectra of energy contained within the horizontal velocity fluctuations, vertical velocity fluctuations, and virtual potential temperature fluctuations for cases S and B, respectively, at heights of $z/z_i = 0.1$, 0.2, 0.3, 0.5, and 0.8. As was done by Khanna and Brasseur (1998), a two-dimensional fast Fourier transform is used to produce a two-dimensional energy spectrum at a given height in the flow field. Then, the spectral energy contained within shells of constant wavenumber magnitude, $\mathbf{k} = \sqrt{k_1 + k_2}$, is computed to produce these logarithmic plots of spectral energy versus wavenumber. Shown in gray on each plot is the $\mathbf{k}^{-5/3}$ line. This line is shown to compare our spectra to the Kolmogorov -5/3 spectrum (Pope 2001, p. 200) for the inertial range of turbulence scales.

Figure 9 shows that in the shear-driven simulation S, in which $-z_i/L = 0$, the amount of low wavenumber (larger lengthscale) energy contained within the vertical velocity fluctuations gradually increases with height because the lengthscale of such fluctuations increase with height above the surface. The same occurs, but to a much lesser degree, with the low wavenumber energy contained within the horizontal velocity fluctuations. This behavior is seen in the spectra of the shear-driven flow ($-z_i/L = 0.44$) computed by Khanna and Brasseur (1998, p. 722). The overall energy contained in the potential temperature fluctuations increases with height above the surface. This could be a result of the mixing of cooler fluid with warmer fluid from the inversion, which causes larger temperature fluctuations near the capping inversion than near the surface. A major difference between our spectra for simulation S and those of Khanna and Brasseur (1998) for their shear-driven simulation is that above $z/z_i = 0.25$, the spectra of Khanna and Brasseur (1998) show a clear inertial range of -5/3 slope for both velocity and temperature. Our shear-driven spectra do not follow a -5/3 slope at any height.

Figure 10 shows that in the case with greatest surface temperature flux, simulation B, in which $-z_i/L = 18$, the amount of low wavenumber energy contained within the vertical velocity fluctuations increases in the range $0.1 \leq z/z_i \leq 0.3$ and then remains fairly constant, which is consistent with the spectra of Khanna and Brasseur (1998, p. 731) in which $-z_i/L = 8$. This increase with height is, again, due to proximity to the surface and the size of vertical velocity lengthscales that the surface permits. Unlike in simulation S, there is no significant increase with height of the low wavenumber energy contained in the horizontal velocity and potential temperature fluctuations, which is also consistent with the spectra of Khanna and Brasseur (1998) for their $-z_i/L = 8$ case. Also unlike in simulation S, the overall amount of energy contained within the potential temperature fluctuations increases up to $z/z_i = 0.2$, and then remains fairly constant. This may be due to the fact that the boundary layer in simulation B is more well mixed, so the energy contained within the horizontal velocity and potential temperature fluctuations is more well mixed. Our spectra from simulation B more closely follow the -5/3 line, especially above z/z_i =0.3, than in simulation S. However, they do not follow the -5/3 line as well as the spectra of Khanna and Brasseur (1998).

The fact than our spectra for both simulations S and B do not have a clear inertial range that follows the -5/3 slope line is of concern. We do not currently have an understanding as to why this is so, but will explore this problem in the near future. One possibility is that, because we are using a finite-volume solver, which is inherently more numerically diffusive than a pseudo-spectral solver, our simulations are reaching the grid cut-off before the inertial range can be appreciably represented. Performing simulations on a finer grid would help validate this idea. The spectra for both simulations S and B, show that the energy cut-off consistently begins around a wavenumber of 0.06, which corresponds to a wavelength of about $6\Delta x = 6\Delta y$.

5.5 *ABL STRUCTURE*

Figure 11 shows flow visualizations that highlight some of the structure present in these ABL flows. Figure 11 (a) is a visualization from case S. The blue isosurfaces represent horizontal velocity fluctuation, $u' = -3u_* = -1.5$ m/s, and the red isosurfaces represent vertical velocity fluctuation, $w' = 0.6$ m/s. This visualization clearly shows the organized streak-like, flow-aligned, horizontal structure within the lowest region of the surface layer. Because this flow is not buoyancy-driven, the vertical velocity fluctuations do not follow an orderly structure; however, the larger of these isosurfaces seems to emanate from regions where there are horizontal velocity deficits (where the blue isosurfaces lie). This compares well to that observed in the LES of Khanna and Brasseur (1998).

Figure 11 (b) is a visualization from case SB1 in which there is moderate surface heating. This visu-

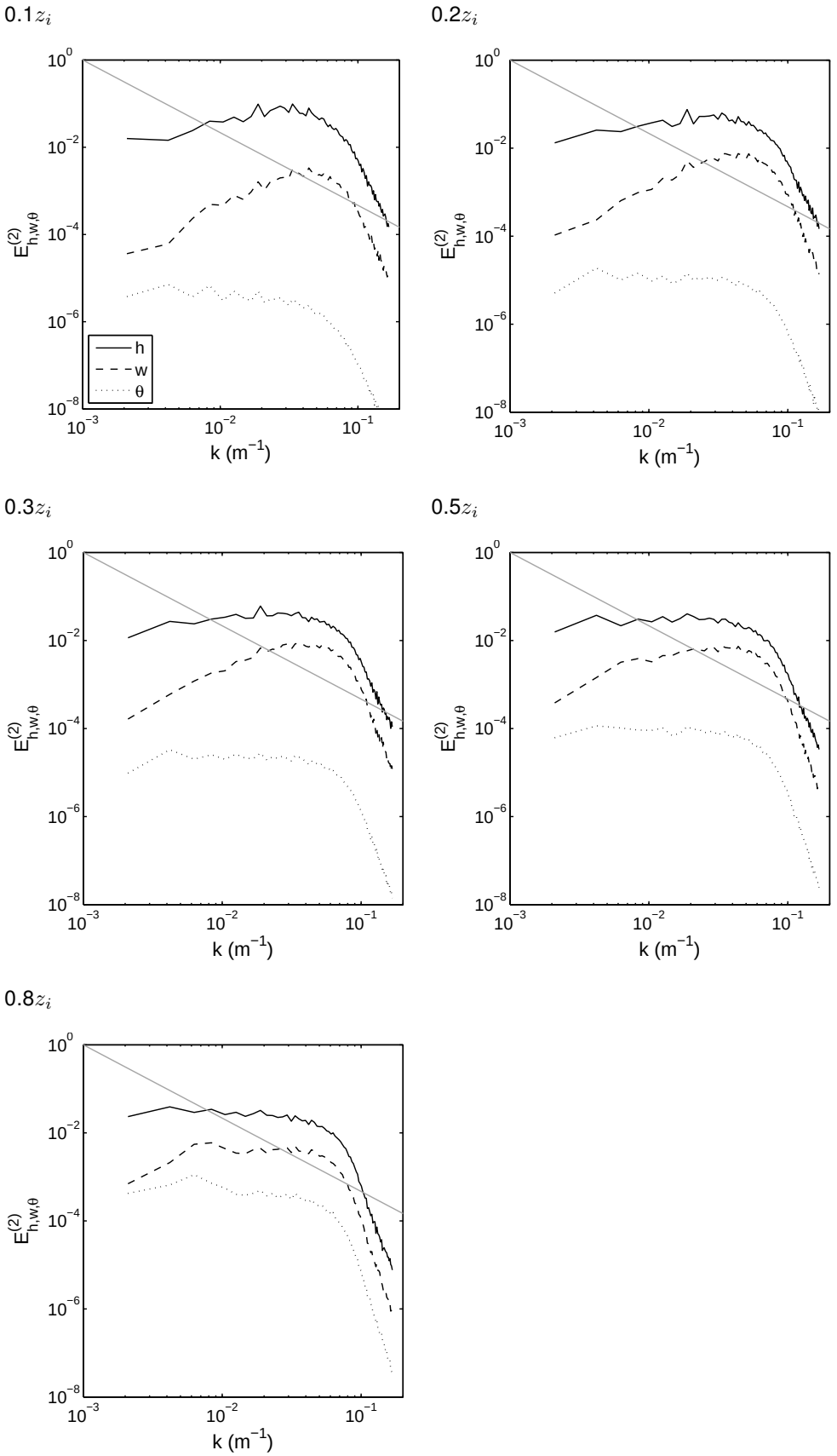


Fig. 9. Two-dimensional spectra of the energy contained in the horizontal velocity fluctuations, h, the vertical velocity fluctuations, w, and the virtual potential temperature fluctuations, θ, taken at various heights in x-y planes from simulation S.

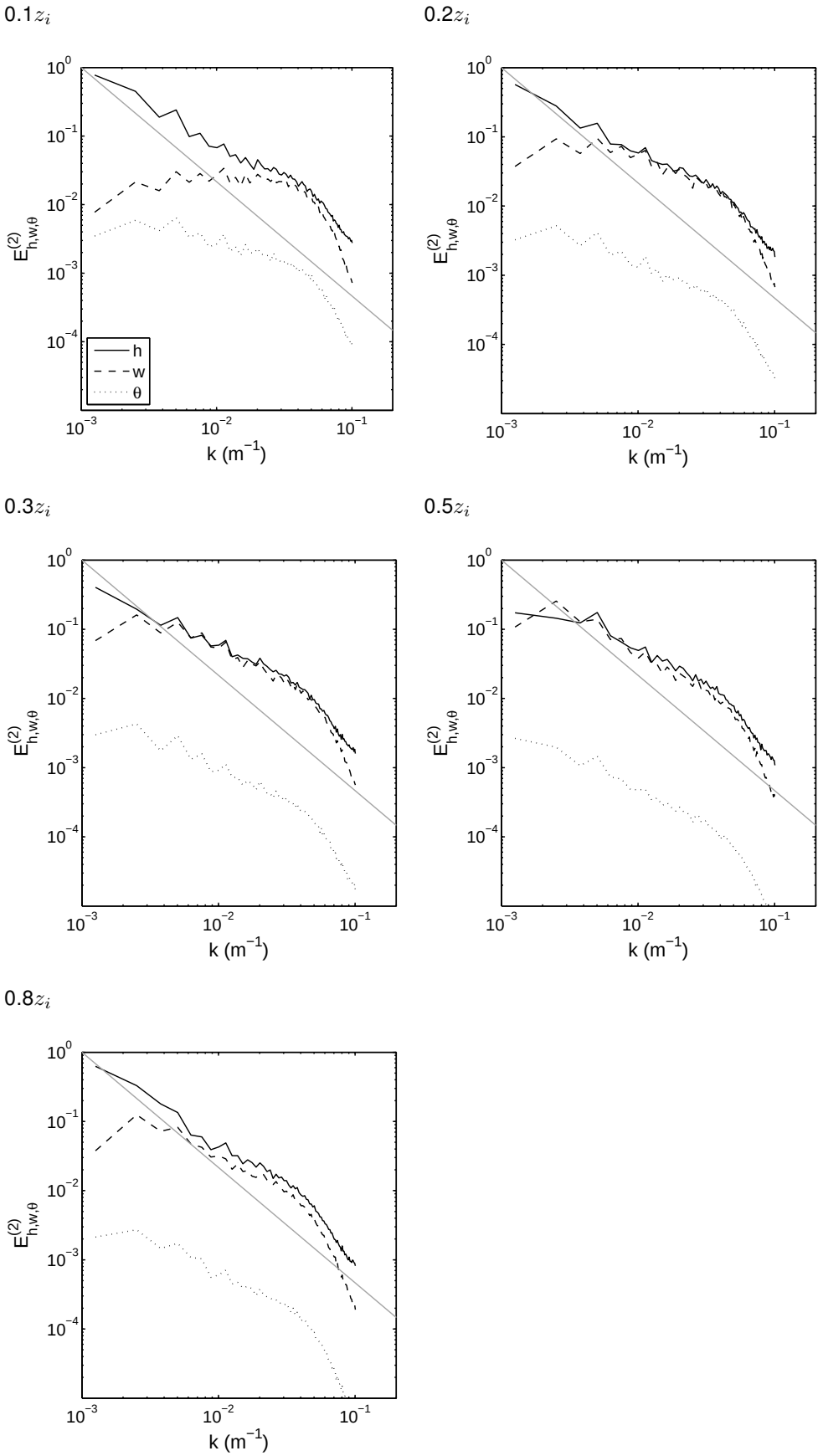


Fig. 10. Two-dimensional spectra of the energy contained in the horizontal velocity fluctuations, h, the vertical velocity fluctuations, w, and the virtual potential temperature fluctuations, θ, taken at various heights in x-y planes from simulation B.

alization well illustrates the roll cells present in this simulation. The red isosurfaces represent vertical velocity fluctuations, $w' = w_* = 0.88$ m/s. The location of these isosurfaces corresponds to the upwelling motion caused by two adjacent roll cells. Below the isosurfaces is a contour plot of temperature. Darker colors correspond to cooler temperatures, and lighter colors correspond to warmer temperatures. There is a clear correlation between regions of warmer temperature and positive vertical velocity fluctuations.

5.6 *THE HIGH ACCURACY ZONE*

Brasseur and Wei (2010) discuss the systematic overshoot in

$$\Phi_m(z) = \kappa \frac{z}{u_*} \frac{\partial U}{\partial z}, \tag{23}$$

which commonly occurs in LESs of turbulent boundary layers (here κ is the von Kármán constant). In order to capture law-of-the-wall (LOTW) scaling, this overshoot must be removed. Brasseur and Wei state that the "negative consequences of the overshoot — spurious stream-wise coherence, large-eddy structure and vertical transport — are enhanced by buoyancy." They also provide three criteria, in terms of nondimensional parameters, that are necessary to remove the overshoot and provide $\Phi_m(z)$ that is constant through the surface layer. The three criteria and their associated nondimensional variables form a parameter space that can be used to move LES into a "high-accuracy zone" in which LOTW is captured.

The first criterion is that the resolved stress maintains dominance over the sub-grid-scale stress at the first grid level, which can be stated mathematically as

$$\Re = \frac{T_R(z_1)}{T_S(z_1)} = \frac{\sqrt{\langle u'w' \rangle^2 + \langle v'w' \rangle^2}}{\sqrt{R_{13}^2 + R_{23}^2}} \sim O(1), \tag{24}$$

where $\Re$ is the ratio of resolved to sub-grid-scale stresses at the first grid level, $T_R(z_1)$ and $T_S(z_1)$ are the resolved and sub-grid-scale horizontal shear stresses, respectively, at the first grid level; $\langle u'w' \rangle$ and $\langle v'w' \rangle$ are the resolved horizontal momentum fluxes at the first grid level; and R_{13} and R_{23} are the sub-grid-scale momentum fluxes at the first grid level. The second criterion is that inertia in the discretized LES governing equation dominates friction caused by the sub-grid-scale viscosity (and by the numerical algorithm, which is impossible to completely quantify). This is restated mathematically as

$$\mathrm{Re}_{LES} = \frac{z_i \, u_*}{\nu^{SGS}(z_1)} > \mathrm{Re}^*_{LES}, \tag{25}$$

where Re_{LES} and Re^*_{LES} are the actual and critical LES Reynolds numbers, respectively. The third criterion is that the number of grid levels within the boundary layer provide a minimum vertical resolution, which is restated as $N_{z_i} > N^*_{z_i}$ where N_{z_i} and $N^*_{z_i}$ are the actual and critical number of grid levels within the boundary layer, respectively. Brasseur and Wei state that in performing LESs of the neutral ABL with the Smagorinsky eddy viscosity closure, reasonable estimates for $\Re^*$, Re_{LES}, and $N^*_{z_i}$ are 1, 350, and 45-50, respectively. Moving an LES into the high-accuracy zone by satisfaction of these criteria can be achieved by adjusting the sub-grid-scale closure constant and vertical resolution of the grid.

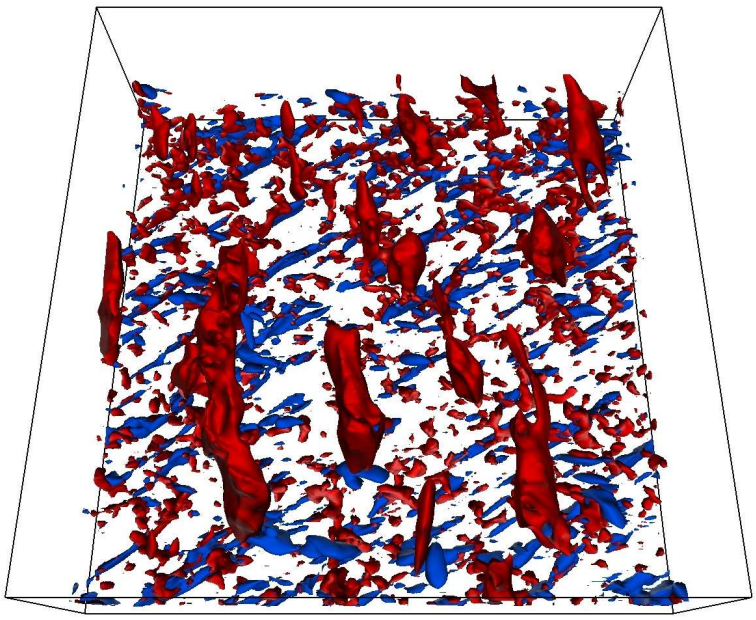
The subject of this section is to show how far the current simulations are from the high-accuracy zone. Because Brasseur and Wei (2010) only focused on the neutral case, we do the same here. Because the ABL continues to develop past the final time that MS94 simulated, we ran our simulation S until it reached a state of better quasi-equilibrium, which is around 46 000 s. Figure 12 is a plot of the Φ_m function (using $\kappa = 0.4$), and it shows that our simulation contains a prominent overshoot in the profile at $z/z_i = 0.06$; the simulation is clearly not in the high-accuracy zone. For this simulation, $\Re = 0.09$, $\mathrm{Re}_{LES} = 100$, and $N_{z_i} = 45$. The only criterion that has been met is the one for vertical grid resolution.

In these simulations, we use a Smagorinsky constant of $C_s = 0.168$, so it can be reduced to move toward the high-accuracy zone. It will be interesting to see how much the friction resulting from the finite-volume formulation (as opposed to a less dissipative pseudo-spectral scheme) contributes to the overshoot and failure to capture LOTW scaling.

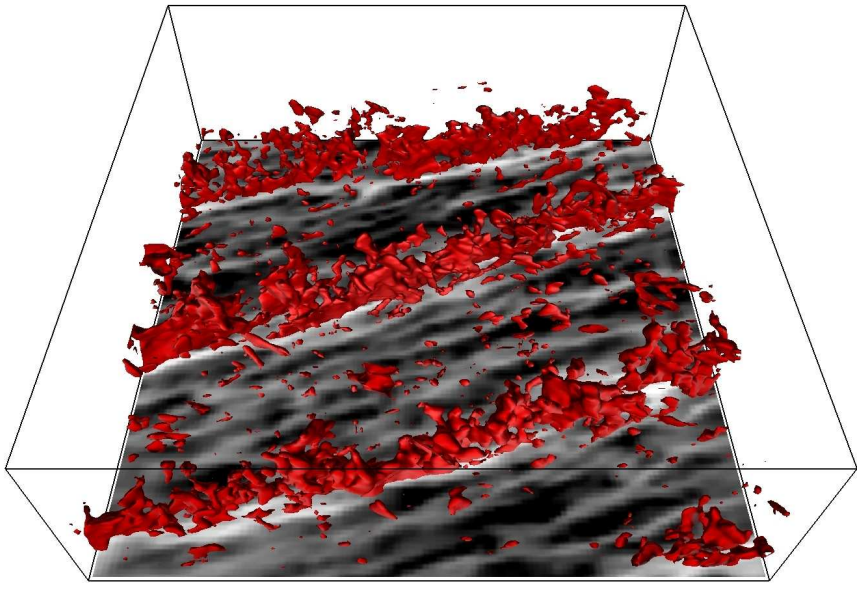
6. CONCLUSIONS AND FUTURE WORK

This work shows that the ABL simulations performed with a second-order accurate, colocated, finite-volume solver compare fairly well with those performed by Moeng and Sullivan (1994) and Khanna and Brasseur (1998) with their pseudo-spectral solvers. However, there are some important differences that must be reconciled in our solver before using it for wind energy applications. One of the most important differences is the fact that the energy spectra from our simulations do not show the -5/3 cascade of energy in the inertial range that is clearly seen in the spectra of Khanna and Brasseur (1998). Also, our shear-driven simulation S does not compare to the corresponding simulation of Moeng and Sullivan (1994) as well as does the more buoyant-driven cases.

This work highlights the fact that there are significant rates of change occurring in vertical profiles of velocity, variance, and fluxes of momentum and variance in the vertical band where turbine rotors lie. It also highlights the large differences between completely shear-driven and shear-buoyancy-driven flow. Wind farms operate in a variety of ABL stability conditions, so it is important to be able to accurately model all of these cases.



(a)



(b)

Fig. 11. (a) Isosurfaces of x-directed velocity fluctuations for $u' = -3u_* = -1.5$ m/s (blue) and of vertical velocity fluctuations for $w' = 0.6$ m/s from the shear-driven case S. (b) Isosurfaces of vertical velocity fluctuations for $w' = w_* = 0.88$ m/s (red) and a contour of temperature θ in which black denotes cooler temperature and white denotes warmer temperature from the shear-buoyancy-driven case SB1.

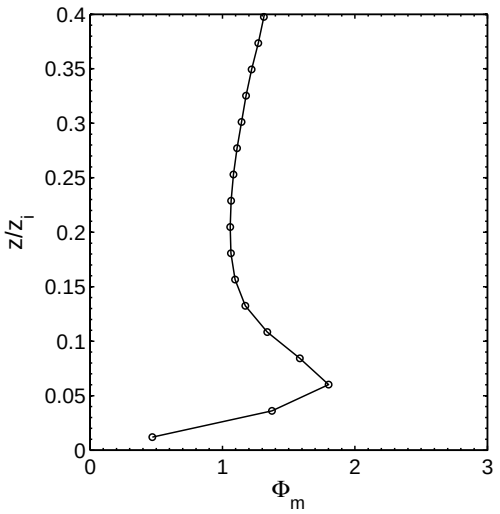


Fig. 12. The non-dimensional velocity gradient function Φ_m from simulation S at time 46 000 s.

Importantly, this work has demonstrated the high-level of flexibility provided by the OpenFOAM framework in creating solvers, boundary conditions, and closures with relative ease compared to modifying an existing code or writing one's own code.

A next step in this work is to incorporate changes to our code to make it more versatile and as similar as possible to the code used by Moeng and Sullivan (1994) such that a better comparison can be made. This means that our code will need to incorporate the one-equation sub-grid-scale closure and surface stress model described in Moeng (1984). Furthermore, validation against high quality field data taken over fairly flat terrain should also be performed. It may also be beneficial to create a solver that uses explicit time advancement; that would make for a simpler algorithm that may be easier to troubleshoot. Such an explicit algorithm is used by Sullivan et al. (1996, 2000) in their colocated pseudo-spectral code.

It is clear that these simulations are far from the "high accuracy zone" described by Brasseur and Wei (2010) and that law-of-the-wall scaling is not being maintained throughout the surface layer as evidenced by the overshoot in the Φ_m parameter observed in these simulations. Work must be done to move these simulations to the high accuracy zone and to investigate how large a role numerical friction plays in keeping the simulation from producing law-of-the-wall scaling. Because this work is being done within a finite-volume setting as it allows more flexibility for wind energy applications (i.e. nonperiodic boundaries), work must be done to understand the grid resolution necessary in the finite-volume setting relative to the pseudo-spectral setting to obtain similar results.

Last, a model for turbines will be added to this solver to perform wind-farm-scale LESs. ABL simulations, like the ones performed here, will be used as precursors to wind farm LESs using this solver. The turbine model under consideration is the actuator line model of Sørensen and Shen (2002).

Acknowledgments. The authors would like to acknowledge Peter Sullivan and Ned Patton of the National Center for Atmospheric Research (NCAR) for their help in resolving solvers problems during the course of this work. Also acknowledged are Branko Kosovic, Bill Mahoney, and Andrezej Wyszogrodzki, of NCAR for their help in determining the important features of an LES solver for this application. Many contributors to the OpenFOAM forum have also helped in answering questions that arose during the creation of the solver and boundary conditions. NREL's High Performance Computing Center provided the compute time necessary to perform these simulations along with help in creating flow field visualizations.

The Alliance for Sustainable Energy, LLC (Alliance), is the manager and operator of the National Renewable Energy Laboratory (NREL). Employees of the Alliance for Sustainable Energy, LLC, under Contract No. DE-AC36-08GO28308 with the U.S. Dept. of Energy, have authored this work. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for United States Government purposes.

References

2009a: *OpenFOAM – The Open Source CFD Toolbox, Programmer's Guide, Version 1.6*. OpenCFD Ltd., 9 Albert Road, Caversham, Reading, Berkshire RG4 7AN, UK.

2009b: *OpenFOAM – The Open Source CFD Toolbox, User's Manual, Version 1.6*. OpenCFD Ltd., 9 Albert Road, Caversham, Reading, Berkshire RG4 7AN, UK.

Brasseur, J. G. and T. Wei, 2010: Designing large-eddy simulation of the turbulent boundary layer to capture law-of-the-wall scaling. *Physics of Fluids*, **22**, 021303, doi:10.1063/1.3319073.

Calaf, M., C. Meneveau, and J. Meyers, 2010: Large eddy simulation study of fully developed wind-turbine array boundary layers. *Physics of Fluids*, **22**, 015110, doi:10.1063/1.3291077.

Conzemius, B., H. Lu, L. Chamorro, Y.-T. Wu, and F. Porte-Agel, 2010: Development and testing of a

wind farm simulator at an operating wind farm. Presentation from AWEA 2010 WindPower Conference and Exhibition, Dallas, Texas.

Cuijpers, J. W. M. and P. G. Duynkerke, 1993: Large eddy simulation of trade wind cumulus clouds. *Journal of the Atmospheric Sciences*, **50**, 3894–3908.

de Villiers, E., 2006: *The Potential of Large Eddy Simulation for Modeling of Wall Bounded Flow*. Ph.D. thesis, Department of Mechanical Engineering, Imperial College of Science, Technology, and Medicine, London, United Kingdom.

Deardorff, J. W., 1972: Numerical investigation of neutral and unstable planetary boundary layer. *Journal of the Atmospheric Sciences*, **29**, 91–115.

Etling, D., 1996: Modelling the vertical abl structure. *Modelling of Atmposheric Flow Fields*, D. P. Lalas and C. F. Ratto, eds., World Scientific Publishing Company, chapter 3, 400–402.

Ferziger, J. H. and M. Perić, 2002: *Computational Methods for Fluid Dynamics*. Springer-Verlag, 3rd, revised edition.

Issa, R. I., 1985: Solution of the implicitly discretized fluid flow equations by operator-splitting. *Journal of Computational Physics*, **62**, 40–65.

Ivanell, S. S. A., 2009: *Numerical Computations of Wind Turbine Wakes*. Ph.D. thesis, Gotland University, Stockholm, Sweden.

Jensen, L. E., 2007: Analysis of array efficiency at Horns Rev and the effect of atmospheric stability. *2007 EWEC Conference, Milan, Italy*.

Johnson, C., 2008: Summary of actual versus predicted wind farm performance - a recap of windpower 2008. Presentation from 2008 AWEA Wind Resource Assessment Workshop, Portland, Oregon.

Katic, I., J. Højstrup, and N. O. Jensen, 1986: A simple model for cluster efficiency. *Proceedings of EWEC'86, Rome, Italy*, W. Palz and E. Sesto, eds., 407–410.

Khanna, S. and J. G. Brasseur, 1998: Three-dimensional buoyancy- and shear-induced structure of the atmospheric boundary layer. *Journal of the Atmospheric Sciences*, **55**, 710–743.

LeMone, M. A., 1973: The structure and dynamics of horizontal roll vortices in the planetary boundary layer. *Journal of the Atmospheric Sciences*, **30**, 1077–1091.

Mason, P. J. and A. R. Brown, 1999: On subgrid models and filter operations in large eddy simulations. *Journal of the Atmospheric Sciences*, **56**, 2101–2114.

Moeng, C.-H., 1984: A large-eddy-simulation model for the study of planetary boundary-layer turbulence. *Journal of the Atmospheric Sciences*, **41**, 2052–2062.

Moeng, C.-H. and P. Sullivan, 1994: A comparison of shear- and buoyancy-driven planetary boundary layer flows. *Journal of the Atmospheric Sciences*, **51**, 999–1022.

Paulson, C. A., 1970: The mathematical representation of wind speed and temperature profiles in the unstable atmospheric surface layer. *Journal of Applied Meteorology*, **9**, 857–861.

Pope, S. B., 2001: *Turbulent Flows*. Cambridge University Press.

Porté-Agel, F., H. Lu, and Y.-T. Wu, 2010: A large-eddy simulation framework for wind energy applications. The Fifth International Symposium on Computational Wind Engineering (CWE2010), Chapel Hill, NC, May 23-27.

Rhie, C. M. and W. L. Chow, 1983: Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA Journal*, **21**, 1525–1532.

Schumann, U., 1975: Subgrid-scale model for finite difference simulation of turbulenct flows in plane channels and annuli. *Journal of Computational Physics*, **18**, 376–404.

Smagorinsky, J., 1963: General circulation experiments with the primitive equations: I. the basic equations. *Monthly Weather Review*, **91**, 99–164.

Sørensen, J. N. and W. Z. Shen, 2002: Numerical modeling of wind turbine wakes. *Journal of Fluids Engineering*, **124**, 393–399.

Stovall, T. D., G. Pawlas, and P. J. Moriarty, 2010: Wind farm wake simualtions in OpenFOAM. AIAA Paper 2010-0825.

Sullivan, P. P., J. C. McWilliams, and C.-H. Moeng, 1996: A grid nesting method for large-eddy simulation of planetary boundary-layer flows. *Boundary Layer Meteorology*, **80**, 167–202.

— 2000: Simulation of turbulent flow over idealized water waves. *Journal of Fluid Mechanics*, **404**, 47–85.

Wharton, S. and J. K. Lundquist, 2010: Atmospheric stability impacts on power curves of tall wind turbines — an analysis of a west coast north american wind farm. Technical report, Lawrence Livermore National Laboratory (LLNL), lLNL-TR-424425, https://e-reports-ext.llnl.gov/pdf/387609.pdf.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
August 2010	Conference Paper	August 2-6, 2010

4. TITLE AND SUBTITLE
Wind Energy-Related Atmospheric Boundary Layer Large-Eddy Simulation Using OpenFOAM: Preprint

5a. CONTRACT NUMBER
DE-AC36-08-GO28308

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
M.J. Churchfield, G. Vijayakumar; J.G. Brasseur, and P.J. Moriarty

5d. PROJECT NUMBER
NREL/CP-500-48905

5e. TASK NUMBER
WE10.3122

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
National Renewable Energy Laboratory
1617 Cole Blvd.
Golden, CO 80401-3393

8. PERFORMING ORGANIZATION REPORT NUMBER
NREL/CP-500-48905

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)
NREL

11. SPONSORING/MONITORING AGENCY REPORT NUMBER

12. DISTRIBUTION AVAILABILITY STATEMENT
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161

13. SUPPLEMENTARY NOTES

14. ABSTRACT *(Maximum 200 Words)*

This paper develops and evaluates the performance of a large-eddy simulation (LES) solver in computing the atmospheric boundary layer (ABL) over flat terrain under a variety of stability conditions, ranging from shear driven (neutral stratification) to moderately convective (unstable stratification).

15. SUBJECT TERMS
atmospheric boundary layer; wind energy; OpenFOAM; large-eddy simulation; wind turbine; ABL; stability conditions

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER (*Include area code*)
Unclassified	Unclassified	Unclassified	UL		

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18