4B.5 Comparison of different implementations of the immersed boundary method in WRF (WRF-IBM)

Jingyi Bao[1]*, Katherine A. Lundquist[2], and Fotini K. Chow [1]
[1] University of California, Berkeley, CA
[2] Lawrence Livermore National Laboratory, Livermore, CA

1 Introduction

Mesoscale models, such as the Weather Research and Forecasting (WRF) model, are increasingly being used for high-resolution simulations. A grid nesting framework, which uses multiple telescoping grids of increasing resolution, provides the ability to pass information across scales. Terrain-following coordinates, used by WRF and other mesoscale atmospheric models, have historically worked well, as fine-scale details of the topography are not resolved on the coarse grid, and the maximum terrain slope is small. As grid resolution increases, fine-scale terrain is represented on the grid, along with increased terrain slopes, posing a challenge to the traditional terrain-following coordinates. In regions of very steep terrain, the native terrain-following coordinates used in the WRF model become highly skewed, resulting in large numerical errors.

An immersed boundary method (IBM), a non-conforming grid technique, was recently implemented into the WRF model [Lundquist et al. (2010, 2012)]. The method alleviates numerical errors associated with steep terrain slopes, and allows simulations over arbitrarily complex terrain. The WRF-IBM model uses a non-conforming grid with the terrain boundary immersed within the grid. Boundary conditions are set through interpolation procedures for grid cells intersected by the immersed surface. The WRF-IBM model has been previously tested for flows over urban and mountainous terrain using a no-slip bottom boundary condition in Lundquist et al. (2010, 2012).

The WRF-IBM model provides the framework and ability to produce high-resolution simulations over complex terrain and urban areas using non-conforming coordinates, however, the previously developed no-slip boundary condition is inappropriate for the coarse resolutions generally used by atmospheric models. Here, we extend WRF-IBM to include a boundary condition which parameterizes surface stresses in the unresolved surface layer using Monin-Obukhov (M-O) similarity theory. We initially consider only neutral stability, in which case M-O theory simplifies to the commonly known "log-law", given in equation 1.

$$\frac{U}{u_*} = \frac{1}{k} \ln \frac{z}{z_0} \tag{1}$$

In this equation, U is the wind speed, u_* is the friction velocity, k is the Von Karman constant, z is the height above the surface, and z_0 is the surface roughness length.

Methods for combining the log-law with the immersed boundary method currently exist in the literature [Choi et al. (2007); Chester et al. (2007); Anderson (2013)]. We implemented several of these methods in WRF, following the implementation as closely as possible (though there are slight differences in the implementation). Additionally, we added a method for using the log-law with WRF's native terrain-following coordinate, a boundary condition treatment commonly applied in idealized large eddy simulations (LES). This allows the surface stress to be parameterized with a surface roughness length scale z_0, facilitating the comparison of results from simulations using terrain-following and IBM coordinates.

While most of the IBM methods worked well for flow over flat terrain, by matching the WRF solution with terrain-following coordinates, all of the IBM implementations from the literature introduced inaccuracies over sloped terrain at the resolutions used here. In this work, a new IBM boundary condition is additionally developed, which specifies both a velocity boundary condition and shear stresses in accordance with similarity theory. In initial tests, this new IBM implementation resulted in the correct velocity profiles for flow over sloped terrain. This paper describes implementation of existing IBM methods, as well as the new method in WRF-IBM. Simulation results from both the existing and the newly

* *Corresponding author address*: Jingyi Bao, University of California, Berkeley, Berkeley, CA 94709, email: jingyib@berkeley.edu

developed method are presented for LES with the Smagorinsky turbulence closure over both flat terrain and for an idealized valley test case. This work represents the first steps in implementing and testing a new surface boundary condition in WRF-IBM, which enables simulations over complex, mountainous terrain.

2 Background

2.1 Immersed boundary method

The immersed boundary method was first used to simulate blood flow through the valve of a heart [Peskin (1977)]. The influence of the immersed boundary acts as an additional body force F_b in the momentum equation, given in equation 2.

$$\frac{\partial \tilde{U}}{\partial t} + \tilde{U} \cdot \nabla \tilde{U} = -\alpha \nabla P + \nu_t \nabla^2 \tilde{U} + F_b \qquad (2)$$

The body force is zero when it is far away from the boundary and has a non-zero value when it is close to boundary.

The immersed boundary method used in this work uses the method of direct forcing, as first presented in Iaccarino and Verzicco (2003). With direct forcing, the velocity is directly modified at the cells near the boundary to enforce the boundary condition, eliminating the need for calculation of the body force term in the numerical algorithm. With WRF-IBM, first, we need to identify cells that are cut by the immersed boundary, which are cells that include both fluid and solid nodes. On the staggered grid used by WRF, cut cells must be determined for each variable for which a boundary condition will be imposed. Then, an interpolation method is used to determine the forcing needed to impose the desired boundary condition. Several different interpolation methods have been employed, ranging from linear interpolation to inverse distance weighting schemes [Iaccarino and Verzicco (2003)]. Finally, the value of the variable at points near the immersed boundary are set, which enforces the desired boundary condition. Further details of the IBM implemenation in WRF are given in Lundquist et al. (2010, 2012).

2.2 WRF's boundary condition

In WRF, surface stresses are set using equation 3 and equation 4.

$$\tau_{sfc_{xz}} = C_d \left|U\right| U \qquad (3)$$

$$\tau_{sfc_{yz}} = C_d \left|U\right| V \qquad (4)$$

In this equation, $(\tau_{sfc_{xz}}, \tau_{sfc_{yz}})$ are the surface shear stresses in the x and y directions, C_d is the drag coefficient, and $|U|$ denotes the wind speed. There are several ways to calculate C_d in WRF. It can be specified directly as an input in the namelist, or M-O theory can be used to calculate u_* in the land surface model in WRF. Additionally, we added a new equilibrium stress model option that calculates the drag coefficient based on a specified surface roughness parameter z_0 and the log law. In this formulation, the coefficient of drag is given by equation 5.

$$C_d = \left(\frac{k}{\ln \frac{z}{z_0}} \right)^2 \qquad (5)$$

Surface stresses are imposed in the vertical direction in native WRF coordinates, rather than in the surface normal direction. In addition to setting the shear stress at the surface, WRF also implements a kinematic boundary condition for the velocity. The set of equations used for the kinematic boundary condition are given by (6) and (7), where $\dot{\eta}$ is the velocity of the vertical coordinate, h is the terrain height, and $(u, v, w)_{surf}$ are the Cartesian components of velocity at the surface.

$$\dot{\eta} = 0 \qquad (6)$$

$$w_{surf} = u_{surf} \frac{\delta h}{\delta x} + v_{surf} \frac{\delta h}{\delta y} \qquad (7)$$

This set of equations specifies no flow in the surface normal direction.

3 IBM boundary condition implementations

Four IBM methods are implemented in WRF here. These include three existing IBM methods from the literature that use the log law in combination with IBM [Choi et al. (2007), Chester et al. (2007), and Anderson (2013)], as well as our new IBM method. The first method is the velocity reconstruction method in Choi et al. (2007), where the velocity is reconstructed at the first fluid node assuming there is a logarithmic profile near surface. The second method is the shear stress reconstruction method in Chester et al. (2007). In this method, several layers of shear stress near the immersed boundary are modified according to the log law. Nodes internal to the terrain are set to have zero ve-

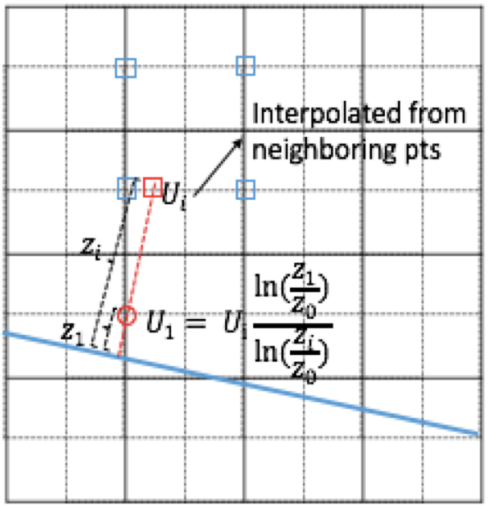

Figure 1: Velocity reconstruction IBM

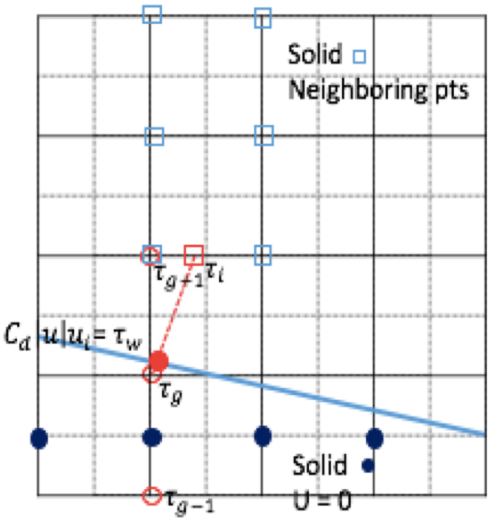

Figure 2: Shear stress reconstruction IBM

locity. The third method, which is called the canopy method, is based on the canopy stress model in Anderson (2013). A body drag force is imposed as a momentum sink at cells near the surface. Internal nodes are also set to have zero velocity. In the fourth, new method, the log law is enforced by setting boundary conditions for both velocity and stress. Velocity is set at a ghost point just underneath the terrain, while shear stress is reconstructed at multiple points above and below the terrain. We test all four methods by simulating flow over flat terrain and an idealized V-shaped valley, and comparing the results to those using WRF's native terrain-following coordinate.

3.1 Velocity reconstruction method

In this method, the velocity is reconstructed at the first fluid node outside of the immersed surface, as represented by the open circle in figure 1. Fadlun et al. (2000) used linear interpolation between a point in the fluid and a no-slip boundary condition at the surface to reconstruct the velocity at a layer of fluid points outside of the immersed surface.

Choi et al. (2007) suggested a similar method where the log-law is used to reconstruct the velocity at the first fluid node. Assuming that multiple nodes reside within the logarithmic layer, and that u_* is constant within this region, equation 8 can be used to calculate the velocity at the first fluid node based on the velocity at the second fluid node above the surface.

$$U_1 = U_2 \frac{\ln \frac{z_1}{z_0}}{\ln \frac{z_2}{z_0}} \tag{8}$$

Figure 1 illustrates the method as implemented in WRF-IBM. Here, velocity is reconstructed at the first fluid node (U_1) above the immersed boundary for each cut cell. An interpolation point (U_i) is found based on projecting the surface normal vector outward until it intersects a cell containing only fluid nodes, so that U_1 and U_i are normal to the immersed boundary. This scheme then uses equation 8 to calculate the value of U_1 at each cut cell, where U_2 and z_2 take values from the interpolated point.

3.2 Shear stress reconstruction method

The shear stress reconstruction method of Chester et al. (2007) is a scheme where shear stress is reconstructed at multiple grid points in the vicinity of the immersed boundary. Shear stress values on the immersed surface are calculated according to the log-law, using equation 3 and equation 4. Shear stress values within a defined distance, external to the immersed boundary, are set to the surface stress value, while stress values interior to the immersed boundary are extrapolated using linear interpolation from the surface value and a value outside of the band of nodes being reconstructed. Velocity is set to zero interior to the immersed boundary, and is not treated outside of the immersed boundary.

The shear stress reconstruction method, as implemented in WRF-IBM, is illustrated in figure 2.

Surface stresses are reconstructed at three layers (open red circles in figure 2), which includes a ghost point just below the terrain (labeled τ_g), the point below the ghost point (labeled τ_{g-1}), and the point above the ghost point (labeled τ_{g+1}). The surface stress is found by equation 3 and equation 4 in combination with equation 5. For each shear stress reconstruction point, an interpolating point (τ_i) is found based on projecting the surface normal outward until it intersects a cell containing only fluid nodes. An interpolation method is used to calculate the value of τ_i. This method then uses linear extrapolation to set τ_g. The values of τ_{g-1} and τ_{g+1} are calculated using the same method, but an independent image location is found for each point (not shown in figure 2). As in the method of Chester et al. (2007), the velocity is set to zero at nodes internal to the terrain, and not treated outside of the terrain. These points are indicated by solid blue points in figure 2.

3.3 Canopy method

In Anderson (2013), the immersed boundary method includes a canopy model to simulate flow over obstacles. In this method, nodes internal to the immersed boundary are set to have zero velocity. A canopy stress model is used to impose a momentum sink for cells near the surface. Cells where the stress model is applied are determined based on surface geometry, and can include single or multiple points, interior and exterior to the immersed surface. The momentum sink is a body force added to the equation for conservation of momentum, and is given by equation 9.

$$f(x,t) = -\frac{1}{2} C_d A(x) \, |U| \, U \tag{9}$$

Here, $A(x)$ is the frontal (impinging) area of the immersed surface. In the streamwise direction, the frontal area is equal to $\Delta x_2 [\frac{\delta h}{\delta x_1}] \Delta x_1$, where x_1 and x_2 represent the streamwise and transverse directions, grid spacing is represented by Δx_1 and Δx_2, and h is the height of the immersed terrain. Similarly, $\Delta x_1 [\frac{\delta h}{\delta x_2}] \Delta x_2$ is the frontal area for flow in the transverse direction. When the force in equation 9 is divided by cell volume ($\Delta x_1 \Delta x_2 \Delta x_3$), equation 10 results, which can be used directly in the equation for conservation of momentum.

$$\frac{f(x,t)}{\Delta x_1 \Delta x_2 \Delta x_3} = -\frac{1}{2} C_d \frac{U}{\Delta x_3} (\vec{U} \cdot \nabla h) \tag{10}$$

Figure 3 illustrates the canopy method as implemented in WRF-IBM. Velocity at interior nodes, represented by the closed red circle labeled U_g are set to zero. The body force is added at the point labeled U_1, but can also be applied at additional points, dependent on the terrain geometry. The algorithm determining points at which the body forces are applied is identical to that presented in Anderson (2013).

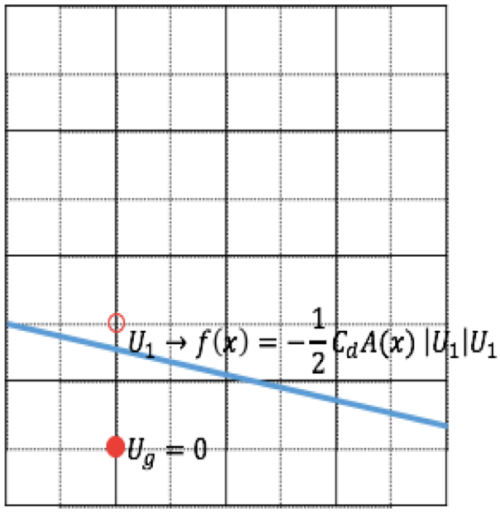

Figure 3: Canopy method IBM

3.4 Ghost-cell velocity and shear stress reconstruction

While several IBM implementations using the log-law have been proposed in the literature, and implemented here, it was found that they could not recreate the WRF solution for flow over sloping terrain. A new IBM implementation, capable of matching the WRF solution for flat and sloped terrain, is developed here. Of the IBM implementations presented here, this method is most similar to WRF's boundary condition, as it requires both reconstruction of the velocity and stress fields in the vicinity of the immersed boundary. As mentioned in section 2.2, WRF implements it's boundary condition by setting the shear stress according to M-O theory, as well as a kinematic boundary condition for velocity.

The new IBM algorithm is illustrated in figure 4, where the treatment of the velocity field is shown in the left panel. Like the velocity reconstruction methods of Choi et al. (2007), we use the log-law (for neutral stratification) to enforce a logarithmic velocity profile near surface. However, unlike the existing method, where velocity is reconstructed at a fluid node, velocity in the new method is reconstructed at a ghost point interior to the terrain. First, ghost cells

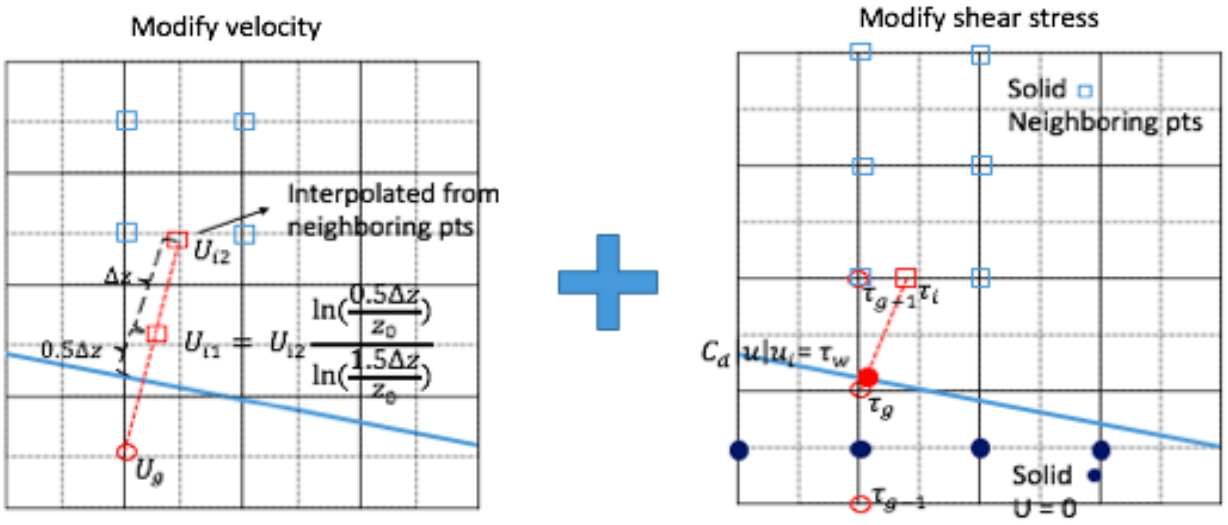

Figure 4: New ghost-cell velocity and shear stress reconstruction IBM algorithm. The algorithm for velocity is depicted in the left panel, and the shear stress algorithm is depicted in the right panel.

(U_g) are identified below the immersed boundary at each cut cell. Then two interpolation points (U_{i1} and U_{i2}) are found based on projecting the surface normal vector outward from U_g a distance of $0.5\Delta z$ and $1.5\Delta z$ from the immersed surface. All three points U_g, U_{i1}, and U_{i2} are aligned in the surface normal direction to the immersed boundary. Equation 8 is used to calculate the value of U_{i1}, where the value of U_{i2} is calculated via interpolation. Linear extrapolation, using the values of U_{i1} and U_{i2}, is used to calculate U_g.

The right panel of figure 4 illustrates the shear stress reconstruction method implemented in this WRF-IBM method. Three layers of shear stress are reconstructed near the immersed boundary, identical to the shear stress reconstruction method described above in section 3.2.

4 Validation

In this section, we examine the implementation of the four log-law boundary conditions at the immersed boundary using two test cases. The first is a pressure-driven neutral atmospheric boundary layer over flat terrain, and the second is an idealized V-shaped valley. We additionally present results using terrain-following coordinates. The WRF model is run in an idealized mode, such that atmospheric processes other than turbulence are not parameterized. This allows us to verify the new surface boundary condition at the immersed boundary without adding the complexity of additional atmospheric processes, although IBM has been coupled to atmospheric parameterizations [Lundquist et al. (2010)].

4.1 Flat terrain simulations

Simulations are carried out for a neutral atmospheric boundary layer over terrain located at a height of 100 m using the four IBM methods and terrain-following coordinates. For each of the four IBM methods, we tested four different grids, shown in figure 5. For each of the four grids, the immersed boundary is held at a constant height of 100 m, while the height of the domain bottom is modified. This leads to the grid aligning with the immersed boundary in multiple configurations. Due to the staggered grid used by WRF, in grid 1 the immersed boundary aligns with w velocity points, and grid 3 aligns with u points, while grids 2 and 4 are aligned half way between staggered nodes. This leads to the horizontal velocity points being various distances above the immersed boundary ranging from 0 to Δz, and tests the robustness of our IBM algorithm, as the different grids allow careful testing of the interpolation method with all possible grid staggering configurations.

In this test case, the domain height is located at 1.5 km, and 71 vertical grid points are used in the terrain-following case, while 73 vertical points are used in the IBM cases to allow for points below the terrain. In both setups a constant vertical grid spacing of $\Delta z = 20$ m is used. Although the Smagorinsky turbulence closure is used in this case, the

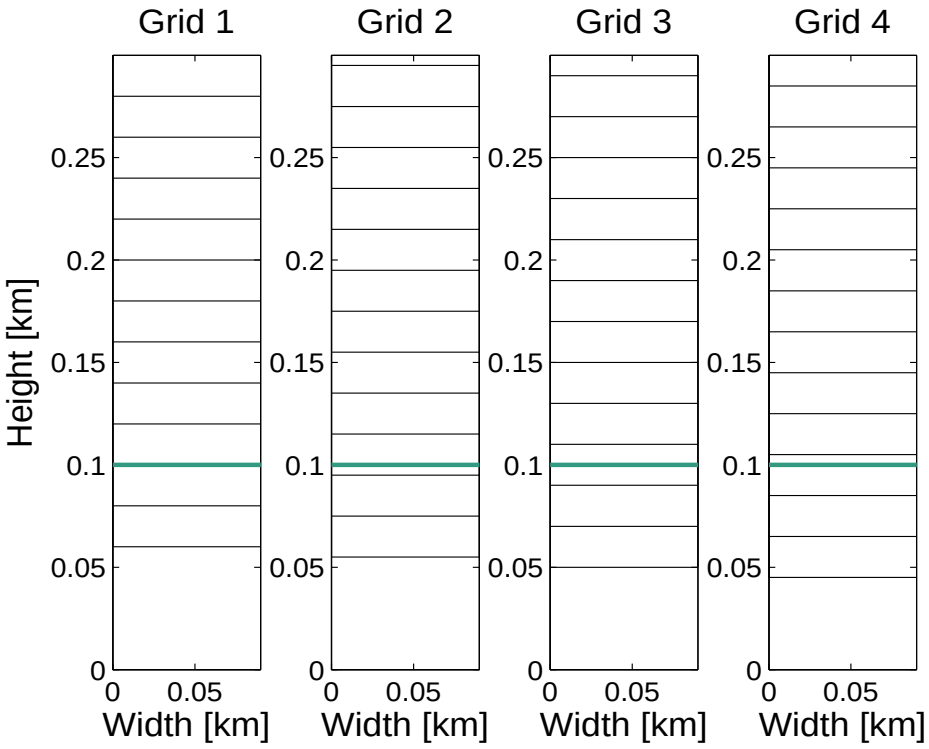

Figure 5: Four different grid setups for the IBM flat terrain case

initialization is not seeded with perturbations, and therefore the solution remains perfectly horizontally homogeneous (i.e. turbulence is not resolved). The grid spacing in the horizontal is $(\Delta x, \Delta y) = (90, 90)$ m, and the simulation uses three grid points in each horizontal direction. The algorithm has been tested with a larger number of horizontal grid points, however, as the solution is horizontally homogeneous, the solution is independent of the number of grid points used in the horizontal.

All cases are initialized with a neutral and dry sounding with a 10 m/s wind in the x-direction. The surface is set to have a constant surface roughness of $z_0 = 0.1$ m, and a pressure gradient drives the flow in the x-direction only. Periodic boundary conditions are used for the lateral boundaries, and the total integration time for this case is 4 hours.

Figure 6 shows instantaneous u velocity profiles at $T = 4$ hours for terrain-following WRF and the four IBM implementations. Simulations are performed for each IBM method on the four different grid setups, but we only show the results from grid 2 in figure 6, as results from all grids appear similar. The velocity profiles for the WRF and WRF-IBM simulations compare well for all methods except the canopy method for all four grid setups.

4.2 Idealized valley simulations

Next, the WRF-IBM boundary conditions are tested for an idealized V-shaped valley case. The terrain and grid are shown in figure 7. The valley terrain is defined using the following equation $h_t = \mathit{offset} + \mathit{slope} * |x_{dist}|$, where the bottom of the valley is $\mathit{offset} = 100$ m, the valley slope is $\mathit{slope} = \frac{2}{9}$, which creates the 12 degree angle of the valley walls , and x_{dist} is the distance from the center of the valley.

The hill slope is relatively small (12 degrees), so that the simulation results from WRF are still reasonable using terrain-following coordinates. With such a mild slope, and the chosen grid resolution, the errors due to the terrain-following coordinate should be small, thus allowing use of the WRF results for validation of the IBM boundary conditions.

In this idealized V-shaped valley test case, the domain height is located at 1.5 km, and 71 vertical grid points are used for the terrain-following grid, while 76 vertical points are used for the IBM grid to allow for points below the terrain. In both setups a constant vertical grid spacing of $\Delta z = 20$ m is used. The grid spacing in the horizontal is $(\Delta x, \Delta y) = (90, 90)$ m, and the simulation uses 20 grid points in each horizontal direction.

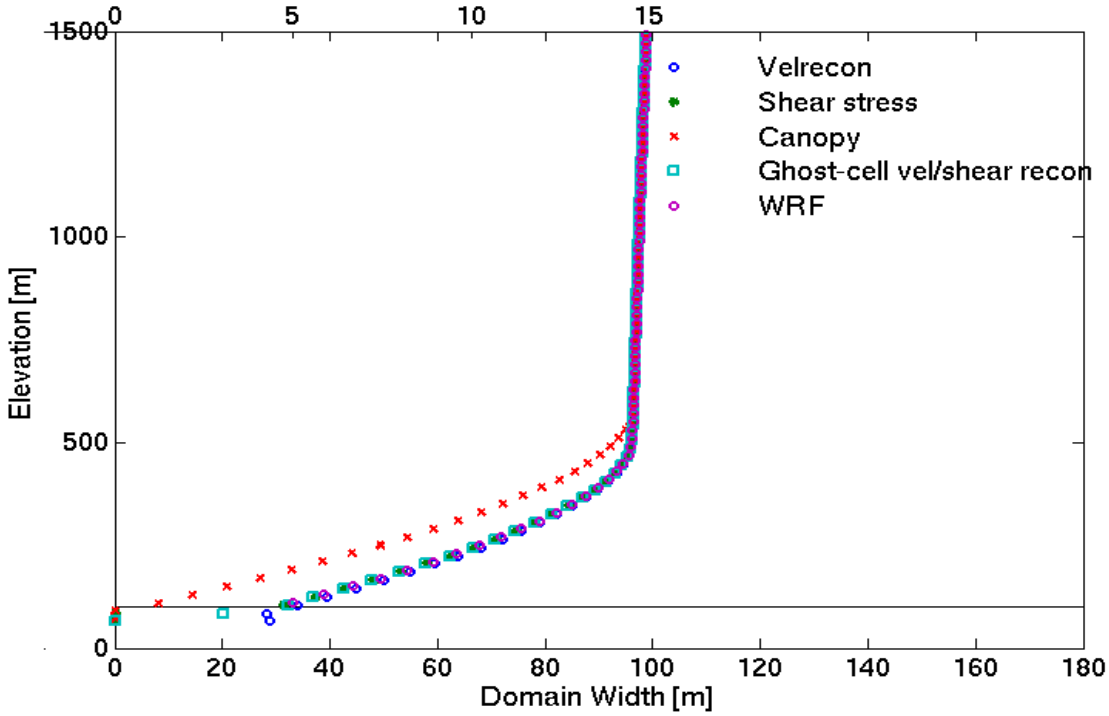

Figure 6: Flat terrain results: u velocity profiles for WRF-IBM .vs. WRF at $T = 4$ hours

As with the flat terrain case, the valley case is initialized with a neutral, dry sounding with a 10 m/s wind in the x-direction. The surface is set to have a constant surface roughness of $z_0 = 0.1$ m, and a pressure gradient drives the flow in the x-direction only. The Smagorinsky turbulence closure is used, and periodic boundary conditions are used for the lateral boundaries. The total integration time for this case is 4 hours.

Figure 8 shows the instantaneous profiles of u velocity at $T = 4$ h for several different locations along the span of the valley for WRF and WRF-IBM with the velocity reconstruction method [Choi et al. (2007)]. Although the Smagorinsky turbulence closure is used, turbulence remains unresolved at $T = 4$ hours, and therefore we are able to compare the WRF and WRF-IBM solutions directly. The surface velocity for the WRF-IBM velocity reconstruction method is too small compared to the solution using terrain-following coordinates. Figure 9 and figure 10 show the same profiles of u velocity for WRF-IBM using the shear stress reconstruction method [Chester et al. (2007)] and the canopy method [Anderson (2013)]. The near-surface velocity is also too small for both of these two methods compared to WRF.

In both the shear stress reconstruction method and the canopy method, velocities for internal nodes below the immersed boundary are all set to zero. During our test cases, we noticed that these zero velocity internal points play a important role in creating smaller surface velocities in figures 9 and 10. Setting velocities for the internal nodes to be zero is commonly used in many traditional IBM methods, as theoretically, the velocities inside a solid obstacle are zero. In our numerical implementations, we found that setting zero velocity at internal nodes is not an appropriate choice, as it affects the surface velocity. In our new ghost-cell velocity and shear stress reconstruction method, instead of setting the velocity at internal points directly to zero, we modified the velocity at ghost points based on the log-law.

Figure 11 shows the instantaneous u velocity profile at $T = 4$ h for the same locations along the span of the valley for WRF and WRF-IBM with the newly developed ghost-cell velocity and shear stress reconstruction method described in section 3.4. The velocity profiles for the WRF-IBM model with this new IBM method match well with the native WRF result for this idealized valley case.

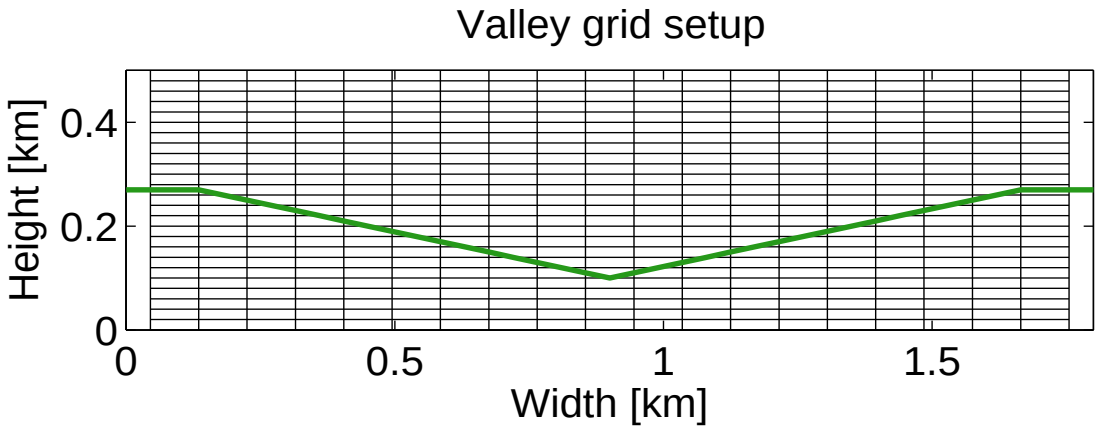

Figure 7: Valley grid setup for WRF-IBM

5 Summary and conclusion

The results above indicate that the velocity reconstruction method [Choi et al. (2007)] and the shear stress reconstruction method [Chester et al. (2007)] perform well over flat terrain, but they produce velocities that are too small near the surface over the idealized valley (figures 8 and 9) when compared to the reference solution which uses terrain-following coordinates. The canopy method [Anderson (2013)] produces smaller velocities near the surface for both flow over flat terrain (figure 6) and over the idealized valley (figure 10) in comparison to the reference solution. In contrast, the newly proposed IBM implementation in section 3.4, is able to reproduce the results of WRF using terrain-following coordinates as seen in figures 6 and 11 for both flat and mildly complex terrain. The newly developed ghost-cell velocity and shear stress method still requires further testing. Additional test cases will include flow over steeper idealized hills and flow over real topography, as well as non-neutral atmospheric stability conditions.

References

Anderson, W. (2013). An immersed boundary method wall model for high-Reynolds-number channel flow over complex topography. *International Journal for Numerical Methods in Fluids*, 71(12):1588–1608.

Chester, S., Meneveau, C., and Parlange, M. B. (2007). Modeling turbulent flow over fractal trees with renormalized numerical simulation. *Journal of Computational Physics*, 225(1):427–448.

Choi, J.-I., Oberoi, R. C., Edwards, J. R., and Rosati, J. A. (2007). An Immersed Boundary Method for Complex Incompressible Flows. *J. Comput. Phys.*, 224(2):757–784.

Fadlun, E. A., Verzicco, R., Orlandi, P., and Mohd-Yusof, J. (2000). Combined Immersed-Boundary

Finite-Difference Methods for Three-Dimensional Complex Flow Simulations. *Journal of Computational Physics*, 161:35–60.

Iaccarino, G. and Verzicco, R. (2003). Immersed boundary technique for turbulent flow simulations. *Applied Mechanics Reviews*, 56(3):331–347.

Lundquist, K. A., Chow, F. K., and Lundquist, J. K. (2010). An Immersed Boundary Method for the Weather Research and Forecasting Model. *Monthly Weather Review*, 138(3):796–817.

Lundquist, K. A., Chow, F. K., and Lundquist, J. K. (2012). An Immersed Boundary Method Enabling Large-Eddy Simulations of Flow over Complex Terrain in the WRF Model. *Monthly Weather Review*, 140(12):3936–3955.

Peskin, C. S. (1977). Numerical analysis of blood flow in the heart. *Journal of Computational Physics*, 25(3):220–252.

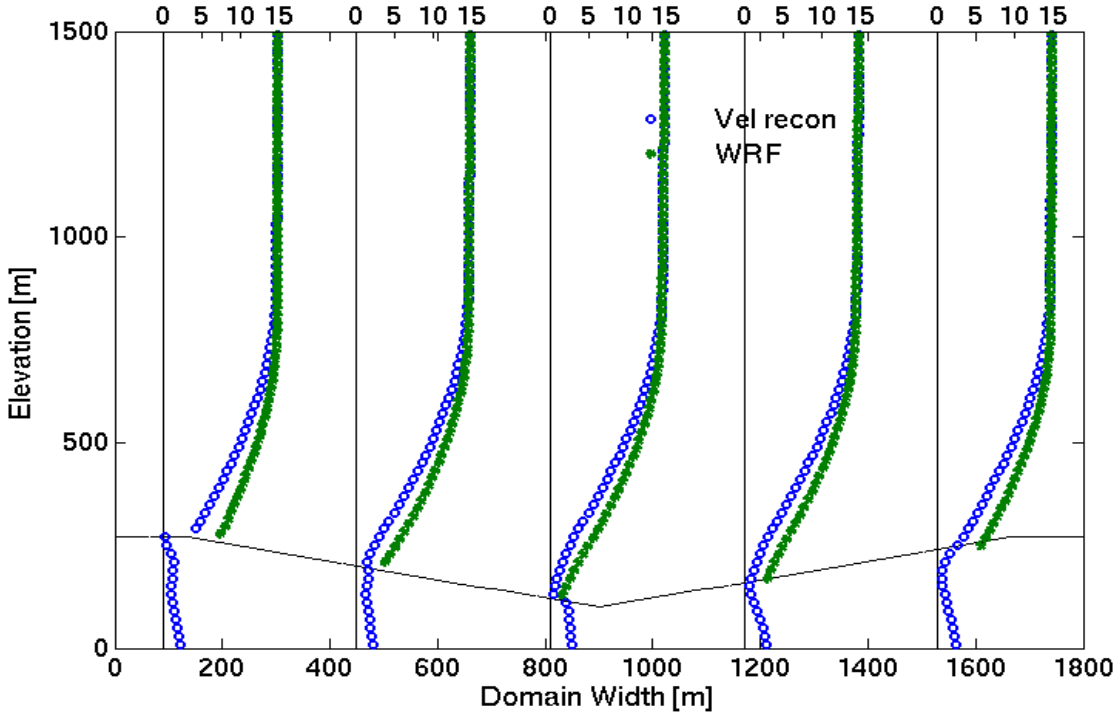

Figure 8: Idealized valley results: u velocity profiles for WRF-IBM (velocity reconstruction method) .vs. WRF at $T = 4$ hours

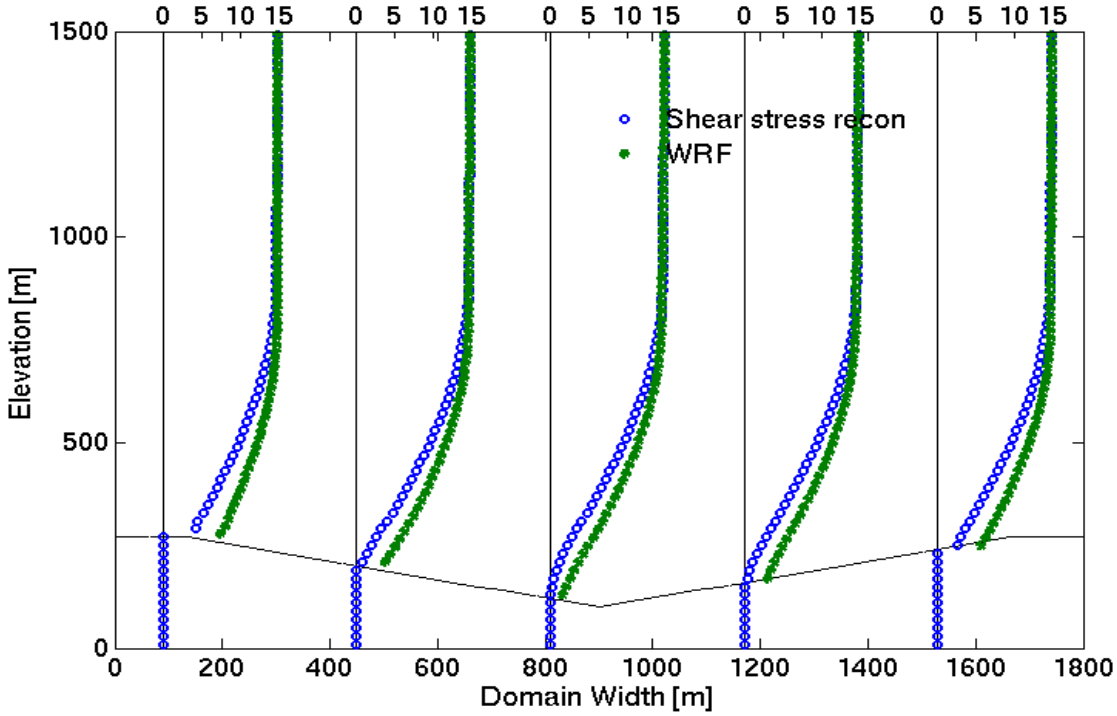

Figure 9: Idealized valley results: u velocity profiles for WRF-IBM (shear stress reconstruction method) .vs. WRF at $T = 4$ hours

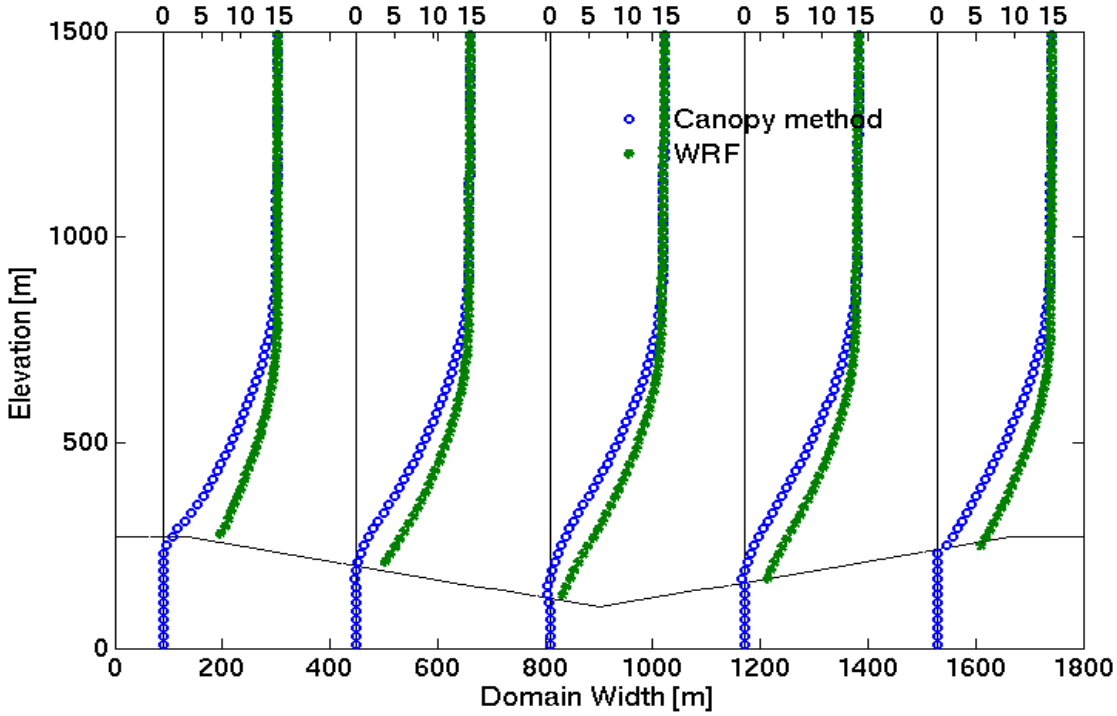

Figure 10: Idealized valley results: u velocity profiles for WRF-IBM (canopy method) .vs. WRF at $T = 4$ hours

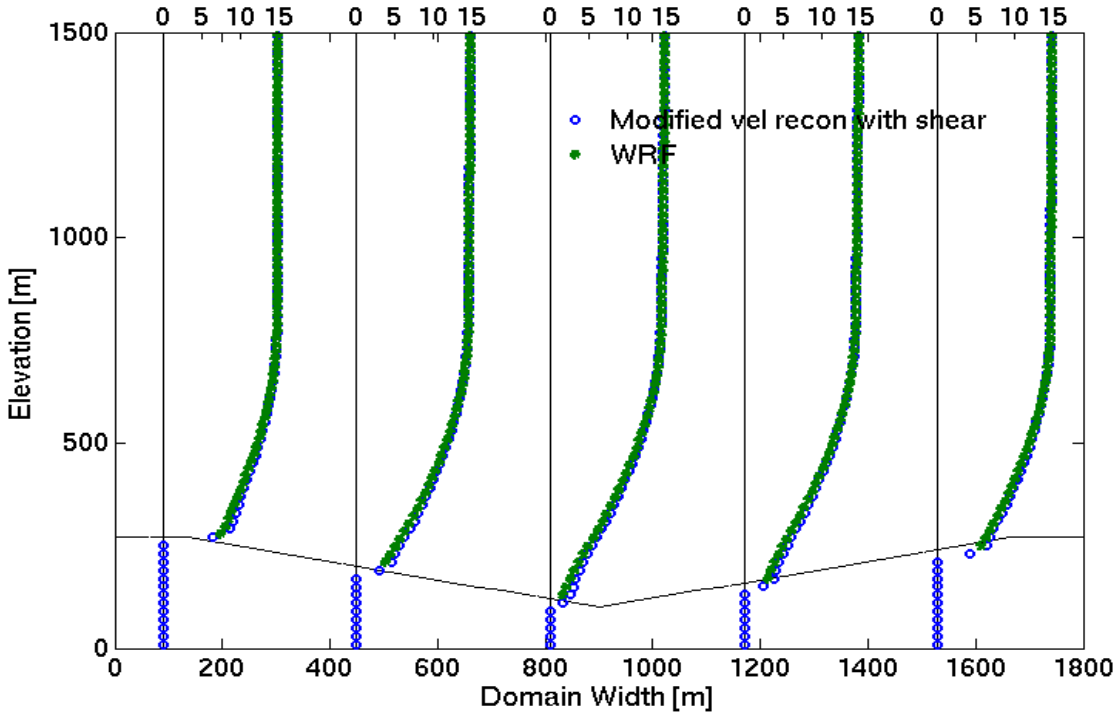

Figure 11: Idealized valley results: u velocity profiles for WRF-IBM (ghost-cell velocity and shear stress reconstruction) .vs. WRF at $T = 4$ hours