

# An Immersed Boundary Method for the Weather Research and Forecasting Model

KATHERINE A. LUNDQUIST AND FOTINI KATOPODES CHOW

*University of California, Berkeley, Berkeley, California*

JULIE K. LUNDQUIST

*Lawrence Livermore National Laboratory, Livermore, California*

(Manuscript received 4 March 2009, in final form 3 September 2009)

## ABSTRACT

This paper describes an immersed boundary method that facilitates the explicit resolution of complex terrain within the Weather Research and Forecasting (WRF) model. Mesoscale models, such as WRF, are increasingly used for high-resolution simulations, particularly in complex terrain, but errors associated with terrain-following coordinates degrade the accuracy of the solution. The use of an alternative-gridding technique, known as an immersed boundary method, alleviates coordinate transformation errors and eliminates restrictions on terrain slope that currently limit mesoscale models to slowly varying terrain. Simulations are presented for canonical cases with shallow terrain slopes, and comparisons between simulations with the native terrain-following coordinates and those using the immersed boundary method show excellent agreement. Validation cases demonstrate the ability of the immersed boundary method to handle both Dirichlet and Neumann boundary conditions. Additionally, realistic surface forcing can be provided at the immersed boundary by atmospheric physics parameterizations, which are modified to include the effects of the immersed terrain. Using the immersed boundary method, the WRF model is capable of simulating highly complex terrain, as demonstrated by a simulation of flow over an urban skyline.

## 1. Introduction

Most mesoscale numerical weather prediction (NWP) models use terrain-following coordinates, which accommodate complex terrain by transforming the physical domain onto a Cartesian grid. Phillips (1957) first introduced this coordinate, using the variable sigma to represent the transformed vertical coordinate. This formulation simplifies the application of lower boundary conditions by aligning the lowest coordinate with the topography. Coordinate lines gradually become smoother and flatter with distance from the ground but retain a signature of the underlying surface shape throughout the entire height of the domain. Metric terms from the coordinate transformation are introduced into the governing equations, and when discretized these terms produce additional numerical errors (Janjic 1977; Klemp et al. 2003). As mesoscale models are increasingly being used for high-

resolution flows over complex terrain, these coordinate transformation errors can significantly degrade the quality of the numerical solution.

Several methods have been proposed to increase the fidelity of simulations with complex terrain, ranging from methods that reduce errors arising from coordinate transformations to methods that eliminate the need for coordinate transformations. Mahrer (1984) noted that the standard computational stencil used to calculate horizontal gradients is inappropriate when the vertical grid spacing is less than the elevation change over the horizontal dimension of the cell. In this case, the accuracy of horizontal gradients can be improved by modifying the stencil to include nodes that more closely follow a Cartesian grid. Schär et al. (2002) proposed a modified sigma coordinate in which grid distortion resulting from small-scale terrain features decays with height more rapidly than distortion caused by large-scale terrain features. The modified coordinate flattens quickly with height and improves the accuracy of the solution by reducing grid distortion aloft. As terrain slopes approach the vertical limit, coordinate transformation errors grow, making reductions of these errors increasingly difficult.

---

*Corresponding author address:* Katherine A. Lundquist, Mechanical Engineering, 6189 Etcheverry Hall, University of California, Berkeley, Berkeley, CA 94720-1742.  
E-mail: katielundquist@berkeley.edu

In environments with very steep slopes, such as urban or mountainous environments, it is desirable to completely eliminate the coordinate transformation. One approach is to represent topography by fitting it to a Cartesian grid, thereby creating a step approximation of the boundary. Large errors have been documented when a zeroth-order terrain approximation is used to model topographically induced circulations (Fast 2003), whereas higher-order boundary representations alleviate these errors by eliminating the sharp corners (Adcroft et al. 1997). Structured or unstructured body-fitted grids are often employed for complex geometries because these methods are capable of higher-order boundary representations. Hanna et al. (2006) present simulations of downtown Manhattan with five different urban computational fluid dynamics (CFD) models, each using conforming grids. Although imposing boundary conditions on these grids is straightforward, there are drawbacks. Body-fitted coordinates require time-consuming manual manipulation to ensure that the grid conforms to the boundaries, while minimizing grid skewness. Unstructured grids produce an irregular data structure, which increases computational cost. In addition to these gridding issues, CFD models are traditionally forced at lateral boundaries with idealized flow, neglecting dynamic forcing resulting from synoptic-scale weather patterns. Furthermore, CFD codes do not generally include options for representing atmospheric processes such as surface fluxes of heat and moisture.

In this work, we introduce an immersed boundary method (IBM) to combine the favorable properties of an NWP code with the ability to handle complex terrain. The IBM enables flows with complex terrain to be simulated on nonconforming grids while retaining the efficiency of a structured solver and the features of an NWP code. Terrain is included by modifying the treatment of nodes near the immersed boundary to impose the effects of the surface. Numerous methods employ this technique and are known by a variety of names, including immersed boundary, embedded boundary, fictitious domain, penalty, and Cartesian-grid methods. The methods differ in the treatment of the nodes near the immersed or embedded boundary. The review articles by Iaccarino and Verzicco (2003) and Mittal and Iaccarino (2005) provide a comprehensive overview of existing methods for nonconforming grids, although they focus on the variants of the immersed boundary method.

The IBM presented here represents a rigid interface at the boundary using direct forcing, as first suggested by Mohd-Yusof (1997). This IBM is uniquely formulated to handle compressible viscous flows and to work with the native isobaric terrain-following coordinate in the

Weather Research and Forecasting (WRF) model. The newly developed algorithm accommodates movement in the vertical pressure coordinate while remaining well conditioned and avoids numerical instabilities noted in other immersed and embedded boundary approaches (Saiki and Biringen 1996; Tseng and Ferziger 2003; Kirkpatrick et al. 2003). Details of the WRF model and the IBM implementation are provided in sections 2 and 3. The IBM is capable of enforcing either Dirichlet or Neumann boundary conditions. A comparison of the native WRF and IBM-WRF boundary conditions is made in section 4, along with the details of a new no-slip option for terrain-following coordinates.

An additional feature, distinguishing the IBM described here, is that it includes coupling to a land surface model, which provides realistic surface forcing. The IBM has been used for atmospheric flows in CFD codes (Tseng et al. 2006; Smolarkiewicz et al. 2007; Shi et al. 2008); however, to the authors' knowledge this is the first IBM that interfaces with atmospheric parameterizations. The modifications required to include the atmospheric physics options are discussed in section 5. By retaining the native isobaric coordinate and coupling the IBM to the land surface model, NWP features such as grid nesting and atmospheric physics function seamlessly with the IBM. Domains where the IBM is used to explicitly resolve complex terrain can be nested into larger mesoscale domains, which use the terrain-following coordinate.

The accuracy of the immersed boundary solver is examined in section 6 by comparing the sigma-coordinate WRF and IBM-WRF solutions for canonical flows. Validation cases include two-dimensional pressure-driven flow over an isolated hill and idealized valley simulations with both specified and parameterized surface fluxes. Results are also presented for flow over two blocks of the New York City skyline, which demonstrate the method's ability to handle extremely complex terrain with sharp corners and vertical terrain gradients.

## 2. Formulation of the numerical solver

### *a. Coordinate definition*

Phillips (1957) first introduced terrain-following coordinates by using pressure as an independent variable representing the vertical coordinate. Laprise (1992) cast the fully compressible Euler equations in this coordinate by first transforming the equation into the hydrostatic pressure coordinate and then applying a second transformation into the terrain-following coordinates. The WRF model is based on this coordinate, which Laprise defines in terms of the dry hydrostatic pressure  $P_{hs}$ . The

coordinate is defined such that it is zero at the top of the computational domain and unity at the terrain surface. This yields the coordinate definition  $\eta = (P_{hs} - P_{hs\_top})/\mu$ , where the column mass per unit area of the fluid is  $\mu(x, y) = P_{hs\_surface} - P_{hs\_top}$ . Note that we refer to the coordinate as “eta” to be consistent with Laprise and the WRF model documentation (Skamarock et al. 2007), although the coordinate is more commonly referred to in the literature with the variable sigma.

With our implementation of the immersed boundary method, the sigma coordinate is retained. The grid is transformed to align with gently sloping background topography but does not conform to terrain that is explicitly resolved by the IBM. Topography is defined by the superposition of small- and large-scale features in domains using IBM. This concept is illustrated in Fig. 1, where urban-scale terrain is centered in a shallow valley. Pressure-based terrain-following coordinates are used to resolve the valley features in an outer, coarser grid domain. In a finer, nested domain, the valley floor is defined by the  $\sigma$  coordinate, whereas the buildings are represented with the IBM.

IBM is often used to eliminate coordinate transformations; however, in the case of flows within complex terrain, it is beneficial to use a hybrid methodology that combines the use of IBM with a transformed background coordinate. As the background sigma coordinate conforms only to gently sloping terrain, grid skewness and the associated errors are largely eliminated by using this IBM, although metric terms appear in the discretization for both the sigma WRF and the IBM–WRF models. Use of a hybrid coordinate facilitates grid nesting and increases the flexibility of the model. Hybrid techniques for using IBM on a curvilinear grid have been pursued in aeronautical (Ghias et al. 2007) and cardiovascular (Ge and Sotiropoulos 2007) applications as well. A difference here is that additional complexities arise in the treatment of nodes near the immersed boundary because of the time-dependent pressure coordinate.

### b. Governing equations

This work uses the Advanced Research WRF (ARW) model dynamics solver, which is a conservative finite-difference model that solves the nonhydrostatic compressible Navier–Stokes equations (Skamarock et al. 2007). The moist Euler equations are transformed into the isobaric terrain-following coordinate  $\eta$ , while additional terms such as diffusion, Coriolis, and parameterized physics (represented by  $F$ ) are computed in physical space. A velocity  $\hat{\eta}$ , defined as the contravariant velocity of the vertical coordinate, is introduced in the coordinate transformation, necessitating the solution of

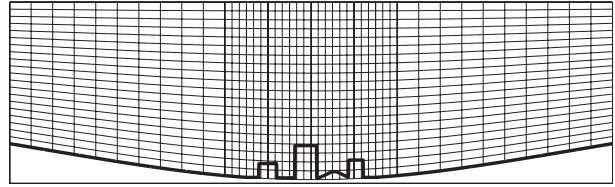


FIG. 1. Nested domains can be used to resolve both coarse and finescale terrain. Terrain-following coordinates conform to the gently sloping valley, whereas finescale terrain (in this case urban features) is resolved with the immersed boundary method.

an additional equation. Perturbation variables are introduced to reduce numerical errors; the perturbations are defined as the deviation from a time-invariant hydrostatically balanced reference state. Pressure  $p$ , specific volume (moist  $\alpha_m$  and dry  $\alpha_d$ ), geopotential  $\phi$ , and dry column mass  $\mu_d$  are cast as mean and perturbation values as  $\phi = \bar{\phi} + \phi'$ , where  $\phi$  represents a generic variable and the overbar indicates the hydrostatic base state. After substitution into the momentum equations, the hydrostatically balanced terms cancel out. The transformed equations are given in Eq. (1):

$$\partial_t \mu'_d + \nabla_\eta \cdot (\mu_d \mathbf{V}) = 0 \quad (1a)$$

$$\begin{aligned} \partial_t (\mu_d \mathbf{V}_H) + \nabla_\eta \cdot (\mu_d \mathbf{V}_H \times \mathbf{V}) + \mu_d (\alpha_m \nabla_\eta p' + \alpha'_m \nabla_\eta \bar{p}) \\ + \frac{\alpha_m}{\alpha_d} [\mu_d \nabla_\eta \phi' + (\nabla_\eta \phi)(\partial_\eta p' - \mu'_d)] = \mathbf{F}, \end{aligned} \quad (1b)$$

$$\begin{aligned} \partial_t (\mu_d w) + \nabla_\eta \cdot (\mu_d \mathbf{V} w) \\ - g \left( \frac{\alpha_m}{\alpha_d} \partial_\eta p' + \frac{\alpha_m - \alpha_d}{\alpha_d} \bar{\mu}_d - \mu'_d \right) = F, \quad \text{and} \quad (1c) \end{aligned}$$

$$\partial_t \phi' + \mathbf{V} \cdot \nabla_\eta \phi - gw = 0. \quad (1d)$$

In the previous equations, the velocity vector is  $\mathbf{V} = (u, v, \eta)$ ,  $\mathbf{V}_H$  includes the horizontal velocities, and  $\nabla_\eta = (\partial_x, \partial_y, \partial_\eta)$  operates on coordinate surfaces. Geopotential is defined as  $\phi = gz$ , so that  $\nabla_\eta \phi$  is a substitute for the Jacobian term  $\nabla_\eta z$ .

In addition to conservation of mass and momentum, a conservation Eq. (2) is solved for additional scalar quantities, such as potential temperature  $\theta$ , water vapor  $q_v$ , ice  $q_i$ , and passive scalars, given here as

$$\partial_t (\mu_d \phi) + \nabla_\eta \cdot (\mu_d \mathbf{V} \phi) = F_\phi. \quad (2)$$

Pressure is then diagnosed from the equation of state below, where  $\gamma_d$  is the ratio of heat capacities of dry air  $C_p/C_v$ ,  $p_o$  is the surface pressure, and  $R_d$  is the universal gas constant:

$$p = p_o \left( \frac{R_d \theta}{p_o \alpha_d} \right)^{\gamma_d}. \quad (3)$$

### c. Discretization schemes

The governing equations are discretized on an Arakawa-C staggered grid. Uniform grid spacing is used in the horizontal directions, and the grid may be stretched in the vertical direction. Second- through sixth-order finite-difference schemes are available for discretization of the advective terms. For the simulations presented here, the default finite-difference scheme for advection is used, which is fifth order for horizontal derivatives and third order for vertical derivatives. Diffusive terms are calculated with a second-order scheme.

A conservative split-explicit time-integration scheme handles the full range of frequencies admitted by the compressible Navier–Stokes equations (Wicker and Skamarock 2002; Klemp et al. 2007). In this scheme a third-order explicit Runge–Kutta method is used for time advancement of meteorologically significant low-frequency physical modes, although a smaller time step is needed to account for the higher-frequency modes such as acoustic waves. Variables from the Navier–Stokes equations and the prognostic equation for potential temperature are split into the value from the most recent Runge–Kutta time step  $t$  and a perturbation on the acoustic time step  $\tau$ , so that  $\varphi = \varphi^t + \varphi^\tau$ . Horizontally propagating acoustic modes are integrated using an explicit forward–backward scheme, and vertically propagating acoustic modes and buoyancy oscillations are treated implicitly.

## 3. Treatment at the immersed boundary

IBM is used to represent the effects of boundaries on a nonconforming structured grid. Boundary conditions are imposed with the addition of a body force term  $F_B$  in the conservation equations for momentum and scalars [Eq. (4)]. The body force term takes a zero value away from the boundaries but it modifies the governing equations in the vicinity of the boundary:

$$\partial_t \mathbf{V} + \mathbf{V} \cdot \nabla \mathbf{V} = -\alpha \nabla p + \nu \nabla^2 \mathbf{V} + \mathbf{g} + F_B \quad \text{and} \quad (4a)$$

$$\partial_t \varphi + \mathbf{V} \cdot \nabla \varphi = \nu_t \nabla^2 \varphi + F_\varphi + F_B. \quad (4b)$$

The exact details of this modification have varied among researchers since IBM was introduced by Peskin in 1972. Generally, IBM methods include a determination of the forcing term and an interpolation scheme to reconstruct the boundary condition at a surface that is not coincident with computational nodes.

### a. Flow reconstruction at the immersed boundary

The IBM used in this work falls into a category commonly referred to as discrete or direct forcing, which first appeared in the work of Mohd-Yusof (1997). With this method, the velocity or scalar value is modified at forcing points near the boundary to enforce the boundary condition, eliminating the need for explicit calculation of the body force term in the numerical algorithm. The method is especially adept at handling rigid boundaries and produces a sharp representation of the fluid–solid interface. Within the direct-forcing class of methods, we have adopted a finite-difference approach, where forcing is applied at ghost cells located in the solid domain. This method has several advantages, most notably the straightforward extension from two to three dimensions. One drawback of this method is that it is not strictly conservative at cut cells. Despite errors in local conservation, successful results using this method have been reported extensively in the literature (Mittal and Iaccarino 2005; Ghias et al. 2007).

In the case that the boundary is coincident with computational nodes, the boundary condition can be imposed by assigning it at the coincident node. However, when the boundary passes through the grid in an arbitrary manner, the grid points are not generally aligned with the boundary. This is always the case with staggered grids, such as the Arakawa-C grid used in the WRF model. An interpolation method must be used to determine the forcing needed at actual computational nodes. The first step in reconstructing the flow around a boundary is to specify the terrain independently of the grid. For implementation into the WRF model, we have allowed specification of terrain height at twice the resolution of the horizontal grid. Terrain elevation is specified as a function of  $x$  and  $y$ , which is compatible with the typical format of raw lidar and digital elevation data.

The next step is to identify cells that are cut by the immersed boundary. With a staggered grid, cut cells must be determined for each flow variable that will have a boundary condition imposed. Each node located at the cell center and on each face of the staggered grid is marked as a solid or fluid node. Cells containing both types of nodes are defined as cut cells. Flow variables can be reconstructed at fluid nodes, as in Fadlun et al. (2000), Gilmanov et al. (2003), Balaras (2004), and Choi et al. (2007), or at solid nodes, as in Gao et al. (2007), Ghias et al. (2007), and Mittal et al. (2008). In this work, flow variables are reconstructed at solid nodes, which is a technique known as the ghost-cell method. Ghost points are identified as the layer of nodes belonging to cut cells that are within the solid region of the domain (as shown in Fig. 2).



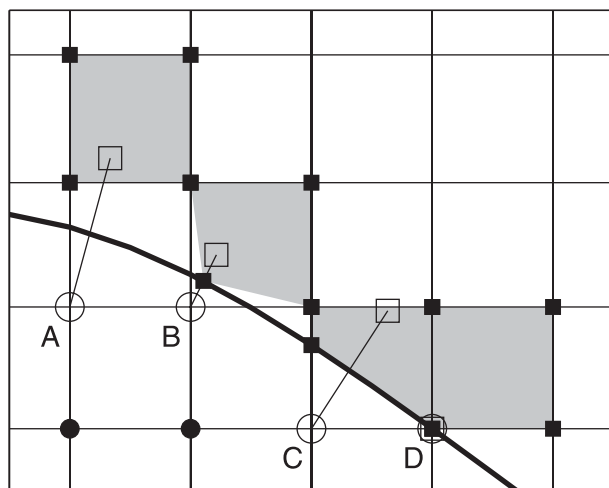


FIG. 2. The four examples of the interpolation scheme used to reconstruct the immersed boundary. Ghost nodes are marked with open circles and image points with open squares. The normal vector to the immersed surface connects the ghost and image points. Interior nodes are indicated with a filled circle and the neighbors used in the interpolation scheme are marked with a filled square. The interpolation region is shaded.

The value of the variable at the ghost cell, which will enforce either a Dirichlet (5a) or Neumann (5b) boundary condition at the immersed surface  $\Omega$ , must be computed:

$$\varphi = \varphi_{\Omega} \quad \text{and} \quad (5a)$$

$$\mathbf{n} \cdot \nabla \varphi = \frac{\partial \varphi}{\partial \mathbf{n}} \Big|_{\Omega}. \quad (5b)$$

Several different interpolation methods have been employed by researchers for the purpose of making this calculation, ranging from linear interpolation to inverse distance weighting schemes (Iaccarino and Verzicco 2003). For the pressure-based compressible flow equations in the WRF model and the associated scalar equations, we have developed a unique bilinear reconstruction scheme appropriate for two-dimensional terrain. Three-dimensional terrain is included by extending the method to use a trilinear interpolation scheme, as researchers have also done in CFD codes (Mittal et al. 2008).

The bilinear interpolation method used in this work is illustrated in Fig. 2. First, the location of the ghost point is reflected across the boundary in the surface normal direction, and this is labeled an image point. When a Dirichlet boundary condition is used, the ghost-point value is related to the image-point value with  $\varphi_G = 2\varphi_{\Omega} - \varphi_I$ . For a Neumann boundary condition, the relationship is  $\varphi_G = \varphi_I - \overline{\text{GI}}(\partial\varphi/\partial\mathbf{n})|_{\Omega}$ , where  $\overline{\text{GI}}$  is the distance between the ghost and the image points.

The value of the image point is calculated by using interpolation over the shaded region in Fig. 2, where the interpolant is

$$\varphi = c_1 + c_2x + c_3z + c_4xz. \quad (6)$$

Four neighboring points are used to define the interpolation region and are chosen as either computational nodes or as points on the boundary. For Dirichlet boundary conditions, the neighbors nearest to the image point are chosen. For Neumann boundary conditions, the neighbors nearest to the intersection of the surface normal with the boundary are chosen.

Because the boundary may intersect the computational cell in an arbitrary manner, the IBM algorithm must include logical directives to choose neighbors for a variety of geometric cases. Examples of the choices made for a Dirichlet boundary condition are shown in Fig. 2, but this illustration is not exhaustive. It can be seen in case A that if all four nodes surrounding the image point belong to the fluid domain, then these are chosen as the nearest neighbors. In case B, one of the surrounding computational nodes lies within the solid domain, so in this case the boundary point in the surface normal direction is chosen as a nearest neighbor. If two of the neighboring computational nodes are solid points, then two boundary points are chosen (as in case C). In case D, the ghost point is in close proximity to the boundary, and with the time variant grid this point can move between the solid and fluid domains. In close proximity cases, flexibility is added to the reconstruction scheme so that it is applied at this node, regardless of being in the solid or fluid domain. If the ghost node lies in the fluid domain, then it is excluded from being used as a neighbor because solid or ghost nodes are never used as neighbors in the interpolation scheme. Additionally, because the WRF grid remains curvilinear as it follows large-scale terrain features, we found that special care must be exercised in determining the interpolation neighbors by accounting for horizontal gradients in the vertical coordinate. This is done by searching for neighbors in each vertical column of nodes independently because adjacent nodes will not have identical vertical heights.

The constants  $\mathbf{c}$ , from the interpolant [Eq. (6)], are determined by solving a system of equations [Eq. (7)] for each ghost point, where the rank is equal to the number of neighbors, given as

$$\mathbf{c} = \mathbf{A}^{-1}\boldsymbol{\varphi}. \quad (7)$$

Here, the matrix  $\mathbf{A}$  and the vector  $\boldsymbol{\varphi}$  are dependent on the neighbors chosen for the interpolation and the

type of boundary condition being imposed. For Dirichlet boundary conditions, Eq. (6) appears in the matrix equation. If the neighbor represented by Eq. (6) is a computational node,  $\phi$  takes the calculated value at the node. If the neighbor is a boundary point, then the boundary condition [Eq. (5a)] is assigned to  $\phi$ . For Neumann boundary conditions, the gradient of the interpolation function [Eq. (6)] is substituted into the boundary condition [Eq. (5b)] and Eq. (8) results, so that

$$\frac{\partial \phi}{\partial \mathbf{n}} = c_2 \mathbf{n}_x + c_3 \mathbf{n}_z + c_4 (\mathbf{n}_z x + \mathbf{n}_x z). \quad (8)$$

For neighbors on the boundary, this equation is used in Eq. (7) instead. Once the interpolation constants are determined, the value of the image point is found with Eq. (6). As a last step, the variable value at the ghost node is calculated and assigned.

The method proposed here is similar to the interpolation schemes used by Tseng and Ferziger (2003) and Mittal et al. (2008) but has additional favorable properties. Tseng and Ferziger used a linear interpolation scheme with three neighbors. The boundary point and the two nearest computational nodes to the ghost point were used for determining the weighting coefficients used in calculating the ghost point value. Large extrapolation coefficients are the result of when the nearest neighbors are close to the boundary point and the ghost point is relatively far away. When large gradients exist in flow variables, such as in velocity when a no-slip boundary condition is used, large weighing coefficients cause the IBM to assign unphysical velocity values to the ghost points, often leading to numerical instabilities. Our method eliminates this problem by choosing neighbors that are closest to the image points when a Dirichlet boundary condition is used and closest to the boundary point when a Neumann boundary condition is used. This ensures that the flow variable assigned at the ghost point has a physical value and that numerical stability is maintained. Mittal et al. (2008) allow neighboring solid nodes to be chosen as interpolation neighbors. These solid nodes are ghost nodes, leading to a coupling of the solution. An additional iterative procedure is used to deal with the coupling of ghost cell values. Our method avoids this additional step by only choosing neighbors in the fluid domain.

## b. IBM in the WRF model

### 1) INITIALIZATION

Standard preprocessing routines are used to create domain input data for the IBM–WRF model, creating a terrain-following grid based on the large-scale topography. Variables are initialized throughout the en-

tire domain, even though some points will be reclassified as solid points during integration. Changes to the WRF model's preprocessing procedure include defining small-scale terrain, which will be resolved with the IBM, separately from the large-scale terrain, to which the terrain-following coordinates conform. For cases with atmospheric physics, the land surface properties are modified to reflect the new location and properties of the small-scale terrain.

An additional domain initialization routine exists in the run-time portion of the WRF model executable and is called for all domains including the parent domain, newly created nested domains, and moving domains. This initialization routine is where the IBM initialization is added and the immersed boundary condition is set for the prognostic variables. For the simulations presented in this paper, velocity fields are initialized with a no-slip condition and scalar fields are initialized with a zero-flux condition.

During initialization, the IBM procedure begins with the determination of ghost points and ends by assigning values at ghost points. The full values (base state plus perturbations) of  $u$ ,  $v$ ,  $w$ ,  $\theta$ , and all scalars are used. First, nodes must be marked as interior or exterior to the domain. To make this determination, the elevation of each grid cell must be converted from geopotential by combining the base and perturbation states and dividing by gravity,  $z = (\bar{\phi} + \phi')/g$ . Geopotential is defined on the horizontal faces of a computational cell, coincident with the  $w$  velocity. Height is averaged to the locations of the other variables, which are located at the cell center and on vertical cell faces. Solid nodes, which are internal to the terrain, are set to zero for velocity variables, but the values of temperature, moisture, and passive scalars are not modified at these points.

Several variables are not modified at the immersed boundary. Pressure and density are defined by equations of state and do not require a boundary condition. The equation for perturbation geopotential [Eq. (1d)] is obtained by rearranging the definition of the coordinate velocity  $\eta$ . Because the coordinate is allowed to evolve independently of the immersed boundary, an immersed boundary condition is not applied.

### 2) INTEGRATION

As noted in section 2, a time-split scheme is used for numerical integration. Velocity and potential temperature are advanced on the small acoustic time step, although advection–diffusion equations for moisture and passive scalars are integrated exclusively on the large Runge–Kutta time step. IBM routines must be called on both time-step frequencies to enforce boundary conditions for the separate integrations. Additionally,

because the isobaric coordinate in the WRF model evolves in time, the IBM procedure of reconstructing the boundary with geometric interpolation must be completed for each iteration of the solver.

During integration,  $u$ ,  $v$ ,  $w$ , and  $\theta$  must be updated during the acoustic time step on which the coupled perturbation variables are advanced,  $(\mu_d u)^\tau$ ,  $(\mu_d v)^\tau$ ,  $(\mu_d w)^\tau$ , and  $(\mu_d \theta)^\tau$ . IBM routines are called after advancement of all variables on the acoustic time step. To set the immersed boundary condition, the full uncoupled variable is calculated as  $\phi = [(\mu_d \phi)^\tau + \Delta\tau + (\mu_d \phi)^t] / \mu_d$ , and the full grid height is calculated as  $z = (\bar{\phi} + \phi'^t + \phi'^{\tau+\Delta\tau}) / g$ . Full variable values are used at nodes included in the interpolation that determines the ghost point value  $\phi_G$ . The value of the coupled acoustic perturbation at the ghost point is then calculated as  $(\mu_d \phi)_G^{\tau+\Delta\tau} = \mu_d \phi_G - (\mu_d \phi)_G^t$ , and this coupled value is used at the ghost point to enforce the boundary condition. Again, a boundary condition is not set at the immersed boundary for  $\phi'^\tau$ , and  $p'^\tau$  and  $\alpha_d'^\tau$  are diagnosed from state equations.

Scalar variables are advanced on the large Runge–Kutta time step. Potential temperature and scalars are collocated, so if the same type of boundary condition is used for all of these variables, then the grid elevation, ghost point locations, and interpolation points from the last acoustic time step are still valid. This information must only be recalculated if the type of boundary condition varies. The IBM procedure, as previously explained, is used to set the boundary condition on each scalar, which for this work only includes water vapor. Lundquist et al. (2008) performed simulations using the IBM with a passive tracer.

During integration the reconstructed solution is enforced at the first grid point within the solid domain; however, several options exist for the treatment of additional interior nodes. When higher-order finite-difference schemes are used, solid nodes are included in the stencil of fluid nodes and can affect the accuracy of the derivatives at these nodes. Iaccarino and Verzicco (2003) used both a second-order and a high-order upwind finite-difference scheme and found that with the direct forcing type of IBM, the treatment of interior nodes has little or no effect on the external flow solution. A fourth-order finite-difference scheme is used in von Terzi et al. (2001), along with four different interior treatments. This study found that in many cases the interior treatment did not affect the outer-fluid flow; however, when the accuracy of near-wall gradients is critical to obtaining the correct flow solution, greater accuracy can be achieved by using a one-sided finite difference at the immersed boundary. We implemented various treatment options for solid nodes and tested

both second-order and higher-order advection schemes, and we concluded that there was little effect for the test cases presented in this manuscript. Therefore, we allow prognostic variables to evolve in accordance with the governing equations within the solid domain and use the default advection scheme (fifth-order horizontal, third-order vertical). Options for eliminating this issue in the future are to reduce the order of the finite-difference scheme at the immersed boundary, apply the reconstruction scheme to multiple layers of ghost nodes, or use one-sided differences at the boundary.

## 4. Boundary conditions

### a. Comparison of native and immersed boundary conditions

The native bottom boundary conditions in the  $\sigma$  WRF model set the contravariant coordinate velocity to zero and use a kinematic boundary condition for the Cartesian vertical velocity. The set of equations given by Eqs. (9a) and (9b) create a free slip bottom boundary condition:

$$\dot{\eta}_{\text{surf}} = 0 \quad \text{and} \quad (9a)$$

$$w_{\text{surf}} = \mathbf{V} \cdot \nabla h. \quad (9b)$$

In Eq. (9b),  $h$  is a function specifying the terrain height. Horizontal velocities are extrapolated to the surface using a quadratic Lagrange polynomial. The shear stress at the boundary is implicitly set to zero unless the effects of friction are taken into account by using log-law similarity theory. Vertical gradients of scalars are zero unless forcing is provided by the atmospheric physics parameterizations.

Our IBM implementation uses a no-slip bottom boundary condition that is commonly used in CFD simulations over complex topography, such as urban simulations with explicitly resolved buildings. To enable comparisons between the IBM–WRF and the WRF models, we have added a no-slip condition to the WRF model. This boundary condition is also useful for high-resolution simulations performed with the WRF solver and enables comparisons between the WRF model and other CFD codes that use a no-slip boundary condition. The no-slip implementation is described in the following section, and this boundary condition is used for the verification simulations presented in section 6.

Wall models, which approximate surface fluxes of momentum, are commonly used in NWP and have also been applied at immersed boundaries. Cristallo and Verzicco (2006) use a two-layer model with the IBM, which reconstructs the velocity field at the first fluid point by solving a simplified boundary layer equation.

Choi et al. (2007) reconstructs the velocity field by fitting the tangent component to a power law. These studies show promising results, and the development of a wall model for the IBM–WRF is in progress (Lundquist 2006).

When specifying surface fluxes, our IBM implementation can apply fluxes in either the surface normal or vertical direction. It is critical that flux boundary conditions be enforced in the surface normal direction over steep slopes, such as vertical building surfaces. The WRF model prescribes vertical fluxes at the surface; therefore, the vertical flux boundary condition can be used when simulations with IBM are compared directly to the WRF model, as is done in the idealized valley simulation. To further simplify comparisons between the WRF and the IBM–WRF models, we have used a constant eddy viscosity. Turbulent viscosity is needed to compute surface fluxes, and variation in this quantity would lead to differences in the enforced boundary condition.

#### b. Implementation of the no-slip boundary condition

As part of this work, a no-slip bottom boundary condition has been added to the WRF model as an additional option beyond the standard boundary conditions available in the code. To add the no-slip option, several modifications were made to the original WRF boundary conditions, which are given in the previous section as Eq. (9). The first equation [Eq. (9a)] is still appropriate because it sets the contravariant velocity of the bottom coordinate to zero. This ensures that the bottom coordinate follows the terrain and does not change position in time. The kinematic boundary condition, given by Eq. (9b), is no longer valid and is replaced with the requirement that the Cartesian vertical velocity  $w = 0$  on the boundary. The need to extrapolate the horizontal velocities to the surface is eliminated and therefore no approximations are made in our formulation of the no-slip boundary condition. With these changes, the no-slip boundary condition is satisfied for the advective fluxes in the continuity and momentum equations. The use of an Arakawa–C staggered grid (illustrated in Fig. 3) eliminates the need to explicitly set the  $u$  and  $v$  velocities equal to zero on the surface. In the case of the vertical derivative in the advective term of the horizontal momentum Eq. (1b), it can be seen that because  $\dot{\eta}$  and  $\mathbf{V}_H$  are both zero at the surface, a one-sided vertical difference is sufficient for calculating this term at the bottom boundary of the domain.

Boundary conditions must be imposed on the diffusive flux terms in the momentum equation in order to achieve a no-slip condition. The WRF model employs eddy viscosity-type turbulence models so that the tur-

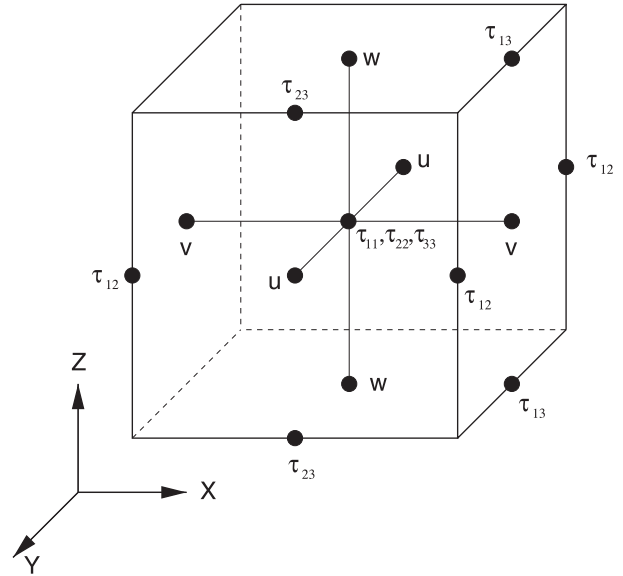


FIG. 3. The Arakawa–C staggered grid used in WRF.

bulent mixing terms take the form given by Eq. (10) and the stress tensor is Eq. (11):

$$F_i = -\frac{\partial \tau_{ij}}{\partial x_j} + \frac{\partial}{\partial z}(\mathbf{V}_H z \cdot \tau_{ij}) \quad \text{and} \quad (10)$$

$$\tau_{ij} = -\mu \nu_t D_{ij}. \quad (11)$$

Diffusive flux terms are calculated in physical space, so that the second term in Eq. (10) arises as the Jacobian of the coordinate transformation, where  $z$  is the physical height of the coordinate. The variable  $\mu$  is the column mass per unit area,  $\nu_t$  is the turbulent eddy viscosity, and  $D_{ij}$  is twice the strain rate or deformation tensor and is defined as

$$D_{ij} = \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - (\mathbf{V}_H z)_j \frac{\partial u_i}{\partial z} - (\mathbf{V}_H z)_i \frac{\partial u_j}{\partial z}. \quad (12)$$

Here, the locations of the components of the deformation and stress tensors are shown in Fig. 3. Diagonal elements of each tensor are located at the cell center, and off-diagonal elements are located at vorticity points, which are centered on the cell edges.

The presence of the Jacobian terms complicates the implementation of the no-slip boundary condition by adding partial derivatives in the vertical direction to each term. The native WRF model boundary condition (which is free slip) uses a quadratic Lagrange polynomial to estimate the  $u$  and  $v$  velocity at the surface for the purpose of calculating  $D_{11}$ ,  $D_{22}$ , and  $D_{12}$ . This step is eliminated for the no-slip boundary condition, and



a value of zero is used for  $u$  and  $v$  on the surface. In the native WRF model boundary condition,  $D_{13}$  and  $D_{23}$  are zero because they are located on the surface. For the no-slip boundary condition, the surface values are calculated using a one-sided difference. These new values of the deformation tensor are used for calculating the turbulent stresses.

Finally, the equation for the diffusion terms includes additional terms arising from the Jacobian of the coordinate transformation, as seen in Eq. (10) previously. With a free-slip boundary condition,  $\tau_{ij}$  is zero on the surface. To satisfy the no-slip boundary condition, certain elements of  $\tau_{ij}$  must now be calculated at various locations on the surface, such as below the cell center, below  $u$  points, etc. The calculation of these additional surface stress terms is consistent with the procedure previously described for evaluating deformation at the surface, and the methods employed are one-sided differences, setting velocities to zero at the surface and averaging surface values to intermediate locations.

## 5. Inclusion of atmospheric parameterizations

Atmospheric parameterizations often provide surface forcing in NWP models. Surface physics interact with the lowest model level when terrain-following coordinates are used; therefore, it was necessary to modify these routines to couple them to the immersed boundary. The set of schemes that have been coupled to the immersed boundary are the Rapid Radiative Transfer Model (RRTM) for longwave radiation, the the fifth-generation Pennsylvania State University–National Center for Atmospheric Research Mesoscale Model (MM5; Dudhia) model for shortwave radiation, MM5 surface layer scheme, and the Noah land surface model. Documentation of these schemes can be found in Skamarock et al. (2007) and the references within.

### a. Radiation models

The radiation schemes currently used in the WRF are column models, where each column is treated independently. The terrain is treated as if it were a horizontal plane at each column. To couple this model with the immersed boundary method, the vertical integration limits are modified to exclude any portion of the atmosphere below the terrain. Additionally, variables such as temperature, pressure, and grid spacing must be modified in the vicinity of the immersed boundary by extrapolating these values onto the surface, using data from fluid points above the immersed terrain. The modified data for the atmospheric column are then used in the existing radiation routines.

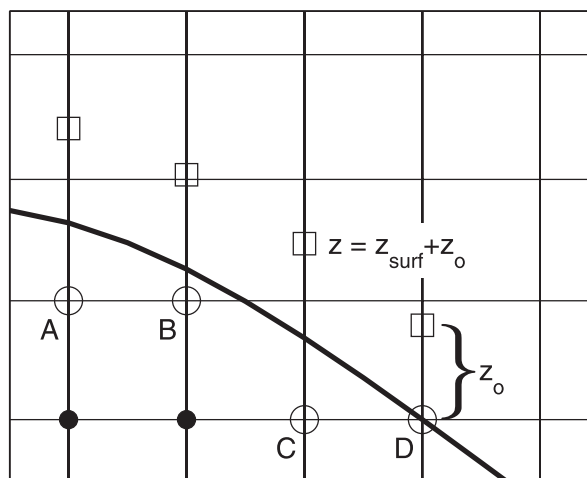


FIG. 4. Examples of the modified reference height  $z$  used at the immersed boundary. Ghost nodes are marked with open circles (○) and modified reference heights with open squares (□).

### b. Surface physics

The WRF model calculates exchange coefficients of heat, moisture, and momentum using the MM5 surface layer model, which is based on Monin–Obukhov similarity theory. Stability regimes are determined by the bulk Richardson number [defined by Eq. (17)], where  $g$  is the gravitational constant,  $z$  is a reference height above the terrain,  $z_{\text{surf}}$  is the terrain level,  $\theta_v$  is virtual potential temperature, and  $V$  is the horizontal wind speed, which has been modified by a vertical convection velocity:

$$\text{Ri}_b = \frac{g[\theta_v(z) - \theta_v(z_{\text{surf}})](z - z_{\text{surf}})}{\theta_v(z)V(z)^2}. \quad (13)$$

Here, the WRF model chooses the reference height  $z$  to be the height of the first computational cell, so that the vertical grid spacing defines the offset  $z_o$  between the reference height and the surface. For the immersed boundary method, the reference height must be modified so that it is located above the terrain rather than at the first grid point in the domain. With the IBM–WRF model, the offset  $z_o$  is added to the terrain height  $z_{\text{surf}}$  (as shown in Fig. 4). This modified reference height is above the immersed boundary; however, unlike in the WRF model case, it does not coincide with a computational node. Velocities and temperature, as well as those quantities needed for the conversion to virtual temperature, must be interpolated to the modified reference height. Additionally, the surface values must be replaced with values on the immersed boundary. The Monin–Obukhov length scale is then calculated based

on the stability class. Next, the 2- and 10-m reference height data are calculated. Finally, the exchange coefficients are calculated for heat and moisture. Again, the values used in these calculations are modified for the immersed boundary, following the example given previously for the bulk Richardson number. It is important to note that although scalar fluxes are calculated with this approach, surface momentum fluxes are not based on Monin–Obukhov theory when the no-slip boundary condition is used because these conditions are incompatible. This is true of both the native WRF and the IBM–WRF model computations shown here.

The WRF model uses the Noah land surface model to provide heat and moisture fluxes at the land surface, using radiative and atmospheric forcing from the radiation and surface layer schemes as well as land surface properties to determine the fluxes. These fluxes serve as the bottom boundary conditions for potential temperature and water vapor. This model also calculates a water budget over four model layers, including physical effects like evapotranspiration and runoff. Many of the quantities used by the land surface model have already been modified to include the effects of the immersed boundary by the models discussed previously. The Noah model also makes use of similarity theory. As before, the reference height is added to the height of the immersed boundary, and the variables used in the calculation are modified accordingly.

## 6. Verification

### a. Witch of Agnesi hill simulations

The performance of the IBM method is examined in this section by considering flow over a two-dimensional hill, defined by the Witch of Agnesi curve. This case is often used to verify that topographic forcing is correct at the bottom boundary and is chosen here because the terrain slopes are shallow enough so that a direct comparison can be made between the WRF and the IBM–WRF models.

The test flow case is the startup flow over a two-dimensional hill. The terrain height  $h_t$  is defined by the Witch of Agnesi curve given in Eq. (14), using a peak height  $h_p$  and mountain half-width  $a$  of 100 m each:

$$h_t(x) = \frac{h_p}{1 + (x/a)^2}. \quad (14)$$

The flow is initialized with a neutral and quiescent sounding and driven with a constant pressure gradient in the  $x$  direction. The number of grid points used in each direction is  $(n_x, n_y, n_z) = (120, 2, 172)$  for a total domain

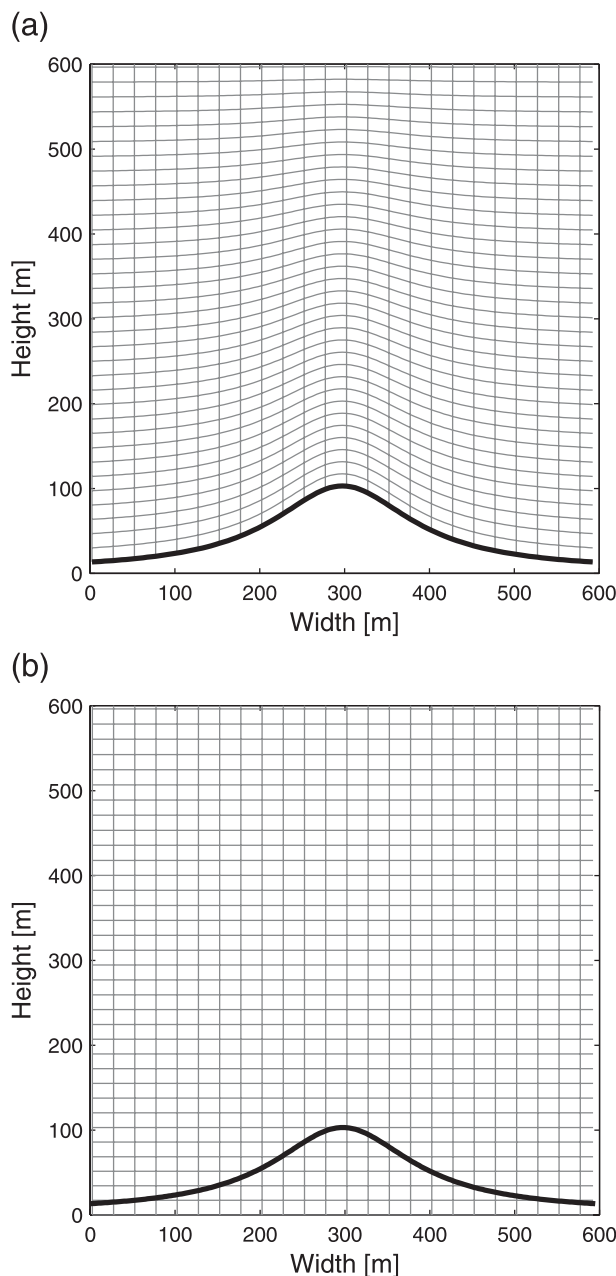


FIG. 5. (top) The WRF grid with terrain-following coordinates is shown in contrast with the (bottom) IBM–WRF grid. Terrain-following coordinates map to the bottom boundary; with IBM the coordinates remain flat. Every fifth grid line is shown.

size of  $(X, Y, Z) = (595, 5, 600)$  m. In the horizontal dimensions, a constant 5-m grid spacing is used. In the vertical direction, the grid is not stretched in  $\eta$ ; however, the pressure coordinate naturally results in stretching in physical space. The WRF and IBM–WRF model grids are shown in Fig. 5. At the initialization, the minimum vertical grid spacing in the WRF grid is  $\Delta z_{\min} \approx 2.85$  m

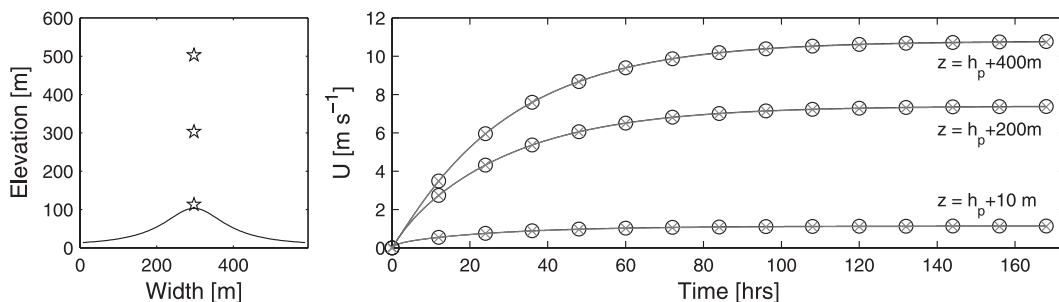


FIG. 6. (left) The  $u$  velocity component is plotted at the domain center for three different heights indicated by stars. The first point is 10 m above the terrain peak  $h_p$ , the second is 200 m, and the third is 400 m. (right) The IBM-WRF is plotted with a black line ( $\ominus$ ), whereas WRF is shown in gray ( $\times$ ). Markers are placed at 12-h intervals. The WRF and IBM-WRF solutions are virtually indistinguishable.

and occurs at the peak of the hill, whereas the maximum spacing is  $\Delta z_{\max} \approx 3.52$  m at the domain top. In the IBM domain, the vertical grid spans the entire 600-m domain height, giving a more uniform (and slightly coarser) grid resolution, with  $\Delta z_{\min} \approx 3.42$  m and  $\Delta z_{\max} \approx 3.60$  m.

Periodic boundary conditions are used at the lateral boundaries. A no-slip boundary condition is set on velocity at the terrain surface, along with a zero-flux condition on temperature. At the top of the domain, the native WRF model boundary condition is used (isobaric and a material surface), with a diffusive damping layer 50 m in depth.

The solution is integrated until a steady-state solution is obtained. Figure 6 shows the  $u$  velocity at the domain center for three different heights for seven days of simulation time. Evolution of the velocity field is nearly identical between the WRF and IBM-WRF model solutions, indicating that the rate of convergence toward steady state is not affected by the IBM.

Steady-state results are shown for the hill case after seven days (168 h) of simulation time. Figure 7 shows  $u$  and  $w$  velocity profiles at several locations across the domain. It can be seen that the IBM solution matches the no-slip WRF model solution almost exactly. To compare the results further, the IBM and WRF model solutions are linearly interpolated onto a common terrain-following grid. Differences in the simulations are quantified in Table 1, where maximum and minimum velocity values for the no-slip WRF solution are presented along with velocity values for the IBM-WRF model solution at the same location. Given the different vertical grid spacing in the two simulations, and the presence of errors associated with terrain-following coordinates in the  $\sigma$ -WRF model simulation, the excellent agreement is remarkable.

Contours of the same data are shown in Fig. 8, where contours of velocity and the difference between the IBM and the WRF model solutions are included. The  $u$  velocity increases as it crests the hill, and areas of weak

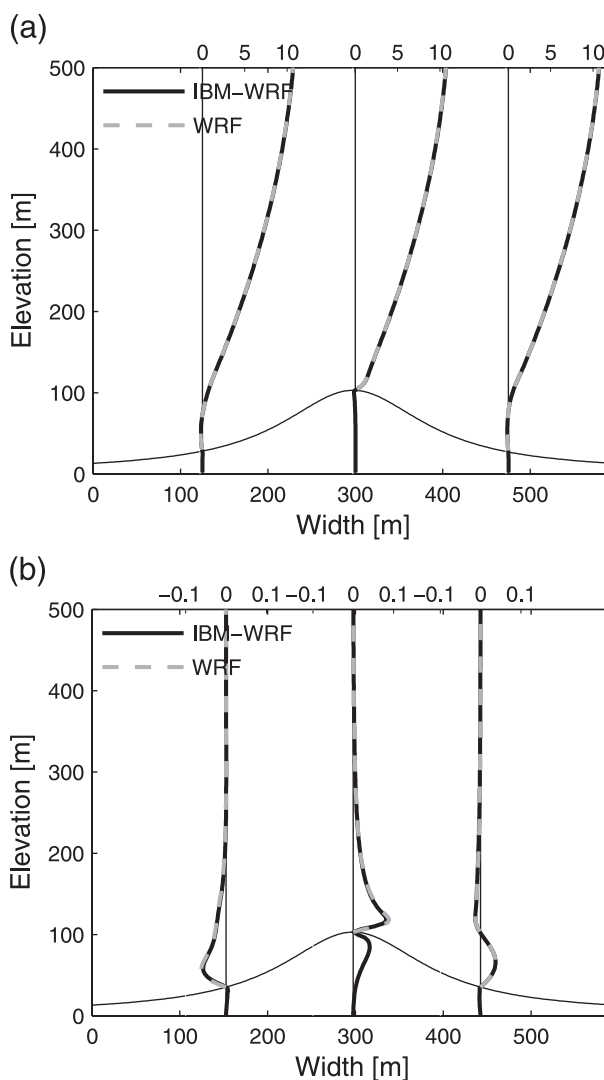


FIG. 7. Profiles of (a)  $u$  and (b)  $w$  velocity ( $\text{m s}^{-1}$ ) for steady-state flow over a hill at 168 h.

TABLE 1. Location of minimum and maximum velocities for steady-state flow over a hill in the  $\sigma$ -WRF simulation and the velocity values at this location for both the IBM-WRF and the  $\sigma$  WRF. Differences between the IBM and WRF solutions are given as both absolute and relative values.

	Location (m)		Velocity ( $\text{m s}^{-1}$ )		Difference	
	$x$	$z$	IBM	WRF	Absolute ( $\text{m s}^{-1}$ )	Relative (%)
$u_{\min}$	122.5	49.6	-0.203	-0.208	0.005	2.4
$u_{\max}$	402.5	500	10.736	10.713	0.023	0.2
$w_{\min}$	192.5	71.6	-0.077	-0.083	0.012	14.5
$w_{\max}$	272.5	107.6	0.192	0.192	$-4.7 \times 10^{-5}$	0

recirculation are observed in the troughs. In the IBM solution, a reverse flow develops under the terrain. This flow can be seen in Fig. 7; however, in Fig. 8 data below the hill are omitted because the IBM solution is interpolated onto a common terrain-following grid. The maximum differences present in Fig. 8 are 0.039 and  $-0.021 \text{ m s}^{-1}$  for the  $u$  and  $w$  fields, respectively. The domain-averaged differences are smaller, with  $\Delta U_{\text{ave}} = 0.017$  and  $\Delta W_{\text{ave}} = 6 \times 10^{-4} \text{ m s}^{-1}$ .

Contours of potential temperature for the IBM solution and differences in the IBM and the WRF solutions are shown in Fig. 9. It can be seen that with no surface heating the IBM solution maintains the initialization temperature of 288 K, whereas there is a very slight increase in temperature in the WRF model domain. It was

found that small errors developed in the WRF solution because of the native bottom boundary condition, which uses a Lagrange polynomial to extrapolate temperature to the surface.

### b. Idealized valley simulations

In the previous section, it was shown that our IBM implementation is able to reproduce results from numerical simulations using the native terrain-following coordinate in the WRF model. A no-slip boundary condition was imposed on the velocity at the terrain surface, and a zero gradient boundary condition was used for potential temperature. These two boundary conditions are adequate for studying a variety of flows and are commonly the only boundary conditions available in a traditional CFD code. For realistic atmospheric flows, it is necessary to represent nonzero fluxes at the surface, such as those of heat and moisture. Here, we examine nonzero-gradient boundary conditions on the immersed boundary in the context of thermally induced slope flows in an idealized valley.

The idealized valley terrain and initialization are similar to those used in the valley winds studies of Rampanelli et al. (2004) and Schmidli et al. (2008). The valley terrain is defined by  $h_t = h_p \times h_x$ , where the peak height is  $h_p = 1.5 \text{ km}$  and the valley profile  $h_x$  is given by Eq. (15). In this equation, the valley floor half width is  $V_x = 0.5 \text{ km}$  and the hill half-width is  $S_x = 9 \text{ km}$ :

$$h_x(x) = \begin{cases} 0, & |x| \leq V_x \\ 0.5 - 0.5 \cos\left(\pi \frac{|x| - V_x}{S_x}\right), & V_x < |x| < V_x + S_x \\ 1, & V_x + S_x \leq |x| \leq V_x + S_x + P_x \\ 0.5 + 0.5 \cos\left[\pi \frac{|x| - (V_x + S_x + P_x)}{S_x}\right], & V_x + S_x + P_x < |x| < V_x + 2S_x + P_x \\ 0, & |x| \geq V_x + 2S_x + P_x \end{cases} \quad (15)$$

Again, the terrain slopes are gentle enough that a direct comparison can be made between the WRF model, using terrain-following coordinates, and the IBM-WRF model. This case is more challenging for the immersed boundary method because the flow is induced by the approximate boundary conditions at the immersed surface itself instead of being induced by a large-scale pressure gradient or mean flow.

The domain is initialized with a quiescent sounding that is moist and stably stratified. The sounding prescribes a constant 40% relative humidity and the potential temperature is specified by  $\theta(z) = \theta_s + \Gamma z + \Delta\theta[1 - \exp(-\beta z)]$ , where  $\theta_s = 280 \text{ K}$ ,  $\Gamma = 3.2 \text{ K km}^{-1}$ ,  $\Delta\theta = 5 \text{ K}$ , and  $\beta = 0.002 \text{ m}^{-1}$ .

For the WRF model simulation with terrain-following coordinates, the total domain size is  $(X, Y, Z) = (60, 0.6, 10 \text{ km})$ , where the number of grid points used in each direction is  $(n_x, n_y, n_z) = (301, 3, 60)$ . When the IBM-WRF model is used, the bottom of the domain is lowered to 200 m below the zero level, so that the domain size is  $(X, Y, Z) = (60, 0.6, 10.2 \text{ km})$ . This allows a minimum of two grid points below the immersed terrain, and the number of grid points becomes  $(n_x, n_y, n_z) = (301, 3, 62)$ . In each simulation, the horizontal grid spacing is  $\Delta X = \Delta Y = 200 \text{ m}$ . For the terrain-following coordinate, the minimum vertical grid spacing occurs at the mountain peaks ( $\Delta Z_{\min} = 95.6 \text{ m}$ ) and the maximum at the domain top ( $\Delta Z_{\max} = 307.8 \text{ m}$ ). For the IBM-WRF model grid,



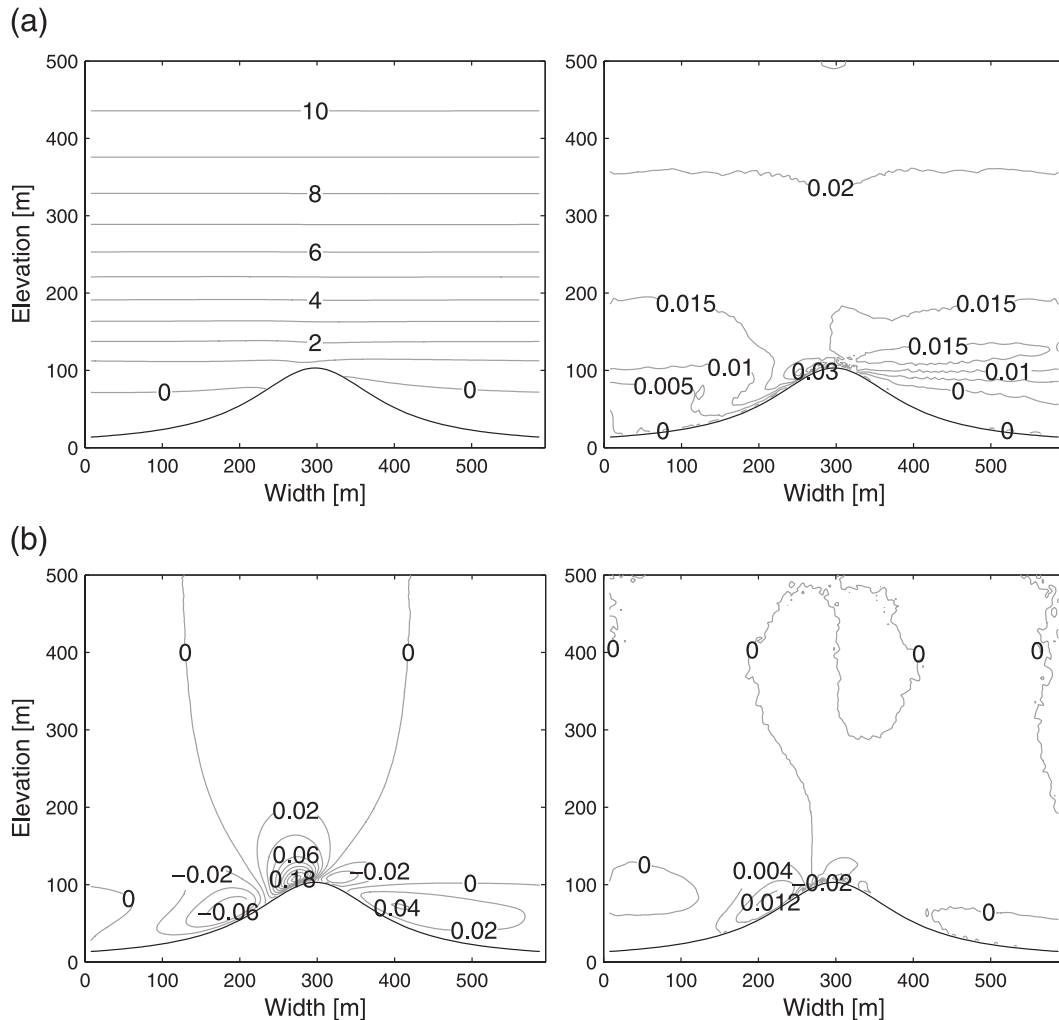


FIG. 8. (left) Contours of (top)  $u$  and (bottom)  $w$  velocity for the IBM–WRF for steady-state flow over a hill at 168 h. Contour intervals are  $1 \text{ m s}^{-1}$  for  $u$  and  $0.02 \text{ m s}^{-1}$  for  $w$ . (right) Contours of the difference between the IBM–WRF and the WRF solution are shown for (top)  $u$  and (bottom)  $w$ , with intervals of  $0.005$  and  $0.004 \text{ m s}^{-1}$ , respectively.

the minimum vertical grid spacing occurs at the lowest grid level, which is below the terrain ( $\Delta Z_{\min} = 101.9 \text{ m}$ ) and the maximum at the domain top ( $\Delta Z_{\max} = 307.4 \text{ m}$ ).

Total integration time for the simulation is 12 h, starting at 0600 UTC and ending at 1800 UTC. A constant eddy viscosity of  $60 \text{ m}^2 \text{ s}^{-1}$  is used in the turbulent diffusion terms, which is consistent with the typical range of daytime eddy diffusivities of  $10\text{--}100 \text{ m}^2 \text{ s}^{-1}$  (Yasuda 1988). When this eddy diffusivity is used, the mixing is strong enough to maintain a neutral stability profile during surface heating, thereby avoiding the formation of thermal convection cells and making instantaneous comparisons between simulations possible.

Here, both uncoupled and coupled cases are considered. In the uncoupled cases, the surface heating is specified as a function of time and there are no surface or land attributes such as vegetation or soil type. This allows

verification of the surface flux condition at the immersed boundary without the added complexity of the land surface model. In the coupled cases, the surface fluxes are prescribed by atmospheric parameterizations, which have been modified to recognize the immersed boundary as the terrain surface. Periodic boundary conditions are used at the lateral boundaries. A diffusive damping layer  $3.8 \text{ km}$  thick is used at the top of the domain, and a no-slip boundary condition is used at the terrain surface.

### 1) SPECIFIED SURFACE HEATING

In this case, the land surface model is turned off and a Neumann boundary condition is applied to potential temperature by defining the surface heat flux to be  $Q(t) = Q_{\max} \sin(\omega t)$ , where the maximum sensible heat flux is  $Q_{\max} = 200 \text{ W m}^{-2}$  and a period of  $\omega = 2\pi/24 \text{ h}$  is used to represent a daytime heating cycle.

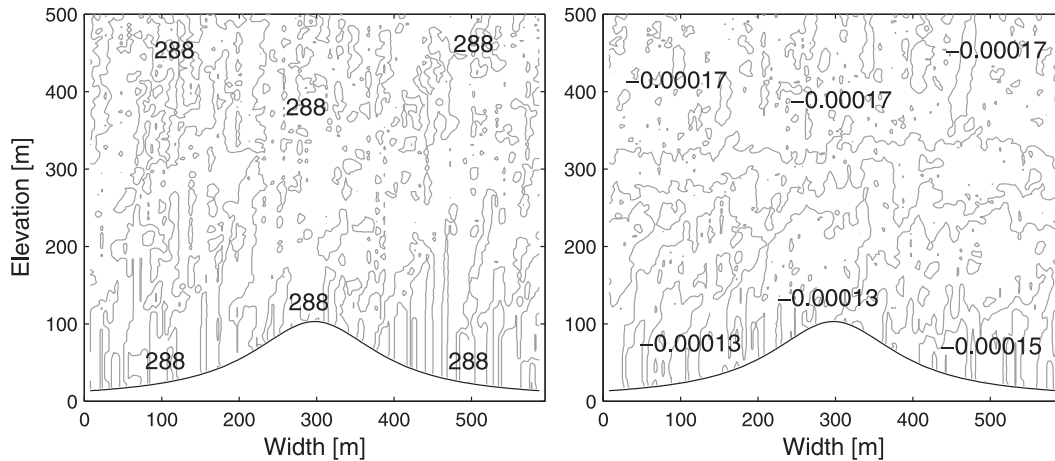


FIG. 9. (left) Contours of potential temperature (K) for the IBM–WRF. (right) The difference in the IBM and the WRF solutions.

Figure 10 depicts profiles of  $u$  and  $w$  velocity at 1200 UTC, which is the time of maximum radiative forcing. A clear circulation pattern forms with anabatic winds converging at the mountain peaks and subsidence at the valley center. The profiles match very well and show that the boundary condition at the immersed boundary is actually achieved. Figure 11 shows instantaneous profiles of potential temperature at the valley center as it evolves in time. The evolution of the profiles for the IBM and WRF model simulations compare well and excellent agreement is achieved in the development of the mixed layer.

To further analyze the differences between the IBM–WRF and the WRF model solutions, both time- and domain-averaged differences are calculated on a common and time-independent terrain-following grid. Once the two solutions are on a common grid they may be compared directly, although the error introduced by the interpolation is unknown. Spatial variations in the two solutions are calculated for a given variable  $\phi$  with Eq. (16a) and temporal variations are calculated with Eq. (16b):

$$\Delta\phi(x, \eta) = \overline{\phi_{\text{IBM}}(x, \eta, t) - \phi_{\text{WRF}}(x, \eta, t)}^t \quad \text{and} \quad (16a)$$

$$\Delta\phi(t) = \left| \overline{\phi_{\text{IBM}}(x, \eta, t) - \phi_{\text{WRF}}(x, \eta, t)} \right|^{x, \eta}. \quad (16b)$$

The results of the time-averaged spatial differences are presented in Figs. 12 and 13. Figure 12 shows time-averaged differences in  $\theta$ , where  $\Delta\theta_{\text{max}} = 0.099$  K and  $\Delta\theta_{\text{min}} = -0.023$  K. The IBM simulation predicts a higher temperature than the WRF model at the surface, with the maximum difference occurring on the valley slopes and a slightly cooler atmosphere aloft in the valley center. The higher surface temperature is attributed to

differences in the vertical grid spacing. In the  $\sigma$ -WRF model case, the average spacing between the first potential temperature node and the surface is  $\Delta z_{\text{ave}} = 51.0$  m, whereas the spacing is  $\Delta z_{\text{ave}} = 63.5$  m for the IBM–WRF model. As the vertical grid spacing increases, a larger temperature increment is needed between the surface and the first vertical grid point to maintain the specified flux boundary condition. Differences in  $\theta$  contribute to the differences in velocity shown in the next two figures. Velocity differences are  $\Delta U_{\text{max}} = \Delta U_{\text{min}} = \pm 0.254$  m s<sup>-1</sup> and  $\Delta W_{\text{max}} = 0.133$  m s<sup>-1</sup> and  $\Delta W_{\text{min}} = -0.068$  m s<sup>-1</sup>. In  $u$ , the largest differences are seen on the valley slopes, specifically inside of the valley. Additionally, the largest differences in  $w$  are seen at the valley peaks, where there is a slight offset in the location of the buoyant plumes. These differences are small considering that the simulations have different grids and use different methods for specifying boundary conditions.

Differences between the simulations can also be viewed as a function of time (as in Fig. 14). It can be seen here that the potential temperature fields are different at initialization. This difference is due to interpolation errors from both interpolating a nonlinear initial sounding profile onto the WRF and IBM–WRF grids and interpolating the solutions onto a common grid for comparison. Excellent agreement is seen in the potential temperature field and the differences are on the same order of magnitude at the end of the simulation as at initialization. The velocity fields are initialized as quiescent; therefore, they are identical at initialization. The differences in velocity rise during the first 30 min of the simulation and then rapidly decrease. A stair-step-type appearance in the flow is observed in the first 30 min of the IBM simulation because velocities at discrete points

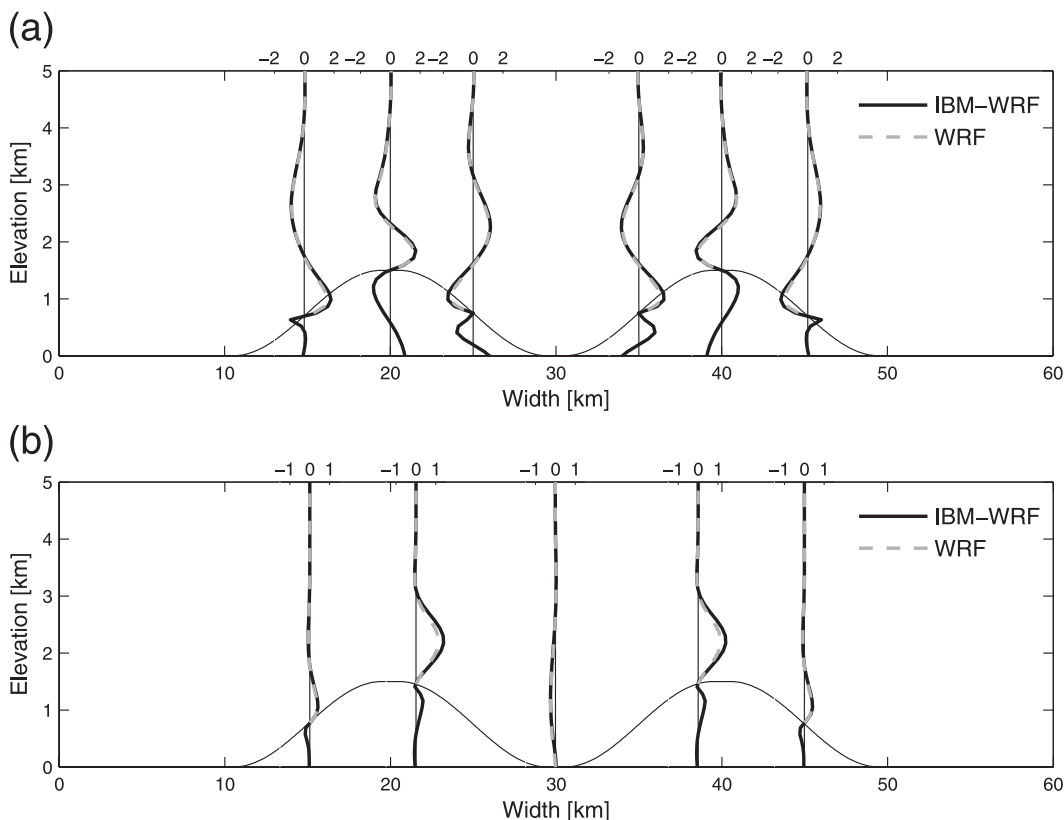


FIG. 10. Profiles of (top)  $u$  and (bottom)  $w$  velocity ( $\text{m s}^{-1}$ ) for the IBM-WRF and WRF, with terrain-following coordinates at 1200 UTC.

adjust to the smooth immersed boundary. As flow patterns develop, the stepped appearance quickly smooths out and disappears. This peak indicates that differences could be reduced by adding spinup time to the simulation. It is important to note that the differences do not grow in time and in fact trend downward as the simulation time increases (even though the solution is time dependent).

## 2) COUPLED SURFACE HEATING

In this section, the simulations are fully coupled, which means that the surface fluxes are calculated by the radiation and land surface schemes. The domain setup is as described for the cases with specified heating; however, in the coupled cases, soil properties are also initialized. The valley is located at  $36^\circ\text{N}$ ,  $0^\circ\text{E}$ . The simulation date is 21 March 2007 and the period is again from 0600 to 1800 UTC. The soil is initialized with uniform properties with a soil type of sandy loam, vegetation type of savannah, and vegetation fraction of 0.1. The soil temperature is initialized at equilibrium with the atmospheric temperature at the surface, and soil moisture is set at a 20% saturation rate. Four vertical levels are used in the soil model with depths of 0.1, 0.3,

0.6, and 1 m, for a total depth of 2 m. The RRTM is used for longwave radiation and the MM5 (Dudhia) scheme is used to model shortwave radiation. The MM5 surface layer model is used along with the Noah land surface model. As described in section 5, these schemes have been modified so that they are coupled to the immersed boundary.

Domain-averaged longwave radiation varies slightly between the two simulations (as seen in Fig. 15). The WRF and IBM-WRF cases match almost exactly at the first data point (0615 UTC) but diverge slightly during the simulation. At the end of the simulation (1800 UTC), the difference in longwave radiation is  $1.15 \text{ W m}^{-2}$ , or 0.44%. Differences are even smaller for shortwave radiation, where nearly perfect results are achieved. The maximum difference between the two solutions is  $0.50 \text{ W m}^{-2}$  or 0.06% and occurs at 1400 UTC.

The spatial variations in longwave and shortwave radiation are documented in Fig. 16, where they are shown at 1200 UTC. There is a difference of about  $56 \text{ W m}^{-2}$  in longwave radiation from the valley floor to peak. In comparison, at noon there is a maximum difference of  $1.17 \text{ W m}^{-2}$  in the two simulations. This shows that any error created by coupling the radiation scheme to the

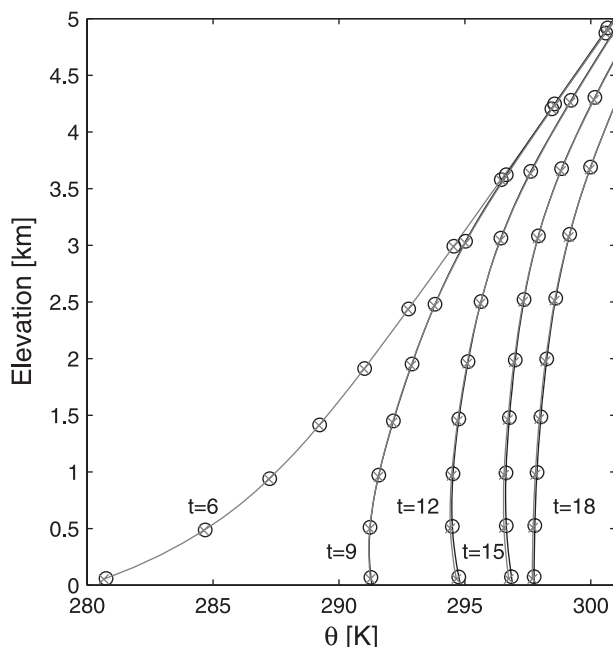


FIG. 11. Profiles of potential temperature (K) at the valley center for 0600, 0900, 1200, 1500, and 1800 UTC. The IBM-WRF is shown as a black line ( $\ominus$ ) and the WRF with terrain-following coordinates is shown in gray ( $\times$ ). Markers are placed at every fourth data point. The WRF and IBM-WRF solutions are virtually indistinguishable.

immersed boundary is negligible in comparison to variations resulting from terrain height. Shortwave radiation varies by about  $31 \text{ W m}^{-2}$  from valley peak to trough, with a maximum difference of  $0.54 \text{ W m}^{-2}$  between the simulations.

Domain-averaged heat and moisture fluxes from the land surface model are plotted in Fig. 17. The peak domain-averaged heat flux occurs at 1230 UTC. In the IBM-WRF model simulation this peak is  $193.80 \text{ W m}^{-2}$ ,

and in the WRF model simulation it is  $193.31 \text{ W m}^{-2}$ . The maximum difference between the two simulations occurs later in the day at 1430 UTC and is a difference of  $1.10 \text{ W m}^{-2}$  or 0.68%. Differences in moisture fluxes are also minimal. The maximum mean difference in the simulations occurs at 1230 UTC, and is  $5.39 \times 10^{-8} \text{ kg m}^{-2} \text{ s}^{-1}$  in magnitude, which is 0.82% of the WRF model value.

Figure 18 shows the spatial variation in the heat and moisture fluxes at three different times: 0900, 1200, and 1500 UTC. At 1200 UTC, spikes in the heat flux are seen near the valley peaks. The cause of this variation is the interpolation of variables to a modified reference height for use in the similarity parameters (as described in section 5). The modified reference height in the IBM case may lie between the immersed surface and the first grid point above the immersed surface or between the first two computational nodes above the surface. This interpolation difference occasionally leads to errors in the calculated heat flux in the IBM-WRF model simulations. These spikes in heat flux do not affect the flow in this case, and plots of the velocity fields look nearly identical between the uncoupled and coupled cases. This is demonstrated in the data given in Table 2, where the values and differences for the IBM and the WRF model simulations are given for the prognostic variables in both the uncoupled and coupled cases. The approximately 5% difference in  $u$  velocity is expected because of the differences in grids, application of the boundary conditions, and the added difficulty of forcing the flow at the immersed boundary rather than at a larger scale. The largest difference of 9.7% is seen in  $w$  when the values are small, and a small absolute difference leads to a large relative difference.

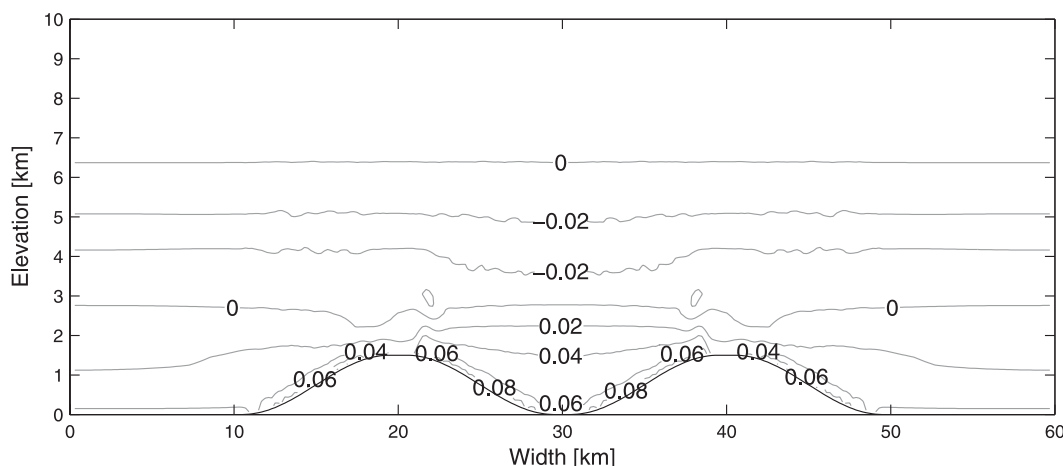


FIG. 12. Time-averaged differences (K) between the IBM-WRF and WRF for potential temperature. Differences are absolute.



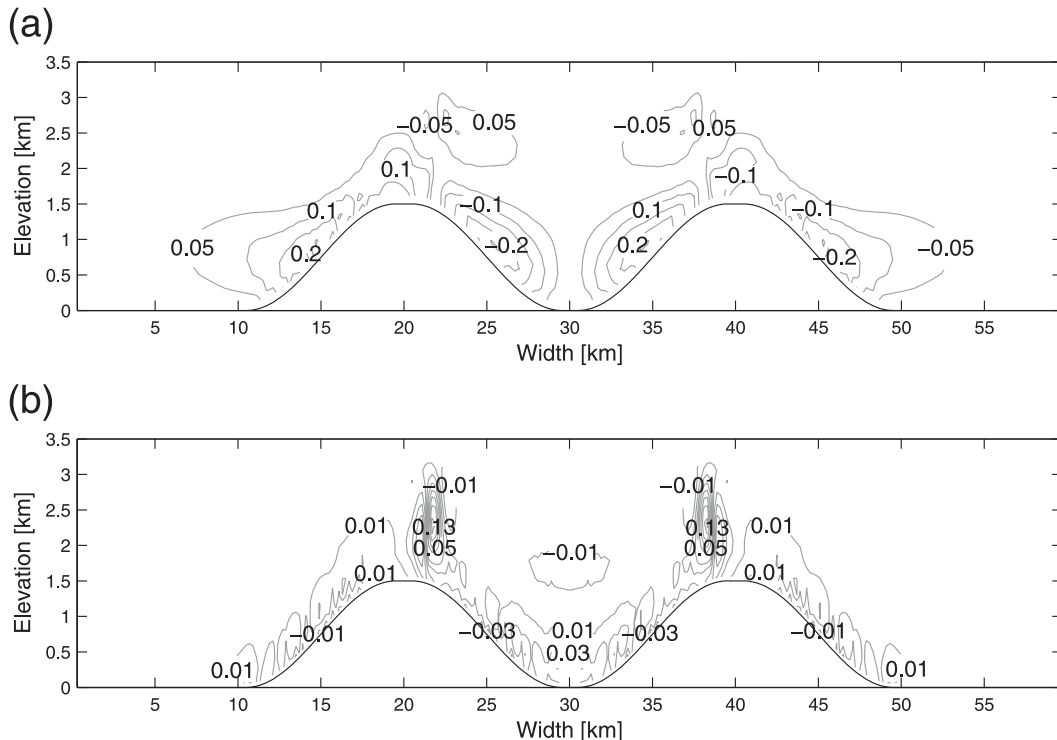


FIG. 13. Time-averaged differences between the IBM-WRF and the WRF for (top)  $u$  and (bottom)  $w$  velocity ( $\text{m s}^{-1}$ ). Differences are absolute. Contour intervals of 0.05 and 0.02  $\text{m s}^{-1}$  are used for  $u$  and  $w$ , respectively, with the zero contour suppressed.

The land surface models for soil moisture can be used independently of atmospheric simulations; however, when they are coupled, the top boundary conditions of the land surface model are set with inputs from the atmosphere. When the IBM is used, it provides boundary conditions to each model (WRF and Noah) simultaneously. A comparison of soil moisture is shown in Fig. 19 for the IBM-WRF and WRF simulations. At the initialization, the soil moisture has a constant volumetric water content of  $0.0868 \text{ m}^3 \text{ m}^{-3}$ . During the simulation, the soil begins to dry as moisture is transferred to the atmosphere. The depth at which this occurs is affected by the terrain height, and it can be seen that the soil remains slightly moister at the terrain peaks than at the valley floor and the terrain outside of the valley. Similar results are achieved in both of the simulations.

### c. Two-dimensional urban terrain simulations

The IBM allows resolution of urban terrain, enabling the WRF model to simulate flows that cannot be computed using a standard terrain-following coordinate. To demonstrate this capability, we have modeled flow over a two-dimensional slice of downtown New York City. The tallest building that is included has a height of

248 m. The domain size is  $(X, Y, Z) = (698, 4, 600 \text{ m})$  with  $\Delta x = 2 \text{ m}$ ,  $\Delta y = 2 \text{ m}$ , and  $\Delta z \approx 2.4 \text{ m}$ . The total number of grid points in each direction is  $(n_x, n_y, n_z) = (350, 3, 250)$ . At the initialization, the atmosphere is at rest and the flow is driven by a constant horizontal

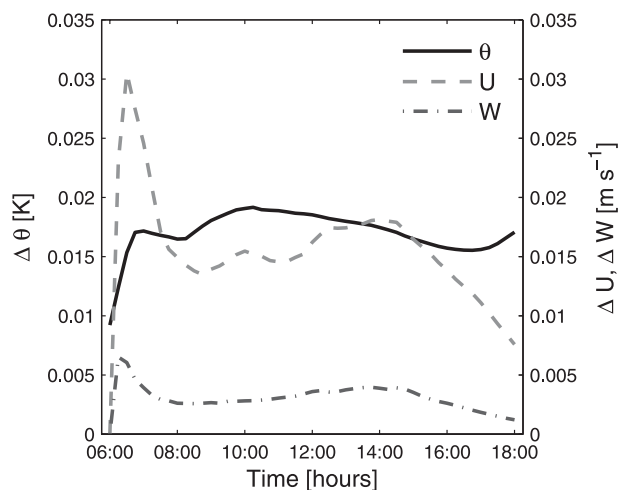


FIG. 14. Domain-averaged differences of  $\theta$ ,  $u$ , and  $w$  as a function of time.

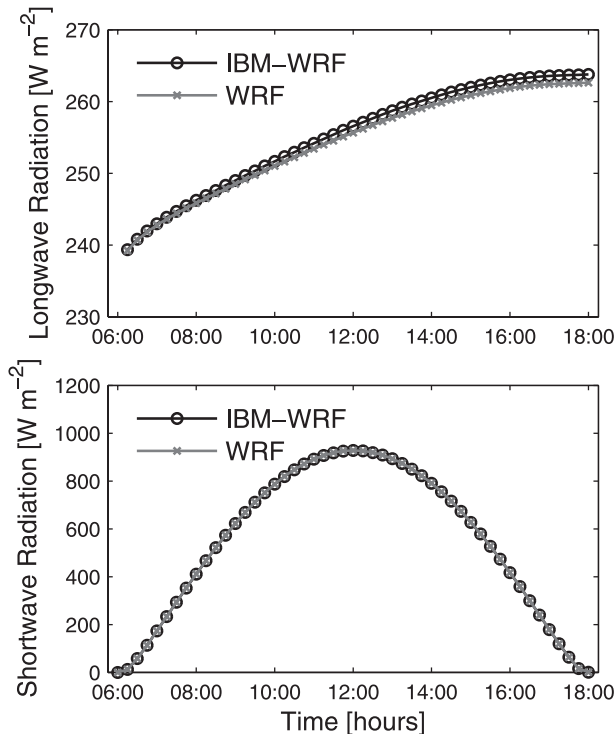


FIG. 15. Domain-averaged (top) downward longwave and (bottom) shortwave radiation for the coupled simulations. Markers are placed at 15-min intervals.

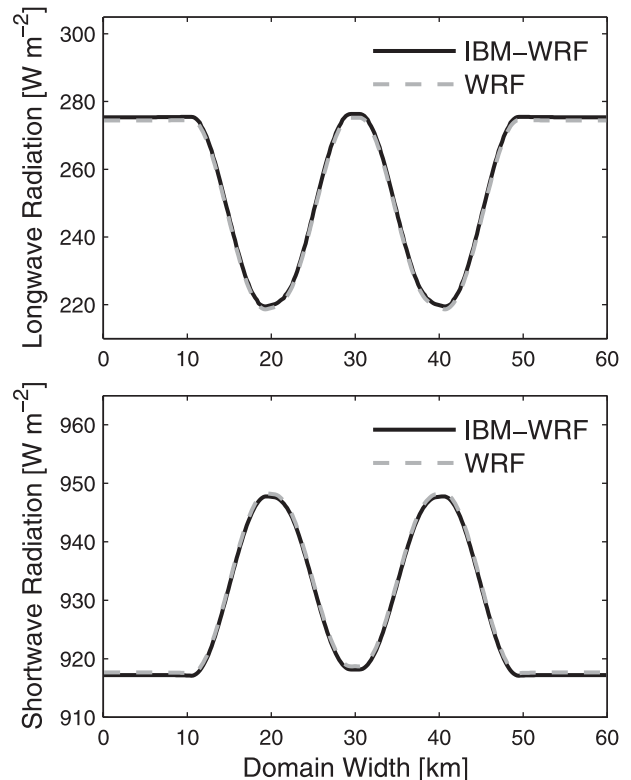


FIG. 16. Instantaneous spatial variation in (top) downward longwave and (bottom) shortwave radiation at 1200 UTC.

pressure gradient. Boundary conditions are identical to those previously used.

Figure 20 shows flow streamlines and the velocity magnitude at three different snapshots in time. After 30 min of flow spinup, recirculation regions form behind several of the buildings. After two hours, there is a large area of recirculation between the buildings and the flow velocities are fairly weak in these regions.

This case demonstrates the ability of our IBM algorithm to handle a wide variety of geometric cases while retaining numerical stability. Several configurations of ghost points and nearest neighbors that were not present in the idealized hill and valley cases are tested in this case. As the isobaric coordinates evolve, it can be challenging to maintain a well-conditioned interpolation matrix in Eq. (7); however, this case demonstrates that we have developed a capable and robust IBM algorithm.

The focus of this work is the development and accuracy of boundary conditions at the immersed surface; however, for complex cases, such as with urban terrain, there are many factors that impact the accuracy of the simulation beyond the treatment of the lower boundary. Subfilter-scale turbulence models and near-wall stress models play a large role in obtaining accurate solutions of the flow in complex terrain. This is true for all types of

grids, regardless of being body or nonbody conforming. Some of these additional challenges for high-resolution simulations of complex flows in the WRF model have been addressed in the work of Mirocha et al. (2007) and Lundquist et al. (2007), where the use of improved turbulence models is investigated.

## 7. Summary and conclusions

We have developed an immersed boundary method for the WRF model, which is sufficiently flexible to work with the time variant isobaric  $\sigma$  coordinate that is native to the WRF and accurately enforces Dirichlet and Neumann boundary conditions within the mode-splitting time-integration scheme. A unique interpolation scheme is used for the IBM, which avoids problematic numerical instabilities and the need for an iterative procedure to decouple the interior and exterior solutions as in previous methods (Tseng and Ferziger 2003; Mittal et al. 2008). The canonical cases of pressure-driven flow over a hill and thermally driven flow in a valley were used to validate the implementation of the IBM by comparing the results to those achieved with the native WRF model terrain-following coordinate. In the hill case, the domain-averaged differences between the simulations at steady

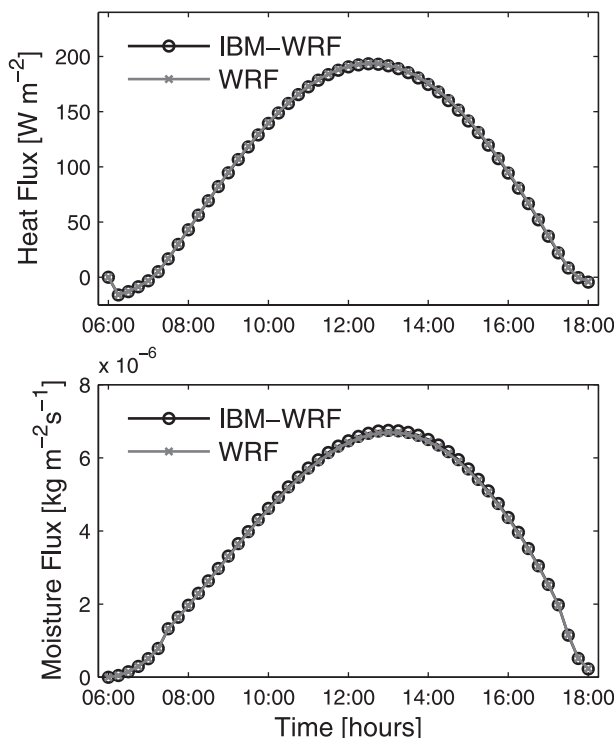


FIG. 17. Domain-averaged upward (top) heat and (bottom) moisture flux for the coupled simulations. Markers are placed at 15 min intervals.

state were  $\Delta U_{\text{ave}} = 0.017$  and  $\Delta W_{\text{ave}} = 6 \times 10^{-4} \text{ m s}^{-1}$ . In the valley case, the domain-averaged differences were  $\Delta U_{\text{ave}} = 0.010$  and  $\Delta W_{\text{ave}} = 0.002 \text{ m s}^{-1}$  (uncoupled) and  $\Delta U_{\text{ave}} = 0.011$  and  $\Delta W_{\text{ave}} = 0.002 \text{ m s}^{-1}$  (coupled), when averaged over the duration of the daytime diurnal cycle. These errors are remarkably small when considering that there were unavoidable differences in the grids and the application of boundary conditions in the WRF and IBM-WRF cases. This success was only achieved after careful development of our IBM within the WRF computational framework. Considerations were made to accommodate the fully compressible Navier-Stokes equations, the transformation of these equations into time-variant pressure-based  $\sigma$  coordinates, and the discretization of the equations with a time-split integration scheme. A simulation of flow over urban terrain demonstrated the ability of the IBM-WRF to handle highly complex terrain, which could not be simulated using  $\sigma$  coordinates.

Our IBM-WRF model implementation is capable of handling highly complex terrain while maintaining atmospheric parameterizations included in the WRF mesoscale model. We coupled a set of surface parameterizations to the immersed boundary, providing realistic land surface forcing. These parameterizations may be

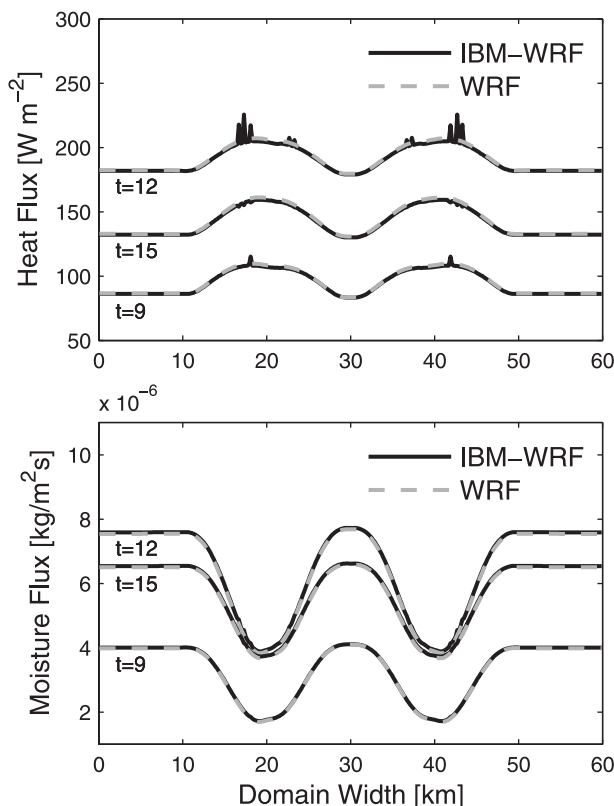


FIG. 18. Spatial variation in (top) upward heat and (bottom) moisture flux at 0900, 1200, and 1500 UTC.

appropriate for some scales where IBM is used, such as in mountainous terrain, but may need to be modified in urban terrain (where traditional similarity theory may not

TABLE 2. The maximum and minimum values of prognostic variables for the uncoupled and coupled idealized valley cases at noon. Differences are instantaneous and calculated at the location of the indicated maximum or minimum value. Here, the units are for velocity ( $\text{m s}^{-1}$ ), potential temperature (K), and the water vapor mixing ratio ( $\text{kg kg}^{-1}$ ).

		IBM	WRF	Difference	
				Absolute	Relative (%)
Uncoupled	$u_{\min}$	-2.029	-1.913	-0.116	6.1
	$u_{\max}$	2.029	1.913	0.116	6.1
	$w_{\min}$	-0.226	-0.243	0.018	7.4
	$w_{\max}$	1.400	1.369	0.032	2.3
	$\theta_{\min}$	293.689	293.684	0.005	0
	$\theta_{\max}$	316.337	316.328	0.009	0
Coupled	$u_{\min}$	-2.011	-1.912	-0.098	5.1
	$u_{\max}$	2.011	1.912	0.098	5.1
	$w_{\min}$	-0.194	-0.215	0.021	9.0
	$w_{\max}$	1.198	1.253	-0.054	4.3
	$\theta_{\min}$	293.149	293.135	0.015	0
	$\theta_{\max}$	316.338	316.328	0.010	0
	$q_{v\min}$	$1 \times 10^{-4}$	$1 \times 10^{-4}$	$2 \times 10^{-7}$	0.2
	$q_{v\max}$	0.002	0.002	$7 \times 10^{-7}$	0

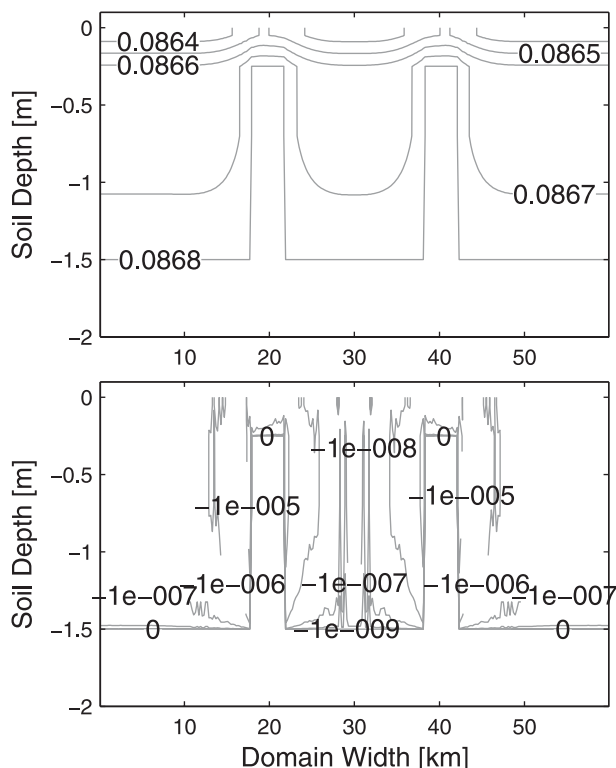


FIG. 19. (top) Volumetric soil moisture ( $\text{m}^3 \text{m}^{-3}$ ) at the end of the IBM–WRF simulation (1800 UTC). This field was initialized with a constant saturation rate. (bottom) The absolute difference between the IBM and the WRF simulations for volumetric soil moisture.

be applicable). Regardless, the capability exists to couple the IBM–WRF model to surface parameterizations when they are available. This means that the IBM–WRF can be used in a grid-nesting configuration to accommodate synoptic-scale lateral boundary conditions in an urban environment, with grid resolution fine enough to explicitly resolve flow around buildings. We can therefore combine the best properties of CFD (capable of resolving buildings) and mesoscale codes (capable of handling atmospheric physics and boundary conditions). A standard terrain-following WRF model simulation can be used at mesoscales, with a seamless transition to the IBM–WRF model at finer scales. Furthermore, IBM eliminates grid distortion, which is present high above the land surface when using  $\sigma$  coordinates over steep terrain.

The drawbacks to this approach are that it can be expensive to maintain fine near-surface grid resolution of the immersed boundary at elevated heights, such as at mountain peaks. The hybrid  $\sigma$  and IBM coordinate may reduce this constraint by allowing gentle slopes in topography to be represented with  $\sigma$  coordinates, whereas steep slopes are represented with the IBM. This could allow for increased resolution near the ground

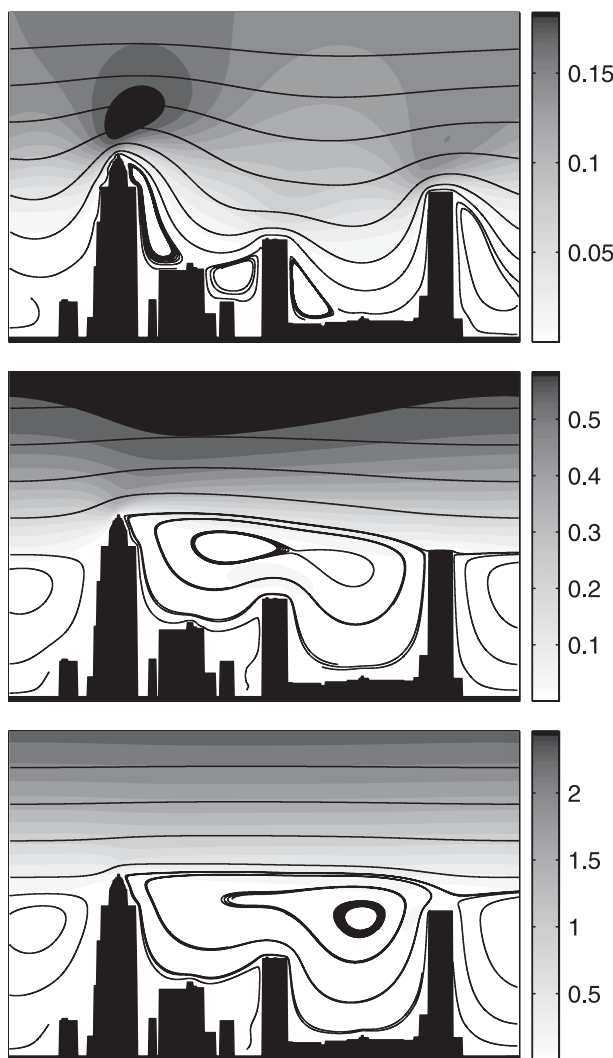


FIG. 20. Velocity contours ( $\text{m s}^{-1}$ ) and streamlines of startup flow over 2D building data from New York City using the IBM–WRF. Three different times are shown: (top) 30 min, (middle) 2 h, and (bottom) 10 h.

while minimizing the number of nodes below the terrain. There are increased computational costs associated with the IBM algorithm; however, the IBM allows simulations of flow over terrain that could not otherwise be handled by the WRF model. Opportunities exist for optimization as the code is developed further.

The ability to seamlessly integrate IBM simulations within a mesoscale model with full atmospheric physics is a significant milestone in the progress of atmospheric boundary layer simulations. The development of the IBM–WRF model enables a range of applications, including building-resolving simulations of urban dispersion for emergency response efforts or even urban air quality modeling. Similarly, simulations of flow in highly complex mountainous terrain with near-vertical slopes



can now be accommodated without compromising the accuracy or stability of the solution. The extension to three dimensions has been relatively straightforward in our preliminary tests and simply requires a larger interpolation stencil for the IBM algorithm. Incorporation of similarity theory for surface momentum fluxes is also in progress (Lundquist 2006) and will allow even more realistic application to atmospheric boundary layer flows.

**Acknowledgments.** Thanks are extended to Jeff Mirocha and Branko Kosović for many useful discussions regarding this work. The authors are grateful for the support of a Lawrence Scholars Program Fellowship from Lawrence Livermore National Laboratory (LLNL). This project was partially funded by the LLNL Laboratory-Driven Research and Development program. The support of National Science Foundation Grant ATM-0645784 [FKC] (Physical Meteorology Program: S. Nelson, Program Director) is also gratefully acknowledged. This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

#### REFERENCES

- Adcroft, A., C. Hill, and J. Marshall, 1997: Representation of topography by shaved cells in a height coordinate ocean model. *Mon. Wea. Rev.*, **125**, 2293–2315.
- Balaras, E., 2004: Modeling complex boundaries using an external force field on fixed Cartesian grid in large-eddy simulations. *Comput. Fluids*, **33**, 375–404.
- Choi, J., R. Oberoi, J. Edwards, and J. Rosati, 2007: An immersed boundary method for complex incompressible flows. *J. Comput. Phys.*, **224**, 757–784.
- Cristallo, A., and R. Verzicco, 2006: Combined immersed boundary/large-eddy-simulations of incompressible three-dimensional complex flows. *Flow Turbul. Combust.*, **77**, 3–26.
- Fadlun, E., R. Verzicco, P. Orlandi, and J. Mohd-Yusof, 2000: Combined immersed boundary finite-difference methods for three-dimensional complex flow simulations. *J. Comput. Phys.*, **161**, 35–60.
- Fast, J., 2003: Forecasts of valley circulations using the terrain-following and step-mountain vertical coordinates in the Meso-Eta model. *Wea. Forecasting*, **18**, 1192–1206.
- Gao, T., Y. Tseng, and X. Lu, 2007: An improved hybrid Cartesian/immersed boundary method for fluid-solid flows. *Int. J. Numer. Methods Fluids*, **55**, 1189–1211.
- Ge, L., and F. Sotiropoulos, 2007: A numerical method for solving the 3D unsteady incompressible Navier–Stokes equations in curvilinear domains with complex immersed boundaries. *J. Comput. Phys.*, **225**, 1782–1809.
- Ghias, R., R. Mittal, and H. Dong, 2007: A sharp interface immersed boundary method for compressible viscous flows. *J. Comput. Phys.*, **225**, 528–553.
- Gilmanov, A., F. Sotiropoulos, and E. Balaras, 2003: A general reconstruction algorithm for simulating flows with complex 3D immersed boundaries on Cartesian grids. *J. Comput. Phys.*, **191**, 660–669.
- Hanna, S., and Coauthors, 2006: Detailed simulations of atmospheric flow and dispersion in downtown Manhattan: An application of five computational fluid dynamics models. *Bull. Amer. Meteor. Soc.*, **87**, 1713–1726.
- Iaccarino, G., and R. Verzicco, 2003: Immersed boundary technique for turbulent flow simulations. *Appl. Mech. Rev.*, **56**, 331–347.
- Janjic, Z., 1977: Pressure gradient force and advection scheme used for forecasting with steep and small scale topography. *Beitr. Phys. Atmos.*, **50**, 186–189.
- Kirkpatrick, M., S. Armfield, and J. Kent, 2003: A representation of curved boundaries for the solution of the Navier–Stokes equations on a staggered three-dimensional Cartesian grid. *J. Comput. Phys.*, **184**, 1–36.
- Klemp, J., W. Skamarock, and O. Fuhrer, 2003: Numerical consistency of metric terms in terrain-following coordinates. *Mon. Wea. Rev.*, **131**, 1229–1239.
- , —, and J. Dudhia, 2007: Conservative split-explicit time integration methods for the compressible nonhydrostatic equations. *Mon. Wea. Rev.*, **135**, 2897–2913.
- Laprise, R., 1992: The euler equations of motion with hydrostatic pressure as an independent variable. *Mon. Wea. Rev.*, **120**, 197–207.
- Lundquist, J. K., F. Chow, J. Mirocha, and K. Lundquist, 2007: An improved WRF for urban-scale and complex-terrain applications. Preprints, *Seventh Symp. on the Urban Environment*, San Diego, CA, Amer. Meteor. Soc., 4.4. [Available online at [http://ams.confex.com/ams/7Coastal7Urban/techprogram/paper\\_127144.htm](http://ams.confex.com/ams/7Coastal7Urban/techprogram/paper_127144.htm).]
- Lundquist, K. A., 2006: Implementation of the immersed boundary method in the weather research and forecasting model. M.S. thesis, Department of Mechanical Engineering, University of California, Berkeley, 59 pp.
- , F. K. Chow, and J. K. Lundquist, 2008: An immersed boundary method for flow over complex terrain. Preprints, *13th Conf. on Mountain Meteorology*, Whistler, BC, Amer. Meteor. Soc., 9A.5. [Available online at <http://ams.confex.com/ams/pdfpapers/141221.pdf>.]
- Mahrer, Y., 1984: An improved numerical approximation of the horizontal gradients in a terrain-following coordinate system. *Mon. Wea. Rev.*, **112**, 918–922.
- Mirocha, J., F. Chow, J. Lundquist, and K. Lundquist, 2007: Improved subfilter turbulence modeling for large-eddy simulation using WRF. Preprints, *Seventh Symp. on the Urban Environment*, San Diego, CA, Amer. Meteor. Soc. 13.1. [Available online at [http://ams.confex.com/ams/7Coastal7Urban/techprogram/paper\\_127107.htm](http://ams.confex.com/ams/7Coastal7Urban/techprogram/paper_127107.htm).]
- Mittal, R., and G. Iaccarino, 2005: Immersed boundary methods. *Annu. Rev. Fluid Mech.*, **37**, 239–261.
- , H. Dong, M. Bozkurtas, F. Najjar, A. Vargas, and A. von Loebbecke, 2008: A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries. *J. Comput. Phys.*, **227**, 4825–4852.
- Mohd-Yusof, J., 1997: Combined immersed boundary/b-spline methods for simulations of flow in complex geometry. Center for Turbulence Research, Annual Research Briefs, NASA Ames/Stanford University, 317–327.
- Peskin, C., 1972: Flow patterns around heart valves: A numerical method. *J. Comput. Phys.*, **10**, 252–271.
- Phillips, N., 1957: A coordinate system having some special advantages for numerical forecasting. *J. Meteor.*, **14**, 184–185.

- Rampanelli, G., D. Zardi, and R. Rotunno, 2004: Mechanisms of up-valley winds. *J. Atmos. Sci.*, **61**, 3097–3111.
- Saiki, E., and S. Biringen, 1996: Numerical simulation of a cylinder in uniform flow: Application of a virtual boundary method. *J. Comput. Phys.*, **123**, 450–465.
- Schär, C., D. Leuenberger, O. Fuhrer, D. Lüthi, and C. Girard, 2002: A new terrain-following vertical coordinate formulation for atmospheric prediction models. *Mon. Wea. Rev.*, **130**, 2459–2480.
- Schmidli, J., and Coauthors, 2008: The T-REX valley wind model intercomparison project. Preprints, *13th Conf. on Mountain Meteorology*, Whistler, BC, Amer. Meteor. Soc., 5B.1. [Available online at [http://ams.confex.com/ams/13MontMet17AP/techprogram/paper\\_140760.htm](http://ams.confex.com/ams/13MontMet17AP/techprogram/paper_140760.htm).]
- Shi, R., G. Cui, Z. Wang, C. Xu, and Z. S. Zhang, 2008: Large-eddy simulation of wind field and plume dispersion in building array. *Atmos. Environ.*, **42**, 1083–1097.
- Skamarock, W., J. Klemp, J. Dudhia, D. Gill, D. Barker, W. Wang, and J. Powers, 2007: A description of the advanced research WRF version 2. NCAR Tech. Note NCAR/TN-468+STR, 88 pp.
- Smolarkiewicz, P., R. Sharman, J. Weil, S. Perry, D. Heist, and G. Bowker, 2007: Building resolving large-eddy simulations and comparison with wind tunnel experiments. *J. Comput. Phys.*, **227**, 633–653.
- von Terzi, D. A., M. N. Linnick, J. Seidel, and H. F. Fasel, 2001: Immersed boundary techniques for high-order finite-difference methods. American Institute of Aeronautics and Astronautics, Rep. AIAA-2001-2918, 17 pp.
- Tseng, Y., and J. Ferziger, 2003: A ghost-cell immersed boundary method for flow in complex geometry. *J. Comput. Phys.*, **192**, 593–623.
- , C. Meneveau, and M. Parlange, 2006: Modeling flow around bluff bodies and predicting urban dispersion using large eddy simulation. *Environ. Sci. Technol.*, **40**, 2653–2662.
- Wicker, L., and W. Skamarock, 2002: Time-splitting methods for elastic models using forward time schemes. *Mon. Wea. Rev.*, **130**, 2088–2097.
- Yasuda, N., 1988: Turbulent diffusivity and diurnal variation in the atmospheric boundary layer. *Bound.-Layer Meteor.*, **43**, 209–221.