



## RESEARCH ARTICLE

10.1029/2024MS004494

## Key Points:

- Cut-cell method, a higher-order geometry representation, and a near-surface velocity gradient correction are implemented in Quick Environmental System-Winds
- Cut-cell method results are mostly independent of grid sizes and represent the wind field more accurately than stair-step, especially at lower resolutions
- The near-wall parameterization improves the velocity gradient near the surface which could lead to better estimation of scalar transport

## Correspondence to:

E. R. Pardyjak,  
pardyjak@eng.utah.edu

## Citation:

Bozorgmehr, B., Willemssen, P., Gibbs, J. A., Stoll, R., Kim, J.-J., Patterson, Z., & Pardyjak, E. R. (2025). Improving the performance of a reduced-order mass-consistent model for urban environments and complex terrain with a higher-order geometrical representation. *Journal of Advances in Modeling Earth Systems*, 17, e2024MS004494. <https://doi.org/10.1029/2024MS004494>

Received 5 JUN 2024

Accepted 31 MAR 2025

## Author Contributions:

**Conceptualization:** Eric R. Pardyjak  
**Methodology:** Eric R. Pardyjak  
**Writing – review & editing:** Eric R. Pardyjak

© 2025 The Author(s). Journal of Advances in Modeling Earth Systems published by Wiley Periodicals LLC on behalf of American Geophysical Union. This is an open access article under the terms of the [Creative Commons Attribution-NonCommercial-NoDerivs License](#), which permits use and distribution in any medium, provided the original work is properly cited, the use is non-commercial and no modifications or adaptations are made.

# Improving the Performance of a Reduced-Order Mass-Consistent Model for Urban Environments and Complex Terrain With a Higher-Order Geometrical Representation

Behnam Bozorgmehr<sup>1</sup>, Pete Willemssen<sup>2</sup>, Jeremy A. Gibbs<sup>1,3</sup> , Rob Stoll<sup>1</sup>, Jae-Jin Kim<sup>4</sup>, Zachary Patterson<sup>2</sup>, and Eric R. Pardyjak<sup>1</sup> 

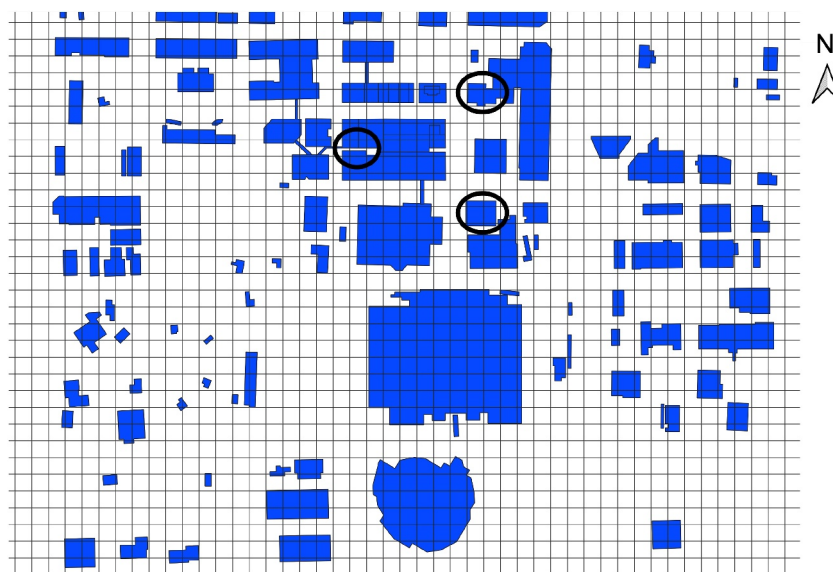
<sup>1</sup>Department of Mechanical Engineering, University of Utah, Salt Lake City, UT, USA, <sup>2</sup>Department of Computer Science, University of Minnesota Duluth, Duluth, MN, USA, <sup>3</sup>National Severe Storms Laboratory, National Oceanic and Atmospheric Administration, Norman, OK, USA, <sup>4</sup>Department of Environmental Atmospheric Sciences, Pukyong National University, Busan, South Korea

**Abstract** Solid structures (buildings and topography) act as obstacles and significantly influence the wind flow. Because of their importance, faithfully representing the geometry of structures in numerical predictions is critical to modeling accurate wind fields. A higher-order geometry representation (the cut-cell method) is incorporated in the mass-consistent wind model, Quick Environmental System (QES)-Winds. To represent the differences between a stair-step and the cut-cell method, an urban case study (the Oklahoma City JU2003 experiments) and a complex terrain case (from the MATERHORN campaign) are modeled in QES-Winds. Comparison between the simulation results with the stair-step and cut-cell methods and the measured data for sensors close to walls and buildings showed that the sensitivity of the cut-cell method to changes in resolution is less than the stair-step method. Another way to improve the effects of solid geometries on the flow is to correct the velocity gradient near the surface. QES-Winds solves a conservation of mass equation and not a conservation of momentum equation. This means that QES-Winds overestimates velocity gradients near the surface which leads to higher rates of scalar transport. The near-surface parameterization is designed to correct the tangential near-surface velocity component using the logarithmic assumption. Results, including the near-wall parameterization, are evaluated with data from the Granite Mountain case (the MATERHORN campaign), which indicates that the parameterization slightly improves the performance of the model for cells near the surface. The new geometry representation and near-wall parameterization added to a mass-consistent platform, enhances the model's ability to simulate the effects of solid geometries on wind fields.

**Plain Language Summary** Solid geometries (buildings and the earth's surface) shape the airflow in cities and over topography. To accurately model the wind field, these important structures need to be represented correctly in numerical models. Quick Environmental System (QES)-Winds previously used a stair-step or block-building method to process solid geometries. To improve geometry representation in QES-Winds, a cut-cell (higher-order) method is developed in QES-Winds. The cut-cell method provides the ability to have cells that are partially occupied by solid geometries. Three test cases are utilized in this research to show the differences between the stair-step and cut-cell methods. The cut-cell results are more consistent over a range of grid sizes which means that higher resolution is not required for accurate modeling. Solid structures greatly influence the rate of change of wind speed near the surface. QES-Winds does not include the momentum effects. As a result, QES-Winds overestimates the rate of change of velocity near the surface which means higher scalar transport from the surface. A near-wall parameterization is incorporated into QES-Winds to modify the velocity gradient near the surface and improve the transport rate. The results for the complex terrain (Granite Mountain) test case showed that the parameterization improves the model performance near the solid surface.

## 1. Introduction

Simulating accurate wind fields over hilly and mountainous terrain and in urban areas is crucial for modeling wildfire physics (Balbi et al., 2020; Moody et al., 2022), dispersion of diseases in agriculture (Ulmer et al., 2023) and air quality in cities (Pardyjak & Brown, 2001; Singh et al., 2008; Williams et al., 2004). Buildings are the most important elements in urban environments, while topography is significant for urban areas and determines the rate of wildfires spread in complex terrain. These solid elements affect the wind flow around them greatly. To



**Figure 1.** Building footprint for the central business district in Oklahoma City with a Cartesian grid on top. The figure illustrates how actual building geometries rarely line up with the simulation grid. Regions circled in black show particularly problematic situations.

improve wind-field modeling in the aforementioned applications, one option is to enhance the physics modules that capture the effects of solid elements on the flow. Another option is to capture every little detail of the building and terrain geometries and implement their effects on the flow field (e.g., modeling the gap between buildings and steep terrain).

Modeling high-resolution wind fields is required for the prediction and optimization of the aforementioned applications. Traditional computational fluid dynamics (CFD) models, such as Reynolds-averaged Navier-Stokes simulations and large-eddy simulations (LES) are expensive and have long execution times (Hayati et al., 2019). Another option is to use a semi-empirical fast-response approach, in which physics-based details are compromised for shorter run-times.

Quick Environmental System (QES)-Winds (Bozorgmehr, Willemsen, Gibbs, et al., 2021; Bozorgmehr, Willemsen, Margairaz, et al., 2021) is a fast-response wind modeling platform. QES-Winds simulates mean wind fields that have been used in several fast-response applications such as modeling wildfire spread (Moody et al., 2022), dispersion of pollution in urban areas (Margairaz et al., 2023; Pardyjak & Brown, 2001; Singh et al., 2008; Williams et al., 2004), and propagation of disease in agricultural fields (Ulmer et al., 2023). There are other mass-consistent wind-modeling software packages similar to QES-Winds that have been described in the literature. Most prominent of these models are WindNinja (Forthofer et al., 2014; Sherman, 1978), Micro SWIFT (Moussafir et al., 2004; Tinarelli et al., 2007), and CALMET (Scire et al., 2000; W. Wang et al., 2008). WindNinja (Forthofer et al., 2014) and CALMET (Scire et al., 2000; W. Wang et al., 2008) yield mass conservative solutions using a terrain-following coordinate system. Micro SWIFT ensures the conservation of mass by solving the Poisson equation on a three-dimensional Cartesian grid (Moussafir et al., 2004; Tinarelli et al., 2007). Micro SWIFT utilizes a stair-step or the block-building method for representing building geometry. Micro SWIFT is not capable of simulating wind flow over complex terrain since it does not process terrain geometry.

QES-Winds utilizes a 3D Cartesian grid to discretize the computational domain. Most often, the grid does not match building edges, even for rectangular buildings, as shown in Figure 1. The situation can be more complex for topography, where the terrain geometry rarely matches the grid. QUIC-URB (Pardyjak & Brown, 2003) and QES-Winds use a “block-building” or “stair-step” method for representing geometry in which the center of each cell is examined to be either in a solid or air. If the center is inside a solid (building or terrain), the whole cell is considered solid, while if the center is in air, the cell is considered air. The stair-step method results in losing

geometrical details as illustrated by the regions inside the black circles in Figure 1. The situation is worse for wind fields over complex terrain. One option to improve the geometry representation in the stair-step method is to increase the resolution of the domain. This means more cells and slower simulations for large domains. Another option is to add higher-order geometrical representations of both building and terrain. Here, we present a higher-order geometry representation approach as an alternative to the stair-step method as the first attempt in the literature to use improved geometry representation in a mass-consistent wind solver.

The immersed boundary method (IBM) was first introduced by Peskin (1972) to simulate cardiac flow. Since then, several modifications and adjustments have been investigated and variants of the approach have been used frequently in CFD (Marella et al., 2005; Mittal & Iaccarino, 2005; Sussman et al., 1999; Udaykumar et al., 1996; Ye et al., 1999) and fluid-structure interaction applications (Hou et al., 2012; Yang & Balaras, 2006). IBMs have been utilized to simulate environmental flows (Arthur et al., 2018, 2020; Bao et al., 2018; Chester et al., 2007; DeLeon et al., 2018; Finnigan et al., 2020; Lundquist et al., 2010, 2012; Salesky et al., 2017, 2019; Umphrey et al., 2017) and shallow-water oceanic flows, for example, shaved-cell approach (Ketefian & Jacobson, 2009, 2011), and Brinkman volume penalization (Kevlahan et al., 2015) to represent the complex geometry as a boundary condition. In all the above-mentioned cases, instead of using a terrain-following vertical grid (where the vertical grid conforms to the terrain), the terrain is represented as an immersed surface cut through the domain grid arbitrarily. Boundary conditions related to the terrain are applied by modifying the governing equations along the immersed surface.

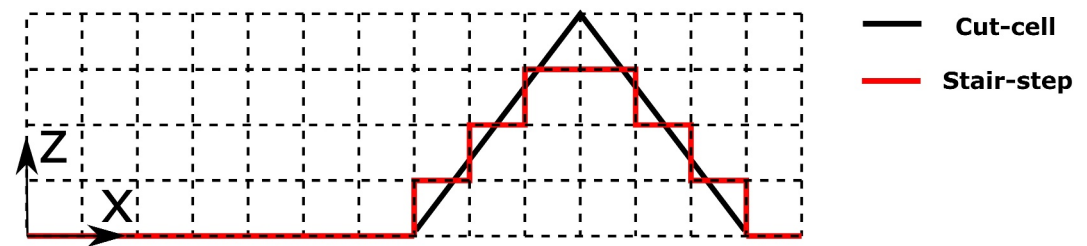
IBMs used in CFD can be categorized into two classes (based on Udaykumar et al. (2001)): First, the immersed boundary can be represented as a diffuse interface and can be smeared over several cells. This class consists of the methods that utilize body (Finnigan et al., 2020; Peskin, 1972; Umphrey et al., 2017) or surface forces (Chester et al., 2007; DeLeon et al., 2018), to represent the effect of the immersed boundary. Approaches that use the volume-of-fluid (VOF) method (Scardovelli & Zaleski, 1999; Yuan & Piomelli, 2014a, 2014b) and phase-field (Anderson et al., 2000; Wang et al., 1993) techniques, are also included in this class. In the second category, the immersed boundary is tracked as a sharp interface in which the boundary condition is applied by modifying the computational grid near the interface (Almgren et al., 1997; Marella et al., 2005; Udaykumar et al., 1996, 2001; Ye et al., 1999). In this method, the computational domain is divided into two separate regions by the sharp interface and, there is a distinct jump in material and flow features between the two sub-domains.

Velocity and shear-stress reconstruction methods, in which the momentum flux at the interface is parameterized using the near-wall modeling, are IBMs that replace the terrain-following scheme in the weather research and forecasting model (Arthur et al., 2018, 2020; Bao et al., 2018). Terrain-following coordinate systems can cause numerical instabilities when the terrain becomes too steep, generating highly skewed grids (Arthur et al., 2018, 2020). IBMs were first used in atmospheric models to resolve the aforementioned issue (Arthur et al., 2018, 2020). DeLeon et al. (2018) used a hybrid method of velocity and shear stress reconstruction in their LES model to simulate turbulent flow over complex terrain. They reconstructed both the velocity and the eddy-viscosity in the wall-normal direction while reconstructing the tangential velocity component using the log-law assumption.

The cut-cell approach (a simplified IBM method) was first introduced by Clarke et al. (1986) for simulating inviscid flow and was later applied to modeling viscous flows (Udaykumar et al., 1996, 2001, 2002; Ye et al., 1999). The Cartesian cut-cell methodology, which provides the capability of having cells partially cut by solid surfaces, has been used in the literature to handle complex terrain and topography (Dang et al., 2017; Ingram et al., 2003; Salesky et al., 2017, 2019; Tucker & Pan, 2000).

In their review of IBM approaches, Mittal and Iaccarino (2005) concluded that the only method in which conservation of mass can be satisfied in the cells near the immersed boundary is the Cartesian cut-cell method. The other methods mentioned are designed to be used in solutions to the Navier-Stokes equations. Because of the underlying Cartesian grid, the cut-cell method is simple to implement. Also, the Cartesian grid, which is required to resolve buildings and building parameterization, makes the cut-cell the best approach to be applied to both terrain and building geometries. As a result, the Cartesian cut-cell method has been added to QES-Winds as the higher-order geometrical representation scheme of choice.

In this paper, effects of a higher-order geometry representation and a near-surface velocity correction on wind fields simulated by a mass-consistent fast-response wind solver (QES-Winds) are evaluated. Details of the cut-



**Figure 2.** Schematic illustrating the differences between cut-cell and stair-step meshing schemes on a uniform grid for a triangular terrain feature.

cell method and how it is applied to the building and terrain geometries are described. This is followed by the presentation of the theory and application of a new near-wall parameterization for terrain that creates more realistic velocities close to the surface. Results of simulations with the stair-step and cut-cell methods are compared to the measurement data for the Askervein Hill (topography) and the Oklahoma City (urban) test cases. We also analyze differences resulting from the application of the stair-step and cut-cell methods, as well as the near-wall parameterization, for the Granite Mountain, a complex terrain test case.

## 2. Methods

QES-Winds uses a type of data assimilation technique (Sasaki, 1958, 1970a, 1970b) to minimize differences between the initial and final velocity fields and to ensure the conservation of mass, based on a model proposed by Röckle (1990). The successive over-relaxation (SOR) method is incorporated to solve the Poisson equation required for assuring the conservation of mass. While QES-Winds does not solve a conservation of momentum equation explicitly, momentum effects (e.g., shear and advection) effects are added to the solution by using empirical parameterizations. Although QES-Winds utilizes reduced physics to model wind flow, the results of fast numerical simulations compare quite well with higher order physics-based models in both idealized (Hayati et al., 2017) and realistic urban cities (Neophytou et al., 2011).

In this section, the cut-cell method implementation is explained, from processing the building and terrain information and calculating the area fraction coefficients to applying the area fraction coefficients to the QES-Winds solver. In the last part of this section, the theory and application of a new near-wall parameterization are described.

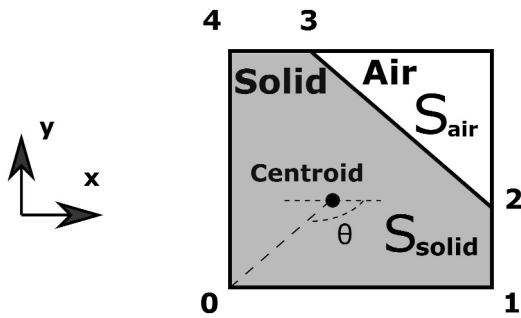
### 2.1. Cut-Cell Method

Figure 2 is a two-dimensional schematic showing how the stair-step and cut-cell methods represent a simple triangular geometry. The stair-step method simplifies the slope to steps, while the cut-cell method captures the actual geometry.

To apply the cut-cell method, an area fraction coefficient for each cell face needs to be calculated. Figure 3 shows a schematic of a cell's face cut by a solid geometry (e.g., mountain terrain or a building). Following the method designed in Salesky et al. (2017, 2019), we first need to define vertices forming the polygon that circumscribes the geometry either on the air or solid side of the cell face. In QES-Winds, the building and terrain data are imported from external files for processing, making it natural to determine points on the solid side for each cell face (points 0 to 4 in Figure 3). The procedure of processing the building and terrain information to determine the list of solid vertices for each cell face is explained in Appendix A and Section 2.1.2, respectively.

#### 2.1.1. Processing Building Geometry

A new shapefile reader function has been added to QES-Winds, which provides the capacity to load Environmental Systems Research Institute (ESRI) shapefiles using Geospatial Data Abstraction Library (GDAL; <https://www.gdal.org>). After the building footprints and heights are loaded from ESRI shapefiles, QES-Winds creates polygon buildings from the imported data. ESRI shapefiles contain building footprints in the shape of general polygons and the height associated with each polygon. We assume that the building walls are vertical (in the  $z$ -direction) and therefore, only the footprint polygon is evaluated to find the intersection points in the first cell.



**Figure 3.** 2D schematic of a cut-cell face in which part of the face is composed of a solid and air part.  $S_{air}$  and  $S_{solid}$  are the air and solid area fractions of the cell face shown. The centroid is the center point of the solid polygon (points 0 to 4). Points 0 to 4 are the intersection of the solid geometry with the cell face.  $\theta$  is defined as the angle of the line connecting point 0 to the centroid, with respect to the horizontal line through the centroid. The angle is positive if the connecting line is above the horizontal line passing through the centroid and negative if below the horizontal line.

Figure 4 displays a schematic of a random building footprint and the way it cuts through the underlying Cartesian grid cells. Figures 5 and 6 show a 2D schematic of the bottom face and a schematic of a vertical array of cells cut through by a building geometry, respectively. Figures 5 and 6 belong to the case in which the building geometry cuts through the bottom, top, left, and right faces (similar to the Case 5 in Figure 4). Details of different ways the building cuts through cells are provided in Appendix A, alongside the procedure of defining the building points in each face of the cell. For the example in Figures 5 and 6, points 1, 2, 4A, and 1A will be added to the building point's list and the building point's list for the lowest cell in Figure 6 includes points 1, 2, 1A, 4A, 1B, 2B, 5B, and 6B.

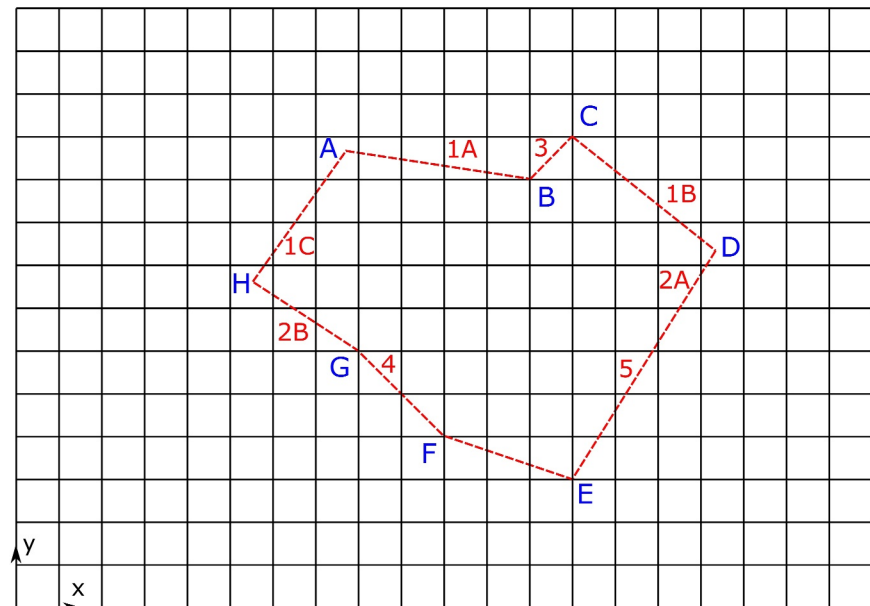
### 2.1.2. Processing Terrain Geometry

Terrain information is a continuous three-dimensional surface, while building data consists of a two-dimensional footprint and a height field. As a result, a different procedure is required to process terrain information and create a list of solid points for the cut-cells. QES-Winds utilizes the GDAL libraries (<https://www.gdal.org>), enabling it to load geo-referenced data sets

of terrain so that the simulations can include the effects of hills, valleys, and mountains. QES-Winds can load Digital Elevation Model (DEM) files, process the height field, and create the terrain geometry based on the domain resolution.

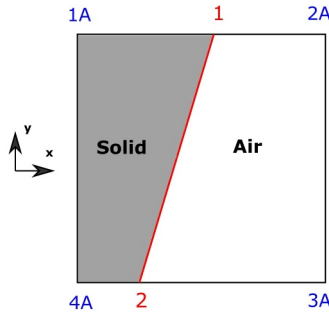
To process terrain cut-cells, the first step is to query the terrain height at each corner of each cell in the horizontal directions (each  $i$  and  $j$  index). Then, moving through cells in the  $z$ -direction, if at least one corner height (there are 4 of them) is greater than the cell's bottom-face height and less than the cell's top-face height, the cell is marked as a cut-cell. Figure 7 shows a schematic of a vertical array of cells cut through by terrain geometry. The red numbers (1–4) indicate the terrain point at the horizontal corners location of the cell, and the red dashed lines represent straight lines connecting the terrain points.

The next step is to find the intersection points between the line connecting every two corners (red dashed lines in Figure 7) and the cell's faces. The intersection points (3B, 4B, 2C, and 3C) are marked by the blue combination of

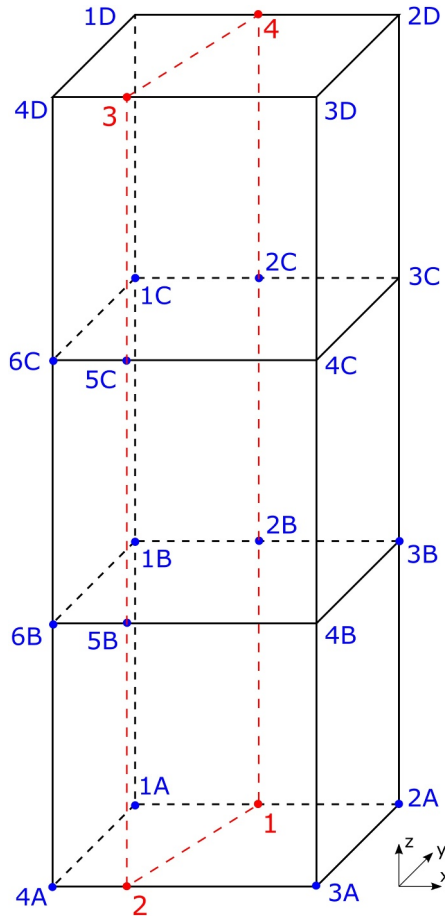


**Figure 4.** 2D schematic of a random building footprint showing how it cuts through underlying Cartesian grid. The dashed red lines are the building polygon lines. The red combination of numbers and letters marks different situations in which a building polygon line cuts through cells. The blue letters indicate the intersection of two building polygon lines.





**Figure 5.** 2D schematic of a random building cutting through the bottom face of the domain. The red numbers (1 and 2) indicate the building intersections with the bottom face. The red line is the building polygon line through the face. Points 1, 2, 4A, and 1A construct the solid area polygon inside the face.



**Figure 6.** Schematic of a vertical array of cells cut through by building geometry. The red numbers (1–4) indicate the building points at the top and bottom faces of the array. The red dashed lines represent straight lines connecting the building points (1–4).

letters and numbers in Figure 7. The intersection points are included in the list of solid points for the two-cell overlap at the position. Also, if the terrain points (the red points) are inside the bounds of the cell in the vertical direction, they are added to the list for the cell. Finally, if corners of the cell are located lower than the terrain points (points with the red number), they will be added to the list. These points are marked by the blue combination of numbers and letters in Figure 7. After this process, the list for each cell contains all the terrain-related solid vertices in that cell. For example, points 1A, 2A, 3A, 4A, 3, 1B, 2B, 3B, 4B, and 5B are included in the terrain point list for the lowest cell in Figure 7. In the last step, the terrain points that are located on each cell's face are added to the list of solid points for that face.

### 2.1.3. Reordering Vertices

The vertices created by the list of solid points for each cut-cell face determined for buildings (Section 2.1.1) and terrain (Section 2.1.2) must be reordered in a counter-clockwise manner because the formulation to calculate the area fraction coefficients only works with this condition. The reordering scheme starts with calculating the centroid (Figure 3) of the solid polygon of each face using the average of all solid vertices. Next, the angle of the line connecting each point to the face centroid, with respect to the horizontal line through the centroid, is determined. The angle marked by  $\theta$  in Figure 3 indicates the calculated angle for vertex 0. Sorting points by their angles increasing from  $-180^\circ$  to  $180^\circ$  results in ordering vertices in a counter-clockwise manner. Points 0 to 4 in Figure 3 are reordered to be counter-clockwise using this algorithm. This reordering scheme only works if vertices feed into it to form a convex polygon (all internal angles are less than  $180^\circ$ ). In the case of concave polygons (at least one internal angle is greater than  $180^\circ$ ), there is no well-known algorithm to reorder vertices. However, by processing the solid geometries in the building and terrain cases here, the chance of creating concave polygons is extraordinarily rare. Therefore, the current reordering scheme works for the terrain and building cases in this study.

### 2.1.4. Calculating Area Fraction Coefficient

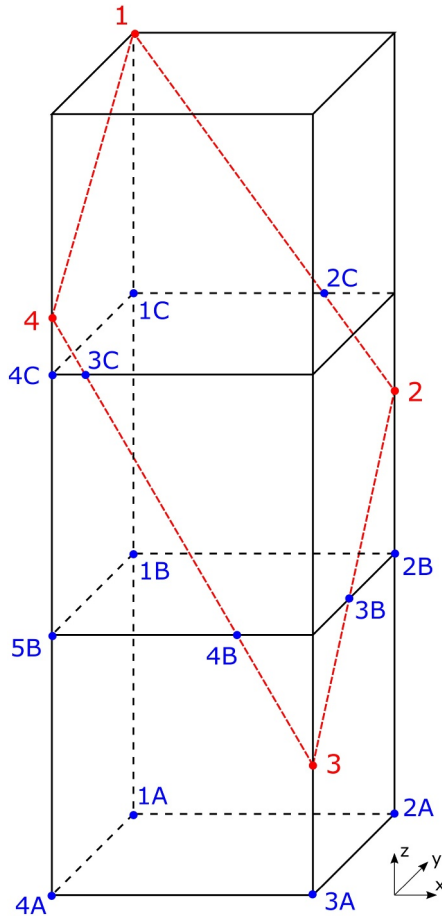
The final step utilizes Equation 1 to calculate the area fraction of the face located at the bottom of the cell, “below face” (Figure 3) ( $A_{i,j,k-\frac{1}{2}}^z$  in this example). The area fraction coefficient is defined as the area of the solid part ( $S_{solid}$ ) divided by the area of the cell face (in this case  $\Delta x \Delta y$ ):

$$A_{i,j,k-\frac{1}{2}}^z = \frac{S_{solid}}{S_{solid} + S_{air}} = \frac{\sum_{l=0}^{n-1} \frac{1}{2} (x_{((l+1) \bmod n)} - x_l) (y_{((l+1) \bmod n)} - y_l)}{\Delta x \Delta y}. \quad (1)$$

where  $i$ ,  $j$ , and  $k$  are the indices of the cell in  $x$ ,  $y$ , and  $z$  directions, respectively.  $l$  is the summation index and,  $n$  is the total number of points on the face, in the case of Figure 3,  $n$  is 5.

$$A_{i,j,k+\frac{1}{2}}^z = \frac{\sum_{l=0}^{n-1} \frac{1}{2} (x_{((l+1) \bmod n)} - x_l) (y_{((l+1) \bmod n)} - y_l)}{\Delta x \Delta y}, \quad (2)$$

$$A_{i-\frac{1}{2},j,k}^x = \frac{\sum_{l=0}^{n-1} \frac{1}{2} (y_{((l+1) \bmod n)} - y_l) (z_{((l+1) \bmod n)} - z_l)}{\Delta y \Delta z}, \quad (3)$$



**Figure 7.** Schematic of a vertical array of cells cut through by terrain geometry. The red numbers (1–4) indicate the terrain point at the horizontal corners of the cell. The red dashed lines represent straight lines connecting the terrain points (1–4) at the horizontal corners of the cell.

scope of this research because it reflects a neutral stability condition which is the case in these simulations.  $u^0$ ,  $v^0$ , and  $w^0$  are the initial wind field components. After solving the Poisson equation (Equation 8) for the Lagrange multipliers, the final velocity field is calculated using the Euler-Lagrange equations:

$$u = u^0 + \frac{1}{(2\alpha_1^2)} \frac{\partial \lambda}{\partial x}, \quad (10)$$

$$v = v^0 + \frac{1}{(2\alpha_1^2)} \frac{\partial \lambda}{\partial y}, \quad \text{and} \quad (11)$$

$$w = w^0 + \frac{1}{(2\alpha_2^2)} \frac{\partial \lambda}{\partial z}. \quad (12)$$

QES-Winds (Bozorgmehr, Willemsen, Gibbs, et al., 2021) utilizes the SOR method to solve the Poisson equation (Equation 8) over a staggered grid. The discretized version of the Poisson equation with the SOR method applied to it, is as follows:

$$\lambda_{i,j,k}^t = \frac{\omega}{e + f + g + h + m + n} [-(\Delta x)^2 R_{i,j,k} + e\lambda_{i+1,j,k}^{t-1} + f\lambda_{i-1,j,k}^{t-1} + A(g\lambda_{i,j+1,k}^{t-1} + h\lambda_{i,j-1,k}^{t-1}) + B(m\lambda_{i,j,k+1}^{t-1} + n\lambda_{i,j,k-1}^{t-1})] + (1 - \omega)\lambda_{i,j,k}^{t-1}. \quad (13)$$

$$A_{i+\frac{1}{2},j,k}^x = \frac{\sum_{l=0}^{n-1} \frac{1}{2} (y_{((l+1) \bmod n)} - y_l) (z_{((l+1) \bmod n)} - z_l)}{\Delta y \Delta z}, \quad (4)$$

$$A_{i,j-\frac{1}{2},k}^y = \frac{\sum_{l=0}^{n-1} \frac{1}{2} (x_{((l+1) \bmod n)} - x_l) (z_{((l+1) \bmod n)} - z_l)}{\Delta x \Delta z}, \quad \text{and} \quad (5)$$

$$A_{i,j+\frac{1}{2},k}^x = \frac{\sum_{l=0}^{n-1} \frac{1}{2} (x_{((l+1) \bmod n)} - x_l) (z_{((l+1) \bmod n)} - z_l)}{\Delta x \Delta z}. \quad (6)$$

Since QES-Winds ensures the conservation of mass, the area fraction coefficients for the fluid side are required. Every face consists of a solid and an air part and thus, the sum of the area fraction coefficients for the air and the solid part is one ( $A_{air} + A_{solid} = 1$ ). Once the area fraction coefficient for the solid part is calculated, the area fraction coefficient for the air side becomes:

$$A_{air} = 1 - A_{solid}. \quad (7)$$

### 2.1.5. Wind Solver for the Stair-Step Method

To obtain a quasi-time-averaged velocity field (mean wind field for every time step), QES-Winds (Bozorgmehr, Willemsen, Gibbs, et al., 2021) solves the Poisson equation for Lagrange multipliers,  $\lambda$ , as follows:

$$\frac{\partial^2 \lambda}{\partial x^2} + \frac{\partial^2 \lambda}{\partial y^2} + \left( \frac{\alpha_1}{\alpha_2} \right)^2 \frac{\partial^2 \lambda}{\partial z^2} = R \quad (8)$$

where  $R$  is the divergence of the initial wind field and is defined as:

$$R = -2\alpha_1^2 \left[ \frac{\partial u^0}{\partial x} + \frac{\partial v^0}{\partial y} + \frac{\partial w^0}{\partial z} \right]. \quad (9)$$

$\alpha_1$  and  $\alpha_2$  are Gaussian precision moduli that define the relative importance of the horizontal and vertical wind flow and are set to one (Bagal, 2005) for the

Here,  $e, f, g, h, m, n$  are defined as cell-centered boundary condition coefficients and are equivalent to  $e_{i,j,k}, f_{i,j,k}, g_{i,j,k}, h_{i,j,k}, m_{i,j,k}, n_{i,j,k}$ . Superscripts  $t$  and  $t - 1$  are representing current and previous iterations, respectively. Equation 13 displays the serial and original SOR equation. In the parallel version (GPU solver), using a red-black coloring scheme, the superscript  $t$  turns into  $t - 1$  for the red sub-iteration, while superscripts  $t - 1$  turns into  $t$  for the black sub-iteration, in the numerator of the right-hand side of Equation 13.  $A = \frac{(\Delta x)^2}{(\Delta y)^2}$ ,  $B = \eta \frac{(\Delta x)^2}{(\Delta y)^2}$ , and  $\eta = \left(\frac{\alpha_1}{\alpha_2}\right)^2$  are domain constants.  $\omega$  is the SOR over-relaxation factor and is set to 1.78 based on the recommendation by Röckle (1990). The boundary condition for solid surfaces is  $\left(\frac{\partial \lambda}{\partial n} = 0\right)$ , and for inlet/outlet surfaces, it is  $\lambda = 0$ .

### 2.1.6. Updated Solver for the Cut-Cell Method

To implement the cut-cell method, the area fraction coefficients need to be incorporated into the solver. First, gradient terms using finite difference discretization become:

$$\frac{\partial u^0}{\partial x} = \frac{1}{\Delta x} (A_{i+\frac{1}{2},j,k}^x u_{i+\frac{1}{2},j,k}^0 - A_{i-\frac{1}{2},j,k}^x u_{i-\frac{1}{2},j,k}^0), \quad (14)$$

$$\frac{\partial v^0}{\partial y} = \frac{1}{\Delta y} (A_{i,j+\frac{1}{2},k}^y v_{i,j+\frac{1}{2},k}^0 - A_{i,j-\frac{1}{2},k}^y v_{i,j-\frac{1}{2},k}^0), \quad (15)$$

$$\frac{\partial w^0}{\partial z} = \frac{1}{\Delta z} (A_{i,j,k+\frac{1}{2}}^z w_{i,j,k+\frac{1}{2}}^0 - A_{i,j,k-\frac{1}{2}}^z w_{i,j,k-\frac{1}{2}}^0), \quad (16)$$

$$\frac{\partial \lambda}{\partial x} = \frac{A_{i-\frac{1}{2},j,k}^x}{\Delta x} (\lambda_{i,j,k} - \lambda_{i-1,j,k}), \quad (17)$$

$$\frac{\partial \lambda}{\partial y} = \frac{A_{i,j-\frac{1}{2},k}^y}{\Delta y} (\lambda_{i,j,k} - \lambda_{i,j-1,k}), \quad (18)$$

$$\frac{\partial \lambda}{\partial z} = \frac{A_{i,j,k-\frac{1}{2}}^z}{\Delta z} (\lambda_{i,j,k} - \lambda_{i,j,k-1}), \quad (19)$$

$$\frac{\partial^2 \lambda}{\partial x^2} = \frac{1}{(\Delta x)^2} [A_{i+\frac{1}{2},j,k}^x \lambda_{i+1,j,k} - (A_{i+\frac{1}{2},j,k}^x + A_{i-\frac{1}{2},j,k}^x) \lambda_{i,j,k} + A_{i-\frac{1}{2},j,k}^x \lambda_{i-1,j,k}], \quad (20)$$

$$\frac{\partial^2 \lambda}{\partial y^2} = \frac{1}{(\Delta y)^2} [A_{i,j+\frac{1}{2},k}^y \lambda_{i,j+1,k} - (A_{i,j+\frac{1}{2},k}^y + A_{i,j-\frac{1}{2},k}^y) \lambda_{i,j,k} + A_{i,j-\frac{1}{2},k}^y \lambda_{i,j-1,k}], \quad \text{and} \quad (21)$$

$$\frac{\partial^2 \lambda}{\partial z^2} = \frac{1}{(\Delta z)^2} [A_{i,j,k+\frac{1}{2}}^z \lambda_{i,j,k+1} - (A_{i,j,k+\frac{1}{2}}^z + A_{i,j,k-\frac{1}{2}}^z) \lambda_{i,j,k} + A_{i,j,k-\frac{1}{2}}^z \lambda_{i,j,k-1}]. \quad (22)$$

In Equation 20,  $A_{i+\frac{1}{2},j,k}^x$  is the area fraction coefficient of face  $i + \frac{1}{2}$ . To follow the same methodology as the stair-step solver, the solver coefficients are defined as:

$$\begin{aligned} e &= A_{i+\frac{1}{2},j,k}^x, & f &= A_{i-\frac{1}{2},j,k}^x, & g &= A_{i,j+\frac{1}{2},k}^y, \\ h &= A_{i,j-\frac{1}{2},k}^y, & m &= A_{i,j,k+\frac{1}{2}}^z, & n &= A_{i,j,k-\frac{1}{2}}^z. \end{aligned} \quad (23)$$

Discretization with cut-cell method of Equation 8 results in:

$$R_{i,j,k} = -2\alpha_1^2 \left[ \frac{(eu_{i+\frac{1}{2},j,k}^0 - fu_{i-\frac{1}{2},j,k}^0)}{\Delta x} + \frac{(gv_{i,j+\frac{1}{2},k}^0 - hv_{i,j-\frac{1}{2},k}^0)}{\Delta y} + \frac{(mw_{i,j,k+\frac{1}{2}}^0 - nw_{i,j,k-\frac{1}{2}}^0)}{\Delta z} \right] \quad (24)$$



where the left hand side of Equation 8 is:

$$\frac{[e\lambda_{i+1,j,k} - (e+f)\lambda_{i,j,k} + f\lambda_{i-1,j,k}]}{(\Delta x)^2} + \frac{[g\lambda_{i,j+1,k} - (g+h)\lambda_{i,j,k} + h\lambda_{i,j-1,k}]}{(\Delta y)^2} + \frac{\left(\frac{\alpha_1}{\alpha_2}\right)^2 [m\lambda_{i,j,k+1} - (m+n)\lambda_{i,j,k} + n\lambda_{i,j,k-1}]}{(\Delta z)^2}. \quad (25)$$

By applying the SOR method and after some manipulation:

$$\lambda_{i,j,k}^t = \frac{\omega}{e+f+g+h+m+n} [-(\Delta x)^2 R_{i,j,k} + e\lambda_{i+1,j,k}^{t-1} + f\lambda_{i-1,j,k}^t + A(g\lambda_{i,j+1,k}^{t-1} + h\lambda_{i,j-1,k}^t) + B(m\lambda_{i,j,k+1}^{t-1} + n\lambda_{i,j,k-1}^t)] + (1-\omega)\lambda_{i,j,k}^{t-1}. \quad (26)$$

The modified Euler-Lagrange equations become:

$$u_{i,j,k} = f_{i,j,k} u_{i,j,k}^0 + \frac{f_{i,j,k}}{(2\alpha_1^2)\Delta x} [\lambda_{i,j,k} - \lambda_{i-1,j,k}], \quad (27)$$

$$v_{i,j,k} = h_{i,j,k} v_{i,j,k}^0 + \frac{h_{i,j,k}}{(2\alpha_1^2)\Delta y} [\lambda_{i,j,k} - \lambda_{i,j-1,k}], \quad \text{and} \quad (28)$$

$$w_{i,j,k} = n_{i,j,k} w_{i,j,k}^0 + \frac{n_{i,j,k}}{(2\alpha_2^2)\Delta z} [\lambda_{i,j,k} - \lambda_{i,j,k-1}]. \quad (29)$$

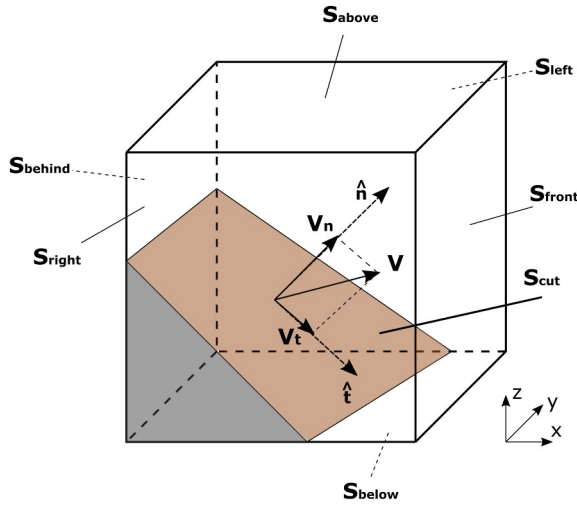
This new version of the solver is more general than the stair-step version. It can handle the stair-step mesh easily as well as the cut-cell mesh. In the stair-step method, the boundary condition coefficients are either zero or one while, in the cut-cell method, the coefficients can have any value between zero and one. This means that Equation 26 can cover both geometry representation methods by changing only the coefficient values. Also, setting the boundary condition coefficients to zero or one in Equation 24 and Equations 27–29 yield the same ones (Equation 9 and Equations 10–12) for the stair-step method.

## 2.2. Near-Wall Parameterization

QES-Winds ensures the conservation of mass for the final velocity field but conservation of momentum is not enforced (Bozorgmehr, Willemsen, Gibbs, et al., 2021; Bozorgmehr, Willemsen, Margairaz, et al., 2021). This results in a mass-consistent velocity field that can create large and unrealistic surface stress estimations in the absence of any parameterizations. Briggs (2015) created a wall-stress model to enforce the logarithmic wind profile at the nearest cell to the surface using an iterative approach. The model was implemented in the stair-step version of QUIC-URB (Pardyjak & Brown, 2003; Singh et al., 2008) and reported improvement in the scalar transport and the wall-stress estimates near the surface (Briggs, 2015). The same approach is adopted here for the cut-cell method applied to terrain geometry. The near-wall parameterization improves momentum (shear stress) transport from the solid surface and adds some momentum effects to the model. This parameterization is the very first empirical parameterization available in QES-Winds for flow over the complex terrain. In this method, the tangential velocity component (Figure 8) of the cell closest to the wall is parameterized to satisfy the logarithmic velocity profile (Monin & Obukhov, 1954; Stull, 1988). This is done using the tangential velocity component from the second cell in the wall-normal direction and the aerodynamic surface roughness.

Enforcing the log-law on the velocity tangential to cut faces requires knowledge of the surface's normal and tangential directions. Figure 8 shows a schematic of the normal and tangential directions for an arbitrary cut face. Following Salesky et al. (2019), the vector normal to the cut face can be calculated using:

$$\hat{n} = \frac{1}{S_{cut}} (\hat{i} \quad \hat{j} \quad \hat{k}) \begin{pmatrix} S_{front} - S_{behind} \\ S_{left} - S_{right} \\ S_{above} - S_{below} \end{pmatrix} \quad (30)$$



**Figure 8.** Schematic of the cut face with normal and tangential directions to the face. Also, the velocity vector in the cell and normal and tangential components of the velocity vector are depicted.

where  $S_{behind}$ ,  $S_{front}$ ,  $S_{right}$ ,  $S_{left}$ ,  $S_{below}$ , and  $S_{above}$  are the surface area of the fluid part of their associated cell faces (Figure 8) and  $S_{cut}$  is the surface of the cut face (Figure 8):

$$S_{behind} = A_{i-\frac{1}{2},j,k}^x \Delta y \Delta z, \quad (31)$$

$$S_{front} = A_{i+\frac{1}{2},j,k}^x \Delta y \Delta z, \quad (32)$$

$$S_{right} = A_{i,j-\frac{1}{2},k}^y \Delta x \Delta z, \quad (33)$$

$$S_{left} = A_{i,j+\frac{1}{2},k}^y \Delta x \Delta z, \quad (34)$$

$$S_{below} = A_{i,j,k-\frac{1}{2}}^z \Delta x \Delta y, \quad \text{and} \quad (35)$$

$$S_{top} = A_{i,j,k+\frac{1}{2}}^z \Delta x \Delta y, \quad (36)$$

$$S_{cut} = \left[ (S_{front} - S_{behind})^2 + (S_{left} - S_{right})^2 + (S_{above} - S_{below})^2 \right]^{1/2}. \quad (37)$$

To define the tangential vector to the cut face, a local coordinate system consisting of normal and tangential components at the face and the velocity vector in the cell is utilized. The velocity vector in the cut cell can be defined as the sum of the normal and tangential vector components:

$$\vec{V} = V_n \hat{n} + V_t \hat{t} \quad (38)$$

where  $V_n$  and  $V_t$  are the normal and tangential components of the velocity vector. The normal component of the velocity vector is defined through the dot product of the velocity vector and the normal to the cut face vector:

$$V_n = \vec{V} \cdot \hat{n} = u_n \hat{n}_i + v_n \hat{n}_j + w_n \hat{n}_k. \quad (39)$$

The tangential velocity components in the Cartesian grid ( $u_t$ ,  $v_t$ , and  $w_t$ ) are calculated using the fact that the sum of tangential and normal components in the Cartesian grid is equal to the velocity component in the cell:

$$u_t = u - u_n, \quad (40)$$

$$v_t = v - v_n, \quad \text{and} \quad (41)$$

$$w_t = w - w_n. \quad (42)$$

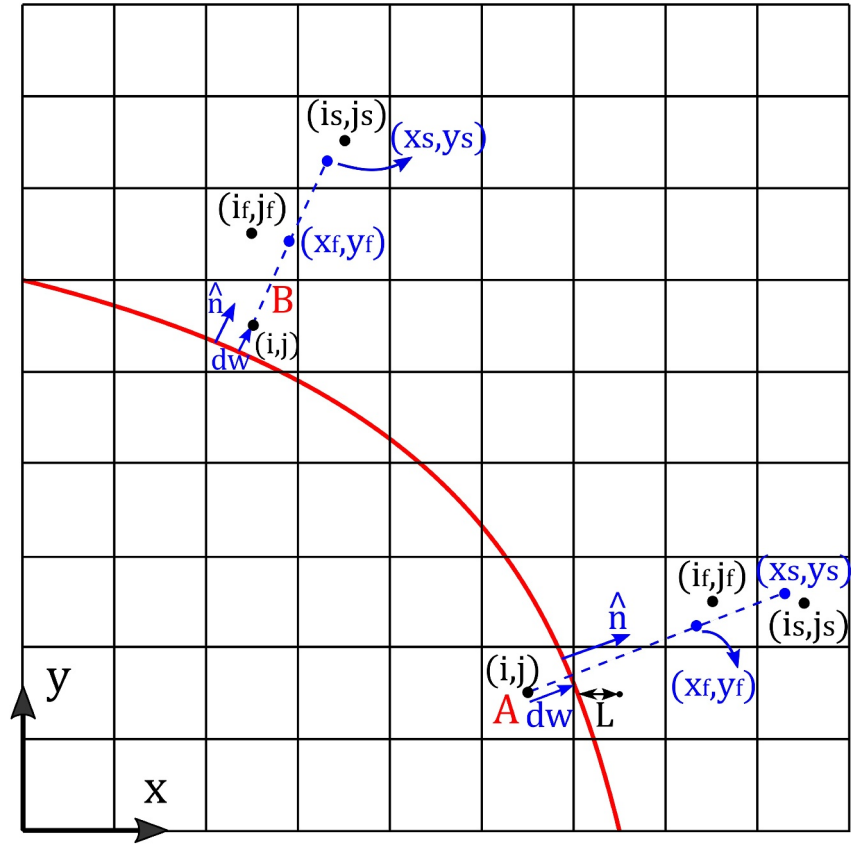
Finally, the tangential unit vector of the cut face is calculated using the tangential velocity components in the Cartesian grid:

$$\hat{t}_i = \frac{u_t}{(u_t^2 + v_t^2 + w_t^2)^{1/2}}, \quad (43)$$

$$\hat{t}_j = \frac{v_t}{(u_t^2 + v_t^2 + w_t^2)^{1/2}}, \quad \text{and} \quad (44)$$

$$\hat{t}_k = \frac{w_t}{(u_t^2 + v_t^2 + w_t^2)^{1/2}}. \quad (45)$$

Salesky et al. (2019) incorporated trilinear interpolation to calculate the velocity vector at a constant distance from the wall. This interpolation is computationally expensive and requires knowledge of the exact location of the wall. Here, only an estimation of the cut face is computed. As a result, the nearest cell center is assumed to be the



**Figure 9.** A two-dimensional representation of how the first and second cells in the wall-normal direction are defined for the terrain. Two examples are included in the plot: the cell-center is inside the terrain (cell marked as A) and the cell-center is outside of the terrain (cell marked as B).

reference location, and the velocity vector co-located at the cell center is utilized. Figure 9 shows a two-dimensional schematic of the first and second cells in the wall-normal direction defined.

To find the first and second cells in the normal direction to the cut face, the following formulation is used:

$$x_f = x(i) + C\hat{n}_i\Delta x, \quad (46)$$

$$y_f = y(j) + C\hat{n}_j\Delta y, \quad (47)$$

$$z_f = z(k) + C\hat{n}_k\Delta z, \quad (48)$$

$$x_s = x(i) + (C+1)\hat{n}_i\Delta x, \quad (49)$$

$$y_s = y(j) + (C+1)\hat{n}_j\Delta y, \quad \text{and} \quad (50)$$

$$z_s = z(k) + (C+1)\hat{n}_k\Delta z, \quad (51)$$

where  $x(i)$ ,  $y(j)$ , and  $z(k)$  are the cut-cell's center location in the  $x$ ,  $y$ , and  $z$  directions, respectively.  $\hat{n}_i$ ,  $\hat{n}_j$ , and  $\hat{n}_k$  are the unit normal vector components in the  $x$ ,  $y$ , and  $z$  directions, respectively, and  $C$  is a constant.  $x_f$  and  $x_s$ ,  $y_f$  and  $y_s$ , and  $z_f$  and  $z_s$  are the  $x$ ,  $y$ , and  $z$  components of the location of the first and second points in the wall-normal direction, respectively. In both examples depicted in Figure 9, the cell marked as A, where the cell-center is inside the terrain, and the cell marked as B, in which the cell-center is not inside the solid, the constant  $C$  is initially set to 1. The first and second cells in the direction normal to the surface are the cells that encompass these locations, which are indicated by the indices  $(i_f, j_f)$  and  $(i_s, j_s)$  in Figure 9, respectively.

To implement the method described in Briggs (2015), the distance of the first and second cell-centers from the wall is required:

$$d_{first} = \left[ (x(i_f) - x(i))^2 + (y(j_f) - y(j))^2 + (z(k_f) - z(k))^2 \right]^{1/2} + d_w, \quad (52)$$

$$d_{second} = \left[ (x(i_s) - x(i))^2 + (y(j_s) - y(j))^2 + (z(k_s) - z(k))^2 \right]^{1/2} + d_w, \quad (53)$$

where  $d_w$  is the distance of the cut-cell's center in the normal direction from the cut face. This value is negative if the cell's center is located inside the solid region (cell A in Figure 9) and positive in case the cell's center is outside of the solid region (cell B in Figure 9). If the value of  $d_{first}$  is greater than the surface roughness ( $z_0$ ), the process continues with the current first and second cells. Otherwise, the constant value ( $C$ ) is increased by one, and the process of defining the first and second cells and calculating the distances from the wall is repeated until the above-mentioned condition is satisfied. Example B in Figure 9 represents the situation that the first guess of the first and second cells satisfies the condition ( $C = 1$ ), while for example, A, the first round of identifying the first and second cells leads to value  $d_{first}$  (L in Figure 9) that is smaller than  $z_0$ . In this situation, the constant is increased by one, and the first and second cells are redefined. The second round of determination satisfies the condition and guarantees non-negative velocity values.

The velocity magnitude in the tangential direction for the second cell is estimated as the absolute value of the dot product between the velocity vector at the second cell and the tangential unit vector. Finally, the velocity magnitude in the tangential direction in the first cell is calculated using a form of the log-law (Briggs, 2015):

$$|V_t^1| = |V_t^2| - \frac{u_*}{\kappa} \ln \left( \frac{d_{second}}{d_{first}} \right) \quad (54)$$

where  $|V_t^1|$  and  $|V_t^2|$  are the velocity magnitude in the tangential direction for the first and second cells, respectively, with the velocity magnitude in the tangential direction for the second cell,  $|V_t^2|$ , is estimated as the absolute value of the dot product between the velocity vector at the second cell and the tangential unit vector,  $\kappa$  is the von Karman constant (0.4), and  $u_*$  is the friction velocity value. The initial value of  $u_*$  is set to 0.1 m/s. After calculating the value of  $|V_t^1|$  using Equation 54 and the initial value of  $u_*$ , the value of  $u_*$  is updated using:

$$u_* = \kappa \frac{|V_t^1|}{\ln \left( \frac{d_{first}}{z_0} \right)} \quad (55)$$

where  $z_0$  is the surface roughness for the cut-cell. Equations 54 and 55 are solved using an iterative method. After 20 iterations, the tangential component of the velocity magnitude for the first cell becomes the parameterized velocity vector in the cell. Since the normal component of the velocity vector is zero for the first cell, to convert the velocity vector to the Cartesian grid, QES-Winds uses the following:

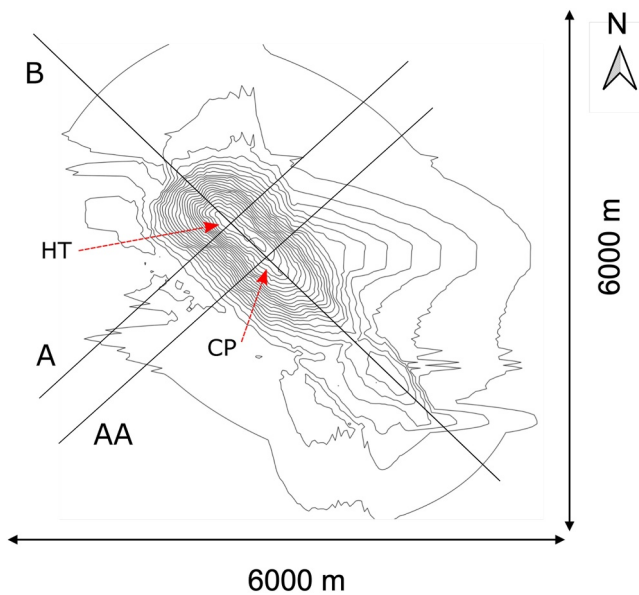
$$u_0 = |V_t^1| \hat{t}_i, \quad (56)$$

$$v_0 = |V_t^1| \hat{t}_j, \quad \text{and} \quad (57)$$

$$w_0 = |V_t^1| \hat{t}_k. \quad (58)$$

### 3. Results and Discussion

In this section, the influence of the new geometry representation scheme (the cut-cell method) and the near-wall parameterization of the wind field are discussed for different complex terrain and urban cases. None of QES's standard wake or street canyon parameterizations are used because the complex fields created by the parameterizations can cover up the effects of geometry representation methods. QES-Winds' SOR solver was run for 500



**Figure 10.** Contour map of Askervein Hill indicating the location of data transects A, AA, and B. The elevation interval is 5 m. The location of the highest point and central point of the Hill are marked as intersections of lines A and AA with line B.

iterations for all the simulations because the velocity field reached the 0.01 m/s threshold (did not change more than that value between iterations), which was established as the convergence criteria in Bozorgmehr, Willemsen, Gibbs, et al., 2021. In these test cases, QES-Winds is applied to real and complex urban and topography geometries. To verify QES-Winds solver with the cut-cell and stair-step methods, the model is applied to a hemisphere as a potential solver and the simulation results are compared to the analytical wind field in Appendix B.

### 3.1. Askervein Hill Test Case

Askervein Hill was selected as the first test case. This is an isolated Hill in Scotland where field experiments have been conducted and data for testing and evaluation exist (Mickle et al., 1988; Taylor & Teunissen, 1987). The Askervein Hill data set is 6 km by 6 km, and the hill is approximately 116 m tall. The data set has been used in the literature to validate fine-scale wind-field simulations over complex terrain (Forthofer et al., 2014). Askervein Hill has a smooth transition from the surrounding flat terrain that averages 8 m above sea level to the highest point of the hill. This feature makes Askervein Hill an appropriate choice for verifying wind fields over complex terrain in QES-Winds, which does not have any terrain-specific parameterizations. Wind measurements used in this study were collected at 10 m above ground level from 40 towers located 100 m apart along three transects (lines A, AA, and B). In Figure 10, lines A and AA are oriented in the southwest-northeast direction, almost perpendicular to the hill orientation. While line B

is oriented in the northwest-southeast direction, on the ridge of the hill. Figure 10 shows a contour map of Askervein Hill with 5-m contour intervals. The location of lines A, AA, and B as well as the location of the highest point (HT) and central point of the Hill are marked in Figure 10.

Forthofer et al. (2014) averaged the original 10-min data over 3-hr period to match their mass-consistent wind solver situation. The original data sets were identified as the MF03-D and TU03B (Taylor & Teunissen, 1987). The chosen test case corresponds to an average velocity of  $8.9 \text{ ms}^{-1}$  from  $210^\circ$  at a reference site about 3 km southwest of the hill. Following Forthofer et al. (2014), an exponential velocity profile that matches the measurements from the upstream tower was created with parameters specific to QES-Winds:

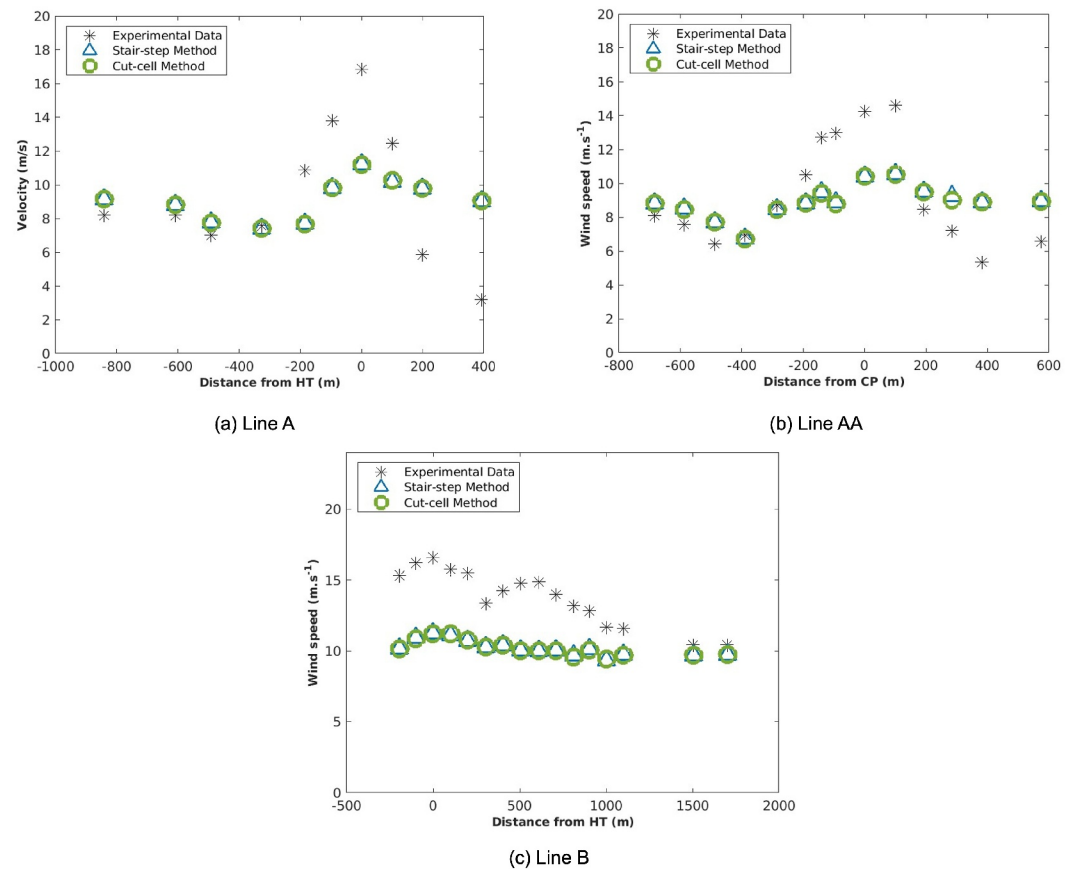
$$u^0(z) = u_{ref} \cdot (z/z_{ref})^{z_0}, \quad (59)$$

where QES-Winds' exponential profile (Equation 59) is matched to the one used by Forthofer et al. (2014) by specifying the surface roughness  $z_0 = 0.21 \text{ m}$ , reference height  $z_{ref} = 10.24 \text{ m}$  and velocity at the reference height  $u_{ref} = 8.97 \text{ ms}^{-1}$ .

The Askervein Hill DEM used here has a 23.35 m by 23.35 m horizontal resolution. Therefore, the first set of simulations was run using 23.35 m as the horizontal cell size ( $x$  and  $y$  directions) and 1 m as the vertical cell size. The domain was discretized using  $257 \times 257 \times 200$  cells in the  $x$ ,  $y$ , and  $z$  directions and the stair-step and cut-cell methods were utilized to process the terrain geometry. Figure 11 displays a comparison between the velocity magnitude for QES simulations with the stair-step and cut-cell methods and the measurement data along transects A, AA, and B.

Simulated wind speeds are within 35% of the measured magnitude everywhere except on the lee side of the hill. QES-Winds only accounts for the conservation of mass and does not solve a momentum equation. As a result, the forces that create fluid motion, such as adverse pressure gradients that lead to flow separation and slowing down of flow on the lee side of the hill are not accounted for. Adding new parametrizations specific to the terrain that mimic the effect of momentum transport is required to improve the accuracy of the velocity field on the lee side of the hill.



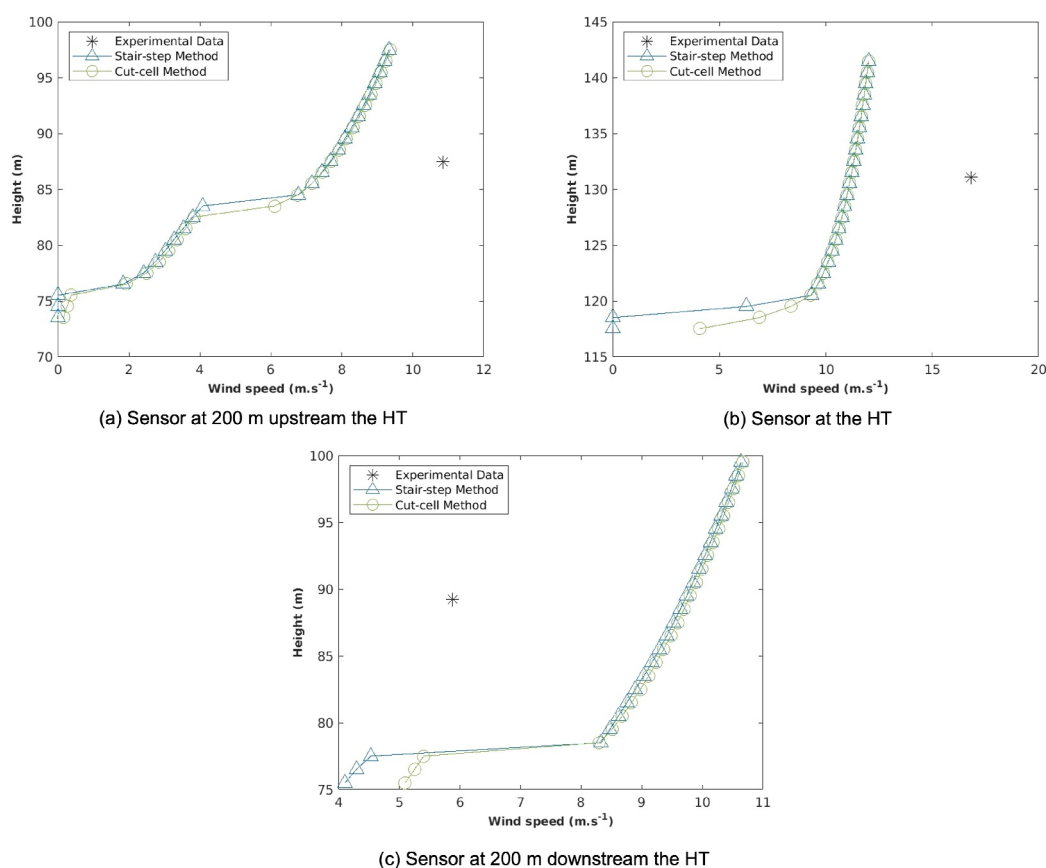


**Figure 11.** Wind speeds for simulations with 23.35 m as the horizontal cell size ( $x$  and  $y$  directions) and 1 m as the vertical cell size for the stair-step and cut-cell methods comparing to measurement data by Taylor and Teunissen (1987). The comparison is for: (a) Line A, (b) Line AA, and (c) Line B. Location of points highest point, and central point, and transects A, AA, and B shown in Figure 10.

Figure 11 indicates that the simulated values for the stair-step and cut-cell methods are nearly identical. In QES-Winds, the divergence of the initial wind field is the driving force behind changes in Lagrange multipliers and, in turn, the final velocity field. In the absence of empirical parameterizations, the effect of geometry representation (i.e., the stair-step or cut-cell method) modifies the divergence value for at most the third to fourth row of air cells above the terrain. Figure 12 shows the vertical velocity magnitude profiles for the cut-cell and stair-step methods versus the measured data at the sensor located at 200 m upstream of the HT, at the HT, and at 200 m downstream of the HT on line A. The plots indicate the differences between the cut-cell and stair-step methods closer to the surface. However, because all sensors are located 10 m above the ground, there is no difference between the two methods, and the comparison does not show improvement in the wind field for the cut-cell method. This test case was included not to indicate the cut-cell method's benefits over the stair-step representation but to show how well the QES-Winds results match the measured data for flow over the hill.

### 3.2. Granite Mountain Test Case

Analysis of the Askervein Hill results suggest that differences between the cut-cell method and the stair-step method when applied to terrain will only be apparent close to geometry objects. This means that comparisons should focus on the near-ground region. Measurements taken at Granite Mountain as part of the Mountain Terrain Atmospheric Modeling and Observations (MATERHORN) (Fernando et al., 2015; Jensen et al., 2017) field campaign meet this requirement and were therefore selected. The MATERHORN field experiments were conducted in two parts, one in the autumn from 26 September to 7 November 2012 and



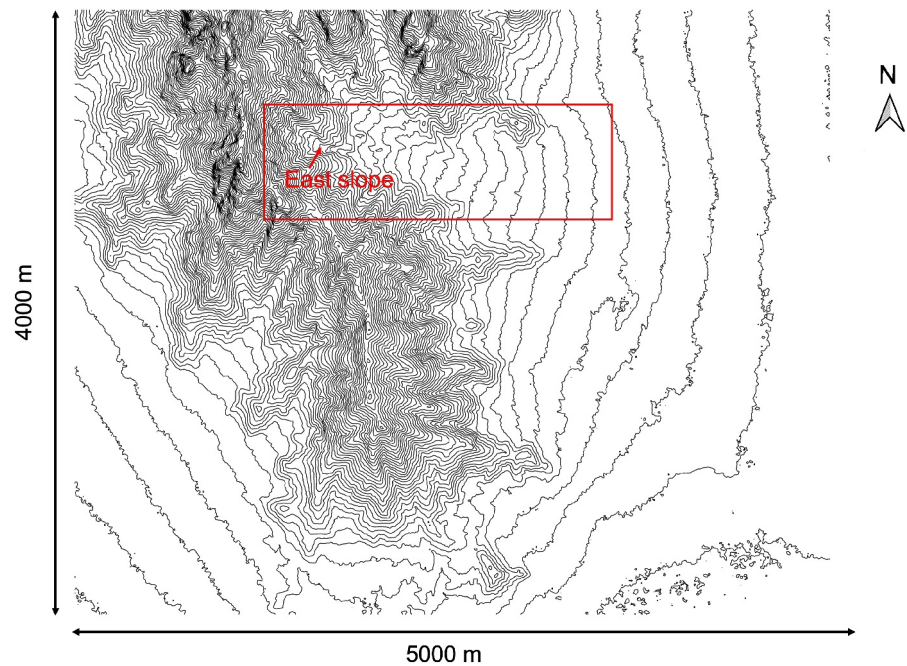
**Figure 12.** Vertical velocity magnitude profiles for the cut-cell and stair-step methods versus the measured data. The comparison is at: (a) Sensor located at 200 m upstream the HT, (b) Sensor at the HT, and (c) Sensor located at 200 m downstream the HT.

one in the spring from 1 May to 6 June 2013. Because of insufficient measurements during the autumn campaign and more uniform wind-forcing direction during the spring campaign, the spring data were chosen as the reference measurement period.

The experimental site was located at the US Army Dugway Proving Ground, approximately 100 km southwest of Salt Lake City, UT, with a focus on Granite Mountain (Fernando et al., 2015). Granite Mountain is a relatively isolated peak with 850 m of elevation prominence with respect to the surrounding valley. Data collected by sensors deployed over the East Slope of Granite Mountain were used. The East Slope is an east-west aligned slope about 4 km long that opens to an alluvial fan about 2 km down-slope from the ridge-line (Figure 13). The Granite Mountain DEM (Figure 13) has a 5 m horizontal resolution.

The list of instruments deployed on the East Slope includes five turbulence towers (ES1-ES5), 12 PWIDS, and 13 Local Energy-budget Measurement Stations (LEMS) (Gunawardena et al., 2022). The location of all of the above-mentioned instruments, except the ES1 tower, is shown in Figure 14. At the time of the experiment, LEMS did not measure wind components, so they were not used in this research. ES1 and PWIDS 37 and 108 are not used here because of equipment failures during the desired time periods. Table 1 summarizes the information about the ES towers and PWIDS based on Lehner et al. (2015).

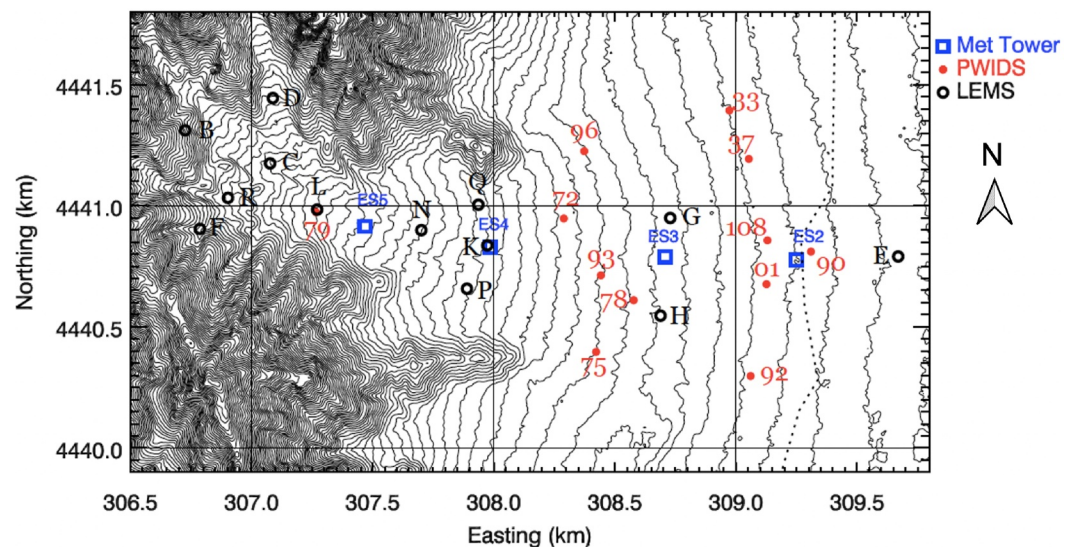
Analysis of the Askervein Hill test case (Section 3.1) demonstrated that because of the lack of terrain flow parameterizations, QES-Winds' ability to reproduce field data on the lee side of a hill is limited. As a result, only time periods when the dominant wind direction is from east or southeast (upslope winds) corresponding to the 13 – 16 MDT time period are considered here. Data were broken into 30-min time periods and averaged for comparison with QES-Winds' mass-consistent wind solver. There were 31 days in the spring campaign creating a



**Figure 13.** Contour map of part of the Granite peak indicating the approximate location of the east slope with the red box. The elevation interval is 10 m for this case.

total of 186 possible 30-min periods between 13 and 16 MDT. However, sufficient data for the ES towers and PWIDS was available only 67 of these time periods starting from 15:30 MDT on 5 May and ending at 16:00 MDT on 17 May 2013.

The computational domain is 5 km by 4 km in  $x$  and  $y$  directions. The highest point of terrain in the domain is about 600 m above the lowest point, therefore, a 750 m vertical domain extent was chosen to allow flow to go over the mountain unimpeded. To investigate the influence of grid size on the results, a range of resolutions has been used to discretize the domain. Table 2 lists the various horizontal and vertical grid sizes used in simulations and the case chosen to indicate each combination.



**Figure 14.** Location of ES2-ES5 turbulence towers, 12 PWIDS and 13 Local Energy-budget Measurement Stations over the contour map of the east slope (Figure courtesy of Huch and Gunawardena).

**Table 1**  
*Specifications of the ES Towers and PWIDS Deployed on the East Slope*

Tower name	Easting (m)	Northing (m)	Slope	z (m) (AGL)
PWIDS 01	309,213.3	4,440,608.7	1.5	2.0
PWIDS 33	308,975.5	4,441,393.3	1.4	
PWIDS 72	308,289.5	4,440,947.4	2.5	
PWIDS 75	308,428.9	4,440,391.9	3.2	
PWIDS 78	308,577.8	4,440,610.3	2.4	
PWIDS 79	307,266.5	4,440,973.4	4.0	
PWIDS 90	309,309.7	4,440,810.6	2.1	
PWIDS 92	309,060.6	4,440,302.6	2.1	
PWIDS 93	308,438.7	4,440,708.2	2.2	
PWIDS 96	308,372.6	4,441,234.1	3.0	
ES2	309,249.9	4,440,776.6	1.6	0.5, 2, 5, 10, 16, 20, 25
ES3	308,707.7	4,440,789.1	1.7	0.5, 2, 5, 10, 20
ES4	307,986.1	4,440,828.5	5.8	0.47, 2.05, 5.12, 10, 20, 26.5
ES5	307,468.8	4,440,914.9	3.6	0.55, 2.14, 5.13, 10.13, 20.08

*Note.* Information in this table was obtained from Lehner et al. (2015). AGL stands for above ground level. UTM zone is 12 for all instruments in this Table.

The 30-min averaged wind speed and direction of the 10 PWIDS (Table 1), is used to initialize the velocity field in QES-Winds. First, QES-Winds created a logarithmic velocity profile (described in Bozorgmehr, Willemsen, Margairaz, et al. (2021)) in the vertical direction for each PWIDS using their wind speed, wind direction, height above the ground (2 m), and surface roughness ( $z_0$ ). The surface roughness value is calculated by using measured  $u_*$  data for the first sensor of the ES2 and ES3 towers, and is set to 0.03 m. Then, the vertical profiles are used to create the initial guess wind field utilizing QES-Winds' two-dimensional Barnes interpolation scheme (Barnes, 1973). Barnes interpolation scheme (Barnes, 1973; Koch et al., 1983) produces a Gaussian weighted average that decreases by increasing distance from the sensor location. The two-dimensional Barnes scheme uses these weighted averages to interpolate the velocity field in horizontal surfaces at each cell height utilizing the vertical profile value at that surface. The two-dimensional Barnes scheme is appropriate for interpolating the initial guess-field in urban areas with negligible topography changes since the reference height of QES-Winds' domain (bottom of the domain) is equivalent to the ground surface everywhere. However, for the cases like Granite Mountain where the terrain changes are significant, the two-dimensional Barnes scheme can create an initial guess of wind speeds of the order of 30 m/s at 2 m above the ground on top of the hill while the measured velocity at the foot of the hill (PWIDS location) is  $\sim 3$ –5 m/s. To prevent generating nonphysical high values of wind speed, the two-dimensional Barnes scheme was modified to accommodate for flow over complex terrain. The new Barnes scheme is called the *terrain-following Barnes scheme* from here on. It generates the Gaussian weighted averages using the horizontal distance from the sensor location (same as before). For the modified method, inside the area affected by the terrain, it interpolates the velocity field in a terrain-following surface with

**Table 2**  
*Different Horizontal and Vertical Grid Resolution and the Case Number Created From the Combination of Them*

Vertical grid size (m)	Horizontal grid size (m)						
	5	10	15	20	25	30	40
1	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7
3	Case 8	Case 9	Case 10	Case 11	Case 12	Case 13	Case 14

*Note.* UTM zone is 12 for all instruments in this Table.

the constant height from the ground level instead of the constant height from the QES-Winds' domain reference height (two-dimensional Barnes scheme). While, outside of the area, the constant height from the QES-Winds' domain reference height (two-dimensional Barnes scheme) is used. The area affected by the terrain is approximated by the atmospheric surface layer (ASL) height since in the ASL, the surface shear stress dominates and the velocity profile exhibits a nearly logarithmic profile (Stull, 1988). The ASL is the bottom layer of the Atmospheric Boundary Layer (ABL) and the ASL height is approximately 5% of the ABL height.

To calculate the ASL height, the ABL height is required to be estimated. The ABL height is dependent on atmospheric stability, synoptic wind speed, and vertical and horizontal scales of the topography (De Wekker & Kossmann, 2015). Stull (1992) identified four different ways the ABL depth changes with respect to the terrain geometry. Deng and Stull (2005) recognized two of the most common types of the ABL behavior over the complex terrain as: first, the terrain-following type in which the ABL top changes with the terrain geometry and the ABL height from the ground is constant. Second way is the constant level type in which the ABL top is constant over the complex terrain and the ABL height from the ground changes with the terrain geometry. They introduced an anisotropic correlation between the two horizontal (two-dimensional Barnes scheme) and the vertical (new modification) factors to interpolate the measured data to the grid cells. Their method is more accurate but it is computationally expensive since the vertical factor can only be partially transferred to the neighboring cells and the transfer is non-linear and iterative. As the number of cells in the horizontal directions and the number of measurement towers increases, calculating the vertical factor becomes more expensive. As a result, this method is not suitable for QES-Winds at this stage of development. However, a similar strategy as described in Deng and Stull (2005) can be used to estimate the ABL height over complex terrain by summing the contribution of the terrain-following and the level type in a linear manner and independent of other domain cells in one iteration. This way, there is no need for more iterations (simplified method) and the computational cost is decreased considerably. The ABL height is defined as:

$$H_{ABL} = 2 * Z_{ref} \{ |Z_g - Z_m| \} \quad \text{if } |Z_g - Z_m| < Z_{ref} \quad (60)$$

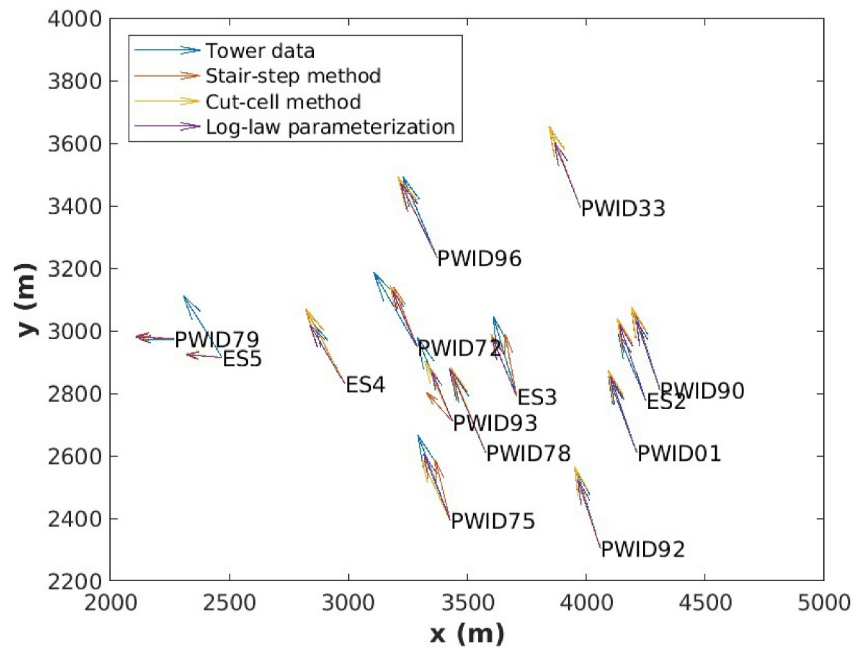
$$H_{ABL} = Z_{ref} \quad \text{if } |Z_g - Z_m| > Z_{ref} \quad (61)$$

where  $H_{ABL}$ ,  $Z_{ref}$ ,  $Z_g$ , and  $Z_m$  are the ABL height estimation at the grid cell, the ABL height approximation depending on the atmospheric stability condition, terrain height at the grid cell, and terrain height at the measurement location, respectively. After approximating the ABL height, the ASL height is estimated as 5% of the ABL height. Finally, the ASL height is used to define the area above the terrain that is affected by the terrain geometry.

After running simulations with the stair-step and cut-cell methods and the near-wall parameterization (log-law parameterization), the velocity vectors for the PWIDS and 2 m above ground level sensors for ES towers were plotted. The velocity vectors at 2 m above ground were plotted to make sure that the simulated wind fields have similar wind directions (especially for the PWIDS) to the measured data. Also, if the majority of PWIDS (especially those close to each other, and to ES2 and ES3 towers) have similar wind directions, there is less turbulence and complex flow structures that QES-Winds is unable to model. QES-Winds uses Gaussian weighted-average values to interpolate the initial field (the Barnes scheme). The Gaussian weighted average is equal to one at the sensor location and decreases with increasing distance from the location. This means that if the simulated wind direction is significantly different from the measured value at the PWIDS location, there is a complex flow structure that QES-Winds is unable to model. The goal of this study is to showcase the differences between the geometry representation methods, and near-wall parameterization, these complex flow structures are undesirable since they can mask the differences between the methods.

Only 28 of the 30-min periods passed the direction test and were analyzed in this section after plotting the velocity vectors for all 67 cases. Figure 15 compares measured velocity vectors at the PWIDS and ES towers with simulated data at the Case 10 resolution for the stair-step and cut-cell methods and with the log-law parameterization for the time period of 14:30–15:00 MDT 14 May 2013. It can be concluded from the velocity-vector plot that the ES4 and ES5 towers have a significant difference between the measured data and the simulations'





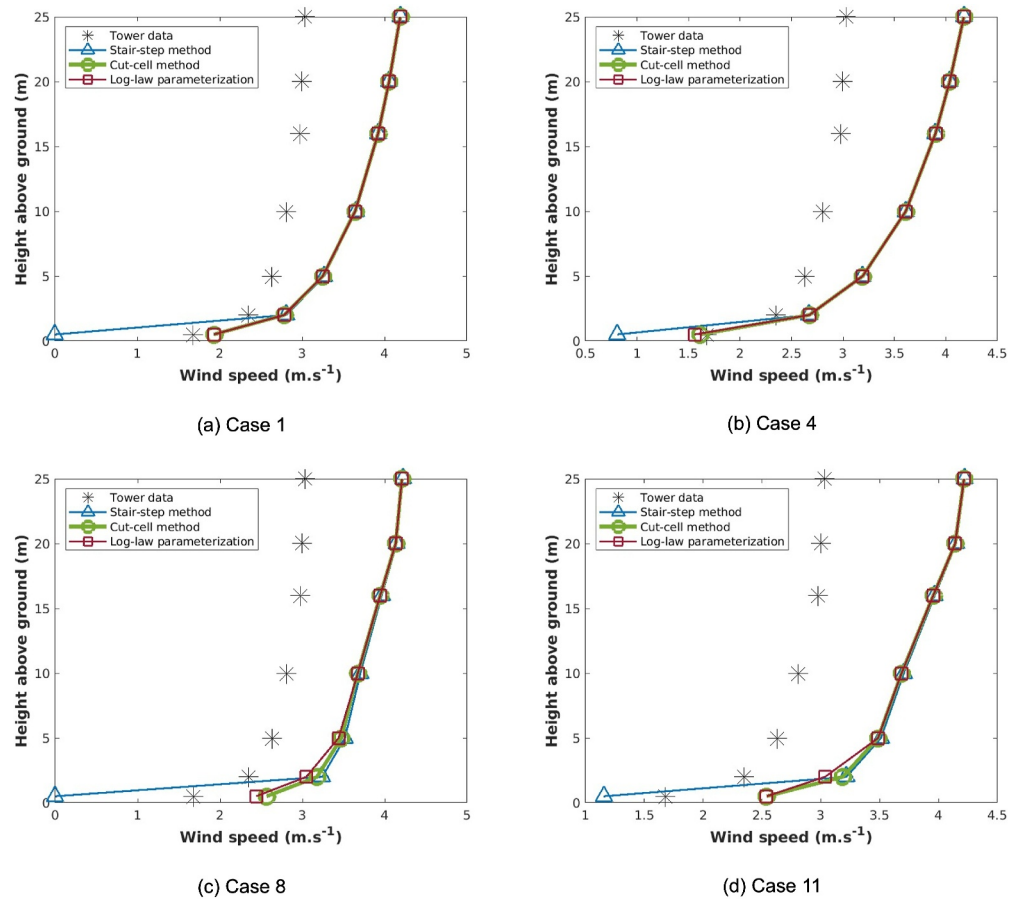
**Figure 15.** Velocity vectors at 2 m above ground level at the location of PWIDS and ES towers for measured data and simulations over the Case 10 resolution. The simulations were run with the stair-step and cut-cell methods and with the log-law parameterization for the time period of 14:30–15:00 14 May 2013.

wind directions. ES4 and ES5 are located on steeper terrain with less open space compared to ES2 and ES3 which results in more complex flow structures near the ES4 and ES5 towers. Therefore, ES4 and ES5 results are excluded from further analysis to prevent errors caused by QES-Winds's simple physics from masking any improvements that result from the use of the cut-cell method and log-law parameterization in the simulations.

We begin our analysis of the Granite Mountain case by comparing the measured vertical profile of the velocity magnitude from the ES2 tower to the simulated velocity magnitudes with grid-size Cases 1, 4, 8, and 11 for the time period of 14:30–15:00 MDT 14 May 2013 (Figure 16). For the finest grid (Case 1), the simulation results for the cut-cell and log-law methods are almost identical. However, the stair-step method creates a big error by calculating zero velocity magnitude. For Case 4 (Figure 16b), the log-law parameterization decreases the velocity magnitude slightly compared to the cut-cell method at the first sensor above the ground (0.5 m). Both the cut-cell method and near-wall parameterization results are in good agreement with the measured data, while the stair-step creates about 50% less wind speed.

As the vertical grid size increases, the log-law results are slightly better than the cut-cell method and significantly superior to the stair-step method. For the Case 8 grid (Figure 16c), the stair-step method places the first sensor inside solid terrain resulting in zero velocity and hence creates a significant error. The stair-step method also generates larger wind speeds for the sensors above to make up for the velocity deficit created near the surface. As the horizontal grid size increases (Case 11, Figure 16d), the stair-step method calculates a non-zero wind speed for the first sensor which is closer to the measured value. The log-law parameterization slows the velocity at the second sensor above the ground compared to the cut-cell method.

Figure 16 demonstrates that the impact of the different geometry representations and near-surface parameterizations is primarily confined to the lowest levels of the simulations. To explore this impact across the 28 selected 30-min periods, scatter plots of the velocity magnitude from simulations using the log-law parameterization and the cut-cell and stair-step methods against the measured data (solid black line) are presented in Figures 17 and 18 for the ES2 tower sensors 1 and 2 (0.5 and 2 m), respectively. The calculations were run over the domain with grid-size Cases 1, 4, 8, and 11 and performance is summarized for each configuration using the root mean square error (RMSE). Table 3 shows the calculated RMSE values between the measured wind speed and simulated



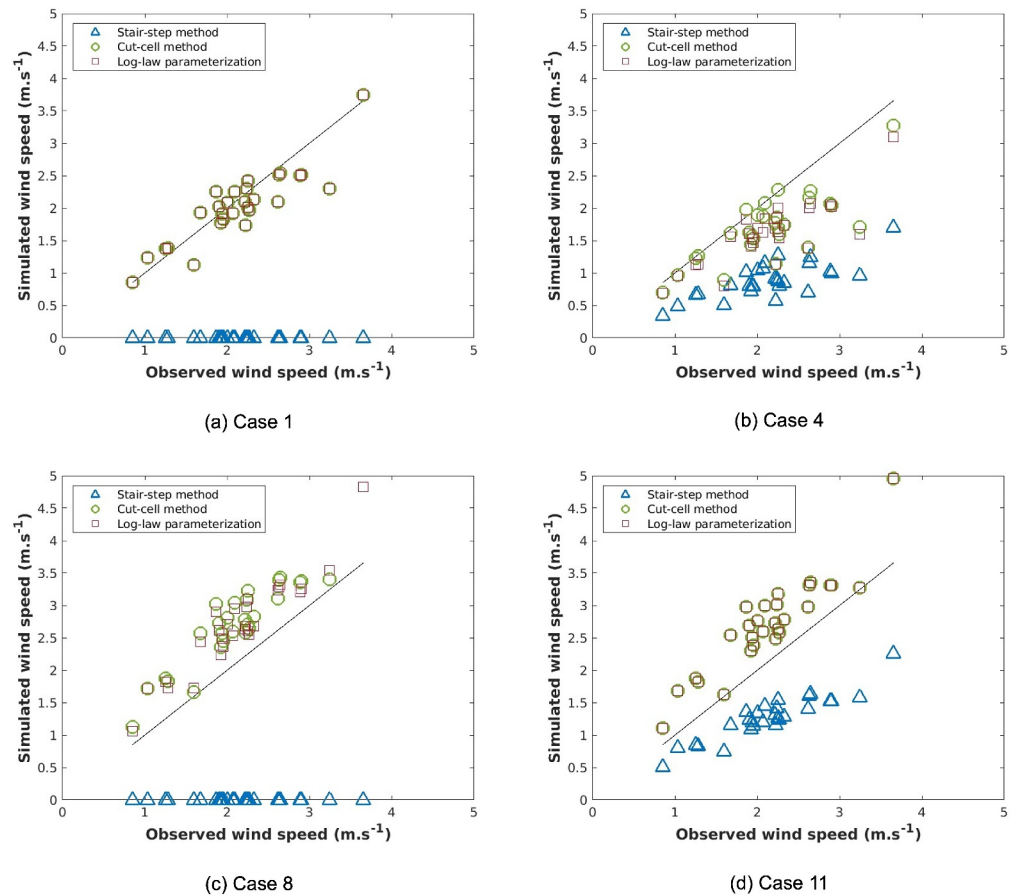
**Figure 16.** Vertical profiles of velocity magnitude for the time period of 14:30–15:00 MDT 14 May 2013 for the ES2 tower. The measured data and results from simulations with the stair-step and cut-cell methods as well as the log-law parameterization over the domain with: (a) Case 1, (b) Case 4, (c) Case 8, and (d) Case 11 as grid sizes.

velocity magnitude for simulations with the stair-step and cut-cell methods and the log-law parameterization for the first and second sensors of ES2 and ES3 towers with Cases 1, 4, 8, and 11.

Both scatter plots support the primary findings from Figure 16. At the finest grid resolution (Case 1), the stair-step performs much worse than the log-law parameterization and the cut-cell method for the lowest sensor. For the second sensor (2 m), all three models have similar values, while the stair-step is slightly further from the measured values. As horizontal grid size increases, while the vertical grid resolution remains constant (Case 4), the stair-step method performs better compared to Case 1 but still performs much worse than the cut-cell method and near-wall parameterization for the 0.5 m sensor. For the second sensor (2 m), all simulation models generate similar values.

Increasing the vertical grid size decreases the cut-cell method and log-law parameterization accuracy significantly while the stair-step performance stays the same for the first sensor. For the Case 8 resolution, this is mainly due to the stair-step method treating the cell as fully solid (the same as Case 1). Although the stair-step does not consider the first sensor to be inside the terrain for Case 11, it still underestimates the velocity magnitude, while the log-law parameterization and the cut-cell method overestimate the value compared to the measured data. For the second sensor (2 m), stair-step simulations using the Cases 8 and 11 grid sizes, overestimate the velocity magnitude to make up for the underestimated flow speed in the cell below. In Cases 8 and 11, the log-law parameterization performs slightly better than the cut-cell and the stair-step methods.

Vertical profiles of velocity magnitude at the ES3 location for simulations run with grid sizes corresponding to Cases 1, 4, 8, and 11 for the 14:30–15:00 MDT 14 May 2013 time period are shown in Figure 19 along with

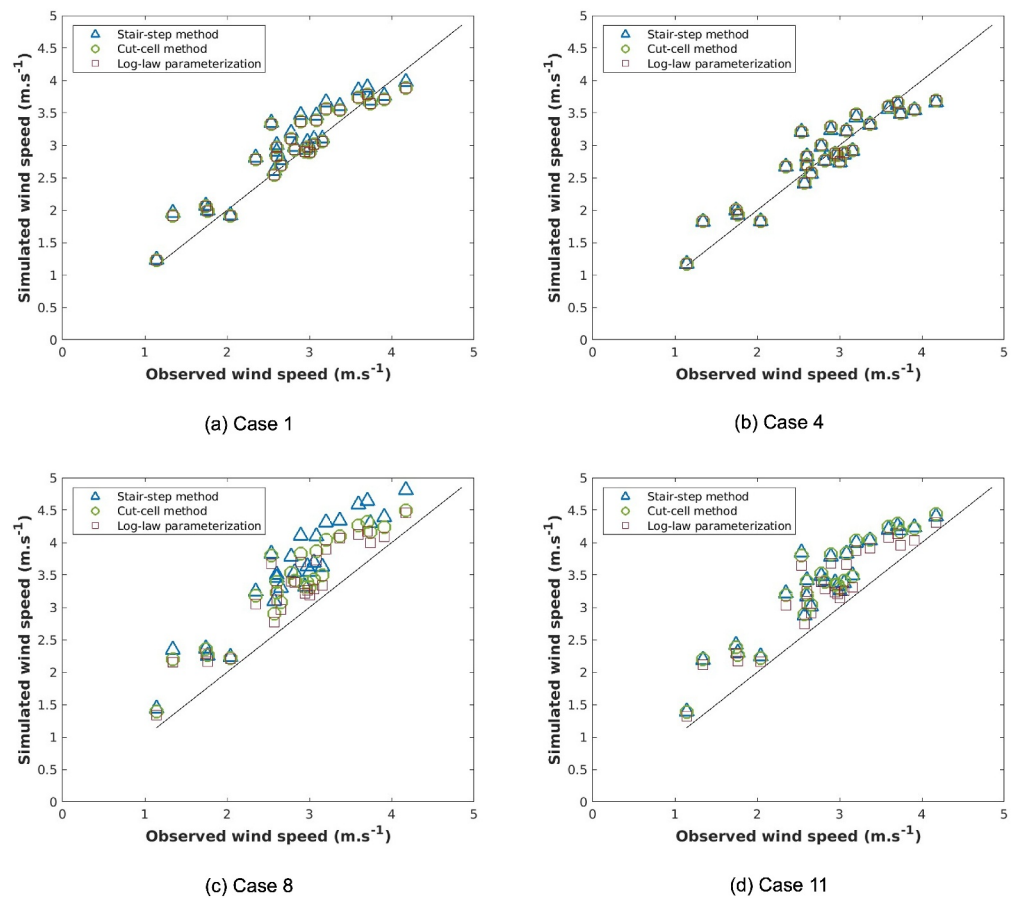


**Figure 17.** Scatter plots of velocity magnitude simulated using the stair-step and cut-cell methods, and the log-law parameterization for the anemometer 0.5 m above the ground level at ES2 (sensor number 1). The black line indicates a perfect match with the measured data. All 28 time periods are plotted and the computational domain was discretized using: (a) Case 1, (b) Case 4, (c) Case 8, and (d) Case 11 as grid sizes.

the tower data. The ES3 tower is located at the height of 39.6 m from the bottom of the QES-Winds domain and as a result, the first and second tower sensors' positions correspond to vertical heights of 40.1 and 41.6 m, respectively from the bottom of the domain. This puts both sensors in the same cell for a 3 m vertical cell size. QES-Winds calculates the velocity components at the cell faces (staggered grid) but outputs the velocity field as an averaged cell-centered value (average of the face values for each component). In this study, the wind speed is evaluated at the cell center that encompasses the sensor position which for ES3 tower, results in the first sensor not being resolved for the simulations with 3 m vertical grid size. This error is caused by a combination of the chosen discretization and the tower location, and results in omitting the first sensor data for 3-m vertical grid size.

It can be concluded from Figure 19 that the cut-cell method performs better than the stair-step method but worse than the log-law for the finest grid (Case 1) at the first sensor. However, at the second sensor, the log-law overestimates the wind speed to make up for the slow down at the first sensor which results in lower accuracy while the stair-step and cut-cell methods generate almost identical values. For Case 4, the cut-cell creates similar results compared to the log-law and better ones than stair-step method at the first sensor while all three models generate almost identical velocity magnitudes at the second sensor.

As the vertical grid size increases (Case 8), the stair-step performs better at the second sensor height. The log-law methods then accelerate the wind at the third sensor height to overcome the lower wind at the second sensor height. For the Case 11, lowest resolution, both the near surface parameterization and cut-cell methods perform better than the stair-step method. While the near-wall parameterization slows down the wind at the third sensor height.



**Figure 18.** Scatter plots of wind speed simulated using the stair-step and cut-cell methods, and the log-law parameterization for the anemometer 2.0 m above the ground level at ES2 (sensor number 2). The black line indicates a perfect match with the measured data. All 28 time periods are plotted and the computational domain was discretized using: (a) Case 1, (b) Case 4, (c) Case 8, and (d) Case 11 as grid sizes.

Scatter plots of the wind speed from simulations using the log-law parameterization and the cut-cell and stair-step methods against the measured data are presented in Figures 20 and 21 for the ES3 tower. The calculations were run over the domain with grid-size Cases 1, 4, 8, and 11 for the second sensor. Because of not resolving the first sensor in simulations with grid-size Case 8 and 11, the scatter plots for the first sensor are only shown for Case 1 and 4. The performance is summarized for each configuration using the RMSE values in Table 3.

The plots for the first sensor indicate that for Case 1, the log-law performs better than the stair-step and cut-cell methods while most of simulations overestimate the wind speed. On the other hand, most of simulations for Case 4 underestimate the wind speed while the cut-cell method and near-wall parameterization estimate the wind speed much better than the stair-step method.

It can be concluded from Figure 21 and Table 3 that for finer resolutions, Case 1 and 4, the differences amongst simulated values are small and calculated wind speeds are in good agreement with the measured velocity magnitudes. For Case 8, the cut-cell method performs slightly better while the log-law has worse performance than the stair-step. However, for Case 11, the stair-step performs worse than the cut-cell and near-wall parameterization.

### 3.3. Oklahoma City Test Case

To illustrate the difference in simulated velocities between the stair-step and cut-cell methods in urban areas, the central business district (CBD) in Oklahoma City during the JU2003 experiment campaign was used (Figure 22).

**Table 3**

*Calculated Root Mean Square Error Values Using the Measured and Simulated Wind Speeds for Cases With the Stair-Step and Cut-Cell Methods and the Log-Law Parameterization for the First and Second Sensors of ES2 and ES3 Towers With Cases 1, 4, 8, and 11*

Method	Case number	ES2		ES3	
		Sensor 1 (m/s)	Sensor 2 (m/s)	Sensor 1 (m/s)	Sensor 2 (m/s)
Stair-step	1	2.22	0.35	0.52	0.35
	4	1.33	0.28	0.70	0.42
	8	2.22	0.83	N/A	0.44
	11	0.93	0.67	N/A	0.67
Cut-cell	1	0.30	0.31	0.42	0.33
	4	0.59	0.28	0.53	0.42
	8	0.69	0.67	N/A	0.40
	11	0.64	0.67	N/A	0.39
Log-law	1	0.30	0.31	0.34	0.39
	4	0.64	0.28	0.56	0.40
	8	0.6	0.57	N/A	0.71
	11	0.64	0.54	N/A	0.39

*Note.* N/A means that RMSE is not applicable at the first sensor of ES3 for Cases 8 and 11 since QES-Winds locates it inside the terrain.

The JU2003 campaign took place in July 2003 and included 10 Intense Observation Periods (IOPs) (Allwine & Flaherty, 2006). IOP8 has been chosen for evaluation because the winds were uniform and the data required for analysis were readily available. IOP8 took place overnight (from 2300 to 0700 CDT) on Thursday, 24 July 2003. A complete list of the deployed instruments and details of the IOPs are included in the report by Allwine and Flaherty (2006). To make the IOP8 measurement suitable for comparison to QES-Winds, IOP8 data were averaged over eight, 1-hr periods.

To initialize the velocity field in the domain, the vertical-velocity information from four of the Indiana University (IU) sonic anemometers (Grimmond et al., 2004) located about 5 km south of the computational domain was used. The IU sensors were mounted on a tower at 19.35 m, 37.26 m, 54.56 m, and 79.63 m above ground level. The hourly averaged data from the IU sonics were utilized here to match a vertical velocity profile. The urban canopy input profile was chosen because it was used in a similar study (Hayati, 2018), and it matches the features of the computational domain (dense urban area) well. The initial urban canopy profile is defined as follows:

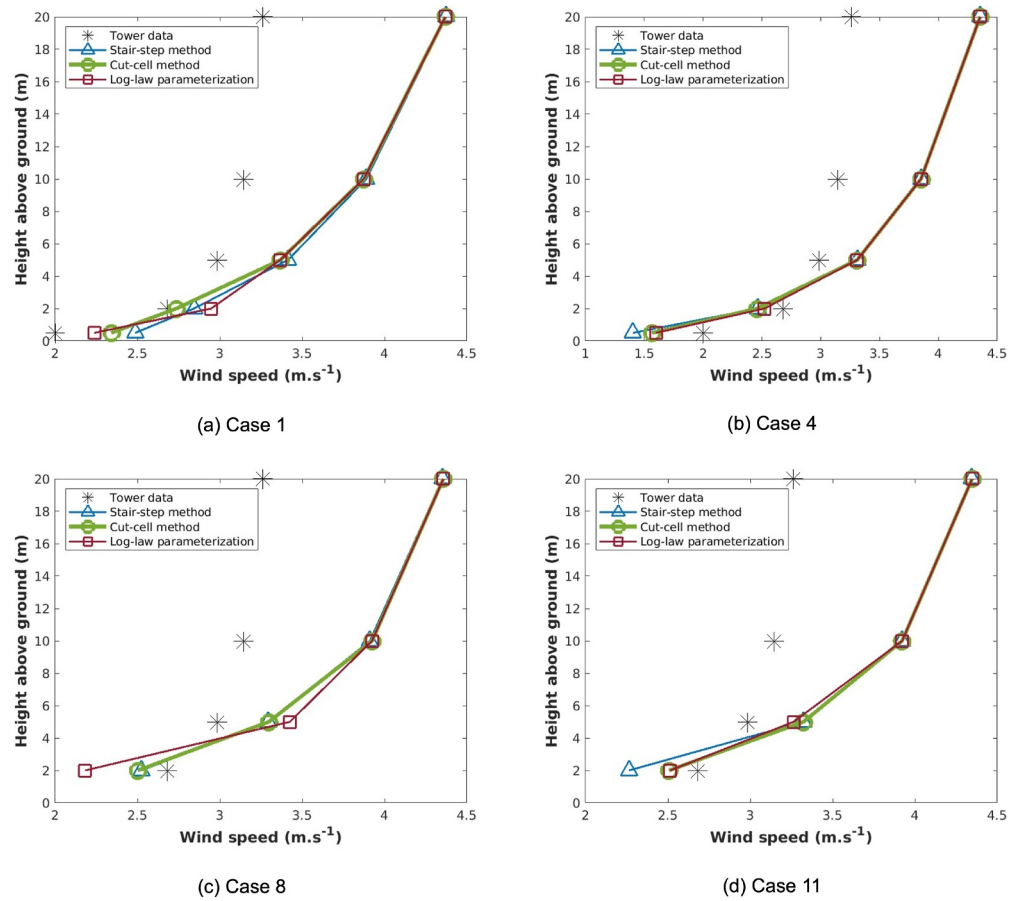
$$u^0(z \leq H) = u(H) \cdot \exp\left(\alpha\left(\frac{z}{H} - 1\right)\right), \quad (62)$$

$$u^0(z > H) = \frac{u_{ref} \left( \ln\left(\frac{z-d}{z_0}\right) \right)}{\ln\left(\frac{z_{ref}}{z_0}\right)}, \quad (63)$$

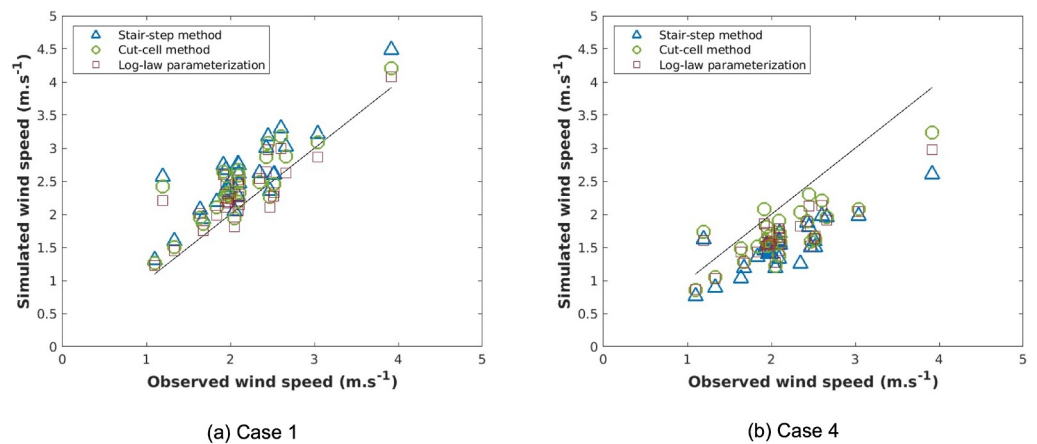
where  $z_0$ ,  $z_{ref}$ ,  $H$ ,  $u(H)$ ,  $d$ ,  $\alpha$ , and  $u_{ref}$  are aerodynamic surface roughness, reference height, canopy height, velocity at the canopy height, displacement height, the attenuation coefficient, and velocity measured at the reference height, respectively.

After fitting Equations 62 and 63 to the data, eight sets of vertical velocity profiles were determined for eight hourly averaged data sets. All the fitted curves are in good agreement with the IU sensor data. To define the wind direction of the input profile, data from the Dugway Proving Ground's PWIDS 15 (Portable Weather Information Display Systems) on the rooftop of the post office building south of the CBD (Allwine & Flaherty, 2006) at 55 m above ground level were averaged hourly over the time period of IOP8. Table 4 displays values required to create

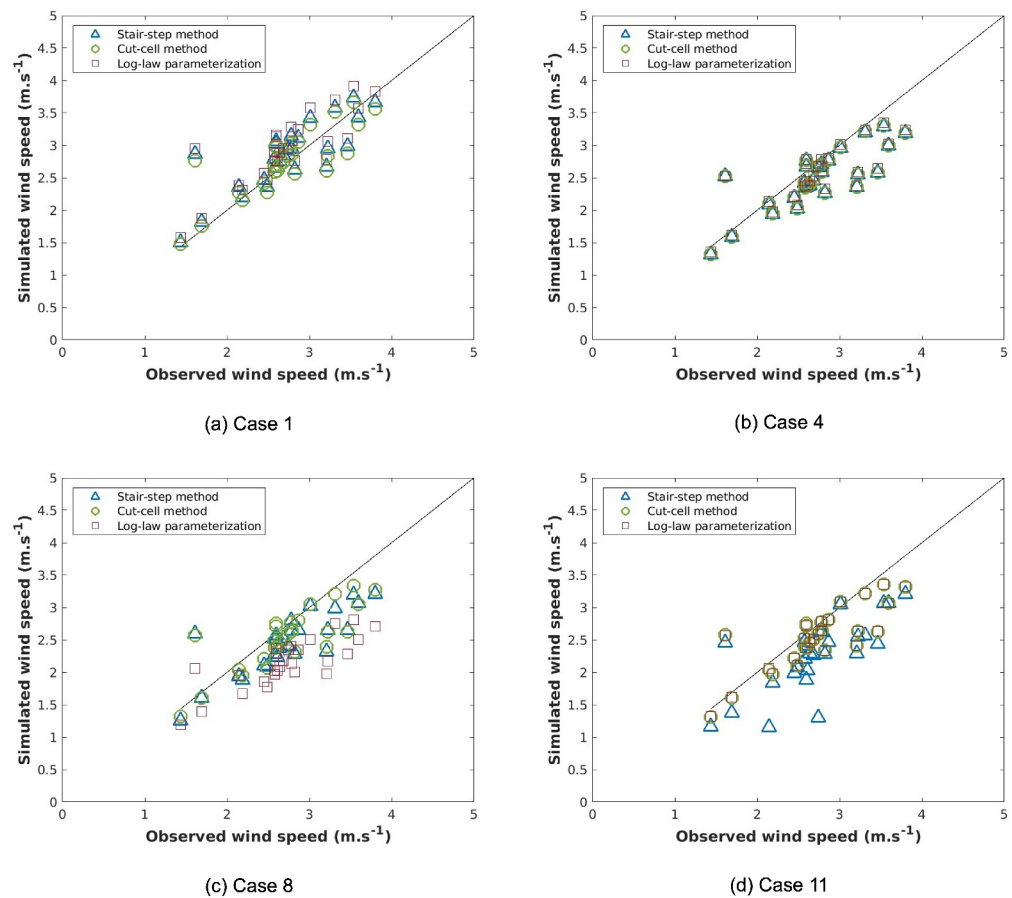




**Figure 19.** Vertical profiles of velocity magnitude for the time period of 14:30–15:00 MDT 14 May 2013 for the ES3 tower. The measured data and results from simulations with the stair-step and cut-cell methods as well as the log-law parameterization over the domain with: (a) Case 1, (b) Case 4, (c) Case 8, and (d) Case 11 as grid sizes.



**Figure 20.** Scatter plots of velocity magnitude simulated using the stair-step and cut-cell methods, and the log-law parameterization for the anemometer 0.5 m above the ground level at ES3 (sensor number 1). The black line indicates a perfect match with the measured data. All 28 time periods are plotted and the computational domain was discretized using: (a) Case 1, and (b) Case 4 as grid sizes.



**Figure 21.** Scatter plots of velocity magnitude simulated using the stair-step and cut-cell methods, and the log-law parameterization for the anemometer 2.0 m above the ground level at ES3 (sensor number 2). The black line indicates a perfect match with the measured data. All 28 time periods are plotted and the computational domain was discretized using: (a) Case 1, (b) Case 4, (c) Case 8, and (d) Case 11 as grid sizes.

the upstream velocity profile for eight hourly averaged periods. Figure 23 shows the fitted urban canopy profile against the IU sensor data for 3–4 CDT.

The computational domain was 1,180 m in length and 1,210 m in width. To investigate the influence of grid size on the results, a range of resolutions has been used to discretize the domain. Table 5 summarizes the various horizontal and vertical grid sizes used in simulations and the case number for their combinations. The computational domain includes several sophisticated structures that make the wind field complicated. Applying building parameterizations to these structures creates an even more complex wind field and can mask the effect of the geometry representation. Therefore, in the simulations related to this test case, all building parameterizations available in QES-Winds (Bagal, Pardyjak, & Brown, 2004; Bagal, Singh, et al., 2004; Gowardhan et al., 2010; Hayati et al., 2017; Kaplan & Dinar, 1996; Pardyjak & Brown, 2001; Pol et al., 2006; Singh et al., 2008) were turned off. It can be concluded from the results of the Askervein Hill test case that the geometry representation effects on the flow field are limited to the cells near the solid geometry. Finding sensors close enough to the buildings is a challenge since researchers usually put the instruments away from the solid elements to avoid interference and inside important flow structures, like wakes behind the building, street canyon, etc. However, the available data used in Nelson et al. (2007), provided enough flow information close to the buildings on the Park Avenue street canyon to represent differences between the two geometry representation methods. Tables 6 and 7 contain information about the towers and the instrument locations as well as where the data used in this study were obtained from Allwine and Flaherty (2006) and Nelson et al. (2007).



**Figure 22.** Plot of the central business district in Oklahoma City domain made using a GIS shapefile. The Park Avenue street canyon study area is shown in the red dashed box.

To begin examining the influence of the building geometry representation on the simulated flow field, Figure 24 presents velocity vectors overlaid on building footprints at  $z = 1$  m for the time period of 23-00 CDT for the Park Avenue street canyon. The results from both the cut-cell and stair-step simulations using the grid resolutions from Cases 1 and 4 are shown. For Case 1 (finer resolution), both methods capture the geometric details of the buildings, and the velocity fields are similar. This is expected as Case 1 corresponds to the shapefile native resolution so that any resulting geometrical errors are only from grid misalignment, which is identical for the stair-step and cut-cell methods. However, in Case 4 (lower resolution), the stair-step method closes gaps between buildings while the cut-cell method is able to model gap-flow winds and capture more details of the buildings.

Examination of the horizontal velocity vectors and magnitude contours gives a qualitative example of how the cut-cell method improves the representation of building geometry at coarser resolution. A quantitative examination of the advantages provided by the cut-cell method can be made using the JU2003 data. Several sensors are close enough to be used to show the differences between the two methods for simulations with various grid sizes. Figure 25 shows the location of the DSTL-3, LANL-2, LANL-5, LANL-9, and LANL-10 towers (information in Tables 6 and 7) in the Park Avenue street canyon that are close enough to the buildings.

Since the DSTL-3, LANL-9, and LANL-10 are close to the buildings and the height of the sensors is less than the height of the buildings, at low resolution the towers will be located inside the building when using the stair-step method.

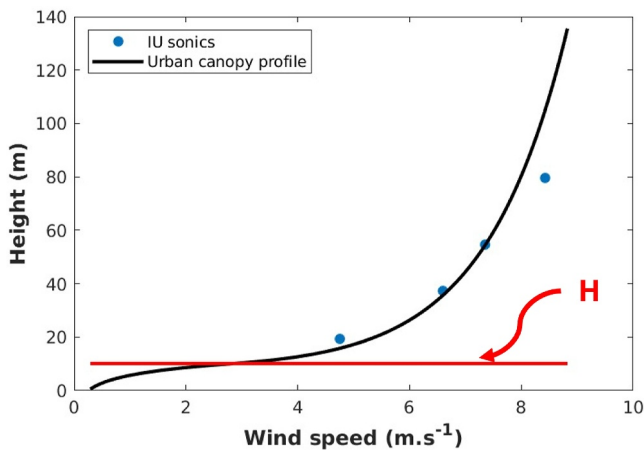
Figures 26–28 plot the tower data against the vertical profiles velocity magnitude for the simulations with the stair-step and cut-cell methods at the location of DSTL-3 (Cases 7 and 11), LANL-9 (Cases 7 and 12) and LANL-10 (Cases 7 and 12), respectively.

The plots indicate that in the lower grid size cases, both methods generate similar velocity profiles, while in the larger grid size cases (lower resolution), the stair-step method considers the location of the tower to be inside the buildings, and as a result, calculates a zero velocity value. The cut-cell method is not without error but its error is significantly less than the stair-step method. Similar vertical profile trends are observed for all 8-hourly averaged time periods for vertical grid sizes of 2 m with the equivalent horizontal grid sizes (Cases 1, 2, 5, and 6).

The next group of sensors considered is those close to buildings but at heights greater than the building height which includes overhanging sensors and sensors on rooftop towers. Figures 29 and 30 present profiles of

**Table 4**  
Values Required to Create the Upstream Velocity Profile for Eight Hourly Averaged Periods

Time period (CDT)	$z_0$ (m)	$z_{ref}$ (m)	$H$ (m)	$u(H)$ (m/s)	$d$ (m)	$\alpha$	$u_{ref}$ (m/s)	Wind direction (degree)
23–0	0.3	17.23	10.1	2.85	2.95	2.41	5.15	153.83
0–1		20.41			2.96		5.61	152.27
1–2		19.92			2.99		5.62	153.67
2–3		18.32			2.90		5.24	158.9
3–4		18.79			2.86		5.23	157.39
4–5		19.19			2.85		5.25	160.68
5–6		17.82			2.93		5.2	158.73
6–7		19.16			2.85		5.25	163.34



**Figure 23.** Urban canopy profile fitted to the Indiana University sensor data for time period of 3–4 CDT on Thursday, 24 July 2003.

tower data against the vertical profiles of the velocity magnitude for the simulations with the stair-step and cut-cell methods at the LANL-2 (Cases 7 and 12) and LANL-5 (Cases 7 and 12) sensor locations, respectively. For LANL-2, the building height is 44 m, and the sensor height is 1.3 m above the rooftop level, which means that the last cell in the vertical direction below the sensor is a cut cell for the cut-cell method but the cell below the sensor treated as completely solid by the stair-step method resulting in the calculation of a zero velocity magnitude (Figure 29a). On the other hand, the cut-cell method correctly simulates a non-zero velocity magnitude value in that cell, and the gradient of velocity is smoother than in the stair-step case. For Figure 29b, for coarser resolution, both methods locate the sensor on a vertical array of cells that are not completely solid cells since the sensor is on the corner of the building and the resolution is coarse enough. The difference is that the cut-cell method treats all cells below the sensor as cut cells and creates higher velocity magnitude because the cells are occupied partially by the building geometry. Meanwhile, the stair-step method generates bigger building geometry which results in slower flows through the cells. The cut-cell method is not perfect, but it is able to model the effect of building geometry on the flow, especially in cells closest to geometry objects, significantly better than the stair-step method.

The LANL-5 tower sensor overhangs the edge of a 47 m tall building (0.5 m outside the building edge), and is located 0.5 m above the building rooftop level. For the fine-resolution cases, both methods correctly consider the cell array below the sensor location as cells outside of the building footprint (Figure 30a). For coarser resolution stair-step results the cell array inside the building geometry and leads to zero velocity magnitude values for cells below the sensor. An additional consequence of this is a discontinuity in the velocity magnitude value at the building height. Meanwhile, the cut-cell method still considers the cell array outside of the building geometry and simulates non-zero velocity magnitude values with a smooth transition at the rooftop level.

Forty seven sensors had continuous or partial measurements during IOP8 (a total of 383 hourly averaged measurements). To statistically compare the cut-cell method and the stair-step method, all available hourly averaged sensor measurements were analyzed. First, the absolute percent errors between the recorded measurements and the simulated values over all 1-hr periods were calculated as

$$\text{Error} = 100 \times \frac{|\text{Measurement} - \text{Simulation}|}{\text{Measurement}}. \quad (64)$$

Next, the percentage of all the measurements in which the percent error of the simulations is in a range for the stair-step and cut-cell methods was calculated. Two threshold values were examined for the tested range of grid sizes given in Table 5: a 50% error threshold, and a 75% error threshold (Figure 31).

At fine horizontal resolutions (<8 m) the stair-step method has slightly better or equal performance compared to the cut-cell method with respect to the percent of sensors with error below either threshold. This suggests that the building geometry of the Park Avenue street canyon is well represented with grid sizes in this range. The better performance of the stair-step method is likely simply a coincidence of grid alignment in this particular street canyon. Once the grid size exceeds 5 m, the cut-cell method starts to show its advantages for both the 50% and

**Table 5**  
*Different Horizontal and Vertical Grid Resolutions and the Corresponding Case Numbers*

Vertical grid size (m)	Horizontal grid size (m)					
	1	2	4	5	8	10
2	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6
3	Case 7	Case 8	Case 9	Case 10	Case 11	Case 12

**Table 6**

*Specifications of the 3D Sonic Anemometers Deployed in the Park Avenue Street Canyon From Allwine and Flaherty (2006) and Nelson et al. (2007)*

Institution	Tower name	Easting (m)	Northing (m)	z (m) (AGL)
Arizona State University	ASU	634,747	3,926,046	2.5, 5, 8.5
Dugway Proving Ground (superPWIDS)	DPG-7	634,595	3,926,049	8
	DPG-8	634,613	3,926,050	
	DPG-9	634,602	3,926,031	
	DPG-10	634,614	3,926,036	
	DPG-11	634,771	3,926,048	
	DPG-12	634,795	3,926,054	
	DPG-13	634,777	3,926,029	
	DPG-14	634,803	3,926,031	
Defense Science and Technology Laboratory	DSTL-1	634,746.2	3,926,038.9	3.5, 5, 6.5
	DSTL-2	634,651.0	3,926,047.3	3, 5, 10
	DSTL-3	634,624.7	3,926,032.1	2
Los Alamos National Laboratory	LANL-1	634,758.6	3,926,031.5	45.3 (1.3 ARL)
	LANL-2	634,766.2	3,926,028.4	47.7 (3.7 ARL)
	LANL-3	634,625.8	3,926,050.4	2.2
	LANL-4	634,725.1	3,926,082.9	47.5 (0.5 ARL)
	LANL-5	634,728.0	3,926,052.2	47.5 (0.5 ARL)
	LANL-6	634,767.9	3,926,052.1	47.5 (0.5 ARL)
University of Oklahoma	OU-1	634,680.4	3,926,047.3	3, 6, 9.9, 15.1
	OU-2	634,685.3	3,926,038.7	1.5, 3, 5.5, 9.9, 15.7
University of Utah	UU-1	634,650.9	3,926,038.3	3.2, 4.2, 5, 7.2, 9.8
	UU-3	634,716.3	3,926,078.9	6 (2 ARL), 7.5 (3.5 ARL), 9 (5 ARL)

*Note.* AGL and ARL stand for above ground level and above rooftop level, respectively. UTM zone is 14 for all instruments in this Table.

75% thresholds. For both methods, the changes in the horizontal grid size affect the accuracy of simulation results more than the influence of the vertical grid size.

#### 4. Conclusions

Solid obstacles change the fluid flow mainly because the flow cannot go through them which means that they significantly influence the velocity field. Improving the representation of geometry details should lead to more accurately simulated wind fields. QES-Winds original geometry representation model, the stair-step or “block

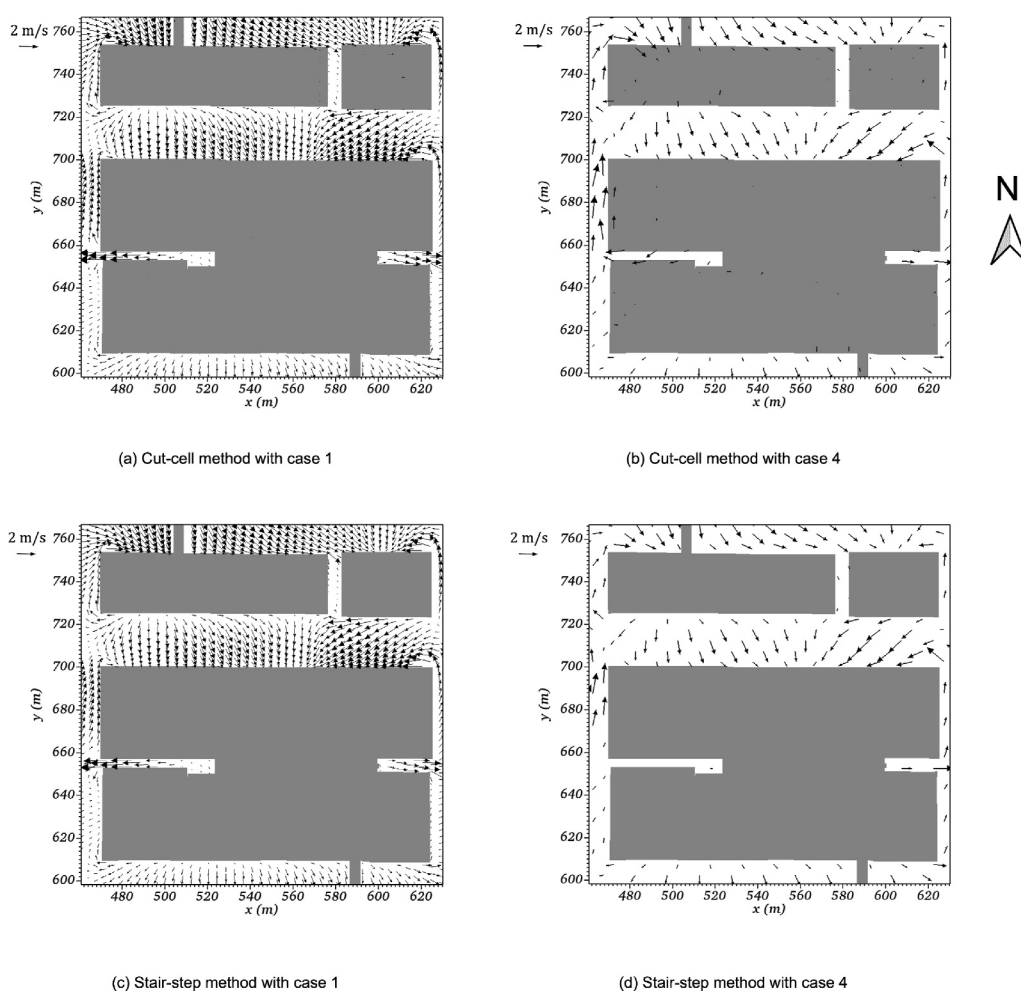
**Table 7**

*Specifications of the 2D Sonic Anemometers Deployed in the Park Avenue Street Canyon*

Institution	Tower name	Easting (m)	Northing (m)	z (m) (AGL)
Los Alamos National Laboratory	LANL-7	634,761.9	3,926,031.4	2.1
	LANL-8	634,756.7	3,926,031.3	
	LANL-9	634,761.7	3,926,052.9	
	LANL-10	634,756.4	3,926,052.9	
	LANL-11	634,625.8	3,926,054.8	
Volpe/University of Central Florida	VOLPE-1	634,619	3,926,053	2
	VOLPE-2	634,619	3,926,032	

*Note.* Information in this table was obtained from Allwine and Flaherty (2006) and Nelson et al. (2007). AGL stands for above ground level. UTM zone is 14 for all instruments in this Table.



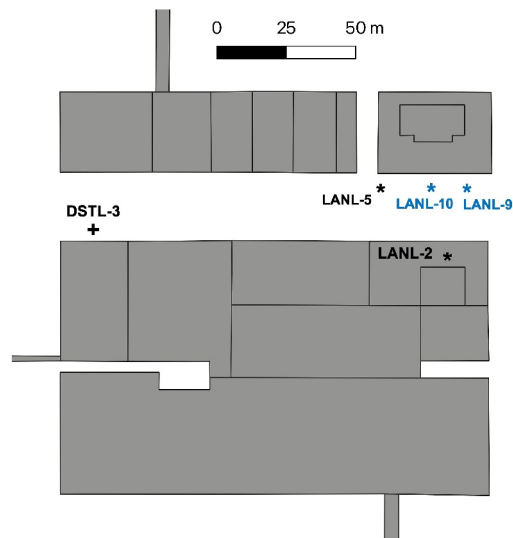


**Figure 24.** Velocity vectors overlaying the building footprint at  $z = 1$  m for the time period of 23-00 CDT for the Park Avenue street canyon from simulations with: (a) Cut-cell method with Case 1, (b) Cut-cell method with Case 4, (c) Stair-step method with Case 1, and (d) Stair-step method with Case 4.

building” method, evaluates cells as fully solid or fully air. Although the stair-step method is simple, easy to implement, and low-cost computationally, as the grid size increases, it misses a great amount of information about the geometry. To include more details of the geometry, a higher-order representation called the cut-cell method has been added to QES-Winds. The cut-cell method provides the ability to have cells that are part solid and part air. The method increases the amount of geometry information inside the cell, which leads to more accurate wind fields. Simulation results of three test cases, Askervein Hill, Granite Mountain (the MATERHORN campaign), and Oklahoma City (JU2003 campaign) have been compared to measured field data to showcase differences between the stair-step and cut-cell methods.

In the Askervein Hill case, a gentle hill with measurement sensors located at 10 m above ground, comparisons to the field data for the stair-step and cut-cell methods are almost identical because the sensor locations are far from the solid geometry. The case was utilized to examine QES-Winds’ general ability to process terrain information and how well the simulation results match the measured data for the flow over complex topography. The comparison also showed the weaknesses and strengths of the mass-consistent approach for flow over complex terrain without any terrain-specific parameterizations.

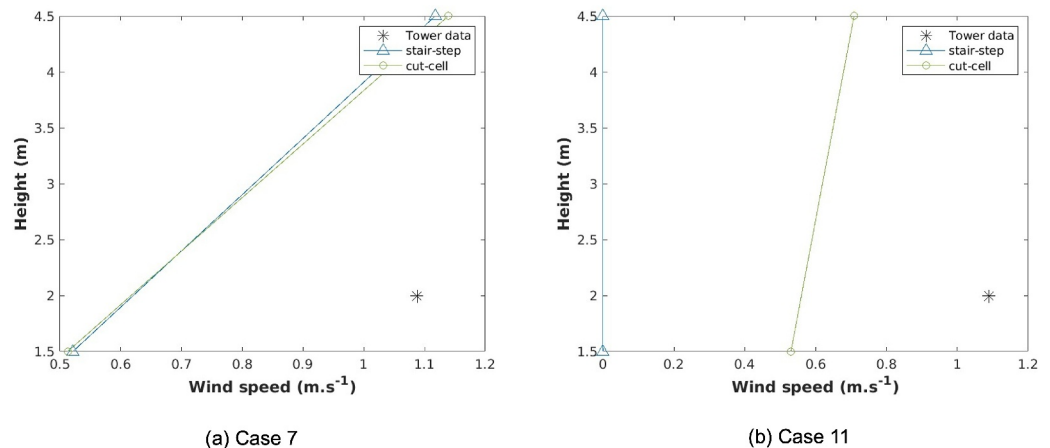
The East Slope of the Granite Mountain, where multiple atmospheric measurement towers were deployed during the spring campaign of the MATERHORN field experiments, was simulated in QES-Winds. A total of 28



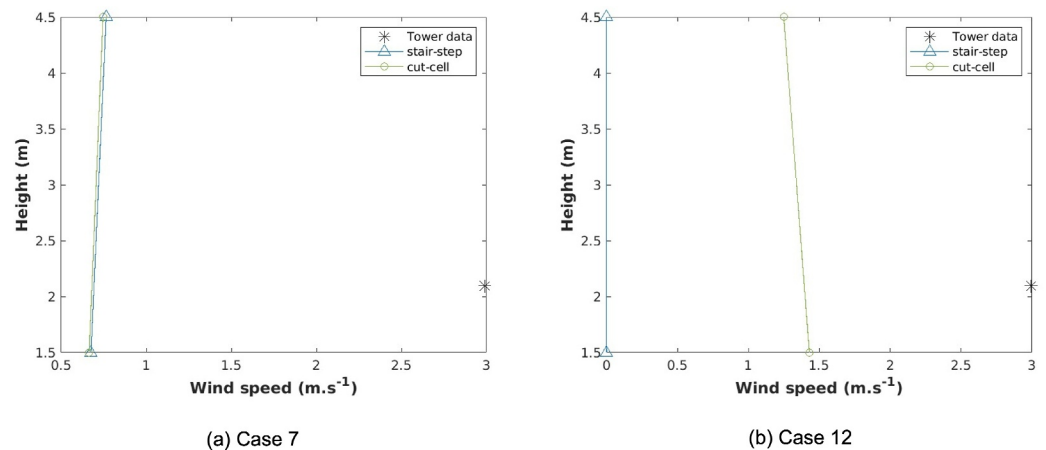
**Figure 25.** Locations of the DSTL-3, LANL-2, LANL-5, LANL-9, and LANL-10 towers in the Park Avenue street canyon (adopted from Allwine and Flaherty (2006)).

instances of 30-min averaged time periods, in which the wind direction was upslope on the eastern side of the Granite Mountain, were analyzed. The computational domain was discretized using a range of horizontal and vertical grid sizes to study the effect of grid resolution on the wind field. The simulation results with the stair-step and cut-cell methods at the location of the ES2 and ES3 towers were compared to the measured data. For most of the finer resolutions, the calculated wind speeds for both methods are almost identical for all measurement locations except for cases where the stair-step method considers the sensor to be in a fully solid cell which results in zero velocity magnitude and a large error. When the vertical grid size is increased, while the cut-cell method does not calculate the most accurate wind speed, the chance of generating a large error by changing the grid size is higher using the stair-step method.

In the final test case, to explore the improvements that the cut-cell method makes over the stair-step method in urban flows, the CBD in Oklahoma City, Oklahoma, USA during the JU2003 field experiments IOP8 time period was modeled. QES-Winds' building wakes and street canyon parameterizations were turned off to isolate the effects of geometry representation methods on the wind field. Several towers were deployed in the Park Avenue street canyon that were either close to building walls or on the rooftop very close to the



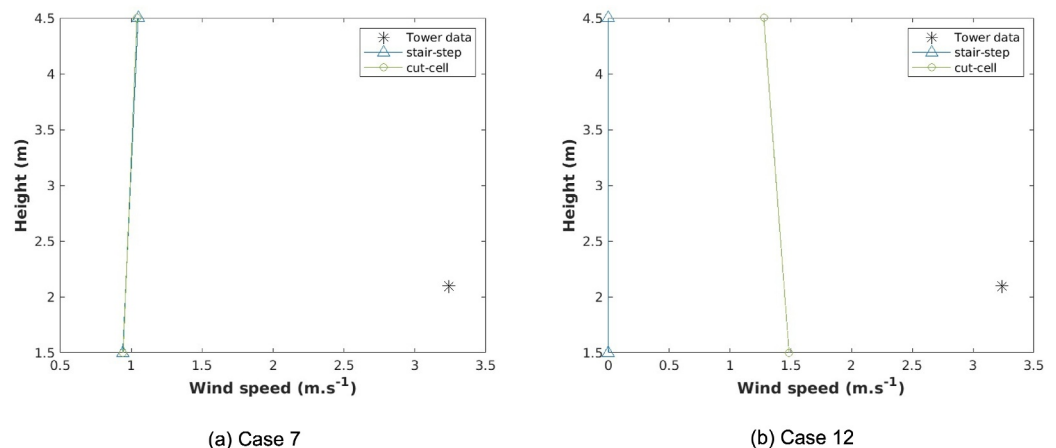
**Figure 26.** Vertical profiles of velocity magnitude at the location of DSTL-3 tower for the time period of 2–3 CDT for simulations with the stair-step and cut-cell methods with resolutions of: (a) Case 7 and (b) Case 11.



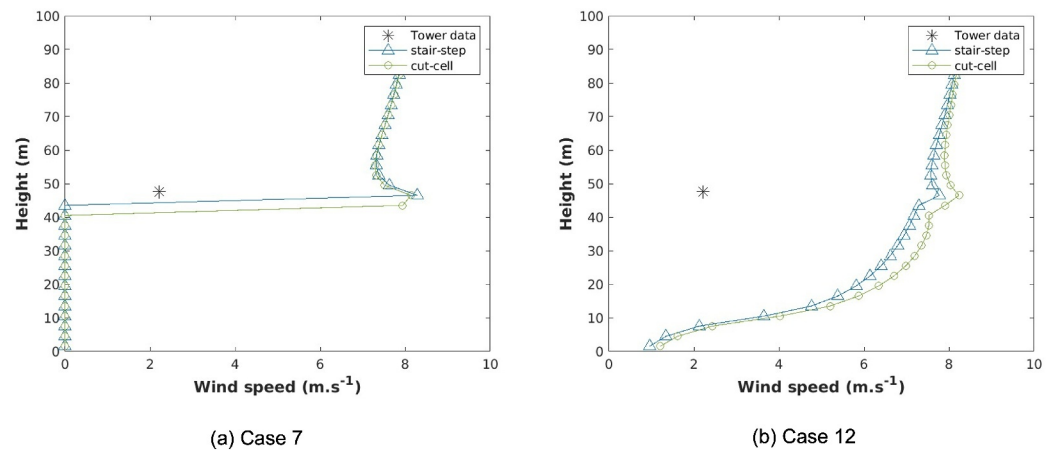
**Figure 27.** Vertical profiles of velocity magnitude at the LANL-9 tower for the time period of 3–4 CDT for simulations with the stair-step and cut-cell methods with the stair-step and cut-cell methods with resolution of: (a) Case 7 and (b) Case 12.

surface. The computational domain was discretized using a range of horizontal and vertical grid sizes. In cases with finer resolution, the simulated results for the stair-step and cut-cell methods are almost identical. As the grid sizes increase, however, the cut-cell method generates less error compared to the stair-step method. The primary reason is that the stair-step method places some sensors in a complete-solid cell or in a full-air cell resulting in the stair-step method generating zero velocity (full-solid cell) at the location of sensors or significantly overestimating the wind speed (full-air cell). Meanwhile, the cut-cell method correctly considers the aforementioned cells as partially solid cells and generates more consistent results over the range of cell sizes. It is expected that the cut-cell method improvements over the stair-step will be more distinctive in modeling particle transport and pollution dispersion in cities. There is a high chance that the stair-step closes the gaps between buildings, forcing particles to go around or over the buildings, especially in higher-density cities with smaller gaps. On the other hand, the cut-cell methods will allow particles to move along gaps which leads to a more realistic simulation of the particle concentration.

In addition to improving QES-Winds near-wall numerical representation, improvements were sought through direct parameterization of the near-wall velocity. This was accomplished through the incorporation of a near-wall, near-surface parameterization. It corrects the velocity component tangent to the surface in the first cell in the wall-normal direction based on the logarithmic assumption. Simulation results that used the new log-law

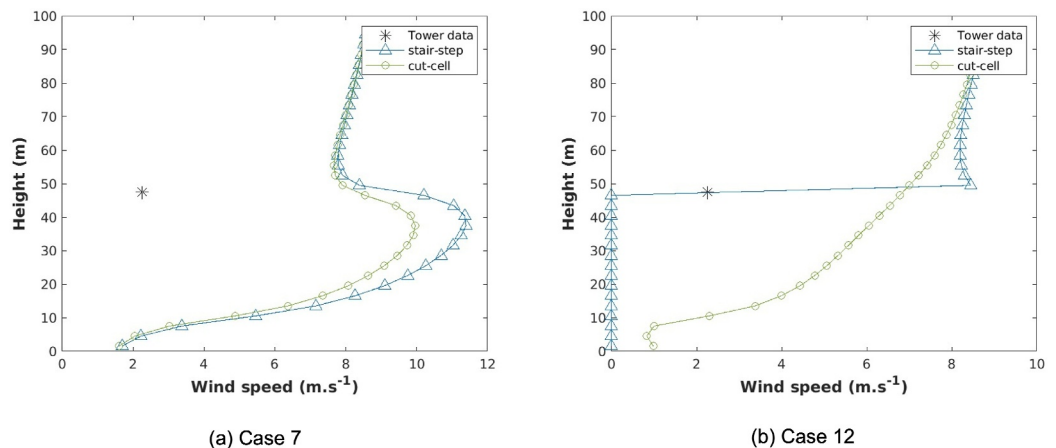


**Figure 28.** Vertical profiles of velocity magnitude at the LANL-10 tower for the time period of 4–5 CDT for simulations with the stair-step and cut-cell methods with resolutions of: (a) Case 7 and (b) Case 12.

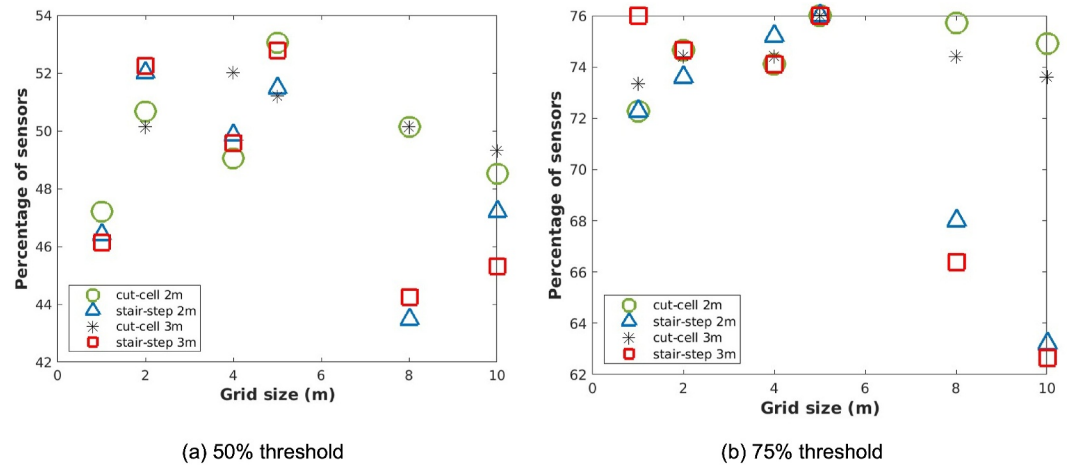


**Figure 29.** Vertical profiles of velocity magnitude at the LANL-2 tower for the time period of 6–7 CDT for simulations with the stair-step and cut-cell methods with the stair-step and cut-cell methods with resolutions of: (a) Case 7 and (b) Case 12.

parameterization for the Granite Mountain case indicated that for both ES towers and most grid sizes, the velocity magnitude is similar to the cut-cell results. This is expected since the log-law parameterization uses the information provided by the cut-cell method to define the normal and tangential directions to the cut face and determine the first cell in the normal direction. The only exceptions are when the first and second sensors at the ES towers are located in the first cell in the direction normal to a cut-cell. In that case, the velocity magnitude in those cells is reduced to match the logarithmic profile. The cells located above the cells affected by the log-law parameterization see a slight increase in the wind speed to make up for lower velocity values in the cells near the ground due to the conservation of mass. We expect the effects of the near-wall parameterization will be more prominent when simulating scalar transport near the surface. A higher velocity gradient near the surface (no parameterization) creates higher transport rate and lower scalar values (temperature or concentration) near the surface.



**Figure 30.** Vertical profiles of velocity magnitude at the LANL-5 tower for the time period of 1–2 CDT for simulations with the stair-step and cut-cell methods with resolutions of: (a) Case 7 and (b) Case 12.



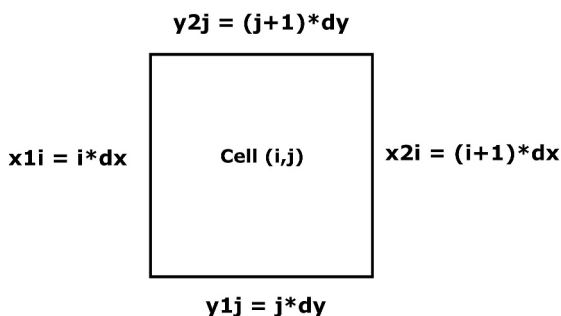
**Figure 31.** Percentage of the simulation results with the stair-step and cut-cell methods with various grid resolution, that are in panels (a) 50% threshold and (b) 75% threshold of the available hourly sensor measurements during IOP8.

## Appendix A: Creating Building Point's List for Cut-Cells

Environmental Systems Research Institute (ESRI) shapefile standards dictate that the points that make up the footprint of a polygon are ordered in a clockwise manner with each line's end point acting as the next line's starting point. Additionally, the last and first points in the list are the same. To process the building footprint, the first step is to evaluate cells that encompass the intersection points (blue letters in Figure 4) of two lines of the building polygon. The starting point of each line is added to the solid vertices for the face below the polygon but the end vertices are added to the list, only if they are located in the same cell as the starting point. For example, point A in Figure 4 is added to the list while processing line AB, but point B is not since it is located outside of the cell that includes point A. Next, the intersection point of the line with the cell's grid lines in the  $x$  and  $y$  directions (as shown in Figure A1), is determined using the slope of the line and the value of  $x$  and/or  $y$  of the grid lines. The intersection point is added to the list of solid vertices for the cell and the associated cell face.

The next step is to loop through all the cells that encompass the line and determine the intersection points of the line with the cell's grid lines (Figure A1). Intersection values,  $y1i$ ,  $y2i$ ,  $x1j$ , and  $x2j$ , are calculated using the equation of a line (line slope), and the cell's grid-line values,  $x1i$ ,  $x2i$ ,  $y1j$ , and  $y2j$ , respectively. If the line does not intersect with any of the cell's grid lines, the associated intersection value will be set to 0. Depending on intersection values, how the line cuts through a cell is defined as one of the categories included in Table A1.

Because the building polygon lines are analyzed one by one, it is impossible to determine the side of the line that is inside the building without additional information. An algorithm that determines whether a point is inside a polygon (PNPOLY-Point Inclusion in Polygon Test (Franklin, 2006)) is utilized to identify which corners of the bottom face of the cell are located inside or outside of the building polygon. If each of the corners is inside the solid, that corner is added to the list of solid points for the cell's faces that intersect at that corner.

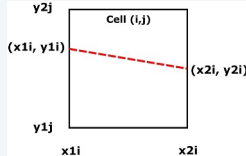
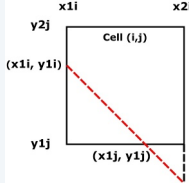
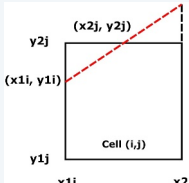
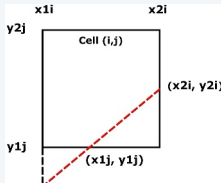
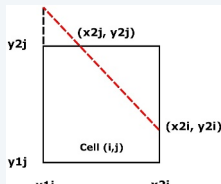
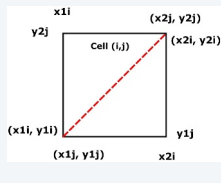


**Figure A1.** Schematic of a cell marked by index  $i$  in the  $x$  direction and index  $j$  in the  $y$  direction. The cell's grid lines are identified by their characteristic values:  $x1i$  and  $x2i$  for vertical lines and  $y1j$  and  $y2j$  for horizontal lines.

Finally, to extend the process in the vertical direction, the height of the building is compared to the cell's top face height. If the building height is greater than or equal to the cell's top face  $z$ -value, the intersection points and corners in the solid are added to the list of solid vertices for the cell's above face. Also, the intersection points and corners in the solid are included in the list of solid vertices for the associated faces (determined by the aforementioned cases) with the cell's top face height. Otherwise, the cell's above face list will be empty and the intersection points and corners in solid are added to the list of solid vertices for the associated faces (determined by the aforementioned cases) with the building height. Table A1 shows different categories for the various possibilities of lines cutting through a cell depending on intersection points.

**Table A1**

*Different Ways Building Wall Cuts Through Grid Cells*

Case	Conditions	Intersection points	Figure
1A	$y1j < y1i < y2j$ $y1j \leq y2i \leq y2j$	$(x1i, y1i)$ $(x2i, y2i)$	
1B	$y1j < y1i < y2j$ $y2i < y1j$	$(x1i, y1i)$ $(x1j, y1j)$	
1C	$y1j < y1i < y2j$ $y2i < y1j$	$(x1i, y1i)$ $(x2j, y2j)$	
2A	$y1j < y2i < y2j$ $y2i < y1j$	$(x2j, y2j)$ $(x1j, y1j)$	
2B	$y1j < y2i < y2j$ $y2j \leq y1i$	$(x2j, y2j)$ $(x2j, y2i)$	
3	$y1j = y1i$ $y2j = y2i$	$(x1j, y1j)$ or $(x1i, y1i)$ $(x2j, y2j)$ or $(x2i, y2i)$	



**Table A1**  
*Continued*

Case	Conditions	Intersection points	Figure
4	$y_{2j} = y_{1i}$ $y_{1j} = y_{2i}$	$(x_{2j}, y_{2j})$ or $(x_{1i}, y_{1i})$ $(x_{1j}, y_{1j})$ or $(x_{2i}, y_{2i})$	
5	$y_{1i} > y_{2j}$ or $y_{1i} < y_{1j}$ $y_{2i} > y_{2j}$ or $y_{2i} < y_{1j}$	$(x_{2j}, y_{2j})$ $(x_{1j}, y_{1j})$	

## Appendix B: Verification Case

As a verification case, we selected potential flow over a hemisphere as complex 3D case with an analytical solution (Cohen & Kundu, 2004):

$$\begin{aligned} v_r &= U \left[ 1 - \left( \frac{a}{r} \right)^3 \right] \cos \theta \\ v_\theta &= -U \left[ 1 + \frac{1}{2} \left( \frac{a}{r} \right)^3 \right] \sin \theta \end{aligned} \quad (\text{B1})$$

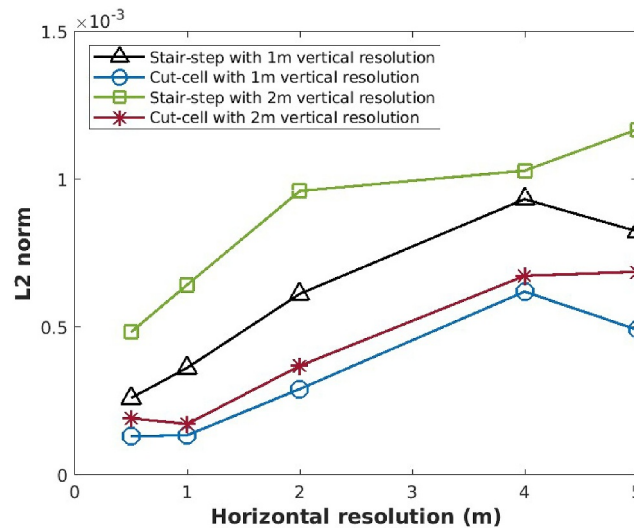
where  $U$  is the uniform upstream velocity,  $a$  is the hemisphere radius, and  $r$  and  $\theta$  are the radial distance and the polar angle in the spherical coordinate system, respectively.

We created a test case for the hemisphere to verify QES-Winds simulations. A DEM with horizontal resolution of 0.5 m is created for the hemisphere with radius of 20 m. We read the DEM into QES-Winds and put it in the middle of a domain that is 200 m by 200 m by 200 m. We run simulations for different horizontal grid sizes of 0.5,

**Table B1**

*Root Mean Square Error and L2 Norm of the Stair-Step and Cut-Cell Method Simulations of a Flow Over a Hemisphere Calculated Against the Analytical Solution for Different Horizontal and Vertical Resolutions*

Horizontal grid size (m)	Method	Vertical grid size (m)	RMSE	L2 norm	Vertical grid size (m)	RMSE	L2 norm
0.5	Stair-step	1	0.327	2.6E-4	2	0.461	4.8E-4
	Cut-cell		0.321	1.3E-4		0.453	1.9E-4
1	Stair-step		0.325	3.6E-4		0.458	6.4E-4
	Cut-cell		0.318	1.4E-4		0.450	1.7E-4
2	Stair-step		0.326	6.1E-4		0.456	9.6E-4
	Cut-cell		0.317	2.9E-4		0.448	3.7E-4
4	Stair-step		0.325	9.3E-4		0.448	10.3E-4
	Cut-cell		0.317	6.2E-4		0.445	6.7E-4
5	Stair-step		0.322	8.2E-4		0.450	11.7E-4
	Cut-cell		0.316	4.9E-4		0.446	6.9E-4



**Figure B1.** Plot of normalized L2 norm as horizontal grid sizes for the cut-cell and stair-step methods and different vertical grid sizes.

1, 2, 4, and 5 m and the vertical grid sizes of 1 and 2 m and with stair-step and cut-cell representations. We also used Equation B1 to create the analytical solution for each resolution.

To quantify the differences between the simulated and analytical wind fields, the RMSE and normalized L2 norm values are calculated:

$$\begin{aligned} \text{RMSE} &= \sqrt{\frac{\sum (U_s - U_a)^2}{N}} \\ \text{L2 norm} &= \frac{\sqrt{\sum (U_s - U_a)^2}}{N} \end{aligned} \quad (\text{B2})$$

where  $U_s$ , and  $U_a$  are the simulated and analytical velocity magnitudes, respectively, and  $N$  is the total number of velocities. Table B1 shows the RMSE and normalized L2 norm values for each method and grid sizes compared to the analytical solution. Figure B1 indicates the normalized L2 norm values versus the horizontal grid sizes for different methods and vertical grid sizes.

These results show that QES-Winds simulations create similar wind fields to the analytical solution. As it can be concluded from the RMSE and normalized L2 norm results and the plot, the cut-cell representation decreases the error and simulates the flow field more accurately than the stair-step method.

### Data Availability Statement

The Quick Environmental System (QES) fast-response wind solver (QES-Winds) has been developed as a collaboration between the University of Utah, the University of Minnesota Duluth, and Pukyong National University. The code is written mainly in C++ and NVIDIA's CUDA language. Version 2.2 of QES-Winds used for development of the terrain and building cut-cell, and the near-wall parameterization, is preserved on GitHub, publicly available via <https://doi.org/10.5281/zenodo.11399262> (Bozorgmehr, Patterson, et al., 2024). The Matlab scripts used for processing simulation data of the three test cases (Askervein Hill, Oklahoma City, and Granite Peak) and the hemisphere verification case and plotting figures in the study along with the processed data are available at Zenodo, publicly available via <https://doi.org/10.5281/zenodo.13988871> (Bozorgmehr, Stoll, & Pardyjak, 2024).

## Acknowledgments

This work was partly supported by a grant from the National Institute of Environment Research, funded by the Ministry of Environment of the Republic of Korea (NIER-SP2019-312), the United States Department of Agriculture National Institute for Food and Agriculture Specialty Crop Research Initiative Award No. 2018-03375, the United States Department of Agriculture Agricultural Research Service through Research Support Agreement 58-2072-0-036, and the United States Department of Agriculture Research Service research project numbers 2072-21000-057-00D and 2072-22000-045-00D.

## References

- Allwine, K. J., & Flaherty, J. E. (2006). *Joint Urban 2003: Study overview and instrument locations* (Tech. Rep.). Pacific Northwest National Lab (PNNL).
- Almgren, A. S., Bell, J. B., Colella, P., & Marthaler, T. (1997). A Cartesian grid projection method for the incompressible Euler equations in complex geometries. *SIAM Journal on Scientific Computing*, 18(5), 1289–1309. <https://doi.org/10.1137/s1064827594273730>
- Anderson, D. M., McFadden, G. B., & Wheeler, A. A. (2000). A phase-field model of solidification with convection. *Physica D: Nonlinear Phenomena*, 135(1–2), 175–194. [https://doi.org/10.1016/s0167-2789\(99\)00109-8](https://doi.org/10.1016/s0167-2789(99)00109-8)
- Arthur, R. S., Lundquist, K. A., & Bao, J. (2018). Evaluating the performance of the immersed boundary method within the grey zone for improved weather forecasts in the HRRR model.
- Arthur, R. S., Lundquist, K. A., Wiersema, D. J., Bao, J., & Chow, F. K. (2020). Evaluating implementations of the immersed boundary method in the weather research and forecasting model. *Monthly Weather Review*, 148(5), 2087–2109. <https://doi.org/10.1175/mwr-d-19-0219.1>
- Bagal, N., Pardyjak, E., & Brown, M. (2004). Improved upwind cavity parameterization for a fast response urban wind model. In *84th annual AMS meeting*.
- Bagal, N., Singh, B., Pardyjak, E., & Brown, M. (2004). Implementation of rooftop recirculation parameterization into the quic fast response urban wind model. In *Proceedings of the 5th AMS urban environment symposium*.
- Bagal, N. L. (2005). *Development and testing of empirical parameterization for quick urban and industrial complex model* (Masters thesis). Department of Mechanical Engineering, University of Utah.
- Balbi, J. H., Chatelon, F. J., Morvan, D., Rossi, J. L., Marcelli, T., & Morandini, F. (2020). A convective–radiative propagation model for wildland fires. *International Journal of Wildland Fire*, 29(8), 723–738. <https://doi.org/10.1071/wf19103>
- Bao, J., Chow, F. K., & Lundquist, K. A. (2018). Large-eddy simulation over complex terrain using an improved immersed boundary method in the weather research and forecasting model. *Monthly Weather Review*, 146(9), 2781–2797. <https://doi.org/10.1175/mwr-d-18-0067.1>
- Barnes, S. L. (1973). Mesoscale objective map analysis using weighted time-series observations. NOAA Technical memorandum ERL ESSL-62.
- Bozorgmehr, B., Patterson, Z., Willemsen, P., Margairaz, F., Gibbs, J., Stoll, R., & Pardyjak, E. R. (2024). Utahefd/qes-public: v2.2.0. *Zenodo*. <https://doi.org/10.5281/zenodo.11399262>
- Bozorgmehr, B., Stoll, R., & Pardyjak, E. R. (2024). Datasets of paper “Improving the performance of a reduced-order mass-consistent model for urban environments and complex terrain with a higher-order geometrical representation”. *Zenodo*. <https://doi.org/10.5281/zenodo.13988871>
- Bozorgmehr, B., Willemsen, P., Gibbs, J. A., Stoll, R., Kim, J.-J., & Pardyjak, E. R. (2021). Utilizing dynamic parallelism in CUDA to accelerate a 3d red-black successive over relaxation wind-field solver. *Environmental Modelling & Software*, 137, 104958. <https://doi.org/10.1016/j.envsoft.2021.104958>
- Bozorgmehr, B., Willemsen, P., Margairaz, F., Gibbs, J. A., Patterson, Z., Stoll, R., & Pardyjak, E. R. (2021). QES-Winds v1.0: Theory and user’s guide.
- Briggs, K. A. (2015). *Evaluation of moisture and heat transport in the fast-response building-resolving urban transport code QUIC Envsim* (Masters thesis). The University of Utah.
- Chester, S., Meneveau, C., & Parlange, M. B. (2007). Modeling turbulent flow over fractal trees with renormalized numerical simulation. *Journal of Computational Physics*, 225(1), 427–448. <https://doi.org/10.1016/j.jcp.2006.12.009>
- Clarke, D. K., Salas, M., & Hassan, H. (1986). Euler calculations for multielement airfoils using Cartesian grids. *AIAA Journal*, 24(3), 353–358. <https://doi.org/10.2514/3.9273>
- Cohen, I. M., & Kundu, P. K. (2004). *Fluid mechanics*. Elsevier.
- Dang, S. T., Johansen, S. T., & Meese, E. A. (2017). A Cartesian cut-cell method, based on formal volume averaging of mass, momentum equations. In *Progress in applied CFD—CFD2017 selected papers from 12th international conference on computational fluid dynamics in the oil & gas, metallurgical and process industries*.
- DeLeon, R., Sandusky, M., & Senocak, I. (2018). Simulations of turbulent flow over complex terrain using an immersed-boundary method. *Boundary-Layer Meteorology*, 167(3), 399–420. <https://doi.org/10.1007/s10546-018-0336-8>
- Deng, X., & Stull, R. (2005). A mesoscale analysis method for surface potential temperature in mountainous and coastal terrain. *Monthly Weather Review*, 133(2), 389–408. <https://doi.org/10.1175/mwr-2859.1>
- De Wekker, S. F., & Kossmann, M. (2015). Convective boundary layer heights over mountainous terrain—A review of concepts. *Frontiers in Earth Science*, 3, 77. <https://doi.org/10.3389/feart.2015.00077>
- Fernando, H., Pardyjak, E., Di Sabatino, S., Chow, F., De Wekker, S., Hoch, S., et al. (2015). The materhorn: Unraveling the intricacies of mountain weather. *Bulletin of the American Meteorological Society*, 96(11), 1945–1967. <https://doi.org/10.1175/bams-d-13-00131.1>
- Finnigan, J., Ayotte, K., Harman, I., Katul, G., Oldroyd, H., Patton, E., et al. (2020). Boundary-layer flow over complex topography. *Boundary-Layer Meteorology*, 177(2), 247–313. <https://doi.org/10.1007/s10546-020-00564-3>
- Forthofer, J. M., Butler, B. W., & Wagenbrenner, N. S. (2014). A comparison of three approaches for simulating fine-scale surface winds in support of wildland fire management. Part I. Model formulation and comparison against measurements. *International Journal of Wildland Fire*, 23(7), 969–981. <https://doi.org/10.1071/wf12089>
- Franklin, W. R. (2006). Pnpoly-point inclusion in polygon test. Web site. [http://www.ecse.rpi.edu/Homepages/wrf/Research/Short\\_Notes/pnpoly.html](http://www.ecse.rpi.edu/Homepages/wrf/Research/Short_Notes/pnpoly.html)
- Gowardhan, A. A., Brown, M. J., & Pardyjak, E. R. (2010). Evaluation of a fast response pressure solver for flow around an isolated cube. *Environmental Fluid Mechanics*, 10(3), 311–328. <https://doi.org/10.1007/s10652-009-9152-5>
- Grimmond, C. S. B., Su, H., Offerle, B., Crawford, B., Scott, S., Zhong, S., & Clements, C. (2004). Spatial variability of heat fluxes in a suburban area of Oklahoma city. In *Preprints, fifth symposium on the urban environment* (Vol. 3). American Meteorological Society.
- Gunawardena, N., Durand, P., Hedde, T., Dupuy, F., & Pardyjak, E. (2022). Data filling of micrometeorological variables in complex terrain for high-resolution nowcasting. *Atmosphere*, 13(3), 408. <https://doi.org/10.3390/atmos13030408>
- Hayati, A. N. (2018). *A computational study of momentum and scalar transport in urban areas* (Doctoral dissertation). The University of Utah.
- Hayati, A. N., Stoll, R., Kim, J., Harman, T., Nelson, M. A., Brown, M. J., & Pardyjak, E. R. (2017). Comprehensive evaluation of fast-response, Reynolds-averaged Navier–Stokes, and large-eddy simulation methods against high-spatial-resolution wind-tunnel data in step-down street canyons. *Boundary-Layer Meteorology*, 164(2), 217–247. <https://doi.org/10.1007/s10546-017-0245-2>
- Hayati, A. N., Stoll, R., Pardyjak, E. R., Harman, T., & Kim, J. (2019). Comparative metrics for computational approaches in non-uniform street-canyon flows. *Building and Environment*, 158, 16–27. <https://doi.org/10.1016/j.buildenv.2019.04.028>
- Hou, G., Wang, J., & Layton, A. (2012). Numerical methods for fluid-structure interaction—A review. *Communications in Computational Physics*, 12(2), 337–377. <https://doi.org/10.4208/cicp.291210.290411s>

- Ingram, D. M., Causon, D. M., & Mingham, C. G. (2003). Developments in Cartesian cut cell methods. *Mathematics and Computers in Simulation*, 61(3–6), 561–572. [https://doi.org/10.1016/s0378-4754\(02\)00107-6](https://doi.org/10.1016/s0378-4754(02)00107-6)
- Jensen, D. D., Nadeau, D. F., Hoch, S. W., & Pardyjak, E. R. (2017). The evolution and sensitivity of katabatic flow dynamics to external influences through the evening transition. *Quarterly Journal of the Royal Meteorological Society*, 143(702), 423–438. <https://doi.org/10.1002/qj.2932>
- Kaplan, H., & Dinar, N. (1996). A Lagrangian dispersion model for calculating concentration distribution within a built-up domain. *Atmospheric Environment*, 30(24), 4197–4207. [https://doi.org/10.1016/1352-2310\(96\)00144-6](https://doi.org/10.1016/1352-2310(96)00144-6)
- Ketefian, G., & Jacobson, M. Z. (2009). A mass, energy, vorticity, and potential enstrophy conserving lateral fluid–land boundary scheme for the shallow water equations. *Journal of Computational Physics*, 228(1), 1–32. <https://doi.org/10.1016/j.jcp.2008.08.009>
- Ketefian, G., & Jacobson, M. Z. (2011). A mass, energy, vorticity, and potential enstrophy conserving lateral boundary scheme for the shallow water equations using piecewise linear boundary approximations. *Journal of Computational Physics*, 230(8), 2751–2793. <https://doi.org/10.1016/j.jcp.2010.11.008>
- Kevlahan, N.-R., Dubos, T., & Aechtner, M. (2015). Adaptive wavelet simulation of global ocean dynamics using a new brinkman volume penalization. *Geoscientific Model Development*, 8(12), 3891–3909. <https://doi.org/10.5194/gmd-8-3891-2015>
- Koch, S. E., DesJardins, M., & Kocin, P. J. (1983). An interactive Barnes objective map analysis scheme for use with satellite and conventional data. *Journal of Applied Meteorology and Climatology*, 22(9), 1487–1503. [https://doi.org/10.1175/1520-0450\(1983\)022<1487:aiboma>2.0.co;2](https://doi.org/10.1175/1520-0450(1983)022<1487:aiboma>2.0.co;2)
- Lehner, M., Whiteman, C. D., Hoch, S. W., Jensen, D., Pardyjak, E. R., Leo, L. S., et al. (2015). A case study of the nocturnal boundary layer evolution on a slope at the foot of a desert mountain. *Journal of Applied Meteorology and Climatology*, 54(4), 732–751. <https://doi.org/10.1175/jamc-d-14-0223.1>
- Lundquist, K. A., Chow, F. K., & Lundquist, J. K. (2010). An immersed boundary method for the weather research and forecasting model. *Monthly Weather Review*, 138(3), 796–817. <https://doi.org/10.1175/2009mwr2990.1>
- Lundquist, K. A., Chow, F. K., & 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. <https://doi.org/10.1175/mwr-d-11-00311.1>
- Marella, S., Krishnan, S., Liu, H., & Udaykumar, H. (2005). Sharp interface Cartesian grid method i: An easily implemented technique for 3d moving boundary computations. *Journal of Computational Physics*, 210(1), 1–31. <https://doi.org/10.1016/j.jcp.2005.03.031>
- Margairaz, F., Singh, B., Gibbs, J. A., Atwood, L., Pardyjak, E. R., & Stoll, R. (2023). QES-Plume v1. 0: A Lagrangian dispersion model. *Geoscientific Model Development*, 16(20), 5729–5754. <https://doi.org/10.5194/gmd-16-5729-2023>
- Mickle, R., Cook, N., Hoff, A., Jensen, N., Salmon, J., Taylor, P., et al. (1988). The Askervein hill project: Vertical profiles of wind and turbulence. *Boundary-Layer Meteorology*, 43(1), 143–169. <https://doi.org/10.1007/bf00153977>
- Mittal, R., & Iaccarino, G. (2005). Immersed boundary methods. *Annual Review of Fluid Mechanics*, 37(1), 239–261. <https://doi.org/10.1146/annurev.fluid.37.061903.175743>
- Monin, A. S., & Obukhov, A. M. (1954). Basic laws of turbulent mixing in the surface layer of the atmosphere. *Contrib. Geophys. Inst. Acad. Sci. USSR*, 151(163), e187.
- Moody, M. J., Gibbs, J. A., Krueger, S., Mallia, D., Pardyjak, E. R., Kochanski, A. K., et al. (2022). QES-Fire: A dynamically coupled fast-response wildfire model. *International Journal of Wildland Fire*, 31(3), 306–325. <https://doi.org/10.1071/wf21057>
- Moussafir, J., Oldrini, O., Tinarelli, G., Sontowski, J., & Dougherty, C. M. (2004). 5.26 a new operational approach to deal with dispersion around obstacles: The MSS (micro swift spray) software suite.
- Nelson, M., Pardyjak, E. R., Klewicki, J., Pol, S., & Brown, M. (2007). Properties of the wind field within the Oklahoma City Park Avenue street canyon. Part I: Mean flow and turbulence statistics. *Journal of Applied Meteorology and Climatology*, 46(12), 2038–2054. <https://doi.org/10.1175/2006jamc1427.1>
- Neophytou, M., Gowardhan, A., & Brown, M. (2011). An inter-comparison of three urban wind models using Oklahoma City joint urban 2003 wind field measurements. *Journal of Wind Engineering and Industrial Aerodynamics*, 99(4), 357–368. <https://doi.org/10.1016/j.jweia.2011.01.010>
- Pardyjak, E. R., & Brown, M. (2003). QUIC-URB v. 1.1: Theory and user's guide.
- Pardyjak, E. R., & Brown, M. J. (2001). *Evaluation of a fast-response urban wind model-comparison to single-building wind tunnel data* (Tech. Rep.). Los Alamos National Lab.
- Peskin, C. S. (1972). Flow patterns around heart valves: A numerical method. *Journal of Computational Physics*, 10(2), 252–271. [https://doi.org/10.1016/0021-9991\(72\)90065-4](https://doi.org/10.1016/0021-9991(72)90065-4)
- Pol, S., Bagal, N., Singh, B., Brown, M., & Pardyjak, E. (2006). Implementation of a rooftop recirculation parameterization into the quic fast response urban wind model.
- Röckle, R. (1990). *Bestimmung der strömungsverhältnisse im bereich komplexer bebauungsstrukturen* (PhD thesis). Vom fachbereich Mechanik, der technischen hochschule Darmstadt.
- Salesky, S., Giometto, M. G., Chamecki, M., Lehning, M., & Parlange, M. B. (2017). Large eddy simulation modeling of particle-laden flows in complex terrain. In *AGU fall meeting abstracts*.
- Salesky, S. T., Giometto, M. G., Chamecki, M., Lehning, M., & Parlange, M. B. (2019). The transport and deposition of heavy particles in complex terrain: Insights from an Eulerian model for large eddy simulation. arXiv preprint arXiv:1903.03521.
- Sasaki, Y. (1958). An objective analysis based on the variational method. *Journal of the Meteorological Society of Japan. Ser. II*, 36(3), 77–88. [https://doi.org/10.2151/jmsj1923.36.3\\_77](https://doi.org/10.2151/jmsj1923.36.3_77)
- Sasaki, Y. (1970a). Some basic formalisms in numerical variational analysis. *Monthly Weather Review*, 98(12), 875–883. [https://doi.org/10.1175/1520-0493\(1970\)098<0875:SBFINV>2.3.CO;2](https://doi.org/10.1175/1520-0493(1970)098<0875:SBFINV>2.3.CO;2)
- Sasaki, Y. (1970b). Numerical variational analysis formulated under the constraints as determined by longwave equations and a low-pass filter. *Monthly Weather Review*, 98(12), 884–898. [https://doi.org/10.1175/1520-0493\(1970\)098<0884:nvafut>2.3.CO;2](https://doi.org/10.1175/1520-0493(1970)098<0884:nvafut>2.3.CO;2)
- Scardovelli, R., & Zaleski, S. (1999). Direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics*, 31(1), 567–603. <https://doi.org/10.1146/annurev.fluid.31.1.567>
- Scire, J. S., Robe, F. R., Fernau, M. E., & Yamartino, R. J. (2000). *A user's guide for the CALMET meteorological model* (Vol. 37). Earth Tech.
- Sherman, C. A. (1978). A mass-consistent model for wind fields over complex terrain. *Journal of Applied Meteorology and Climatology*, 17(3), 312–319. [https://doi.org/10.1175/1520-0450\(1978\)017<0312:amcmfw>2.0.co;2](https://doi.org/10.1175/1520-0450(1978)017<0312:amcmfw>2.0.co;2)
- Singh, B., Hansen, B. S., Brown, M. J., & Pardyjak, E. R. (2008). Evaluation of the quic-urb fast response urban wind model for a cubical building array and wide building street canyon. *Environmental Fluid Mechanics*, 8(4), 281–312. <https://doi.org/10.1007/s10652-008-9084-5>
- Stull, R. B. (1988). *An introduction to boundary layer meteorology* (Vol. 13). Springer Science & Business Media.

- Stull, R. B. (1992). A theory for mixed-layer-top levelness over irregular topography. In *Preprints, 10th symposium on turbulence and diffusion*. American Meteorological Society.
- Sussman, M., Almgren, A. S., Bell, J. B., Colella, P., Howell, L. H., & Welcome, M. L. (1999). An adaptive level set approach for incompressible two-phase flows. *Journal of Computational Physics*, 148(1), 81–124. <https://doi.org/10.1006/jcph.1998.6106>
- Taylor, P., & Teunissen, H. (1987). The Askervein hill project: Overview and background data. *Boundary-Layer Meteorology*, 39(1), 15–39. <https://doi.org/10.1007/bf00121863>
- Tinarelli, G., Brusasca, G., Oldrini, O., Anfossi, D., Castelli, S. T., & Moussafir, J. (2007). Micro-swift-spray (MSS): A new modelling system for the simulation of dispersion at microscale. General description and validation. In *Air pollution modeling and its application xvii* (pp. 449–458). Springer.
- Tucker, P., & Pan, Z. (2000). A Cartesian cut cell method for incompressible viscous flow. *Applied Mathematical Modelling*, 24(8–9), 591–606. [https://doi.org/10.1016/s0307-904x\(00\)00005-6](https://doi.org/10.1016/s0307-904x(00)00005-6)
- Udaykumar, H., Mittal, R., & Rampunggoon, P. (2002). Interface tracking finite volume method for complex solid–fluid interactions on fixed meshes. *Communications in Numerical Methods in Engineering*, 18(2), 89–97. <https://doi.org/10.1002/cnm.468>
- Udaykumar, H., Mittal, R., Rampunggoon, P., & Khanna, A. (2001). A sharp interface Cartesian grid method for simulating flows with complex moving boundaries. *Journal of Computational Physics*, 174(1), 345–380. <https://doi.org/10.1006/jcph.2001.6916>
- Udaykumar, H., Shyy, W., & Rao, M. (1996). A mixed Eulerian–Lagrangian method for fluid flows with complex and moving boundaries. *International Journal for Numerical Methods in Fluids*, 22(8), 691–712. [https://doi.org/10.1002/\(sici\)1097-0363\(19960430\)22:8<691::aid-fld371>3.0.co;2-u](https://doi.org/10.1002/(sici)1097-0363(19960430)22:8<691::aid-fld371>3.0.co;2-u)
- Ulmer, L., Margairaz, F., Bailey, B. N., Mahaffee, W. F., Pardyjak, E. R., & Stoll, R. (2023). A fast-response, wind angle-sensitive model for predicting mean winds in row-organized canopies. *Agricultural and Forest Meteorology*, 329, 109273. <https://doi.org/10.1016/j.agrformet.2022.109273>
- Umphrey, C., DeLeon, R., & Senocak, I. (2017). Direct numerical simulation of turbulent katabatic slope flows with an immersed-boundary method. *Boundary-Layer Meteorology*, 164(3), 367–382. <https://doi.org/10.1007/s10546-017-0252-3>
- Wang, S.-L., Sekerka, R., Wheeler, A., Murray, B., Coriell, S., Braun, R., & McFadden, G. (1993). Thermodynamically-consistent phase-field models for solidification. *Physica D: Nonlinear Phenomena*, 69(1–2), 189–200. [https://doi.org/10.1016/0167-2789\(93\)90189-8](https://doi.org/10.1016/0167-2789(93)90189-8)
- Wang, W., Shaw, W. J., Seiple, T. E., Rishel, J. P., & Xie, Y. (2008). An evaluation of a diagnostic wind model (CALMET). *Journal of Applied Meteorology and Climatology*, 47(6), 1739–1756. <https://doi.org/10.1175/2007jamc1602.1>
- Williams, M. D., Brown, M. J., Boswell, D., Singh, B., & Pardyjak, E. R. (2004). *Testing of the quic-plume model with wind-tunnel measurements for a high-rise building* (Tech. Rep.). Los Alamos National Laboratory.
- Yang, J., & Balaras, E. (2006). An embedded-boundary formulation for large-eddy simulation of turbulent flows interacting with moving boundaries. *Journal of Computational Physics*, 215(1), 12–40. <https://doi.org/10.1016/j.jcp.2005.10.035>
- Ye, T., Mittal, R., Udaykumar, H., & Shyy, W. (1999). An accurate Cartesian grid method for viscous incompressible flows with complex immersed boundaries. *Journal of Computational Physics*, 156(2), 209–240. <https://doi.org/10.1006/jcph.1999.6356>
- Yuan, J., & Piomelli, U. (2014a). Estimation and prediction of the roughness function on realistic surfaces. *Journal of Turbulence*, 15(6), 350–365. <https://doi.org/10.1080/14685248.2014.907904>
- Yuan, J., & Piomelli, U. (2014b). Numerical simulations of sink-flow boundary layers over rough surfaces. *Physics of Fluids*, 26(1), 015113. <https://doi.org/10.1063/1.4862672>