Autor's response to Reviewer

Referee's comment-1: Very last line of page 1, when talking about staggered-grid finite-difference: It would be good to be more precise in this statement. If it's the typical staggered-grid for the E-field equation, then it's the tangential electric field, I believe, that's continuous between cells, with no built-in condition on the current. This is probably getting far too pedantic, so feel free to ignore. But, it would be possible to formulate the typical integral equation approach (whether one that doesn't work for high contrast or one that does) using, e.g., unstructured tetrahedral meshes, to accurately model a real-life ore-body, etc. And it is possible to formulate staggered-grid finite-difference (or "finite-volume") for, e.g., unstructured tetrahedral meshes. I'd argue, therefore, that it's not so much the methods themselves that are limited to rectangular discretizations but the implementations that people have derived and coded up.

Author's response:

Thank you for this comment. Referee #1 is correct in pointing out that I should be more specific about the field continuity properties. In typical staggered-grid formulations for the E-field equation, it is indeed the tangential electric field components that are continuous between cells, without built-in conditions on current density. This distinction is essential for accurately describing how electromagnetic boundary conditions are naturally enforced in these schemes.

To clarify the distinction between implementation and theory, your key point about differentiating the capabilities of theoretical methods from the limitations of practical implementation is particularly valuable, and I will emphasize this in my revision after revisiting the integral equation and staggered-grid schemes. I agree with referee#1 that both integral equation approaches and staggered-grid schemes can, in principle, be formulated on unstructured tetrahedral meshes. This allows for accurate modeling of complex geometries, such as those found in real-world ore bodies. The apparent limitation of rectangular discretizations reflects choices in implementation rather than fundamental mathematical constraints. I look forward to revising and improving my manuscript to provide clearer explanations and corrections.

Referee's comment-2: Lines 37-38: One problem with the simple rectilinear meshes is the poor mesh quality, and hence ill-conditioning of the system, resulting from extending padding zones out to the boundary of the domain, which ends up with very pizza-box-like cells. This is very nicely illustrated by your Figure 5.

Author's response:

Thank you very much for this comment. The above comments can support the concept of our mesh algorithm. Therefore, I will add this issue to our revised manuscript to clarify the existing context.

Referee's comment-3: What field do you use for your boundary conditions: the electric field in a homogeneous halfspace (and if so, what's the conductivity of the halfspace?), or in a horizontally layered Earth, or in 2D models?

Author's response:

As with other FE works, for each side of the domain, the boundary is the 2-D boundary. So, the boundary conditions for the electric field are calculated depending on the resistivity structure on the boundary side of the models. For the first model, the boundary conditions are calculated using a two-layered model with resistivities of 100 ohm-m and 10

ohm-m. For the second model, the resistivity structure on each side of the domain is a three-layer model with 10 ohm-m, 100 ohm-m, and 0.1 ohm-m; therefore, the boundary conditions must be calculated from this layered model. For the third model, the resistivity structure on each side remains a 100 ohm-m half-space model, although the trapezoidal region is located at the center of the domain. The boundary condition is calculated based on the half-space model.

Note that Figure 1 will be revised to make it clearer and satisfy the boundary conditions and interior Air-Earth interface conditions, as in other FE works.

Referee's comment-4: Why use a mesh made from quadrilateral elements and not one made from triangular elements, and then extrude those triangles downwards?

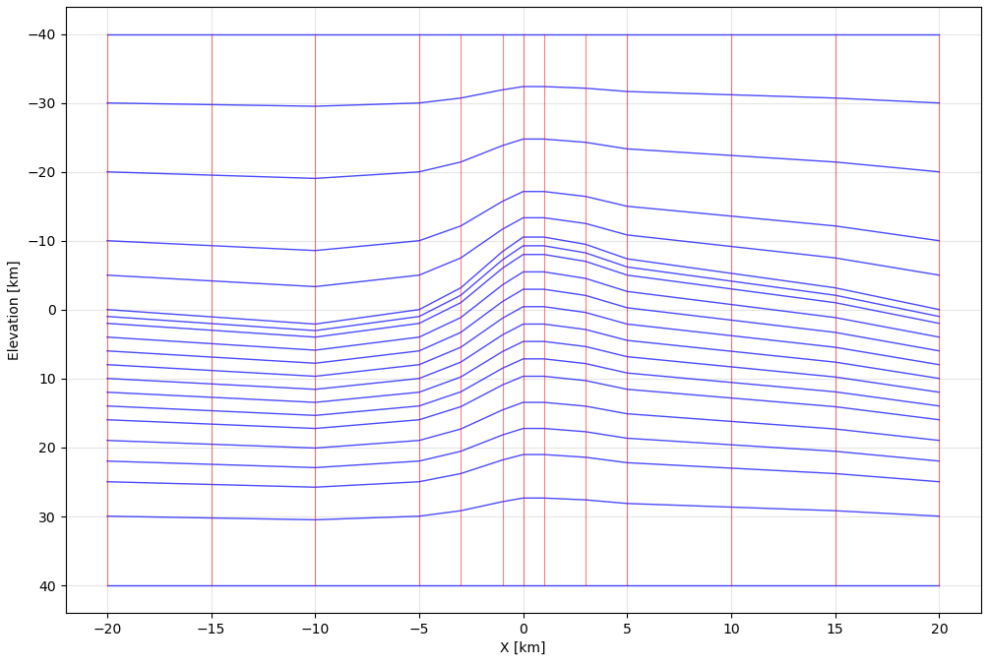
Author's response:

As we know, to date, the unstructured conformal hexahedral mesh is not available for surface meshes. We aim to utilize a semi-structured conformal hexahedral mesh (our method is based on the same concept as sweeping or extrusion) for the edge-based finite element method in 3D MT modeling. Suppose we start with a triangular face and exclude it in the z-direction. The obtaining element is a (distorted) triangular prism that is not hexahedral. Each face of a hexahedron must be a quadrangle. The extrude from the quadrilateral mesh forms the conformal hexahedral mesh. The extrusion for each mesh section may not be uniform, and for each section, it can be adjusted to handle irregular surfaces (such as terrain or non-flat seafloor). However, the number of mesh points or nodes for each mesh section must be constant.

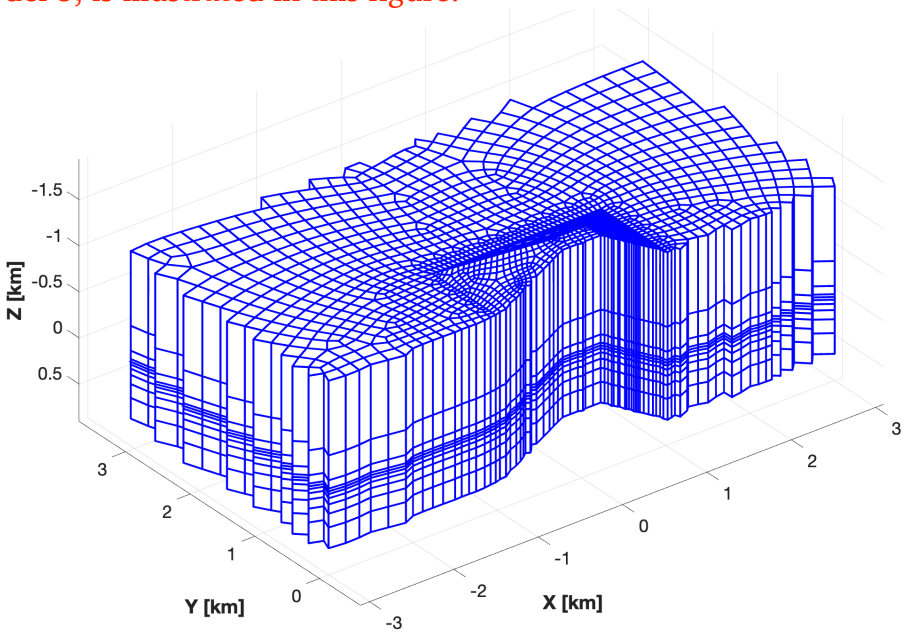
Referee's comment-5: Lines 96-97: This is perhaps slightly important. Can your meshing, and hence the forward-modelling code, handle cells of variable vertical extent? This is what the sentence over these two lines suggests. However, there is nothing else in the manuscript that mentions this. If your method (meshing & MT modelling) can handle this, it would be good to see this shown off and demonstrated in an example (perhaps the trapezoidal hill model of Nam et al.).

Author's response:

In this case, the model that will be mentioned may include irregularly embedded anomalies, irregular layers, or an irregular earth surface, among others. The first two benchmark models include only regular anomalies. So, our mention will not be relevant. For the last model, this feature can be applicable if we show the mesh section both above and beneath the Earth's surface. For this model, our approach is to build upon the work of Name et al. (2007). For the mesh section at the Earth's surface, the coordinates of some mesh nodes are designed to accommodate the trapezoidal hill; otherwise, the coordinate mesh nodes lie at the same depth, $z = 0$ km. For the other mesh sections, the z-coordinates of the nodes are adjusted in the topography projection while maintaining the validity of the mesh. When the mesh section is far away from the Earth's surface (both above in air or beneath in the Earth's subsurface), the z-coordinates of nodes will be less adjusted and become closer to or the same as the nodes that are located outside that projection to make the mesh more relaxed and of better quality (less distortion). This concept is illustrated in the following example figures.



The above is the 2D case. The concept for the 3-D case with a topographic model, especially model 3, is illustrated in this figure.



The other views are shown in the replying comment-10. Note that the elements, which are all part of the volume located inside the specified rectangular subdomain and mesh sections, are filtered to be plotted. Note that we will include these Figures or an improved version in the revised version to make this mesh-type feature more relevant.

Referee's comment-6: Figure 1: In this figure you're meaning to indicate on which boundaries of the domain the (non-zero) tangential component of the background electric field is being applied for the two different polarizations, is that correct? But you still need to provide conditions on the other boundaries, it's just that these involve forcing the tangential electric field to zero?

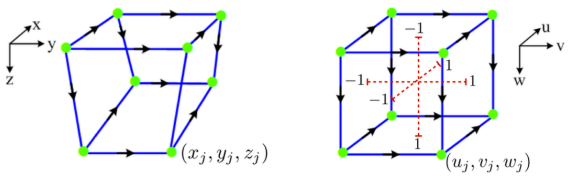
Author's response:

As you mention. In our code, we set it up as you describe. So, our Figure 1 needs to be edited to ensure it is valid for practical boundary conditions.

Referee's comment-7: Line 149: Can't you use analytic formulae for the integrals? Why use a numerical integration technique? Doesn't this take quite a bit longer?

Author's response:

Usually, if the elements are rectangular prisms, we can compute analytically (see "The Finite Element Method in Electromagnetics" by Jian-Ming Jin). However, our elements are distorted hexahedral elements. As we have stated, the distorted hexahedral elements in the original coordinates (x, y, z) need to be transformed into the 2-unit cubic component in the (u, v, w) coordinate system. The transformation is illustrated in the following figure.



(a) Hexahedral element in cartesian coordinate

(b) 2-unit cubic element in reference coordinate

The coordinate transformation is defined by

$$x = \sum_{i=1}^{8} N_i^e(u, v, w) x_i,$$

$$y = \sum_{i=1}^{8} N_i^e(u, v, w) y_i,$$

$$z = \sum_{i=1}^{8} N_i^e(u, v, w) z_i$$

where

$$N_i^e(u, v, w) = \frac{1}{8}\left(1 + u_i u\right)\left(1 + v_i v\right)\left(1 + w_i w\right),$$

for $i = 1,2,3,\ldots,8$.

The vector basis functions $\mathbf{W}_j^e$ need to be classified into three groups. Each group contains four vector basis functions. For the first group which includes the edges parallel to the u-direction, the vector basis functions are defined by

$$\mathbf{W}_j^e = \frac{l_j^e}{8}\left(1 + v_j v\right)\left(1 + w_j w\right)\nabla u, \quad \text{for } j = 1,2,3,4.$$

For the second group that contains the edges parallel to the v-direction, the vector basis functions are defined by

$$\mathbf{W}_j^e = \frac{l_j^e}{8}\left(1 + u_j u\right)\left(1 + w_j w\right)\nabla v, \quad \text{for } j = 5,6,7,8.$$

For the last group that contains the edges parallel to the w-direction, the vector basis functions are defined by

$$\mathbf{W}_j^e = \frac{l_j^e}{8}\left(1+u_j u\right)\left(1+v_j v\right)\nabla w, \quad \text{for } j = 9,10,11,12$$

where l_j^e is the length of the jth edge and the gradients $\nabla u, \nabla v,$ and ∇w are defined by

$$\nabla u = \frac{\partial u}{\partial x}\hat{u} + \frac{\partial u}{\partial y}\hat{v} + \frac{\partial u}{\partial z}\hat{w}$$

$$\nabla v = \frac{\partial v}{\partial x}\hat{u} + \frac{\partial v}{\partial y}\hat{v} + \frac{\partial v}{\partial z}\hat{w}$$

$$\nabla w = \frac{\partial w}{\partial x}\hat{u} + \frac{\partial w}{\partial y}\hat{v} + \frac{\partial w}{\partial z}\hat{w}$$

Based on new coordinate, equation (14) in manuscript can be expressed as

$$\mathbf{F}_{ij}^e = \int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1}\left[\left(\nabla\times\mathbf{W}_i^e\right)\cdot\left(\nabla\times\mathbf{W}_j^e\right) - i\omega\mu\sigma^e\mathbf{W}_i^e\cdot\mathbf{W}_j^e\right]\left|\mathbf{J}_3\right| du\,dv\,dw,$$

where $\mathbf{J}_3$ is the Jacobian matrix of transformation defined by

$$\mathbf{J}_3 = \begin{bmatrix} \frac{\partial x}{\partial u} & \frac{\partial y}{\partial u} & \frac{\partial z}{\partial u} \\ \frac{\partial x}{\partial v} & \frac{\partial y}{\partial v} & \frac{\partial z}{\partial v} \\ \frac{\partial x}{\partial w} & \frac{\partial y}{\partial w} & \frac{\partial z}{\partial w} \end{bmatrix},$$

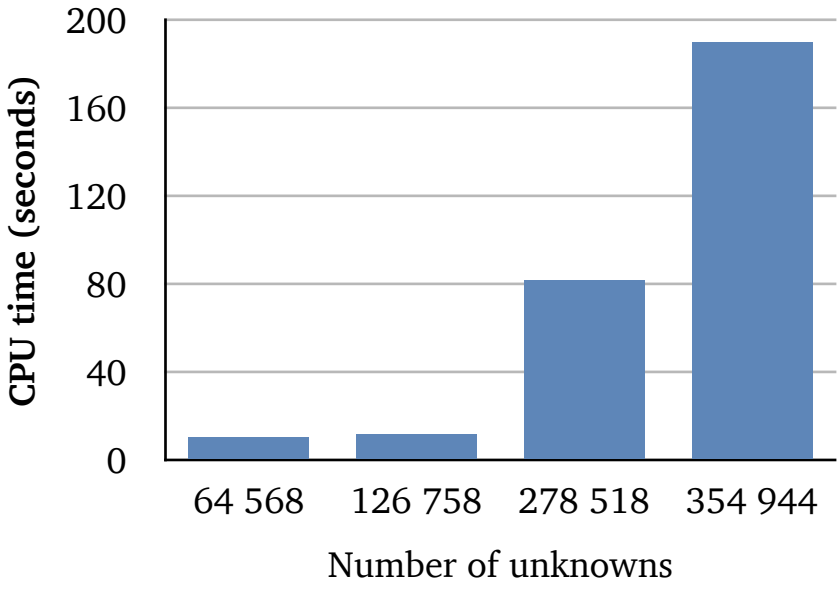
And $\left|\mathbf{J}_3\right|$ is the determinant of $\mathbf{J}_3$. We can see that the formula $\mathbf{F}_{ij}^e$ is more complicated to compute analytically. In contrast, the formula $\mathbf{F}_{ij}^e$ is the standard formula for the Gauss-quadrature rule. The computation relies on the 8-point rule, and the accuracy is acceptable when compared to the analytical value (for the rectangular case). With the high performance of the computing tool. The CPU time for assembling the elemental matrix into the global matrix using the numerical method may be longer than that of the analytical process. However, it seems to have no significant difference when compared to the CPU time used for solving the global system of equations.

Referee's comment-8: Line 165 and thereabouts: It's not 100% clear: Do you use a sparse, direct solver from NumPy (or elsewhere), or an iterative solver? (If an iterative solver, are you doing any kind of divergence correction, or using a specially designed preconditioner, to help with convergence?)

Author's response:

To avoid the instability of the iterative solver, we use a direct solver that is compatible with sparse storage. As we have stated in the manuscript, our coefficient matrix is stored in Compressed Sparse Row (CSR) format. After we submit the manuscript, we

continue to explore and find a better choice for what the direct solver can speed up our algorithm. The PARDISO is one option, but the license is now required. At this time, we found that the Multifrontal Massively Parallel Sparse (MUMPS) Direct Solver (https://mumps-solver.org/index.php , related publications: https://mumps-solver.org/index.php?page=doc) provides a more efficient computation. However, the MUMPS requires a Coordinate (COO) format as input for the sparse matrix (the COO format is represented by three arrays: one for row indices, one for column indices, and one for the corresponding matrix values). Therefore, our coefficient matrix in the CSR format must be transformed into the COO format before implementation. Note that MUMPS is available for both Fortran 95 and Python. Therefore, this context will be revised to clarify it, and our experiment results in the submitted manuscript may need to be updated to account for CPU times and the computing tools used. However, the accuracy of the algorithms remains unchanged. Note that the CPU time for converting from CSR to COO is very short and has no significant impact compared to the main Factorization and substitution process in the MUMPS solver (less than 0.05%). Initial tests reveal the relationship between the number of unknowns in the system of equations and approximate CPU time usage, as shown in the following figures.



Note that the tests are made on MacBook Pro (M1) 14-inch with 16 GB RAM under macOS 15.6.1 (Sequoia)

Referee's comment-9: Figure 5: What does the vertical cross-section at y=0 km look like? That would be interesting to see, including the cells up under the observation locations. Presumably the aspect ratios of those cells are good? Same for Figure 7.

Author's response:

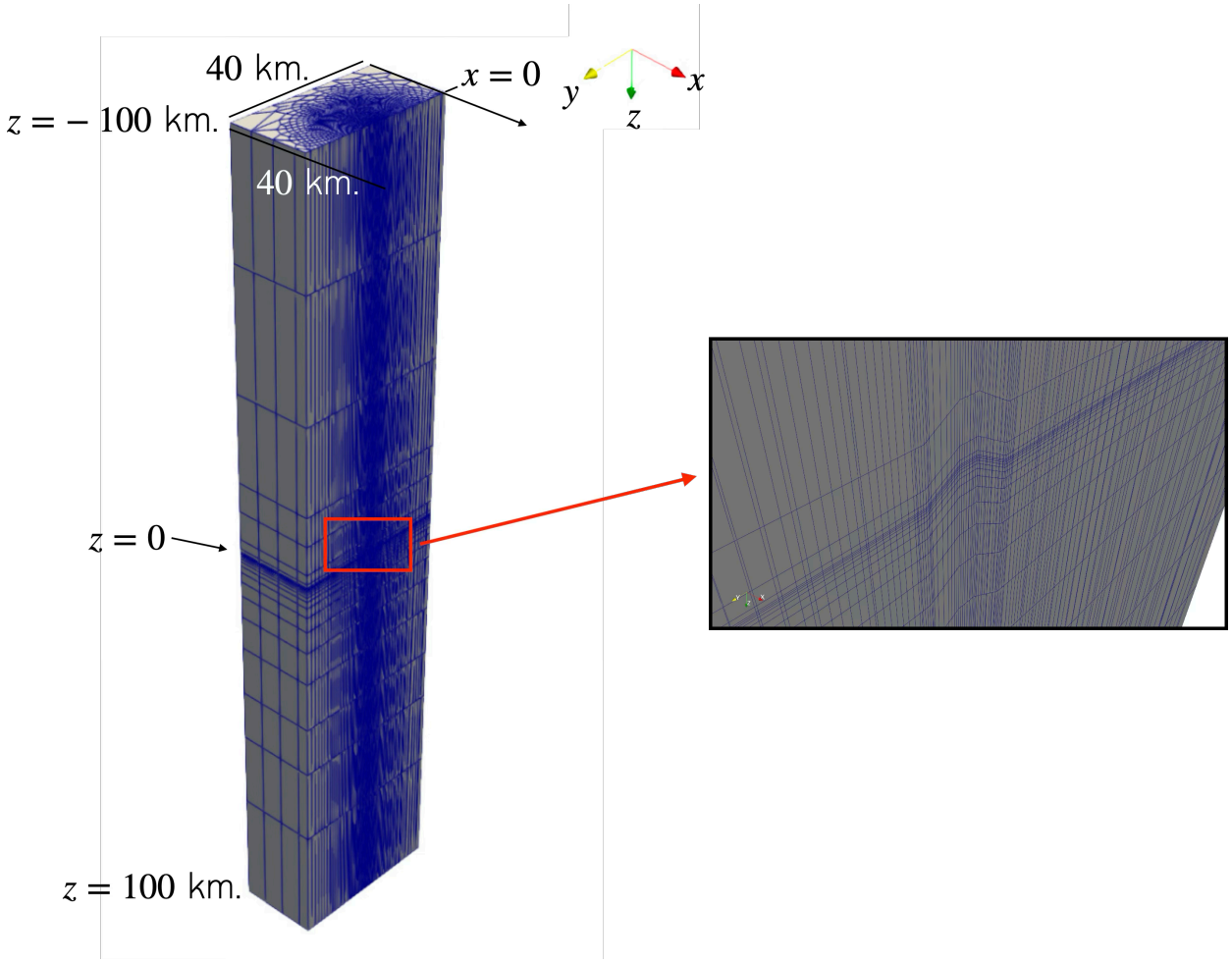
You would like to see the mesh section in the Air layer, from the interface up to the top air boundary at -100 km. The smallest length is the same as the Earth's layer for each model, but only the transition rate is fixed at 3 for all testing. This can be seen in the replying comment-10. The transition rate in the Air is set to a higher rate to reduce the number of elements in the air while maintaining the mesh quality and relaxation. From our

calculations, it may have a less significant effect on accuracy because the skin depth of the EM wave is too high in air. The context of setting the transition rate in the Air will be added in the revised version.

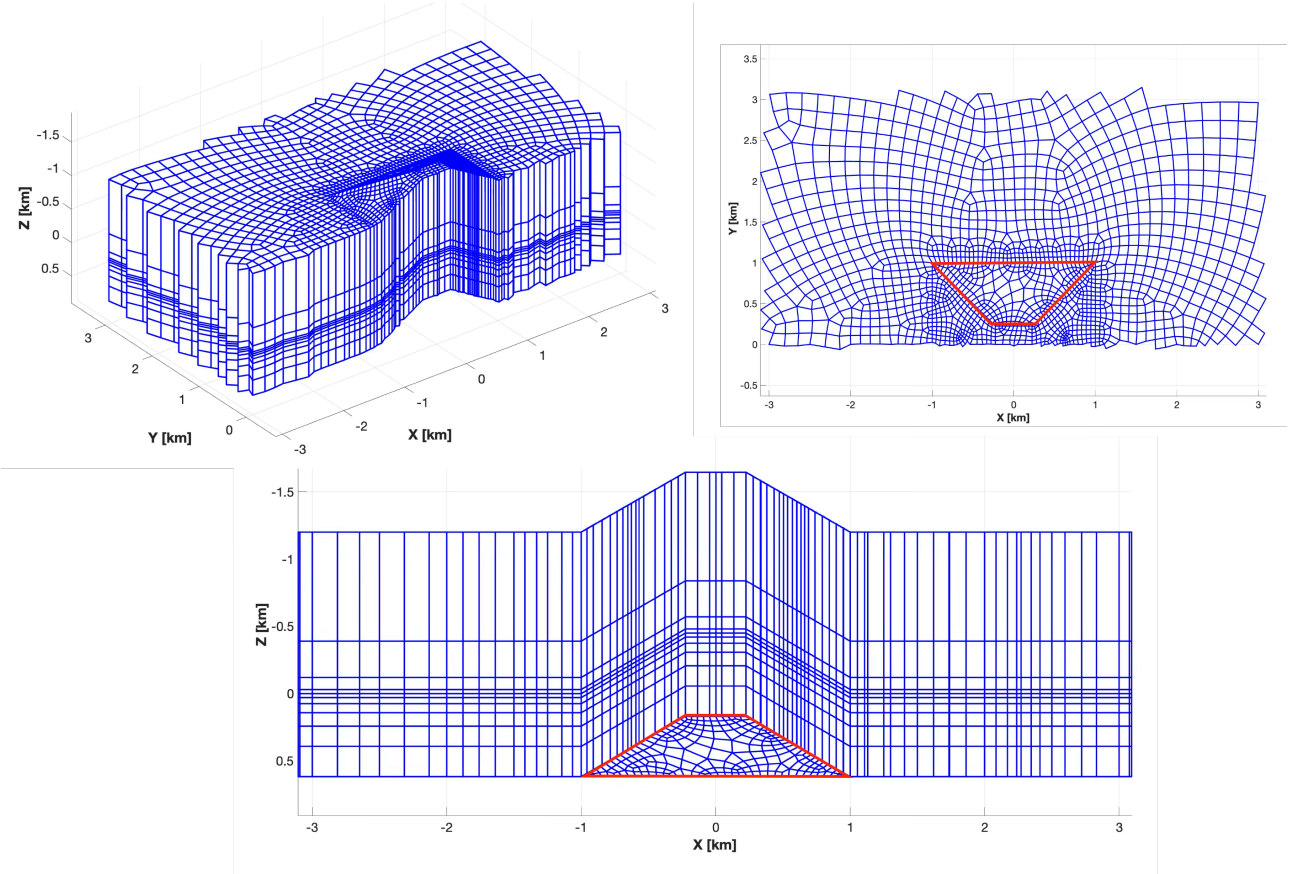
Referee's comment-10: Figure 11: It would be interesting to see a vertical slice through the mesh along the line of observation locations. Do the vertical extents of the cells in a layer vary?

Author's response:

For the previous two cases, the vertical extents of the elements are equal if they are located in the same mesh layer in the z-direction. For this case, the terrain is included. The elements that are outside the topographic zone and located in the same mesh layer have the same vertical extent. In contrast, the elements located inside the topographic zone may have a different extent to handle the topography shape. In the mesh layer far away in the z-direction, the other vertical extents are reduced and become uniform. The concept is illustrated in the following figure.



If we create a filter plot for the elements whose volumes are all located within the specified rectangular subdomain and mesh sections, centered around the topographic zone, the corresponding meshes are shown in the following figures.



The area in the red frame is the projection of the quadrilateral mesh, which is the face of the hexahedral mesh, onto the xy-plane and xz-plane.

Some graphics presented here may be selected for inclusion in the revised manuscript, along with their corresponding explanations. In the real world, with its varied terrain, the Earth's surface is often non-flat in many places. We can adapt this concept to the whole Earth's surface.

Referee's comment-11: Figure 12: I think it would be good to re-do this figure. Perhaps digitize the data points from the figure in Nam et al. (their Figure 12, isn't it?) as this would allow you to plot their data on a new graph that you then have total control over. And don't include the 2D results, which show elevated apparent resistivities in the middle of the survey line for the yx-polarization, contrary to what the 3D results indicate: this is just distracting. Even having the crosses (2D results) on the graph of xy apparent resistivities is distracting and takes away from how close your results are to the 3D results of Nam et al.

Author's response:

Thank you for your suggestions—the new comparison between our results and those of Nam et al. (2007) will improve the quality and validity of Figure 12 in the revised manuscript.

Finally, note that the author would like to make some additional improvements to the revised version to enhance the reader's understanding of the clarification on the efficiency, accuracy, and usefulness of this work.

Assistant Prof.Weerachai Sarakorn, Ph.D.