Authors' Response

We would like to once again express our gratitude to both referees for their valuable suggestions and insightful comments. Their thoughtful review of our work has greatly contributed to the substantial improvement of the manuscript. Referee comments are included in **Black**. Our detailed responses are highlighted in **Blue**. The changes implemented in the revised manuscript are indicated in **Red**.

Response to Referee #1

1. line 21: the 35 degree as threshold value needs some more context, since it is heavily relied upon throughout the paper. What mechanism causes the instability at 35 degrees, and why can some models tolerate steeper gradients?

We thank the reviewer for the comment. We agree that a 35° threshold is too specific for the introduction. In fact, this is an empirical value that depends on several factors. Numerical instabilities can arise regardless of the simulated period or weather conditions, since they are ultimately linked to static terrain features that induce strong flow gradients. These instabilities are related to the model's limited ability to numerically resolve steep orography on a discretized grid, which can produce unrealistically large pressure-gradient and vertical-velocity errors. These effects are further modulated by grid spacing and time-integration settings, which explains why some model configurations can tolerate steeper slopes than others. However, atmospheric conditions can also influence their onset. As explained in the manuscript, slope thresholds of 30° and 35° were required for our mesoscale and microscale simulations to complete successfully, respectively. Nevertheless, this threshold should be adapted for each simulated case, selecting the highest value that ensures numerical stability, thus preserving the terrain as faithfully as possible.

We have modified this paragraph, providing a general approach to the problem and specifying that these are empirical limits: "This issue is of particular concern when terrain slopes exceed **empirical critical values of around 30–40º. In such cases, numerical instabilities can primarily arise from the interaction between steep orography and terrain-following coordinates at finite grid resolution, which amplifies pressure-gradient discretization errors and spurious vertical velocities (e.g., Schär et al., 2002; Klemp et al., 2003). The effective stability also depends on local resolution and time-integration settings (e.g., Courant number), so tolerable slopes may vary across models and configurations."**

2. line 32: what is a 'closest approach'?

We thank the reviewer for this question. We meant “a method that more closely approximates to localized smoothing while remaining global.”

To avoid ambiguity, we have revised the sentence (see response to Specific Comment 2 of Referee #2).

3. line 38: 'thorough' - I do not disagree, but adjectives like this do not belong in scientific text.

We thank the reviewer for the comment. We have removed that word.

That sentence now reads: “The most effective method was selected and implemented, following a ~~**thorough**~~ performance analysis [...]”.

4. line 71: The grid spacing, probably not the resolution, increases with height.

We thank the reviewer for this comment. We have made it clearer in the text.

See implemented changes in the response to Specific Comment 6 of Referee #2.

5. Section 2.1.2: I find the inner and extended FastEddy domains a bit confusing: it seems that only results from the extended domain are presented, so what's the use of the inner domain in this paper? Also, the naming implies that these are two (nested) simulations (like for WRF), which is not the case, if I understand it correctly.

We thank the reviewer for the comment. FastEddy does not use nested dynamical simulations here. We define two *domains* for different purposes: an inner domain (15×15 km) where the LES is actually integrated, and an extended domain (20×20 km) used only at the preprocessing stage (*SimGrid*) to derive the geographical inputs (terrain elevation, land use, roughness length) needed by the inner domain. The smoothing algorithms are applied and compared over the extended domain because it contains more steep-slope points and thus represents a more demanding test for terrain-related instabilities. The simulations themselves are run only in the inner domain. We have revised the text to clarify this separation.

We have included some clarifications in Section 2.1.2: “The FastEddy **simulations are performed over** a 15×15 km **inner (simulation)** domain approximately centered on the city of Murcia (see Fig. 1 (b)), with a horizontal spatial resolution of 10 m ((N_x, N_y) = (1536, 1530)). The vertical extent reaches up to 2700 m, divided into 90 levels using vertical stretching. This setup ensures an approximately isotropic resolution of 10 m near the surface, with the vertical grid spacing gradually increasing with height. Figure 1 (b) also shows the 20×20 km extended **(preprocessing)** FastEddy domain **used only in the *SimGrid* step to generate the geographical fields (terrain elevation, land use, roughness length) for the inner domain.** The different terrain-smoothing methods are applied **and compared** over this extended domain **because it contains a larger fraction of steep-slope points, providing a more demanding test. The LES integration is performed exclusively in the inner domain to assess whether the smoothing yields numerically stable simulations. Nevertheless, any smaller domain extracted from the smoothed extended domain is expected to run without terrain-driven numerical instabilities.** ”

6. Figure 1 caption: Typo (crush instead of crash).

We thank the reviewer for pointing this out. We have corrected the typo.

The last sentence of the caption now reads as follows: “The zooms to the white squares show the vertical component of the wind right before the models **crash**”.

7. line 139: How exactly is the terrain upscaled? Since the paper is about terrain data processing, this is an important detail.

We thank the reviewer for this comment. The upscaling is performed employing the *block_reduce* function of the *skimage.measure* Python module. This function is used to extract the mean value inside each 25x25 m block of the original terrain. This upscaling is applied only to the testing section, for the sake of simplicity. Once we have selected the final method, the smoothing was directly applied to the original resolutions in Section 3.2.

We have included a short explanation of how the upscaling was performed: “For the sake of simplicity, the domain is upscaled to 25 m (Fig. 3) **by block-averaging over non-overlapping 25×25 m tiles**, which significantly reduces the computational cost of the analysis while preserving the validity of the conclusions.”

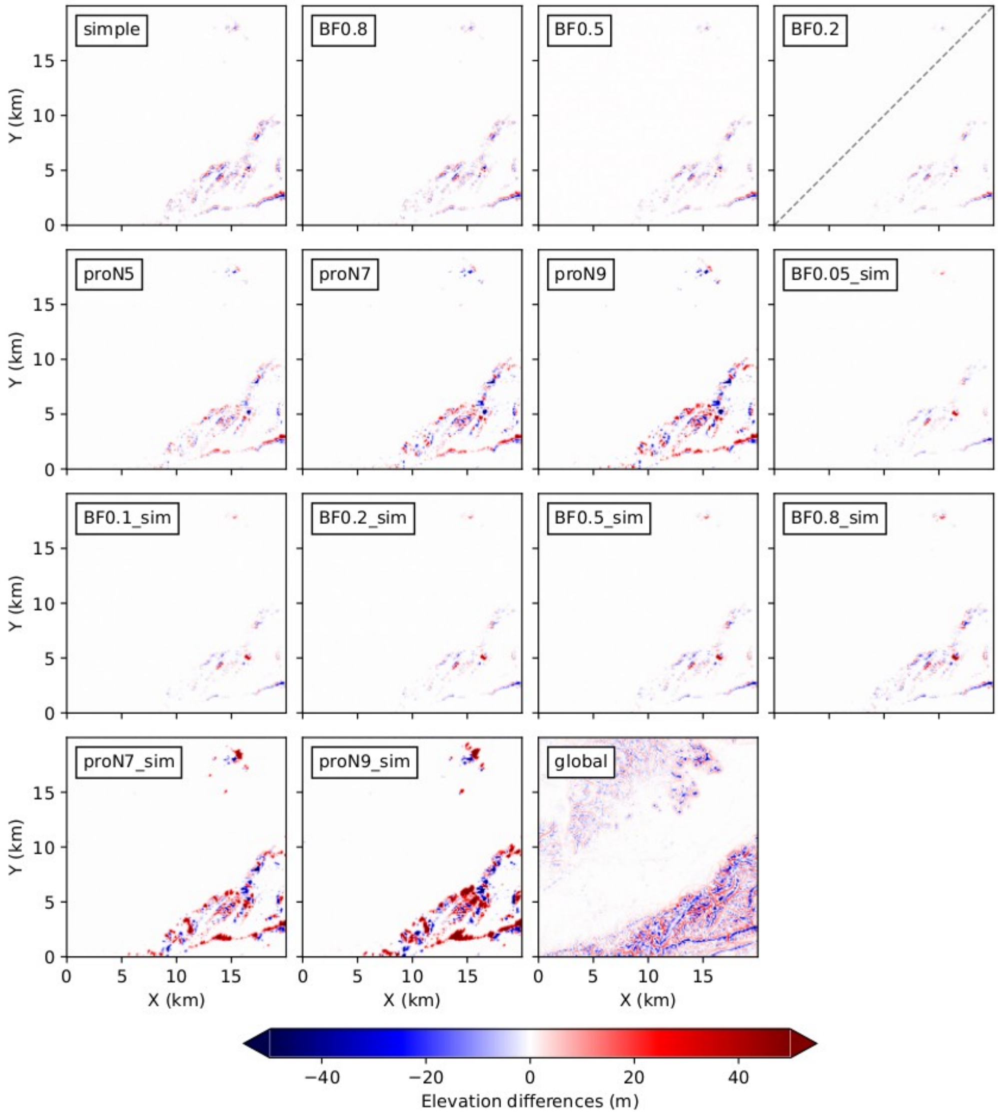
8. line 149: it should be more explicitly stated what convergence entails.

We thank the reviewer for the comment. In this context, “reach convergence” means reaching the point where the smoothed result has no steep-slope points anymore, so the maximum slope falls below the imposed threshold.

We have added a brief clarification of how *convergence* is used at its first occurrence in our analysis (Section 2.3): “The first part of the analysis focuses on a comparison of the 15 smoothing configurations in terms of computational time and number of iterations to convergence **(i.e., maximum slope below the prescribed threshold, no remaining steep-slope points)**, aiming to identify the most time-efficient approaches. Next, […]”

9. Figure 6: What is the purpose of the metric relative elevation differences? A certain elevation difference caused by the smoothing will have a relative elevation difference that depends on the location in the domain. Why not present absolute elevation difference or slope difference?

We thank the reviewer for this question. Our goal was to compare smoothing impacts across areas with very different topographic scales. Because the lowest points in the domain are close to sea level, the main reason behind this decision was to prevent absolute elevation differences (dominated by mountainous regions) from potentially masking changes of similar relative magnitude in lower-lying areas. The relative elevation difference provides a scale-aware, dimensionless measure. We agree that absolute differences and slope differences can be more informative in some cases. We have generated an absolute-difference version of Fig. 6 (see figure below). It highlights mountainous points slightly more strongly (although they are within reasonable relative differences), but the spatial patterns and conclusions remain consistent with those from the relative metric. We therefore keep the original relative-difference version.



10. Figure 7: In the slope density distribution, it is not clear what each color means.

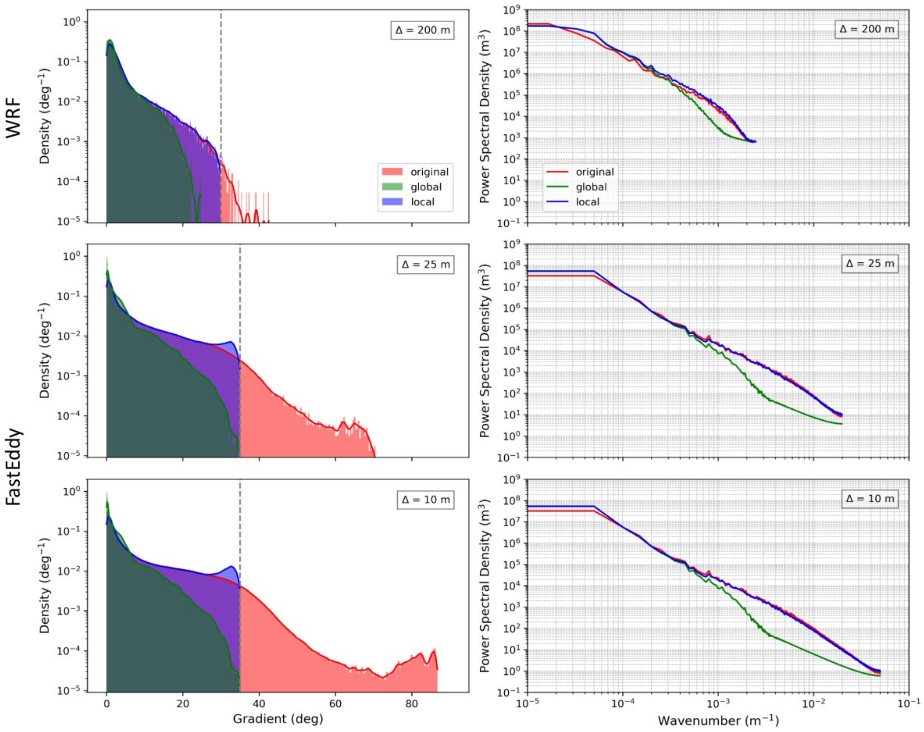
We thank the reviewer for the comment. We have removed excessive transparency in Fig. 7, included KDE lines for more clarity, and added y-axis units (see response to the next question).

11. Figure 7: All three resolutions have the same maximum wavenumber: 2 10^{-2} m^{-1}, which corresponds to 50 m. Since the three panels use different resolutions, it is not clear what is meant with the wavenumber. Also, the y axes lack units.

We thank the reviewer for this comment. Due to a typo, the three spectra were computed with the same grid spacing (Δ = 25 m), which made them share the same maximum wavenumber. We have corrected this, as can be seen in the figure below. Consequently, all three spectra now end according to their Nyquist limit ($k_{Ny} = 1/(2\Delta)$). Therefore, with Δ = 25 m, k_{Ny} = 0.02 m^{-1}, and the spectrum must end there for this resolution.

Regarding the y-axis units, the original spectra were reported in arbitrary units. We have now recomputed them as normalized 1-D power spectral densities *S(k)* with physical units of m^3, such that the integral of *S(k)* equals the variance of elevation (m^2). The revised figure leads to the same conclusions, and only minor wording changes were needed in the accompanying text.

We have modified the text in Section 3.1.3, according to the new figure: “A slight reduction of power is observed between ~~10^{-3}~~ **10^{-4}** and 10^{-2} m^{-1}, corresponding to intermediate**-to-large** terrain features.”



12. line 265: 'the developed local smoothing method ensures numerical stability', this conclusion is a bit too strong (with one case study).

We thank the reviewer for the comment. We have corrected it.

That sentence now reads: “the developed local smoothing method **helps ensure** numerical stability”.

Response to Referee #2

Major comments

1. The Nyquist cutoff limits information represented on a grid to 2*dx. If topography data from a higher resolution dataset is sampled onto a dx resolution grid, there will be aliasing of higher wavenumbers onto the grid. These can create noise that is generally removed by applying terrain smoothing. If the local filtering approach is applied, how is aliasing addressed across the rest of the domain?

We thank the reviewer for this important comment. We agree that aliasing may arise when high-resolution topography is directly sampled onto a coarser grid without appropriate filtering. However, the objective of the present work is not to address aliasing, but to provide a terrain-smoothing tool aimed at mitigating terrain-induced numerical instabilities while preserving local terrain features. In this context, aliasing should be handled during the terrain preprocessing stage where the high-resolution dataset is adapted to the target model grid. Applying global smoothing to remove aliasing would lead to an unnecessary loss of terrain detail, whereas proper preprocessing (e.g., spatial averaging) can act as an appropriate low-pass filter to prevent aliasing.

More importantly, the terrain datasets used in this study were preprocessed to prevent aliasing. In the case of WRF, terrain is not obtained through direct subsampling. Instead, the *geogrid* step was set to apply spatial averaging (with *interp_option* = *average_gcell(4.0)+four_pt+average_4pt* for all domains), which effectively filters out unresolved high-wavenumber components. The local smoothing method is then applied to this preprocessed terrain. For the FastEddy case, the terrain used in this study was treated externally (outside the FastEddy’s preprocessing system) and preprocessed using block averaging over non-overlapping areas (10×10 m or 25×25 m depending on the target resolution). This ensures that the resulting terrain (to which the different smoothing methods are applied), and therefore the results presented in this work, are free of aliasing and consistent with the model resolution.

However, we note that, in its current implementation, FastEddy’s preprocessing (*SimGrid*) can generate terrain through two different strategies: decimation (without prior filtering) or interpolation, which may introduce aliasing. Therefore, the reviewer is correct that aliasing could be present in the terrain passed to the smoothing algorithm under standard settings of this preprocessing step (more specifically when the differences between the simulation's grid spacing and the terrain resolution are significant). As noted above, removing aliasing is not the purpose of the smoothing method, which is designed to act on the already preprocessed terrain to control excessive slopes and improve numerical stability. Nevertheless, following the reviewer's comment, a new preprocessing option based on block averaging (and thus consistent with the approach used in this study) has been implemented in FastEddy to ensure that no aliasing is present in the terrain provided to the local smoothing function. This option is available through the *SimGrid* namelist (*topo_average_opt*) and is included in the upcoming model release (version 5.0, April-May 2026; see the release notes for further details). We thank the reviewer again for raising this important issue.

Additional clarifications have been incorporated into the manuscript to address these points. We emphasize that the terrain used in this study was already preprocessed to account for the resolution disparity and to prevent possible aliasing effects, and therefore the results presented in this work do not require any modification.

We have clarified how the terrain is interpolated onto the model grid in both WRF and FastEddy configurations:

"[…] Therefore, the Shuttle Radar Topography Mission dataset with a spatial resolution of 90 m (SRTM90; Farr et al., 2007) was used instead. **Terrain data are interpolated onto the model grids using WPS spatial averaging options, which effectively filter out unresolved high-wavenumber components and prevent aliasing.** In the high-resolution domain, terrain slopes locally exceed the empirical thresholds recommended for mesoscale modeling. Specifically, […]."

"The terrain elevation data were extracted from the Spanish National 2nd-Coverage Digital Terrain Model (MDT02; 2015–2021; Instituto Geográfico Nacional, 2015), with a native grid spacing of 2 m, obtained from the download center of the National Geography Institute of Spain. **The data were aggregated to the target FastEddy resolution (10 m) using spatial averaging. Under this representation,** problematic points emerge in areas with slopes exceeding 35° (Fig. 1 (b)). Details on land use, surface properties, the selected physical parametrizations, and the coupling with WRF are provided in Appendix B."

Furthermore, we have clarified the role of terrain smoothing in relation to aliasing, explicitly distinguishing between terrain preprocessing prior to smoothing and smoothing:

"[…] Furthermore, the use of a localized method ensures that the majority of the grid points are not being modified in any way, as only the steep-slope points (and their immediate surroundings) are smoothed out.

It is worth noting that global terrain smoothing is sometimes used in modeling workflows as a practical way to mitigate aliasing effects arising from the direct sampling of high-resolution datasets onto coarser grids. The local smoothing approaches proposed in this study are not intended to address aliasing, but rather to control excessive terrain slopes while preserving local features. Therefore, the input terrain is assumed to be already consistent with the target model resolution, with unresolved high-wavenumber components removed during the initial preprocessing stage (e.g., through spatial averaging). In the present work, this condition is ensured by the use of averaging-based interpolation in WRF and block-averaging procedures to preprocess the terrain datasets used in FastEddy.

Two smoothing strategies are considered: sequential, in which only the steepest point is treated at each iteration, and simultaneous, where all problematic points are treated in a single iteration. […]"

In addition, we have clarified in the implementation to the models workflow that, when applying the local smoothing method, the terrain must be preprocessed to ensure consistency with the model resolution prior to smoothing. For the WRF configuration (see response to Specific Comment 20 for full details):

"[…] It is therefore conceived as a complementary preprocessing step in cases where the default *geogrid* smoothing is disabled or is not sufficient to prevent terrain-induced numerical instabilities. **In the former case, appropriate interpolation options (e.g., spatial averaging) should be selected in the *geogrid* step to mitigate aliasing during preprocessing.** […]"

For FastEddy, we have also included a reference to the corresponding preprocessing option in *SimGrid*:

"[…] The algorithm is executed during the preprocessing *SimGrid* step, which performs the generation of the domain grid files including all geophysical variables. **To ensure consistency with the model resolution and to minimize possible aliasing effects, the terrain should be aggregated onto the target grid prior to the application of the smoothing method (e.g., using the block-averaging option available in *SimGrid*).** […]"

2. Explain why global smoothing is a bad thing, given that aliasing is an issue when sampling from a higher resolution dataset. There could be an argument that there are many uncertainties in the topography and land cover datasets, but it seems odd to adjust the grid points only in the steep terrain sections of the domain.

We thank the reviewer for this comment. As noted above, the terrain used in our simulations is obtained through spatial aggregation (i.e., averaging) onto the model grid. This preprocessing step, standard in modeling systems such as WRF and similarly applied here, already mitigates potential aliasing by removing subgrid-scale variability. The local smoothing is applied after this step and is therefore not intended to address aliasing.

Global smoothing is not inherently “incorrect”, but it applies uniform modifications across the entire domain, including regions where the terrain is already well represented and produces stable simulations at the target resolution. This can unnecessarily modify physically relevant features. In contrast, the proposed local approach only modifies grid points where slopes exceed certain thresholds, preserving terrain structure elsewhere while preventing numerical instabilities associated with steep gradients. This selective approach is motivated by the fact that terrain-induced numerical instabilities are typically localized in regions with large slopes, rather than being a domain-wide issue. Therefore, applying modifications only where they are needed minimizes unnecessary alterations to the terrain while ensuring numerical stability. These aspects are already discussed in the manuscript.

3. There is no evaluation of the meteorological forecast differences with global vs local smoothing. Does the local smoothing actually give better simulation results? Why bother with a more expensive local method if the global method works quickly, and the raw topo data needs filtering anyway to remove aliasing? Can the benefits be highlighted more?

We thank the reviewer for this comment. We agree that validation against observations is an important aspect in many modeling studies. However, the primary objective of this work is to present a terrain preprocessing methodology aimed at improving numerical stability and preserving terrain representation in high-resolution simulations, rather than to assess forecast skill against observations.

In this context, the analysis focuses on two key aspects: (i) the ability of the simulations to remain numerically stable under realistic conditions, and (ii) maximizing the degree to which terrain features are preserved under different smoothing strategies. These are essential prerequisites for realistic simulations, particularly at high resolution. As such, numerical stability and consistency must be achieved before any assessment of forecast skill can be considered. Without appropriate terrain treatment, simulations may fail due to CFL-related instabilities, while excessive global smoothing can significantly distort the terrain, effectively undermining the benefits of high-resolution modeling.

The proposed local smoothing approach therefore provides two main advantages: it helps ensure numerical robustness by preventing terrain-induced instabilities, and it preserves terrain features more effectively than global smoothing methods by limiting modifications to steep-slope areas.

From a physical perspective, improved simulations rely on an accurate representation of the modeled domain. Therefore, a more accurate terrain representation is expected to contribute to improved simulation results. However, this is not guaranteed, as model performance depends on multiple factors beyond topography, including physical parameterizations and initial and boundary conditions. High-resolution details and numerical stability and consistency do not necessarily translate into improved forecast skill, but they are prerequisites for physically meaningful simulations. A comprehensive evaluation against observations would therefore require a dedicated experimental design and is beyond the scope of the present study. In addition, observational datasets suitable for a detailed evaluation of the FastEddy simulations are not available for the selected case.

While the local approach introduces a slightly higher computational cost compared to global smoothing, this difference remains small in practice, and the smoothing is performed only once during preprocessing. Furthermore, the terrain used at the smoothing stage has already been made consistent with the model resolution (e.g., through spatial averaging procedures, which are commonly used in workflows such as WRF), mitigating potential aliasing effects, as discussed above. The benefits in terms of improved terrain representation and numerical robustness therefore justify this minor additional cost.

A brief discussion of these aspects has been added at the end of the Results and discussion section.

“Although the global smoothing method also ensures numerical stability in both models, it substantially compromises fine-scale terrain features. These high-resolution details are essential for accurately representing local processes, particularly in simulations aimed at capturing small-scale atmospheric dynamics. **Without appropriate terrain treatment, simulations may fail due to terrain-driven instabilities, while excessive global smoothing can significantly distort the topography, effectively undermining the benefits of high-resolution modeling.**

From a physical perspective, the realism of a simulation relies on an accurate representation of the modeled domain. Therefore, a more realistic terrain depiction is expected to contribute to improved simulation results. However, a detailed evaluation of this effect in terms of forecast skill is beyond the scope of the present study.

The local smoothing method offers improved preservation of the original fine-scale topography, allowing for more realistic terrain representation without sacrificing numerical stability. **It introduces a slightly higher computational cost compared to global smoothing, but this difference remains small in practice. The benefits in terms of improved terrain representation and numerical robustness justify this minor additional cost. Furthermore**, its local nature makes it suitable to also remove noise or other artifacts from terrain datasets that would similarly lead to numerical instabilities.”

Specific comments

1. Lack of meteorological validation of the smoothed-terrain simulations: the paper demonstrates the locally smoothed terrain allows previously failing simulations to complete without CFL errors with real cases. With that said the paper shows no comparison of any comparison with observation. Even a qualitative comparison (10m-Wind, T2, vertical wind components…) would substantially strengthen the paper.

We thank the reviewer for this comment. These aspects are addressed in our response to Major Comment 3. As discussed above, such an analysis is beyond the scope of the present study, which focuses on terrain preprocessing, numerical stability, and terrain representation.

2. Line 32 - "introduced a closest approach to localized" Since the idea of localized smoothing has not yet been introduced, perhaps rephrase this.

We thank the reviewer for this comment. We have rephrased the sentence to avoid introducing this concept before it is formally defined. In addition, we have included a definition of localized smoothing in the following paragraph.

"Sheridan et al. (2023) **proposed a targeted smoothing method that partially addresses the limitations of uniformly applied global approaches."**

"This study presents the development and implementation of a local terrain smoothing approach designed to mitigate numerical instabilities in a mesoscale model (WRF) and a microscale LES model (FastEddy®; hereafter referred to as simply as FastEddy), and that is easily adaptable to other models. **In this context, "local" refers to a smoothing strategy that selectively modifies only the grid points where terrain slopes exceed predefined thresholds associated with numerical stability, while leaving the rest of the domain unchanged.** Various smoothing techniques were evaluated, […]."

3. Lines 35-45 - The last paragraph with the 'table of contents' could be merged with the previous paragraph to make it seem less repetitive.

We thank the reviewer for this suggestion. The paragraph describing the structure of the paper has been condensed and integrated into the end of the previous paragraph to avoid repetition.

"[…] The most effective method was selected and implemented, following a performance analysis, considering the number of iterations required for convergence, computational cost, and, most importantly, the degree of terrain distortion. **This methodology is described in Sect. 2. Section 3 presents the results, including the application of the selected method to a case study where the simulation failed due to CFL-related**

instabilities in both WRF and FastEddy. Finally, Sect. 4 summarizes the main conclusions.”

4. Mention which namelist parameters specify the typical terrain smoothing in WRF and FastEddy.

We thank the reviewer for this comment. In FastEddy, terrain smoothing is applied internally as part of the terrain preprocessing and is not exposed as a user-configurable namelist option. The standard global smoothing is embedded in the preprocessing workflow. The local smoothing method developed in this study replaces the global approach within the code.

In WRF, typical terrain smoothing is applied during preprocessing in the *geogrid* step and is controlled through the *GEOGRID.TBL* settings for the terrain field (*HGT_M*). The default smoothing is defined via *smooth_option = smth_desmth_special* with a single iteration (*smooth_passes = 1*). This method consists of a smoothing step followed by a desmoothing step, both using a 1-2-1 approach. Note that this is a global method applied uniformly across the domain.

The proposed local smoothing method is applied to the terrain after the *geogrid* preprocessing stage. It is conceived as a complementary WPS step in cases where the default *geogrid* smoothing is disabled or is not sufficient to prevent terrain-induced numerical instabilities, and further iterations of the global smoothing would lead to an excessive distortion of the terrain. The tool developed in this work provides flexibility, allowing users to apply the local smoothing approach according to their specific needs and modeling configurations.

Clarifications regarding the typical terrain smoothing approaches in both models have been incorporated into the revised manuscript.

“Once the optimal smoothing method was selected, it was implemented in both the WRF and FastEddy modeling systems. For WRF, the algorithm was integrated as an additional step within the WPS. **The default terrain smoothing is controlled through the *GEOGRID.TBL* file through the parameters *smooth_option* and *smooth_passes*, which define a global smoothing applied uniformly across the domain.**”

“**In FastEddy, terrain smoothing is handled as part of the preprocessing and is not exposed as a user-configurable namelist option.** The selected local smoothing method has been fully integrated into the model source code, replacing the previously used global smoothing approach. Version 3.0 of the model (National Center for Atmospheric Research (NCAR), 2025) already includes this improvement.“

5. Line 55-56: The vertical grid mentions 45 levels with increased resolution near the surface with no further detail. What is the lowest model level height? Further details on vertical levels seem to be relevant information because the slope angle that causes instability depends on the grid aspect ratio. The stretching factor and other grid parameters should be given.

We thank the reviewer for this comment. We agree that the vertical grid configuration is relevant, as the sensitivity to slope-induced numerical instability depends on the grid aspect ratio.

In this study, the vertical grid in WRF is defined explicitly through 45 prescribed eta levels, rather than by using a stretching factor. The full list of eta levels is provided in the namelist configuration (eta_levels = 1.00000, 0.99629, 0.99257, 0.98879, 0.98486, 0.98071, 0.97622, 0.97130, 0.96585, 0.95977, 0.95299, 0.94540, 0.93692, 0.92744, 0.91686, 0.90507, 0.89195, 0.87737, 0.86120, 0.84331, 0.82356, 0.80181, 0.77793, 0.75181, 0.72335, 0.69246, 0.65911, 0.62329, 0.58506, 0.54455, 0.50195, 0.45755, 0.41175, 0.36503, 0.31802, 0.27144, 0.22617, 0.18317, 0.14344, 0.10788, 0.07710, 0.05132, 0.03028, 0.01343, 0.00000). This results in enhanced resolution near the surface, with a progressively increasing vertical spacing with height. The lowest model level is located at approximately 15 m above ground level (domain mean). The subsequent levels are located at approximately 46 m, 78 m, 110 m, 144 m, and 180 m, with a vertical spacing that gradually increases with height. We have included these details in the revised manuscript to better document the vertical grid configuration.

"The vertical grid includes 45 **eta** levels **explicitly defined in the model configuration,** with enhanced resolution near the surface, and the model top set at 50 hPa. **The lowest model level is located at approximately 15 m above ground level (domain mean). Subsequent levels are located at approximately 46 m, 78 m, 110 m, 144 m, and 180 m, with a vertical spacing that gradually increases with height.**"

6. Line 68-70: Same applies with FastEddy domain using 80 vertical levels with stretching to achieve ~10m resolution near the surface. What's the near-surface vertical spacing exactly? Any detailed information needed to assess the effective resolution and the relevance of the 35° slope threshold.

We thank the reviewer for this comment. In FastEddy, the vertical grid is defined using a uniform spacing (d_zeta = 23 m) together with a vertical deformation (verticalDeformSwitch = 1). The deformation is controlled by verticalDeformFactor (set to 0.264 in this study, with verticalDeformQuadCoeff = 0.0), which compresses the grid near the surface and stretches it aloft, providing enhanced resolution near the surface. The lowest model level is located at approximately 6 m above ground level. The vertical spacing near the surface is approximately 12 m and increases gradually with height. We have included these details in the revised manuscript.

“The vertical extent reaches up to 2700 m [...]. **The vertical grid is defined using a uniform spacing of 23 m together with vertical deformation (with a vertical deformation factor of 0.264), enhancing the resolution near the surface. The lowest model level is located at approximately 6 m above ground level, with a vertical spacing of approximately 12 m near the surface that gradually increases with height.**”

7. Figure 1 caption - 'model crush' → crash

We thank the reviewer for this comment. The typo has been corrected.

8. Line 69 - 2700 m is quite a low model top for complex terrain!

We thank the reviewer for this comment. In FastEddy, the model top is fixed at 2700 m above sea level. The maximum terrain elevation within the domain is 488.7 m, resulting in an effective vertical extent of more than 2 km above ground level across the domain. Given that FastEddy is used as a microscale LES model focused on resolving boundary-layer processes, this vertical extent is sufficient to capture the relevant flow structures while maintaining a reasonable computational cost. We acknowledge the reviewer’s concern and have clarified this point in the revised manuscript.

“The vertical extent reaches up to 2700 m **above sea level and is** divided into 90 levels using vertical stretching**, corresponding to more than 2 km above ground level across the domain. This configuration is appropriate for a microscale LES setup focused on resolving boundary-layer processes for the stability conditions considered herein**.”

9. Line 72 - explain why you need an extended domain?

We thank the reviewer for this comment. The role of the extended domain has been clarified in the revised manuscript following the recommendations of both reviewers. The extended domain is used exclusively during the preprocessing stage to generate the geophysical input fields (terrain elevation, land use, roughness length) required by the inner simulation domain. The smoothing methods are applied and evaluated over this larger domain because it contains a greater number of steep-slope points, providing a more demanding test. The LES simulations themselves are performed only over the inner domain. Any smaller domain extracted from the smoothed extended domain is expected to run without terrain-driven numerical instabilities.

A brief clarification has been included in the revised manuscript (see response to Comment 5 of Referee #1).

10. Line 75 - 'once resampled' --> the 2m resolution topography should be smoothed to 10 m -- it shouldn't be resampled because that retains wavelengths smaller than the resolution of the grid.

We thank the reviewer for this comment. As previously stated, we agree that directly sampling high-resolution topography onto a coarser grid without appropriate filtering would retain unresolved small-scale variability and potentially introduce aliasing. In our workflow, however, the terrain is not obtained through direct subsampling. Instead, the high-resolution topography is mapped onto the model grid using spatial aggregation (i.e., averaging), which acts as an implicit low-pass filter and removes subgrid-scale variability, ensuring that wavelengths smaller than the model resolution are not retained.

The manuscript has been revised to clarify that the terrain is generated through spatial averaging rather than direct subsampling, as discussed in the response to Major Comment 1.

11. Line 98 - "the use of a localized method ensures that the majority of the grid points are not being modified in any way, as only the steep-slope points (and their immediate surroundings) are smoothed out" - again this means you have aliasing noise.

We thank the reviewer for this comment. The issue of potential aliasing and its relation to the smoothing strategy is addressed in our response to Major Comment 1. As discussed there, aliasing is handled at the preprocessing stage, as the terrains used in this study are generated through spatial aggregation to ensure consistency with the model resolution and to avoid aliasing effects. The localized smoothing method is therefore not intended to remove high-wavenumber variability, but to reduce excessive slopes that may lead to terrain-driven numerical instabilities while preserving terrain features elsewhere. A brief clarification has been added to the paragraph in question to reflect this point (see the response to Major Comment 1).

12. Line 101 - "the maximum number of iterations is set to 1.5 times the number of grid points with steep slopes." Why?

We thank the reviewer for this comment. This limit was imposed to avoid potentially unbounded loops in cases where a given smoothing strategy needed an excessive number of iterations to converge (i.e., maximum slope falls below the imposed threshold). This upper bound was defined as 1.5 times the number of grid points initially identified as problematic in order to scale the stopping criterion with the number of steep-slope points. This choice is particularly relevant for sequential methods (treating only one steep-slope 3×3 area per iteration), for which convergence may require a number of iterations comparable to the number of steep-slope points. The selected factor was intended as a practical limit, large enough to allow convergence when feasible while preventing excessive computational cost. In practice, the iterative process is stopped as soon as it converges, so this upper limit is not necessarily reached. Furthermore, it can be adjusted by the user if needed; however, for the selected method, convergence is achieved well before reaching this limit. We have included a brief clarification in the revised manuscript.

"In all cases, the maximum number of iterations is set to 1.5 times the number of grid points with steep slopes. **This upper limit is introduced to prevent potentially unbounded iterations. The iterative process stops once the maximum slope falls below the prescribed threshold, so this upper limit is in the majority of instances not reached.**"

13. Line 139 - Why not just start with the 25 m resolution domain and put that in Figure 1 instead? It seems unnecessary to add it here and add an extra figure (Fig 3) for this.

We thank the reviewer for this comment. The use of the 25 m resolution domain serves a different purpose from that of the 10 m domain. The coarser resolution is used during the development and testing phase of the smoothing methods, as it allows for a significantly faster evaluation, particularly for methods that may not converge within the maximum number of allowed iterations.

The 10 m resolution domain, shown in Fig. 1, represents the target configuration for which the smoothing approach is ultimately intended. Using both resolutions therefore allows us to efficiently explore and assess different methods while ensuring that the final evaluation is performed at the resolution of interest.

In addition, the comparison between the two resolutions highlights how the number of steep-slope points increases with resolution within the same domain, reinforcing the motivation for the proposed approach.

We have clarified this aspect in the revised manuscript.

"[...] For the sake of simplicity, the domain is upscaled to 25 m (Fig. 3) by block-averaging over non-overlapping 25×25 m tiles, which significantly reduces the computational cost of the analysis while preserving the validity of the conclusions. **The 25 m resolution domain is used only for method development and evaluation, whereas the FastEddy and WRF domains at their target resolutions (Fig. 1) represent the configurations for which the smoothing approach is ultimately intended and where the simulations are carried out.** Once the most suitable method is identified, it is applied to the full-resolution FastEddy domain (Δ = 10 m) and the fourth WRF domain (Δ = 200 m). [...]"

14. Line 166 - "The global smoothing method is by far the fastest, followed by the simultaneous methods" and Line 170 - "Based on computational time alone, any of the simultaneous methods would be suitable candidates for selection." But 859 seconds is pretty long if you can do it in 0.39 s.

We thank the reviewer for this comment. We note that the value of 859 s corresponds to the simple sequential method, not to the simultaneous methods. As indicated in the manuscript, the computational times for the simultaneous approaches range from approximately 1.5 to 51 s, with the selected method requiring about 2 s. While this is higher than the global smoothing method (0.39 s), the difference remains negligible in practice, especially considering that the smoothing procedure is performed only once during preprocessing. The selected method provides a substantially improved representation of the terrain while maintaining a very low computational cost relative to the overall simulation time.

15. Can you prove that local smoothing gives you better simulation results? Why bother with a more expensive local method if the global method works quickly, and the raw topo data needs filtering anyway to remove aliasing?

We thank the reviewer for this comment. This point is addressed in detail in our response to Major Comment 3.

16. Figure 4 - Why does the max slope increase with # of iterations? (bottom right panel)

We thank the reviewer for this question. The increase in maximum slope observed in the bottom panels (both left and right) is a consequence of the local nature of the smoothing procedure. When a 3×3 area is modified, the elevation differences between the edge of the smoothed area and adjacent untreated grid points may temporarily increase, leading to a local increase in slope. This effect can occur especially in regions where steep slopes extend over areas larger than the smoothing window. In such cases, the maximum slope may increase during an iteration but is subsequently reduced in the following iterations as the affected points are progressively smoothed. We have clarified this aspect in the paragraph related to Fig. 4.

"Figure 4 shows [...]. **A temporary increase in maximum slope can be observed in some cases during the smoothing procedure, as modifications to a localized area may increase elevation differences with adjacent untreated grid points. This increase is subsequently reduced in the following iterations as the affected points are progressively smoothed.**"

17. Figure 7 - Explain why there is an increase in light blue near the cutoff.

We thank the reviewer for this comment. We note that Fig. 7 contains two groups of panels, and it is not entirely clear whether the comment refers to the slope distributions or to the spectral representations. We therefore address both interpretations.

If the comment refers to the spectral representations, the increase in light blue near the cutoff corresponds to a slight increase in power at high wavenumbers. As discussed in the manuscript (line 190 of the original version), this effect is attributed to small-scale variability introduced during the blending process associated with the local smoothing.

If the comment refers to the slope distributions, the increase near the threshold is a direct consequence of the smoothing procedure. Grid points with slopes initially above the threshold are reduced to values usually just below it after smoothing, leading to an accumulation of values in that range. In contrast, global smoothing tends to redistribute slopes toward lower gradients, rather than concentrating values near the threshold. We have included a clarification of this observed behavior in the revised manuscript.

“The local method effectively limits slopes to the defined threshold, while the global method produces a smoother terrain, yet some maximum slopes remain close to the threshold value. **Local smoothing leads to an increase in the frequency of slopes just below the threshold, as points originally exceeding it tend to cluster near this limit when smoothed. In contrast, global smoothing redistributes slopes toward lower gradients.** Additionally, […].”

18. Line 190 "increase in power is observed for wavenumbers greater than 10^{-2} m^{-1}, likely reflecting small-scale noise introduced during the blending process" - this extra noise could be introducing error into the simulation as well. Why is this noise ok to include but the error from global smoothing is not ok?

We thank the reviewer for this comment. We agree that some small-scale noise can be introduced during the blending process (especially for high-resolution cases), reflected as an increase in power at high wavenumbers. However, this effect must be considered in the context of the overall spectral representation of the terrain.

As shown in Fig. 7, the locally smoothed terrain retains a spectral distribution that is much closer to the original topography across a wide range of wavenumbers, whereas the global smoothing method significantly reduces power over a broad portion of the spectrum. In this sense, the local approach preserves the relevant terrain variability more effectively, despite the presence of a slight increase in energy at high wavenumbers.

This aspect is acknowledged in the manuscript (end of the paragraph), where it is noted that the local method can introduce minor, localized distortions, while remaining significantly more faithful to the original terrain than the global smoothing approach.

19. Figure 8 - this is presented as though the original terrain is the correct one. But this is higher resolution data sampled on the coarser domain and thus contains incorrect frequencies/values.

We thank the reviewer for this comment. We clarify that the “original” terrain shown in Fig. 8 does not correspond to the raw high-resolution dataset directly sampled onto the model grid. Instead, it has been preprocessed through spatial averaging (over non overlapping square blocks) to match the target model resolution, thereby removing subgrid-scale variability and avoiding the inclusion of unresolved frequencies (aliasing). In this context, the “original” terrain represents the best available approximation of the topography at the working resolution and is used solely as a reference to assess the deformation introduced by the different smoothing methods. To clarify this point, we have revised the caption of Fig. 8 to explicitly state that the “original” terrain is spatially aggregated to the model resolution.

“Figure 8. Comparison of original and smoothed topographies across the three modeling domains. Each row represents a different domain. From top to bottom: the fourth WRF domain (Δ = 200 m), the upscaled extended FastEddy domain (Δ = 25 m), and the extended FastEddy domain (Δ = 10 m). Columns show, from left to right: the original topography **(spatially aggregated to the corresponding model resolution)**, the terrain after global smoothing, the terrain after local smoothing, and the percentage elevation differences between the locally smoothed and the original topographies.”

20. Line 199 - "The subsequent steps of the WPS workflow remain unchanged" - doesn't this mean that WPS applies its own smoothing on top of this?

We thank the reviewer for this comment. In the WPS workflow, terrain smoothing can be applied during the *geogrid* step (typically a single iteration of a smooth-desmooth 1-2-1 filter). The local smoothing method proposed in this study is intended to be applied to the terrain after the *geogrid* step, i.e., to the already preprocessed topography (whether it is already smoothed or not).

This approach serves as a complementary preprocessing step in cases where the default smoothing is not sufficient to prevent terrain-induced numerical instabilities. Applying additional global smoothing iterations would further modify the entire domain without specifically targeting problematic regions, potentially degrading terrain features unnecessarily. In contrast, the proposed local method selectively adjusts only those areas where excessive slopes occur.

Alternatively, the method can also be applied when the *geogrid* smoothing is disabled. In such cases, appropriate interpolation options (e.g., spatial averaging via *average_gcell*) should be used in the *geogrid* step to mitigate aliasing during preprocessing.

The statement that “the subsequent steps of the WPS workflow remain unchanged” refers specifically to the following stages (*ungrib* and *metgrid*), which are executed as usual and do not introduce any additional modifications to the terrain fields. We have included a brief clarification in the revised manuscript.

"For WRF, the algorithm was integrated as an additional step within the WPS. The default terrain smoothing is controlled through the *GEOGRID.TBL* file through the parameters *smooth_option* and *smooth_passes*, which define a global smoothing applied uniformly across the domain. **The local smoothing method developed in this study is applied to the terrain after the *geogrid* preprocessing, i.e., to the already interpolated topography, whether or not the *geogrid* smoothing has been applied. It is therefore conceived as a complementary preprocessing step in cases where the default *geogrid* smoothing is disabled or is not sufficient to prevent terrain-induced numerical instabilities. In the former case, appropriate interpolation options (e.g., spatial averaging) should be selected in the *geogrid* step to mitigate aliasing during preprocessing. T**he smoothing algorithm produce**s** a new **geographical** file with locally smoothed topography, with all other fields preserved. The subsequent steps of the WPS workflow **(*ungrib* and *metgrid*)** remain unchanged."

21. Line 216 - "(epssm ≥ 0.9)." This is really high! There are other ways to make the simulation run, such as coarsening dz, because the ratio of dz/dx and terrain slope is what affects stability.

We thank the reviewer for this comment. The WRF documentation recommends keeping *epssm* values below 0.5 (see WRF Users Guide), as larger values introduce excessive numerical damping. The values required here (*epssm* ≥ 0.9) are therefore well beyond commonly recommended settings. We agree that such high values of *epssm* are not desirable, and this is precisely why this approach is not considered suitable in this case, as already discussed in the manuscript.

We also agree that modifying the vertical resolution (increasing Δz) is a valid alternative to improve numerical stability. However, in this study we aim to preserve the chosen vertical resolution near the surface, rather than reducing it to compensate for terrain-induced instabilities. In the WRF configuration considered here, the lowest model level is located at about 15 m above ground level and the near-surface vertical spacing is about 30 m, so further coarsening of the vertical grid would reduce the intended resolution of the lower atmosphere. Similarly, in FastEddy, maintaining high resolution near the surface is essential for properly resolving boundary-layer processes.

For these reasons, coarsening the vertical grid is not considered here, and the focus is instead placed on addressing steep slopes to ensure numerical stability while preserving the desired spatial resolution. A brief clarification has been added to the revised manuscript to reflect this point.

“Without any terrain smoothing, the simulation could run successfully by applying high values of the time off-centering parameter for vertically propagating sound waves (*epssm* ≥ 0.9). However, as previously discussed in the Introduction, increasing this parameter may result in unphysical behavior and is therefore not recommended (Arnold et al., 2012). **Alternatively, increasing the vertical grid spacing could also improve stability, but this would reduce the intended near-surface resolution and is therefore not considered in this study.** High-resolution simulations aim to maximize accuracy, making such adjustments undesirable.”

22. Line 217 - "may result in unphysical behavior" Why unphysical? It's just reducing the order of accuracy of the time advancement for the vertical implicit scheme from 2nd to 1st order?

We thank the reviewer for this comment. We agree that increasing *epssm* effectively reduces the order of accuracy of the time advancement for the vertical implicit scheme, making it more dissipative. However, the concern is not only numerical accuracy, but also the physical implications of this additional damping. As discussed in the WRF documentation and in previous studies, increasing *epssm* acts as a form of artificial damping of vertically propagating motions (i.e., damping 3-D divergence), particularly affecting the vertical velocity component. This can suppress physically meaningful dynamics rather than resolving the underlying cause of the instability.

In this sense, the resulting behavior is considered unphysical, as also noted by Arnold et al. (2012):

“Practical workarounds to cope with numerical stability problems include the use of an adaptive time-step (getting shorter if the CFL number is detected to exceed a threshold value) or the use of w-damping, i.e., locally reducing the vertical velocity component (which is often the most critical in high-resolution integrations over complex terrain) if instability in the vertical advection is detected. The latter solution is however unphysical and therefore not recommended.”

This reference is already included in the manuscript.

23. Table B1 and B2 - this information doesn't seem relevant to this paper unless you add actual simulation results to this paper. (Typo: Table B2: Deciduoud broadleaf forest -> Deciduous broadleaf forest)

We thank the reviewer for this comment. We agree that this information is not central to the main findings of the manuscript, which is why it is included in the appendix rather than in the main text. However, these tables document the modifications introduced with respect to the default model configuration and are provided to ensure transparency and reproducibility of the experiments. In particular, the simulations presented in the manuscript (i.e., those that initially fail due to terrain-induced instabilities and successfully run after applying the proposed method) depend on these configuration choices. Documenting them is therefore important to allow reproducibility of the numerical behavior discussed in this work.

The typo in Table B2 has been corrected. We thank the reviewer for pointing this out.