Response to Reviewer 1

We are incredibly grateful for your efficient review process, providing your feedback in just a few days. Your insightful comments have provided valuable guidance for revising the manuscript. We have revised the manuscript according to your suggestions and will respond to your comments paragraph by paragraph. The comments are given below in black, our responses are in blue, and proposed changes to the manuscript are in red. Additional references are provided at the end of this document. The final revisions and specific locations corresponding to the manuscript will be marked uniformly after receiving feedback from other reviewers.

Major comments:

Based on Figure 1 and Figure 2, this computational method appears somewhat simple. There is relatively little research on the keyword "quasi double-precision." The Authors could explain why their work is novel compared to the existed methods already published nowadays.

Thank you for your comments. We have added the significance and novelty of our work in section: introduction and section: Conclusions and discussion and have been summarized below.

Introduction

Most works involving numerical models that reduce numerical precision adopt a mixed-precision scheme, where some variables use single precision while others remain in double precision to ensure integration stability, as demonstrated in the work of Chen et al. (2024). Currently, there are very few studies that almost entirely employ low precision (32-bit) in numerical models, only applied in IFS by Váňa et al. (2016). However, they only utilize single precision without considering error compensation for it. In this study, all variables in the numerical model were implemented using single precision, and error compensation was applied to key variables. By using error compensation methods (quasi double-precision), we can maintain integration stability comparable to that applying double precision scheme while significantly reducing memory requirements by lowering the numerical precision of all variables and improved the accuracy comparable to that applying the single precision. This approach not only reduces communication pressure but also allows for substantial increases in computational speed through vectorization optimization.

Conclusion and discussion

We have replaced the first and second paragraphs of the Conclusion section with the following revised paragraph:

The quasi double-precision algorithm can compensate for round-off errors by keeping corrections in addition of large and small numbers. And in numerical models, the basic field is generally much larger than the tends, which aligns with the principles of quasi double-precision, as well as the time integration process. Based on the it, we have established a strategy for applying the quasi double-precision algorithm within the MPAS-A. Through the implementation of quasi double-precision methods, we maintain accuracy similarly to the tests using double precision and achieve comparable integration stability to the tests comparing to single precision tests. The error of surface pressure of

4 cases are reduced by 68%, 75%, 97%, 96% (see Section 3). Overall, QDP using quasi double-precision algorithm demonstrates higher accuracy than the SGL, suggesting the potential for applying quasi double-precision algorithm in numerical models.

We don’t apply it to spatial discretization process, because spatial discretization primarily involves subtraction, specifically the subtraction of a small number from a large number or the subtraction of two close values. Whether this algorithm is applicable in spatial discretization remains uncertain, therefore, we don’t consider it in this context.

While mixed-precision approaches, where certain variables retain double precision for stability (e.g., Chen et al., 2024), are common for reducing numerical precision in models, and they don’t consider the error compensation. This study distinguishes itself by implementing single precision for all model variables and applying error compensation for critical variables.

When applied the quasi double-precision algorithm in MPAS-A, we achieved to reduce all double precision to single precision, although increased few local variables and arrays in every time-integration variable, these have little impact on the overall memory reduction. In general, memory has been reduced by almost half, with a corresponding computational increase of just 6%, 0.3%, 2%, and 18% in the respective cases (see section 3.4), demonstrating a substantial improvement in computational efficiency.

The main point of applying signal precision computing methods is ensuring predictive, and reducing computational costs. The iterative precision compensation increases the computation load. Has this study considered the issue of computational efficiency? For example, runtime, reduced computational cost, they were mentioned in introduction literature review, but not studied in this study.

We apologize for the oversight. The additional computational cost incurred from using error compensation is minimal; it only adds six global arrays to the entire model code. Therefore, we neglected to elaborate on this aspect. We appreciate your suggestion, following your insightful comment, we found the previously reported figures were indeed rough estimations, so we have re-evaluated the exact computational performance using a measurement tool to determine the runtimes. So, we will:

1. Add a dedicated section 3.4 in the paper to describe the computational performance. The size of the computational performance will be represented in terms of runtime, and we will discuss the runtime for each case in tabular form.
2. Revise the description of computational performance in the abstract to reflect these updated and more accurate measurements.

3.4 Computational performance

In comparison with the SGL, although there is a slight increase in runtime, it is minimal, at only 6% (Jablonowski and Williamson baroclinic wave), 0.3% (Super-cell), 2% (Real data with resolution of 120km) and 18% (Real data with resolution of 240km) (Table 1). This slight increase is attributed to the addition of a small number of global variable arrays when using quasi double-precision. And compared to DBL, QDP demonstrated relatively better performance across different cases, reducing the runtime by 29% (Jablonowski and Williamson baroclinic wave), 29% (Super-cell), 21% (Real data with resolution of 120km) and 6% (Real data with resolution of 240km) (Table 1).

Table 1. Elapsed time of DBL, SGL and QDP test (unit:s).

Case name	DBL	SGL	QDP
Jablonowski and Williamson baroclinic wave	1768	1191	1263
Super-cell	1507	1073	1077
Real data with resolution of 120km	19126	14765	15092
Real data with resolution of 240km	1397	1118	1317

Abstract

The content ‘The round-off error of surface pressure is reduced by 68%, 75%, 97%, 96% in cases, the memory has been reduced by almost half, while the computation increases only 2%, significantly reducing computational cost.’ will be revised to ‘The bias of surface pressure are reduced respectively by 68%, 75%, 97% and 96% in cases, the memory has been reduced by almost half, while the computation increases only 6%, 0.3%, 2%, and 18% in cases, significantly reducing computational cost.’

The model is described poorly. The solution method for the equations is not even mentioned. For example, the finite difference scheme is mainly used to calculate the primary equations for variables studied in this work. And at which step of the equation is the quasi double-precision algorithm specifically applied? The strategy used to compute cell edge, dry air density, potential temperature with quasi double-precision algorithm is also difficult to understand from Figure 3.

Thank you for your valuable feedback regarding the clarity of our model description and the implementation of the quasi double-precision algorithm. We recognize that the solution method for the equations was not sufficiently detailed in our original manuscript. To address this, we will:

1. Add the clear description of the solution method for the equations including temporal integration scheme and spatial discretization scheme in section 2.2. Moreover, the algorithm applied here primarily addresses the rounding error compensation between large and small numbers in addition. Currently, it is only applicable to the time integration process and has not been implemented in the spatial discretization process. I apologize for not mentioning that part in the manuscript. I will revise and provide the supplement. See 2.2 additional content below.
2. Specify at which step the quasi double-precision algorithm is applied within the computation process in section 2.3 and replace the figure (corresponding to Figure 3 in the manuscript) and explain this process using formulas and explanations. See 2.3 additional content below.

2.2 Additional content

The MPAS-A solves the fully compressible, nonhydrostatic equations of motion (Skamarock et al. 2012). The spatial discretization uses a horizontal (spherical) centroidal Voronoi mesh with a terrain-following geometric-height vertical coordinate and C-grid staggering for momentum. The temporal discretization uses the explicit time-split Runge–Kutta technique from Wicker and Skamarock (2002) and Klemp et al. (2007).

The algorithm applied here primarily addresses the rounding error compensation between large and small numbers in addition. Currently, it is only applicable to the time integration process and has not been implemented in the spatial discretization process. Therefore, this section will provide a detailed introduction to the time integration scheme. For the spatial discretization scheme, please refer to Skamarock et al. (2012), and it will not be introduced upon here.

The formulation of the scheme can be considered in on dimension as equation Wicker and Skamarock (2002):

$$\frac{\partial \emptyset}{\partial t} = RHS_{\emptyset} \tag{1}$$

The variable $\emptyset$ represents any prognostic variable in the prognostic equations, while RHS represents the right-hand side of the prognostic equations (i.e., the spatial discretization equation). In MPAS-A, a forward-in-time finite difference is used, and it can be written as Eq. (2):

$$\frac{{\emptyset_i}^{n+1} - {\emptyset_i}^{n}}{\Delta t} = RHS_{\emptyset} \tag{2}$$

Where superscript represent the time step, and subscript represent the position of grid zone.

The two-order Runge-Kutta time scheme is used in MPAS-A as described in Gear et al. (1971):

$$\emptyset^{*} = \emptyset^{t} + \frac{\Delta t}{2} * RHS(\emptyset^{t}) \tag{3}$$

$$\emptyset^{**} = \emptyset^{t} + \frac{\Delta t}{2} * RHS(\emptyset^{*}) \tag{4}$$

$$\emptyset^{t+\Delta t} = \emptyset^{t} + \Delta t * RHS(\emptyset^{**}) \tag{5}$$

2.3 Additional content

According to Equation Eq. (3), (4) and (5), it can be observed that in the time integration scheme, each step involves the process of adding tends on the basic field $\emptyset^{t}$. In numerical models, the basic field is generally much larger than the tends, which aligns with the principles of numerical computation regarding the addition of large and small numbers, as well as the time integration process. It is important to note that the quasi double-precision algorithm currently only addresses time integration and has not been validated during the spatial discretization process. The spatial discretization primarily involves subtraction, specifically the subtraction of a small number from a large number or the subtraction of two close values. Whether this algorithm is applicable in spatial discretization remains uncertain, therefore, we will not apply it in this context.

Based on the application principles of the algorithm, which involve the processes of adding large and small numbers as well as the time integration process, we have established a strategy for applying the quasi double-precision algorithm within the MPAS-A. Specific improvements are provided based on the predictive equations:

$$\frac{\partial \boldsymbol{V}_H}{\partial t} = -\frac{\rho_d}{\rho_m}\left[\nabla_\zeta\left(\frac{p}{\zeta_z}\right) - \frac{\partial z_H p}{\partial \zeta}\right] - \eta \boldsymbol{k} \times \boldsymbol{V}_H - \boldsymbol{v}_H \nabla_\zeta \cdot \boldsymbol{V} - \frac{\partial \Omega v_H}{\partial \zeta} - \rho_d \nabla_\zeta K - eWcos\alpha_r - \frac{v_H W}{r_e} + \boldsymbol{F}_{V_H} \quad (6)$$

$$\frac{\partial W}{\partial t} = -\frac{\rho_d}{\rho_m}\left[\frac{\partial p}{\partial \zeta} + g\tilde{\rho}_m\right] - (\nabla \cdot \boldsymbol{v}W)_\zeta + \frac{uU+vV}{r_e} + e(Ucos\alpha_r - Vsin\alpha_r) + F_W \quad (7)$$

$$\frac{\partial \Theta_m}{\partial t} = -(\nabla \cdot \boldsymbol{V}\theta_m)_\zeta + F_{\Theta_m} \quad (8)$$

$$\frac{\partial \tilde{\rho}_d}{\partial t} = -(\nabla \cdot \boldsymbol{V})_\zeta \quad (9)$$

The meaning of each variable in the equations exactly follows Skamarock et al. (2012), so that we don't repeating explanation. For a numerical model, the most crucial variables are the prognostic variables. Therefore, In the MPAS-A model we applied the quasi double-precision algorithm to the time integration process of these prognostic variables, including horizontal momentum ($\boldsymbol{V}_H$) , dry air density ($\tilde{\rho}_d$), potential temperature (Θ_m) and vertical velocity (W) , that is , the process in red of Eq. (6), (7), (8) and (9). (Only the predictive equations for the dynamic core are presented here, without the scalar transport.)

The color scheme in Figure 5 is very hard to distinguish; the solid purple area is too large, making the gradient difficult to see. Figure 10 has the same issue. It is recommended to refer to the classic NCL color scheme. https://www.ncl.ucar.edu/Applications/era40.shtml

Thank you very much for your suggestions and the website link. I have revised Figures 5 and 10 according to the guidelines provided in the link. Please see below for Figure 1 (corresponding to Figure 5 in the manuscript) and Figure 2 (corresponding to Figure 10 in the manuscript).

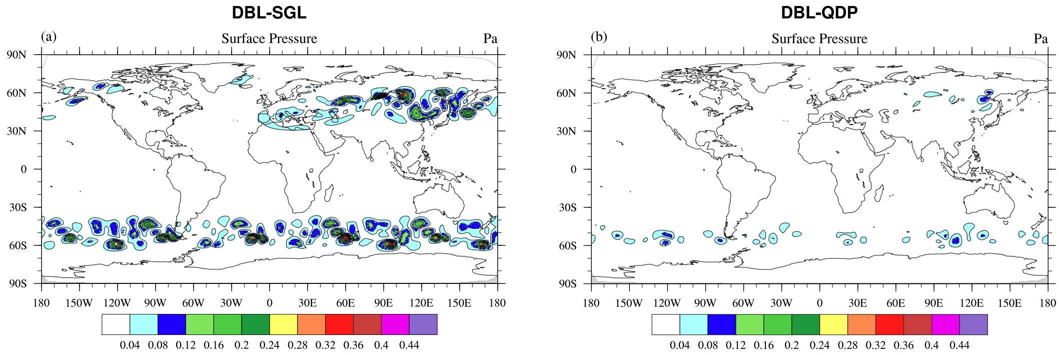

Figure 1. Spatial distributions of averaged (1-15days) difference of surface pressure (units: Pa) between DBL and (a) SGL simulations, (b) QDP simulations (round-off error has reduced) in case of Jablonowski and Williamson baroclinic wave.

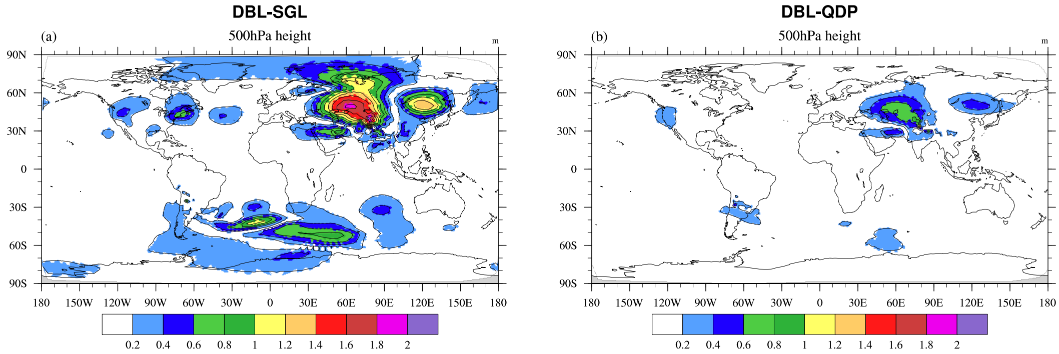

Figure 2. Spatial distributions of averaged (1-15days) difference of 500hPa height (units: m) between DBL and (a) SGL simulation, (b) QDP simulation (resolution: 240 km × 240 km). The RMSE of 500hPa height between DBL and (a) SGL simulation is 2.80×10^{-1} m, (b) QDP simulation is 1.40×10^{-1}m (round-off error has reduced).

Figure 6 (a) shows that DBL-SGL decreases after 1.0, which appears to be caused by a coding error. Please check and confirm the validity of the data.

Thank you for your insightful comment regarding Figure 6(a), we carefully reviewed our code based on your comments and identified that the issue was a problem with the data processing and plotting. We have corrected this issue, and the revised Figure 3 is provided below:

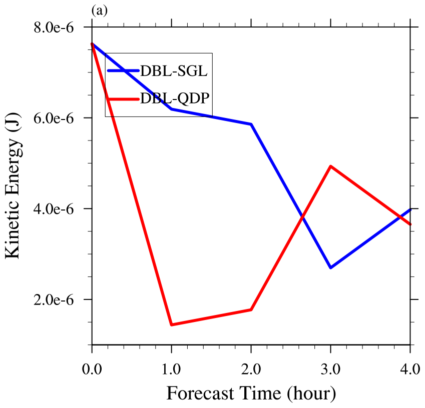

Figure 3. The temporal evolution of spatially averaged difference of kinetic energy between DBL and SGL, as well as difference between DBL and QDP in case of super-cell.

I apologize for needing to provide additional information regarding this section. In all cases presented in the manuscript, the time-evolution plots (Figures 4, 6, and 8 in the manuscript) currently utilize kinetic energy and surface pressure, which correspond to the conservation of energy and mass, respectively. However, I realize that directly using total energy and total mass would offer a more accurate representation. Please see the figures 4, 5 and 6 showing the temporal variations of energy and mass for all cases. If you allow it, I would be happy to replace the existing Figures 4, 6, and 8 (in the manuscript) with these updated versions (figures 4, 5 and 6). I want to emphasize that this change would not affect the overall results or conclusions of the study, which remain consistent with those currently presented in the manuscript. I think your suggestions are greatly appreciated and have been very helpful in improving the manuscript.

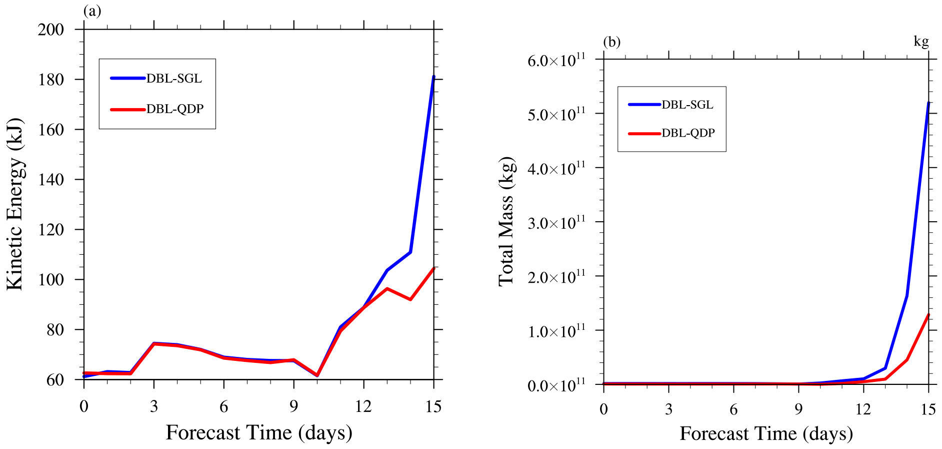

Figure 4. The temporal evolution of spatially averaged difference of (a) total energy, (b) total mass between DBL and SGL, as well as difference between DBL and QDP in case of Jablonowski and Williamson baroclinic wave.

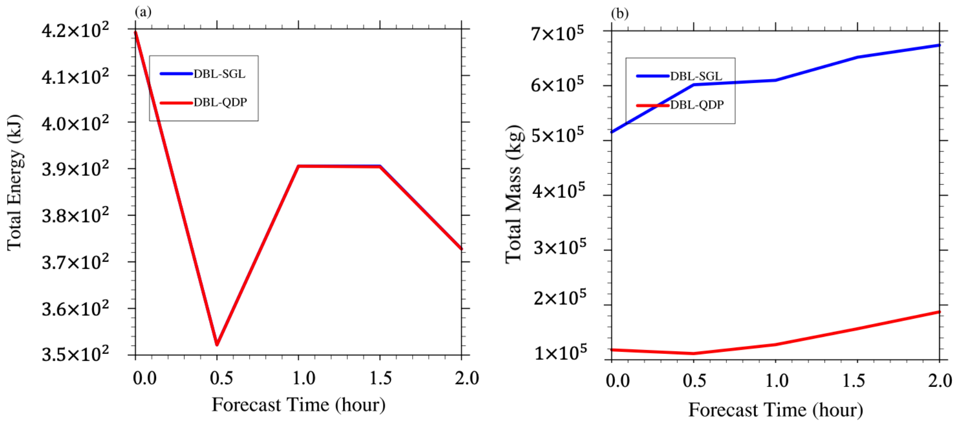

Figure 5. The temporal evolution of spatially averaged difference of (a) total energy, (b) total mass between DBL and SGL, as well as difference between DBL and QDP in case of super-cell.

It is important to note that, for enhanced clarity and to facilitate a better understanding of the trend, the x-axis unit for the figure representing Case 7 has been changed to hours (Fig. 6).

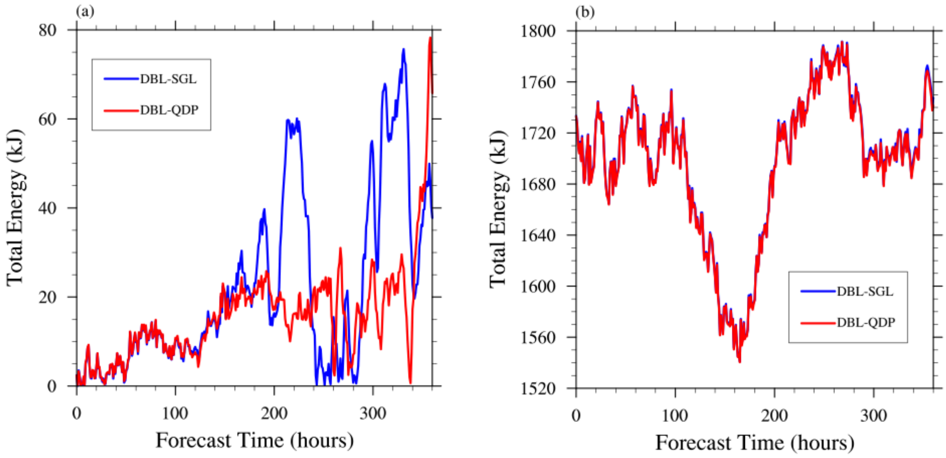

Figure 6. The temporal evolution of spatially averaged difference of total energy between DBL and SGL, as well as difference between DBL and QDP in case of real data, with resolution of (a) 240 km × 240 km, (b) 120 km × 120 km.

Minor comments:

The color bar in Figure 7 seems almost useless.

Thank you very much for your careful review and valuable comments on my manuscript. Regarding the issue you raised about the color bar in the figure 7. After adjusting the color bar, two distinct areas of improvement are now evident, compared to only one in the original figure. I sincerely apologize that the previous color bar setting was not appropriate. I have adjusted the color bar according to your suggestion, and the revised figure is shown as Figure 7 (corresponding to Figure 7 in the manuscript).

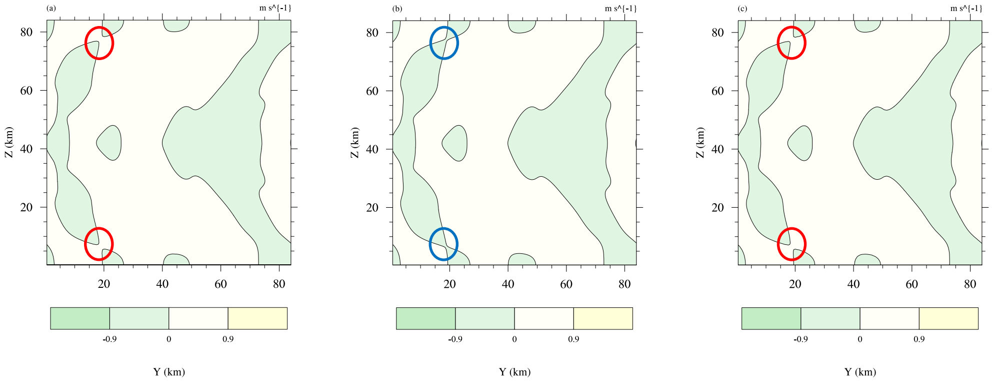

Figure 7. Perturbation theta in super-cell development at 5400s in the (a) DBL simulation, (b) SGL simulation and (c) QDP simulation (bias has reduced), unit: K, the circle represents the clearest error.

Please mention “round-off error” in relevant figures' captions.

Thank you very much for your suggestions. I have revised it. You can see it in the manuscript after receiving feedback from other reviewers.

Figure 11 appears to be somewhat blurry.

Thank you very much for your suggestions. I have revised it. Please see below for Figure 8 (corresponding to Figure 11 in the manuscript).

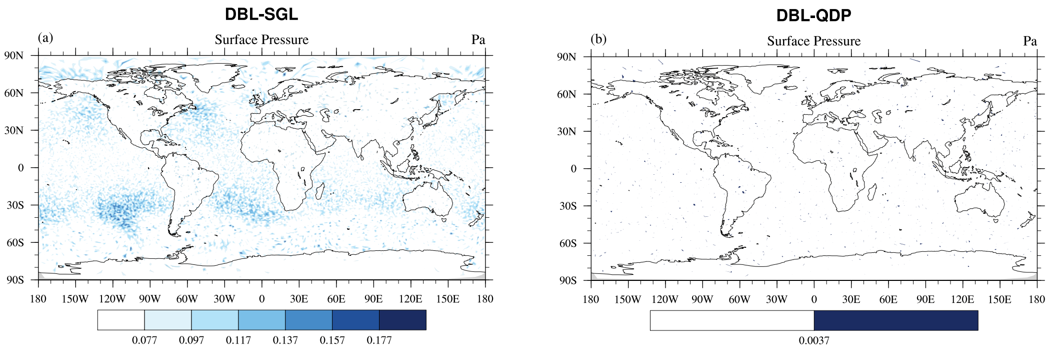

Figure 8. distributions of averaged (1-15days) difference of surface pressure (units: Pa) between DBL and (a) SGL simulation, (b) QDP simulation (resolution: 120 km × 120 km) (round-off error has reduced). The RMSE of surface pressure between DBL and (a) SGL simulation is 6.33×10^{-2} Pa, (b) QDP simulation is 2.25×10^{-2} Pa.

It's important to note that the color bars for Figures (a) and (b) are intentionally set differently due to the significant disparity in their respective threshold ranges. If identical color scales were applied, Figure (b) would appear entirely white.

Wicker, L. J., & Skamarock, W. C.: Time-splitting methods for elastic models using forward time schemes. Monthly Weather Review, 130(8), 2088. https://www.proquest.com/scholarly-journals/time-splitting-methods-elastic-models-using/docview/198148677/se-2, 2002.

Gear, C. W.: Numerical initial value problems in ordinary differential equations[M]. Englewood Cliffs, N.J: Prentice-Hall, 1971.