Theoretical Time Evolution of Numerical Errors When Using Floating Point Numbers in Shallow-Water Models

Yamaura, Tsuyoshi

Nishizawa, Seiya

Tomita, Hirofumi

(Citation)
Journal of Advances in Modeling Earth Systems (JAMES),11(10):3235-3250

(Issue Date)
2019-10

(Resource Type)
journal article

(Version)
Version of Record

(URL)
https://hdl.handle.net/20.500.14094/90007233

JAMES | Journal of Advances in Modeling Earth Systems

RESEARCH ARTICLE
10.1029/2019MS001615

Theoretical Time Evolution of Numerical Errors When Using Floating Point Numbers in Shallow-Water Models

Tsuyoshi Yamaura[1,2], **Seiya Nishizawa**[1], **and Hirofumi Tomita**[1]

[1]RIKEN Center for Computational Science, Japan, [2]Research Center for Urban Safety and Security, Kobe University, Japan

Key Points:
- Quantitative evaluation of the numerical errors that occur when using floating point numbers (FPN errors) in shallow-water equations
- The FPN errors evolve as a random walk, which can be represented by stochastic forcing
- The random walk evolution can be observed during the initial stages of the time evolution, even when the simulation is in an unstable state

Correspondence to:
T. Yamaura,
tyamaura@riken.jp

Citation:
Yamaura, T., Nishizawa, S., & Tomita, H. (2019). Theoretical time evolution of numerical errors when using floating point numbers in shallow-water models. *Journal of Advances in Modeling Earth Systems*, *11*, 3235–3250. https://doi.org/10.1029/2019MS001615

Received 15 JAN 2019
Accepted 11 SEP 2019
Accepted article online 1 OCT 2019
Published online 23 OCT 2019

Abstract We carried out a theoretical investigation of the impact of the numerical errors caused by using floating point numbers (FPNs) in simulations, such as rounding errors. Under the presupposition that model variables can be written as the linear sum of the true value and the numerical error, equations governing the time evolution of numerical errors due to FPNs (FPN errors) are obtained by considering the total errors of the results of simulations of shallow-water models and estimating the errors incurred by using FPNs with varying precision. We can use the time evolution equations to estimate the behavior of the FPN errors, then confirm these estimations by carrying out numerical simulations. In a geostrophic wind balance state, the FPN error oscillates and gradually increases in proportion to the square root of the number of time steps, like a random walk. We found that the error introduced by using FPNs can be considered as stochastic forcing. In a state of barotropic instability, the FPN error initially evolves as stochastic forcing, as in the case of the geostrophic wind balance state. However, it then begins to increase exponentially, like a barotropic instability wave. These numerical results are obtained by using a staggered-grid arrangement and stable time-integration method to retain near-neutral numerical stability in the simulations. The FPN error tends to behave as theoretically predicted if the numerical stability is close to neutral.

1. Introduction

The computational resources required by the high-performance computing (HPC) systems used for meteorological and climatological simulations are increasing continuously, as researchers strive for more accurate and higher resolution results using larger numbers of ensembles, modeled by more sophisticated physical processes, etc. The spatial and temporal resolutions that are currently in use are still insufficient for numerical weather/climate simulations to accurately model various meteorological phenomena (Shapiro et al., 2010; Shukla et al., 2010). Huge HPC systems are inevitably used for such computations. As a result, we are now facing various hardware issues.

One of the most crucial hardware issues with HPC systems is the data transfer speed during simulations. For meteorological and climatological simulations, data transfer between memory-CPU and computational nodes is one of the costliest operations. Double-precision (DP) floating point numbers (FPNs) are used to simulate most numerical weather/climate models. These are intended to be resistant to rounding errors in contrast to single-precision (SP) FPNs. However, excessive accuracy increases the amount of unnecessary computation, in terms of both data transfer and processing. One straightforward way of addressing this issue is to reduce the significant bit-width of the FPN, enabling it to tolerate a larger rounding error (Lingamneni et al., 2011). For example, according to the IEEE standard for floating-point arithmetic (ANSI/IEEE Std. 754-2008; Hereafter, IEEE754) which is employed by many current computers, the significant bit-width of an SP FPN is less than half that of a DP FPN. Using FPNs with smaller bit-widths is expected to reduce the loads imposed by communication, CPU cache, and memory. Some researchers have previously investigated acceleration effects using low-precision FPNs in a numerical weather/climate model (e.g. Düben & Palmer, 2014; Gan et al., 2015; Lingamneni et al., 2011). Düben and Palmer (2014) reported a 25% reduction in computing time for a 10-day weather simulation using SP FPNs. Recently, Váňa et al. (2017) reduced the run time of the realistic simulations by approximately 40% with SP FPNs, using the Integrated Forecast System (IFS) spectral model managed by the European Centre for Medium-Range Weather Forecasts. Nakano et al. (2018) used an

SP FPN and the dynamic core of a global, fully compressible non-hydrostatic model to achieve a 46% reduction in run time when performing baroclinic wave simulations with small errors. More rapid techniques such as employing general-purpose computing on graphics processing units (GPGPUs) and field-programmable gate arrays (FPGAs) in numerical simulations will become more important in future geophysical models. Yamagishi and Matsumura (2016) applied mixed-precision FPNs when preconditioning the Poisson/Helmholtz solver of a non-hydrostatic ocean model featuring a GPU. It was concluded that the GPU-implemented ocean model was 4.7 fold faster than a comparable CPU model; also, the former model lacked significant numerical errors. Thus, reducing the bit widths of FPNs has potential advantages in terms of speeding up simulations.

Although accelerating computations using low-precision FPNs are effective for meteorological and climatological simulations, we should consider the negative effects of using low-precision FPNs for these simulations, namely numerical errors. Researchers have previously studied the effects of the numerical errors caused by the use of low-precision FPNs, compared with the results of using high-precision FPNs. For example, Düben and Palmer (2014) showed that the difference between numerical simulations with SP and DP FPNs is smaller than the standard deviation of the ensemble simulations when carrying out numerical integration for 10 days of an atmospheric general circulation model. On the other hand, it is unclear how the numerical errors that occur when using lower precision FPNs, such as half-precision FPNs, affect the results of numerical simulations. It will soon become necessary to reduce the significant bit-width of the FPN in experiments with higher spatial resolutions, due to various computational problems. For example, the ratio of the memory capacity and bandwidth to floating-point operations per second (FLOPS) will decrease (Shalf, 2010). Therefore, one simple question arises: how far can we reduce the precision of the FPNs used in experiments at higher spatial resolutions? To answer this question, we should theoretically investigate the impact of the numerical errors that arise when using low-precision FPNs. Specifically, we need to establish and evaluate the governing equations of the numerical errors of meteorological and climatological simulations. However, recent weather/climate models are too complex to allow us to understand the impact; we thus begin by using a simple atmospheric system such as shallow-water equations. If a theory of FPN errors could be established, low-precision FPN calculations could aid the theoretical development of weather/climate models that lack significant numerical errors. Although there are some limitations with respect to the theory of FPN errors to a realistic weather/climate model, this work nevertheless represents the first step for establishing the theory. It is important to understand the behavior of numerical errors using low-precision FPNs so that we can develop not only numerical weather/climate models but also various stencil calculations that are required for computational fluid dynamics.

The rest of this paper is organized as follows: We present the theoretical background of the numerical errors that arise when modeling simple atmospheric systems in Section 2. We then verify how numerical errors propagate through prognostic variables as time advances in numerical simulations in Section 3. Finally, we discuss our results and draw our conclusions in Section 4.

2. Theory of the Time Evolution of FPN Errors

Initially, we explain the types of numerical error, i.e., truncation error, overflow, rounding error, loss of trailing digits, and loss of significant digits. Truncation errors are caused by discretizing differential equations; this is a well-known fact and has been extensively investigated. Rounding errors are often caused when computers handle real numbers. Loss of trailing digits indicates that a small number is neglected when added to a very large number, while loss of significant digits arises when two nearly equal numbers are subtracted. Hereinafter, we term the last three errors “FPN errors”. This study focuses on the effects of FPN errors on numerical simulations. Although FPNs may create many numerical errors, the FPN format better expresses real numbers in numerical simulations than do integers and fixed-point numbers, enhancing computational speed. For convenience, we treat rounding errors as FPN errors, although these errors may also occur when using integers or fixed-point numbers. We do not consider overflow errors in this study, as they are caused by using exponential bits with inadequate widths. Significant bits are more closely linked to the accuracy of numerical simulations than exponential bits. Furthermore, overflows are easy to detect by catching the floating-point exceptions raised by the CPU. Hence, we assume that the numerical results of this study are free from such errors.

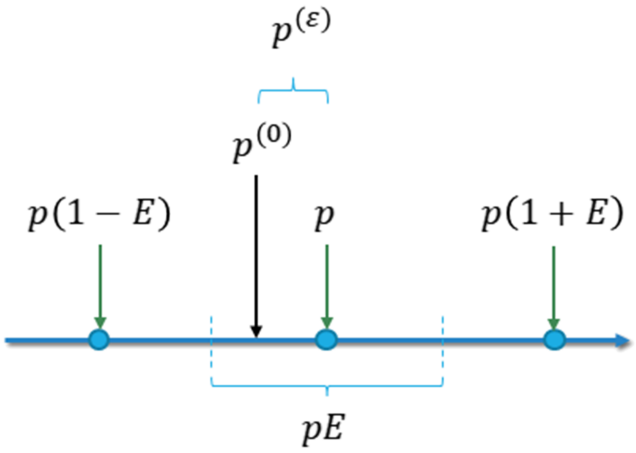

Figure 1. Schematic diagram of the floating-point error when using banker's rounding, after substituting a value into the model variable p.

Model variables used to represent prognostic parameters can be written as the linear sum of the true value and the numerical error after first substituting the relevant value:

$$p = p^{(0)} + p^{(\epsilon)}, \tag{1}$$

where p is a model variable, $p^{(0)}$ is the true value of the prognostic variable, and $p^{(\epsilon)}$ is the FPN error of the prognostic variable. Figure 1 is a schematic diagram of the relationship between p, $p^{(0)}$, and $p^{(\epsilon)}$, after substituting the value for the model variable p. Note that p is a FPN, while $p^{(0)}$ and $p^{(\epsilon)}$ are real numbers. As a FPN is a discretized number, the FPN error is caused by rounding in this situation. Hence, if using "round to nearest, ties to even" rule, where rounding is one to the nearest value with an even least significant digit, the maximum size of $p^{(\epsilon)}$ is less than the half of pE, where E is the machine epsilon. This rounding-mode is the standard for IEEE754 and is referred to as banker's rounding. When using very low-precision FPNs, the magnitude of $p^{(0)}$ is almost the same as that of $p^{(\epsilon)}$, implying that such numerical simulations are meaningless. Thus, we assume that the magnitude of $p^{(0)}$ is much larger than that of $p^{(\epsilon)}$.

The aim of this work was to investigate the time evolution of the FPN error theoretically. However, we cannot measure $p^{(\epsilon)}$ directly because we do not know the correct values of $p^{(0)}$ and $p^{(\epsilon)}$ in p. To cancel $p^{(0)}$, we consider the difference between high- and low-precision FPNs:

$$p^{(L)} - p^{(H)} = p^{(\epsilon_L)} - p^{(\epsilon_H)} \equiv p^{(\delta)} \tag{2}$$

where $p^{(H)}$ is a variable represented by a high-precision FPN ($p^{(H)} = p^{(0)} + p^{(\epsilon_H)}$), and $p^{(L)}$ is one represented by a low-precision FPN ($p^{(L)} = p^{(0)} + p^{(\epsilon_L)}$). If $p^{(\epsilon_H)}$ is sufficiently smaller than $p^{(\epsilon_L)}$, then $p^{(\delta)}$ is approximately equal to $p^{(\epsilon_L)}$. According to Figure 1, the magnitude of $p^{(\epsilon)}$ is less than half that of pE. Then we can interpret the above condition as

$$\left|\frac{pE_H}{2}\right| \ll \left|\frac{pE_L}{2}\right| \Rightarrow E_H \ll E_L,$$

where E_H and E_L are the machine epsilons of high- and low-precision FPNs, respectively. Because the value of the machine epsilon under the IEEE754 is 2^{1-d} where d is the length of the significant bit, the above condition can be rearranged as

$$E_H \ll E_L \Rightarrow 2^{1-d_H} \ll 2^{1-d_L} \Rightarrow 2^{d_L - d_H} \ll 1, \tag{3}$$

where d_H and d_L are the lengths of the significant bits of the high- and low-precision FPNs, respectively. When Equation (3) is established, the FPN error can be estimated by the following approximation: ($p^{(\delta)} \approx p^{(\epsilon_L)}$).

We consider the limitations associated with establishment of the above investigation after the time evolution of p. If the system evolving p yields identical results using high- and low-precision FPNs, the time evolution of $p^{(0)}$ does not differ because a FPN can be divided into $p^{(0)}$ and $p^{(\epsilon)}$. The time evolution of p is the linear sum of the time integration of the true value and the numerical error as long as the error remains much smaller than

the true value. Therefore, the difference between high- and low-precision FPNs after time evolution cancels $p^{(0)}$ and extracts the difference between $p^{(\varepsilon_H)}$ and $p^{(\varepsilon_L)}$. However, many thresholds are employed in real weather/climate models. These can qualitatively change numerical simulations when the values of prognostic variables vary slightly. For example, if a convective parameterization scheme is active in low-precision FPNs and non-active for high-precision FPNs at the threshold, the difference between high- and low-precision FPNs no longer reflects the FPN error term. Considering the above limitation, we investigated the behavior of FPN errors in one of the simplest atmospheric systems (that represented by shallow-water equations) in this study. The model shares the dynamic characteristics of the three-dimensional model, but is simpler to discuss. In future, the behavior of FPN errors in real weather/climate models will be investigated.

2.1. Time Evolution Equations of FPN Errors in Shallow-Water Equations

We now introduce the governing equations for the two-dimensional shallow-water model used in this study. The set of governing equations for the shallow-water model is obtained by considering the momentum equations in the x- and y-directions and the continuity equation, as follows:

$$\begin{aligned}\frac{\partial u}{\partial t} &= -u\frac{\partial u}{\partial x} - v\frac{\partial u}{\partial y} - \frac{\partial \phi}{\partial x} + fv,\\ \frac{\partial v}{\partial t} &= -u\frac{\partial v}{\partial x} - v\frac{\partial v}{\partial y} - \frac{\partial \phi}{\partial y} - fu,\\ \frac{\partial \phi}{\partial t} &= -u\frac{\partial \phi}{\partial x} - v\frac{\partial \phi}{\partial y} - \phi\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right),\end{aligned} \tag{4}$$

where u and v are the velocity components in the x- and y− directions (m s^{-1}), respectively, ϕ is the geopotential (m^2 s^{-2}), and f is the Coriolis parameter (s^{-1}). The momentum equations include terms for the advection, pressure gradient force, and Coriolis force, while the continuity equation takes into account the advection and the divergence. In numerical simulations, the differential terms in Equation (4) are discretized as

$$\begin{aligned}\frac{\Delta_N u_{I,J,N}}{\Delta t} &= -u_{I,J,N}\frac{\Delta_I u_{I,J,N}}{\Delta x} - v_{I,J,N}\frac{\Delta_J u_{I,J,N}}{\Delta y} - \frac{\Delta_I \phi_{I,J,N}}{\Delta x} + fv_{I,J,N},\\ \frac{\Delta_N v_{I,J,N}}{\Delta t} &= -u_{I,J,N}\frac{\Delta_I v_{I,J,N}}{\Delta x} - v_{I,J,N}\frac{\Delta_J v_{I,J,N}}{\Delta y} - \frac{\Delta_J \phi_{I,J,N}}{\Delta y} - fu_{I,J,N},\\ \frac{\Delta_N \phi_{I,J,N}}{\Delta t} &= -u_{I,J,N}\frac{\Delta_I \phi_{I,J,N}}{\Delta x} - v_{I,J,N}\frac{\Delta_J \phi_{I,J,N}}{\Delta y} - \phi_{I,J,N}\left(\frac{\Delta_I u_{I,J,N}}{\Delta x} + \frac{\Delta_J v_{I,J,N}}{\Delta y}\right),\end{aligned} \tag{5}$$

where I and J are grid numbers in the x- and y-directions, respectively, and Δx and Δy are grid distances in the x- and y-directions, respectively. N is the number of time steps and Δt is the time increment. We introduce difference operators so that we can simply describe the difference terms: Δ_I, Δ_J, and Δ_N are the difference operators in the x-, y-, and t-directions, respectively. Hereinafter, to avoid mathematical complications in theoretical analysis, equations are written with a simple A-grid arrangement and the first-order forward difference operator: $\Delta_I u_{I,J,N} = u_{I+1,J,N} - u_{I,J,N}$. We assume that the theoretical nature of the time evolution of FPN errors studied here does not depend on numerical choices, such as the grid-arrangement or time-step scheme. We assume neutral numerical stability in the theoretical discussion that follows. The validity of this assumption is discussed at the end of this subsection.

A non-negligible numerical error arises when Equation (4) is converted into Equation (5), namely the truncation error. The Taylor expansion of the first derivative of p can be written as

$$\frac{\partial p}{\partial t} = \frac{\Delta_N p_{I,J,N}}{\Delta t} + T(\Delta t) = \frac{\Delta_N p^{(0)}_{I,J,N}}{\Delta t} + \frac{\Delta_N p^{(\varepsilon)}_{I,J,N}}{\Delta t} + T(\Delta t), \tag{6}$$

where $T(\Delta t)$ is the residual function of the Taylor series, which implies a truncation error. When using the first-order forward difference, the magnitude of $T(\Delta t)$ is equal to that of $O(\Delta t)$. Although we generally believe that the rounding error is smaller than the truncation error, it is not obvious that this is the case in high-resolution simulations, because the truncation error decreases with Δt, but the rounding error does not. To evaluate the rounding error in Equation (3), we introduce the difference between the discretized derivatives obtained using high- and low-precision FPNs:

$$\frac{\Delta_N p^{(L)}_{I,J,N}}{\Delta t} - \frac{\Delta_N p^{(H)}_{I,J,N}}{\Delta t} + T'(\Delta t) = \frac{\Delta_N p^{(\delta)}_{I,J,N}}{\Delta t} + T'(\Delta t) = \frac{\partial p^{(\delta)}}{\partial t}. \tag{7}$$

Note that $T'(\Delta t)$ in Equation (7) indicates the truncation error function of the rounding error. Therefore, $T'(\Delta t)$ is not the same as $T(\Delta t)$ in Equation (6). As such, we can express the time evolution of $p^{(\delta)}$ in terms of the difference between the discretized derivative term when using high- and low-precision FPNs.

From Equation (5), we obtain the following equations:

$$\begin{aligned}
\frac{\Delta_N u^{(\delta)}_{I,J,N}}{\Delta t} &= -\left(u^{(0)}_{I,J,N}\frac{\Delta_I u^{(\delta)}_{I,J,N}}{\Delta x} + u^{(\delta)}_{I,J,N}\frac{\Delta_I u^{(0)}_{I,J,N}}{\Delta x}\right) - \left(v^{(0)}_{I,J,N}\frac{\Delta_J u^{(\delta)}_{I,J,N}}{\Delta y} + v^{(\delta)}_{I,J,N}\frac{\Delta_J u^{(0)}_{I,J,N}}{\Delta y}\right) \\
&\quad -\frac{\Delta_I \phi^{(\delta)}_{I,J,N}}{\Delta x} + f v^{(\delta)}_{I,J,N} + F_u(I,J,N), \\
\frac{\Delta_N v^{(\delta)}_{I,J,N}}{\Delta t} &= -\left(u^{(0)}_{I,J,N}\frac{\Delta_I v^{(\delta)}_{I,J,N}}{\Delta x} + u^{(\delta)}_{I,J,N}\frac{\Delta_I v^{(0)}_{I,J,N}}{\Delta x}\right) - \left(v^{(0)}_{I,J,N}\frac{\Delta_J v^{(\delta)}_{I,J,N}}{\Delta y} + v^{(\delta)}_{I,J,N}\frac{\Delta_J v^{(0)}_{I,J,N}}{\Delta y}\right) \\
&\quad -\frac{\Delta_J \phi^{(\delta)}_{I,J,N}}{\Delta y} - f u^{(\delta)}_{I,J,N} + F_v(I,J,N), \\
\frac{\Delta_N \phi^{(\delta)}_{I,J,N}}{\Delta t} &= -\left(u^{(0)}_{I,J,N}\frac{\Delta_I \phi^{(\delta)}_{I,J,N}}{\Delta x} + u^{(\delta)}_{I,J,N}\frac{\Delta_I \phi^{(0)}_{I,J,N}}{\Delta x}\right) - \left(v^{(0)}_{I,J,N}\frac{\Delta_J \phi^{(\delta)}_{I,J,N}}{\Delta y} + v^{(\delta)}_{I,J,N}\frac{\Delta_J \phi^{(0)}_{I,J,N}}{\Delta y}\right) \\
&\quad -\phi^{(0)}_{I,J,N}\left(\frac{\Delta_I u^{(\delta)}_{I,J,N}}{\Delta x} + \frac{\Delta_J v^{(\delta)}_{I,J,N}}{\Delta y}\right) - \phi^{(\delta)}_{I,J,N}\left(\frac{\Delta_I u^{(0)}_{I,J,N}}{\Delta x} + \frac{\Delta_J v^{(0)}_{I,J,N}}{\Delta y}\right) + F_\phi(I,J,N),
\end{aligned} \tag{8}$$

where F_u, F_v, and F_ϕ are residual terms due to linearization of Equation (8), which are defined

$$\begin{aligned}
F_u(I,J,N) &= NL_u(\varepsilon_L) - NL_u(\varepsilon_H) + RN(u^{(\varepsilon_L)}_{I,J,N}) - RN(u^{(\varepsilon_H)}_{I,J,N}) + T'(\Delta t) + T'(\Delta x) + T'(\Delta y), \\
F_v(I,J,N) &= NL_v(\varepsilon_L) - NL_v(\varepsilon_H) + RN(v^{(\varepsilon_L)}_{I,J,N}) - RN(v^{(\varepsilon_H)}_{I,J,N}) + T'(\Delta t) + T'(\Delta x) + T'(\Delta y), \\
F_\phi(I,J,N) &= NL_\phi(\varepsilon_L) - NL_\phi(\varepsilon_H) + RN(\phi^{(\varepsilon_L)}_{I,J,N}) - RN(\phi^{(\varepsilon_H)}_{I,J,N}) + T'(\Delta t) + T'(\Delta x) + T'(\Delta y),
\end{aligned} \tag{9}$$

and

$$\begin{aligned}
NL_u(\varepsilon) &= -u^{(\varepsilon)}_{I,J,N}\frac{\Delta_I u^{(\varepsilon)}_{I,J,N}}{\Delta x} - v^{(\varepsilon)}_{I,J,N}\frac{\Delta_J u^{(\varepsilon)}_{I,J,N}}{\Delta y}, \\
NL_v(\varepsilon) &= -u^{(\varepsilon)}_{I,J,N}\frac{\Delta_I v^{(\varepsilon)}_{I,J,N}}{\Delta x} - v^{(\varepsilon)}_{I,J,N}\frac{\Delta_J v^{(\varepsilon)}_{I,J,N}}{\Delta y}, \\
NL_\phi(\varepsilon) &= -u^{(\varepsilon)}_{I,J,N}\frac{\Delta_I \phi^{(\varepsilon)}_{I,J,N}}{\Delta x} - v^{(\varepsilon)}_{I,J,N}\frac{\Delta_J \phi^{(\varepsilon)}_{I,J,N}}{\Delta y} - \phi^{(\varepsilon)}_{I,J,N}\left(\frac{\Delta_I u^{(\varepsilon)}_{I,J,N}}{\Delta x} + \frac{\Delta_J v^{(\varepsilon)}_{I,J,N}}{\Delta y}\right).
\end{aligned}$$

RN is the FPN error obtained when solving a dynamical process, and arises due to the rounding of $p^{(L)}$ and $p^{(H)}$. We include the loss of both the trailing digit and the significant digit in the FPN error. The values of the residual terms may differ between machines, experimental designs, implementations, optimization levels used for compilation, etc. We treat these terms as stochastic variables because some numerical errors may be acceptable for representing the subgrid-scale variability (Düben & Dolaptchiev, 2015). For example, the order of operations affects the rounding errors: If $A>B>C>D$, the rounding errors will be larger for $(((A+B)+C)+D)$ than for $(A+(B+(C+D)))$. If A is very large ($A=A^{(0)}+A^{(\epsilon)}$, $A^{(\epsilon)}>B>C>D$), the former summation is equal to A given the loss of trailing digits, whereas the latter summation is not necessarily equal to A because $(B+(C+D))$ may be larger than $A^{(\epsilon)}$. This effect is important when evaluating FPNs accurately, but it is too difficult to derive an accurate value deterministically during simulation. Therefore, we consider that the effect of operation order with respect to rounding errors is stochastic because the magnitudes of such errors are determined by $O(A^{(\epsilon)})$ in the above case. We confirmed the behavior of the residual terms by carrying out the numerical simulations described in Section 3.1. The most important issue addressed in this work is to understand the behavior of Equation (8).

Equation (8) is based on the assumption that FPN errors, $p^{(\delta)}$, are caused by differences in the results obtained when using the high- versus low-precision FPNs shown in Equation (2). Most importantly, both $p^{(H)}$ and $p^{(L)}$ are solutions that include numerical errors according to the numerical scheme used. The numerical errors (other than the FPN errors) attributable to the schemes are prevented using our subduction. Although squared errors (the numerical errors of $p^{(\delta)}$) may be present in residual terms, we assume that such

errors are smaller than $p^{(\delta)}$ in a weak non-linear system. Then, any difference in grid arrangement is also prevented by our subduction, because we can ignore the residuals of T'. We use the fourth-order Runge-Kutta (RK4) method for time integration in the numerical simulation, to retain near-neutral numerical stability. Although the number of operations differs among the time integration schemes, the expected mean and variance of random forcing are the same for all schemes. We only consider the magnitude of rounding errors, which is almost identical among the various time integration schemes. We obtained almost the same conclusions using the third-order Runge-Kutta (RK3) method for time integration. Therefore, we assume that the theoretical time evolution of FPN errors studied here does not depend on numerical choices. The FPN error tends to behave as predicted theoretically if the numerical stability is close to neutral.

2.2. Initial Value Problem in a Geostrophic Wind Balance State

We consider the geostrophic wind balance state as one of the steady states that arises when the Coriolis force and pressure gradient force are dynamically balanced. An initial condition under geostrophic wind balance is as follows:

$$\begin{aligned} u &= -\frac{1}{f}\frac{d\phi}{dy}, \\ v &= 0, \\ \phi &= \phi(y). \end{aligned} \tag{10}$$

In this case, all of the difference terms in the x-direction are equal to zero. Note that the $v^{(0)}$ terms are eliminated, but $v^{(\delta)}$ is nonzero. We can now write Equation (8) as

$$\begin{aligned} \frac{\Delta_N u^{(\delta)}_{I,J,N}}{\Delta t} &= -v^{(\delta)}_{I,J,N}\frac{\Delta_J u^{(0)}_{I,J,N}}{\Delta y} + f v^{(\delta)}_{I,J,N} + F_u(I,J,N), \\ \frac{\Delta_N v^{(\delta)}_{I,J,N}}{\Delta t} &= -\frac{\Delta_J \phi^{(\delta)}_{I,J,N}}{\Delta y} - f u^{(\delta)}_{I,J,N} + F_v(I,J,N), \\ \frac{\Delta_N \phi^{(\delta)}_{I,J,N}}{\Delta t} &= -v^{(\delta)}_{I,J,N}\frac{\Delta_J \phi^{(0)}_{I,J,N}}{\Delta y} - \phi^{(0)}_{I,J,N}\frac{\Delta_J v^{(\delta)}_{I,J,N}}{\Delta y} + F_\phi(I,J,N). \end{aligned} \tag{11}$$

We subtract the second equation at N from the equation at N+1 and divided by Δt to obtain the following equation:

$$\frac{\Delta_N^2 v^{(\delta)}_{I,J,N}}{\Delta t^2} = -\frac{1}{\Delta y}\left(\frac{\Delta_N \phi^{(\delta)}_{I,J+1,N}}{\Delta t} - \frac{\Delta_N \phi^{(\delta)}_{I,J,N}}{\Delta t}\right) - f\frac{\Delta_N u^{(\delta)}_{I,J,N}}{\Delta t} + \frac{F_v(I,J,N+1) - F_v(I,J,N)}{\Delta t}, \tag{12}$$

where

$$\Delta_N^2 v^{(\delta)}_{I,J,N} = \Delta_N\left(\Delta_N v^{(\delta)}_{I,J,N}\right) = \Delta_N v^{(\delta)}_{I,J,N+1} - \Delta_N v^{(\delta)}_{I,J,N} = v^{(\delta)}_{I,J,N+2} - 2v^{(\delta)}_{I,J,N+1} + v^{(\delta)}_{I,J,N},$$

which is the second-order forward difference operation. Substituting the first and the third equations into Equation (12) yields an inhomogeneous linear wave equation:

$$\begin{aligned} \frac{\Delta_N^2 v^{(\delta)}_{I,J,N}}{\Delta t^2} &= \phi^{(0)}_{I,J+1,N}\frac{\Delta_J^2 v^{(\delta)}_{I,J,N}}{\Delta y^2} + \frac{\Delta_J \phi^{(0)}_{I,J+1,N}}{\Delta y}\frac{\Delta_J v^{(\delta)}_{I,J,N}}{\Delta y} - f^2 v^{(\delta)}_{I,J,N} \\ &\quad - f F_u(I,J,N) + \frac{F_v(I,J,N+1) - F_v(I,J,N)}{\Delta t} + \frac{F_\phi(I,J+1,N) - F_\phi(I,J,N)}{\Delta y}, \end{aligned} \tag{13}$$

from Equation (10). Equation (13) means that $v^{(\delta)}$ oscillates in the y-direction with angular frequency $\sqrt{\phi^{(0)}}$ and increases gradually with the residual terms, F_u, F_v, and F_ϕ. That is, the residual terms may break the steadiness of the geostrophic wind balance. Note that the magnitude of the FPN error oscillates in a wave-like manner, with stochastic amplification.

2.3. Initial Value Problem in a Barotropic Instability State

We now discuss the barotropic instability state, which is representative of the unstable states of the shallow-water equations. We add the small disturbances (v=$v(x)$,$u \gg v$) from Equation (10) as initial conditions. We

assume a rigid lid for the upper condition. This is often used to investigate barotropic instabilities of shallow-water equations. We then omit the divergence terms from Equation (8):

$$\begin{aligned}\frac{\Delta_N u^{(\delta)}_{I,J,N}}{\Delta t} &= -\left(v^{(0)}_{I,J,N}\frac{\Delta_J u^{(\delta)}_{I,J,N}}{\Delta y} + v^{(\delta)}_{I,J,N}\frac{\Delta_J u^{(0)}_{I,J,N}}{\Delta y}\right) - \frac{\Delta_I \phi^{(\delta)}_{I,J,N}}{\Delta x} + f v^{(\delta)}_{I,J,N} + F_u(I,J,N),\\ \frac{\Delta_N v^{(\delta)}_{I,J,N}}{\Delta t} &= -\left(u^{(0)}_{I,J,N}\frac{\Delta_I v^{(\delta)}_{I,J,N}}{\Delta x} + u^{(\delta)}_{I,J,N}\frac{\Delta_I v^{(0)}_{I,J,N}}{\Delta x}\right) - \frac{\Delta_J \phi^{(\delta)}_{I,J,N}}{\Delta y} - f u^{(\delta)}_{I,J,N} + F_v(I,J,N),\end{aligned} \tag{14}$$

where

$$\frac{\Delta_I u^{(0)}_{I,J,N}}{\Delta x} = \frac{\Delta_J v^{(0)}_{I,J,N}}{\Delta y} = \frac{\Delta_I u^{(\delta)}_{I,J,N}}{\Delta x} = \frac{\Delta_J v^{(\delta)}_{I,J,N}}{\Delta y} = 0.$$

To eliminate the pressure gradient force terms in Equation (14), we subtract the first equation at J from the equation at J+1 and divide by Δy; then we subtract the second equation at I from the equation at I+1 and divide by Δx. We then obtain the following equation by subtracting the former from the latter:

$$\begin{aligned}\frac{\Delta_N}{\Delta t}\left(\frac{\Delta_I v^{(\delta)}_{I,J,N}}{\Delta x} - \frac{\Delta_J u^{(\delta)}_{I,J,N}}{\Delta y}\right) &= -u^{(0)}_{I,J,N}\frac{\Delta_I^2 v^{(\delta)}_{I,J,N}}{\Delta x^2} - u^{(\delta)}_{I,J,N}\frac{\Delta_I^2 v^{(0)}_{I,J,N}}{\Delta x^2}\\ &+ v^{(0)}_{I,J,N}\frac{\Delta_J^2 u^{(\delta)}_{I,J,N}}{\Delta y^2} + v^{(\delta)}_{I,J,N}\frac{\Delta_J^2 u^{(0)}_{I,J,N}}{\Delta y^2}\\ &+ \frac{F_v(I+1,J,N) - F_v(I,J,N)}{\Delta x} + \frac{F_u(I,J+1,N) - F_u(I,J,N)}{\Delta y}.\end{aligned} \tag{15}$$

This is the vorticity equation of the FPN error. $v^{(0)}$ is much smaller than $u^{(0)}$, which means that the second and third terms on the right-hand side are much smaller than the first and fourth terms, respectively. We then introduce the stream function ψ of the FPN error, as follows:

$$\begin{aligned}\frac{\Delta_N}{\Delta t}\left(\frac{\Delta_I^2 \psi^{(\delta)}_{I,J,N}}{\Delta x^2} + \frac{\Delta_J^2 \psi^{(\delta)}_{I,J,N}}{\Delta y^2}\right) &= -u^{(0)}_{I,J,N}\frac{\Delta_I}{\Delta x}\left(\frac{\Delta_I^2 \psi^{(\delta)}_{I,J,N}}{\Delta x^2} + \frac{\Delta_J^2 \psi^{(\delta)}_{I,J,N}}{\Delta y^2}\right) + \frac{\Delta_I \psi^{(\delta)}_{I,J,N}}{\Delta x}\frac{\Delta_J^2 u^{(0)}_{I,J,N}}{\Delta y^2}\\ &+ \frac{F_v(I+1,J,N) - F_v(I,J,N)}{\Delta x} + \frac{F_u(I,J+1,N) - F_u(I,J,N)}{\Delta y},\end{aligned} \tag{16}$$

where $u^{(\delta)}_{I,J,N} = -\frac{\Delta_J \psi^{(\delta)}_{I,J,N}}{\Delta y}$, $v^{(\delta)}_{I,J,N} = \frac{\Delta_I \psi^{(\delta)}_{I,J,N}}{\Delta x}$, and $\frac{\Delta_I}{\Delta x}\left(\frac{\Delta_J^2 \psi^{(\delta)}_{I,J,N}}{\Delta^2 y}\right) = 0$. Here, we assume the normal mode solution ($\psi^{(\delta)}$=$\Psi^{(\delta)}(y)e^{ik(x-ct)}$) and substitute this solution into Equation (16):

$$\frac{\Delta_J^2 \Psi^{(\delta)}_{I,J,N}}{\Delta y^2} - \Psi^{(\delta)}_{I,J,N}\left(-k^2\left(\frac{e^{ik\Delta x}-1}{k\Delta x}\right)^2 + \frac{1}{u^{(0)} + c\left(\frac{e^{-ick\Delta t}-1}{ck\Delta t} \Big/ \frac{e^{ik\Delta x}-1}{k\Delta x}\right)}\frac{\Delta_J^2 u^{(0)}_{I,J,N}}{\Delta y^2}\right) = D, \tag{17}$$

where c=c_r+ic_i, i is the imaginary unit, k is the zonal wave number, c_r is the phase speed, kc_i is the growth rate of small disturbances, and

$$D = \frac{e^{-ik(x-ct)}\left(\frac{F_v(I+1,J,N) - F_v(I,J,N)}{\Delta x} + \frac{F_u(I,J+1,N) - F_u(I,J,N)}{\Delta y}\right)}{u^{(0)}k\left(\frac{e^{ik\Delta x}-1}{k\Delta x}\right) + ck\left(\frac{e^{-ick\Delta t}-1}{ck\Delta t}\right)}.$$

When Δt, Δx, and Δy tend to zero, Equation (17) corresponds to the second-order linear inhomogeneous differential equation for the growth rate of the usual barotropic instability waves, because

$$\begin{aligned}\lim_{\Delta t \to 0}\frac{e^{-ick\Delta t}-1}{ck\Delta t} &= -i,\\ \lim_{\Delta x \to 0}\frac{e^{ik\Delta x}-1}{k\Delta x} &= i.\end{aligned}$$

As the solution of an inhomogeneous equation can be expressed by superposing the solutions of a homogeneous equation, thus solving the eigenvalue problem of Equation (17), we now introduce the growth rate of the FPN error at each time-step. That is, the growth rate of the FPN error in the case of barotropic instability is the same as that of barotropic instability waves commencing with small disturbances.

3. Numerical Simulations to Verify the Time Evolution of the FPN Errors

The equations for time evolution of the FPN errors obtained by analyzing the differences between high- and low-precision FPNs were established in the previous section. According to these equations, the residual terms may break the steady state of the geostrophic wind balance. However, barotropic instability may dominate the growth of the FPN error. We validated our time evolution equations by carrying out numerical simulations.

We now describe how we set up the shallow-water model used in the numerical simulations. In the physical constant part of the pre-processing stage, we set up the values of the physical constants, such as the Coriolis parameter. In the grid stage, we defined the spatial grid and temporal increments as Δx, Δy, and Δt respectively. The initialization stage was used to set the initial values of the prognostic variables. In the main body of the simulation, the dynamics core carried out numerical time integration and updated the values of the prognostic variables after each increment. In the output stage, we stored the values of the prognostic variables in a file. Both of the modules were called repeatedly, until the time integration loop was exited. To ensure that numerical stability was maximally neutral, we used the second-order central difference as the spatial difference in an Arakawa C-grid (Arakawa & Lamb, 1977) and carried out the time integration using the RK4 method. This type of scheme is often used to model geophysical fluid dynamics. We used periodic lateral boundary conditions.

To satisfy the condition defined in Equation (3), we employed a quadruple-precision (QP) FPN as $p^{(H)}$. On the other hand, we use DP and SP FPNs as when calculating $p^{(L)}$. The lengths of the significant bits of the QP, DP and SP FPNs were 112, 53, and 24, respectively. Because $p^{(\varepsilon_H)}$ is much smaller than $p^{(\varepsilon_L)}$ ($2^{53-112}\approx10^{-18}\ll1$, $2^{24-112}\approx10^{-27}\ll1$), it is clear that $p^{(\delta)}$ is almost the same as $p^{(\varepsilon_L)}$.

3.1. Geostrophic Wind Balance Experiment

Following Equation (10), the initial values that satisfy the geostrophic wind balance are:

$$\begin{aligned} u_{I,J,N} &= -\frac{1}{f}\frac{\Delta_J \phi_{I,J,N}}{\Delta y}, \\ v_{I,J,N} &= 0, \\ \phi_{I,J,N} &= \Phi - fU\Delta y \tanh\left(\left|J-\frac{J_{max}}{2}\right|-\frac{J_{max}}{4}\right), \end{aligned} \tag{18}$$

where U and Φ are the given values of the zonal velocity and geopotential, respectively, and J_{max} is the number of grids in the y-direction. The distribution of the zonal velocity contains Bickley jets ($u = U/\cosh^2 y$) in the north and south regions, as shown in Figure 2. In this experiment, we set U=10^1, Φ=10^5, and f=10^{-4}, which are typical values for the velocity, geopotential, and Coriolis force in the mid-latitude troposphere, respectively. The grid interval (Δy) is set to 10^4. For this grid distance, the time interval (Δt) should be 10^1 to satisfy the Courant-Friedrichs-Lewy condition for the phase speed of the surface gravity waves ($\sqrt{\Phi}$). The number of grids in the y-direction (J_{max}) was 100.

As the initial condition is given, let us simplify Equation (13) using the scale analysis described in the geostrophic wind balance experiment. We assumed that the magnitude of $p^{(\delta)}$, defined in terms of the difference operator, is equal to the value calculated without the difference operator. The statistical average of the stochastic variable, $p^{(\delta)}$, is zero because prognostic variables barely change during a geostrophic wind balance experiment. This means that the dynamic ranges of prognostic variables remain the same during simulation. That is,

$$O\left(\Delta_N p^{(\delta)}_{I,J,N}\right) = O\left(p^{(\delta)}_{I,J,N+1}\right) - O\left(p^{(\delta)}_{I,J,N}\right) = O\left(p^{(\delta)}_{I,J,N}\right).$$

Therefore, the second and third terms of the right-hand side are much smaller than the first term:

$$O\left(\phi^{(0)}_{I,J+1,N}\frac{\Delta_J^2 v^{(\delta)}_{I,J,N}}{\Delta y^2}\right) \gg O\left(f u^{(0)}_{I,J,N}\frac{\Delta_J v^{(\delta)}_{I,J,N}}{\Delta y}\right) \gg O\left(f^2 v^{(\delta)}_{I,J,N}\right).$$

We expect the solution of this equation to behave in a wave-like manner. We now consider the following question: how large are the residual terms that behave like external forcing? The function *RN* in Equation (9) is the key to determining the magnitude of the residual terms. As the magnitude of the new addition error caused by rounding at each step can also be estimated from Figure 1, the magnitude of *RN*

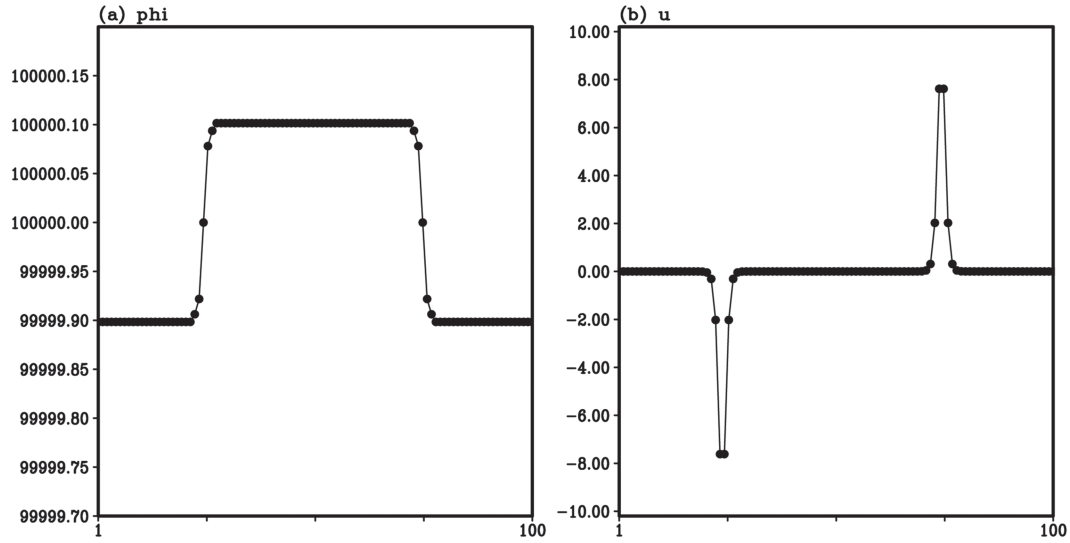

Figure 2. The initial distribution of (a) ϕ and (b) u in the geostrophic wind balance experiment. The ordinate indicates the value of each variable. The abscissa is the grid coordinate in the y-direction.

is greater than or equal to $O\left(p^{(\epsilon)}_{I,J,N}\right)$. The non-linear terms, and the truncation error in terms of rounding error, are smaller than RN. Hence, these terms are eliminated by the rounding operation. That is, Equation (9) can be written as

$$F_p(I,J,N) = O\left(RN\left(p^{(\epsilon_L)}_{I,J,N}\right)\right).$$

In Equation (1), the loss of significant digits indicates that $p^{(0)}$ becomes relatively small but the magnitude of $p^{(\epsilon)}$ is not changed. Although this error is important when performing numerical simulations, we can ignore the error in Equation (8) because we subtract $p^{(0)}$ from p. That is, this error does not act as an external forcing. In other words, the loss of significant errors cannot be detected using our method. Therefore, numerical checking tools, including the Control of Accuracy and Debugging for Numerical Applications (CADNA) and Verificarlo, become important when exploring the error. However, the loss of trailing digits indicates that a very small number is added to p but p is not changed because its magnitude is smaller than $p^{(\epsilon)}$. In Equation (8), this error is thus now unimportant because of its low magnitude. Therefore, the rounding error dominates the RN function. We then update the values of the variables in the shallow-water model in Equation (5), the largest term of which is the time difference term. The magnitude of the rounding error incurred when solving the equations governing the dynamical process can be written as

$$O\left(RN\left(p^{(\epsilon_L)}_{I,J,N}\right)\right) = O\left(\frac{p^{(\epsilon_L)}_{I,J,N}}{\Delta t}\right).$$

Therefore, the fourth and fifth terms of the right-hand side of Equation (13) are much smaller than the sixth term:

$$O\left(\frac{F_\phi(I,J,N)}{\Delta y}\right) \gg O(fF_u(I,J,N)) \gg O\left(\frac{F_v(I,J,N)}{\Delta t}\right).$$

We can now simplify Equation (13) to

$$\frac{\Delta_N^2 v^{(\delta)}_{I,J,N}}{\Delta t^2} = \phi^{(0)}_{I,J+1,N}\frac{\Delta_J^2 v^{(\delta)}_{I,J,N}}{\Delta y^2} + \frac{F_\phi(I,J+1,N) - F_\phi(I,J,N)}{\Delta y}. \tag{19}$$

We use the same method as used for $v^{(\delta)}$ to obtain the equations for $\phi^{(\delta)}$:

$$\frac{\Delta_N^2 \phi^{(\delta)}_{I,J,N}}{\Delta t^2} = \phi^{(0)}_{I,J,N}\frac{\Delta_J^2 \phi^{(\delta)}_{I,J,N}}{\Delta y^2} - \frac{F_\phi(I,J,N+1) - F_\phi(I,J,N)}{\Delta t}, \tag{20}$$

while the form of the equation for the time evolution of $u^{(\delta)}$ differs from these two equations. The first equation of Equation (11) implies that the time evolution of $u^{(\delta)}$ is proportional to $v^{(\delta)}$. The second term on the right-hand side is much smaller than the first term, which means that we can rewrite the equation as

$$\frac{\Delta_N u^{(\delta)}_{I,J,N}}{\Delta t} = -v^{(\delta)}_{I,J,N}\frac{\Delta_J u^{(0)}_{I,J,N}}{\Delta y} + F_u(I,J,N). \tag{21}$$

Note that both of the terms in Equation (21) may act as stochastic forcing for $u^{(\delta)}$.

We can visualize the time evolution of the FPN error easily by introducing the root-mean-square (RMS) of the FPN error of the spatial distribution:

$$RMS\left(p^{(\delta)}_N\right) = \sqrt{\frac{1}{I_{max}}\Sigma^{I_{max}}_{I=1}\frac{1}{J_{max}}\Sigma^{J_{max}}_{J=1}\left(p^{(\delta)}_{I,J,N}\right)^2}, \tag{22}$$

where I_{max} is the number of grids in the x-direction. Note that $p^{(\delta)}$ is forced by the stochastic variable in the residual terms. Compared to other rounding methods, as banker's rounding is bias-free for all real numbers, we expect that equal numbers of values are rounded up and down by this rounding operation. That is, this stochastic forcing operation corresponds to a random walk. The average time-integrated value of a prognostic variable that is undergoing a random walk is zero, and its variance increases proportionally to the size of the time-step (N). Therefore, the RMS of the FPN error increases proportionally to the square root of the time-step, as follows:

$$RMS\left(p^{(\delta)}_N\right) = \sqrt{RMS\left(p^{(\delta)}_0\right)^2 + \gamma_p^2 N},$$

where γ_p is the random forcing coefficient of $p^{(\delta)}$. For the initial condition, we can estimate the RMS of $\phi^{(\delta)}$ from Equation (22). Note that $u^{(\delta)}$ is calculated using Equation (18) and $v^{(\delta)}$ is equal to zero. That is,

$$\begin{aligned} O\left(RMS\left(\phi^{(\delta)}_0\right)\right) &= O(\phi E_L) = O(10^5 E_L), \\ O\left(RMS\left(u^{(\delta)}_0\right)\right) &= O\left(\frac{1\Delta_J\phi^{(\delta)}}{f\ \ \Delta y}\right) = O(10^5 E_L). \end{aligned}$$

The magnitude of the random forcing coefficient corresponds to that of the residual term as

$$\begin{aligned} O\left(\gamma_\phi\right) &= O\left(F_\phi(I,J,N)\Delta t\right) = O(\phi E_L) = O(10^5 E_L), \\ O\left(\gamma_v\right) &= O\left(F_\phi(I,J,N)\frac{\Delta t^2}{\Delta y}\right) = O\left(\phi E_L\frac{\Delta t}{\Delta y}\right) = O(10^2 E_L), \end{aligned}$$

which we estimate using Equations (19) and (20). The time evolution of $u^{(\delta)}$ in Equation (21) has two random forcing terms:

$$O(\gamma_u) = O\left(v^{(\delta)}_{I,J,N}\frac{\Delta_J u^{(0)}_{I,J,N}}{\Delta y}\Delta t\right) + O(F_u(I,J,N)\Delta t) = O\left(10^{-3}v^{(\delta)}_{I,J,N}\right) + O\left(10^1 E_L\right), \tag{23}$$

where we assume that $O\left(\frac{\Delta_J u^{(0)}_{I,J,N}}{\Delta y}\right) = 10^{-4}$, as shown in Figure 2b. These two random forcing terms are mutually exclusive due to the rounding operation. The magnitude of the random forcing coefficient is $10^1 E_L$ when the magnitude of $v^{(\delta)}$ is less than $10^4 E_L$. That is, the ideal equations of the FPN error are:

$$\begin{aligned} RMS\left(\phi^{(\delta)}_0\right)_{ideal} &= O\left(10^5 E_L\right)\cdot\alpha_\phi\sqrt{1+N}, \\ RMS\left(u^{(\delta)}_0\right)_{ideal} &= O(10^5 E_L)\cdot\alpha_u\sqrt{1+10^{-12}\alpha_v^2 N^2 + 10^{-8}N}, \\ RMS\left(v^{(\delta)}_0\right)_{ideal} &= O(10^2 E_L)\cdot\alpha_v\sqrt{N}, \end{aligned}$$

where α_p is the fitting parameter for each prognostic variable, which makes it easier to understand the relationship between the ideal equations and the experimental results. In this case, we set α_ϕ=0.5, α_u=0.5, and α_v=1.5, respectively. According to these equations, the time evolution of $\phi^{(\delta)}$ and $v^{(\delta)}$ is proportional to $\sqrt{N}$. Given the selectiveness of random forcing terms in the second equation, the third term in the square-root is

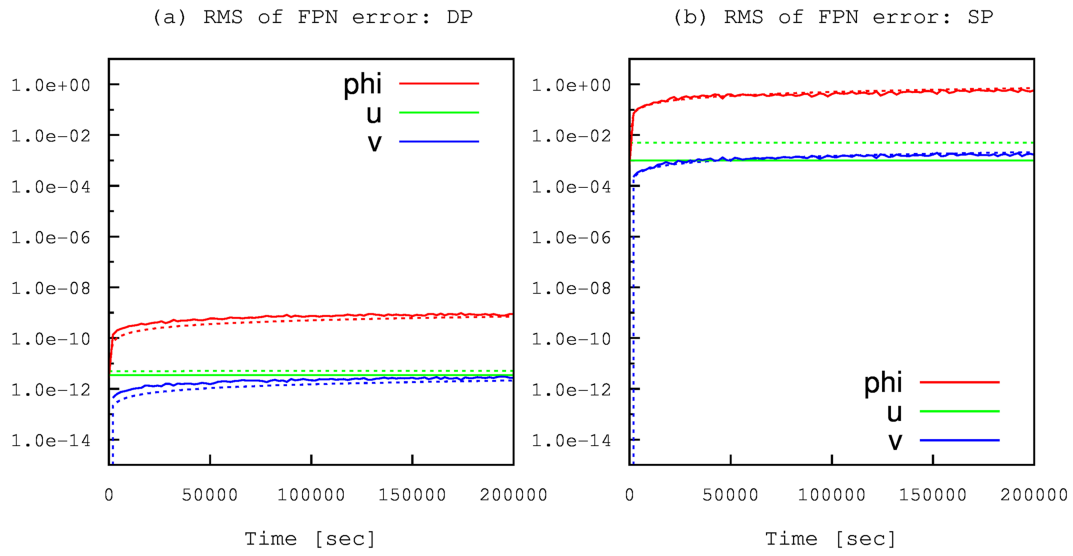

Figure 3. (a) The time evolution of the root-mean-square of the floating point number (FPN) errors from the geostrophic wind balance experiments using double-precision (DP) FPNs. The output is recorded every 2,000 seconds. The solid (dotted) lines are the experimental (ideal) values of the FPN errors. The ideal equation is described in detail in the text. (b) The same as (a), except for single-precision (SP) FPNs.

ignored when N is smaller than 10^4. On the other hand, both the second and third terms in the square-root are much smaller than 1 when N is smaller than 10^6. In that case, the time evolution of $u^{(\delta)}$ is almost constant.

Figure 3 shows the time evolution of FPN errors obtained in the geostrophic wind balance experiments up to 200,000 seconds (N=2×10^4). Figure 3a is the result obtained using DP FPNs; $O(E_L) = 10^{-16}$. In contrast, Figure 3b is the result obtained using SP FPNs; $O(E_L) = 10^{-7}$. The experimental results for each prognostic variable clearly follow the ideal lines. We emphasize that the parameters of the ideal lines, except those of the fitting factors, are defined before execution of numerical simulations. Thus, it is reasonable to assume that

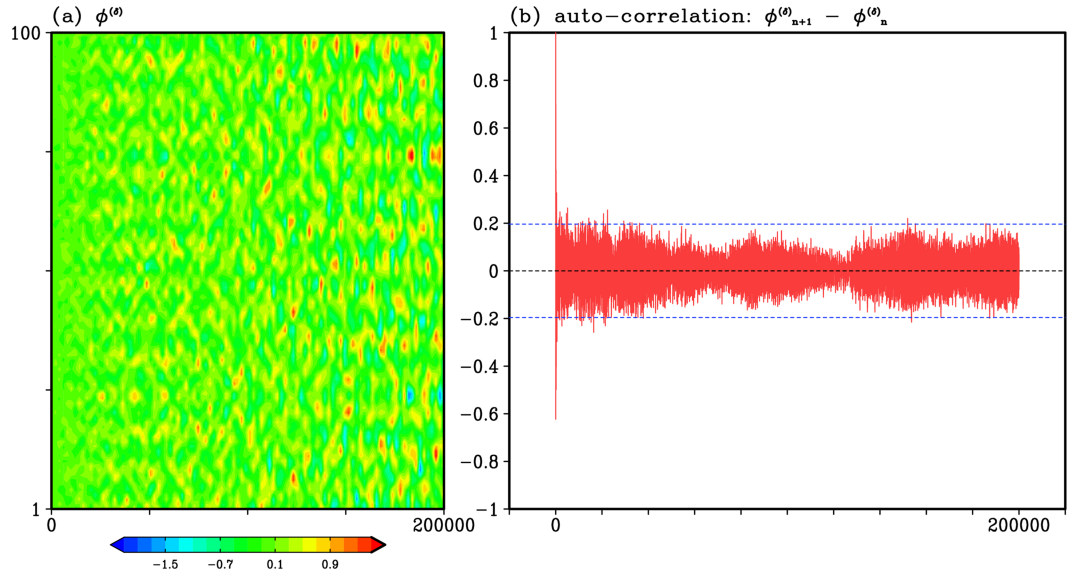

Figure 4. (a) The difference in geopotential ($\phi^{(\delta)}$) between the quadruple-precision (QP) FPN and SP FPN analyses (SP - QP) in the geostrophic wind balance experiment. The abscissa is integrated time at 2,000 second intervals; the ordinate is the y-grid number. (b) The auto-correlation time series of the one-step difference of $\phi^{(\delta)}$ in the geostrophic wind balance experiment. The abscissa is delayed time; the ordinate shows correlation coefficients. The blue dashed line indicates the 5% significance level (100 samples).

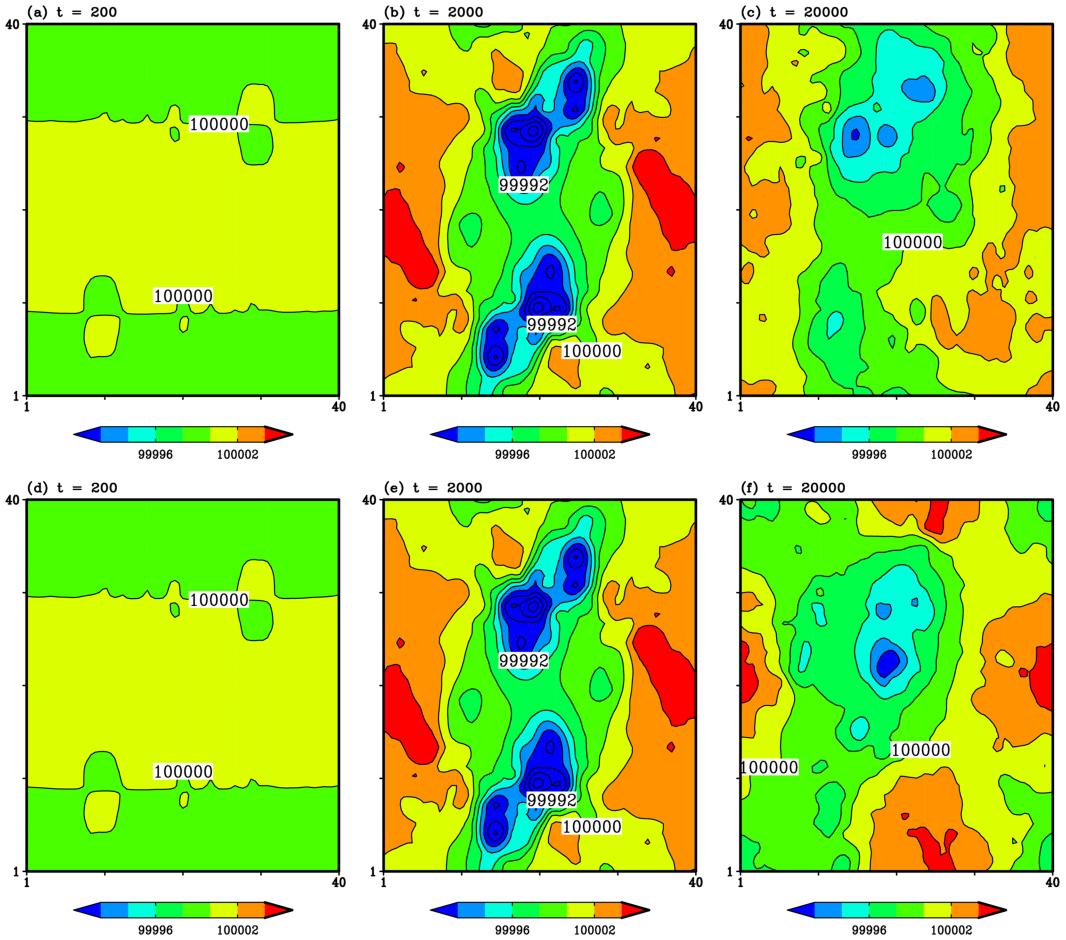

Figure 5. The spatial pattern of geopotential obtained from the barotropic instability experiment after (a) 200 seconds, (b) 2,000 seconds, and (c) 20,000 seconds (QP FPNs). (d), (e), and (f) are as (a), (b), and (c), except that DP FPNs were used.

the time evolution equations of $p^{(\delta)}$ are captured by Equation (11) in terms of the geostrophic wind balance, and that the residual terms are stochastic forcing terms.

To confirm the behavior of FPN errors in the numerical simulations, we display the difference between the QP and SP FPN results. Figure 4a shows the geopotential difference between the QP and SP results in the geostrophic wind balance experiment. The error in this experiment develops with the square root of time, which corresponds to what is shown in Figure 3b. Figure 4b shows the auto-correlation coefficient of the one-step difference of $\phi^{(\delta)}$; this reflects auto-correlation of the sum of FPN errors in the one-step operation. The auto-correlation peaks early and reduces rapidly toward zero during simulation; this supports the suggestion that the FPN errors exhibit stochastic behavior.

3.2. Barotropic Instability Experiment

We now discuss the barotropic instability state, which is a representative unstable state of the shallow-water equations. We define the initial conditions in terms of the small disturbances that occur when the system is in the geostrophic wind balance state. The initial values that satisfy this condition are:

$$\begin{aligned} u_{I,J,N} &= -\frac{1\Delta_J \phi_{I,J,N}}{f \quad \Delta y}, \\ v_{I,J,N} &= V\sin\left(\frac{2\pi k}{L_x}\left(I-\frac{I_{max}}{2}\right)\right) \quad , \\ \phi_{I,J,N} &= \Phi - fU\Delta y \tanh\left(\left|J-\frac{J_{max}}{2}\right|-\frac{J_{max}}{4}\right), \end{aligned}$$

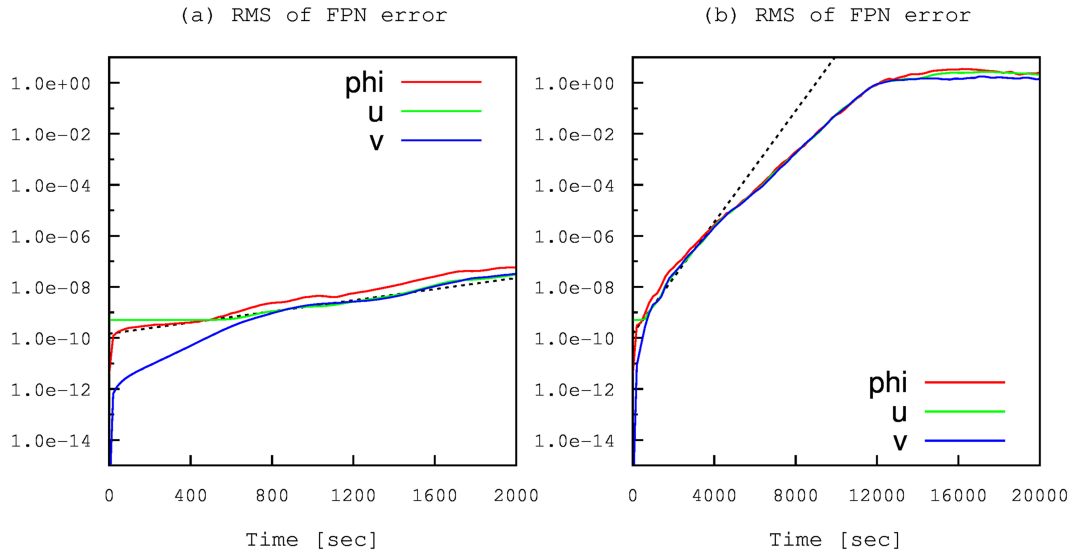

Figure 6. (a) The time evolution of the FPN error in the barotropic instability experiment, up to 2,000 seconds. The output interval is 20 seconds. The solid lines indicate the experimental values of the FPN error. The dotted black line indicates the theoretical growth rate of the barotropic instability wave. (b) As (a), except time integration proceeds to 20,000 seconds and the output interval is 200 seconds.

where $L_x=I_{max}\Delta x$ is the zonal length. The grid and time intervals are $\Delta x=\Delta y=10^2$ (m) and $\Delta t=10^{-1}$ (s), respectively. There are $I_{max}=J_{max}=40$ grid points in the x- and y-directions, respectively. We can specify small disturbances in the meridional velocity when $V=10^{-2}$ by using the zonal wave number, k, the value of which is between 1 and half the number of zonal grids, 20.

Figure 5 shows the spatial patterns of the geopotential obtained in the barotropic instability experiment using QP and DP FPNs. As in the geostrophic wind balance experiment, the geopotential and zonal velocity initially exhibited Bickley jet patterns, as shown in Figure 2. After 200 seconds, small disturbances began to develop due to barotropic instability, although the spatial patterns of the geopotential were still similar to the initial conditions shown in Figures 5a and d. The disturbances then developed into a clear meandering jet pattern after 2,000 seconds. This implies that the disturbances developed due to barotropic instability, as shown in Figures 5b and e. The QP and DP FPNs remained very similar until this point. By the time that 20,000 seconds had passed, the spatial distributions of the geopotential generated by the QP and DP FPNs were completely different, as shown in Figures 5c and f, which implies that the magnitude of the FPN error is comparable to that of the barotropic instability waves.

The growth rate of the barotropic instability wave of the Bickley jet pattern has already been investigated in Kuo (1973). Because barotropic instability waves grow exponentially, we can write the RMS of the FPN error as

$$RMS\left(p_N^{(\delta)}\right) = RMS\left(p_0^{(\delta)}\right)\exp\left(\frac{2\pi\sigma U}{L_y}t\right), \tag{24}$$

where $L_y=J_{max}\Delta y$ is the meridional length and $\sigma\approx0.16$ is the maximum growth rate on the f-plane (Kuo, 1973). Note that the initial value of this exponential function, $p_0^{(\delta)}$, is not the initial condition; $p^{(\delta)}$ when the error growth of barotropic instability is larger than that of the square root of time. This is due to the previously discussed properties of Equation (17).

Figure 6 shows the time evolution of the FPN error obtained from the barotropic instability experiments. Until 400 seconds had passed (Figure 6a), the RMS of the FPN error in the value of the geopotential was well represented by stochastic forcing, as shown in the analysis of the geostrophic wind balance state. The RMS of the FPN error grew proportionally to the barotropic instability. In this case, we set $RMS\left(p_0^{(\delta)}\right)=10^{-9}$ in Equation ((24)). Up to 4,000 seconds, the FPN errors were close to the theoretical growth rate of the barotropic instability waves, as shown in Figure 6b. After that, the growth rate decreased slightly, and continued to decrease up to 12,000 seconds, at which time the error almost stopped developing. Saturation of FPN error is attributable to the existence of a lateral boundary. At this point, the FPN errors were similar to the value of

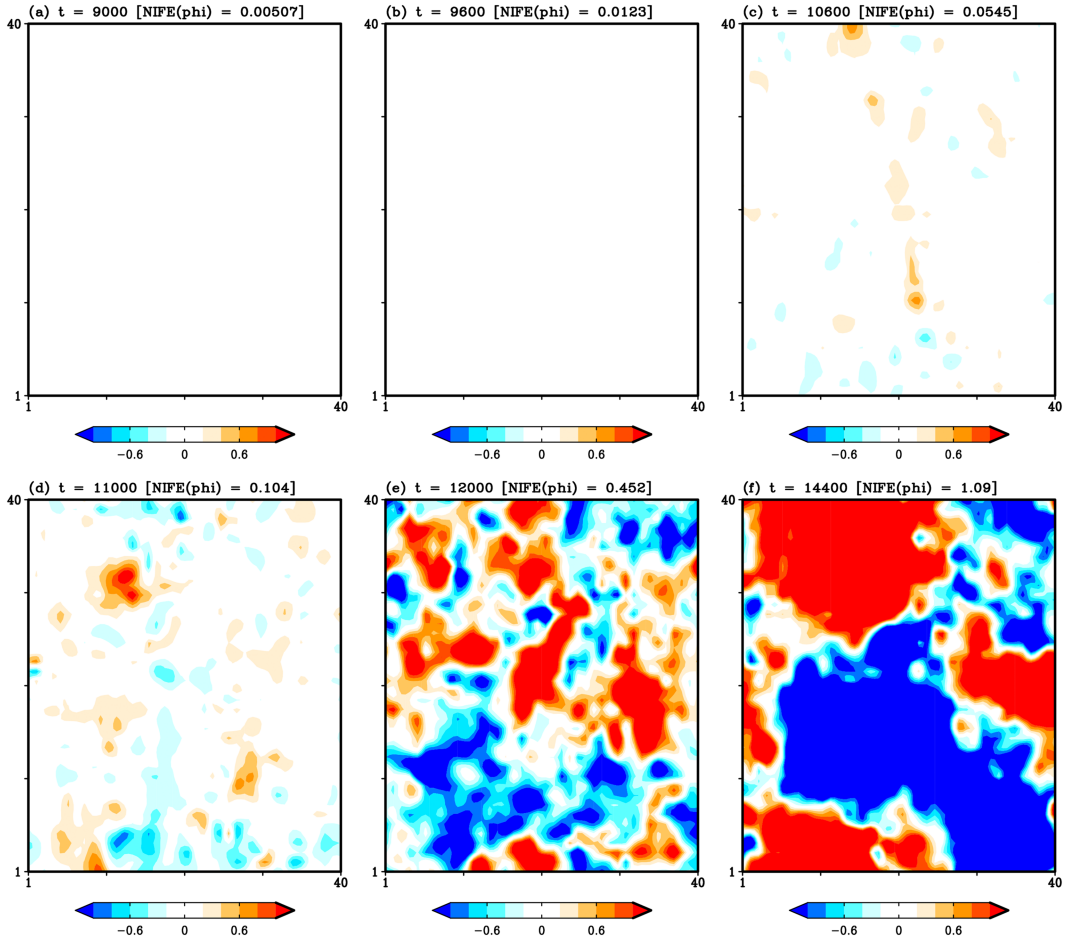

Figure 7. Using DP FPNs, the FPN error patterns of the geopotentials obtained from the barotropic instability experiment after (a) 9,000 seconds, (b) 9,600 seconds, (c) 10,600 seconds, (d) 11,000 seconds, (e) 12,000 seconds, and(f) 14,400 seconds. The shading scale is shown at the bottom of each panel and the geopotential normalized index of FPN error (NIFE) is shown at the top of each panel.

the standard deviation of the waves ($RMS(p^{(\delta)})\approx 1$). This indicates that the results were far from the true values, as shown in Figures 5c and f. Although the one-shot result obtained using the low-precision FPN does not yield an acceptable solution to the initial-value problem, multiple results obtained using FPNs varying precision are meaningful. These can be considered as results of ensemble experiments because the FPN error acts like stochastic forcing in the governing equations, as shown in the geostrophic wind balance experiment. Düben and Dolaptchiev (2015) suggested that the numerical errors induced by low-precision FPNs can be used for ensemble forecasting. Numerical weather simulations using FPNs of various precisions may be useful as simple ensemble forecasting tools in the future.

Although we cannot directly compare the FPN errors of the different variables because their units differ, we can determine which FPN errors are the most important. To explore the relative sizes of FPN errors of prognostic variables, it is necessary to normalize the FPN error units. For example, we can evaluate the physical accuracy of simulations performed using low-precision FPNs by defining the normalized index of the FPN error (NIFE) as

$$NIFE(p_N) = \frac{RMS\left(p_N^{(\delta)}\right)}{STD(p_N)},$$

where STD is the standard deviation of the spatial distribution of a model variable. The maximum NIFE of model variables indicates the extent to which FPN errors influence the results of numerical weather/climate simulations. When the NIFE is close to 1, we can conclude that the magnitude of the FPN error

introducedvia stochastic forcing is comparable to that of physical dispersion of the prognostic variable. Figure 7 shows the pattern of DP FPN geopotential error obtained in the barotropic instability experiment. The maximum/minimum values of the color scale correspond to 10% of the geopotential standard deviation; the color interval is 2% of the standard deviation. Therefore, shaded regions appear when the NIFE is larger than 0.05 (Figure 7c, d, e, f). We found similar patterns for the FPN errors of zonal and meridional winds. Thus, the index yields objective information on FPN errors. For example, we can decide that the FPN error can be neglected in a numerical simulation if the average NIFE for each prognostic variable is sufficiently small.

4. Concluding Remarks and Discussion

In this study, we theoretically investigated the impact of numerical errors due to the use of FPNs when simulating the shallow-water model. Time evaluation equations of the numerical errors caused by FPNs (FPN errors) were obtained by analyzing the difference between the values of the model variables when using high- and low-precision FPNs, under the presupposition that model variables can be written as the linear sum of the true value and the numerical error. We assume that the theoretical essence of the time evolution of FPN errors studied here does not depend on numerical choices, such as the grid arrangement or time-step scheme. This assumption is based on the fact that the FPN errors are caused by differences in the results obtained using the high- versus low-precision FPNs. The numerical errors (excluding the FPN errors) attributable to the schemes are prevented by our subduction. We use a C-grid arrangement and the RK4 method for time integration, to retain near-neutral numerical stability in the simulations. Thus, the FPN error does not differ greatly according to the type of discretization employed. We derived Equation (8) based on these considerations. We carried out numerical experiments using the shallow-water model to confirm our theoretical results in two cases; geostrophic wind balance (steady state) and barotropic instability (unstable state). We evaluated the FPN errors in the numerical simulations by calculating the RMS of the FPN error, which is defined by Equation (22). The results of this study are summarized as follows.

First, we considered the theoretical time development of the FPN error of the geostrophic wind balance, defined by Equation (10). We derived Equation (11) for the time evolution of the FPN error. The solution of this equation implies that the FPN error oscillates with a gradually increasing amplitude, as in Equation (12). According to our numerical simulation of the geostrophic wind balance, the RMS of the FPN error increases with the magnitude of the stochastic forcing, as shown in Figure 3. We also investigated the theoretical growth of the FPN error in a barotropic instability state, in which case we add a small disturbance to the meridional velocity of the geostrophic wind balance state. The vorticity of the FPN error is calculated using Equation (15). The solution of this equation implies that the FPN error evolves in a similar manner to the barotropic instability waves defined in Equation (17). According to our numerical simulation of the barotropic instability, the RMS of the FPN error initially increases with the stochastic forcing, as in the case of the geostrophic wind balance. The RMS of the FPN error then increases exponentially, like the barotropic instability waves as shown in Figure 6. This indicates that the magnitude of the FPN error may be larger than the disturbances due to physical instability over short-time scales because the stochastic forcing of the FPN error is proportional to the square-root of the elapsed time. Therefore, researchers should exercise caution when using low-precision FPNs to conduct numerical simulations to solve the initial-value problem such as when carrying out short-term weather forecasting. To display the magnitude of FPN errors both easily and objectively, we introduce the NIFE, which is the ratio of the magnitude of the FPN error to the physical dispersion of the prognostic variable.

Despite using the shallow-water model, which is a very primitive atmospheric model, the results of this work could help researchers to improve their numerical weather/climate models. This is because the dynamics of the shallow-water model are fundamental to current numerical weather/climate models. To extend the theory of FPN error to real weather/climate models, we should investigate the impact of the numerical errors caused by using low-precision FPNs in more realistic atmospheric models. In this study, the time evolution of p is the linear sum of the time integration of the true value and the numerical error, where the error is much smaller than the true value. It is important that the difference between high- and low-precision FPNs after time evolution cancels $p^{(0)}$, and to extract the difference between $p^{(\epsilon_H)}$ and $p^{(\epsilon_L)}$. However, many thresholds are employed in real weather/climate models that can qualitatively change the simulation results when the values of prognostic variables vary slightly (e.g., the convective parameterization scheme is active or non-active with small adjustment int the prognostic variable). In these cases, the difference between the

high- and low-precision FPNs no longer reflects the FPN error term. We will aim to confirm whether the models remain physically accurate, and whether the use of low-precision FPNs improves the computational performance of simulations of realistic atmospheric models.

Acknowledgments
This work was supported by CREST and AIP, Japan Science and Technology Agency (Grant Number: JPMJCR1312, JPMJCR19U2). The source codes and experimental configurations used in this study are archived in the Open Science Framework (https://osf.io/3vcgx). We thank two anonymous reviewers and an associate editor for their highly constructive comments.

References

Arakawa, A., & Lamb, V. R. (1977). Computational design of the basic dynamical processes of the UCLA general circulation model. *Methods in Computational Physics*, *17*, 173–265.

Düben, P. D., & Dolaptchiev, S. I. (2015). Rounding errors may be beneficial for simulations of atmosphric flow: Results from the forced 1D Burgers equation. *Theoretical and Computational Fluid Dynamics*, *29*, 311–328. https://doi.org/10.1007/s00162-015-0355-8

Düben, P. D., & Palmer, T. N. (2014). Benchmark tests for numerical weather forecasts on inexact hardware. *Monthly Weather Review*, *142*, 3809–3829. https://doi.org/10.1175/MWR-D-14-00110.1

Gan, L., Fu, H., Luk, W., Yang, C., Xue, W., Huang, X., et al. (2015). Solving the global atmospheric equations through heterogeneous reconfigurable platforms. *ACM Transactions on Reconfigurable Technology and Systems*, *8*, 1–16. https://doi.org/10.1145/2629581

Kuo, H. L. (1973). Dynamics of quasigeostrophic flows and instability theory. *Advances in Applied Mechanics*, *13*, 247–330.

Lingamneni, A., Enz, C., Nagel, J. L., Palem, K. V., & Piguet, C. (2011). Energy parsimonious circuit design through probabilistic pruning. In *Proceedings of Design, Automation and Test in Europe Conference Exhibition (DATE)* (pp. 1–6). Grenoble: IEEE. https://doi.org/10.1109/DATE.2011.5763130

Nakano, M., Yashiro, H., Kodama, C., & Tomita, H. (2018). Single Precision in the Dynamical Core of a Nonhydrostatic Global Atmospheric Model: Evaluation Using a Baroclinic Wave Test Case. *Monthly Weather Review*, *146*, 409–416. https://doi.org/10.1175/MWR-D-17-0257.1

Shalf, J. (2010). Exascale Computing Trends: Adjusting to the "New Normal" for Computer Architecture. *Computing in Science and Engineering*, *15*, 16–26. https://doi.org/10.1109/MCSE.2013.95

Shapiro, M., Shukla, J., Brunet, G., Nobre, C., Béland, M., Dole, R., et al. (2010). An earth-system prediction initiative for the twenty-first century. *Bulletin of the American Meteorological Society*, *91*, 1377–1388. https://doi.org/10.1175/2010BAMS2944.1

Shukla, J., Palmer, T., Hagedorn, R., Hoskins, B., Kinter, J., Marotzke, J., et al. (2010). Toward a new generation of world climate research and computing facilities. *Bulletin of the American Meteorological Society*, *91*, 1407–1412. https://doi.org/10.1175/2010BAMS2900.1

Váňa, F., Düben, P., Lang, S., Palmer, T., Leutbecher, M., Salmond, D., & Carver, G. (2017). Single Precision in Weather Forecasting Models: An Evaluation with the IFS. *Monthly Weather Review*, *145*, 495–502. https://doi.org/10.1175/MWR-D-16-0228.1

Yamagishi, T., & Matsumura, Y. (2016). GPU acceleration of a non-hydrostatic ocean model with a multigrid Poisson/Helmholtz solver. *Procedia Computer Science*, *80*, 1658–1669. https://doi.org/10.1016/j.procs.2016.05.502