Computing the ecRad radiation scheme with half-precision arithmetic

Anton Pershin[1], Matthew Chantry[2], Peter Dominik Dueben[2], Robin J Hogan[1,2], and Tim Palmer[1]

[1]Oceanic and Planetary Physics, University of Oxford
[2]European Centre for Medium-Range Weather Forecasts

September 30, 2023

Abstract

Numerical simulations of weather and climate models are conventionally carried out using double-precision floating-point numbers throughout the vast majority of the code. At the same time, the urgent need of high-resolution forecasts given limited computational resources encourages development of much more efficient numerical codes. A number of recent studies has suggested the use of reduced numerical precision, including half-precision floating-point numbers increasingly supported by hardware, as a promising avenue. In this paper, the possibility of using half-precision calculations in the radiation scheme ecRad operationally used in the ECMWF's Integrated Forecasting System (IFS). By deliberately mixing half-, single- and double-precision variables, we develop a mixed-precision version of the Tripleclouds solver, the most computationally demanding part of the radiation scheme, where reduced-precision calculations are emulated by a Fortran software rpe. By employing two tools that estimate the dynamic range of model parameters and identify problematic areas of the model code using ensemble statistics, the code variables were assigned particular precision levels.

It is demonstrated that heating rates computed by the mixed-precision code are reasonably close to those produced by the double-precision code. Moreover, it is shown that using the mixed-precision ecRad in OpenIFS has a very limited impact on the accuracy of a medium-range forecast in comparison to the original double-precision configuration. These results imply that mixed-precision arithmetic could successfully be used to accelerate the radiation scheme ecRad and, possibly, other parametrization schemes used in weather and climate models without harming the forecast accuracy.

Computing the ecRad radiation scheme with half-precision arithmetic

Anton Pershin[1], Matthew Chantry[2], Peter D. Düben[2], Robin J. Hogan[2], Tim Palmer[1]

[1]Atmospheric, Oceanic and Planetary Physics, University of Oxford, Oxford, United Kingdom
[2]European Centre for Medium-Range Weather Forecasts, Reading, United Kingdom

Key Points:

- Half-precision floating-point numbers can be used to accelerate the radiation transfer computation
- Mixed-precision approach is employed to yield sufficiently accurate results while keeping most of the variables in half precision
- Ensemble-based rounding error analysis can successfully identify parts of the code suffering from the reduction of precision

Corresponding author: Anton Pershin, anton.pershin@physics.ox.ac.uk

Abstract

Numerical simulations of weather and climate models are conventionally carried out using double-precision floating-point numbers throughout the vast majority of the code. At the same time, the urgent need of high-resolution forecasts given limited computational resources encourages development of much more efficient numerical codes. A number of recent studies has suggested the use of reduced numerical precision, including half-precision floating-point numbers increasingly supported by hardware, as a promising avenue. In this paper, the possibility of using half-precision calculations in the radiation scheme ecRad operationally used in the ECMWF's Integrated Forecasting System (IFS). By deliberately mixing half-, single- and double-precision variables, we develop a mixed-precision version of the Tripleclouds solver, the most computationally demanding part of the radiation scheme, where reduced-precision calculations are emulated by a Fortran software rpe. By employing two tools that estimate the dynamic range of model parameters and identify problematic areas of the model code using ensemble statistics, the code variables were assigned particular precision levels. It is demonstrated that heating rates computed by the mixed-precision code are reasonably close to those produced by the double-precision code. Moreover, it is shown that using the mixed-precision ecRad in OpenIFS has a very limited impact on the accuracy of a medium-range forecast in comparison to the original double-precision configuration. These results imply that mixed-precision arithmetic could successfully be used to accelerate the radiation scheme ecRad and, possibly, other parametrization schemes used in weather and climate models without harming the forecast accuracy.

Plain Language Summary

Weather and climate forecasts can be made more realistic by using more complex models of physical processes or by resolving finer scales. Any of these approaches requires a significant increase of computational power. Recent studies have demonstrated that the accuracy may be improved at no computational cost by reducing numerical precision which defines the accuracy of individual arithmetic operations. In particular, it looks attractive to replace double-precision numbers (a single number is stored in 64 bits) with half-precision ones (a single number is stored in 16 bits) whose support by new hardware is increasingly being expanded. This papers describe how this can be done for the radiation scheme ecRad operationally used in ECMWF's Integrated Forecasting System (IFS). By estimating the spread of values of code variables and identifying problematic code parts using ensemble statistics, all the variables were assigned half-, single- or double-precision levels. The resulting mixed-precision version of ecRad is shown to produce the output which barely differs from the original one even if the majority of variables are stored as half-precision numbers. Moreover, replacing the original radiation scheme in the full forecasting model with the mixed-precision one has negligible effect on the accuracy of the 10-day forecast.

1 Introduction

Weather and climate prediction simulations are known to be computationally demanding and can require petascale computing facilities and large data storage to produce high-resolution forecasts. Further progress in their quality and realism, often attributed to the use of much higher resolution and model complexity, is largely limited by the available computational resources. To overcome this problem, a number of improvements of computational efficiency has been proposed, from porting the code to heterogeneous hardware architectures to replacing some of the model components with machine-learning surrogate models (Bauer et al., 2021). Among these suggestions, one of the most promising directions is reduction of the numerical precision of variables used throughout the code (T. N. Palmer, 2014; T. Palmer, 2015). Traditionally, most of the variables are represented by double-precision floating-point numbers. Given their extremely wide dynamic range and tiny relative error, one may find that computations performed with respect to this format are overly accurate and turn the code variables to single- or half-precision floating-point numbers, bfloat16 (Kalamkar et al., 2019) or posits (Gustafson & Yonemoto, 2017; Klöwer et al., 2019) without a notable loss of accuracy thereby significantly accelerating the simulation. For example, replacing double-precision variables with half-precision ones can theoretically lead to 4x memory saving and computation speedup. An important assumption behind the successful use of precision reduction is that induced rounding errors must not exceed the level of uncertainties associated with initial conditions and physical parametrizations of the model (T. Palmer, 2015).

Reduced precision has already been used to improve the performance of numerical codes in linear algebra algorithms (Baboulin et al., 2009; Abdelfattah et al., 2021) and machine learning (Gupta et al., 2015) where the working precision of neural networks in both training and inference modes can be as low as one bit (Hubara et al., 2017). In the realm of weather and climate modelling, reduction from double to single precision has proved to be widely successful. Performing the vast majority of operations in single precision in ECMWF's Integrated Forecasting System (IFS) led to about 40% reduction of the run time with no forecasting skill degradation in comparison to double precision (Düben & Palmer, 2014; Váňa et al., 2017) which enabled higher vertical resolution in operational forecasts making them substantially more accurate despite the same computational time (Rodwell et al., 2021; Lang et al., n.d.). A similar runtime reduction was observed in the MeteoSwiss's COSMO forecasting system where single precision was introduced in all parts of the code except for the radiation scheme and is now used operationally (Rüdisühli et al., 2013). A similar mixed-precision approach efficiently combining the use of single- and double-precision variables was employed to speed up the linear solver in the dynamical core of the Met Office's Unified Model (Maynard & Walters, 2019) and the ocean model NEMO (Tintó Prims et al., 2019).

More radical precision reduction in weather and climate applications, typically implying the use of 16-bits floating-point numbers, may be a non-trivial task requiring a deeper understanding of how the rounding errors spread through the code. It is additionally complicated by the fact that there are several alternative formats of floating-point numbers (see Klöwer et al. (2020) for a brief review). Even though 16-bits floating-point numbers become increasingly supported by GPUs (e.g., NVIDIA P100, V100 and A100), Google Tensor Processing Units and even general-purpose CPU (e.g., Fujitsu processor A64FX implementing Armv8.2-A instruction set architecture), most of the studies exploring the prospects of half-precision arithmetic have been carried out using software emulators (Dawson & Düben, 2017) owing to their flexibility. Reducing precision in an atmospheric general circulation model with simplified parametrizations SPEEDY down to 10 significant bits, which is equivalent to half precision, demonstrated that the resulting rounding errors do not exceed the model uncertainty and, thus, half- and double-precision medium-range ensemble forecasts appear to be statistically equivalent. Low-precision climate simulations of the same model lead to similar conclusions (Paxton et al., 2021). The potential of using low-precision calculations in the Open Integrated Forecasting System (OpenIFS), a portable version of IFS, was studied by Chantry et al. (2019) who demonstrated that calculations in the spectral space in OpenIFS can be done mostly in half precision if the largest scales are represented with double precision.

Importantly, while reducing the number of bits in the significand, the aforementioned studies set the exponent of floating-point numbers to be equivalent to that of single or double precision. Forcing the exponent also to comply with a half-precision format dramatically decreases the dynamic range of variables which often leads to large errors or even program crashes due to overflow floating-point exceptions. Rescaling and shifting as well as promoting variables with a large dynamic range to single precision are possible remedies as was demonstrated by Klöwer et al. (2020) who managed to run a shallow water equation model using 16-bits arithmetic with a control of exponent and ported a half-precision version of this model on real hardware reporting about 4x speedup (Klöwer et al., 2021).

In this paper, we build on this body of research and explore the perspective of using reduced precision in ecRad, a radiation scheme used operationally in the IFS from July 2017 (Hogan & Bozzo, 2018). Compared to other parametrization schemes, it consumes a significant amount of computational time which causes the use of a coarser grid and calling it with a time step several times larger than the main model. Making its computations more efficient would allows for more frequent calls of the radiation scheme which would improve the accuracy of weather forecasts. This makes the radiation scheme an appropriate candidate for acceleration. For example, the gas optics module of ecRad has recently been a target for neural-networks acceleration (Ukkonen et al., 2020). In contrast, we will focus on longwave and shortwave solvers, the most time-consuming parts of the radiation scheme as measured by Hogan and Bozzo (2018), and explore how we can make use of precision reduction there. In the next section, we explain how reduced precision was introduced in the code and then, in Section 3, discuss "naive" precision reduction in ecRad which appeared to be unsatisfactory. To develop a more advanced mixed-precision version of the code, we needed to explore typical issues caused by using half-precision floating-point numbers (the lowest levels of precision investigated in our work), and possible ways to overcome them. This topic is covered in Sections 4 and 5 where the latter introduces ensemble-based rounding error analysis, a useful approach for finding variables and operations causing numerical instabilities in the context of reduced precision. In Sections 6 and 7, we provide evidence that an advanced mixed-precision version of ecRad, where half-precision variables are deliberately mixed with single- and double-precision variables, yields adequate accuracy compared to the double-precision version both in terms of instantaneous heating rates and medium-range forecast skill. Finally, we summarize our results and discuss possible caveats on the way towards porting this mixed-precision version on the real hardware in the last section concluding the paper.

2 Implementing reduced precision

Real numbers are typically represented using floating-point numbers as defined by the IEEE 754 standard (Zuras et al., 2008). An N-bits floating-point number x consists of one sign bit, r bits of the exponent and p bits of the significand which we will refer to as sbits. Its decimal representation has the following form:

$$x = (-1)^s 2^{e-e_{\text{bias}}} \left(1 + \sum_{j=1}^{p} m_j 2^{-j} \right), \tag{1}$$

where s is the sign bit, e is the exponent stored as an integer number, e_{bias} is the exponent bias as defined by the IEEE standard and m_j is the jth bit of the significand. Formula (1) represents *normalized numbers* when $e \neq 0$. To reduce the number of underflow exceptions, *subnormal numbers* were introduced in the IEEE 754 standard to represent numbers smaller than the smallest normalized number. Subnormal numbers are represented by an N-bits floating-point number x when $e = 0$ using the following formula:

$$x = (-1)^s 2^{-e_{\text{bias}}+1} \left(\sum_{j=1}^{p} m_j 2^{-j} \right). \tag{2}$$

We are particularly interested in double- (64 bits), single- (32 bits) and half-precision (16 bits) formats defined by the standard. Their corresponding characteristics are shown in table 1. It is important to say that the relative error of the approximation of an arbitrary real number lying within the dynamic range of normalized numbers with a floating-point number is bounded by a constant known as the *machine epsilon* and equal to 2^{-p}.

Table 1: Characteristics of floating-point types defined by the IEEE 754 standard. Only positive numbers are considered for convenience.

Type	Total bits	Exponent bits	Significant bits	Machine epsilon	Smallest subnormal number	Dynamic range without subnormals
Double	64	11	52	2.22×10^{-16}	4.94×10^{-324}	2.23×10^{-308} to 1.80×10^{308}
Single	32	8	23	1.19×10^{-7}	1.40×10^{-45}	1.18×10^{-38} to 3.40×10^{38}
Half	16	5	10	9.77×10^{-4}	5.96×10^{-8}	6.10×10^{-5} to 65504

In this study, we aimed at keeping most of the variables in half precision. The hardware support of half precision is limited which motivated us to use the Fortran library rpe allowing for the emulation of half precision (Dawson & Düben, 2017). In addition to the IEEE half-precision emulation, it offers a combined floating-point number format where one can arbitrarily vary the number of sbits while using the IEEE double-precision exponent thereby eliminating potential issues with the dynamic range and focusing only on studying the effect of reduced precision on computations. To introduce the emulation of floating-point arithmetic in the code, one only needs to replace types in declarations of real variables with a special derived type `rpe_var`. The assignment, arithmetic and logic operators as well as many Fortran intrinsic procedures are overloaded for this type so that precision reduction is applied at all the intermediate operations in compound expressions thereby emulating similar processes in hardware (Dawson & Düben, 2017). For all `rpe_var`'s, the number of sbits can be adjusted individually, and a fine-grained precision analysis can be performed. It should be noted that it is usually impractical to replace all the real types with `rpe_var` since any complicated code is likely to call either intrinsic procedures, not overloaded in the rpe library, or procedures from external libraries, for example, related to the input-output operations.

3 Naive precision reduction in ecRad

Using the rpe library, we explored to which extent the radiation scheme ecRad can benefit from low-precision computations. We focused on its most computationally expensive part, the shortwave and longwave solvers computing the radiative transfer. In particular, we developed a reduced-precision version of the Tripleclouds solver (Shonk & Hogan, 2008) which, being relatively slow in comparison to the operationally used solver McICA as shown by Hogan and Bozzo (2018), was a good target for improvement in terms of computational efficiency. This required turning 80 real variables to type `rpe_var` which allowed us to control their precision via the number of significant bits. To test the accuracy of the reduced-precision version of ecRad, we compare the shortwave and longwave heating rates profiles computed by the reduced-precision and double-precision versions. As test inputs, we use a set of vertical profiles prepared for ecRad from the ERA5 reanalysis data for the year 2001 with 6-hour step and 1.5-degree resolution. After computing the heating rate profiles for each time, latitude and longitude, we calculate the root-mean-square error (RMSE), averaged over time and horizontal coordinates, with respect to the double-precision outputs and report the resulting error profiles in figure 1. Here, we present the error profiles for versions with intermediate precision, gradually decreasing from single precision (23 sbits) to half precision (10 sbits) while keeping the double-precision exponent for all of the `rpe_var` variables. Apart from changing the num-

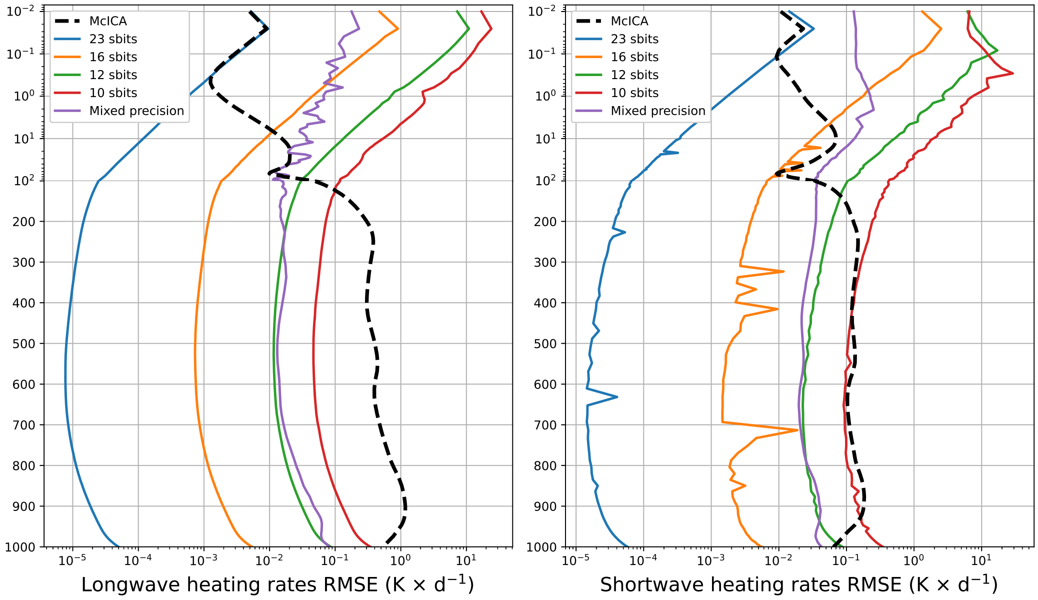

Figure 1: Space- and time-averaged root-mean-square errors of profiles of instantaneous longwave (left) and shortwave (right) heating rates computed with different versions of the Tripleclouds solver (coloured curves) with respect to its double-precision version. The black dashed curve denotes a reference for comparisons: the root-mean-square deviation of profiles computed with the double-precision McICA solver from those computed with the double-precision Tripleclouds solver.

ber of sbits for all the `rpe_var` variables, no additional interventions to the code were made in these versions of ecRad which allows us to treat them as examples of "naive" precision reduction. It is important to note that, instead of heating rates, the ecRad radiation scheme outputs irradiance flux profiles which then need to be differentiated with respect to the pressure and scaled to get heating rate profiles. As a result, differentiation additionally amplifies intrinsic errors of ecRad calculations. Nonetheless, we demonstrate all the error plots with respect to the heating rate since it is the quantity that is eventually used to update the tendencies of the prognostic variables in IFS.

We first read from figure 1 the general pattern of the precision-induced error which tends to be smaller in the troposphere and larger closer to the surface and in the stratosphere and mesosphere. Reducing the number of significant bits unsurprisingly leads to the overall increase of the RMSE from $O(10^{-5}\ \mathrm{K}\times\mathrm{d}^{-1})$ (single precision) to $O(10^{-1}\ \mathrm{K}\times\mathrm{d}^{-1})$ (half precision) in the midtroposphere. The error magnitude becomes significantly larger in the stratosphere and mesosphere taking unacceptable values up to $O(10\ \mathrm{K}\times\mathrm{d}^{-1})$ in the mesosphere for the 10-sbits version. The physical reason of this susceptibility of the stratosphere and mesosphere to reduced precision lies in the calculation of heating rate which, for the i-th layer, reads

$$HR = -\frac{g}{C_p} \cdot \frac{F^{\mathrm{net}}_{i+1/2} - F^{\mathrm{net}}_{i-1/2}}{p_{i+1/2} - p_{i-1/2}}, \tag{3}$$

where $F^{\mathrm{net}}_{i+1/2}$ is the net flux (down flux minus up flux) between layers i and $i+1$ (counting down from the top), $p_{i+1/2}$ is the pressure between layers i and $i+1$, g is the acceleration due to gravity and C_p is the specific heat of air. Based on our observations, the numerator $F^{\mathrm{net}}_{i+1/2} - F^{\mathrm{net}}_{i-1/2}$ takes $O(10^{-4})$ values at 0.1 hPa thereby leading to large errors in heating rates. In contrast, the numerator value is of order of $O(10^{-1})$ at 750 hPa which is easier to handle in half precision.

One can also note error spikes in the shortwave heating rates especially pronounced for the 23-sbits and 16-sbits curves. These are the consequence of numerical instabilities occurring in a subroutine computing the shortwave reflection and transmission.

To make sense of the magnitude of RMSE values shown in figure 1, we introduce a new reference: a root-mean-square difference between the heating rates produced by the ecRad radiation scheme with double-precision McICA and Tripleclouds solvers (dashed black line in figure 1). It should be noted that the McICA solver is stochastic and has noise in cloudy profiles which should average to zero over a long period (Räisänen et al., 2005; Hill et al., 2011; Hogan & Bozzo, 2018). Therefore, our reference measures rather the instantaneous noise in McICA (e.g., as shown by blue curves in figures 4(c) and 4(d) by Hogan and Bozzo (2018)) than the systematic difference between McICA and Tripleclouds solvers which is in fact much smaller. It is reasonable to expect that RMSE values of a reduced-precision version of the Tripleclouds solver should not exceed this reference measuring the difference between two solvers. We can however observe that this clearly does not hold for the 10-sbits version.

4 Difficulties in using IEEE half precision

There are two challenges when running a numerical code in IEEE half-precision. (1) The dynamic range: Half precision can only represent numbers (other than zero) with absolute values between 5.96×10^{-8} (including subnormals) and 65504. Smaller numbers will be truncated to zero, which can cause model crashes in subsequent divisions. Larger numbers will cause an overflow which will also result in a crash of the program. (2) The decimal precision: Half precision numbers can only represent a decimal precision of three digits. Rounding errors will therefore grow quickly. In particular, for subtractions of similar numbers or summations of a small to a large number.

Due to the limited dynamic range, it is crucially important to be able to control the range of values taken by IEEE half-precision variables. A practical way of doing this is multiplicative rescaling, i.e. multiplying a variable by a constant to shift the variable range so that it fits the half-precision range. Rescaling has two important limitations. First, while changing the range in absolute values, rescaling preserves the dynamic range of the variable, i.e. the ratio between the largest and smallest absolute values, and, therefore, cannot fit variables whose dynamic range exceeds 10^9 into the half-precision normalized number range. Since we must guarantee that all half-precision variable values are less than 65504, the compromise would be to tolerate an increased number of subnormal and flushed-to-zero values.

The second limitation stems from the fact that rescaling is difficult to employ unless the variable transformations occurring between scaling and unscaling are linear as in the Legendre transform (Hatfield et al., 2019), linear terms of differential equations (Klöwer et al., 2020) or derivatives in training neural networks (Micikevicius et al., 2018). In contrast to these examples, the ecRad radiation scheme, as many other physical parametrization schemes, contains a long sequence of both linear and nonlinear calculations accompanied with conditional statements seriously complicating the use of rescaling. As a result, if rescaling or expression reordering cannot be used, we simply promote problematic variables to single precision.

It is now clear that prior to any decision regarding the choice of precision for a particular variable, we need to assess its range. To facilitate this assessment, we extended the rpe software and added an automatic collection of statistics of values assigned to any `rpe_var` variable. This extension to the rpe library, akin in spirit to the package Sherlogs.jl written in Julia (Klöwer et al., 2021), provides sufficient amount of information about the range of values assigned to a particular `rpe_var` variable to conclude whether the variable can in principle be turned to half precision. Since the sign of values does not provide any useful information, all the collected statistics are related to absolute values only. The quantities gathered for all the `rpe_var` variables include the total number of assignments, minimum, maximum and mean absolute values, the number of zero assignments and the histogram of absolute values with bins defined by $\pm\infty$ and $10^{\pm k}$, where $k \in \{1, 3, 5, 7, 16\}$. The statistics are dumped to a file individually for each variables by calling a dedicated subroutine. To collect this information, the extension updates internal data of an `rpe_var` variable every time some value is assigned to it which of course slows down the overall calculations.

We collect necessary statistics by running the code with the ecRad input data corresponding to a single day from the ERA5 reanalysis data mentioned above with all the `rpe_var` variables set to double precision. An example of the resulting statistics can be seen from figure 2 where we demonstrate statistics of all the real variables used in a subroutine computing the shortwave irradiances for both double- and mixed-precision version of the ecRad shortwave solver. One can observe that several variables, e.g. `od_total` (optical depth of gas+aerosol+cloud in a given layer and given spectral interval), tend to take values close to the largest normalized half-precision number and, therefore, require either rescaling or promoting to single precision. At the same time, the majority of variables are likely to take values below the smallest normalized half-precision number. Turning some of them to half precision, e.g. `scat_od` (scattering optical depth of gas+aerosol) and `od_total` involved into the calculation of single-scattering albedo and asymmetry factor of gas-cloud combination within a given layer of the atmosphere, significantly increase the relative error of computations and may lead to division-by-zero exceptions. For `scat_od` and `od_total`, two different strategies were used to mitigate these issues: `scat_od` was rescaled (see the right part of figure 2) and `od_total` was promoted to single precision because its dynamic range appeared to be greater than that of half precision. Other variables were analysed in a similar fashion. While our extension for the variable statistics collection does not allow us to obtain any information about the range of temporary variables, it still provides us with enough knowledge to conclude what variables could potentially be turned to half precision and, if appropriate, how they should be rescaled.

It is worth pointing out that some variables, such as fluxes or albedo values, can be flushed to zero without underflow exceptions because it would never make physical sense to divide by these numbers which makes them perfect candidates for half-precision conversion. However, higher precision can still be necessary for them to guarantee the required accuracy of further calculations (e.g., higher precision of fluxes is important for computing heating rates).

5 Ensemble-based rounding error analysis

Without additional tools, the identification and localization of errors caused by the extensive use of reduced-precision calculations requires a lot of manual labour. Moreover, these errors may be invisible for typical measures of accuracy, such as the root-mean-square error (RMSE), smoothing out extreme fluctuations of the error occurring, for example, due to their spatial localization. Figure 3 demonstrates the most prominent example of such a localized error we identified while adapting the radiation scheme ecRad to reduced precision. It was a spontaneous error occurring at specific latitude, longitude and time, i.e it was localized both in space and time. The source of this error was found in a subroutine computing the shortwave reflection and transmission for a given level using the formulas from Meador and Weaver (1980). To solve the problem, the computation was changed to be calculated at native precision which did not only eliminate this error, but also improved the overall accuracy of the results.

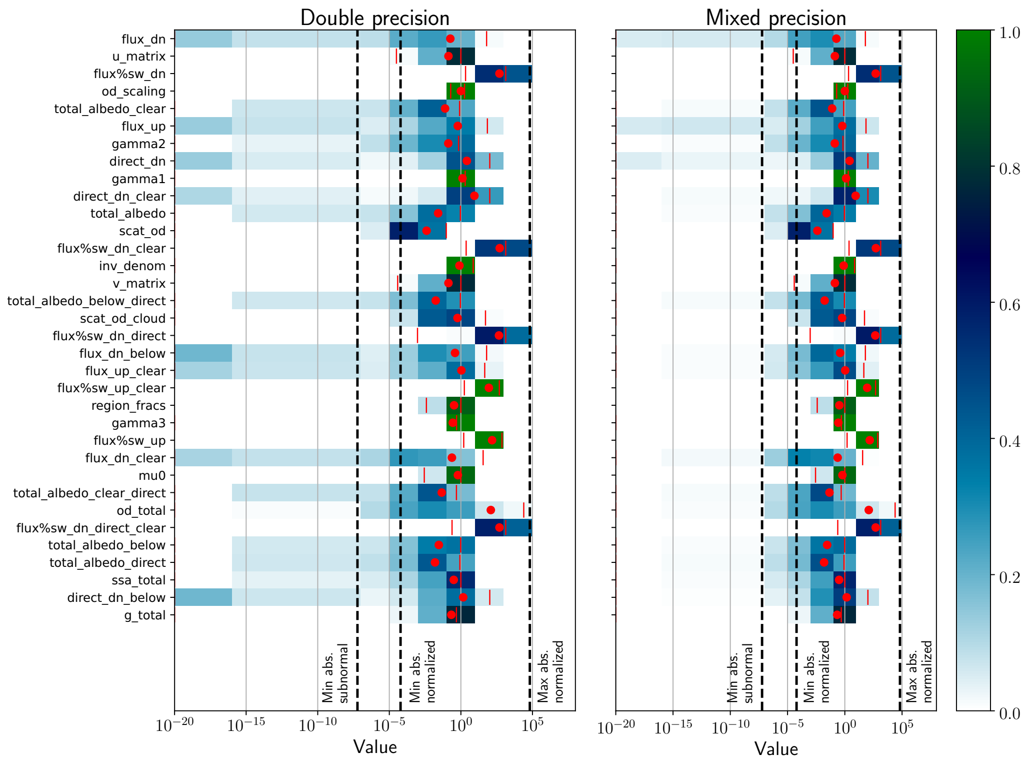

Figure 2: Histograms of all the absolute values assigned to the local variables used in the main subroutine of the shortwave Tripleclouds solver. They were built by running double-precision (left) and mixed-precision (right) versions of the solver. Red dots show the sample mean of distributions whereas red dashes show the maximum and minimum non-zero absolute values. The rightmost color bar is associated with the probability of getting the value lying in a histogram bin. Note that the presented histograms only weakly depend on a choice of inputs from the prepared ERA5 dataset (see the main text for details), but do vary if the radiation scheme is used within the OpenIFS.

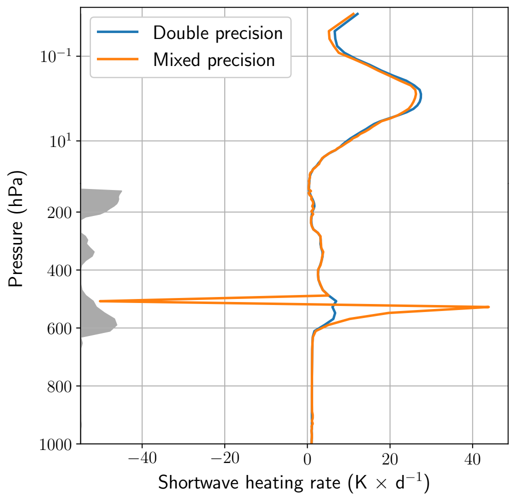

Figure 3: Comparison of profiles of instantaneous shortwave heating rates computed at a fixed latitude and longitude with double- and mixed-precision versions of the Tripleclouds solver. Gray regions on the left show the cloud cover spanning from 0 to 1. Note that the numerical instability observed around 500 hPa is a rare event: there is no similar instability at any of the nearby latitudes nor longitudes.

Searching for particular places in the code causing errors similar to the one shown in figure 3 can be extremely laborious, especially if the code is used to simulate nonlinear dynamics and is heterogeneous with respect to arithmetic operations and intrinsics being involved in calculations. We have therefore developed a tool to find parts of the code where rounding errors start growing excessively. A straightforward approach to automating this process would be to modify the rpe software so that it could compute every operation in double precision in parallel to emulating reduced precision, track the difference between the reduced-precision and double-precision outcomes, and alerting the user when they diverge too much. However, in the presence of sensitivity to tiny perturbations in initial conditions typical for chaotic systems, this approach fails to recognize problematic code lines since double-precision and reduced-precision forecasts may start diverging due to chaotic properties of the underlying system completely unrelated to the quality of computation. An alternative approach taking into account this feature and introduced in this paper as *ensemble-based rounding error analysis* is to compare ensemble predictions, i.e. double-precision and reduced-precision distributions of each variable, at every operation in the code. Its core idea is illustrated by figure 4 where two modes of rounding error analysis are presented. The first mode, shown in sketch 4(c), implies comparing a single computation in reduced precision to the double-precision ensemble which is suitable for non-chaotic calculations. However, if the variable follows chaotic dynamics, this mode may not be able to detect a significant deviation of the reduced-precision forecast from the reference since the former still fits well into the double-precision distribution of the variable. In this case, the second mode, shown in sketch 4(d), should be used: it implies comparing reduced-precision and double-precision ensembles. In both modes, we identify the problematic line of code as follows: if the reduced-precision value (or ensemble mean) deviates from the double-precision mean by more than 3σ, where σ is the double-precision ensemble standard deviation, the program is interrupted due to either an artificially introduced floating-poing exception or debugger breakpoints. Both ways output a particular file and line whose inspection should help identify problematic operation and variables. The aforementioned approach was implemented as an exten-

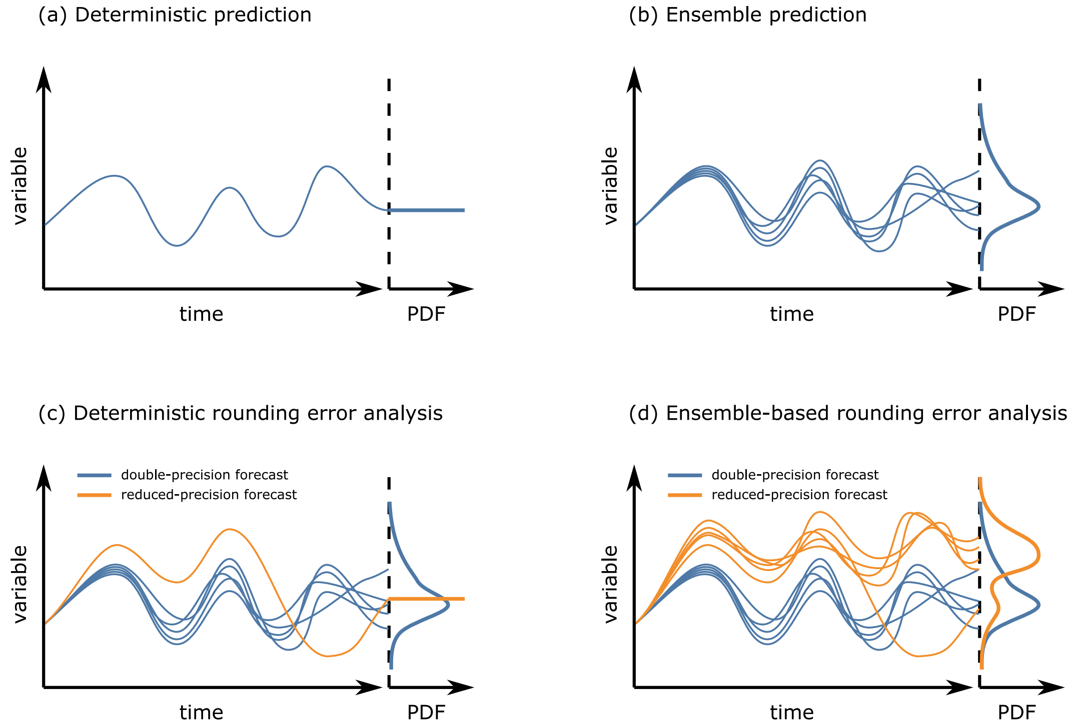

Figure 4: Illustration of deterministic (sketch (a)) and ensemble (sketch (b)) prediction modes together with two types of rounding error analysis: deterministic (sketch (c)) and ensemble-based (sketch (d)). Blue (orange) curves correspond to forecasts of some variable made by a double-precision (reduced-precision) code. The left part of each sketch display the time-evolution of the variable, whereas the right part shows its probability density function (PDF) estimation.

sion to the rpe library where each `rpe_var` variable stores an ensemble of values in a way akin to the ensemble format suggested by Düben (2018).

In combination with the variable statistics, this tool helped find parts of the code causing a sudden increase of the error and mitigate it using rescaling, reordering or promoting to single precision. In particular, we found several places where operations involving subnormal numbers resulted in a several order-of-magnitude increase of the relative error spreading further in the code which reinforces the importance of rescaling the corresponding variables (Klöwer et al., 2021). Figure 5 shows a particular scenario of how a significant growth of the relative error can occur during the calculations and how it can be identified using our extension. It is exemplified by a simple piece of code aimed at computing the centre of mass where all the variables are represented by half-precision floating point numbers. Its first four lines correspond to simple assignments within the dynamic range of normal numbers, so that the rounding error inevitably introduced in each assignment is bounded by the machine epsilon $\epsilon \approx 9.8 \times 10^{-4}$ (see the right plot showing the relative error of computations). The scaling employed then in the fifth line pulls the value of the variable out of the normal range making it subnormal. Given that there is only a limited and very sparse set of subnormal numbers, the consequence of this operation is a drastic increase of the relative error of the computation. This is exactly the place where our extension will throw an exception warning about an error since the reduced-precision value has dropped out of the 3σ-window of the reference ensemble. If we ignored it, the final line of the code would yield the value inheriting a large relative error from the previous line. In this particular case, the problem can be solved by a proper

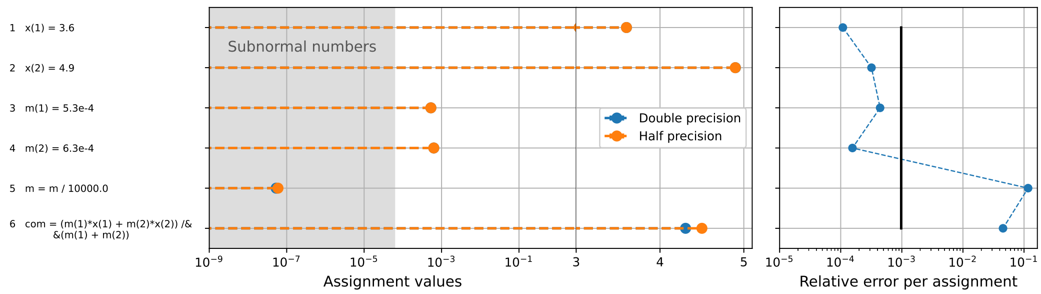

Figure 5: Code example leading to a significant growth of the relative rounding error of calculations. Numbered codes lines are shown on the left. The left plot shows values assigned at each line of the code while computing in double (blue) and half (orange) precision. The right plot shows the relative difference between double- and half-precision computations.

rescaling of variable `m` since it its dynamic range is smaller than that of half-precision floating-point numbers.

6 Mixed-precision version of ecRad

Making use of the aforementioned techniques and tools, we developed a mixed-precision version of the Tripleclouds solver with IEEE half-precision variables. Native precision was still kept in three subroutines: one of them computes the delta-Eddington scaling (Joseph et al., 1976) and two other ones compute the shortwave and longwave reflection and transmission at a given height. They all appeared to be particularly sensitive to lowering precision of their variables. The extensions to the rpe library helped identify a set of variables requiring either rescaling or promoting to single precision which was important to avoid floating-point overflows and improve the accuracy of results. After all adjustments, about 75% of `rpe_var` variables in the mixed-precision version of the solver were handled in half precision and the rest in single precision (see Appendix A for details on precision of subroutines used within the solver). Importantly, these include optical and cloud properties, the key inputs necessary for the Tripleclouds solver to compute the shortwave and longwave irradiance profiles. They are listed in table 2 alongside their variable precision used in the mixed-precision version of the code. Almost all of the variables could be reduced to half precision. Two exceptions, optical depth and single scattering albedo, are characterized by a wide dynamic range whose truncation directly affects the accuracy which urged us to use single precision for them.

The RMSE of the heating rate profiles for the mixed-precision version is shown in purple in figure 1. When compered to naive 10-sbits precision reduction (red curve), it does in general decrease the RMSE and, importantly, reduces the overly large errors in the mesosphere by two orders-of-magnitude. RMSE values of the mixed-precision version are smaller than the reference (RMSE values for the McICA solver) in the troposphere, but they clearly exceed the reference values in the stratosphere and mesosphere. The latter however is related to the fact that the McICA and Tripleclouds solvers only differ in how they represent cloud structure. As a result, the reference errors tend to be smaller above the troposphere which is especially pronounced for longwave heating rates. Taking this into account, we can conclude that the mixed-precision version of the Tripleclouds-powered radiation scheme compares well with the reference.

Table 2: Precision of input variables passed to the Tripleclouds solver

Name	Number of significant bits	
	Longwave solver	Shortwave solver
Layer optical depth (`od`)	10	23
Single scattering albedo (`ssa`)	10	23
Asymmetry factor (`g`)	10	10
In-cloud optical depth (`od_cloud`)	10	10
In-cloud single scattering albedo (`ssa_cloud`)	10	10
In-cloud asymmetry factor (`g_cloud`)	10	10
Planck function at half levels (`planck_hl`)	10	–
Longwave emission from the surface (`lw_emission`)	10	–
Longwave albedo of the surface (`lw_albedo`)	10	–
Direct shortwave albedo of the surface (`sw_albedo_direct`)	–	10
Diffuse shortwave albedo of the surface (`sw_albedo_diffuse`)	–	10
Incoming shortwave flux at top-of-atmosphere (`incoming_sw`)	–	10

It is also important to ensure that the bias, caused by precision reduction and shadowed by the RMSE measure, does not become unreasonably large. This information can be deduced from figure 6 showing the difference between double-precision Tripleclouds outputs and various reduced-precision versions of the code as a function of pressure. For the case of longwave heating rates and the mixed-precision solver (left plot), we can clearly observe a cooling bias growing with height whose magnitude however is bounded by -0.2 K$\times$d^{-1}, the value achieved in the middle mesosphere where the temperature prediction is known to tend to be unrealistic (Hogan & Bozzo, 2018). We need to mention that it is typical for the ecRad radiation scheme to develop an increasing warming bias in the upper stratosphere and above (Hogan & Bozzo, 2018). Therefore, we may consider a slight longwave cooling induced by the mixed-precision ecRad as acceptable. At the same time, no shortwave bias is observed even though the bias variations become unreasonably large for 12-sbits and 10-sbits results (right plot in figure 6). However, using the mixed-precision solver drastically decreases their magnitude making them at least one order-of-magnitude smaller. This gives a strong evidence that the mixed-precision version of the Tripleclouds solver can successfully be used for weather and climate forecasting.

7 Influence of precision reduction on medium-range forecast

So far we have been examining the deviation of instantaneous heating rates calculated by the ecRad radiation scheme subject to precision reduction from their double-precision companions. In this Section, we take a step forward and assess the influence of reduced-precision outputs of the radiation scheme on the forecast skill of OpenIFS, a portable version of IFS. Namely, we make 10-day weather forecasts based on the latest available version of OpenIFS corresponding to IFS Cycle 43R3 used operationally from July 2017 to June 2018 and for the first time employing ecRad as the operational radiation scheme. The resolution being used in our study is T_l255 corresponding to 78-km horizontal spacing and 91 vertical levels. The time step of the model is 45 minutes, and the radiation scheme is invoked every 3 hours. We embed the reduced-precision code of ecRad into OpenIFS and run this model for 10 days starting from 1 November 2019.

We start from exploring the difference in the geopotential height, 2-meter temperature and surface downwelling shortwave and longwave radiation being formed after 10 days of forecast. The corresponding maps are shown in figure 7. As usual, the double-

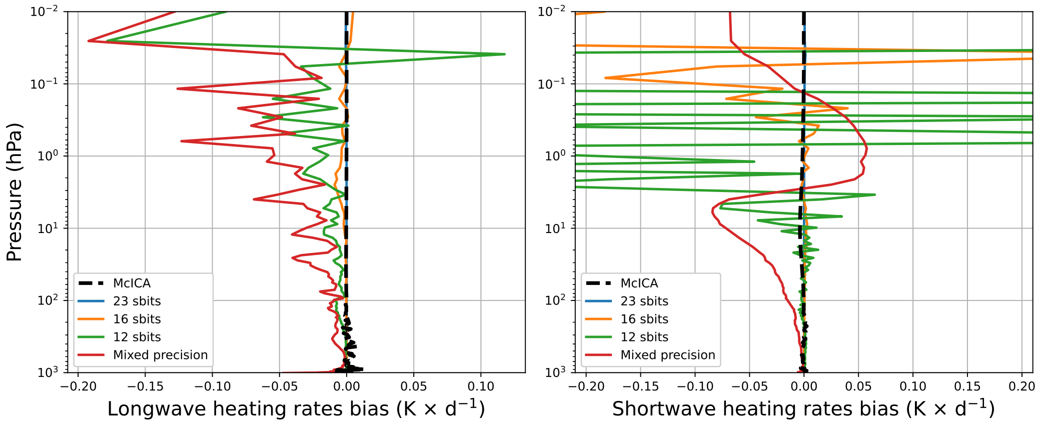

Figure 6: Space- and time-averaged difference between profiles of instantaneous longwave (left) and shortwave (right) heating rates computed with the double-precision version of the Tripleclouds solver and its reduced-precision modifications (coloured curves). The black dashed curve is a reference as explained in figure 1.

precision calculations obtained with the Tripleclouds solver act as a ground truth to which we compare 23-sbits and mixed-precision results. For reference, we compare to an alternative solver,McICA, run at double precision. One can easily observe that no strong bias is developed in either configuration. For the geopotential height, occasional regions where they differ start developing roughly between 50° and 70° latitudes in both hemispheres. Noticeable deviations can be found at the same latitudes for the 2-meter temperature and surface downwelling longwave radiation, but only in the Northern Hemisphere. However, these regions also appear in our reference calculations where the McICA solver is used in ecRad which implies that these deviation patterns are likely to stem from mere chaotic sensitivity to perturbations in the radiation scheme rather than any systematic bias induced by precision reduction. Observed differences can be acceptable as long as they remain small compared to the uncertainty of predictions which can be quantified by the impact of stochastic parametrization schemes such as stochastically perturbed parametrization tendency (SPPT) routinely used in ECMWF for medium-range probabilistic forecasts (Buizza et al., 1999). This is a driving motivation for the successful use of imprecise computing in weather forecasting (Düben & Palmer, 2014; T. N. Palmer, 2014).

Similar conclusions can be drawn if we examine the time-evolution of the forecast error of the temperature at different heights. To make a fair comparison, we track how the root-mean-square (RMS) forecast error, i.e. the RMS deviation of the forecast from the ERA5 data, computed for the reduced-precision Tripleclouds or double-precision McICA changes with respect to the double-precision Tripleclouds. The corresponding changes are shown in figure 8. As expected from a perturbed chaotic system, they typically grow with time for all the considered ecRad configurations with 300-hPa temperature displaying marginally larger variations. We can note that the mixed-precision change does not seem to differ significantly from the single-precision and McICA values neither in the trend nor magnitude. This provides additional evidence that the inaccuracy induced by careful precision reduction in the radiation scheme is likely to be sufficiently small for forecasting in the presence of uncertainties.

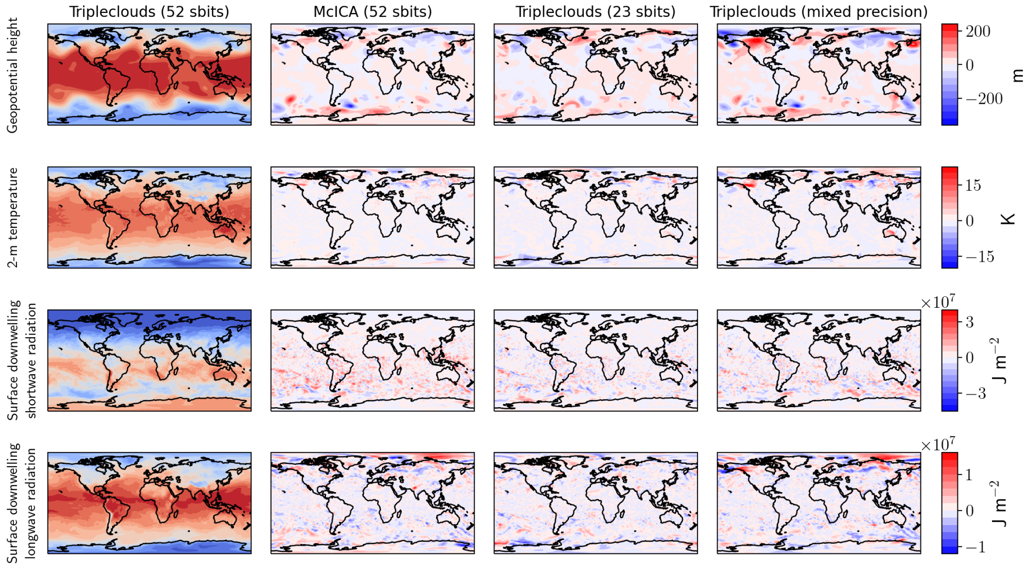

Figure 7: Geopotential height at pressure 500 hPa, 2-meter temperature and surface downwelling shortwave and longwave radiation after 10 days of the OpenIFS simulation displayed for double-, single- and mixed-precision versions of the Tripleclouds solver (the first, third and forth columns) and the McICA solver (the second column) where the latter serves as a reference. Fields for McICA, single- and mixed-precision Tripleclouds solvers are shown as deviations from the forecast made with the double-precision Tripleclouds solver. Deviation values are described by color bars on the right.

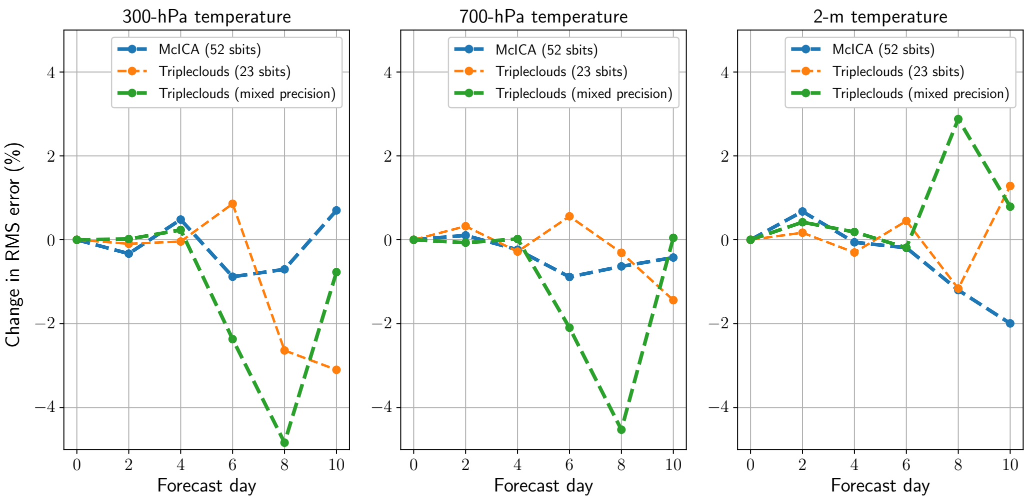

Figure 8: Time evolution of the change in root-mean-square forecast error of the temperature at pressure 300 and 700 hPa and at 2 meters induced by reduced precision in the radiation scheme and computed with respect to the double-precision version of the radiation scheme. The forecast error is defined as the deviation of the forecast from the ERA5 data. As a reference, same quantity is plotted for the change induced by replacement of the double-precision Tripleclouds solver with the double-precision McICA solver.

8 Conclusion

In this paper, we have considered the reduced-precision versions of the radiation scheme ecRad operationally used in the ECMWF's IFS. Namely, we introduced precision reduction in the Tripleclouds solver, the most computationally expensive component of the radiation scheme, using a Fortran emulator of reduced precision named rpe (Dawson & Düben, 2017). We have demonstrated that "naive" precision reduction, where the number of significant bits is reduced and fixed for all the real-valued variables, leads to a strong deviation of the resulting heating rates from the ground-truth double-precision calculations. To overcome this problem, we explored a mixed-precision approach where the whole set of real-valued variables is split into three subsets containing variables with double, single and half precision respectively. The flexibility of the mixed-precision approach allows one to adjust a trade off between the accuracy and the speed of computations by changing the ratio of double-, single- and half-precision variables. Splitting was performed based on the dynamic range of variables and the effect of their precision on the overall accuracy of calculations. To facilitate the process of finding a proper partition, we developed two extensions to the rpe library: the first one automatically gathers all the necessary statistics about the range of values assigned to reduced-precision variables, and the second one helps tracking the divergence between double- and reduced-precision calculations line-by-line thereby making it possible to localize particular parts of the code and even variables causing undesirable loss of accuracy. Based on ERA5 reanalysis data for the year 2001, we have demonstrated that heating rates produced by the resulting mixed-precision version of the Tripleclouds solver are close to their double-precision companions if measured relative to the inter-model difference between the double-precision McICA and Tripleclouds solvers. Additionally, we have shown that replacing the OpenIFS' radiation scheme with its mixed-precision version has only a small influence on the accuracy of a medium-range forecast well comparable to the difference appearing when the Tripleclouds solver is replaced with McICA in the radiation scheme.

It is important to say that as we have only emulated reduced precision in this work, we cannot present any assessments of speed-up which is an ultimate goal of introducing precision reduction. This can only be done if the radiation scheme together with OpenIFS are ported on the hardware natively supporting half-precision floating-point numbers. A notable example of such hardware is the Fujitsu microprocessor A64FX. This process may require additional changes of the mixed-precision radiation scheme because mixing variables of various precision levels may slow down certain parts of the code diminishing the potential speed-up. Moreover, the A64FX microprocessor is known to handle half-precision subnormal numbers slowly which may become another obstacle to successful porting (Klöwer et al., 2021). Subnormal numbers are unlikely to be encountered when dealing with double- or single-precision variables, but they appear much more frequently for half precision. A possible solution is to try to avoid using subnormal numbers for half-precision variables, which can be achieved by a proper rescaling of variables, and flush them to zero if they occasionally appear (Klöwer et al., 2021).

Another avenue to explore is the use of stochastic rounding instead of currently used round-to-nearest approach. There is now a growing body of evidence suggesting that rounding errors can efficiently be mitigated if stochastic rounding is used for half-precision variables (Croci & Giles, 2020; Paxton et al., 2021). We performed a set of experiments similar to that in Section 3 on a limited set of data with enabled stochastic rounding and found that, even though stochastic rounding does not improve the RMSE values, it completely removes the longwave cooling bias observed in Figure 6 which is an undoubtedly significant improvement of the mixed-precision radiation scheme. Further investigation is however needed to come to a final conclusion.

We believe that our results are promising enough to suggest that mixed-precision arithmetics can be useful for the radiation scheme ecRad and, in perspective, other parametriza-

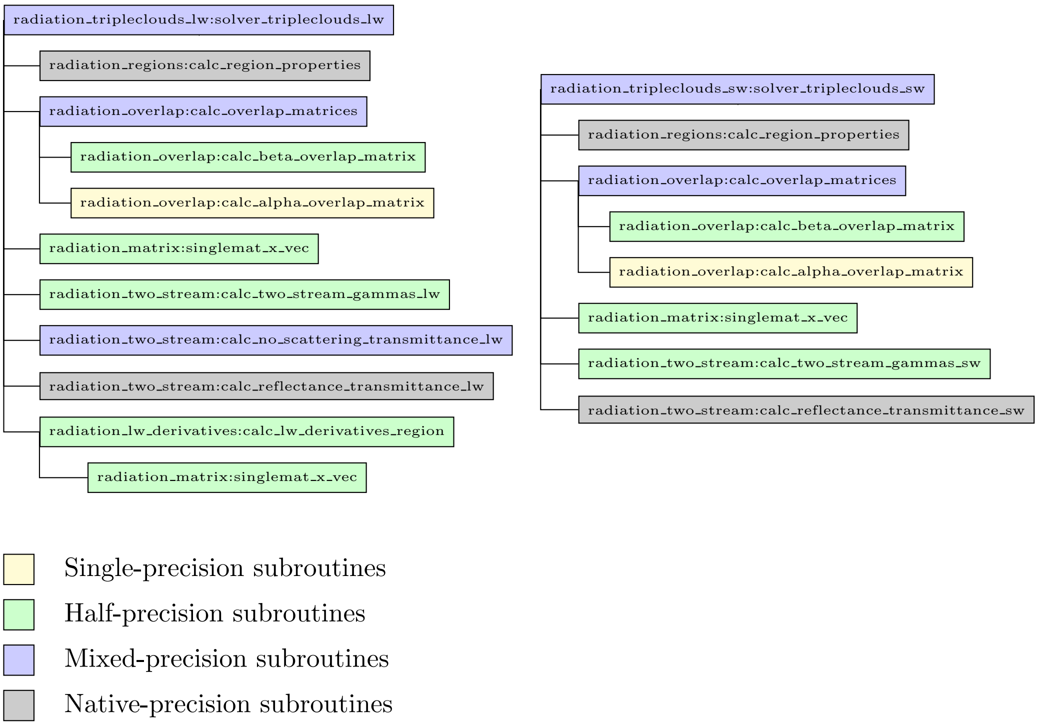

Figure A1: Call structure of Tripleclouds subroutines showing precision used within. Only subroutines where there exist local or allocated variables are presented.

tion schemes used in weather and climate models. More extensive benchmarking is however necessary to continue towards operational use.

Appendix A Local precision of the Tripleclouds subroutines

In this Appendix, we present a detailed arrangement of local precision in all the subroutines used in the mixed-precision version of the Tripleclouds solver. By local precision, we understand precision of local variables used in a subroutine whereas precision of input variables is assumed to be set in the outer scope of a subroutine. The information about local precision of subroutines is summarized in figure A1 where single-, mixed- and half-precision subroutines are highlighted with yellow, blue and green colours. Some subroutines particularly sensitive to precision reduction are left in native precision (gray).

Open Research

Availability Statement

The rpe library is open and available at `https://github.com/aopp-pred/rpe`. The ecRad radiation scheme code is open and available at `https://github.com/ecmwf-ifs/ecrad`. OpenIFS is free to use, but a license from ECMWF is required: `https://www.ecmwf.int/en/research/projects/openifs`.

Acknowledgments

This paper is supported by funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement No. 741112). PD gratefully acknowledges funding from the Royal Society for his University Research Fellowship as well as the ESiWACE2, ESiWACE3 and MAELSTROM projects under Horizon 2020 and the European High-Performance Computing Joint Undertaking (JU; grant agreement No 823988, 101093054 and 955513). The JU receives support from the European Union's Horizon 2020 research and innovation programme and United Kingdom, Germany, Italy, Luxembourg, Switzerland, Norway.

References

Abdelfattah, A., Anzt, H., Boman, E. G., Carson, E., Cojean, T., Dongarra, J., ... Yang, U. M. (2021). A survey of numerical linear algebra methods utilizing mixed-precision arithmetic. *The International Journal of High Performance Computing Applications*, *35*(4), 344-369.

Baboulin, M., Buttari, A., Dongarra, J., Kurzak, J., Langou, J., Langou, J., ... Tomov, S. (2009). Accelerating scientific computations with mixed precision algorithms. *Computer Physics Communications*, *180*(12), 2526–2533.

Bauer, P., Dueben, P. D., Hoefler, T., Quintino, T., Schulthess, T. C., & Wedi, N. P. (2021). The digital revolution of earth-system science. *Nature Computational Science*, *1*(2), 104–113.

Buizza, R., Milleer, M., & Palmer, T. N. (1999). Stochastic representation of model uncertainties in the ecmwf ensemble prediction system. *Quarterly Journal of the Royal Meteorological Society*, *125*(560), 2887–2908.

Chantry, M., Thornes, T., Palmer, T., & Düben, P. (2019). Scale-selective precision for weather and climate forecasting. *Monthly Weather Review*, *147*(2), 645–655.

Croci, M., & Giles, M. B. (2020). Effects of round-to-nearest and stochastic rounding in the numerical solution of the heat equation in low precision. *arXiv preprint arXiv:2010.16225*.

Dawson, A., & Düben, P. D. (2017). rpe v5: an emulator for reduced floating-point precision in large numerical simulations. *Geoscientific Model Development*, *10*(6), 2221–2230.

Düben, P. D. (2018). A new number format for ensemble simulations. *Journal of Advances in Modeling Earth Systems*, *10*(11), 2983–2991.

Düben, P. D., & Palmer, T. (2014). Benchmark tests for numerical weather forecasts on inexact hardware. *Monthly Weather Review*, *142*(10), 3809–3829.

Gupta, S., Agrawal, A., Gopalakrishnan, K., & Narayanan, P. (2015). Deep learning with limited numerical precision. In *International conference on machine learning* (pp. 1737–1746).

Gustafson, J. L., & Yonemoto, I. T. (2017). Beating floating point at its own game: Posit arithmetic. *Supercomputing frontiers and innovations*, *4*(2), 71–86.

Hatfield, S., Chantry, M., Düben, P., & Palmer, T. (2019). Accelerating high-resolution weather models with deep-learning hardware. In *Proceedings of the platform for advanced scientific computing conference* (pp. 1–11).

Hill, P., Manners, J., & Petch, J. (2011). Reducing noise associated with the monte carlo independent column approximation for weather forecasting models. *Quarterly Journal of the Royal Meteorological Society*, *137*(654), 219–228.

Hogan, R. J., & Bozzo, A. (2018). A flexible and efficient radiation scheme for the ecmwf model. *Journal of Advances in Modeling Earth Systems*, *10*(8), 1990–2008.

Hubara, I., Courbariaux, M., Soudry, D., El-Yaniv, R., & Bengio, Y. (2017). Quantized neural networks: Training neural networks with low precision weights and activations. *The Journal of Machine Learning Research*, *18*(1), 6869–6898.

Joseph, J. H., Wiscombe, W., & Weinman, J. (1976). The delta-eddington approximation for radiative flux transfer. *Journal of Atmospheric Sciences*, *33*(12), 2452–2459.

Kalamkar, D., Mudigere, D., Mellempudi, N., Das, D., Banerjee, K., Avancha, S., ... others (2019). A study of bfloat16 for deep learning training. *arXiv preprint arXiv:1905.12322*.

Klöwer, M., Düben, P., & Palmer, T. (2020). Number formats, error mitigation, and scope for 16-bit arithmetics in weather and climate modeling analyzed with a shallow water model. *Journal of Advances in Modeling Earth Systems*, *12*(10), e2020MS002246.

Klöwer, M., Düben, P. D., & Palmer, T. N. (2019). Posits as an alternative to floats for weather and climate models. In *Proceedings of the conference for next generation arithmetic 2019* (pp. 1–8).

Klöwer, M., Hatfield, S., Croci, M., Düben, P., & Palmer, T. (2021). Fluid simulations accelerated with 16 bit: Approaching 4x speedup on a64fx by squeezing shallowwaters.jl into float16. *Earth and Space Science Open Archive*, 26. doi: 10.1002/essoar.10507472.2

Lang, S. T. K., Dawson, A., Diamantakis, M., Dueben, P., Hatfield, S., Leutbecher, M., ... Wedi, N. (n.d.). More accuracy with less precision. *Quarterly Journal of the Royal Meteorological Society*, *n/a*(n/a).

Maynard, C. M., & Walters, D. N. (2019). Mixed-precision arithmetic in the endgame dynamical core of the unified model, a numerical weather prediction and climate model code. *Computer Physics Communications*, *244*, 69–75.

Meador, W., & Weaver, W. (1980). Two-stream approximations to radiative transfer in planetary atmospheres: A unified description of existing methods and a new improvement. *Journal of Atmospheric Sciences*, *37*(3), 630–643.

Micikevicius, P., Narang, S., Alben, J., Diamos, G., Elsen, E., Garcia, D., ... Wu, H. (2018). Mixed precision training. In *International conference on learning representations.* Retrieved from `https://openreview.net/forum?id=r1gs9JgRZ`

Palmer, T. (2015). Modelling: Build imprecise supercomputers. *Nature News*, *526*(7571), 32.

Palmer, T. N. (2014). More reliable forecasts with less precise computations: a fast-track route to cloud-resolved weather and climate simulators? *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *372*(2018), 20130391.

Paxton, E. A., Chantry, M., Klöwer, M., Saffin, L., & Palmer, T. (2021). Climate modelling in low-precision: Effects of both deterministic & stochastic rounding. *arXiv preprint arXiv:2104.15076*.

Räisänen, P., Barker, H. W., & Cole, J. (2005). The monte carlo independent column approximation's conditional random noise: Impact on simulated climate. *Journal of Climate*, *18*(22), 4715–4730.

Rodwell, M., Diamantakis, M., Düben, P., Janoušek, M., Lang, S., Polichtchouk, I., ... Váňa, F. (2021). Ifs upgrade provides more skilful ensemble forecasts. *ECMWF Newsletter*.

Rüdisühli, S., Walser, A., & Fuhrer, O. (2013). Cosmo in single precision. *Cosmo Newsletter*(14), 5–1.

Shonk, J. K., & Hogan, R. J. (2008). Tripleclouds: An efficient method for representing horizontal cloud inhomogeneity in 1d radiation schemes by using three regions at each height. *Journal of Climate*, *21*(11), 2352–2370.

Tintó Prims, O., Acosta, M. C., Moore, A. M., Castrillo, M., Serradell, K., Cortés, A., & Doblas-Reyes, F. J. (2019). How to use mixed precision in ocean models: exploring a potential reduction of numerical precision in nemo 4.0 and roms 3.6. *Geoscientific Model Development*, *12*(7), 3135–3148.

Ukkonen, P., Pincus, R., Hogan, R. J., Pagh Nielsen, K., & Kaas, E. (2020). Accelerating radiation computations for dynamical models with targeted machine learning and code optimization. *Journal of Advances in Modeling Earth Systems*, *12*(12), e2020MS002226.

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*(2), 495–502.

Zuras, D., Cowlishaw, M., Aiken, A., Applegate, M., Bailey, D., Bass, S., … others (2008). Ieee standard for floating-point arithmetic. *IEEE Std*, *754*(2008), 1–70.