Theor. Comput. Fluid Dyn. (2015) 29:311–328
DOI 10.1007/s00162-015-0355-8

ORIGINAL ARTICLE

Peter D. Düben · Stamen I. Dolaptchiev

Rounding errors may be beneficial for simulations of atmospheric flow: results from the forced 1D Burgers equation

Received: 16 February 2015 / Accepted: 27 May 2015 / Published online: 20 June 2015
© The Author(s) 2015. This article is published with open access at Springerlink.com

Abstract Inexact hardware can reduce computational cost, due to a reduced energy demand and an increase in performance, and can therefore allow higher-resolution simulations of the atmosphere within the same budget for computation. We investigate the use of emulated inexact hardware for a model of the randomly forced 1D Burgers equation with stochastic sub-grid-scale parametrisation. Results show that numerical precision can be reduced to only 12 bits in the significand of floating-point numbers—instead of 52 bits for double precision—with no serious degradation in results for all diagnostics considered. Simulations that use inexact hardware on a grid with higher spatial resolution show results that are significantly better compared to simulations in double precision on a coarser grid at similar estimated computing cost. In the second half of the paper, we compare the forcing due to rounding errors to the stochastic forcing of the stochastic parametrisation scheme that is used to represent sub-grid-scale variability in the standard model setup. We argue that stochastic forcings of stochastic parametrisation schemes can provide a first guess for the upper limit of the magnitude of rounding errors of inexact hardware that can be tolerated by model simulations and suggest that rounding errors can be hidden in the distribution of the stochastic forcing. We present an idealised model setup that replaces the expensive stochastic forcing of the stochastic parametrisation scheme with an engineered rounding error forcing and provides results of similar quality. The engineered rounding error forcing can be used to create a forecast ensemble of similar spread compared to an ensemble based on the stochastic forcing. We conclude that rounding errors are not necessarily degrading the quality of model simulations. Instead, they can be beneficial for the representation of sub-grid-scale variability.

Keywords Inexact hardware · Stochastic parametrisation · Numerical precision · Turbulent closure · Ensemble methods

1 Introduction

Resolution in numerical models for atmosphere and ocean is limited by the available performance of today's supercomputers. It was shown in a series of papers [7–9,26] that the use of inexact hardware offers great potential for modelling weather and climate. Inexact hardware could reduce computational cost, due to a reduced energy demand and an increase in performance and could therefore allow higher-resolution simulations

Communicated by Rupert Klein.

P. D. Düben (✉)
Atmospheric, Oceanic and Planetary Physics, University of Oxford, Oxford, UK
E-mail: dueben@atm.ox.ac.uk
Tel.: +0043 1865 282428

S. I. Dolaptchiev
Institut für Atmosphäre und Umwelt, Johann Wolfgang Goethe-Universität Frankfurt/Main, Frankfurt, Germany
E-mail: dolaptchiev@iau.uni-frankfurt.de

within the same budget for computation. This would push parametrisation schemes towards smaller scales. However, rounding errors will influence model simulations.

From the perspective of hardware development, the main motivation behind inexact hardware is to trade a reduction in numerical precision against reduced computational cost and an increase in performance. This trade-off is discussed in the literature since a couple of years (see for example [22,23]). The first approach to inexact hardware was to use so-called stochastic processors that allow hardware errors to occur within numerical simulations while power consumption is reduced by reducing the applied voltage to the floating-point unit (see for example [16,20]). A second approach is "pruning" for which the physical size of the floating-point unit is reduced, by removing parts that are either hardly used or do not have a strong influence on significant bits in the results of floating-point operations [7,17]. Pruning of the floating-point unit can be combined with the use of inexact memory [10]. A third approach is to use hardware that allows reduced precision floating-point arithmetic, such as field programmable gate arrays (FPGAs). Here, the amount of bits that are used to represent fixed point or floating-point numbers can be customised to the application (the hardware is "information efficient", see [24]). Recent studies showed that the use of dataflow engines shows huge potential for atmospheric modelling [12,21,29], especially if floating-point precision is reduced in parts of the models. Dataflow engines are based on FPGAs and stream data from memory through the dataflow engine where thousands of operations can be performed within one computing cycle at relatively low clock frequency.

Studies of emulated inexact hardware in a spectral dynamical core [8,9] showed that the use of reduced floating-point precision or pruning has large potential to reduce computational cost for atmosphere models. For a spectral dynamical core of an atmosphere model at T85 resolution, numerical precision can be reduced to only 6 bits in the exponent and 8 bits in the significand of floating-point numbers within model parts that cause about 98 % of the computational cost for a double precision control simulation with no strong affect on the quality of results and forecast errors [9].

In this paper, we will study the use of inexact hardware in a model that simulates the one-dimensional Burgers equation with stochastic forcing. The model is using the stochastic parametrisation scheme of [5] for turbulent closure. We will perform simulations with emulated reduced precision that identify the minimal level of precision that can be used with no significant increase in model error, and we will provide estimates for the reduction in computational cost that can be expected for reduced precision simulations in comparison with control simulations in double precision.

In the second half of the paper, we will investigate the nature of the forcing due to rounding error and compare the rounding error distribution against the random noise that is used within the stochastic parametrisation scheme. Parametrisation schemes are used within global atmosphere and ocean models to represent sub-grid-scale processes that cannot be represented explicitly at the available numerical resolution. Unfortunately, these processes include many influential physical phenomena, such as deep convection. In a first approach, parametrisation schemes are typically developed with physical reasoning and based on a statistical analysis of observations or high-resolution model output (for example from large eddy simulations). However, the final parameter tuning is often rather based on "trial and error" adjustment than on stringent physical principles or measurements. Parametrisation schemes can only approximate sub-grid-scale behaviour.

Since deterministic parametrisation schemes tend to underestimate sub-grid-scale variability, stochastic parametrisation schemes gain more and more momentum in the development of atmosphere and ocean models. Stochastic parametrisation schemes use random numbers to mimic the stochastic nature of forcings from sub-grid-scale processes. They do not only improve the ensemble spread, but also correct systematic model errors and represent model uncertainty (see, for example, [25]). Stochastic parametrisations include stochastically perturbed physical parametrisation tendencies [3,27], stochastic kinetic energy back-scatter [2,30], cellular automata [1,25], quasi-equilibrium statistical mechanics parametrisations (for example [28]), stochastic differential equations (for example [11,13]) and Markov chains (for example [4,14]). Stochastic forcings are often introduced into the numerical model (for example [2,3,30]) to represent sub-grid-scale variability explicitly. These schemes allow an analysis of forecast uncertainty via an evaluation of the spread of ensemble forecasts.

There is no doubt that it is a difficult challenge to find a suitable distribution for random noise pattern that represents sub-grid-scale variability realistically. However, the papers above suggest several approaches to find appropriate stochastic forcings to be used in parametrisation schemes. The random numbers that are used in stochastic parametrisation schemes (e.g. via Gaussian distributed white noise) are typically of very high quality with a specific mean and standard deviation and with very long repetition times (for example in [5]).

One might argue that this is over-engineered when the large model error in geophysical applications and the rather arbitrary adjustment procedure of parametrisation schemes is taken into account.

The stochastic parametrisation scheme which is used in the Burgers equation model is based on the stochastic mode reduction strategy by [19]. In ongoing research, the same approach is now tested in two- and three-dimensional models of atmospheric dynamics. We use the Burgers model to study the properties of rounding errors and compare the magnitude of rounding errors to the forcing of the stochastic parametrisation scheme. The used setup allows a relatively simple comparison between the stochastic forcing within the stochastic parametrisation scheme and rounding errors due to inexact hardware in a one-dimensional model that is meaningful for atmospheric modelling. We argue that the stochastic forcing of the stochastic parametrisation scheme can be used to hide rounding errors and that the magnitude of the stochastic forcing can serve as a first guess for the upper limit of the magnitude of rounding errors that allow simulations with no change of model statistics.

There are many publications that show that random numbers can be used in stochastic forcings of stochastic parametrisation schemes to improve atmospheric modelling (see [26] and references therein). On the other hand, rounding errors will show almost no correlations in space and time and have similarities to random numbers. Therefore, we argue that rounding errors can also have a beneficial influence in chaotic models, following the same motivation that is used to justify stochastic forcings in stochastic parametrisation schemes—for example to represent sub-grid-scale variability. Obviously, it is unlikely that the shape of the rounding error forcing will be appropriate to represent sub-grid-scale noise without adaptation. However, if the shape and magnitude of the rounding error forcing do not match the desired distribution, changes to the model allow to influence rounding errors. Two examples: (1) A change of the length of the time step (Δt) will have an influence on the net forcing, similar to the influence of a change of the time step in stochastic differential equations if the different scaling of deterministic and stochastic terms with Δt is ignored. (2) A change of the time-stepping scheme will change the shape of the forcing.

If rounding errors are engineered towards a desired distribution, the random forcings that are derived for stochastic parametrisation schemes and designed to improve model simulations by representing sub-grid-scale variability are obviously promising targets for the rounding error distribution. We will study if it is possible to represent such a forcing sufficiently, using only rounding errors.

Section 2 of this paper introduces the model and the used parametrisation schemes, and Sect. 3 explains the emulator for reduced precision. In Sect. 4, we show that numerical precision can be reduced heavily before rounding errors start to reduce the quality of the used model. In Sect. 5, we compare the possible gain in resolution against the possible increase in model error when inexact hardware is used. In Sect. 6, we compare the stochastic forcing of the stochastic parametrisation scheme against the rounding error forcing at different levels of precision. In Sect. 7, we describe similarities between rounding errors and random noise, engineer rounding errors towards the distribution of the stochastic forcing of the stochastic sub-grid-scale parametrisation scheme and show results for simulations that replace the stochastic forcing by the engineered rounding errors. In Sect. 8, we show that rounding errors can be used to generate ensemble simulations that are similar to ensemble forecasts based on stochastic parametrisation schemes. In Sect. 9, a summary and the conclusions are present.

2 1D Burgers equation with stochastic forcing and stochastic sub-grid-scale closure

This section presents the model for simulations of the 1D Burgers equation. We use the stochastic sub-grid-scale parametrisation scheme by [5] based on the MTV stochastic mode reduction strategy by **M**ajda, **T**imofeyev and **V**anden-Eijnden [19]. The so-called MTV strategy separates the modes in the system into resolved and unresolved ones based on their characteristic timescales. This information is used to derive deterministic and stochastic terms to correct the dynamics of resolved modes for interactions to unresolved modes. In contrast to previous studies that used the MTV method for global basis functions (for example [18] and [11]), [5] applied this strategy locally to a finite-difference grid point model. This allows an application of the method to very high-dimensional systems and to consider relatively large numbers of resolved modes (see also [6]).

The 1D Burgers equation with dissipation and random forcing f is given by:

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(\frac{u^2}{2} - \nu\frac{\partial u}{\partial x}\right) = f\,(x,t)\,, \tag{1}$$

where u is velocity and ν is the constant diffusion coefficient. The domain is periodic with a length L. Following [5], the domain is divided into a fine grid with 512 equidistant grid points. Additionally, the domain is also divided into 32 equidistant coarse grid cells, such that 16 gridpoints of the fine grid are situated within each coarse-grained grid cell. The velocity at each fine grid cell is represented as a sum of the mean value x_i inside the coarse grid cell and the local variation y_j:

$$u_j = x_i + y_j \qquad \text{and} \qquad x_i = \frac{1}{16} \sum_{k=16\cdot i}^{16\cdot(i+1)-1} u_k. \tag{2}$$

We discretise Eq. (1) in space using an energy conserving finite-difference discretisation for the quadratic nonlinearities and rewrite the discrete model equation using the large- and small-scale variables x_i and y_j. We use a stochastic forcing f that has the shape of large-scale waves with wavenumbers 1, 2, and 3 over the entire domain. The forcing will act on the large-scale parameters x_i only. According to [5], the new set of equations reads:

$$\begin{aligned}\dot{x}_i =& F_i^x + \sum_{k=0}^{31} L_{ik}^{xx} x_k + \sum_{k=0}^{31}\sum_{l=0}^{31} B_{ikl}^{xxx} x_k x_l + \sum_{k=0}^{31}\sum_{l=0}^{511} B_{ikl}^{xxy} x_k y_l \\ &+ \sum_{k=0}^{511}\sum_{l=0}^{511} B_{ikl}^{xyy} y_k y_l + \sum_{k=0}^{511} L_{ik}^{xy} y_k,\end{aligned} \tag{3}$$

$$F_i^x = \sum_{k=1}^{3} \frac{\sqrt{2}\alpha_k}{100\cdot\sqrt{k\Delta t}} \cos\left(2\pi\left(\frac{kin\Delta x}{L} + \phi_k\right)\right), \tag{4}$$

$$\begin{aligned}\dot{y}_j =& \sum_{k=0}^{31} L_{jk}^{yx} x_k + \sum_{k=0}^{31}\sum_{l=0}^{31} B_{jkl}^{yxx} x_k x_l + \sum_{k=0}^{31}\sum_{l=0}^{511} B_{jkl}^{yxy} x_k y_l \\ &+ \sum_{k=0}^{511}\sum_{l=0}^{511} B_{jkl}^{yyy} y_k y_l + \sum_{k=0}^{511} L_{jk}^{yy} y_k,\end{aligned} \tag{5}$$

with random numbers α_k and ϕ_k and $\Delta x = L/512$. The terms $B^{\alpha\beta\gamma}$ cover quadratic nonlinearities. The $L^{\alpha\beta}$ terms result from the dissipation term. For both $B^{\alpha\beta\gamma}$ and $L^{\alpha\beta}$, the first subscript α denotes the mode onto which they project and β and γ indicate the modes involved. The exact terms are listed in "Appendix" (see [5] for more details). The summations in the equation suggest that we consider global interactions. However, the interactions in the used finite-difference scheme are local and the resulting tensors are sparse. We work with a Runge–Kutta time-stepping scheme of third order and choose a time step of $\Delta t = 0.01$ and $\nu = 0.02$.

We simulate the system either with the full set of large- and small-scale parameters x_i and y_j, or in a parametrised setup for which the small-scale parameters y_j are truncated and the system is reduced to the large-scale parameters x_i. We follow the nomenclature in [5] and denote simulations of the full set of Eqs. (3)–(5) as direct numerical simulations (DNS). To parametrise the y_j variables, we use the stochastic sub-grid-scale closure parametrisation scheme by [5] as standard configuration and a deterministic Smagorinsky closure for comparison. The model with Smagorinsky parametrisation solves the following equation:

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(\frac{u^2}{2} - \nu\frac{\partial u}{\partial x}\right) = f(x,t) - 0.5\frac{\partial \tau}{\partial x}, \qquad \text{with} \qquad \tau = -2C_s\Delta_x^2\left|\frac{\partial u}{\partial x}\right|\frac{\partial u}{\partial x}, \tag{6}$$

on the coarse grid with the Smagorinsky constant $C_s = 0.2$. We denote simulations that use Smagorinsky closure with the acronym "smg", following [5].

For the sub-grid-scale parametrisation scheme derived in [5], the full system [Eqs. (3)–(5)] is reduced to:

$$
\begin{aligned}
dx_i(t) = & \sum_{k,l=i-1}^{i+1} B_{ikl}^{xxx} x_k x_l dt + \sum_{k=i-1}^{i+1} L_{ik}^{xx} x_k dt + F_i^x dt \\
& + \sum_{k=i-2}^{i+2} M_{ik} x_k dt + \sum_{k,l=i-2}^{i+2} Q_{ikl} x_k x_l dt + \sum_{k,l,m=i-2}^{i+2} C_{iklm} x_k x_l x_m dt \\
& + \sigma^{(1)} dW_i^{(1)} + \sigma_i^{(2)}(x) dW_i^{(2)}. \qquad (7)
\end{aligned}
$$

Here, $W^{(1)}$ and $W^{(2)}$ denote vectors with independent Wiener processes. The first three terms of the right-hand side of the equation discretise the 1D Burgers equation without any sub-grid-scale closure. The fourth, fifth and sixth term represent the deterministic part of the parametrisation scheme with linear, quadratic and cubic corrections. The last two terms form the stochastic part of the parametrisation scheme with additive and multiplicative noise. The specific form of the terms can be found in Appendix D of [5].

3 Emulated reduced precision

In this section, we provide a short introduction to the representation of real numbers as floating points and the used emulator for reduced precision. Today's hardware is typically only capable to represent floating-point numbers in single (32 bits) or double (64 bits) precision. According to the IEEE754 standard [15], a double precision floating-point number ϵ is represented by a sequence of 64 bits. Each bit can take the value 0 or 1. The 64 bits comprise a sign bit (s), eleven bits for the exponent ($c_0, c_1, c_2, ..., c_{10}$) and 52 bits for the mantissa or significand ($b_{-1}, b_{-2}, \ldots, b_{-52}$). The relationship between ϵ and its bit representation is given by

$$
\epsilon = (-1)^s \left(1 + \sum_{i=1}^{52} b_{-i} 2^{-i}\right) 2^E, \qquad \text{where} \qquad E = \left(\sum_{i=0}^{10} c_i 2^i\right) - 1023.
$$

In this paper, we investigate the use of hardware that allows the use of floating-point numbers at various levels of precision. We focus on a reduced number of bits in the significand. Although there are several approaches towards building such hardware (see Sect. 1), it is still only available for use by us in form of FPGAs that are very difficult to programme. Therefore, we need to emulate reduced precision arithmetic within the numerical simulations. The used emulator will round results of floating-point operations to the closest number that can be represented with a reduced number of bits. The resulting rounding error is slightly different to the expected rounding error on real reduced precision floating-point hardware, for which intermediate steps in the calculation of a floating-point operations will also be represented by a reduced number of bits and will therefore also be affected by rounding errors. Guarding bits do, however, keep rounding errors for intermediate steps to a minimum. We therefore expect differences between the emulated and the real hardware rounding to be small.

In simulations with emulated reduced precision, the emulator is acting on all floating-point operations within the entire model run. This includes the calculation of the resolved scales but also the sub-grid-scale parametrisation scheme.

4 Simulations of the Burgers equation with reduced floating-point precision

This section presents results for simulations with the model with stochastic closure on emulated reduced precision hardware. We compare results for simulations with reduced precision to the control simulation in double precision. We will use the same diagnostics as [5].

Figure 1 shows the energy spectra, the autocorrelation function and the kurtosis[1] for the sub-grid-scale parametrised simulations with reduced significand, the control simulation with sub-grid-scale parametrisation in double precision, the simulation with Smagorinsky closure and the direct numerical simulation. All simulations are run for 600,000 nondimensional units. Table 1 presents results for the variance and the integrated autocorrelation function. While simulations with 8 bit significand are clearly perturbed, the simulations with more bits (10 and 12) do not show strong differences to the double precision control simulation for all diagnostics, except for small changes of the kurtosis for 10 bits in the significand. Differences between the double

[1] The kurtosis is the normalised lagged fourth-order moment. Kurtosis is a measure for deviations from Gaussianity.

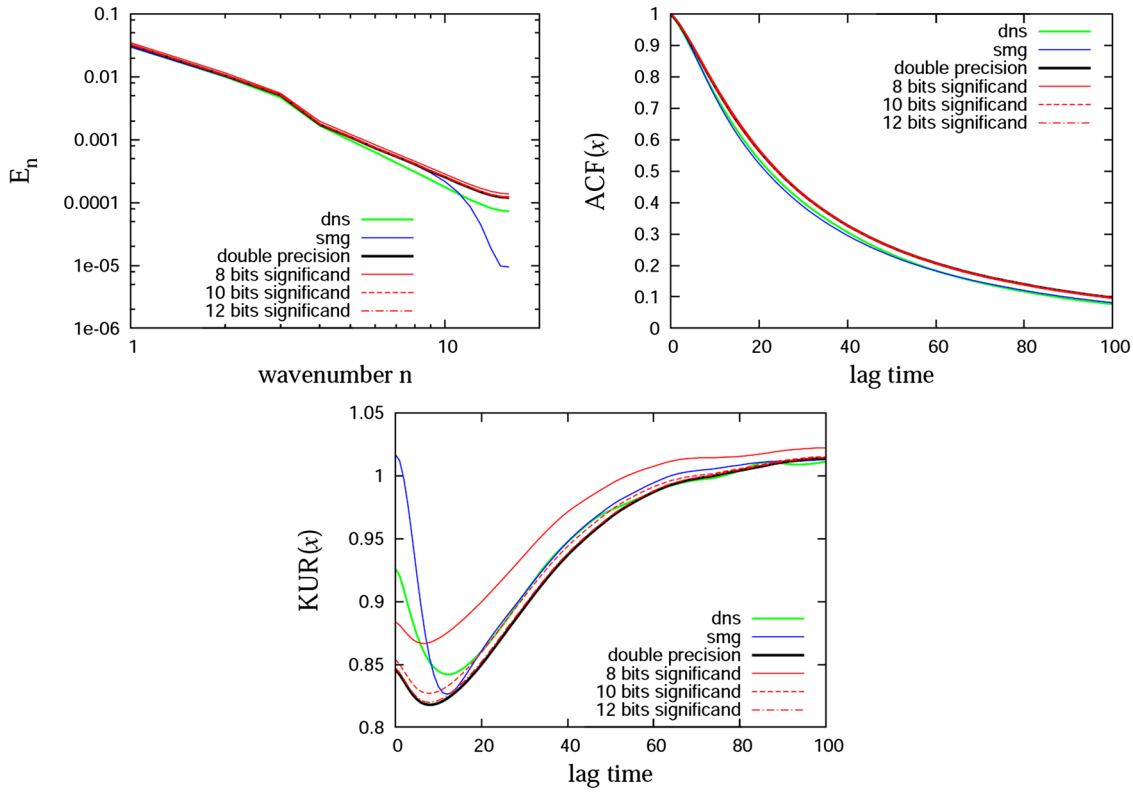

Fig. 1 Energy spectra (*left*), spatially averaged time autocorrelation function (*right*) and kurtosis (*bottom*) for the direct numerical simulation (DNS), the simulation with Smagorinsky parametrisation (smg), the sub-grid-scale parametrised simulation in double precision (double precision), and the sub-grid-scale parametrised simulations with emulated low precision and different numbers of bits in the significand

Table 1 Variance (VAR) and integrated autocorrelation function (Int. ACF) for model simulations with different precision

	DNS	smg	Double prec.	8 bit sig.	10 bit sig.	12 bit sig.
VAR	0.0250	0.0252	0.0268	0.0293	0.0271	0.0270
Int. ACF	144.1	144.9	159.1	159.5	158.0	159.5

precision control simulation and the reduced precision simulations with 10 and 12 bits are much smaller than differences between the control simulation and the direct numerical simulation which suggests that reduced precision will not have a strong influence on the quality of model simulations

5 Performance analysis

In the previous section, we tested how a reduction in precision will reduce the quality of simulations. In this section, we argue that the use of inexact hardware can be beneficial for model simulations. We present results to answer the question whether an increase in resolution, which is made possible by a reduction in the power consumption or an increase in performance due to reduced precision, can be beneficial for model simulations in such a way that the reduction in model quality due to reduced precision is balanced. It needs to be checked whether the level of precision that has been tested to be sufficient for the original model setup is also sufficient if resolution is increased. We consider a setup with emulated reduced floating-point precision to 19 bits (12 bits for the significand, 6 bits for the exponent and one sign bit). The number of bits is significantly reduced in comparison with double precision (52 bits for the significand, 11 bits for the exponent and one sign bit) and single precision (23 bits for the significand, 8 bits for the exponent and one sign bit). An estimate for a possible cost reduction with reduced precision that seems to be fair is the ratio between the number of bits that

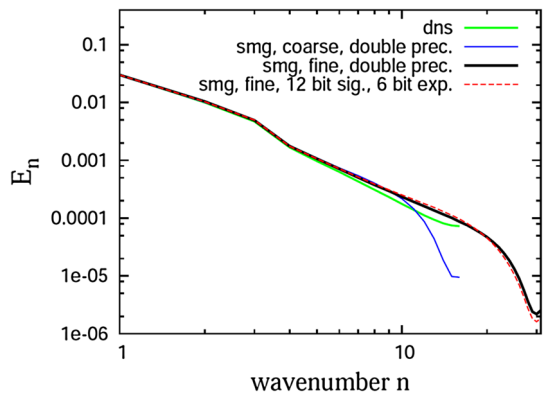

Fig. 2 Energy spectra for simulations with double and reduced precision for the standard setup (coarse) and a setup with higher resolution (fine)

can be used. This will certainly depend on the used approach to inexact hardware (see list in introduction). If FPGAs are used, the speed-up will be approximately proportional to the ratio of the number of bits used. An inexact CPU setup with a pruned floating-point unit and inexact memory might even scale better than the ratio between the number of bits [7,10]. We conclude that *the reduced precision setup will be more than three times faster* compared to the standard double precision simulation ($64/19 \approx 3.4$).

Figure 2 shows results for the direct numerical simulations and for simulations with Smagorinsky closure calculated either with double or with reduced precision at two different resolutions. While the standard setup is using 32 grid points, the "high"-resolution setup is using 64 grid points. Both setups are compared to the direct numerical simulation with 512 grid points, the latter projected on 32 coarse grid cells. Due to the increase in resolution by a factor of two, we expect computational cost *to increase approximately by a factor of four* (a factor of two for each dimension in space and time). Therefore, simulations with reduced precision on the fine grid will be only slightly more expensive than double precision simulations on the coarse grid. However, results in Fig. 2 show that an increase in resolution will improve model simulations significantly between wavenumbers 10 and 16 since the impact of diffusion can be pushed to higher wavenumbers. The energy spectra of the simulation with reduced precision are almost identical to the spectra of the high-resolution simulation with double precision, despite the large difference in computational cost.

We used simulations with the Smagorinsky scheme for comparison, since it is difficult to make fair estimates of the computational cost if the stochastic sub-grid-scale scheme is used. Since the stochastic closure model produces a continuous power-law slope in the inertial range, unlike the Smagorinsky model, we do not expect significant improvement in this spectral range, if we run the model with increased resolution but with reduced precision. However, we expect a better representation of the statistical moments, as suggested from Tables 2 and 3 from [5], which show that the relative error in the moments decreases for increasing resolution.

6 Analysis of the rounding error forcing in comparison to the stochastic forcing of the parametrisation scheme

To learn more about the forcing due to rounding errors when using inexact hardware, we study the properties of rounding errors and compare the stochastic forcing of the stochastic parametrisation scheme with rounding errors at different levels of precision for the significand of floating-point numbers. We test whether rounding errors can be hidden in the stochastic forcing of the parametrisation scheme.

In a first approach, we compare the stochastic forcing of the stochastic parametrisation scheme with the noise pattern that is present if a time step is calculated at different levels of precision in Fig. 3. To generate the Figure, we run the model and perform each time step in three different settings:

1. We calculate the next time step with the parametrised model in double precision (full Eq. (7)).
2. We calculate the same time step with the parametrised model in double precision omitting the stochastic part of the parametrisation scheme (the terms $\sigma^{(1)} dW_i^{(1)}$ and $\sigma_i^{(2)} dW_i^{(2)}$ in Eq. (7)).
3. We perform the same step as in setting 1. with emulated reduced precision.

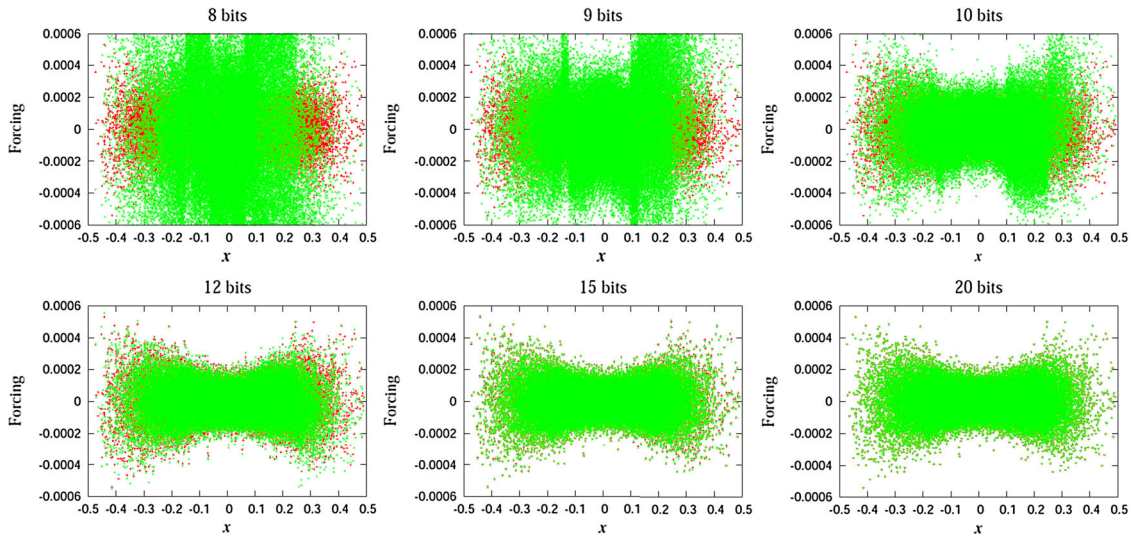

Fig. 3 Stochastic forcing of the sub-grid-scale parametrisation scheme in the control simulation (*red*) and stochastic forcing plus rounding error forcing when the time step is calculated with reduced precision to 8, 9, 10, 12, 15 and 20 bits in the significand of floating-point numbers (*green*, *top left* to *bottom right*) plotted against the velocity value x at the specific grid point (colour figure online)

Table 2 Mean and standard deviation for the stochastic forcing of the sub-grid-scale parametrisation scheme in the control simulation and the stochastic forcing plus rounding error forcing when the time step is calculated with reduced precision in the significand, and the correlation between the two forcings

	Mean	SD	Correlation
Stochastic forcing	−2.49E−8	9.27E−5	1.0000
8 bit significand	−3.33E−5	4.85E−4	0.0685
9 bit significand	6.36E−6	2.65E−4	0.3040
10 bit significand	−7.48E−6	1.58E−4	0.5887
12 bit significand	−1.95E−6	9.82E−5	0.9448
15 bit significand	−2.85E−7	9.28E−5	0.9991
20 bit significand	−3.25E−8	9.27E−5	1.0000

In each setting, the next time step will use the state vector calculated in setting 1. as initial condition. We can now compare the stochastic forcing of the parametrisation scheme (contribution of right-hand side for setting 1. minus contribution of right-hand side for setting 2.) with the noise caused by rounding errors added to the stochastic forcing of the parametrisation scheme (contribution of right-hand side for setting 3. minus contribution of right-hand side for setting 1.) for each time step. Fig. 3 shows that rounding errors decrease with increasing number of bits in the significand, as expected. Changes in the distribution due to rounding errors are hardly visible for 12, 15 and 20 bits in the significand, while the rounding error generates a larger spread of the noise for inexact hardware with 8,9 or 10 bits in the significand. Table 2 confirms these results since simulations with $\geq$12 bits in the significand show a correlation between the reduced precision and the standard random forcing of more than 0.9 and a change in the standard deviation by less than 6%.

Figure 4 shows the probability distributions of the forcing pattern for certain intervals of the x variable in Fig. 3. The distributions of rounding errors have a reasonable shape close to a Gaussian distribution, and the results confirm that the stochastic forcing is closely matched for 12 bits in the significand, while the distributions are clearly different for 9 or 10 bits in the significand. The distributions with inexact hardware are not centred perfectly around zero which might cause problems.[2] However, the mean of the entire forcing is small, as seen in Table 2.

[2] Rounding errors with nonzero mean can appear if the distribution of results of a specific floating-point operation does not have the same probability for "rounding up" and "rounding down" or due to the nonlinear terms of the model.

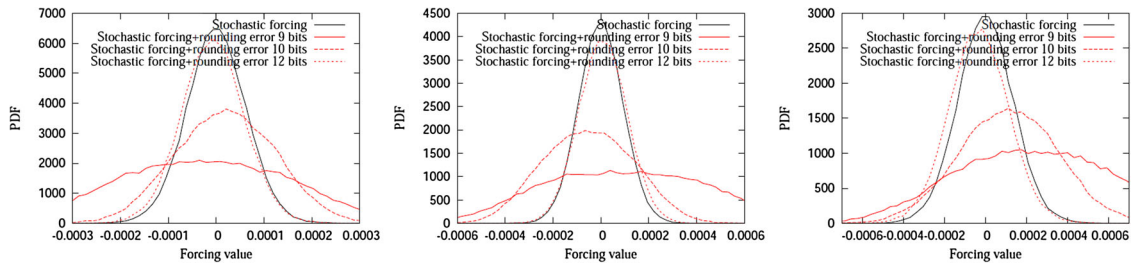

Fig. 4 PDF of the forcing due to rounding errors plus stochastic forcing for intervals of the prognostic velocity parameter (x in Figure 3) between 0.02 and 0.05, 0.15 and 0.2, and 0.25 and 0.5 (from *left* to *right*) for an emulated significand with 9, 10 or 12 bits, compared to the stochastic forcing of the sub-grid-scale parametrisation scheme only

From the results in Figs. 3, 4 and Table 2, we would expect that model simulations with more than 10 bits in the significand will be hardly affected by rounding errors since the errors will be hidden within the stochastic forcing of the sub-grid-scale parametrisation, while there might be changes to model results for $\leq$10 bits in the significand.

This result fits nicely to the conclusion from Fig. 1 and Table 1 that show that model simulations can be disturbed for less than 10 bits in the significand, while the quality of model simulations is comparable for larger numbers of bits. We conclude that rounding errors can be hidden in the noise pattern for stochastic parametrisation schemes if the rounding error forcing is smaller than the stochastic forcing caused by stochastic parametrisation schemes. If the rounding error forcing does not exceed the stochastic forcing, the quality of the model is hardly influenced by rounding errors. The magnitude of the stochastic forcing can serve as a first guess for the maximal rounding error that is acceptable before model results will be changed significantly.

7 The use of rounding errors to represent sub-grid-scale variability

In this section, we discuss the properties of rounding errors in general. We show similarities between rounding errors and random noise and exploit these similarities by engineering rounding errors that mimic the stochastic forcing of the sub-grid-scale parametrisation scheme. Finally, we perform simulations that replace the stochastic forcing by the rounding error of inexact hardware. We compare results of these simulations with model simulations of the full stochastic parametrisation scheme in double precision.

Figure 5 shows the noise pattern when random input numbers (x-axis) are rounded to either four or ten bits in the significand. Rounding errors show linear pattern (see panel a in Fig. 5) with jumps if one input value is rounded down, while its right neighbour is rounded up. The entire process of rounding is deterministic in nature, and it might be misleading to assign properties of random noise to the noise pattern of rounding errors. However, if the input values cover only the range of one exponent and if the distance between the "rounding-up" and "rounding-down" interval is very small compared to the range of input variables, the pattern shows no visible difference to additive white noise with uniform distribution (compare b and c in Fig. 5). The linear noise patterns would obviously be recovered if we would zoom into a smaller range of x-variables, or if we would consider a much larger sample of points. However, we can assume that the rounding error—and therefore the "rounding-up" and "rounding-down" interval—needs to be relatively small in comparison with the range of the signal, to allow reasonable results for model simulations. The range of the rounding error is fixed by the value of the exponent E and the number of bits m that represent the significand of floating-point numbers. The rounding error ζ for a random number is limited by $-\frac{2^E}{2^{(m+2)}} < \zeta \leq \frac{2^E}{2^{(m+2)}}$.

If the range of input values covers several exponents, the maximal rounding error is changing with the exponent by a factor of two with each step. The length of the interval in which a specific exponent is valid is also changing by a factor of two with each step. A straight line can therefore describe the position of all points at which the maximal absolute rounding error is changing with the exponent. This line is described by the equation $h(x) = \frac{x}{2^{(m+1)}}$ and plotted in Fig. 5 d and e. The range of the noise pattern is obviously discontinuous at changes of the exponent. However, the noise pattern has similar properties to multiplicative noise (compare d and e with f in Fig. 5).

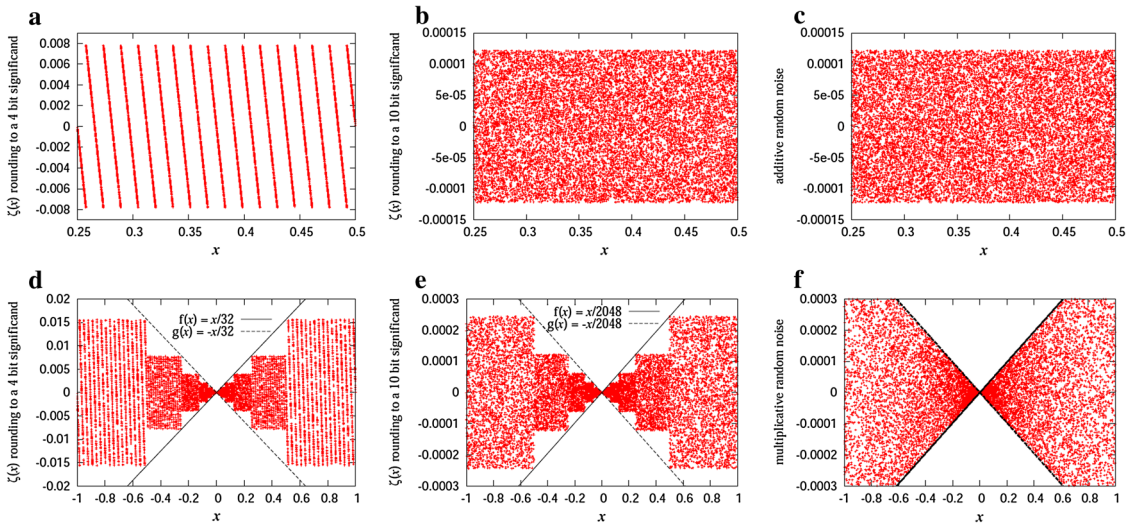

Fig. 5 The rounding error ζ calculated for random double precision input values x when rounding to a significand of 4 bits (*left*) or 10 bits (*middle*) for two different intervals of random input values (*top and bottom*). The plots on the right show additive (*top*) and multiplicative (*bottom*) white noise of equivalent magnitude

According to a profiler,[3] the calculation of the additive and multiplicative noise term of the stochastic parametrisation scheme [the terms $\sigma^{(1)} dW_i^{(1)}$ and $\sigma_i^{(2)} dW_i^{(2)}$ in Eq. (7)] is creating approximately two-thirds of the computational cost of the entire model simulation in terms of computing time.[4] It can be assumed that the calculation of the stochastic forcing will not create a significant ratio of the total cost for model simulations of a full atmosphere or ocean model. Still, for the model under investigation computing time will be reduced significantly if it is possible to replace the stochastic forcing of the sub-grid-scale parametrisation scheme with an engineered rounding error of inexact hardware, additional to savings due to the use of inexact hardware.

If we assume that the stochastic forcing of the sub-grid-scale parametrisation scheme offers a correct representation of sub-grid-scale variability and if we can replace this forcing by engineered rounding errors, we would show that rounding errors can be used to represent sub-grid-scale variability and *can therefore be beneficial* for the given model.

However, we are not aware of an effective method to create rounding error forcings that mimic arbitrary multiplicative noise terms involving interactions between different modes. Therefore, the rounding error in this paper will never be able to show the same quality as the stochastic forcing of the sub-grid-scale parametrisation, and an important and cost-intensive ingredient in the generation of the noise is missing. However, in the present application, it turned out that omitting those interactions in the rounding error forcing does not play an important role.

To get a better impression of the magnitude and the shape of the noise caused by rounding errors when the calculation of the stochastic noise is not included, we run a model and perform each time step in three different settings:

1. We calculate the next time step with the parametrised model in double precision [full Eq. (7)].
2. We calculate the same time step with the parametrised model in double precision omitting the stochastic part of the parametrisation scheme [the terms $\sigma^{(1)} dW_i^{(1)}$ and $\sigma_i^{(2)} dW_i^{(2)}$ in Eq. (7)].
3. We perform the same step as in setting 2. with emulated reduced precision.

The next time step is based on the new state vector calculated in setting 1. for each setting. We can now compare the stochastic forcing of the parametrisation scheme (contribution of right-hand side for setting 1. minus contribution of right-hand side for setting 2.) with the noise caused by rounding errors (contribution of right-hand side for setting 3. minus contribution of right-hand side for setting 2.) for each time step. Fig. 6 and Table 3 compare the two different forcings for simulations with different numbers of bits in the significand. Rounding errors decrease with increasing the number of bits, as expected. While the rounding errors are smaller

[3] gprof applied to the model compiled with the Intel Fortran compiler and -O3 optimisation.

[4] This is due to the relatively large number of interactions entering the multiplicative noise term amplitude [see Eq. (61) in [5].]

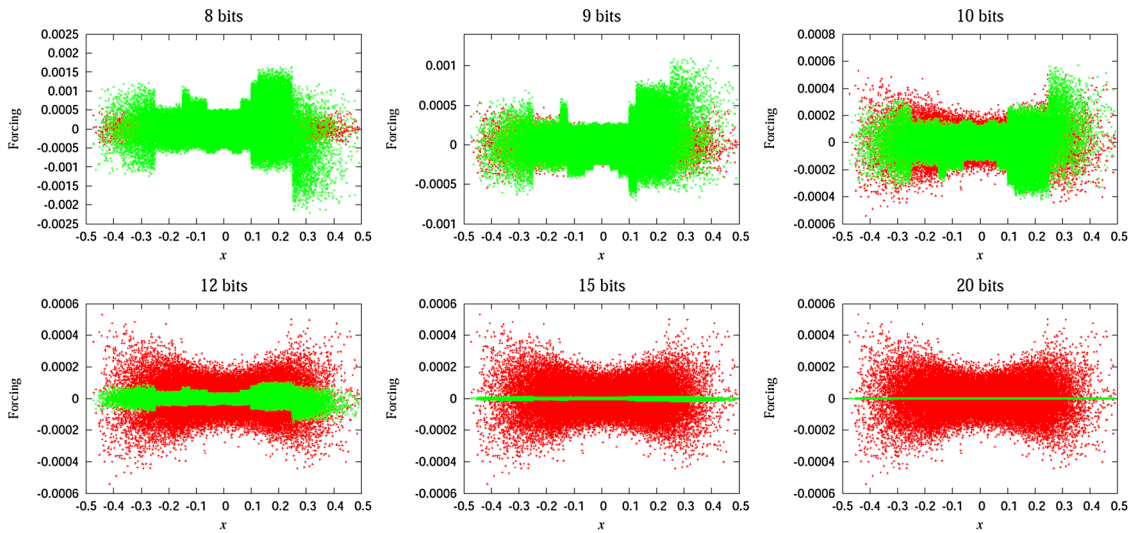

Fig. 6 Noise pattern caused by numerical rounding to a reduced number of bits in the significand compared to the random noise of the stochastic forcing of the sub-grid-scale parametrisation scheme for 8, 9, 10, 12, 15 and 20 bits (*top left* to *bottom right*) plotted against the velocity value at the specific grid point

Table 3 Mean and standard deviation for the stochastic forcing and the forcing due to rounding errors to a reduced number of bits in the significand

	Mean	SD
Stochastic forcing	−2.49E−8	9.27E−5
8 bit significand	4.94E−5	4.51E−4
9 bit significand	2.40E−5	2.27E−4
10 bit significand	−1.25E−5	1.14E−4
12 bit significand	3.14E−6	2.86E−5
15 bit significand	3.75E−7	3.57E−6
20 bit significand	8.27E−9	1.12E−7

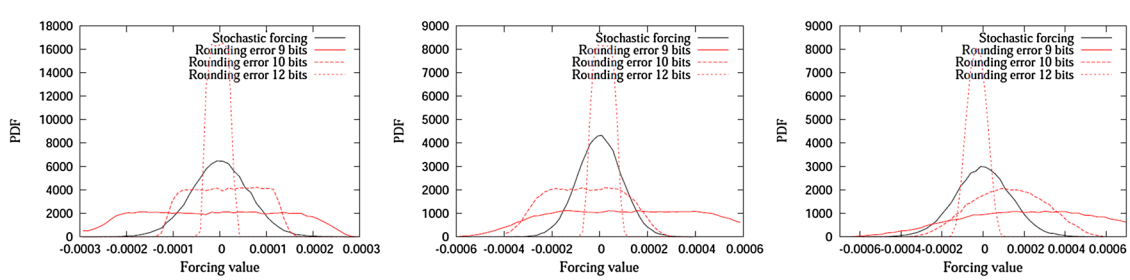

Fig. 7 PDF of the forcing due to rounding errors for intervals of the prognostic velocity parameter (x in Fig. 6) between 0.02 and 0.05, 0.15 and 0.2, and 0.25 and 0.5 (from *left* to *right*) for an emulated significand with 9, 10 or 12 bits, compared to the stochastic forcing of the sub-grid-scale parametrisation scheme

or of similar magnitude in comparison with the stochastic forcing for 10, 12, 15 and 20 bits in the significand, rounding errors exceed the stochastic forcing for 8 or 9. bits in the significand.

Figure 7 shows the probability distributions of the forcing pattern for certain intervals of the x variable in Fig. 6. Most of the distributions of the rounding error have a reasonable shape close to a Gaussian distribution. Although the distributions are not always centred around zero, the total mean error averaged over all x is small (see Table 3).

We will engineer the pattern of rounding errors to match the stochastic forcing of the sub-grid-scale parametrisation scheme as close as possible. Basically, we want the green distribution in Fig. 6 to match the red distribution. As discussed above, rounding errors can behave similar to both additive and multiplicative

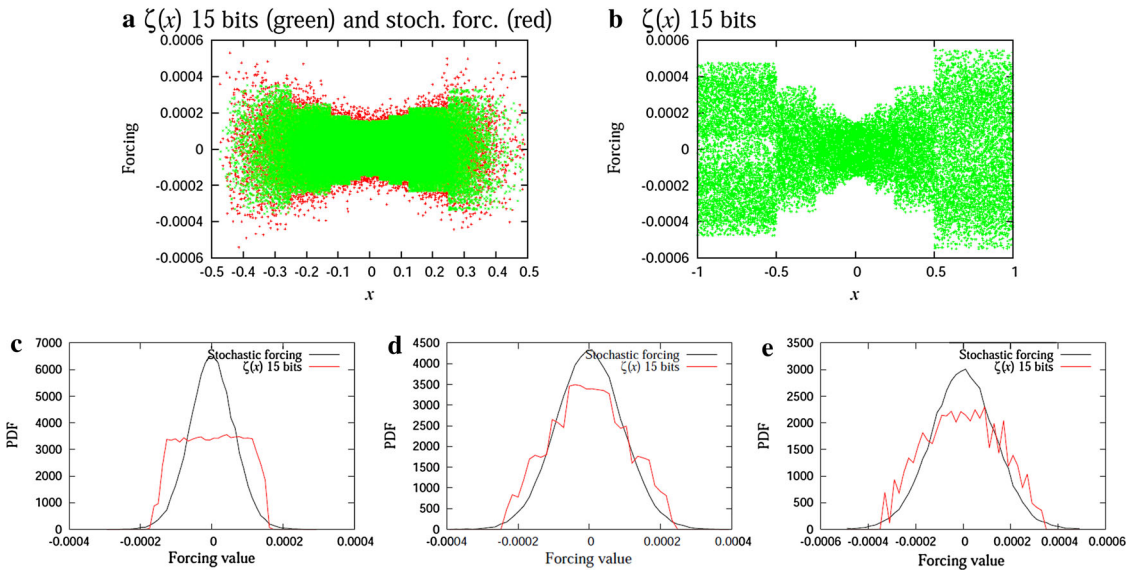

Fig. 8 **a** Noise pattern of the engineered rounding errors (*green*) compared to the stochastic forcing of the sub-grid-scale parametrisation scheme (*red*). The rounding errors were engineered by tuning α and β in Eqs. (8) and (9) to fit the random noise as good as possible. **b** Error distribution of the engineered forcing if the distribution of x changes from $-0.5 < x < 0.5$ to $-1.0 < x < 1.0$. **c**, **d**, **e** Error distribution of the forcing due to engineered rounding errors for intervals of velocity (x in the *plots* above) between 0.02 and 0.05, 0.15 and 0.2, and 0.25 and 0.5 (from *left* to *right*) (colour figure online)

noise. Since we cannot reduce the rounding error for a given number of bits in the significand, we will start from the setup with 15 bits in the significand, for which the rounding error is much smaller than the stochastic forcing (see Fig. 6). Our approach assumes that the used hardware can work with multiple levels of exactness. We perform the following operations to obtain additive ζ_a and multiplicative ζ_m noise:

$$\zeta_a(x_i) = \alpha \cdot (r(r(x_i + 2.5) - 2.5) - x_i) \tag{8}$$

$$\zeta_m(x_i) = \beta \cdot (x_i - r(x_i)) \tag{9}$$

The function $r(\ldots)$ denotes a rounding to 11 bits in the significand. All other operations are performed with emulated 15 bits in the significand. Both terms would be zero if no rounding errors were present. We add 2.5 to x_i for additive noise, to shift the range of the x_i values to make them share the same exponent. For multiplicative noise, we can simply round the parameter value to its representation with 11 bits in the significand and subtract the original value of x_i since the range of parameters is symmetric around zero and spread over a wide range of exponents. We use $\alpha = 0.3$ and $\beta = 3.6$ to get a proper adjustment of the noise.

Figure 8 shows the error patterns that are engineered to fit the stochastic forcing induced by the stochastic parametrisation scheme. The engineered forcing has a standard deviation of 1.05E−4. This is relatively close to the mean and standard deviation of the stochastic forcing (9.27E−5). The mean of the engineered forcing is small as well (−9.62E−7). The engineered noise will not change its behaviour drastically if the prognostic parameters x_i leave the range between $-0.5 < x < 0.5$, as can be seen in panel *b* in Fig. 8. Figure 9 shows the correlation of the stochastic forcing of the sub-grid-scale parametrisation scheme and the engineered rounding errors in space and time. As expected, it can be seen that no correlation is visible even to the nearest neighbours for both forcings. The slightly negative correlation in space for the stochastic forcing is caused by the constraint of zero mean for the stochastic forcing in each time step.

Finally, we perform model simulation that use the engineered rounding error to replace the stochastic forcing of the stochastic parametrisation scheme. Results for the energy spectra, the autocorrelation function and the kurtosis are plotted in Fig. 10. The values for variance and integrated autocorrelation function are given in Table 4. Only the kurtosis is disturbed slightly, the other climate-type diagnostics show good results for the simulations with engineered rounding errors, especially when compared against the differences between the sub-grid-scale parametrised control simulation in double precision and the direct numerical simulation. Figure 10 is also showing the mean forecast error in x for simulations on the coarse grid. The forecast error is calculated against direct numerical simulations:

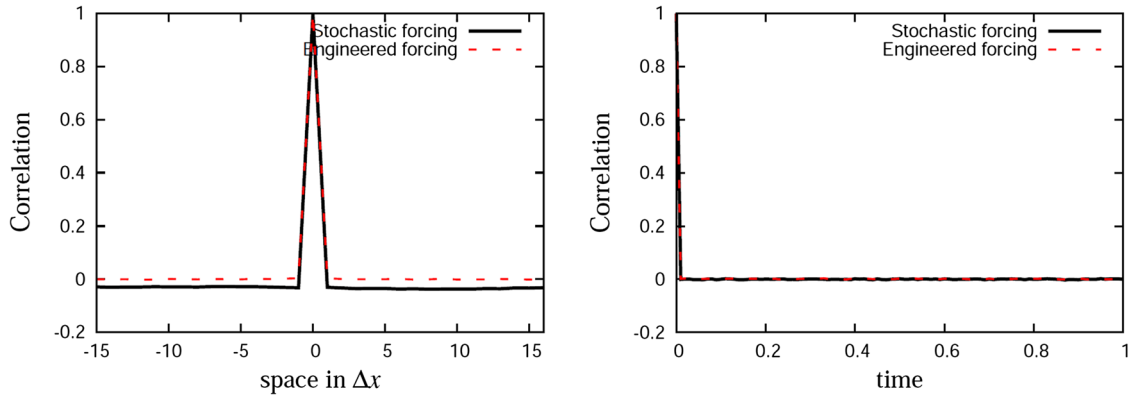

Fig. 9 Correlation in space (*left*) and time (*right*) for the stochastic forcing of the sub-grid-scale parametrisation scheme and the engineered rounding error forcing calculated from 1,000,000 time steps

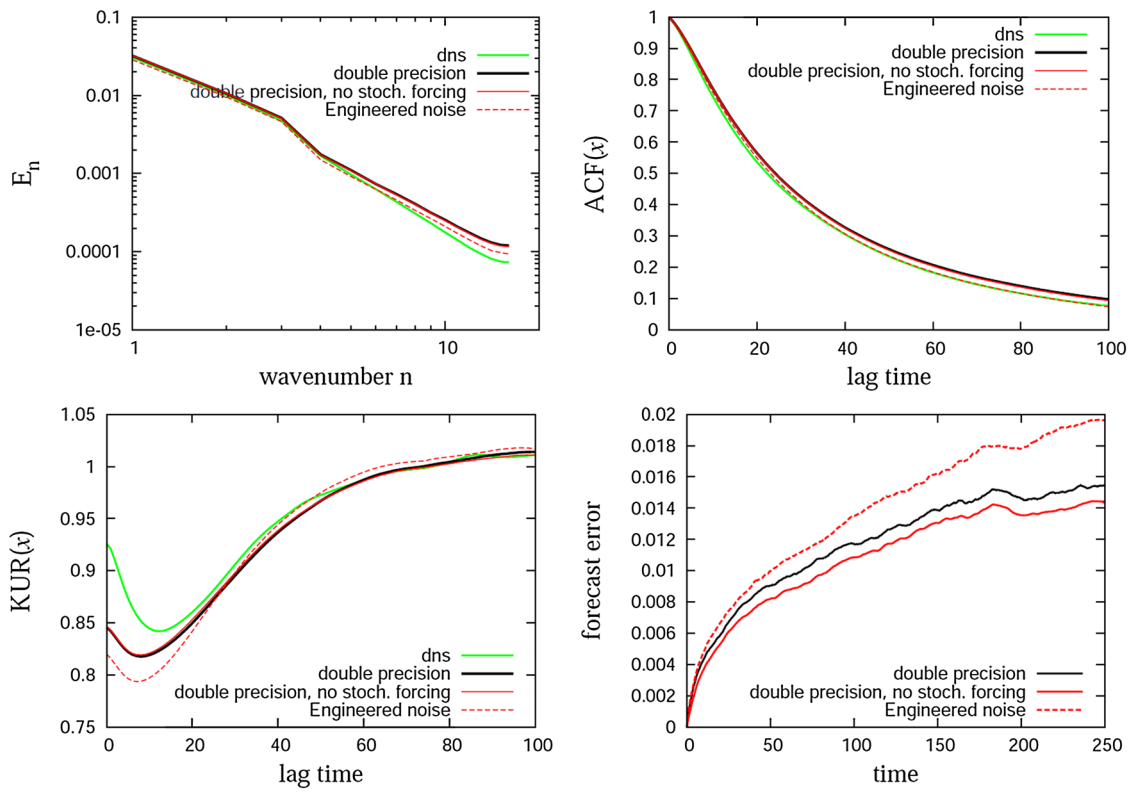

Fig. 10 Energy spectra (*top left*), autocorrelation function (*top right*), kurtosis (*bottom left*) and forecast error (*bottom right*) for the direct numerical simulation (DNS), the sub-grid-scale parametrised simulation with double precision, the double precision simulation with no stochastic forcing and the sub-grid-scale parametrised simulation with engineered rounding errors that replace the stochastic forcing terms

$$\text{Forecast error}(t) = \frac{1}{32 \cdot N_f} \sum_{i=0}^{31} \sum_{j=1}^{N_f} abs\left(x_{i,j}^{dns}(t) - x_{i,j}^{c}(t)\right).$$

$N_f = 50$ is the number of evaluated forecasts, each separated by 1000 nondimensional units. $x_{i,j}^{dns}$ and $x_{i,j}^{c}$ are the mean values in the coarse grid cells for the direct numerical simulations and simulations on the coarse grid.

The increase in forecast error that can be seen with engineered rounding errors compared to the simulation with added stochastic forcing seems to be in a reasonable range given the possible reduction in computational

Table 4 Variance (VAR) and integrated autocorrelation function (Int. ACF) for the direct numerical simulation, the double precision control simulation, a simulation in double precision with no stochastic forcing and the simulation with engineered rounding errors

	DNS	Double precision	No stoch. forcing	15 bits, engineered
VAR	0.0250	0.0268	0.0268	0.0237
Int. ACF	144.1	159.1	157.2	144.7

cost. It should be mentioned that a model simulation that is calculated in double precision with no stochastic forcing provides reasonable results as well (see Fig. 10; Table 4). However, such models can only be used in ensemble methods using perturbed initial conditions. The forecast error is actually increased slightly for simulations with stochastic forcing compared against simulations with no added stochastic forcing.

8 Inexact hardware used to generate ensemble simulations

An important property of stochastic parametrisation schemes is the ability to generate ensembles. Typically, different sets of random numbers are used to calculate the stochastic forcing in stochastic parametrisation schemes for different members of an ensemble of simulations, for example, by using different seeds in random number generation. The resulting difference in the stochastic forcing is causing a divergence of ensemble members in time. Since the stochastic forcing is typically adjusted to represent sub-grid-scale variability, the spread of the ensemble can provide important information about predictability and an estimate of the forecast errors. This is very important for numerical weather prediction. For deterministic parametrisation schemes, initial value perturbations need to be used to generate ensembles.

We argue that the use of inexact hardware allows similar ensemble setups compared to ensemble methods with stochastic parametrisation schemes. If the initial conditions are perturbed only slightly, within the order of magnitude of the rounding errors, the resulting forcing pattern due to rounding has no obvious correlation with the equivalent forcing pattern in an unperturbed simulation. To illustrate this, we consider the same random numbers that were used to generate the plots of rounding errors, when rounding to a significand of 4 bits in Fig. 5, for a second time. We assume that each x represents an initial condition of a simulation. We add a uniformly distributed random number η to x with $-m \leq \eta < m$ where m is the maximal amplitude of the rounding error. The result x_2 is taken as new "initial condition", and a rounding to the closest representation with 4 bits in the significand is performed. Figure 11 shows the rounding error ζ if x_2 is rounded to four bits in the significand, plotted against the initial value x. We use $m \approx 0.008$ for panel a and $m \approx 0.016$ for panel b. The resulting pattern shows no visible difference to white noise within the range of a given exponent and appears to be sufficiently uncorrelated compared to the original rounding error with no initial perturbation that showed clear "rounding-up/rounding-down" pattern (compare Figs. 11 to 5a and d). In difference to pure initial value ensembles, for which only the initial values are perturbed, the effective forcing due to rounding is changed throughout the simulation compared to the equivalent forcing in the unperturbed simulation. We expect two simulations with inexact hardware that use x and x_2 as initial conditions to diverge from each other in the same way as ensemble members that use stochastic parametrisation schemes with different seeds.

We calculate 50 ensemble simulations with 50 ensemble members each. The ensembles were initialised along a simulation of the model in 6000 nondimensional units distance and integrated for 250 nondimensional units. Figure 12 shows results for the mean ensemble standard deviation for ensembles based on different ensemble methods. The first ensemble is solely based on the stochastic forcing of the sub-grid-scale parametrisation scheme with different random seeds for the different ensemble members. All other ensembles have a tiny perturbation of a uniform distributed random number between $-0.0005 < \eta \leq 0.0005$ added to the initial condition of each prognostic variable x_i. It can be seen that the used initial perturbations are hardly influencing model simulations since simulations with the full sub-grid-scale parametrisation scheme with different random seeds and added initial perturbation behave in the same way to simulations without initial perturbation and since the sub-grid-scale parametrised simulations with no stochastic forcing show only a small standard deviation which is decreasing with time since the stochastic large-scale forcing is the same for all simulations. For the simulations in which the engineered rounding error is replacing the stochastic term of the sub-grid-scale parametrisation, we obtain an ensemble spread of similar shape and the same order of magnitude compared to the sub-grid-scale parametrised simulations with stochastic forcing.

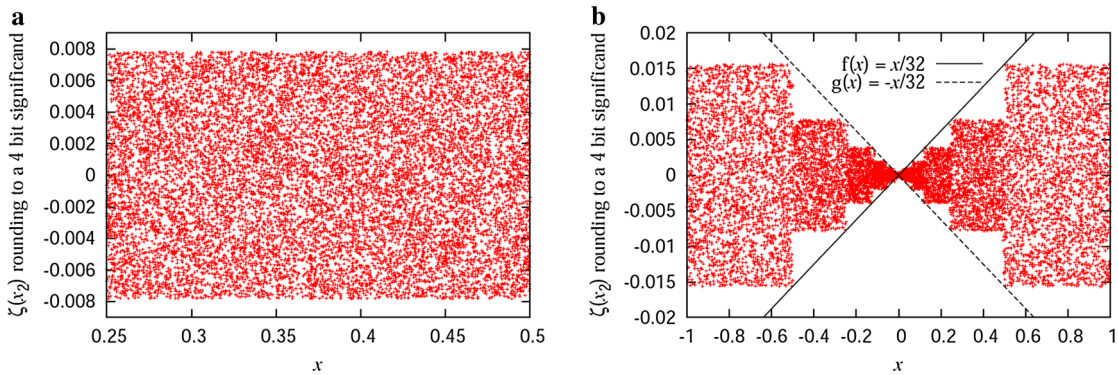

Fig. 11 Rounding error ζ when the perturbed initial condition $x_2 = x + \eta$ is rounded to 4 bits in the significand plotted against x. The same x values as in Fig. 5 are used

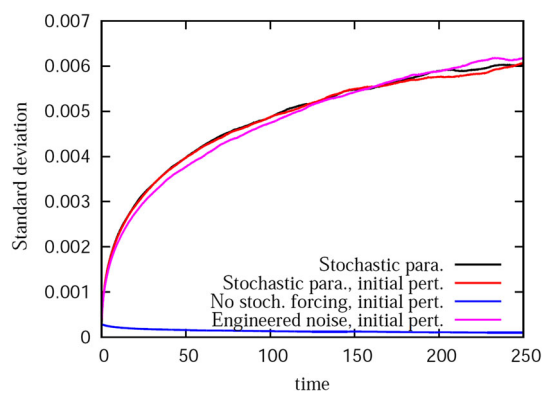

Fig. 12 Mean ensemble standard deviation plotted against time for ensemble simulations with 50 ensemble members, averaged over 50 ensemble simulations. The ensemble spread was caused by: *a* the stochastic parametrisation scheme, *b* the stochastic parametrisation scheme and an initial perturbation, *c* the initial perturbation only for simulations without stochastic forcing of the sub-grid-scale parametrisation, *d* the initial perturbation and the engineered rounding error term

If the results of Fig. 12 are compared against the plot of the forecast error in Fig. 10, it can be seen that the ensemble spread is under-dispersive, meaning that the forecast error has larger values than the ensemble spread. Ensemble forecasts will be over-confident since the ensemble spread will not be a realistic representation of the model error that should be expected. Given the underdispersiveness of the forecast, we argue that the small differences in ensemble standard deviations between the engineered rounding error forcing and the stochastic forcing are not significant. However, the use of the engineered rounding error forcing increased the forecast error slightly.

9 Summary and conclusion

We investigate the use of emulated inexact hardware to integrate a model of the randomly forced 1D Burgers equation with stochastic sub-grid-scale parametrisation. Results show that numerical precision can be reduced to only 12 bits in the significand of floating-point numbers—instead of 52 bits for double precision— with no serious degradation in results for all diagnostics considered (see Sect. 4). Simulations that use inexact hardware on a grid with higher spatial resolution show results that are significantly better compared to simulations in double precision on a coarser grid. The reduction in computational cost due to the use of reduced precision is approximated to be more than a factor of three in comparison with double precision simulations, while the use of higher resolution will increase computational cost by approximately only a factor of four (see Sect. 5).

We study the properties of the induced rounding errors and show for the 1D Burgers model that the stochastic forcing of the stochastic parametrisation scheme can be used to hide rounding errors and that the magnitude of the stochastic forcing can serve as a first guess for the upper limit of the magnitude of rounding errors that allow simulations with no significant change of the model statistics (see Sect. 6). We argue that rounding errors

can show similarities to additive and multiplicative noise when the interval of interest covers one or several values of exponents of floating-point numbers, as long as the range of rounding errors is small compared to the considered interval. The rounding error noise is almost uncorrelated for the model under investigation. We use the similarity to additive and multiplicative noise and engineer rounding errors in model simulations with inexact hardware to fit the distribution of the stochastic forcing of the stochastic parametrisation scheme as close as possible, by adding a small number of operations to the calculation of each prognostic variable in each time step (see Sect. 7). We use the engineered rounding errors to replace the stochastic forcing of the stochastic parametrisation scheme and obtain model results that show a similar quality compared to double precision control simulations with stochastic forcing. We show that rounding errors allow new perspectives for ensemble methods since a very small initial perturbation, within the order of magnitude of the rounding errors, is changing the rounding error forcing over the entire simulation. The rounding error forcing is sufficiently uncorrelated to form an ensemble of similar spread and quality compared to ensemble simulations that are based on stochastic forcings (see Sect. 8).

While the large potential for the use of inexact hardware in atmosphere models was already shown in previous studies [7–9], this is the first study to test the use of inexact hardware in a grid point instead of a spectral model. We show, in a very idealised setup, that rounding errors can improve numerical simulations in ensemble methods in the same way stochastic forcings of stochastic parametrisation schemes can do. Therefore, rounding errors are not necessarily degrading the quality of models in simulations of a system with chaotic dynamics. Instead, they can be beneficial for the representation of sub-grid-scale variability and can induce a reasonable spread in ensemble simulations. We do not argue that rounding errors will always be beneficial or that no stochastic parametrisation should be used in a model with reduced floating-point precision. Certainly, a first approach to the use of inexact hardware should always try to keep rounding errors as small as possible and random number generators are much easier to tune towards a desired forcing distribution, compared to rounding errors. The use of engineered rounding errors has certainly limitation: it is unlikely that it will be possible to tune rounding error forcings to replace very sophisticated stochastic forcings that show correlations in either space and time or state dependence (other than multiplicative noise) with no degradation in results. However, it is likely that a combination of rounding errors and stochastic forcings can be used in these cases. Furthermore, biases in rounding error patterns (see Fig. 4) need to be treated carefully. It is also obvious that results in this paper are very idealised, especially those with engineered rounding errors, and that robustness needs to be checked in larger and more realistic model setups. However, we believe that the increase in accuracy that would be possible when using inexact but ultra-efficient hardware will compensate for the increased model error due to rounding errors for weather and climate models and that an increased variability due to hardware errors *can* actually be beneficial for simulations of a system, such as the atmosphere, for which sub-grid-scale errors are inherently stochastic.

Acknowledgments We thank Tim Palmer and Krishna Palem for interesting discussions that have been essential for this paper. We also thank two anonymous reviewer for very useful feedback. The position of Peter Düben is funded by an ERC grant (Towards the Prototype Probabilistic Earth-System Model for Climate Prediction, project reference 291406).

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Appendix: Terms of the discrete two-scaled system of the Burgers equation

The following terms define the discretised system of the Burgers model and the used sub-grid-scale parametrisation scheme. No changes were made to the terms derived in [5]. In the following, the index c is the index of the coarse cell in which a small-scale parameter y_j is situated. cr and cl denote the index of the coarse cells in which either the right or the left neighbour of the small-scale parameter is located. jr and jl mark the index of the small-scale parameter which is located closest to a boundary between two coarse grid cells.
The terms that define quadratic interactions are:

$$\sum_{k=0}^{31}\sum_{l=0}^{31} B_{ikl}^{xxx} x_k x_l = -\frac{1}{6n\Delta x}\left(x_{i+1}^2 + x_i x_{i+1} - x_i x_{i-1} - x_{i-1}^2\right),$$

$$\sum_{k=0}^{511}\sum_{l=0}^{511} B_{ikl}^{xyy} y_k y_l = -\frac{1}{6n\Delta x}\left(y_{jr}^2 + y_{jr}y_{jr+1} + y_{jr+1}^2 - y_{jl}^2 - y_{jl}y_{jl-1} - y_{jl-1}^2\right),$$

$$\sum_{k=0}^{31}\sum_{l=0}^{511} B_{ikl}^{xxy} x_k y_l = -\frac{1}{6n\Delta x}\left(2x_i y_{jr} + x_i y_{jr+1} + x_{i+1}y_{jr} + 2x_{i+1}y_{jl+1}\right.$$
$$\left.-2x_i y_{jl} - x_i y_{jl-1} - x_{i-1}y_{jl} - 2x_{i-1}y_{jl-1}\right),$$

$$\sum_{k=0}^{31}\sum_{l=0}^{511} B_{jkl}^{yxy} x_k y_l = -\frac{1}{6n\Delta x}\left(2x_{cr}y_{j+1} + x_{cr}y_j + x_c y_{j+1} - 2x_{cl}y_{j-1} - x_{cl}y_j - x_c y_{j-1}\right)$$
$$+\frac{1}{6n\Delta x}\left(2x_c y_{jr} + x_c y_{jr+1} + x_{cr}y_{jr} + 2x_{cr}y_{jr+1}\right.$$
$$\left.-2x_{cl}y_{jl-1} - x_c y_{jl-1} - x_{cl}y_{jl} - 2x_c y_{jl}\right)$$

$$\sum_{k=0}^{31}\sum_{l=0}^{31} B_{jkl}^{yxx} x_k x_l = -\frac{1}{6n\Delta x}\left(x_{cr}^2 + x_c x_{cr} - x_{cl}^2 - x_c x_{cl}\right)$$
$$+\frac{1}{6n\Delta x}\left(x_{c+1}^2 + x_c x_{c+1} - x_{c-1}^2 - x_c x_{c-1}\right).$$

The following terms result from the discretisation of the diffusion term:

$$\sum_{k=0}^{31} L_{ik}^{xx} x_k = \frac{\nu}{n\Delta x^2}\left(x_{i+1} - 2x_i + x_{i-1},\right),$$

$$\sum_{k=0}^{511} L_{ik}^{xy} y_k = \frac{\nu}{n\Delta x^2}\left(y_{jr+1} - y_{jr} - y_{jl} + y_{jl-1}\right),$$

$$\sum_{k=0}^{31} L_{jk}^{yx} x_k = \frac{\nu}{\Delta x^2}\left(x_{cr} + x_{cl} - 2x_c\right) - \frac{\nu}{n\Delta x^2}\left(x_{c+1} - 2x_c + x_{c-1}\right),$$

$$\sum_{k=0}^{511} L_{jk}^{yy} y_k = \frac{\nu}{\Delta x^2}\left(y_{j+1} - 2y_j + y_{j-1}\right) - \frac{\nu}{n\Delta x^2}\left(y_{jr+1} - y_{jr} + y_{jl} - y_{jl-1}\right).$$

References

1. Bengtsson, L., Steinheimer, M., Bechtold, P., Geleyn, J.F.: A stochastic parametrization for deep convection using cellular automata. Q. J. R. Meteorol. Soc. **139**(675), 1533–1543 (2013)
2. Berner, J., Shutts, G.J., Leutbecher, M., Palmer, T.N.: A Spectral stochastic kinetic Energy backscatter scheme and its impact On flow-dependent predictability in the ECMWF ensemble prediction system. J. Atmos. Sci. **66**, 603 (2009). doi:10.1175/2008JAS2677.1
3. Buizza, R., Miller, M., Palmer, T.N.: Stochastic representation of model uncertainties in the ECMWF ensemble prediction system. Q. J. R. Meteorol. Soc. **125**, 2887–2908 (1999). doi:10.1002/qj.49712556006
4. Crommelin, D., Vanden-Eijnden, E.: Subgrid-Scale parameterization with conditional Markov chains. J. Atmos. Sci. **65**, 2661–2675 (2008). doi:10.1175/2008JAS2566.1
5. Dolaptchiev, S., Achatz, U., Timofeyev, I.: Stochastic closure for local averages in the finite-difference discretization of the forced burgers equation. Theor. Comput. Fluid Dyn. **27**(3–4), 297–317 (2013)
6. Dolaptchiev, S., Timofeyev, I., Achatz, U.: Subgrid-scale closure for the inviscid burgers-hopf equation. Commun. Math. Sci. **11**(3), 757–777 (2013)
7. Düben, P.D., Joven, J., Lingamneni, A., McNamara, H., De Micheli, G., Palem, K.V., Palmer, T.N.: On the use of inexact, pruned hardware in atmospheric modelling. Philos. Trans. R. Soc. A (2014)
8. Düben, P.D., McNamara, H., Palmer, T.: The use of imprecise processing to improve accuracy in weather & climate prediction. J. Comput. Phys. **271**(0):2–18 (2014). Frontiers in Computational Physics Modeling the Earth System
9. Düben, P.D., Palmer, T.N.: Benchmark tests for numerical forecasts on inexact hardware. Mon. Weather Rev. **142**, 3809–3829 (2014). doi:10.1175/MWR-D-14-00110.1

10. Düben, P.D., Schlachter, J., Parishkrati, Yenugula, S., Augustine, J., Enz, C., Palem, K., Palmer, T.N.: Opportunities for energy efficient computing: a study of inexact general purpose processors for high-performance and big-data applications . In: Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition (DATE '15). EDA Consortium, San Jose, CA, USA, pp. 764–769 (2015)
11. Franzke, C., Majda, A.J., Vanden-Eijnden, E.: Low-order stochastic mode reduction for a realistic barotropic model climate. J. Atmos. Sci. **62**, 1722–1745 (2005)
12. Gan, L., Fu, H., Luk, W., Yang, C., Xue, W., Huang, X., Zhang, Y., Yang, G.: Accelerating solvers for global atmospheric equations through mixed-precision data flow engine. In: 23rd International Conference on Field Programmable Logic and Applications (FPL), pp. 1–6 (2013)
13. Hasselmann, K.: Stochastic climate models part i. theory. Tellus **28**(6), 473–485 (1976)
14. Horenko, I., Dolaptchiev, S.I., Eliseev, A.V., Mokhov, I.I., Klein, R.: Metastable decomposition of high-dimensional meteorological data with gaps. J. Atmos. Sci. **65**, 3479–3496 (2008)
15. IEEE: IEEE standard for binary floating-point arithmetic. ANSI/IEEE Std, pp. 754–1985 (1985). doi:10.1109/IEEESTD.1985.82928
16. Kahng, A., Kang, S., Kumar, R., Sartori, J.: Slack redistribution for graceful degradation under voltage overscaling. In: 15th Asia and South Pacific on Design Automation Conference (ASP-DAC), pp. 825–831 (2010)
17. Lingamneni, A., Enz, C., Nagel, J.L., Palem, K., Piguet, C.: Energy parsimonious circuit design through probabilistic pruning. In: Design, Automation Test in Europe Conference Exhibition (DATE), pp. 1–6 (2011)
18. Majda, A., Timofeyev, I., Vanden-Eijnden, E.: Stochastic models for selected slow variables in large deterministic systems. Nonlinearity **19**, 769–794 (2006)
19. Majda, A.J., Timofeyev, I., Vanden Eijnden, E.: A mathematical framework for stochastic climate models. Commun. Pure Appl. Mathe. **54**(8), 891–974 (2001)
20. Narayanan, S., Sartori, J., Kumar, R., Jones, D.L.: Scalable stochastic processors. In: Proceedings of Design, Automation and Test in Europe Conference, pp. 335–338 (2010)
21. Oriato, D., Tilbury, S., Marrocu, M., Pusceddu, G.: Acceleration of a meteorological limited area model with dataflow engines. In: 2012 Symposium on Application Accelerators in High Performance Computing (SAAHPC), pp. 129–132 (2012)
22. Palem, K.: Energy aware computing through probabilistic switching: a study of limits. IEEE Trans. Comput. **54**(9), 1123–1137 (2005)
23. Palem, K.V.: Energy aware algorithm design via probabilistic computing: from algorithms and models to Moore's law and novel (semiconductor) devices. In: Proceedings of CASES, pp. 113–116 (2003)
24. Palem, K.V.: Inexactness and a future of computing. Philos. Trans. R. Soc. A math. Phys. Eng. Sci. **372**(2018), 20130281 (2014)
25. Palmer, T.N.: A nonlinear dynamical perspective on model error: a proposal for non-local stochastic-dynamic parametrization in weather and climate prediction models. Q. J. R. Meteorol. Soc. **127**(572), 279–304 (2001)
26. Palmer, T.N.: Towards the probabilistic earth-system simulator: a vision for the future of climate and weather prediction. Q. J. R. Meteorol. Soc. **138**(665), 841–861 (2012)
27. Palmer, T.N., Buizza, R., Doblas-Reyes, F., Jung, T., Leutbecher, M., Shutts, G.J., Steinheimer, M., Weisheimer, M.: Stochastic parametrization and model uncertainty. Technical Memorandum, 598 ECMWF: Reading, UK (2009)
28. Plant, R.S., Craig, G.C.: A stochastic parameterization for deep convection based on equilibrium statistics. J. Atmos. Sci. **65**, 87–105 (2008)
29. Russell, F.P., Düben, P.D., Niu, X., Luk, W., Palmer, T.N.: Hardware acceleration and precision analysis of a chaotic system with application to atmospheric modelling, accepted. In: Proceedings of FCCM (2015)
30. Shutts, G.: A kinetic energy backscatter algorithm for use in ensemble prediction systems. Q. J. R. Meteorol. Soc. **131**, 3079–3102 (2005)