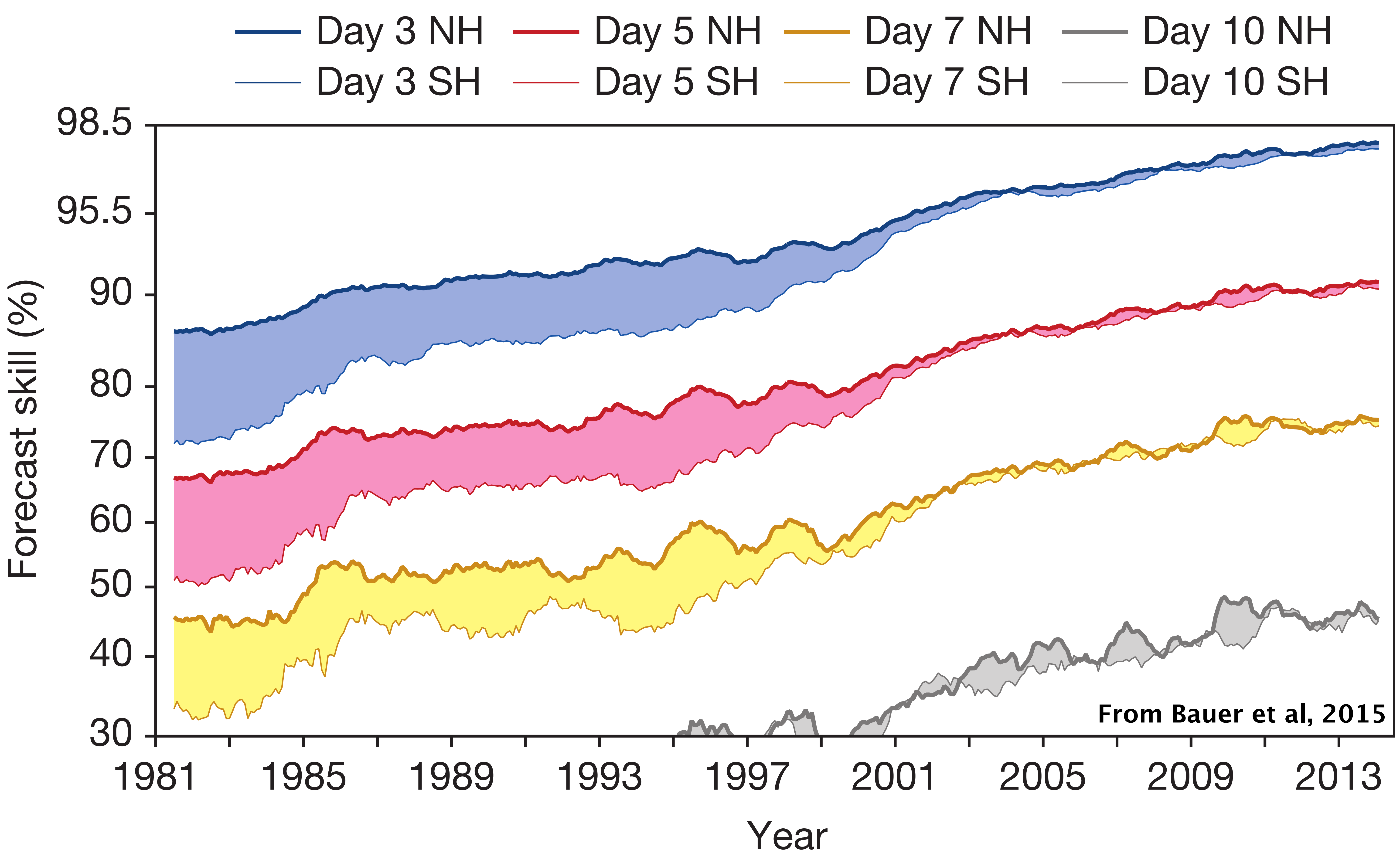
Towards 16bit weather and climate models: Posits as an alternative to floats

Milan Klöwer[1], Peter Düben[2], Tim Palmer[1]

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

Weather forecasting: Where are we?

Posits for weather and climate models
CoNGA'19
Milan Klöwer



Forecast skill at the European Weather Centre (UK)

Forecast skill mainly improved through

- **Satellite data**
- **Better models**
- **Faster supercomputers**

Strongly linked to horizontal resolution:

- **9km (2019)**
- **25km (2009)**

Goal for the next decades

- **~1km (cloud resolving)**

Requires exascale supercomputers

Origin of weather/climate forecast errors

1. Initial and boundary condition errors (observations, data assimilation, climate change)
2. Model error (approximations in equations of motion; clouds, precipitation, radiation, …)
3. Discretisation error (finite spatial & temporal resolution)

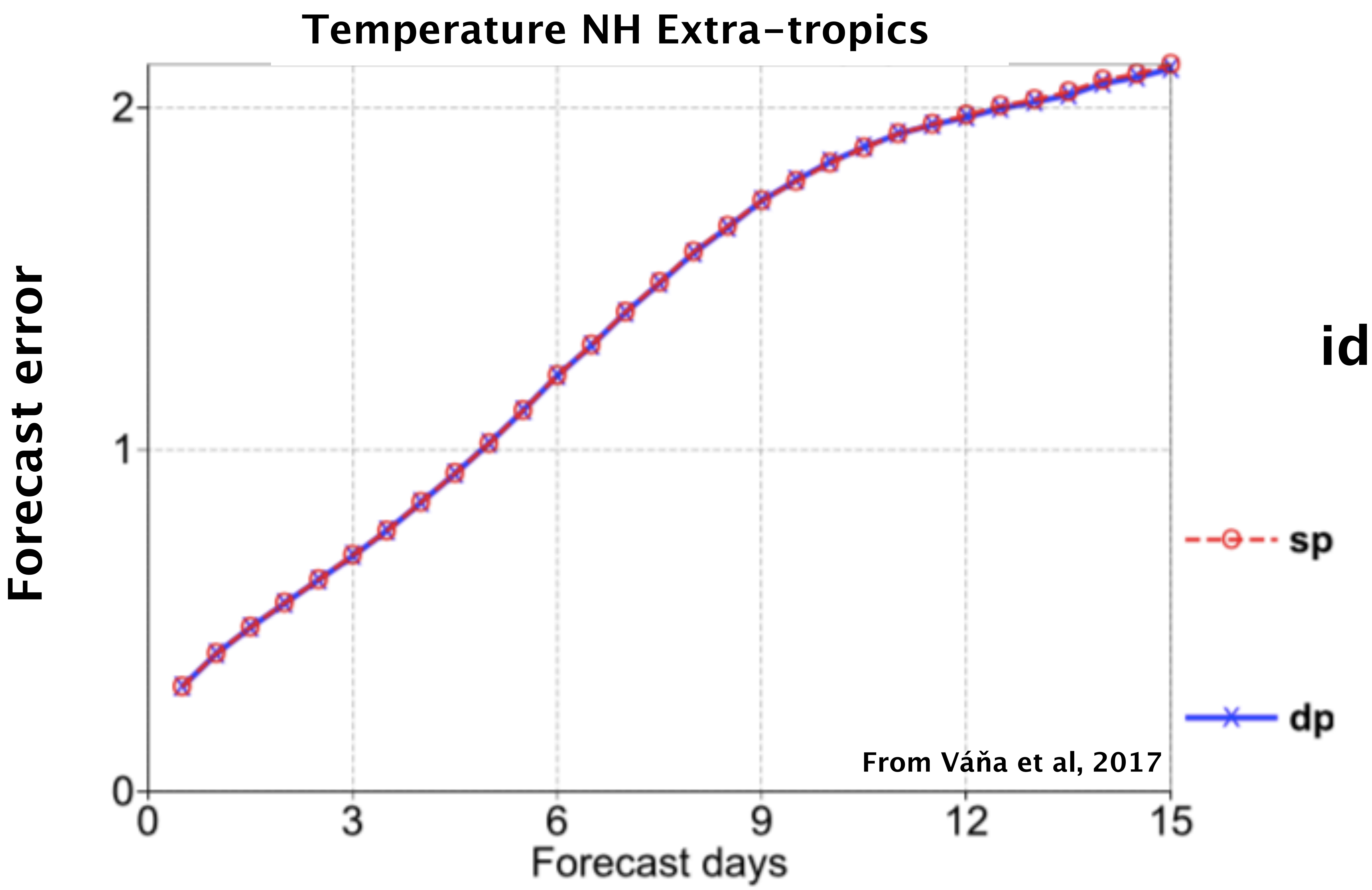
HUGE

4. Rounding error of double precision floats

tiny.

Allow for bigger rounding errors they will be dwarfed by other errors anyway!

Weather forecasting in single precision



Single precision yields almost identical results in ~60% of runtime.

Next step: 16bit?

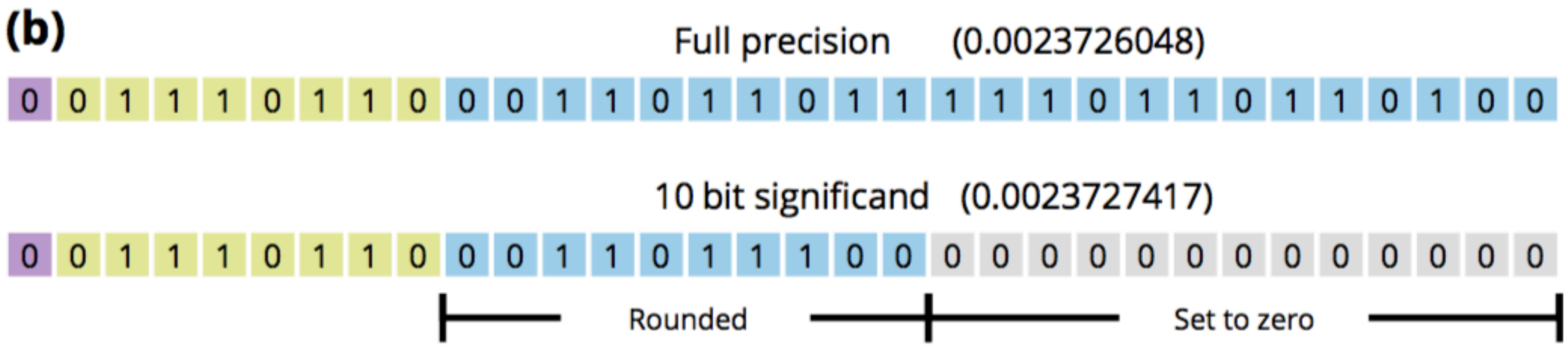
…

Half precision floats
bfloat16

…

Reduced precision software emulator

$$c = \mathscr{R}_{10}(a + b), \qquad a, b, c \quad :: \textbf{DOUBLE}$$



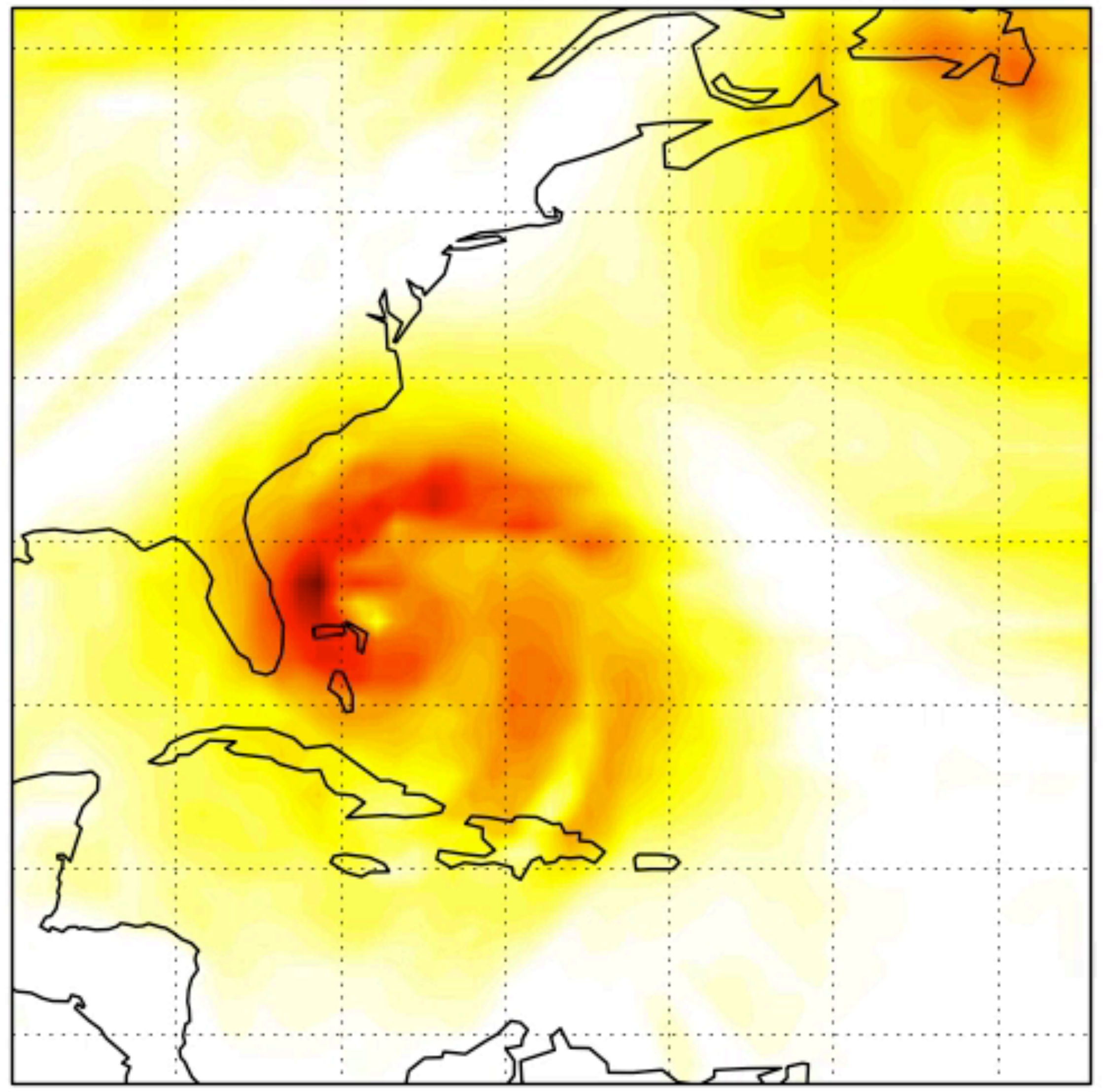
From Dawson & Düben, 2017

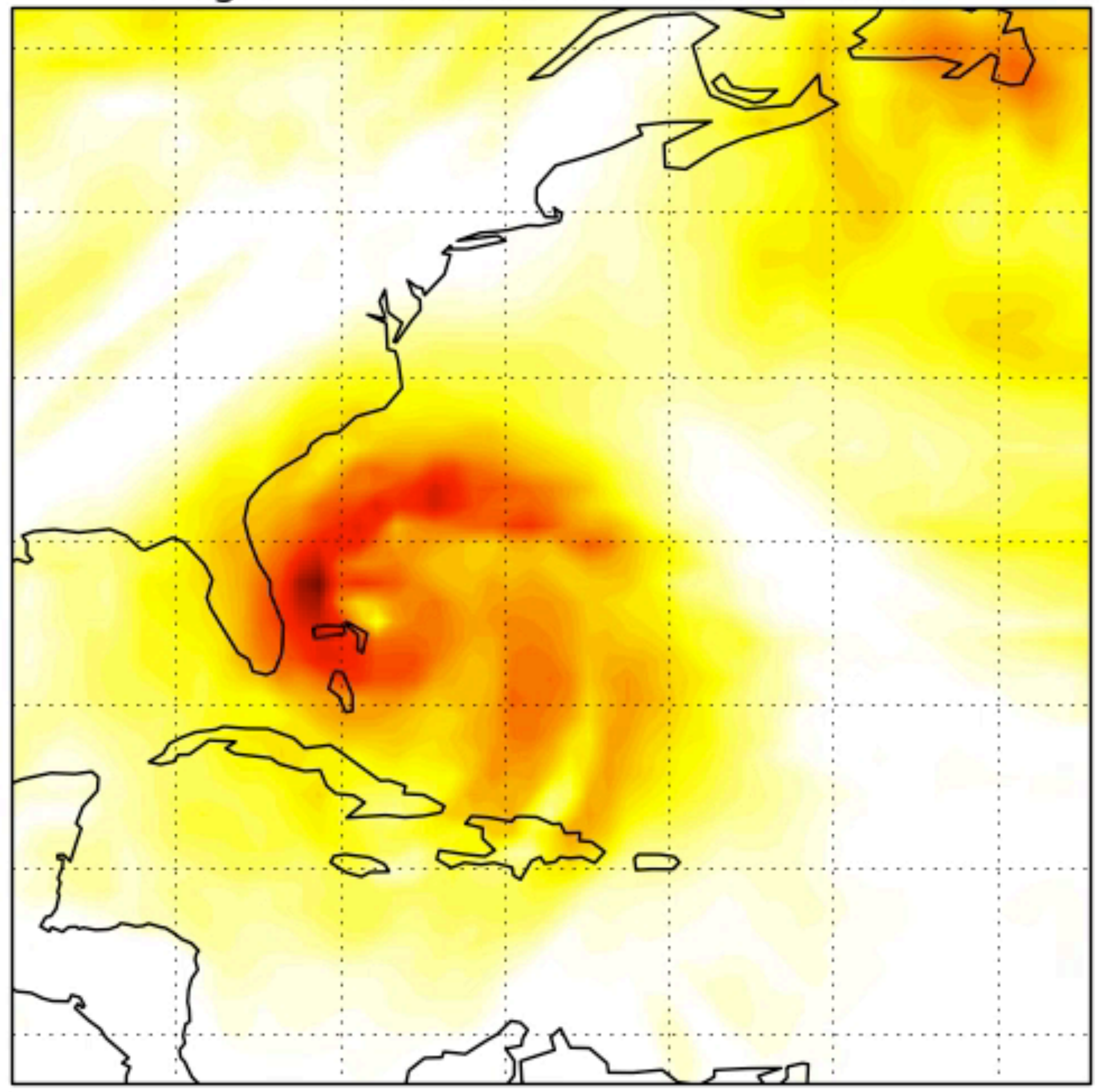
Allows testing of weather and climate models without specialised hardware.

Software emulation: Reduced precision

Posits for weather and climate models
CoNGA'19
Milan Klöwer

Double precision 52sbits

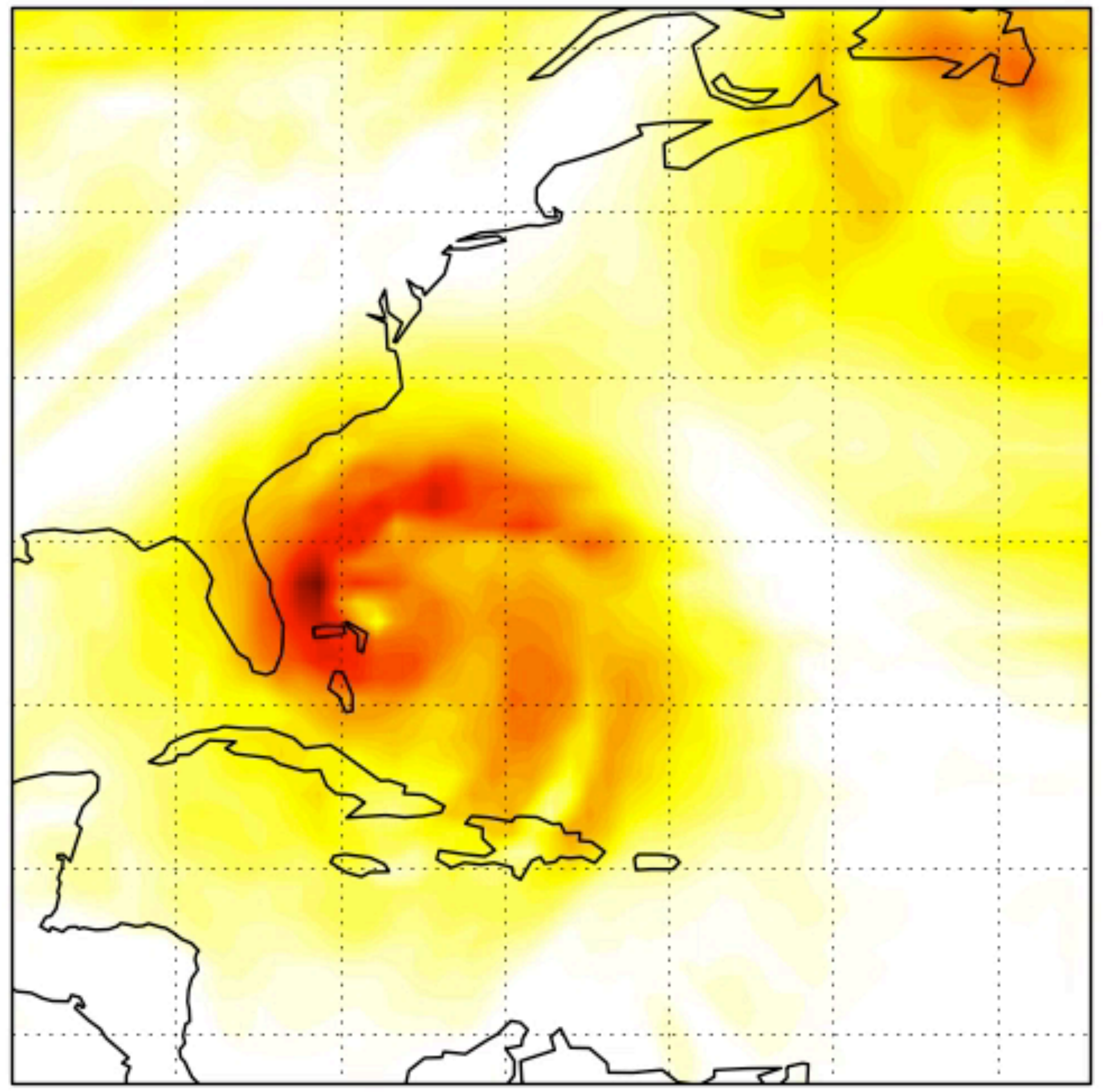




Single precision 23sbits

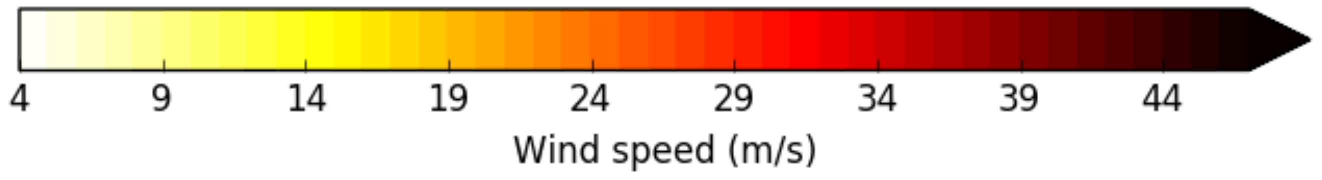
From Matthew Chantry

8 significant bits



Hurricane Sandy

27/10/12 00:00
850hPa wind speed
T255L91 ~ 80km



Grid point calculations

Spectral transform

Spectral inverse

Linear terms
in reduced precision

8sbits not a representative potential for the whole model!

The complexity of a weather/climate model

Posits for weather and climate models
CoNGA'19
Milan Klöwer

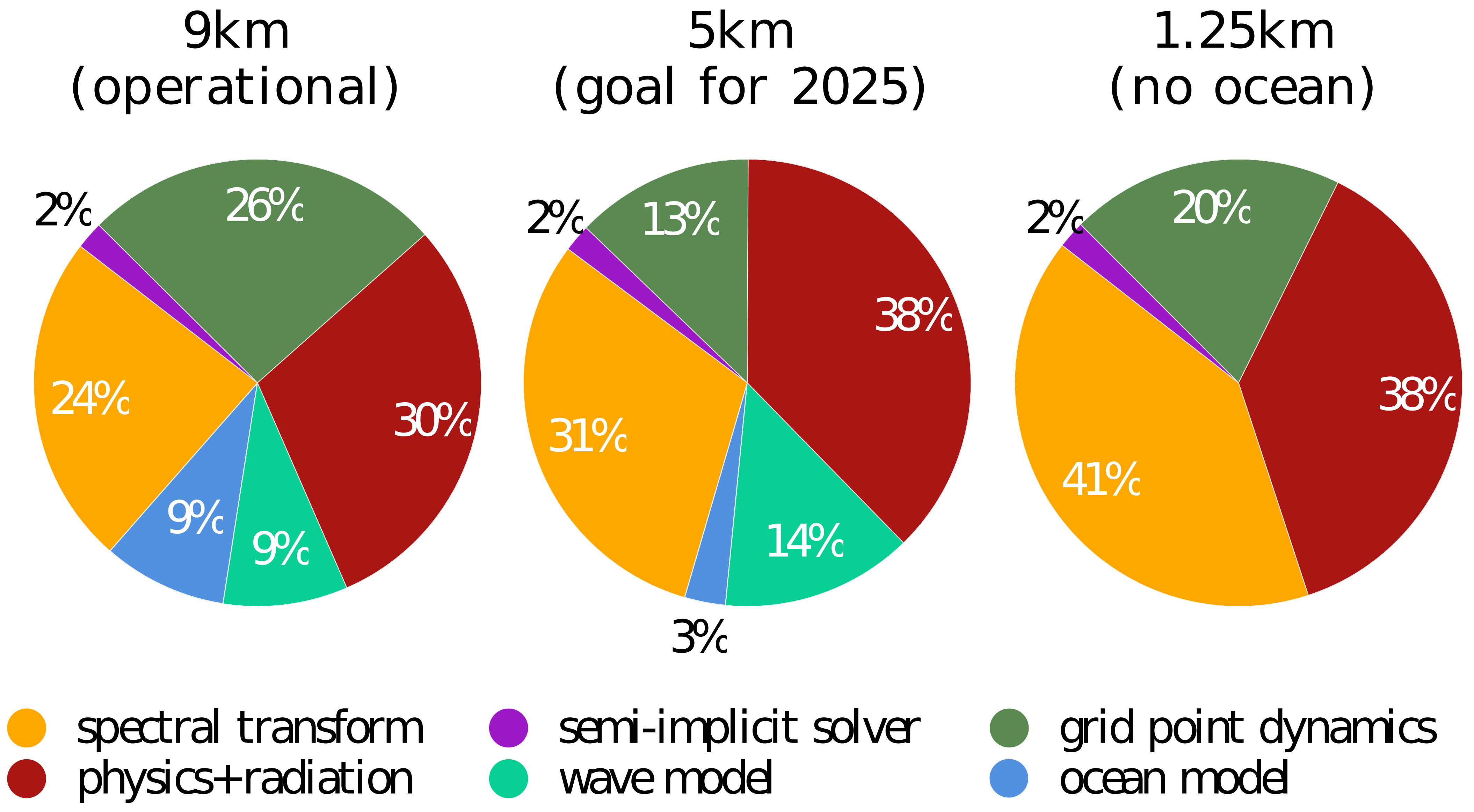
Cost (% of CPU time)

MPI comm
Cubic interpolation

Elliptic solver,
Multi-grid precon

FFT
Legendre (Mat x Mat)
MPI comm

Finite difference,
Semi-implicit solver,
Interpolations,
MPI comm

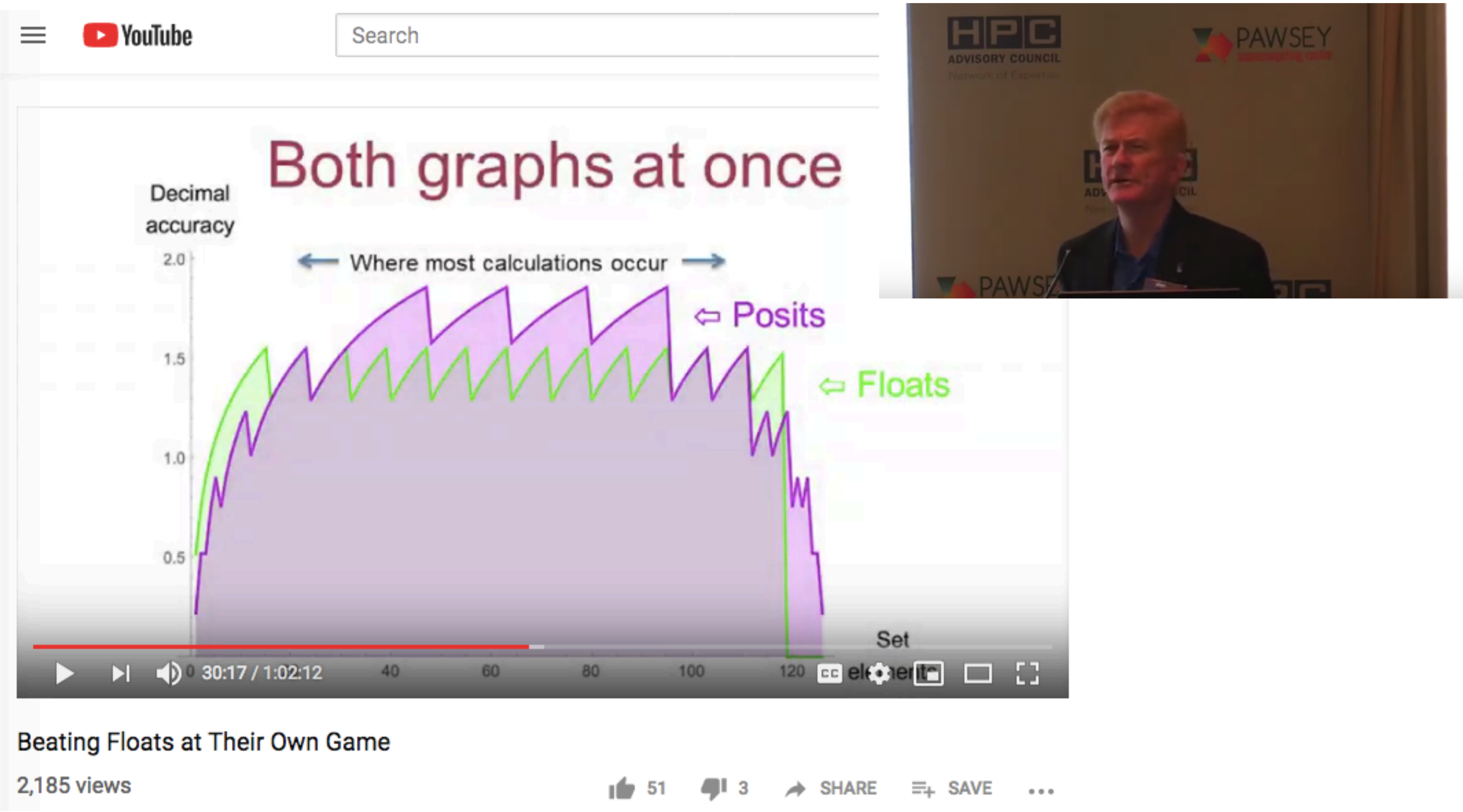


Communication volume

IFS (spectral), global MPI comm	427 TB	2880 MPI procs	12min runtime
DG (grid point), local MPI comm	34 TB	2880 MPI procs	4h runtime

A posit of posits?

Posits for weather and climate models
CoNGA'19
Milan Klöwer



Published Aug 4, 2017

Posits:
A number format much better suited for weather and climate models?

Julia-based posit emulator

written by Isaac Yonemoto

```
julia> # define posit environment
julia> using SigmoidNumbers
julia> Posit161 = Posit{16,1};
julia> # convert float to 16bit posit, add
julia> a = Posit161(12.3);
julia> c = a+a;
julia> # bits split in sign, regime, exponent and fraction
julia> bits(c," ")
"0 1110 0 1000100110"
julia> Float64(c)    # convert back to double
24.59375
```

Figure 2.3: Example use of the posit emulator *SigmoidNumbers* in the Julia shell.

The simplest chaotic system: Lorenz 1963

Posits for weather and climate models
CoNGA'19
Milan Klöwer

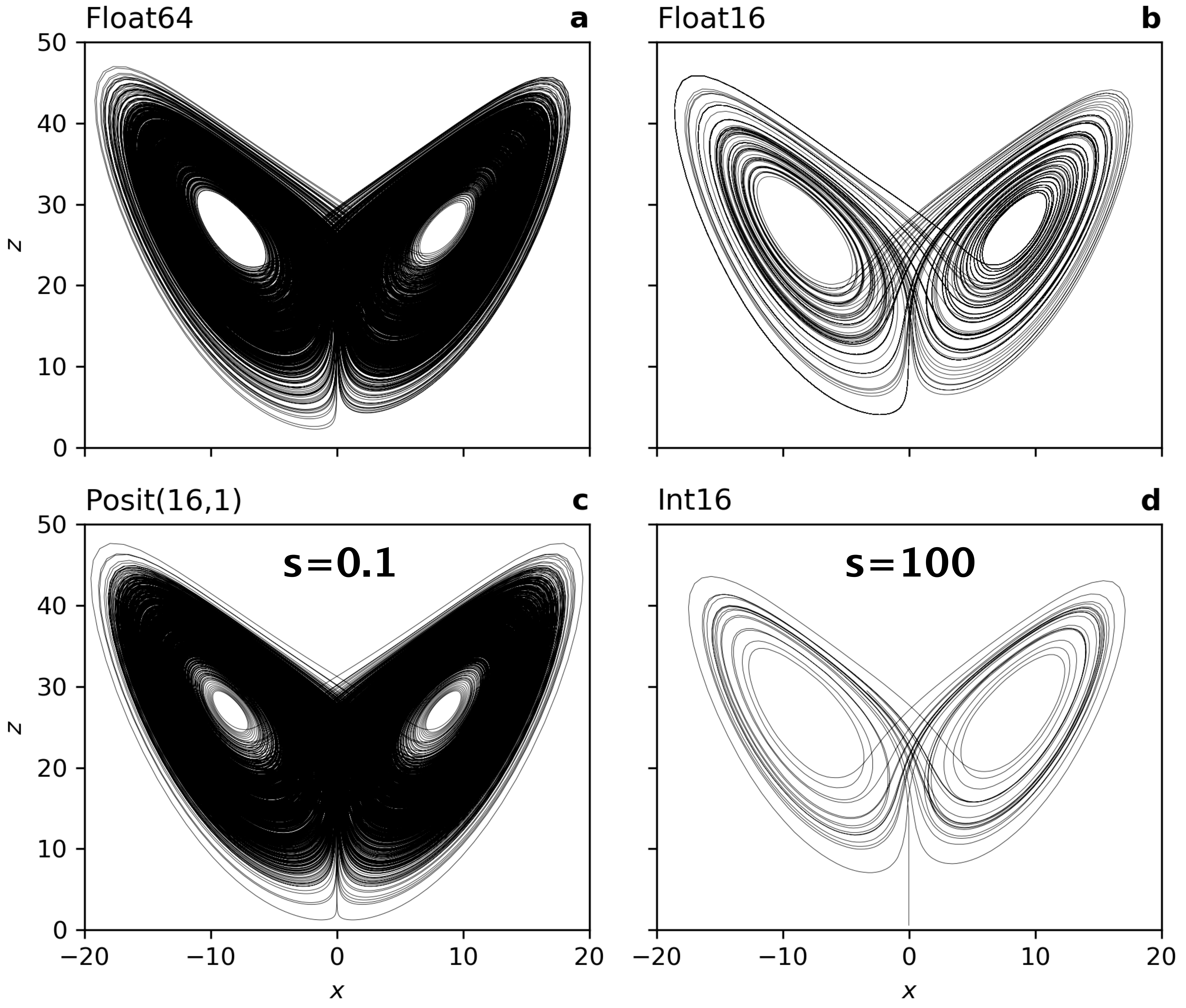
$$\dot{x} = \sigma(y - x)$$

$$\dot{y} = x(\rho - z) - y$$

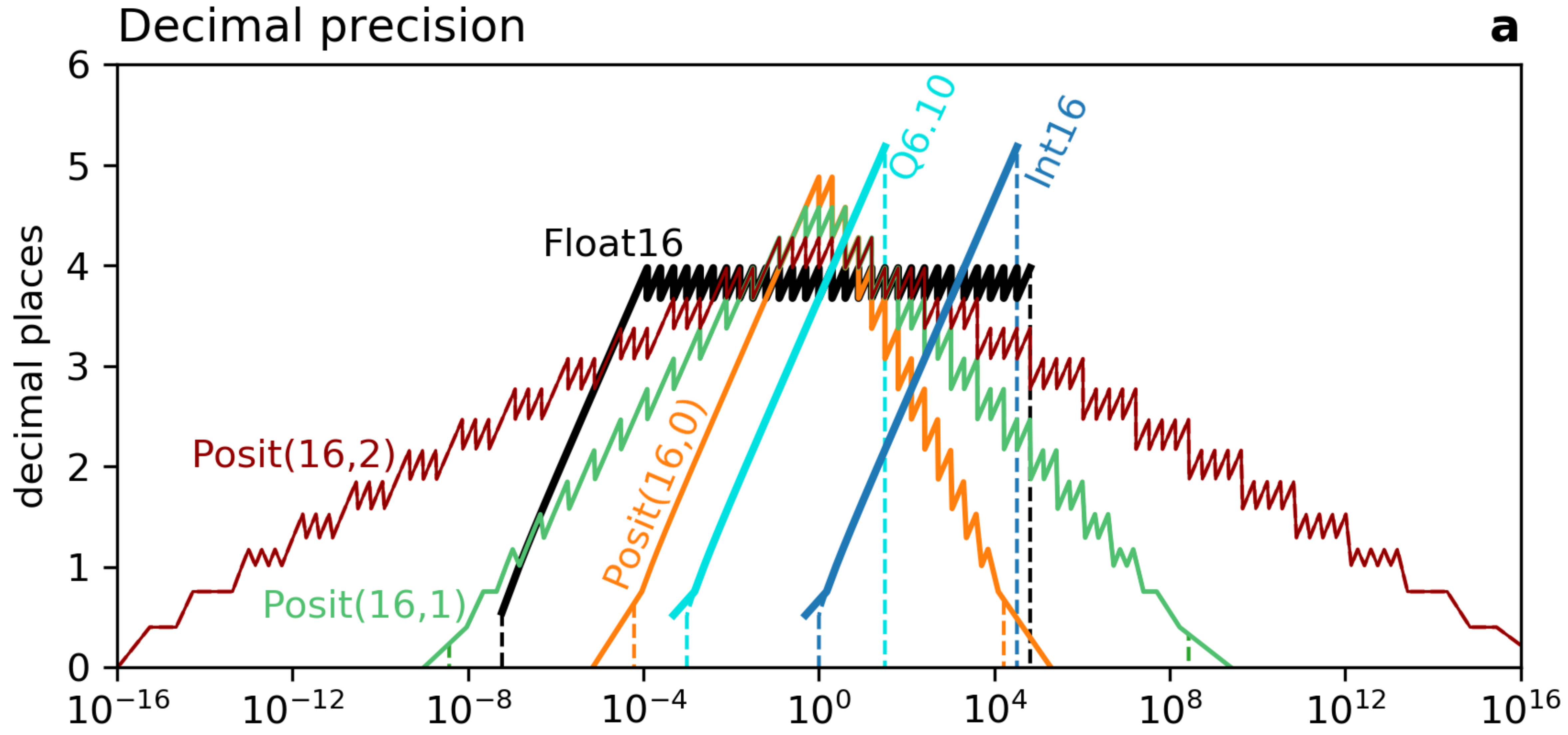
$$\dot{z} = xy - \beta z$$

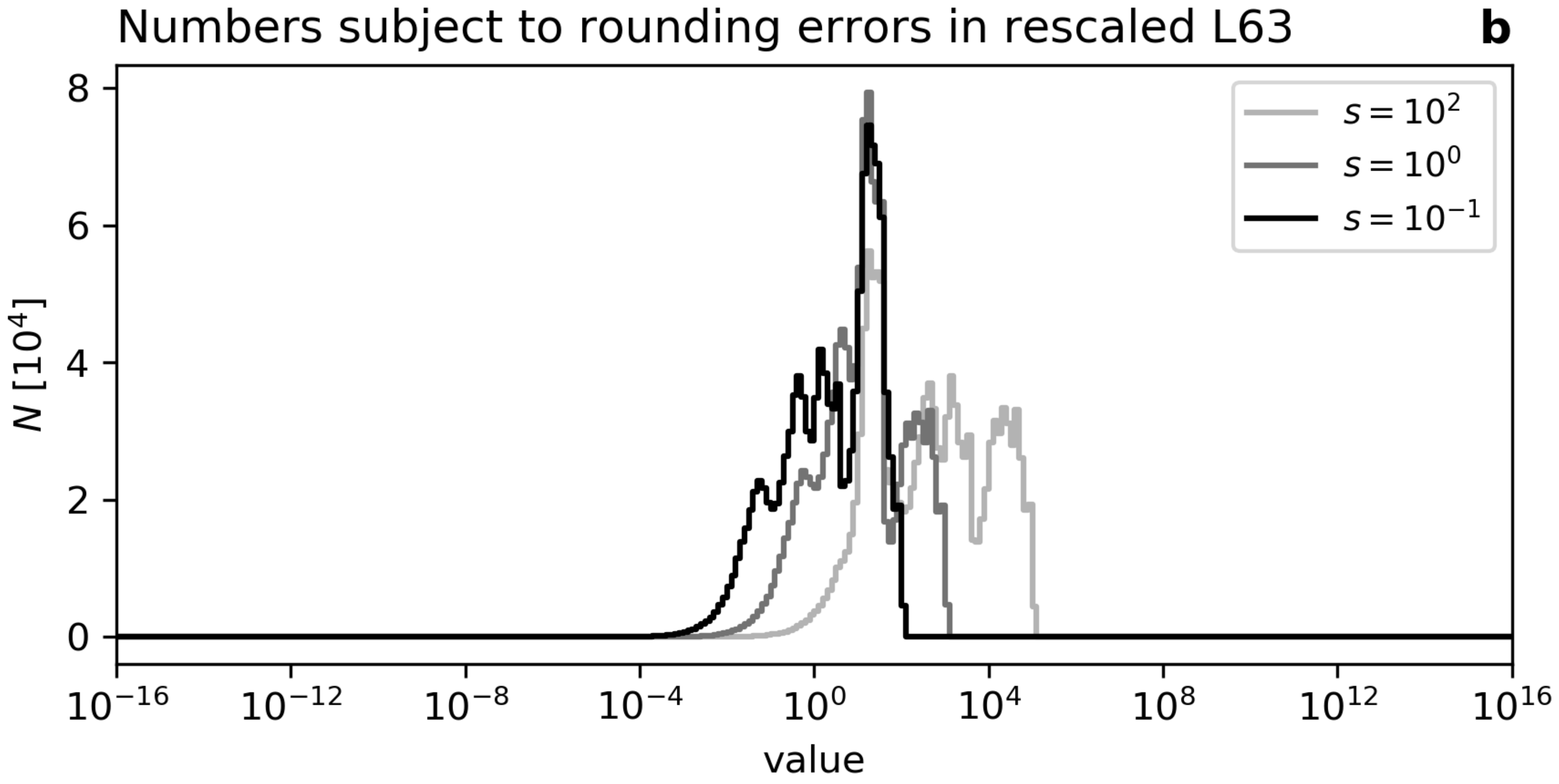
Rescale the system:

$$\hat{x} = sx$$



Decimal precision





Posits for weather and climate models
CoNGA'19
Milan Klöwer

$$\textbf{decimal precision} = -\log_{10}\left|\log_{10}\left(\frac{x_{repr}}{x_{exact}}\right)\right|$$

What would be the optimal decimal precision distribution?

Reduced precision modelling:

- **Rescale equations to fit number system (limited)**
- **Find number system that fits computed numbers**

Shallow water equations I

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} + f\hat{\mathbf{z}} \times \mathbf{u} = -g\nabla\eta - \nu\nabla^4\mathbf{u} - r\mathbf{u} + \mathbf{F}$$

Conservation of momentum

$$\frac{\partial \eta}{\partial t} + \nabla \cdot (\mathbf{u}h) = 0.$$

Conservation of mass

$$\frac{\partial q}{\partial t} + \mathbf{u} \cdot \nabla q = 0.$$

Conservation of tracer (temperature, humidity, salinity, …)

Some features

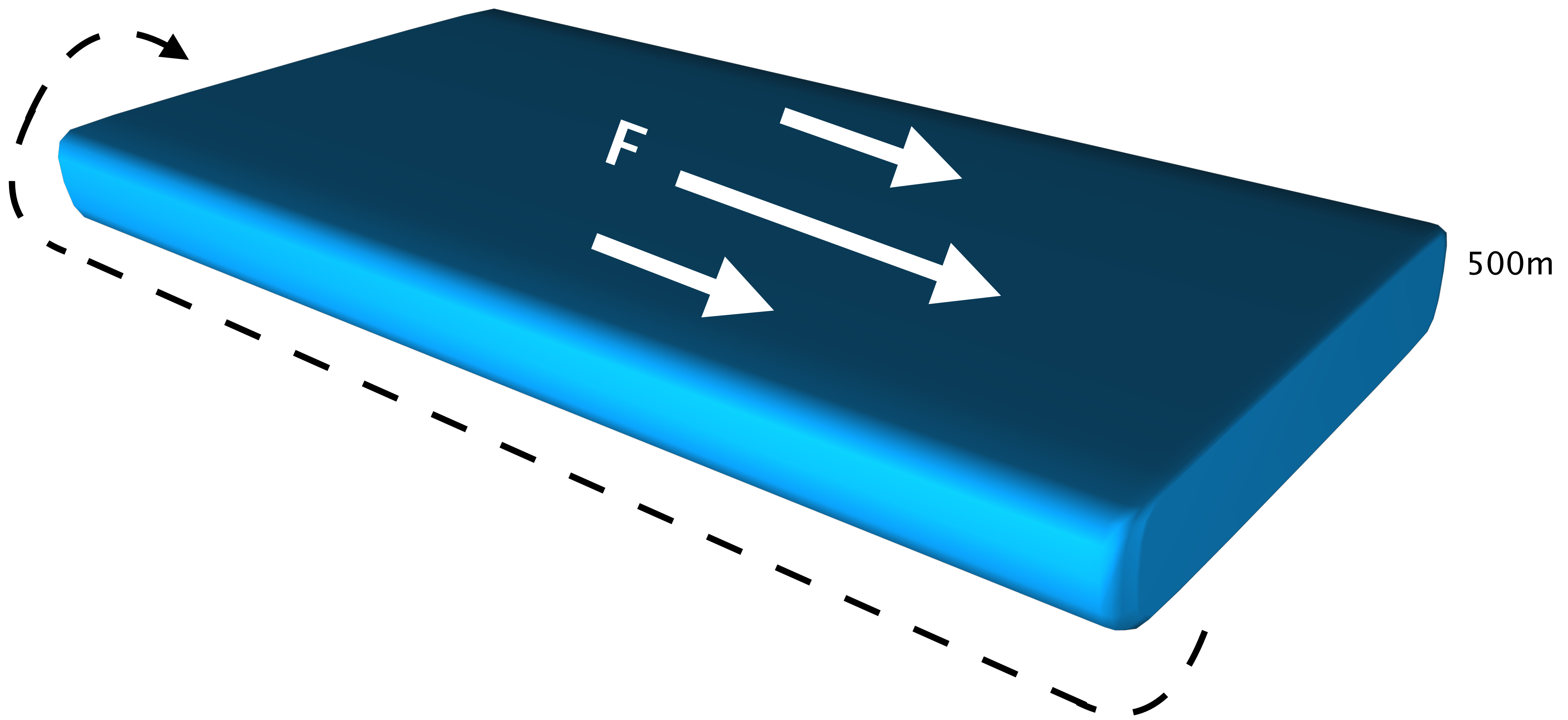
- **Energy and enstrophy conserving mom adv (Arakawa & Hsu, 1990)**
- **Smagorinsky-like biharmonic diffusion (Griffies & Hallberg, 2000)**
- **Time splitting (ideas from NEMO, ROMS)**
- **Semi-Lagrangian tracer advection (Diamantakis, 2014)**

Shallow water equations II

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} + f\hat{\mathbf{z}} \times \mathbf{u} = -g\nabla\eta - \nu\nabla^4\mathbf{u} - r\mathbf{u} + \mathbf{F}$$

$$\frac{\partial \eta}{\partial t} + \nabla \cdot (\mathbf{u}h) = 0.$$

$$\frac{\partial q}{\partial t} + \mathbf{u} \cdot \nabla q = 0.$$



Prognostic variables

- u **meridional velocity**
- v **zonal velocity**
- η **surface height, pressure**
- q **Tracer, e.g. temperature**

Basically a 2D layer of the Navier-Stokes equations.

Rescaling: Biharmonic diffusion

$$\frac{\partial u}{\partial t} = \ldots - \nu \nabla^4 \mathbf{u}$$

For 10km model: $\nu \approx 10^{10}\,\mathrm{m}^4\,\mathrm{s}^{-1}$ $\quad \nabla^4: \mathcal{O}(\frac{1}{\Delta^4}) \approx 10^{-16}\,\mathrm{m}^{-4}$

Rescaling algorithms is very important for 16bit arithmetics:

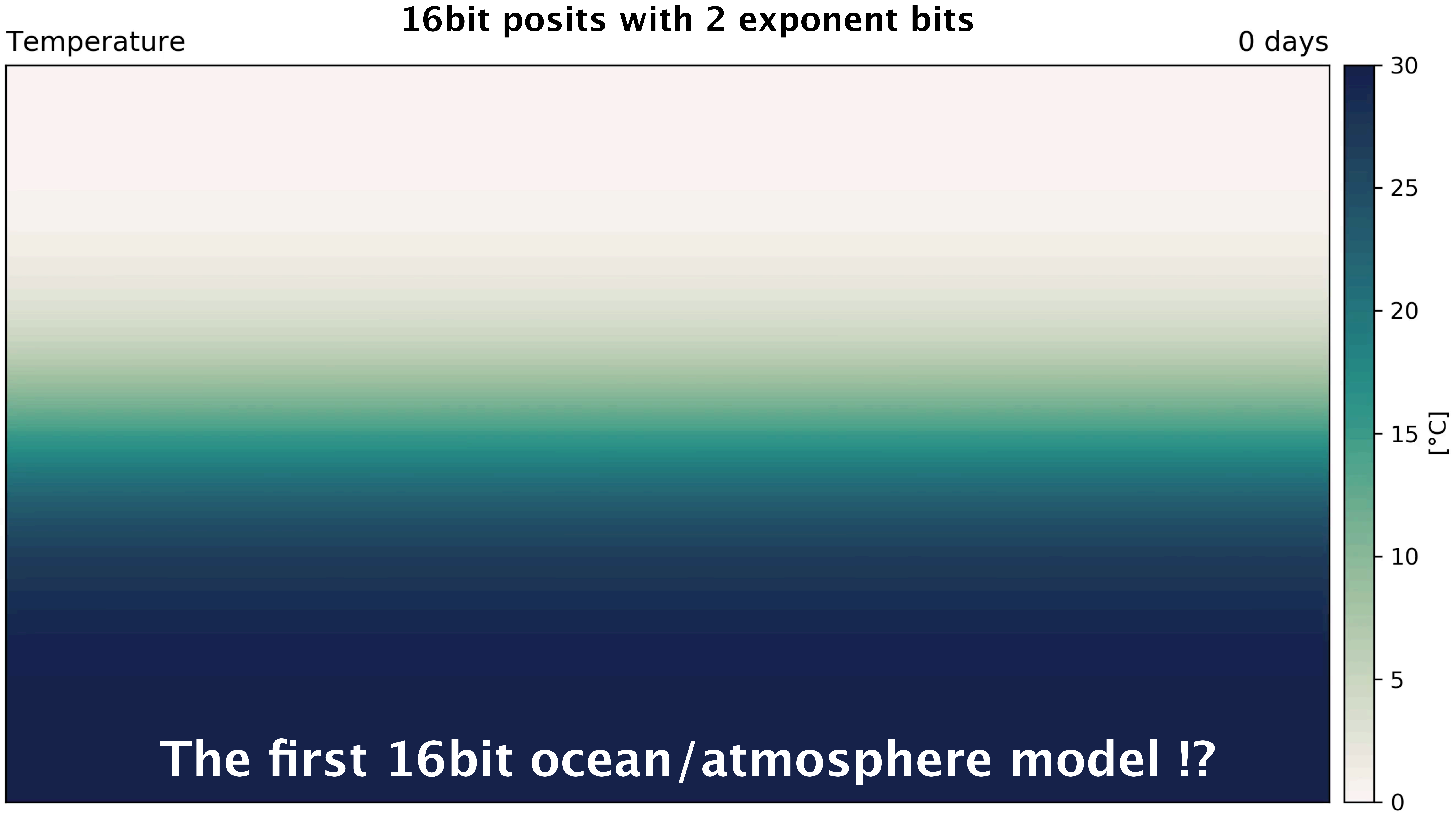
$$\mathbf{u}_{n+1} = \mathbf{u}_n + (\frac{\Delta t}{\Delta})(\ldots - \tilde{\nu}\,\tilde{\nabla}^4 \mathbf{u})$$

$\frac{\Delta t}{\Delta} \approx 10^{-2}\,\mathrm{sm}^{-1}$ $\quad \tilde{\nu} \approx 10^{-2}\,\mathrm{ms}^{-1}$ $\quad \tilde{\nabla}^4: \mathcal{O}(1)$

Which is indeed computable with 16bit arithmetics!

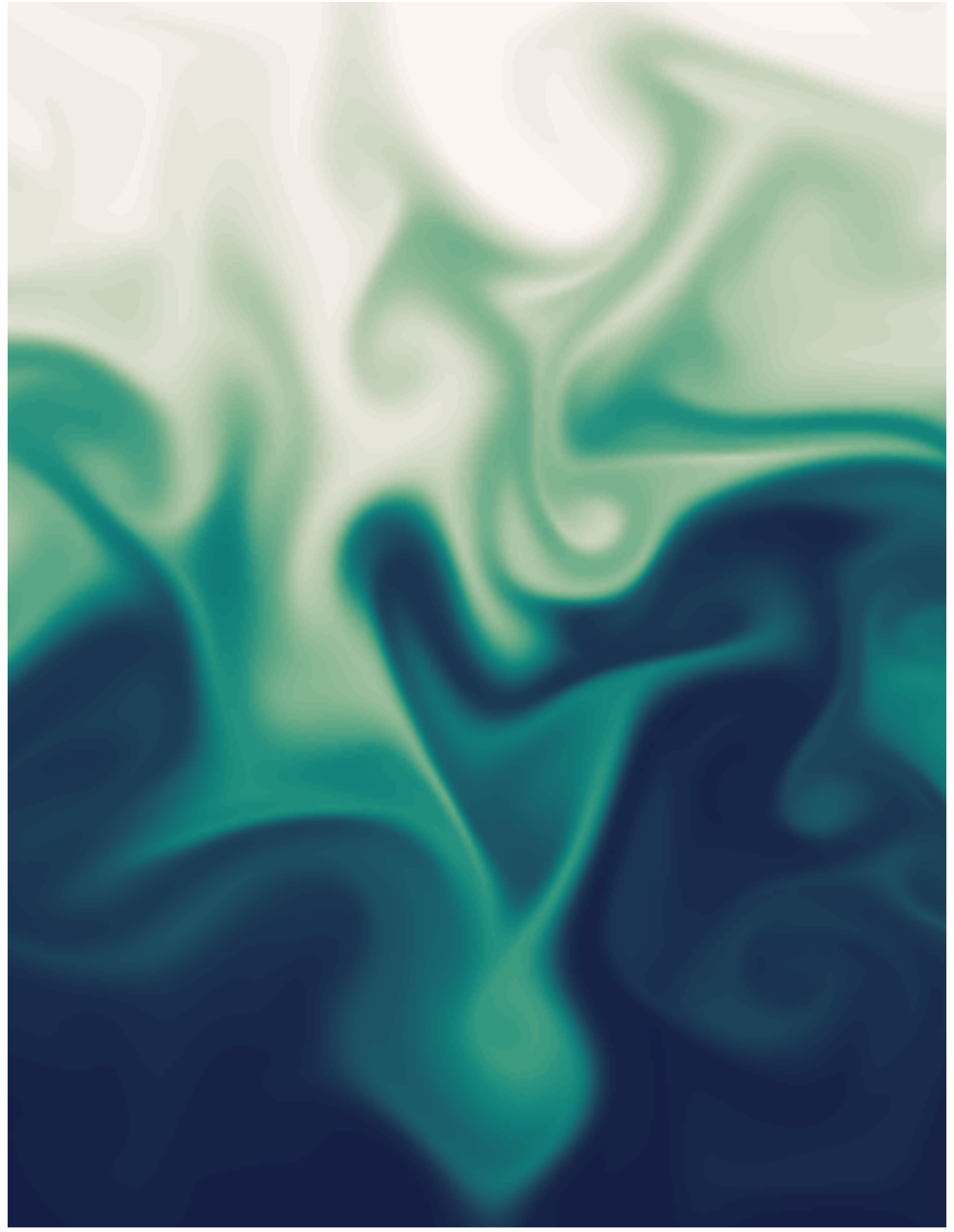
Shallow water simulation

Posits for weather and climate models
CoNGA'19
Milan Klöwer



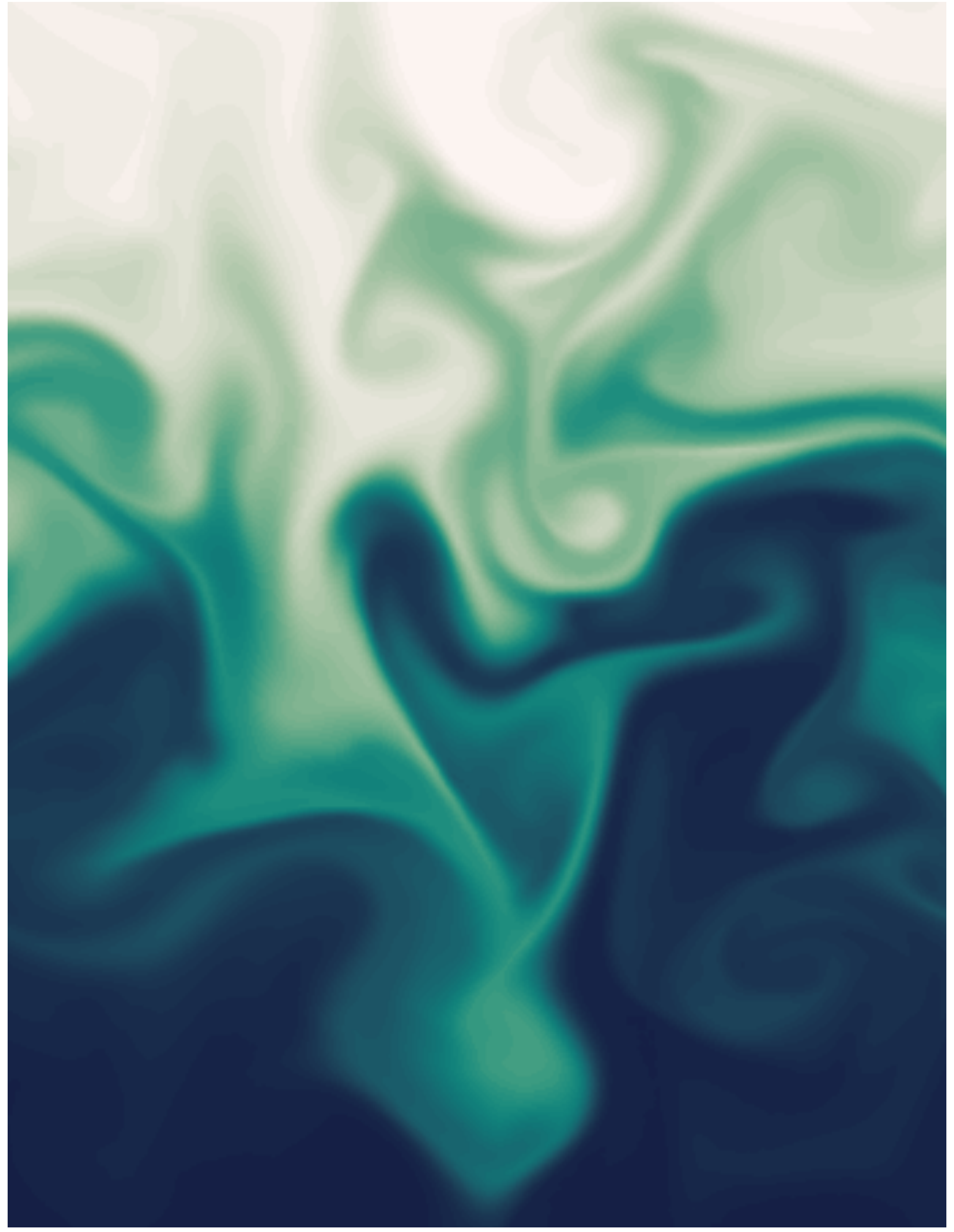
Forecasts in the shallow water model

Float64

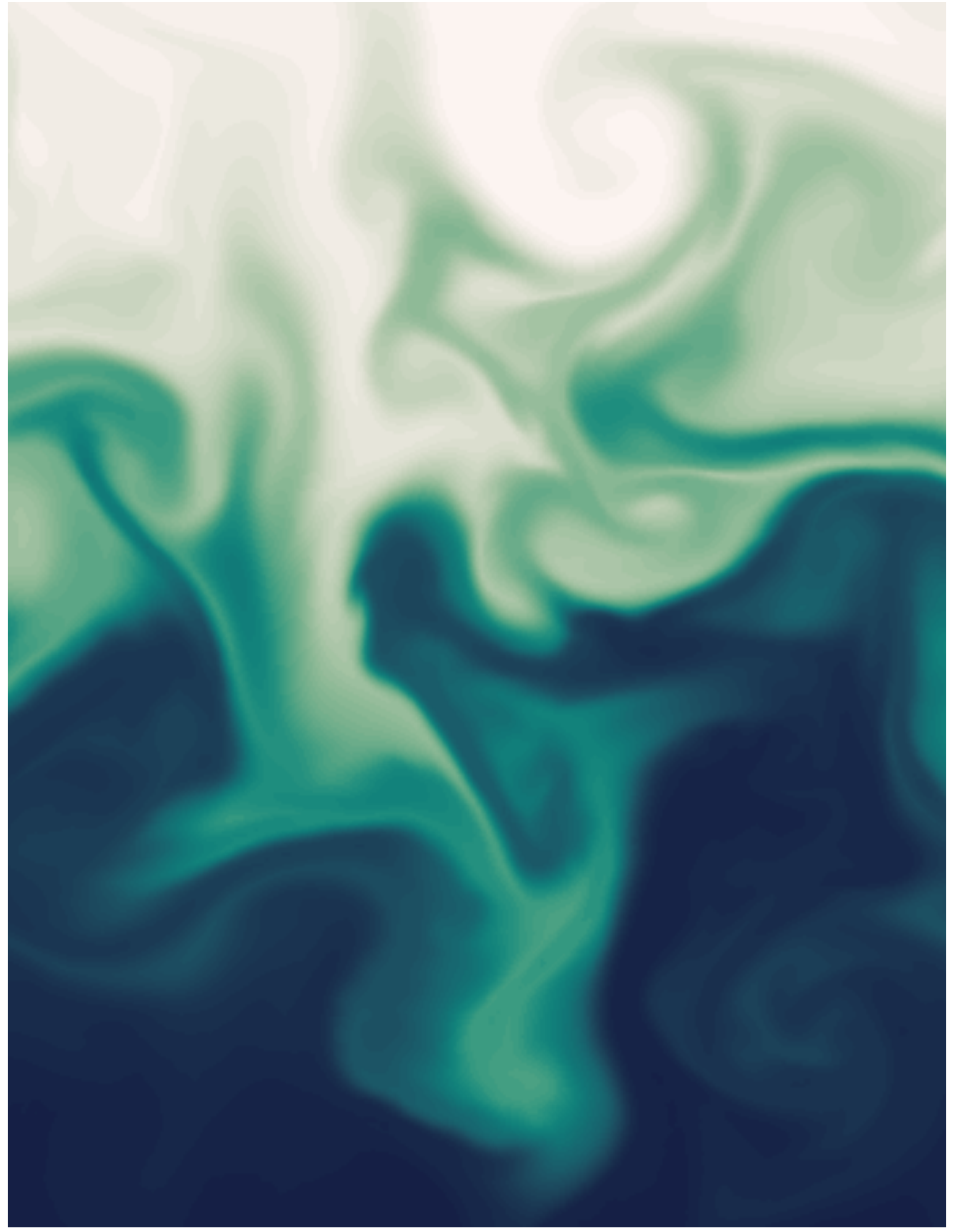


Truth

Posit(16,2)



Float16

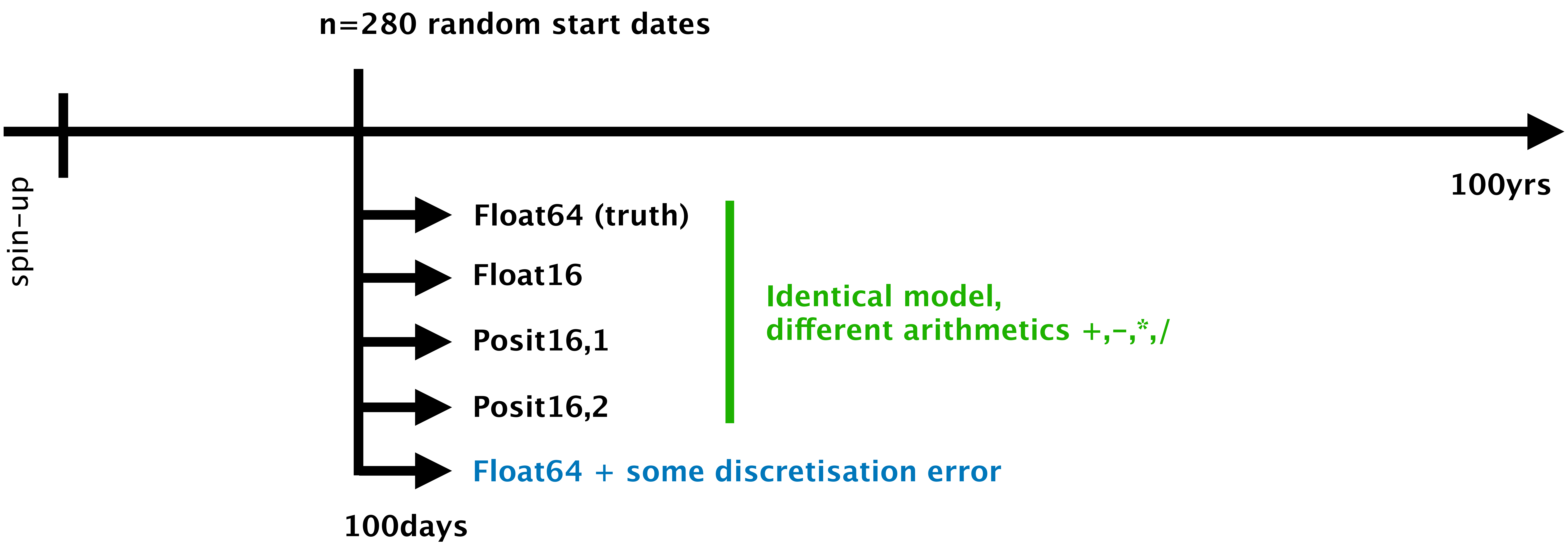


Spot the error!

Satellite

Forecasts in the shallow water model

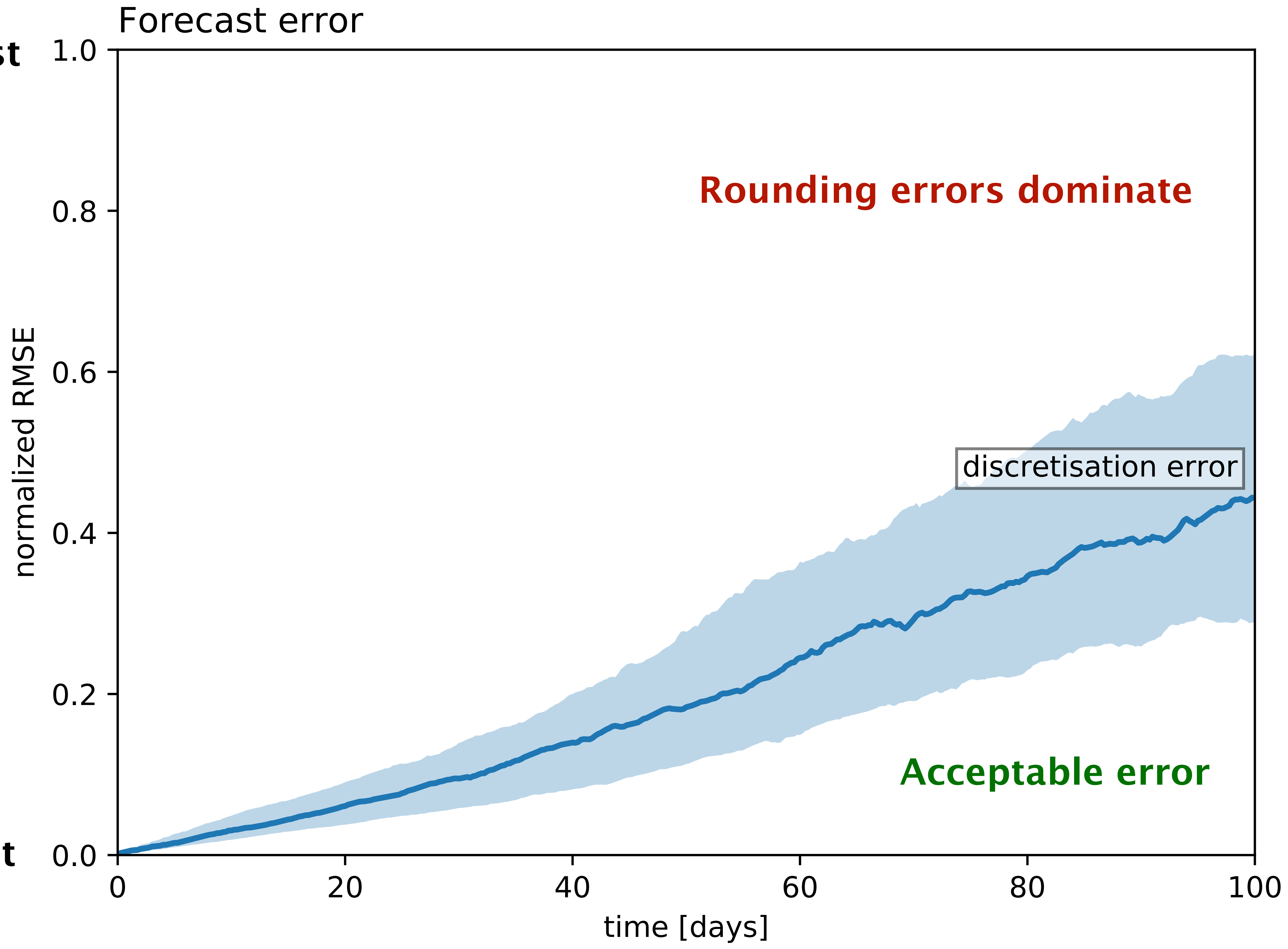
Control simulation



Forecasts in the shallow water model

Posits for weather and climate models
CoNGA'19
Milan Klöwer

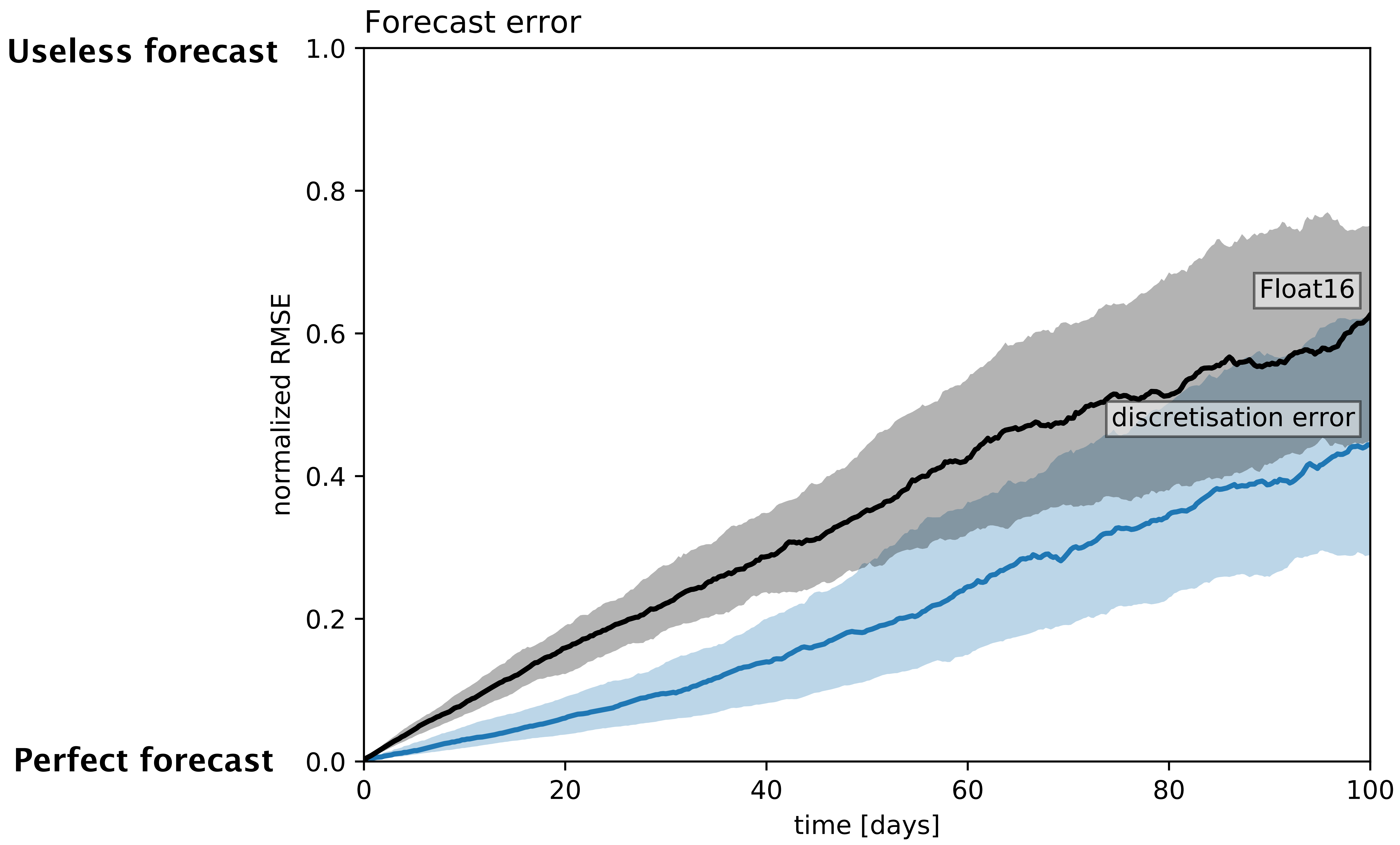
Useless forecast



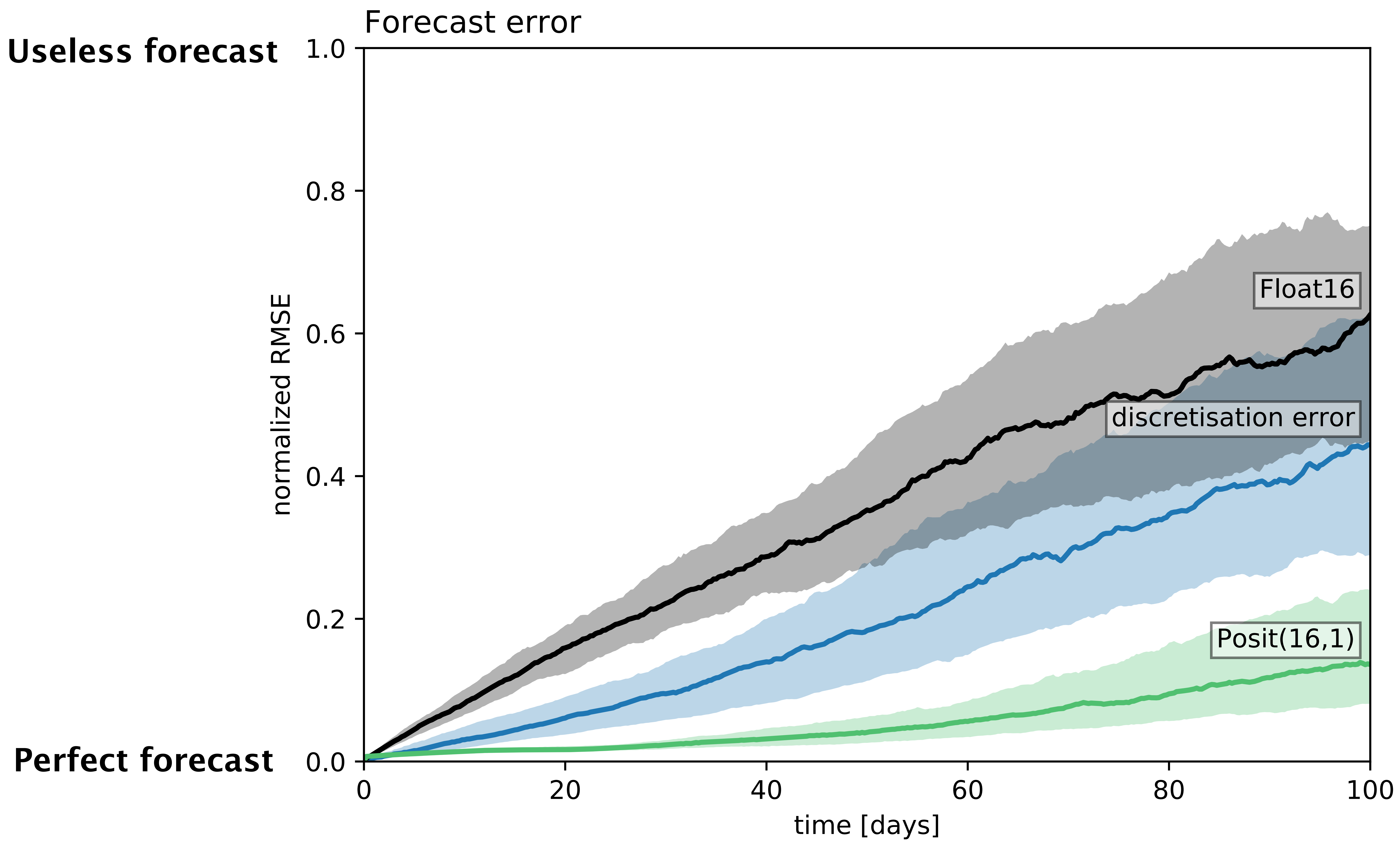
Perfect forecast

Forecasts in the shallow water model

Posits for weather and climate models
CoNGA'19
Milan Klöwer



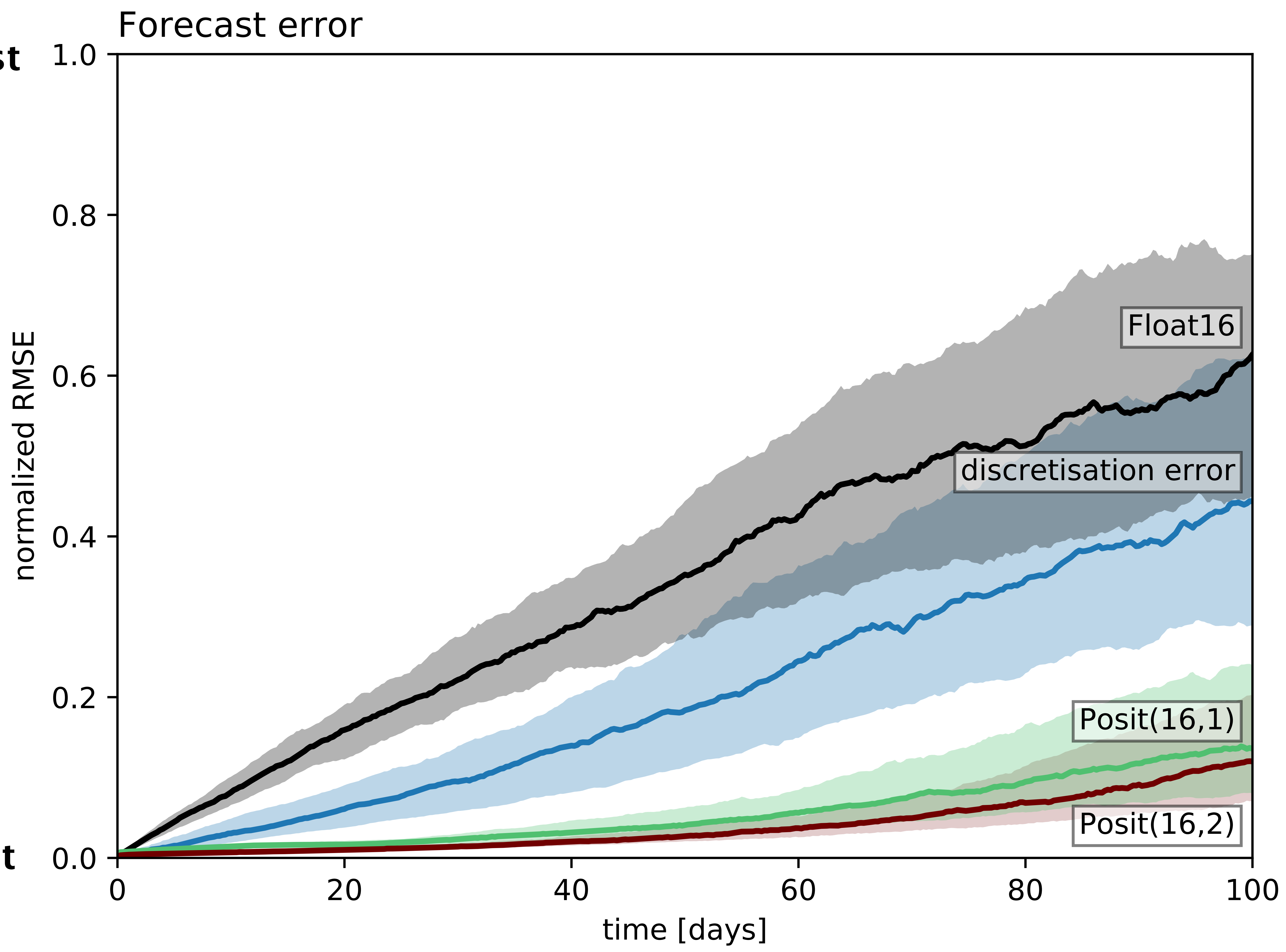
Forecasts in the shallow water model



Forecasts in the shallow water model

Useless forecast

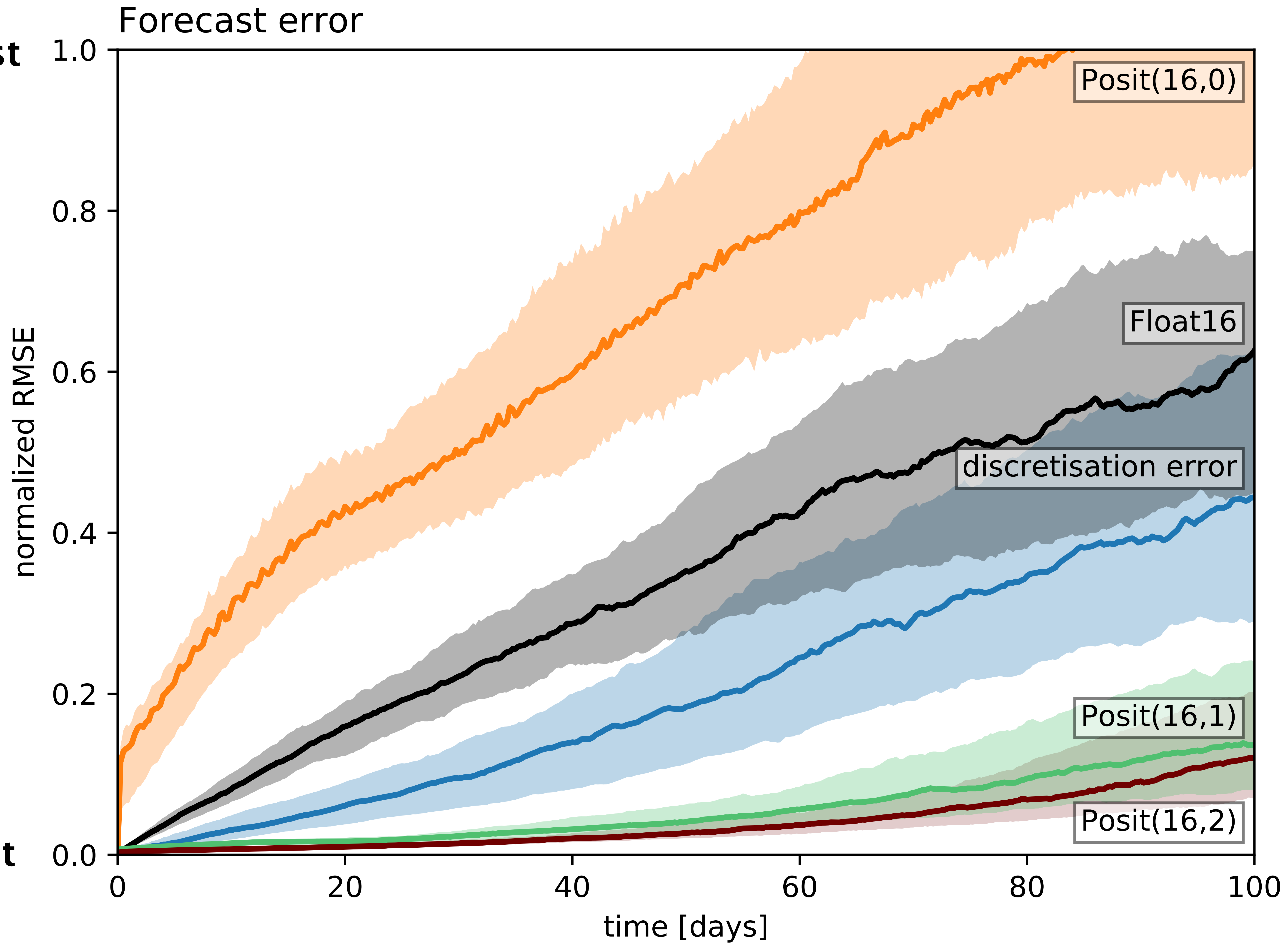
Perfect forecast



Forecasts in the shallow water model

Posits for weather and climate models
CoNGA'19
Milan Klöwer

Useless forecast



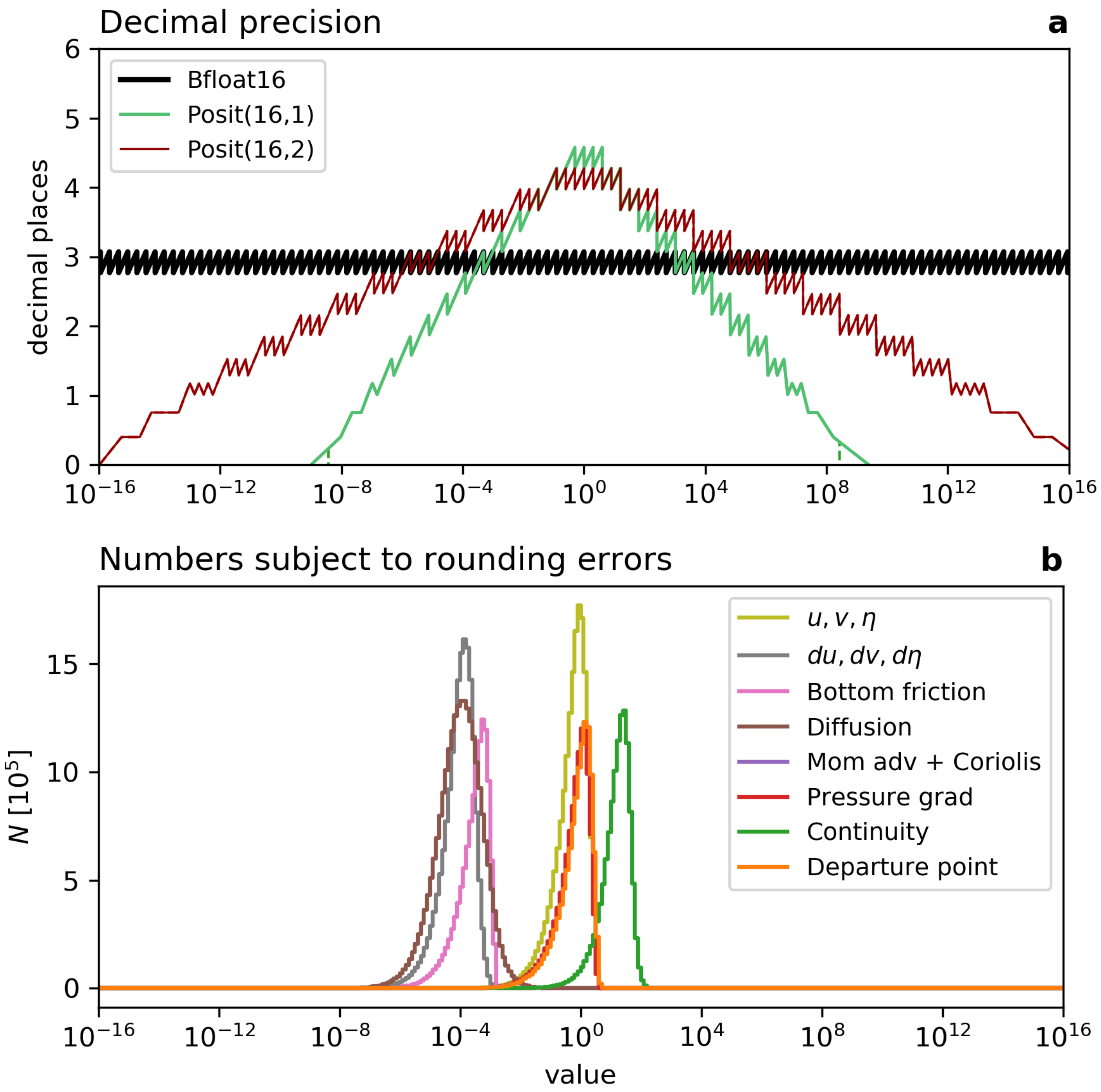
Perfect forecast

Posits clearly outperform floats!

Posit errors will be dwarfed by other sources of error.

Dynamic range of Posit(16,2) has a great potential for more complex models

Conclusion on posits in the shallow water model



Posits for weather and climate models
CoNGA'19
Milan Klöwer

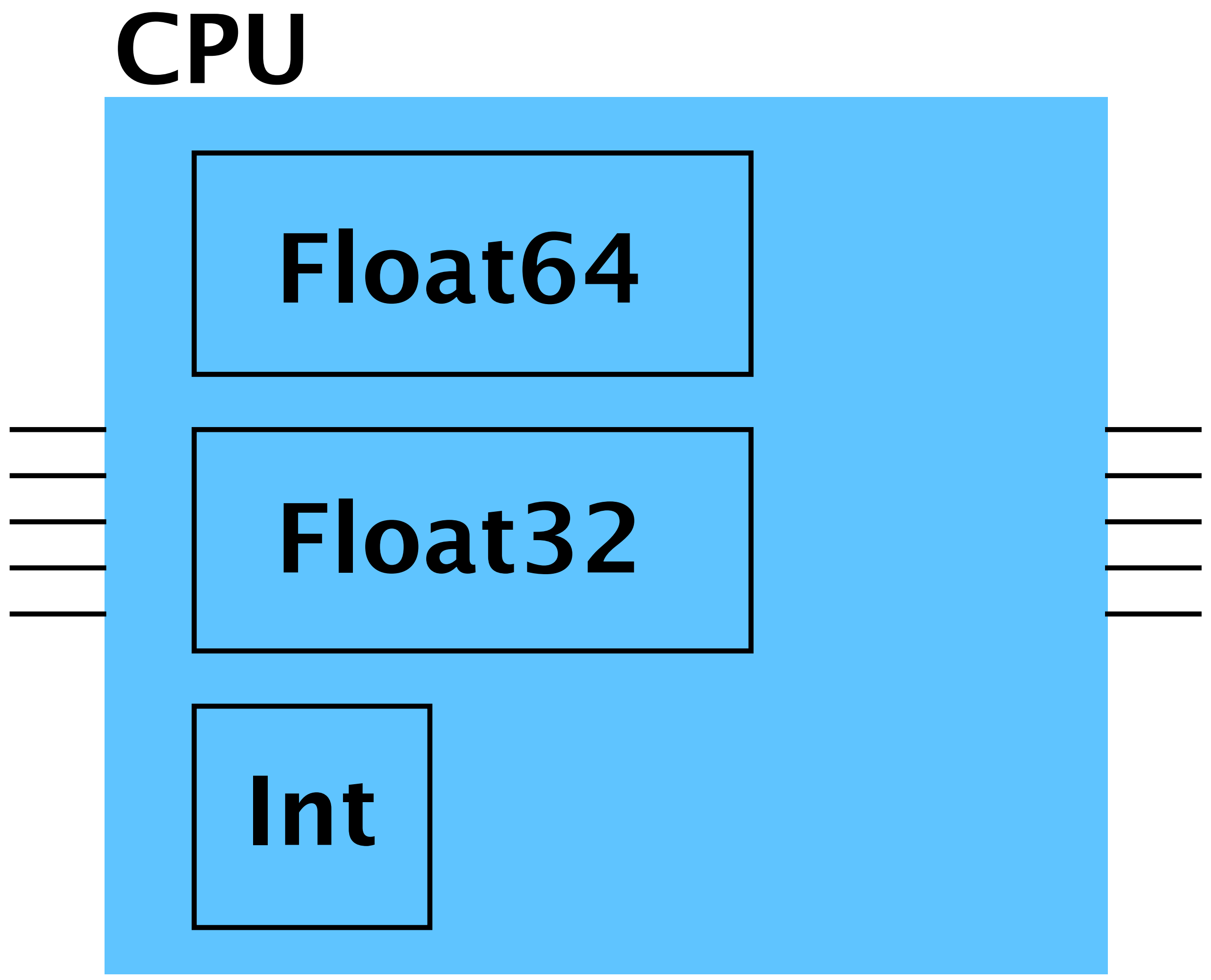
PDE-type problems / computational fluid dynamics benefit from a 10^{16} – 10^{-16} dynamic range and the increased precision around 1.

Float16 okay, but bfloat16 useless.

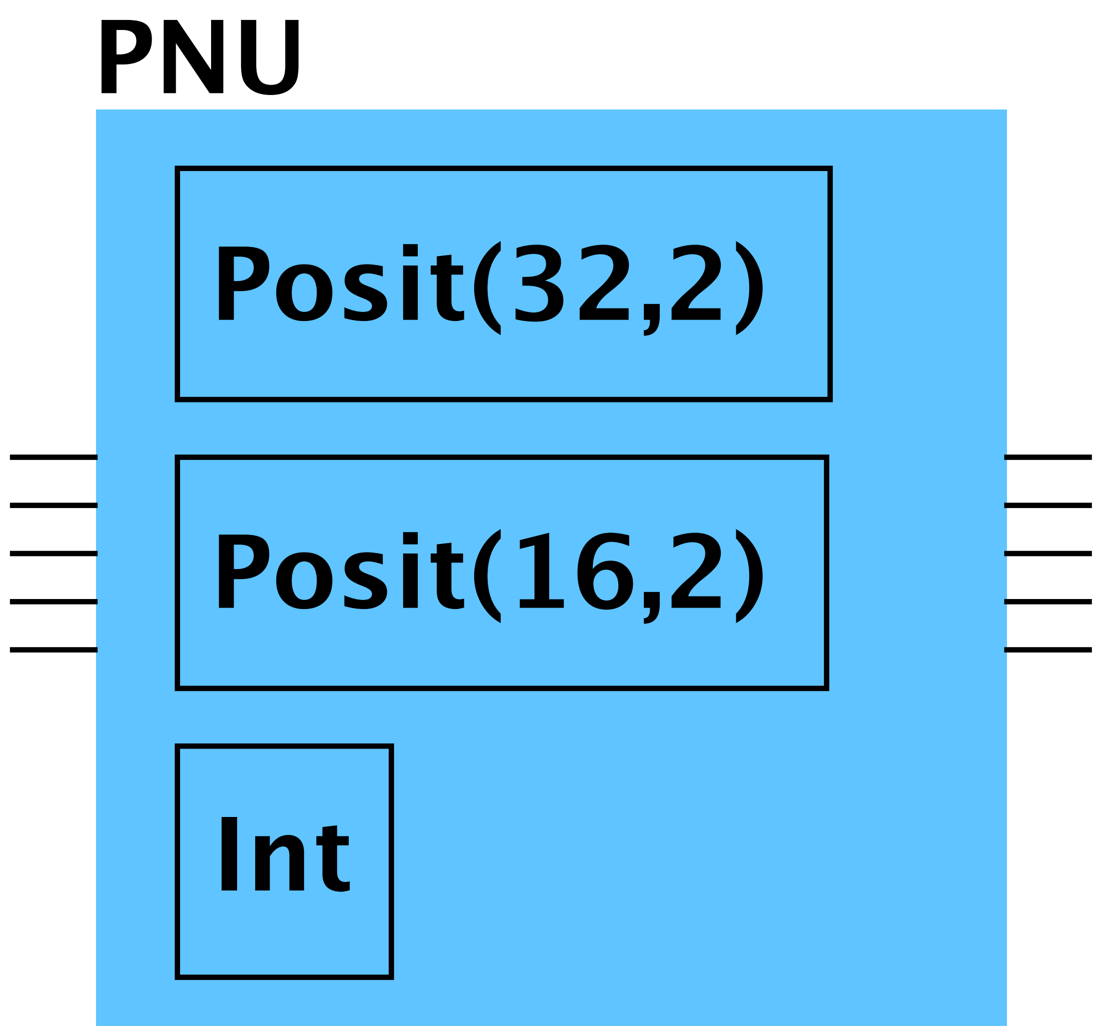
Posit(16,2) has a great potential for Earth-system models!

Summary

What we have



What we want



For now

> Use models with 32bit floats, 40-50% speed-up
> Rewrite algorithms for low precision
> Test Posit(16,2) in various models
> Wait

For future

> Switch Float32 -> Posit32
> Test Posit32 -> Posit16

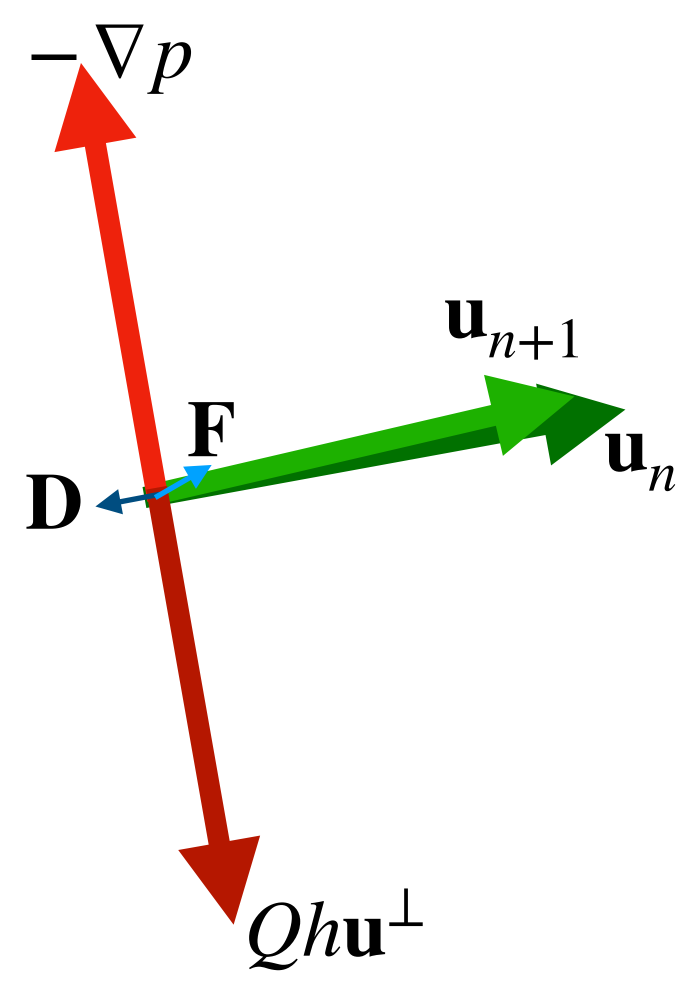
Get in touch!

milan.kloewer@physics.ox.ac.uk
@milankloewer
github.com/milankl/juls

Perspectives for quires

$$u^{n+1} = u^n + RK_u\left(Qhv + \partial_x p + D_x + F_x\right)$$

(1) (2) (3) (4) (5)



Which is the dominant balance of the sum? Only clear at runtime! In Float16

$$-0.0002 \leftarrow 10^{-1} + 1 - 1 + 10^{-4} - 10^{-1}$$

$$-0.0005 \leftarrow 10^{-1} + 1 - 10^{-1} + 10^{-4} - 1$$

But actually this is

$$0.0001 = 1 - 1 + 10^{-1} - 10^{-1} + 10^{-4}$$