# Physics-Preserving AI-Accelerated Simulations of Turbulent Transport

Frank Jenko, Robin Greif, and Nils Thuerey

Max Planck Institute for Plasma Physics & Technical University of Munich

*Vienna, November 30, 2023*

# Fusion research: At the forefront of supercomputing since the '70s

## NERSC HISTORY

### Powering Scientific Discovery Since 1974

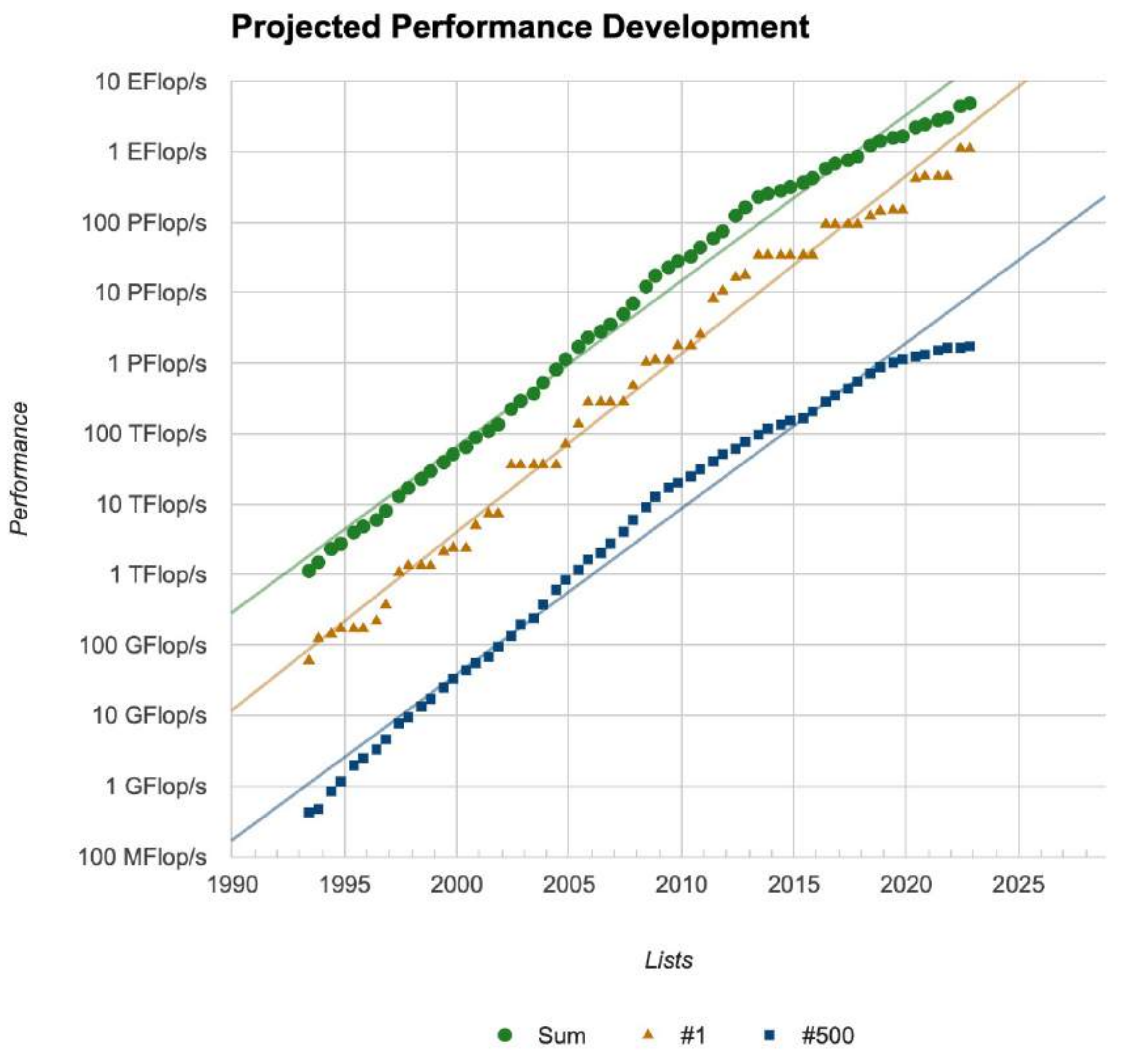
**Contact: Jon Bashor, jbashor@lbl.gov, +1 510 486 5849**

The oil crisis of 1973 did more than create long lines at the gas pumps — it jumpstarted a supercomputing revolution.

The quest for alternative energy sources led to increased funding for the Department of Energy's Magnetic Fusion Energy program, and simulating the behavior of plasma in a fusion reactor required a computer center dedicated to this purpose. Founded in 1974 at Lawrence Livermore National Laboratory, the Controlled Thermonuclear Research Computer Center was the first unclassified supercomputer center and was the model for those that followed.

Over the years the center's name was changed to the National Magnetic Fusion Energy Computer Center and later the National Energy Research Supercomputer Center (NERSC). In 1983 NERSC's role was expanded beyond the fusion program, and it began providing general computing services to all of the programs funded by the DOE Office of Energy Research (now the Office of Science). The current name was adopted in 1996 when NERSC relocated to Lawrence Berkeley National Laboratory and merged with Berkeley Lab's Computing Sciences program. The name change — from "Supercomputer Center" to "Scientific Computing Center" — signaled a new philosophy, one of making scientific computing more productive, not just providing supercomputer cycles.

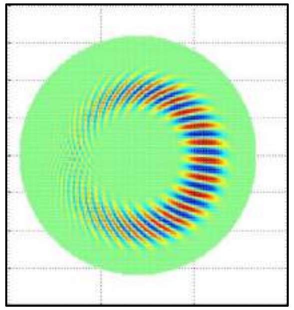
# Performance development of supercomputers since the ‘90s

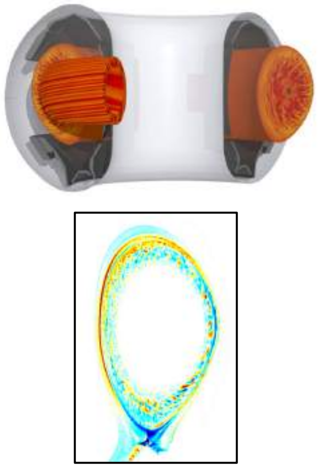


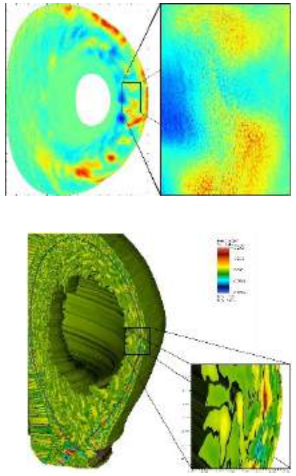

www.top500.org

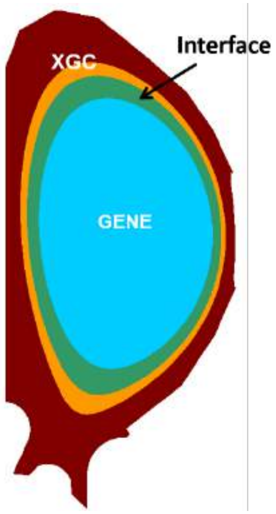
# From highly idealized models to virtual fusion systems

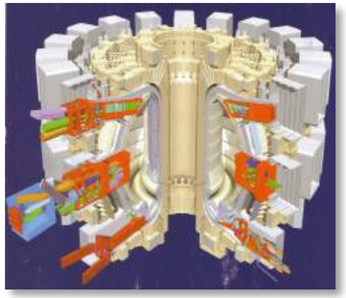
Increasing fidelity & modeling capability with increasing computing power

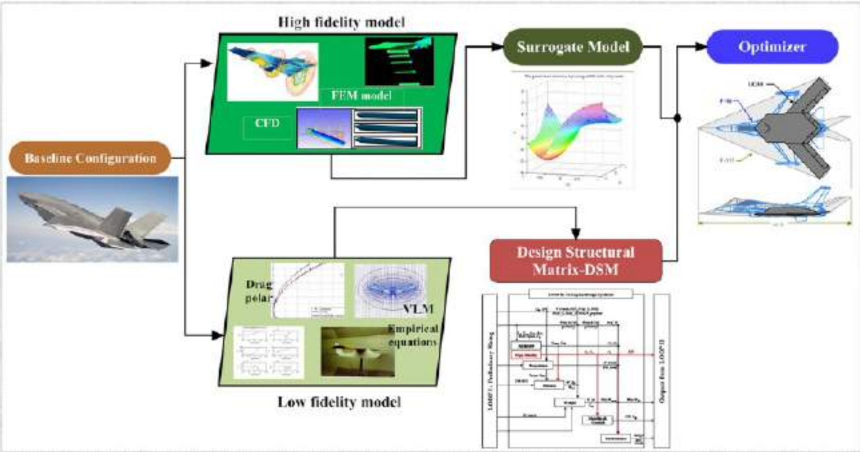











more complete physics

faster

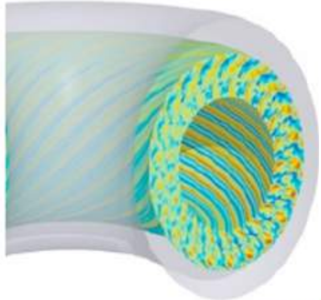
**Multi-fidelity approach:**

- HiFi models for reliable extrapolation/prediction
- LoFi models (based on HiFi models) for high-throughput computing & real-time applications (incl. control)

Both are needed – together

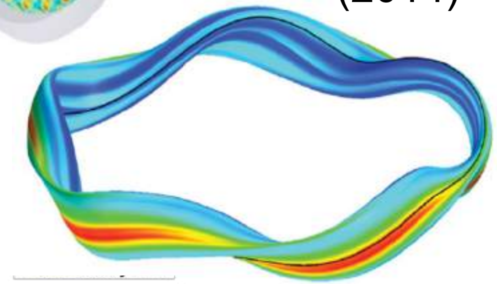
# The GENE family of grid-based gyrokinetic turbulence codes

T flux tube (2000)
S flux tube (2002)

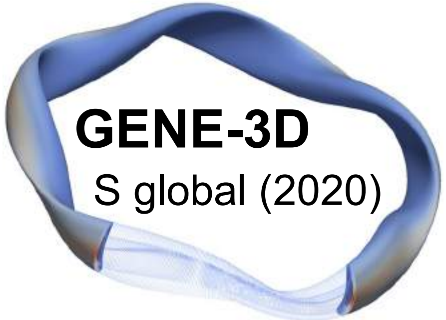


## GENE

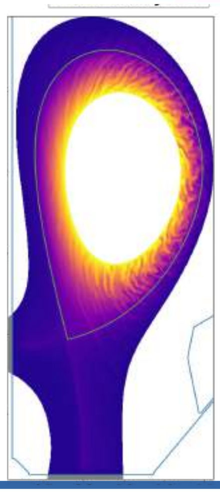
S full surface (2014)



T global (2011)



**GENE-3D**
S global (2020)

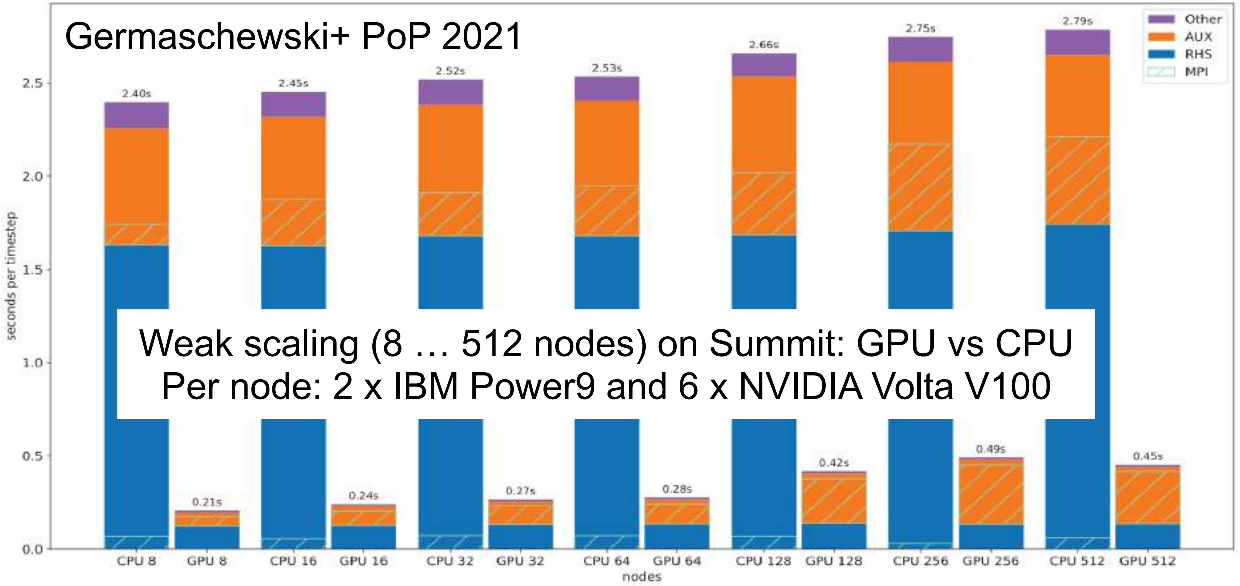


**GENE-X**
T/S global (2021)

T = tokamak
S = stellarator

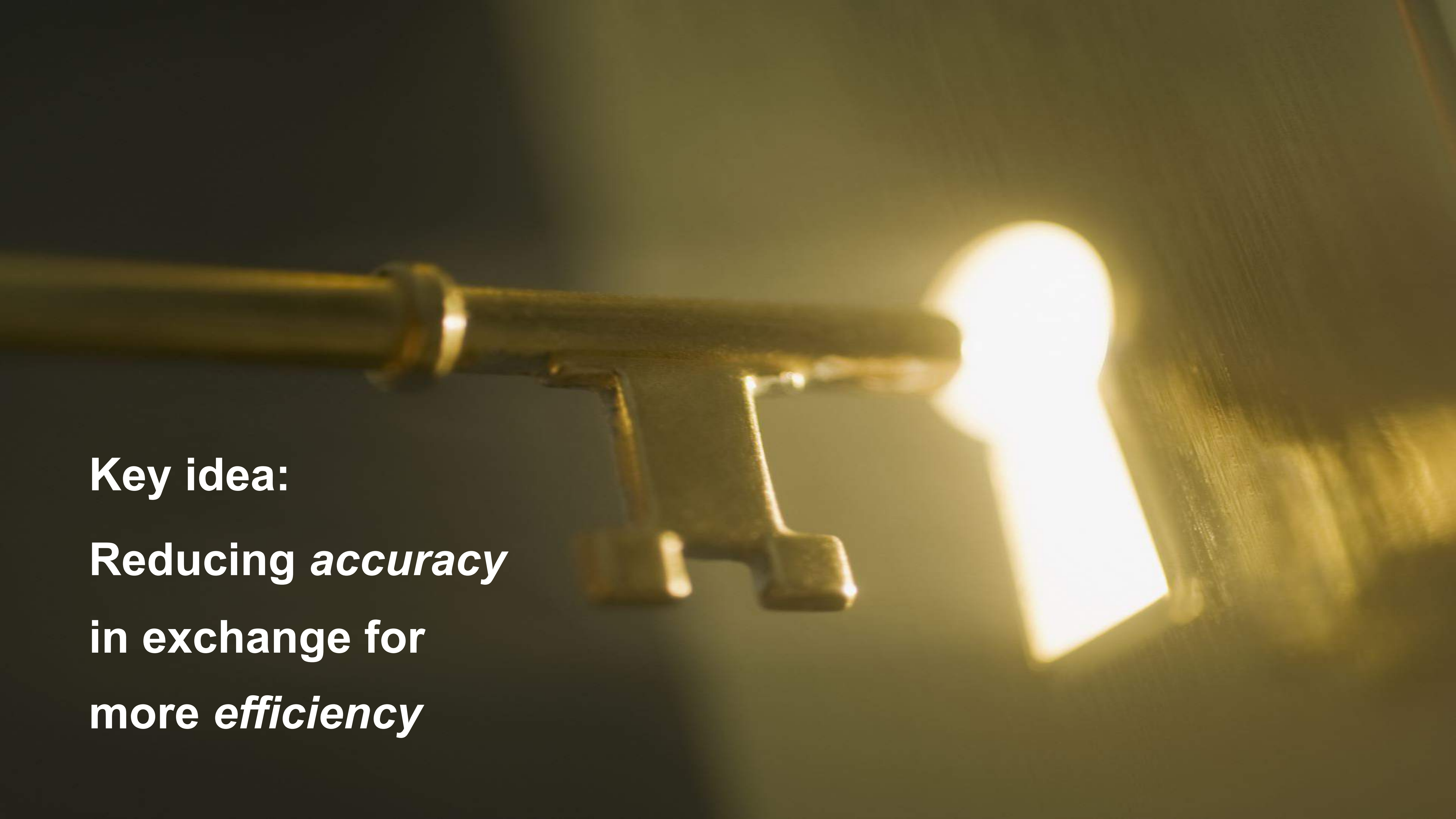
**Entering the exascale era (ECP & Plasma-PEPSC projects)**

Germaschewski+ PoP 2021




Weak scaling (8 … 512 nodes) on Summit: GPU vs CPU
Per node: 2 x IBM Power9 and 6 x NVIDIA Volta V100

GENE User Sites



**Key idea:**

**Reducing *accuracy***

**in exchange for**

**more *efficiency***

# AI Meets Large-Scale Computing

## “Science at extreme scales: Where big data meets large-scale computing”


**Interdisciplinary Long Program @UCLA**

September 12 - December 14, 2018

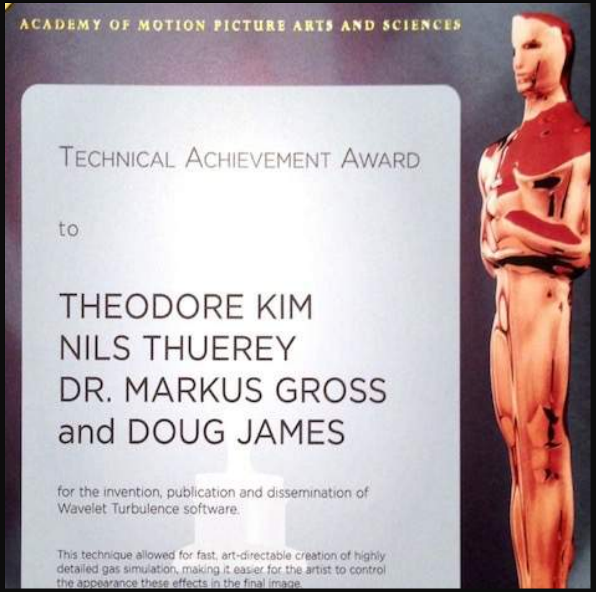
200+ participants, 50+ long-term participants

Speaker list includes:

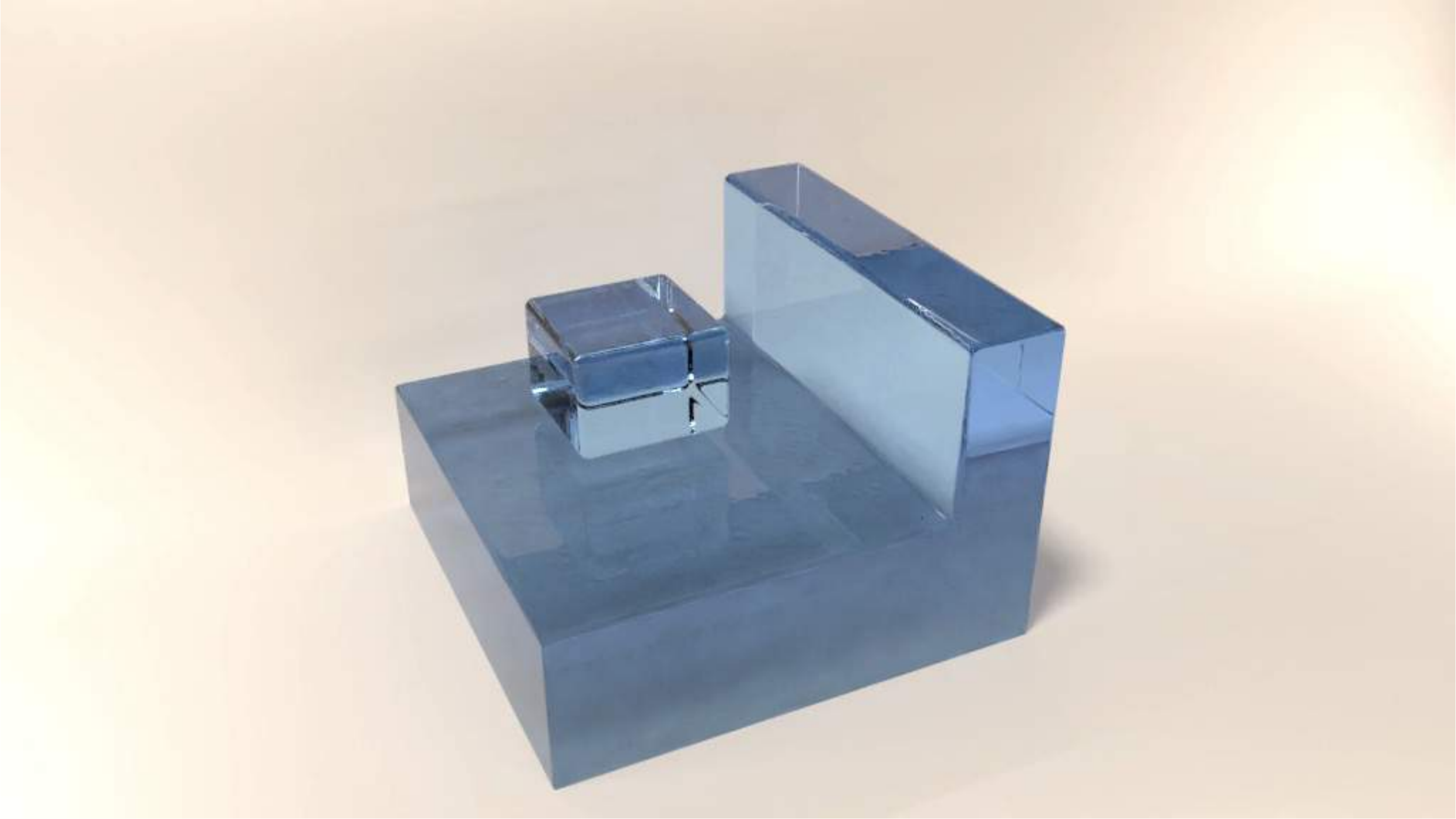
- Yann LeCun (Director of AI Research @Facebook)
- Emmanuel Candes (Stanford University)
- Rajat Monga (Google)
- Matthias Troyer (Microsoft)
- James Sexton (IBM)
- Adrian Tate (Cray)
- Alan Lee (AMD)

# Tech Oscar 2013 for Nils Thuerey

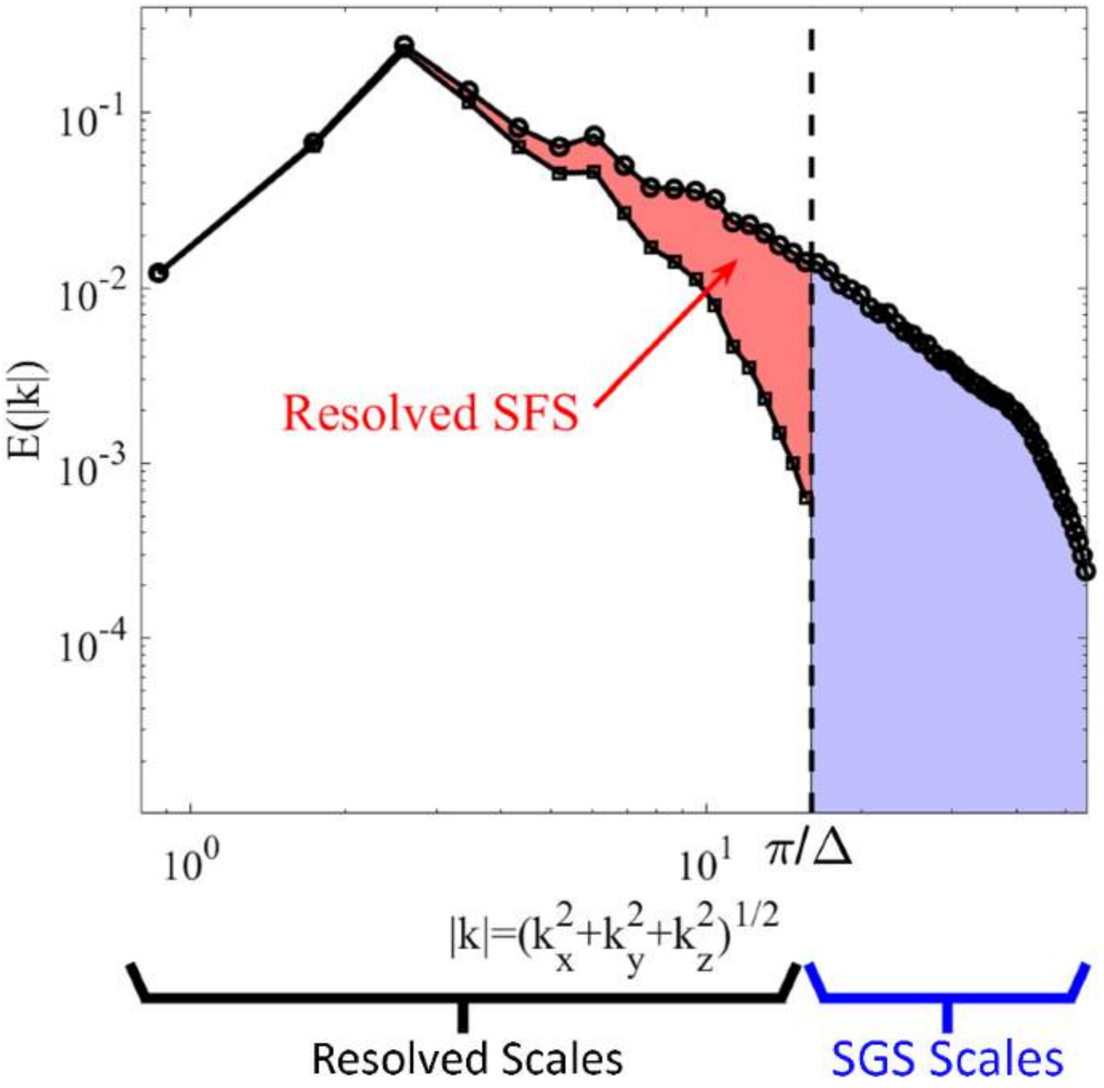
The quartet was honoured for developing a technique that makes it easier for visual effects artists to control the appearance of gas and smoke on film




# AI Acceleration of Fluid Dynamics Simulations



# Large Eddy Simulations



# 2D Hasegawa-Wakatani Model

# Two-Fluid Model for Plasma Turbulence

$$\partial_t n = c_1(n - \phi) - [\phi, n] - \kappa_n \partial_y \phi - \nu \nabla^{2N} n$$
$$\partial_t \Omega = c_1(n - \phi) - [\phi, \Omega] \qquad\qquad - \nu \nabla^{2N} \phi$$

$$with \ \ \Omega = \nabla_\perp^2 \phi$$

$n$: Density (t, y, x)

$\Omega$: Vorticity (t, y, x)

$\phi$: Potential (t, y, x)

# Two-Fluid Model for Plasma Turbulence

$$\partial_t n = c_1(n - \phi) - [\phi, n] - \kappa_n \, \partial_y \phi - \nu \nabla^{2N} n$$
$$\partial_t \Omega = c_1(n - \phi) - [\phi, \Omega] \qquad\qquad - \nu \nabla^{2N} \phi$$

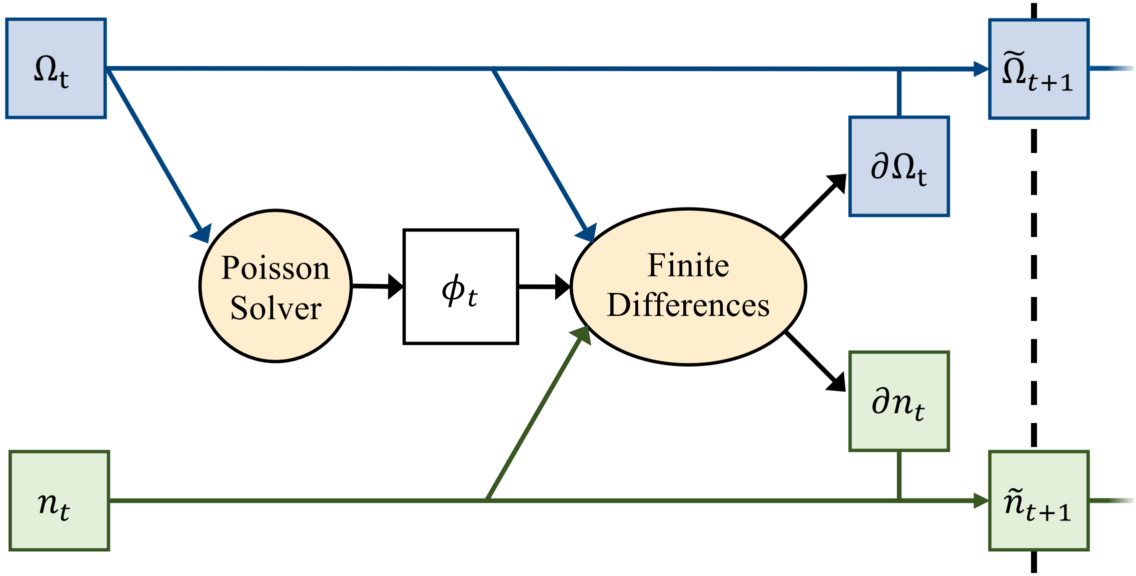
$$with \;\; \nabla_\perp^{-2} \Omega = \phi$$

| | |
|---|---|
| $\partial_t$ | Runge Kutta 4th order, Euler, leapfrog methods |
| $[\cdot,\cdot]$ | Arakawa Scheme |
| $\partial_y$ | Central Finite Difference |
| $\nabla^{2N}$ | Repeated Central Finite Difference |
| $\nabla^{-2}$ | Fourier Poisson Solver |

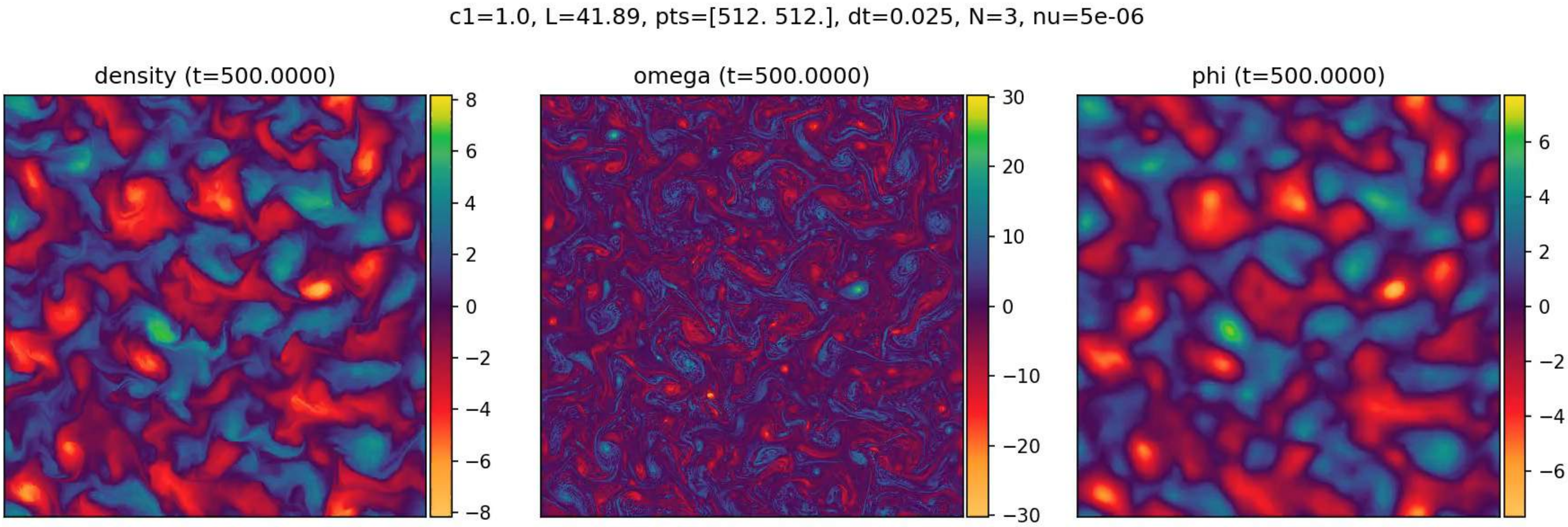
# Two-Fluid Model for Plasma Turbulence

$$\partial_t n = c_1(n - \phi) - [\phi, n] - \kappa_n \, \partial_y \phi - \nu \nabla^{2N} n$$
$$\partial_t \Omega = c_1(n - \phi) - [\phi, \Omega] \qquad\qquad - \nu \nabla^{2N} \phi$$

$$with \;\; \phi = \nabla_\perp^{-2} \Omega$$



# Dynamics of the 2D Hasegawa-Wakatani Model



# Typical Resolution Requirements

$$\partial_t n = c_1(n - \phi) - [\phi, n] - \kappa_n \, \partial_y \phi - \nu \nabla^{2N} n$$
$$\partial_t \Omega = c_1(n - \phi) - [\phi, \Omega] \qquad\qquad - \nu \nabla^{2N} \phi$$

$$with \;\; \nabla_\perp^{-2} \Omega = \phi$$

Spatial Resolution: 512 x 512

Time Step Size: 0.025

Time Steps Required: 1,000+

# Properties to Preserve

$$\partial_t n = c_1(n - \phi) - [\phi, n] - \kappa_n \, \partial_y \phi - \nu \nabla^{2N} n$$
$$\partial_t \Omega = c_1(n - \phi) - [\phi, \Omega] \qquad\qquad\qquad - \nu \nabla^{2N} \phi$$

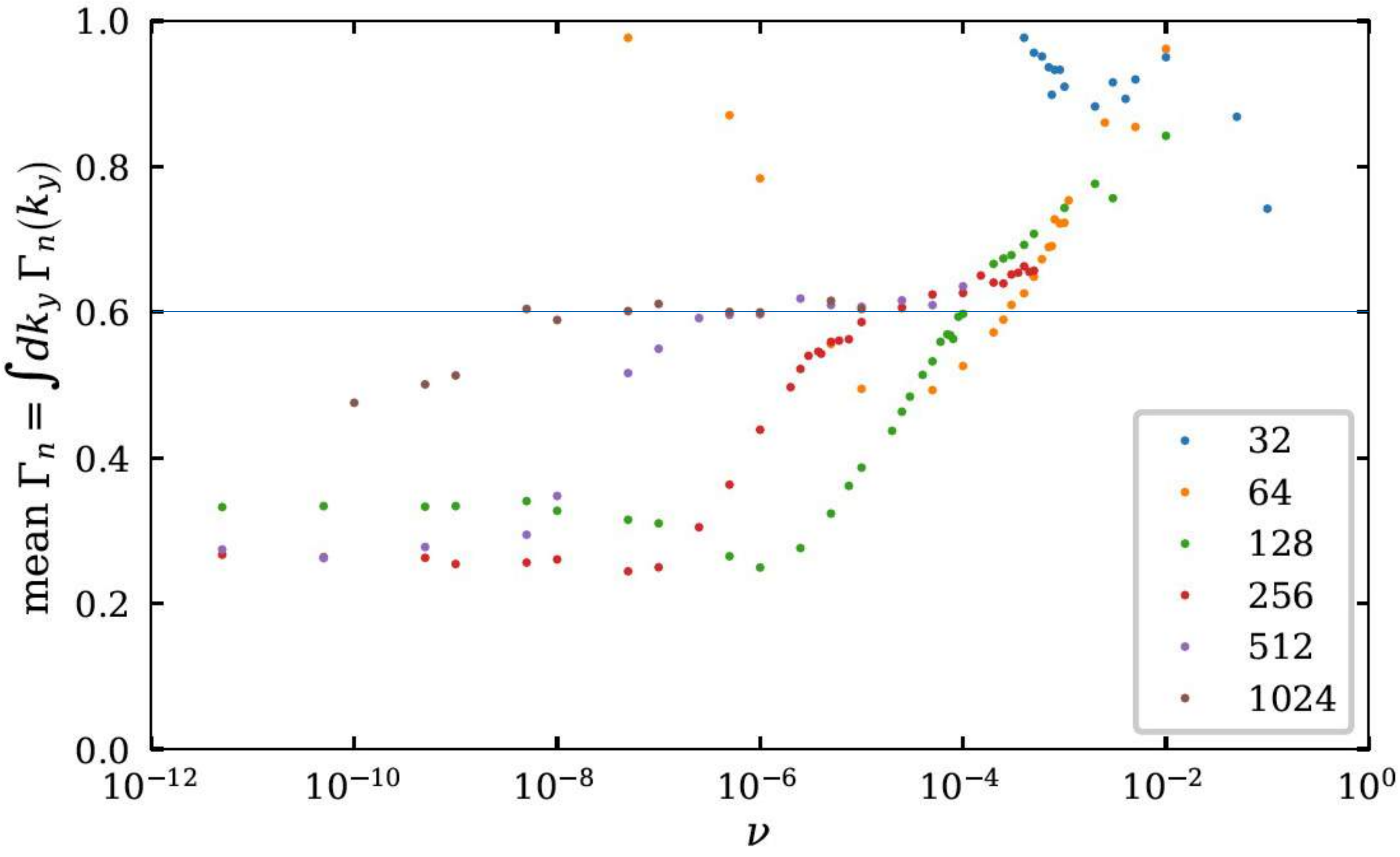
$$with \;\; \nabla_\perp^{-2} \Omega = \phi$$

$$\Gamma_n(t) = -\iint d^2x \;\, n \, \partial_y \phi = -\int d_{k_y} \;\, i \, k_y \, n(k_y) \, \phi(k_y)^* \quad \Big] \; source$$

$$\Gamma_c(t) = c_1 \iint d^2x \, (n \; - \; \phi)^2$$
$$\partial_t E = \Gamma_n \; - \; \Gamma_c \; - \iint d^2x \, (n \, \nabla^{2N} n \; - \; \phi \nabla^{2N} \phi)$$
$$\partial_t U = \Gamma_n + \iint d^2x \, (n \; - \; \Omega)(\nabla^{2N} n \; - \; \nabla^{2N} \phi)$$

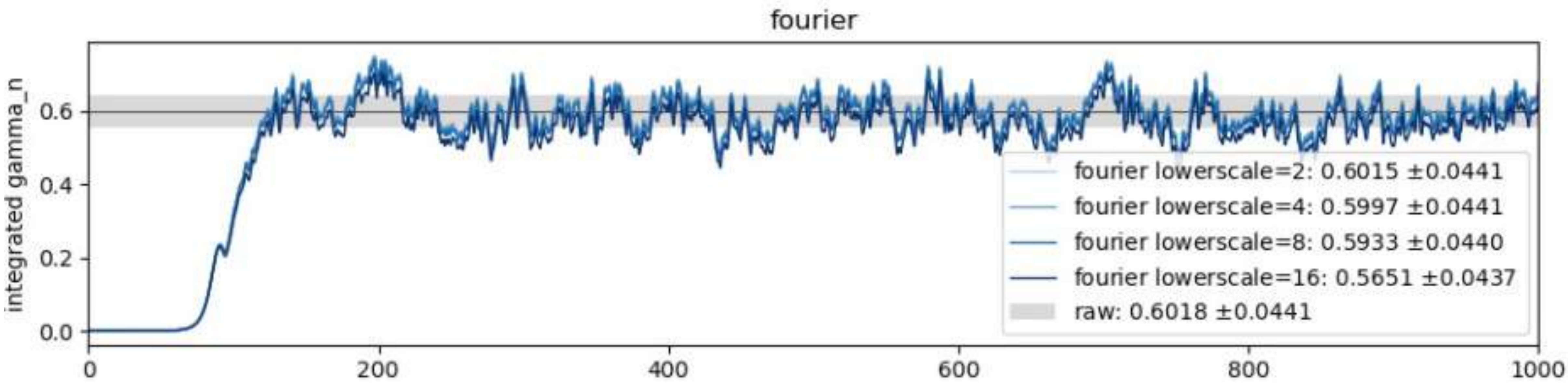
$\Big]$ *sinks*

# What is the correct baseline simulation?



# Large-Scale Quantities of Interest

# $\Gamma_n$: Downsampling in Fourier Space




*A posteriori* reduction of 256x is possible w/o significant loss of accuracy

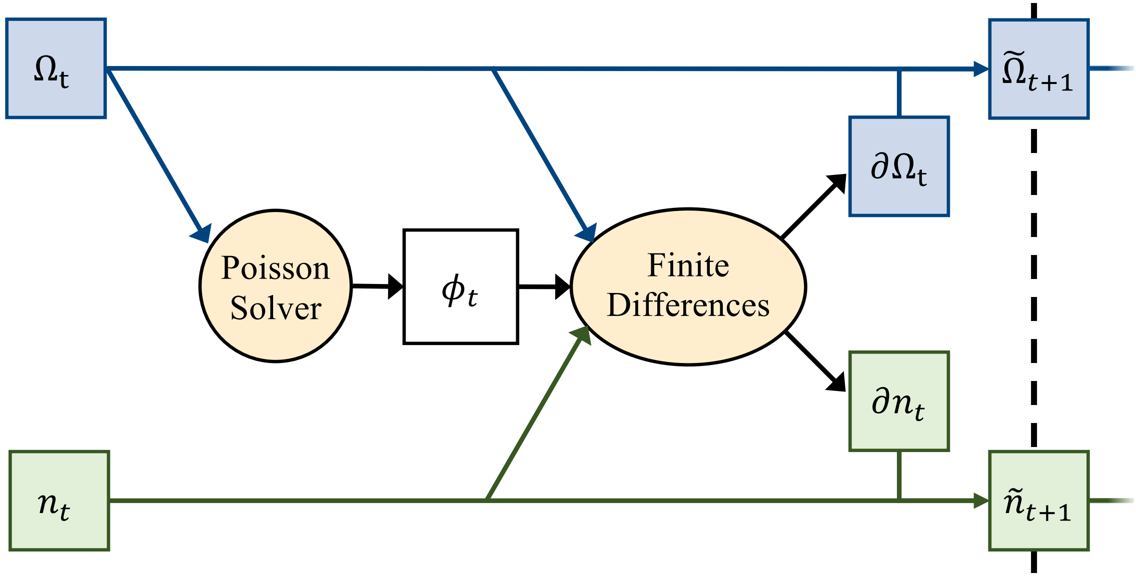
*A priori* reduction results in significant errors

# AI-Accelerated Simulations

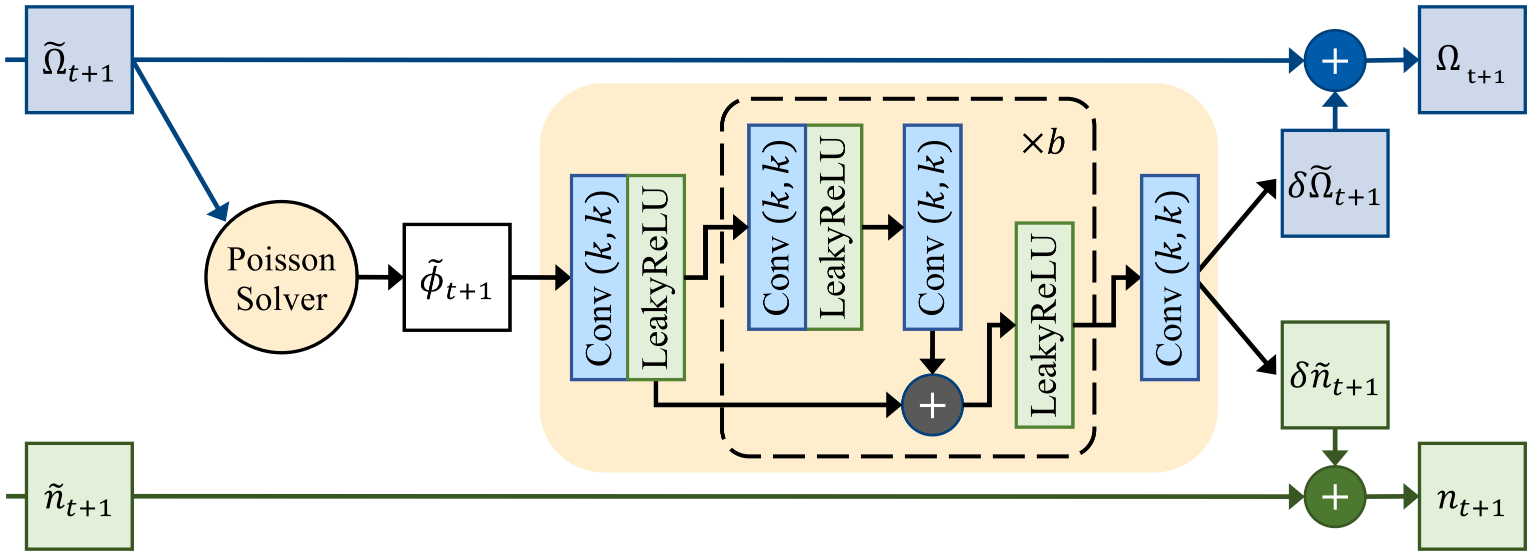
# Predictor Step at Reduced Resolution

$$\partial_t n = c_1(n - \phi) - [\phi, n] - \kappa_n \, \partial_y \phi - \nu \nabla^{2N} n$$
$$\partial_t \Omega = c_1(n - \phi) - [\phi, \Omega] \qquad\qquad - \nu \nabla^{2N} \phi$$

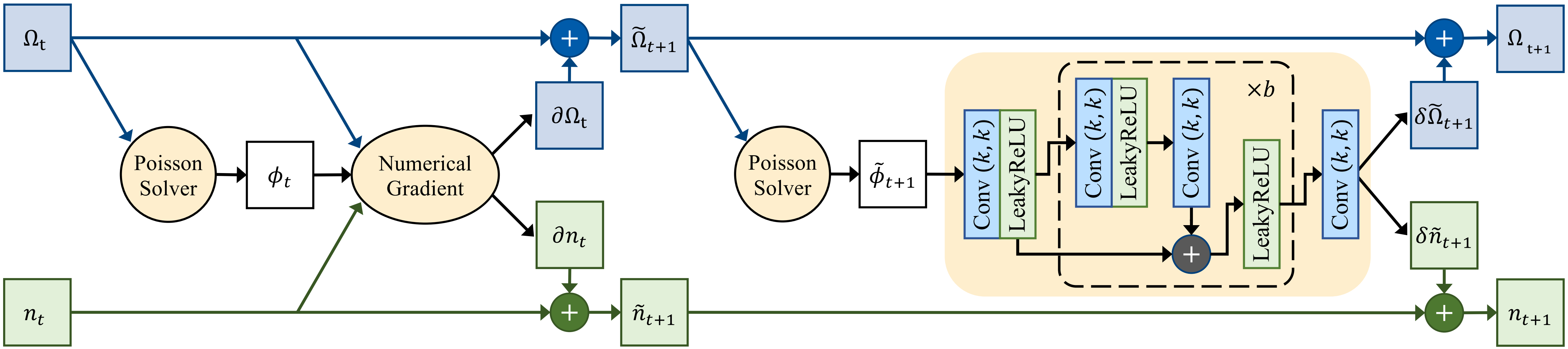
$$with \;\; \nabla_{\perp}^{-2} \Omega = \phi$$



# ML-Based Corrector Step

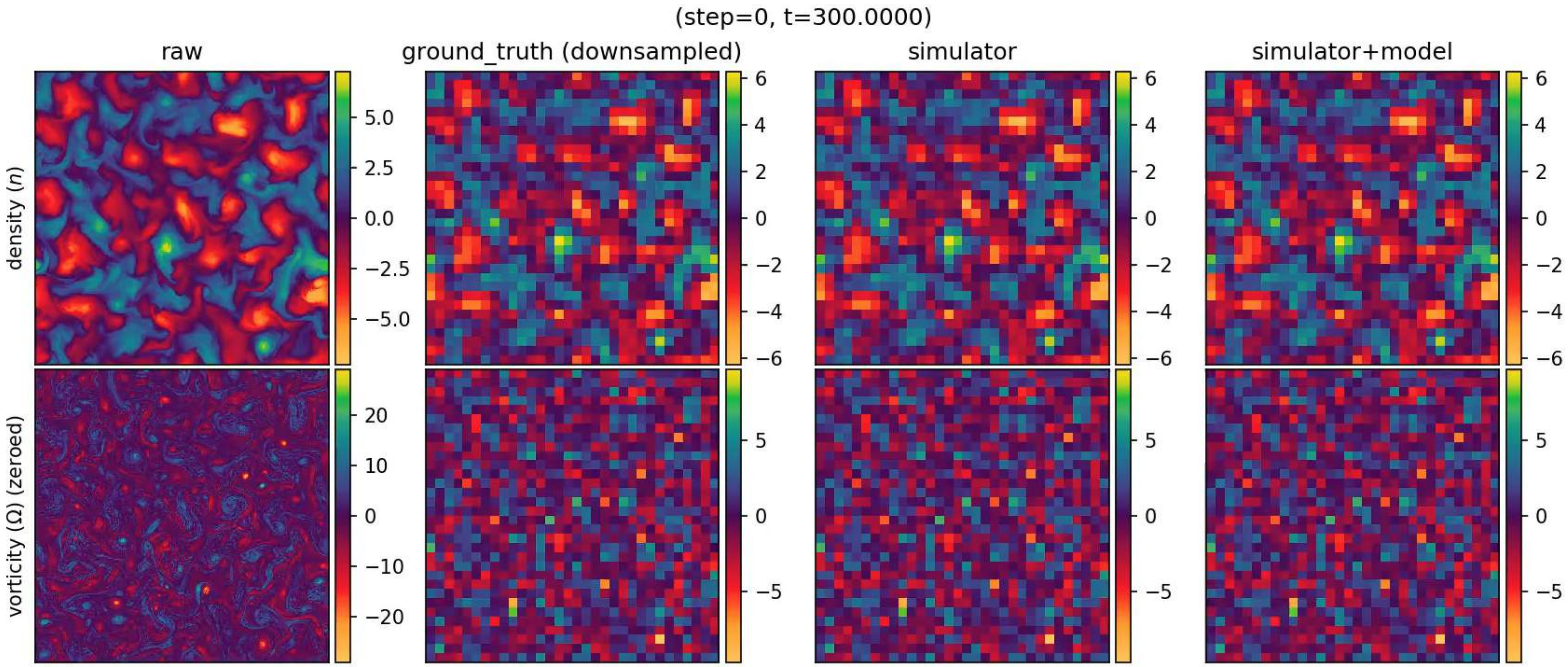


# Putting Things Together

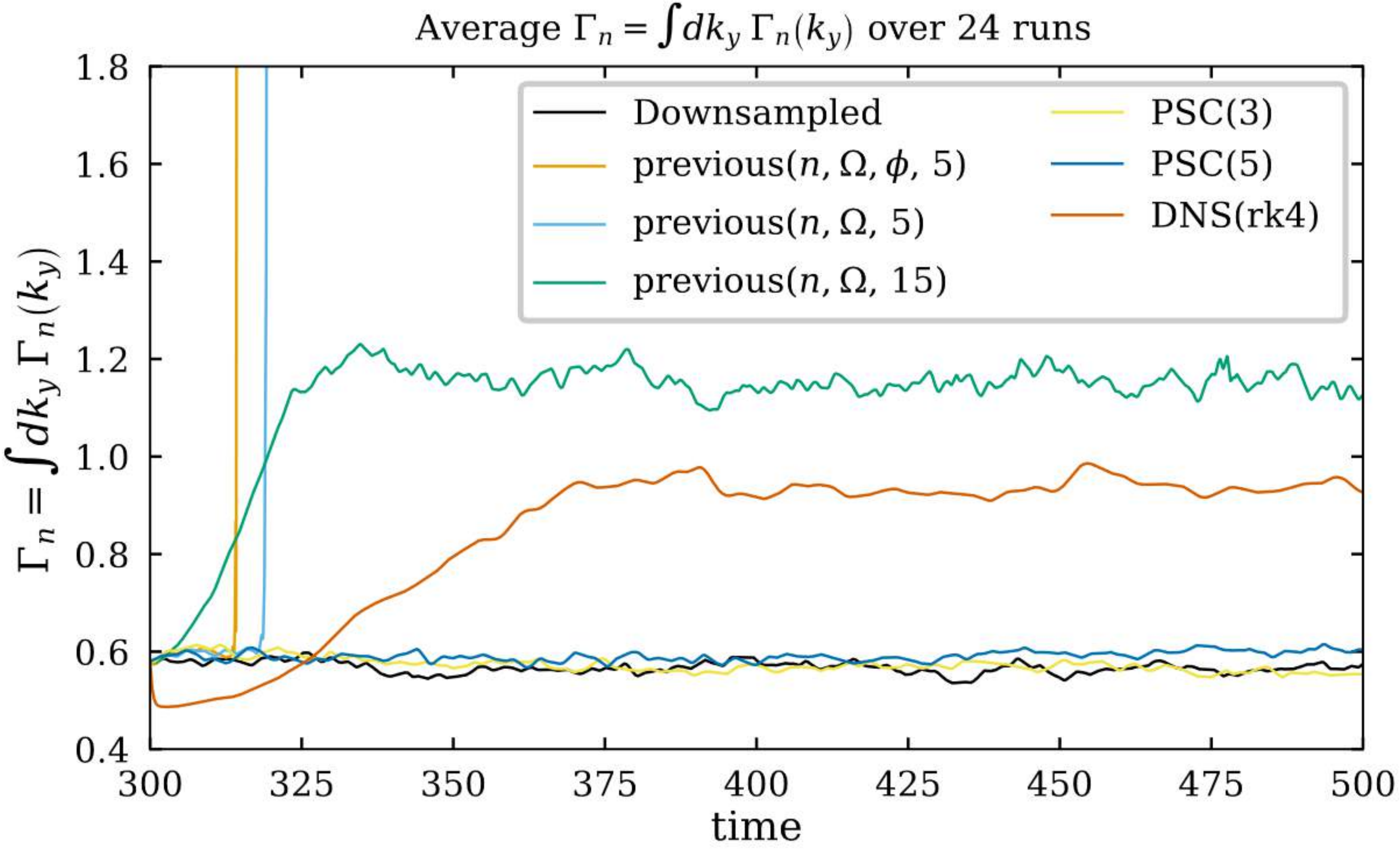


# Results

# Spatial Dynamics, Stable for $10^6$ Time Steps



# Time Traces: Long-Time Stability (!)



# Averages

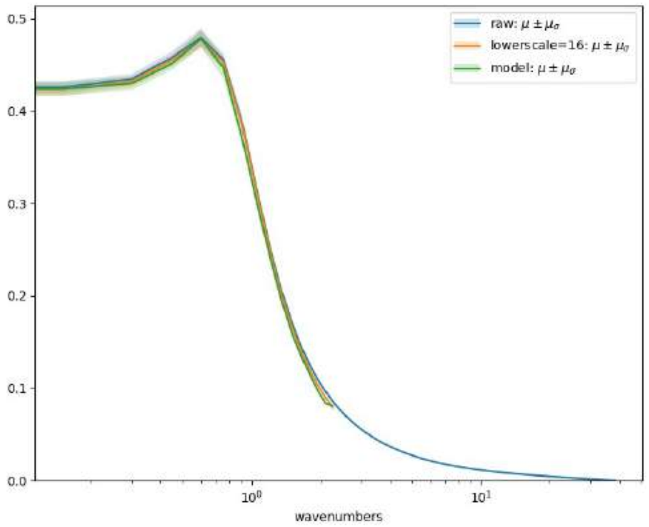
**Table 1. Physical values at 32x32**

| Variant | $\Gamma_n \pm \delta\Gamma_n$ | $\Gamma_c \pm \delta\Gamma_c$ | $E \pm \delta E$ | $U \pm \delta U$ |
|---|---|---|---|---|
| Downsampled | 0.57±0.05 | 0.45±0.03 | 3.69±0.29 | 8.16±0.41 |
| **Our Model** | **0.57±0.05** | **0.46±0.03** | **3.33±0.24** | **8.08±0.40** |
| Previous | 0.84±0.10 | 1.11±0.14 | 8.38±0.81 | 17.23±1.61 |
| DNS | 0.92±0.13 | 0.51±0.08 | 10.89±1.65 | 15.87±2.20 |

# Ja, ja – but is it really physical?!

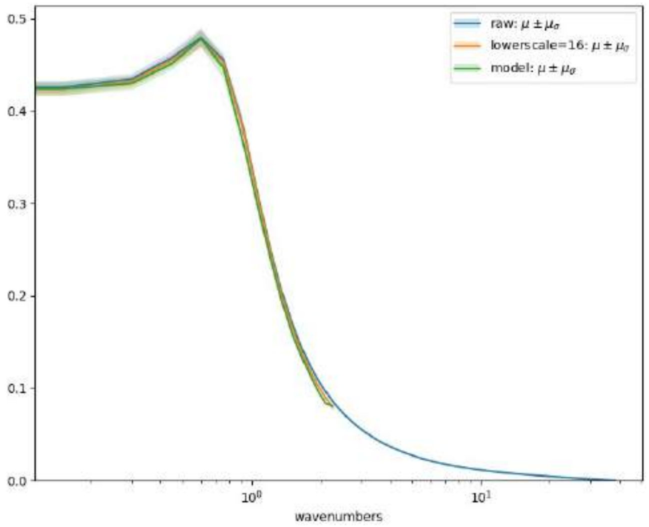
# Preserving Spectral Properties

$k_y$ for density

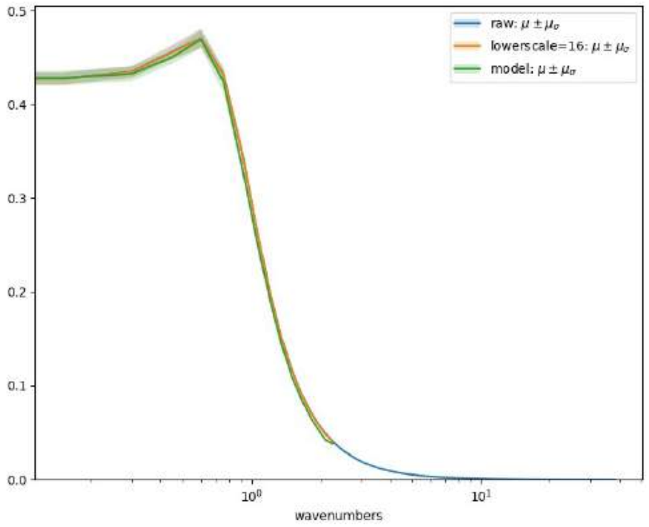


# Preserving Spectral Properties

$k_y$ for density

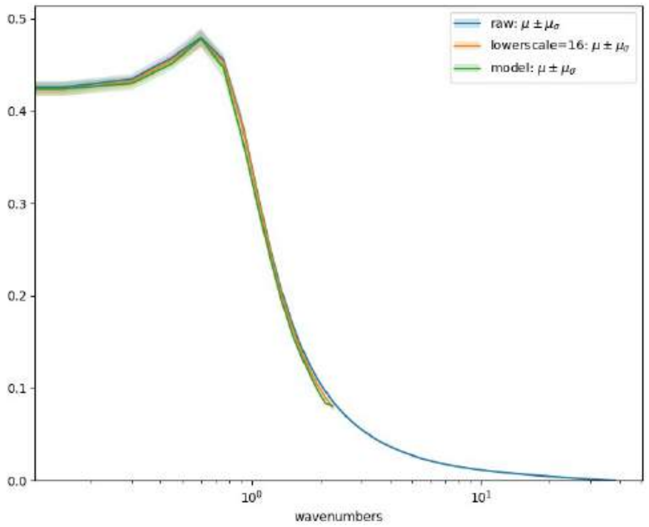



$k_y$ for phi

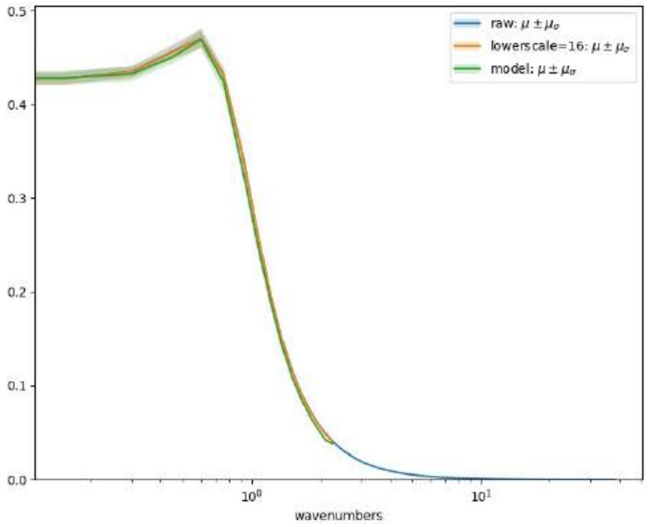


# Preserving Spectral Properties

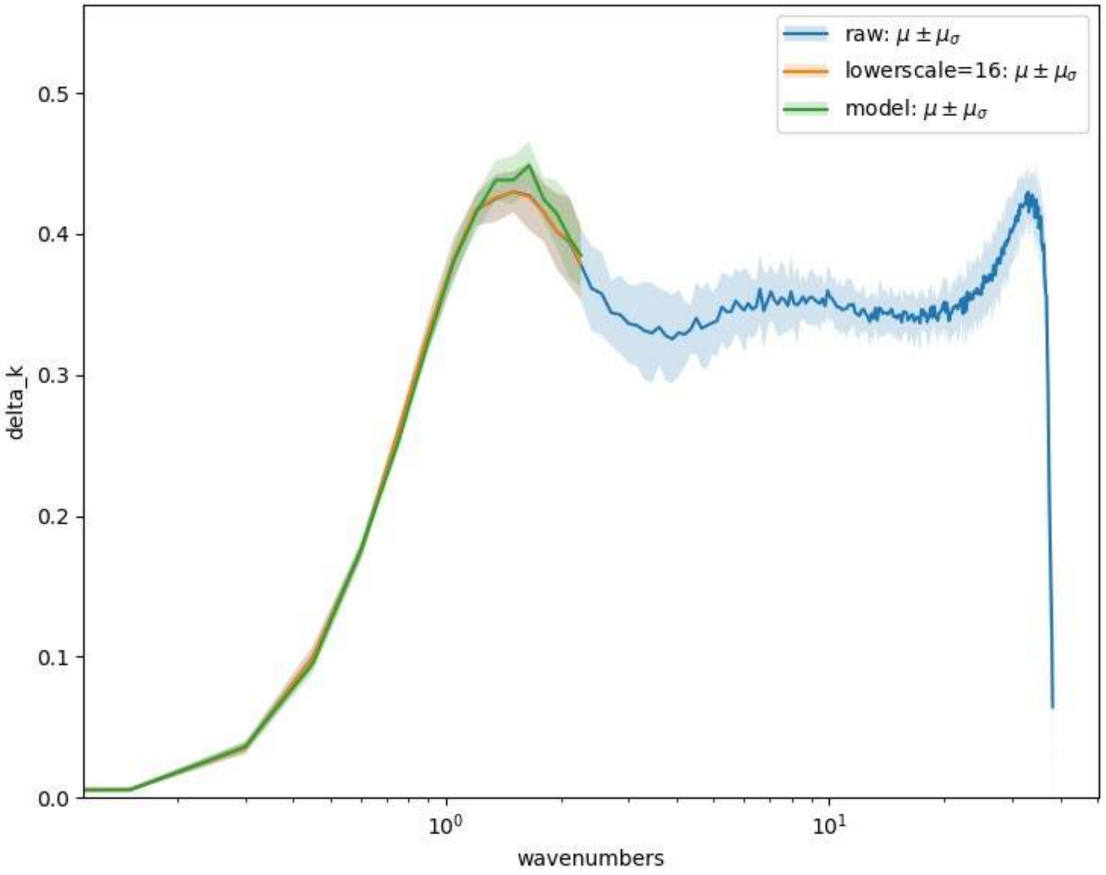
$k_y$ for density




$k_y$ for phi

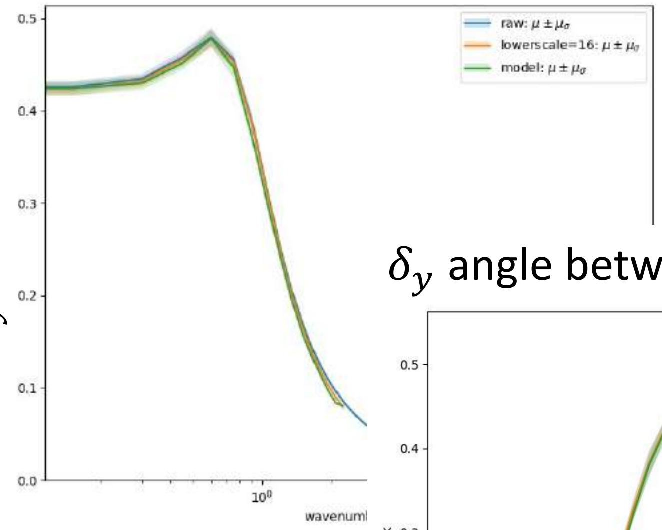



$\delta_y$ angle between density and phi

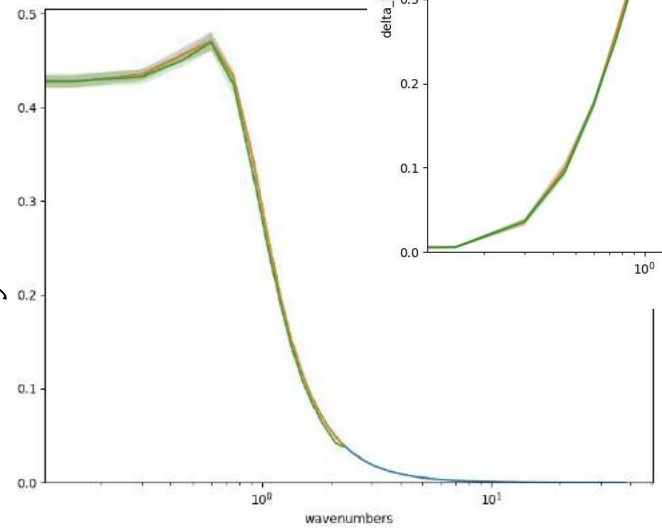


# Preserving Spectral Properties

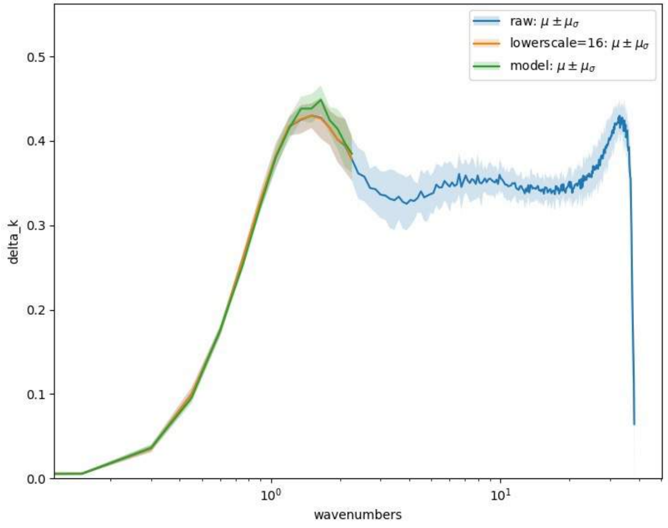
$k_y$ for density



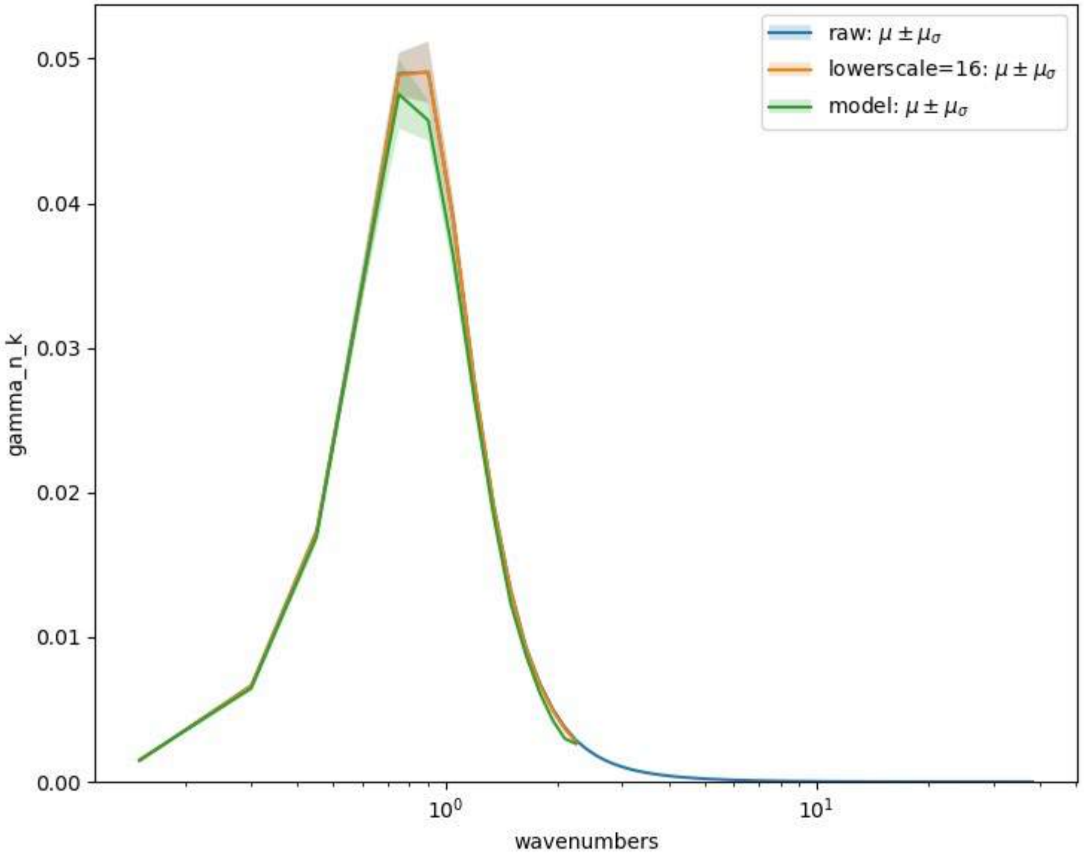

$k_y$ for phi




$\delta_y$ angle between density and phi

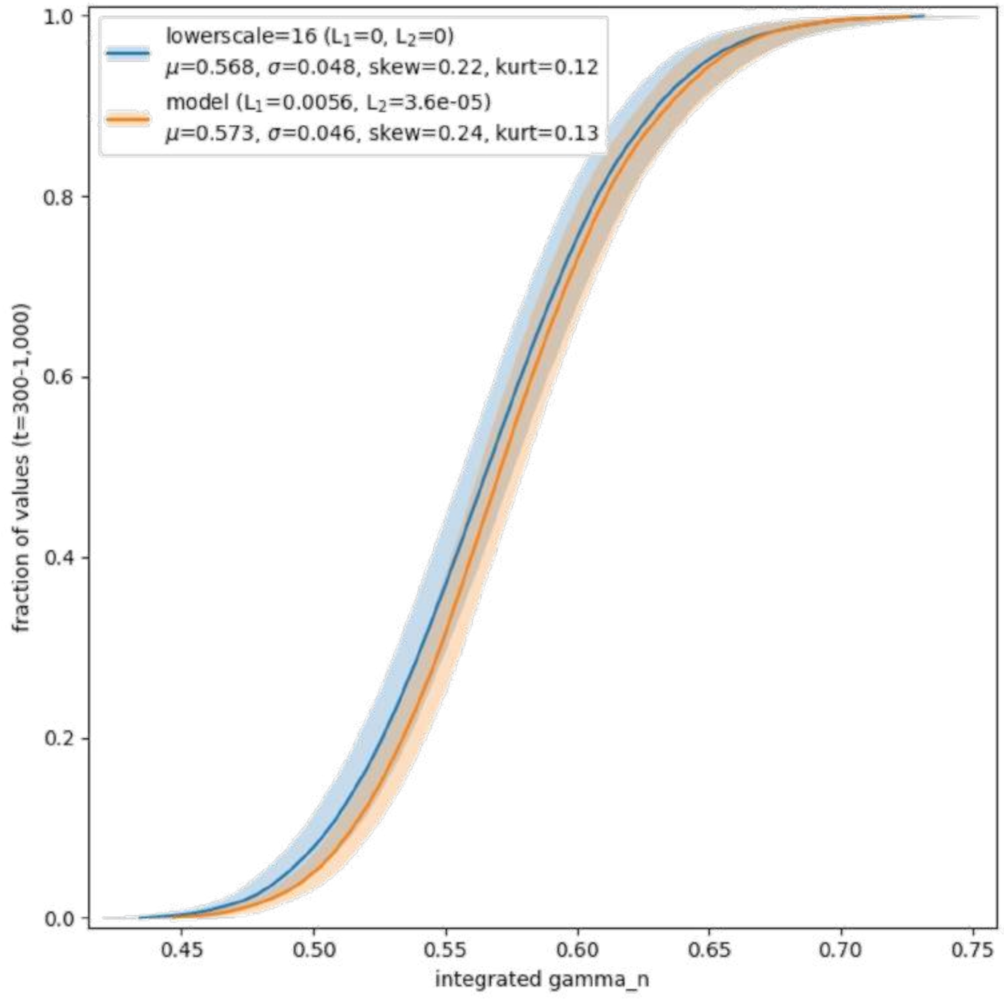



$\Gamma_n(k_y)$ source term spectrum

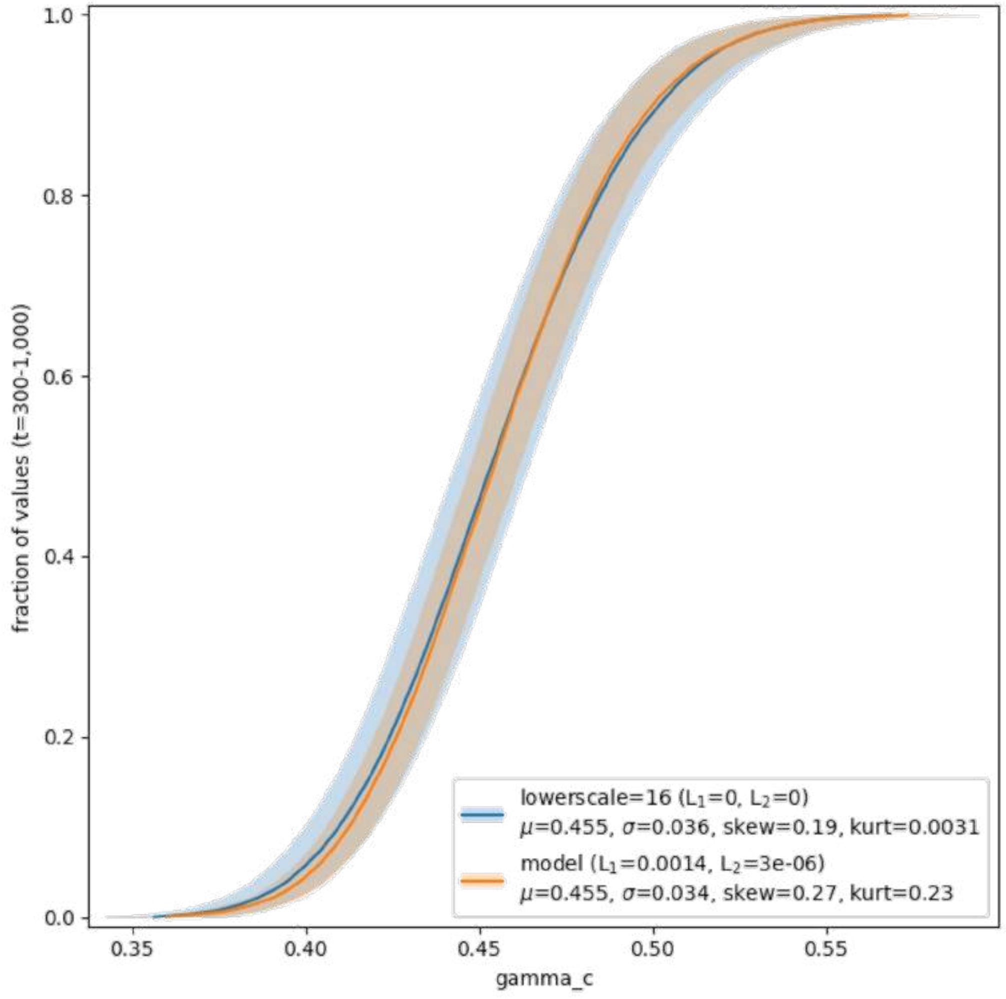


# Preserving Statistical Properties

$\Gamma_n$ cumulative distribution function

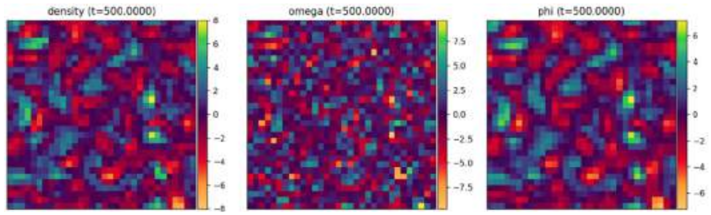



$\Gamma_c$ cumulative distribution function



# What do we gain?

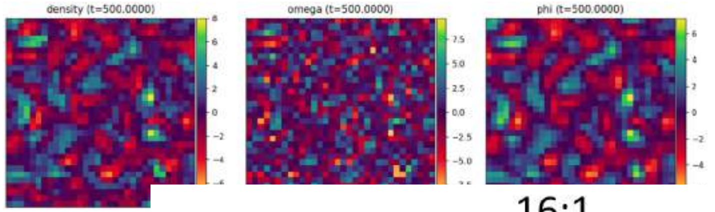
- **Preserved visual dynamics***




***within 1$\sigma$ of mean**

# What do we gain?

- **Preserved visual dynamics***
- **Preserved physical metrics***



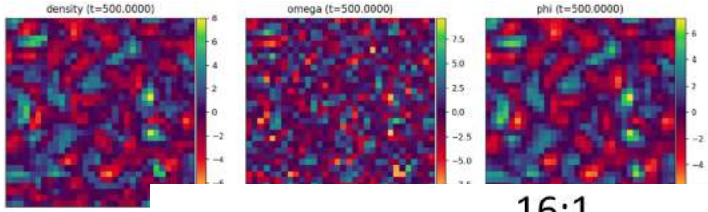

16:1

| | | |
|---|---|---|
| High Resolution | $\Gamma_n = 0.60 \pm 0.01$ | $\delta\Gamma_n = 0.05 \pm 0.004$ |
| Downsampled | $\Gamma_n = 0.57 \pm 0.01$ | $\delta\Gamma_n = 0.05 \pm 0.004$ |
| Greif et al. | $\Gamma_n = 0.58 \pm 0.01$ | $\delta\Gamma_n = 0.05 \pm 0.005$ |

***within $1\sigma$ of mean**

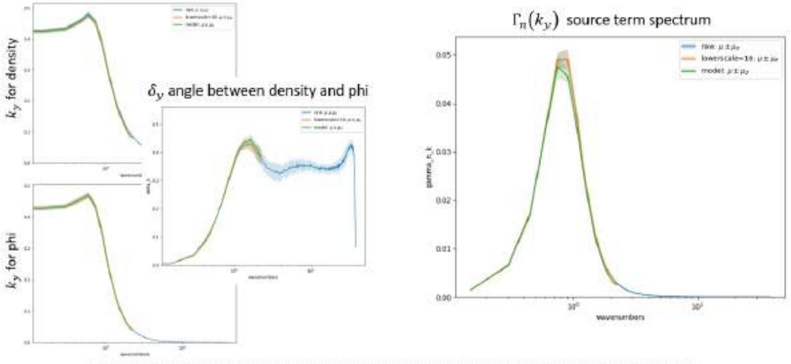
# What do we gain?

- **Preserved visual dynamics***
- **Preserved physical metrics***
- **Preserved spectral properties***




16:1

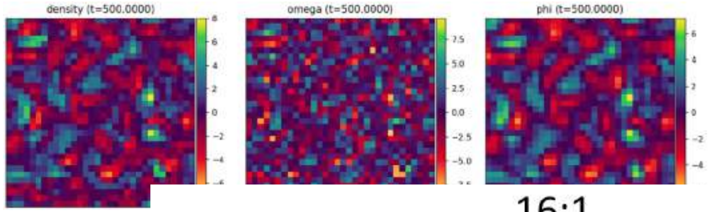
| | | |
|---|---|---|
| High Resolution | $\Gamma_n = 0.60 \pm 0.01$ | $\delta\Gamma_n = 0.05 \pm 0.004$ |
| Downsampled | $\Gamma_n = 0.57 \pm 0.01$ | $\delta\Gamma_n = 0.05 \pm 0.004$ |
| Greif et al. | $\Gamma_n = 0.58 \pm 0.01$ | $\delta\Gamma_n = 0.05 \pm 0.005$ |




**$*$within $1\sigma$ of mean**

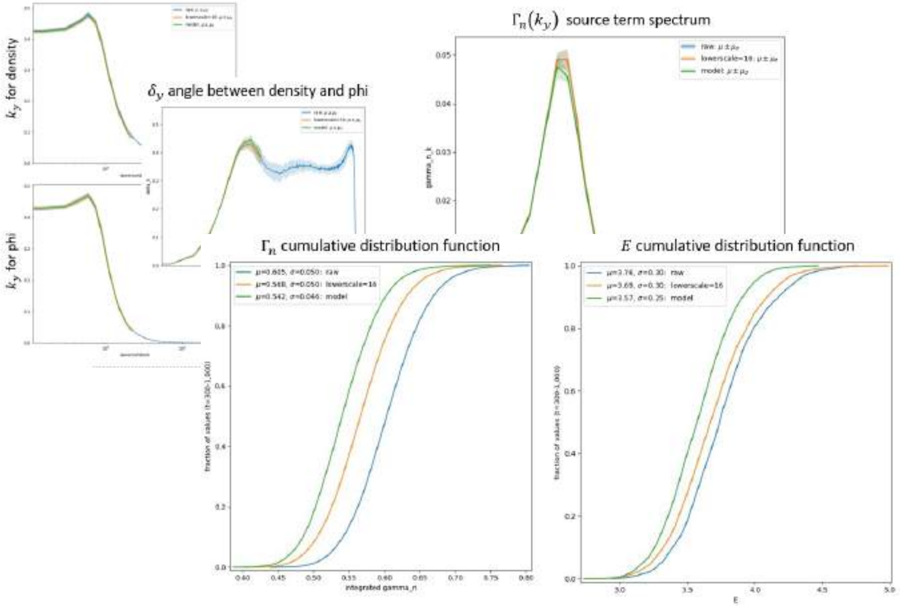
# What do we gain?

- **Preserved visual dynamics***
- **Preserved physical metrics***
- **Preserved spectral properties***
- **Preserved statistical distributions***




16:1

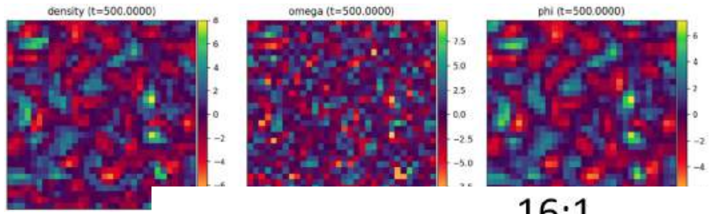
| | | |
|---|---|---|
| High Resolution | $\Gamma_n = 0.60 \pm 0.01$ | $\delta\Gamma_n = 0.05 \pm 0.004$ |
| Downsampled | $\Gamma_n = 0.57 \pm 0.01$ | $\delta\Gamma_n = 0.05 \pm 0.004$ |
| Greif et al. | $\Gamma_n = 0.58 \pm 0.01$ | $\delta\Gamma_n = 0.05 \pm 0.005$ |




***within 1$\sigma$ of mean**

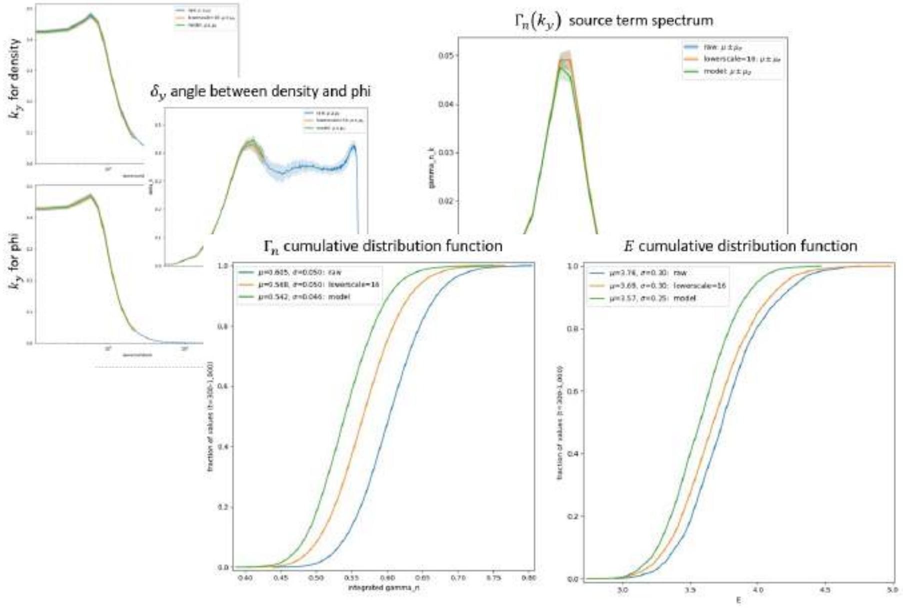
# What do we gain?

- **Preserved visual dynamics***
- **Preserved physical metrics***
- **Preserved spectral properties***
- **Preserved statistical distributions***
- **Speedup:**
  - Network scales O(n)
  - Downsampling in $x$ and $y$, 16x each: 256x
  - Using fewer gradients, RK4 → $Euler$: 4x
  - Increasing time step compared to RK4: 5x
  - Up to ~5,000x faster in theory
  - Speedup of ~600x in practice (70s, down from ~12h)




16:1

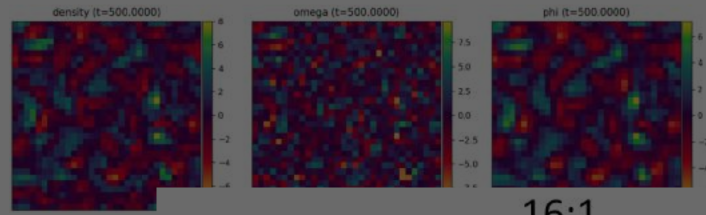
| | | |
|---|---|---|
| High Resolution | $\Gamma_n = 0.60 \pm 0.01$ | $\delta\Gamma_n = 0.05 \pm 0.004$ |
| Downsampled | $\Gamma_n = 0.57 \pm 0.01$ | $\delta\Gamma_n = 0.05 \pm 0.004$ |
| Greif et al. | $\Gamma_n = 0.58 \pm 0.01$ | $\delta\Gamma_n = 0.05 \pm 0.005$ |




***within 1$\sigma$ of mean**

# What do we gain?

- **Preserved visual dynamics***
- **Preserved physical metrics***
- **Preserved spe**
- **Preserved stat**
- **Speedup:**
  - Network scal
  - Downsampli
  - Using fewer
  - Increasing ti
  - Up to ~5,00
  - Speedup of ~600x in practice (70s, down from ~12h)

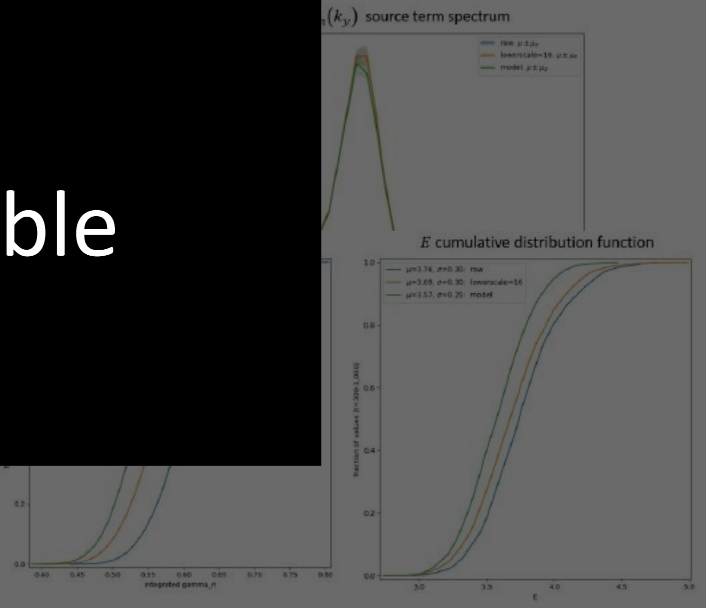



16:1

$\pm 0.004$

$\pm 0.004$

$\pm 0.005$




Up to 5,000x faster

Statistically indistinguishable

***within 1$\sigma$ of mean**