DeepMind

Learning general-purpose CNN-based simulators for astrophysical turbulence

Alvaro Sanchez-Gonzalez*,[1] Kimberly Stachenfeld*,[1] Drummond Fielding,[2] Dmitrii Kochkov,[3] Miles Cranmer,[4] Tobias Pfaff,[1] Jonathan Godwin,[1] Can Cui,[2] Shirley Ho,[2] Peter Battaglia[1]

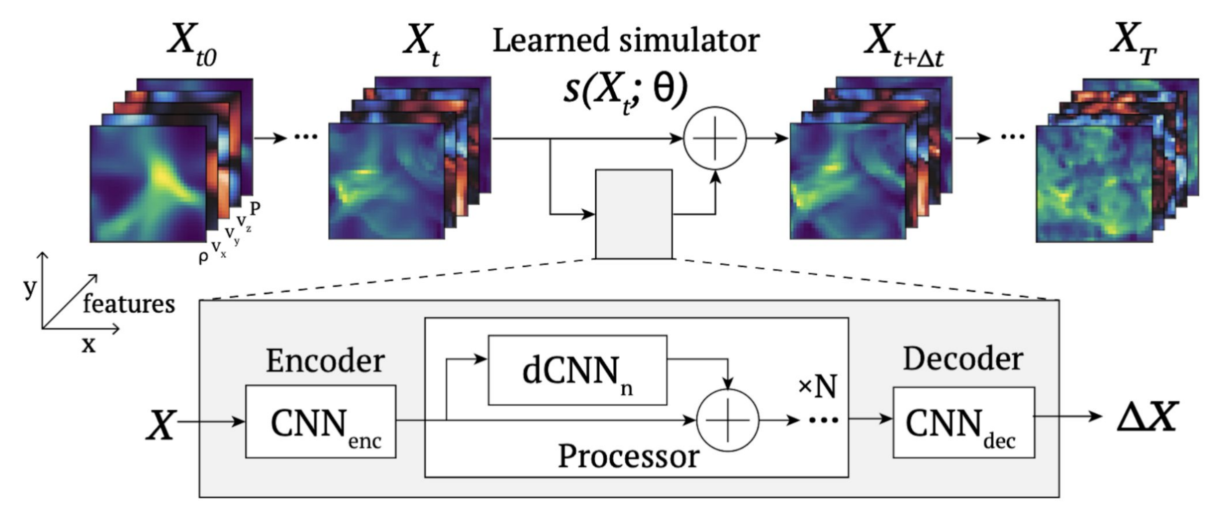
[1] DeepMind
[2] Flatiron Institute
[3] Google Research
[4] Princeton University

Introduction

- Turbulence simulation is important to understanding astrophysical phenomena like galaxy formation.
- Numerical solvers (like Athena++) are computationally expensive at high resolution, and very inaccurate at low resolution.
- **Question**: To what extent can learned models supplement or replace traditional simulators for astrophysics?
- **Highlights:**
  - Domain-general, fully-learned convolutional models for simulation
  - Same architecture can learn to predict different types of turbulence
  - Comparisons to coarsened numerical solver in terms of spatial and temporal resolution, numerical stability, and generalization performance

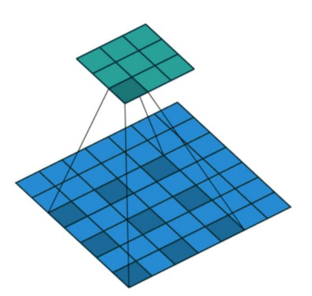
Model

We train a general-purpose model to learn the transition function between pairs of states for 1D, 2D, or 3D grids. At test time, we apply the model multiple times from an initial state.

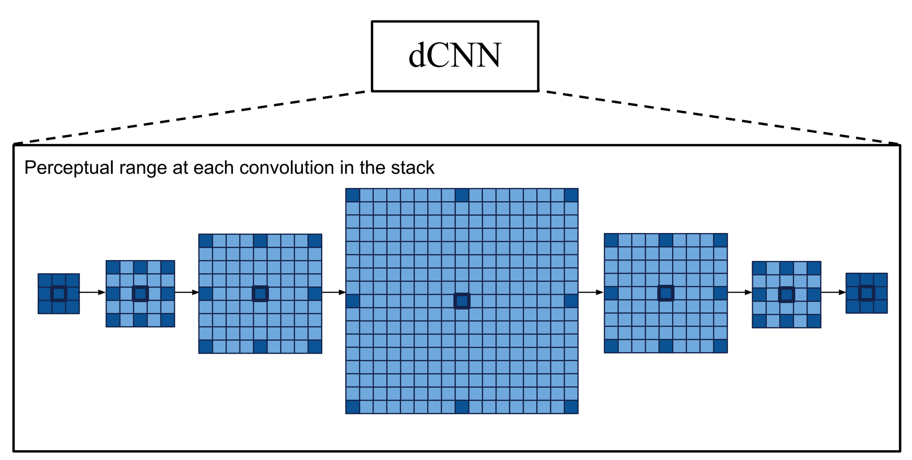


Dilated Convolution
Yu & Koltun (2015)

Perceptual range of a single dilated convolution with a dilation rate of 2



U-shaped stack
See also: Ronneberger et al. (2015)

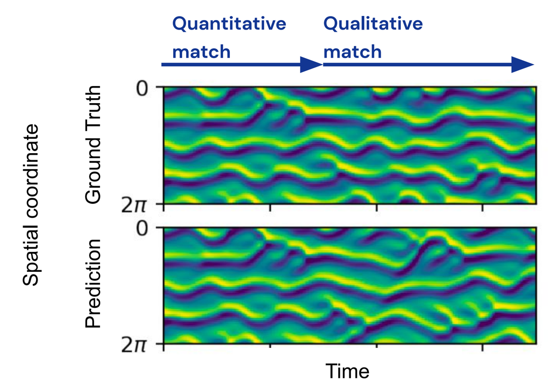


dCNN block: stack of 7 dilated CNNs to gradually increase and decrease perceptual range. Our model uses N=4 of these stacks.

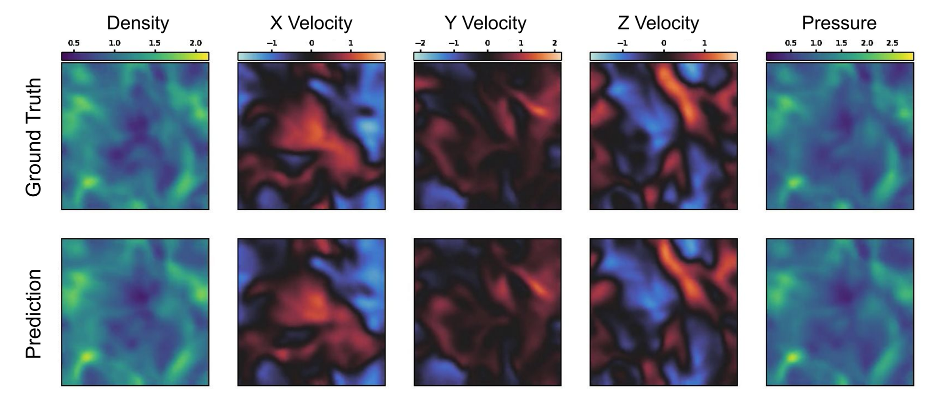
Datasets and domain generality

One model → Accurate performance in 4 different domains

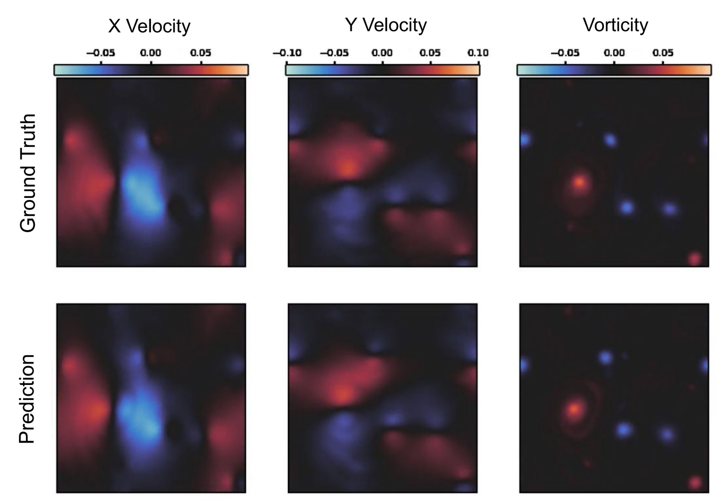
1D Kuramoto-Sivashinsky (KS) Equation
Chaotic, unstable, nonlinear dynamics



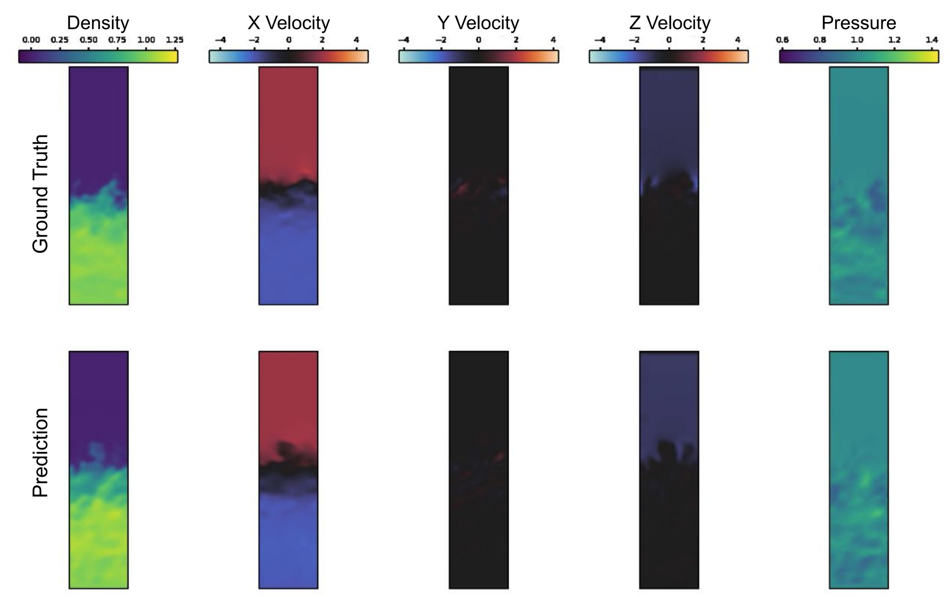
3D Uniform Compressible Decaying Turbulence
Relevant to planet, star, black hole, and galaxy formation



2D Incompressible Turbulence
Describes atmospheric flows on planets



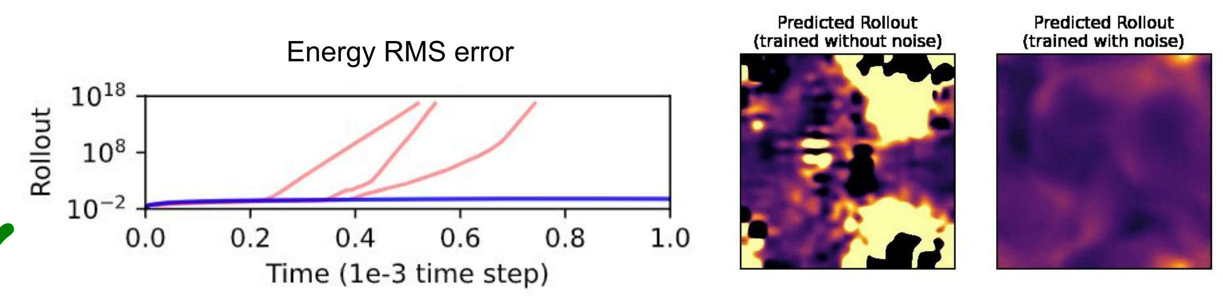
3D Mixing Layer Turbulence with Radiative Cooling
Captures cooling during galaxy formation



Stability

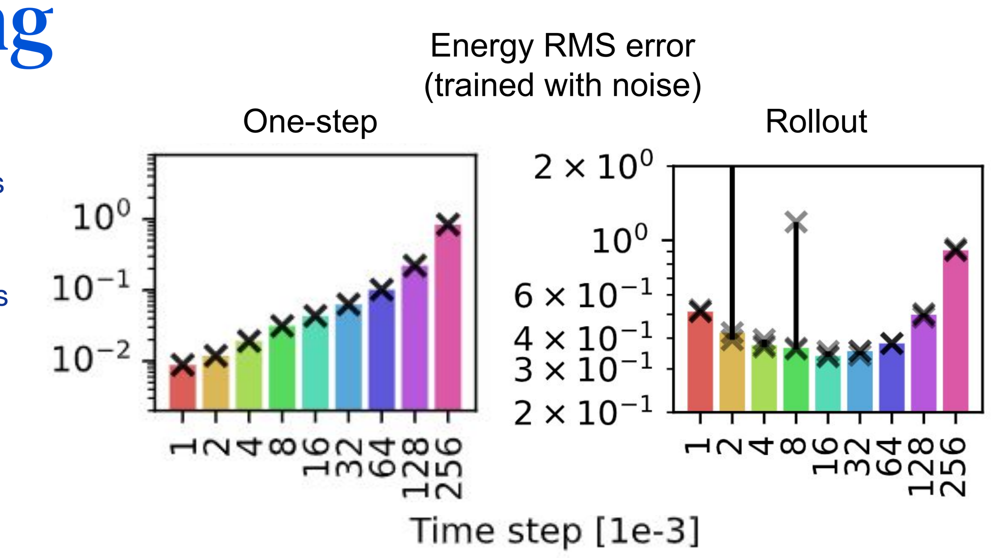
Models **trained without noise** can be unstable

Using **training noise** helps fixing stability ✓



Temporal coarsening

Unlike the ground truth solver, the learned model is **not very sensitive to** the specific integration **timestep**, and can operate well on many timesteps ✓



Running time

- Athena++
  - Scales O(resolution[4])
  - CPU only
- Learned model:
  - **Up to 1000x faster** than Athena at 128

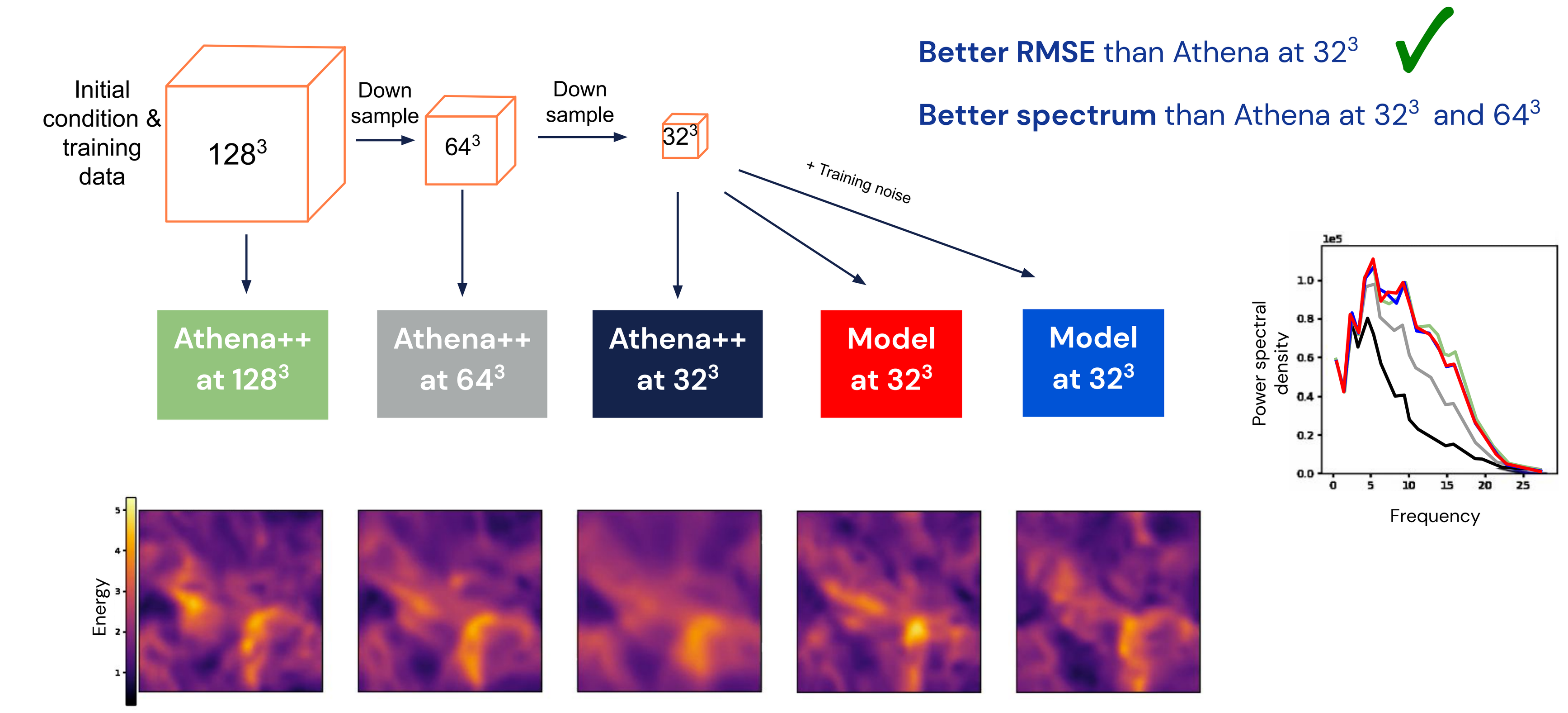
Simulator	Time (s)
Athena++ 32^3	~4
Athena++ 64^3	~60
Athena++ 128^3	~1000
Model $128^3 \rightarrow 32^3$	~20-30
Model $128^3 \rightarrow 32^3$ (GPU)	~1

Spatial coarsening

The learned model can operate at coarser discretizations of the problem **more effectively than the ground truth solver**

Better RMSE than Athena at 32^3 ✓

Better spectrum than Athena at 32^3 and 64^3 ✓



Constraint preservation

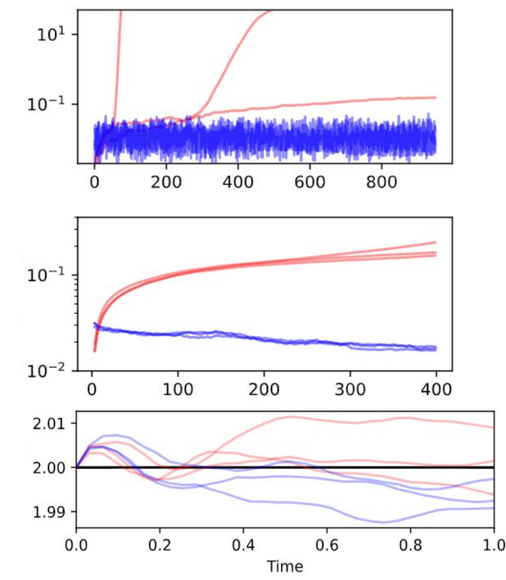
Training without noise does not preserve constraints

Training with noise helps....sometimes..

✓ 1D KS total momentum (should stay at 0)

✓ 2D Incompressible velocity field divergence (should be 0)

✗ 3D turbulence (should stay at 2)



Generalization to different states

Generalization to longer trajectories:

✗ Does not generalize to **more developed turbulence**

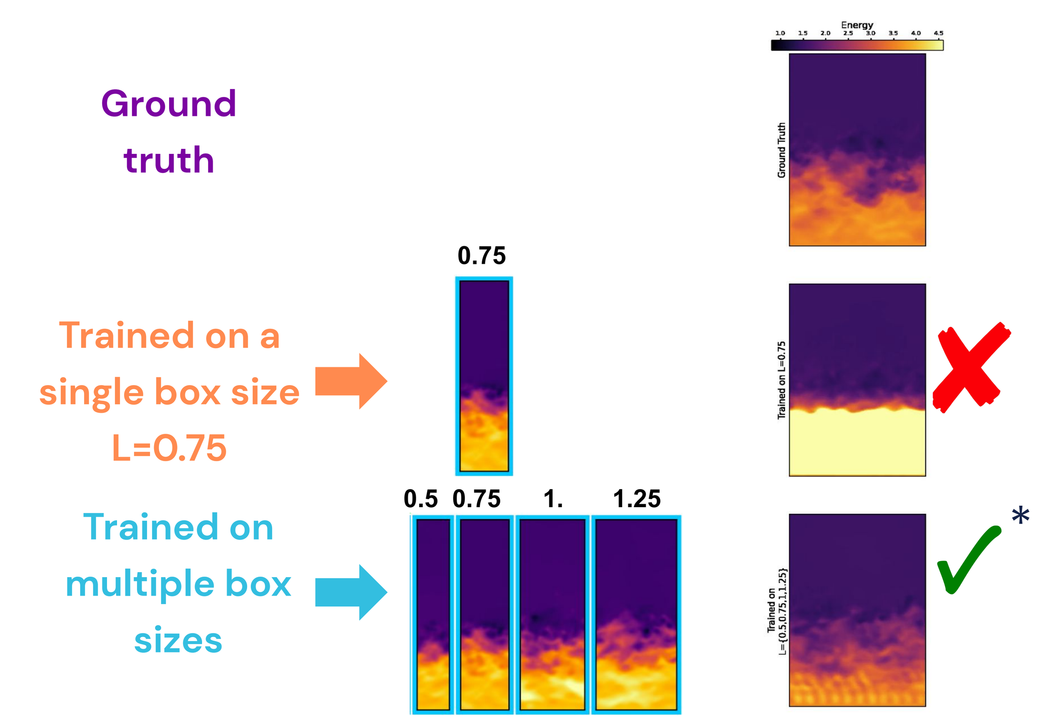
Generalization to different initial conditions:

✓ Generalizes to **higher solenoidal components**

✗ Fails to generalize to **higher compressive components**

Generalization to larger boxes

The model requires training on multiple box sizes to be able to produce plausible predictions for larger box sizes



When extrapolating to a box size larger or smaller than those seen during training, the **model trained on multiple box sizes does not predict the expected cooling velocity**, which is a function of the box size.