

WOUDSCHOTEN CONFERENCE · 2 OCTOBER 2026

What a coarse grid cannot know

Learned turbulence closures, and how AI agents help us find out

Syver Døving Agdestein · CWI Amsterdam · with Benjamin Sanderse

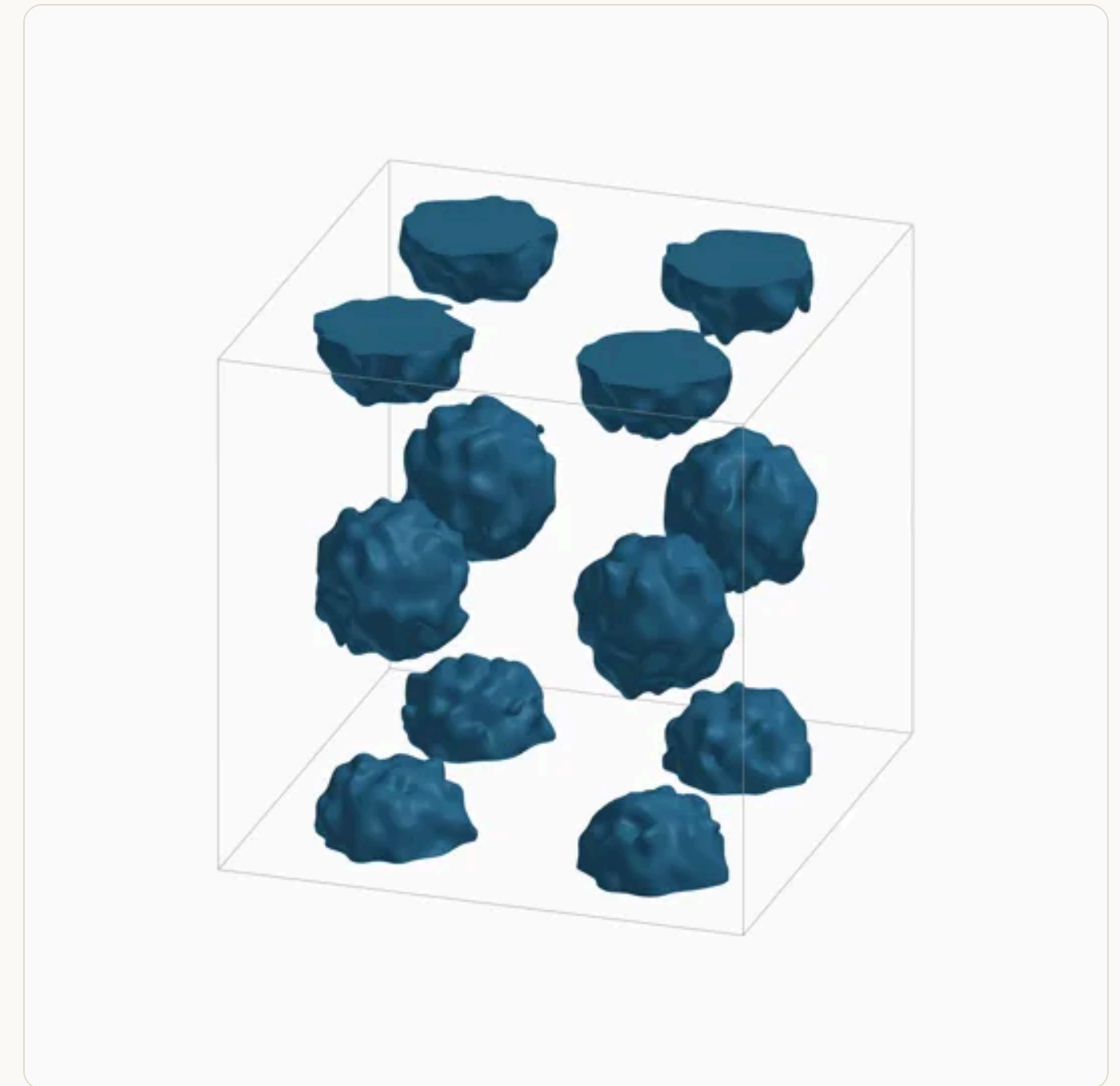
Supported by NWO (VI.Vidi.193.105, OCENW.XS25.1.149)

Compute on Snellius: SURF (EINF-15798)

THE QUESTION

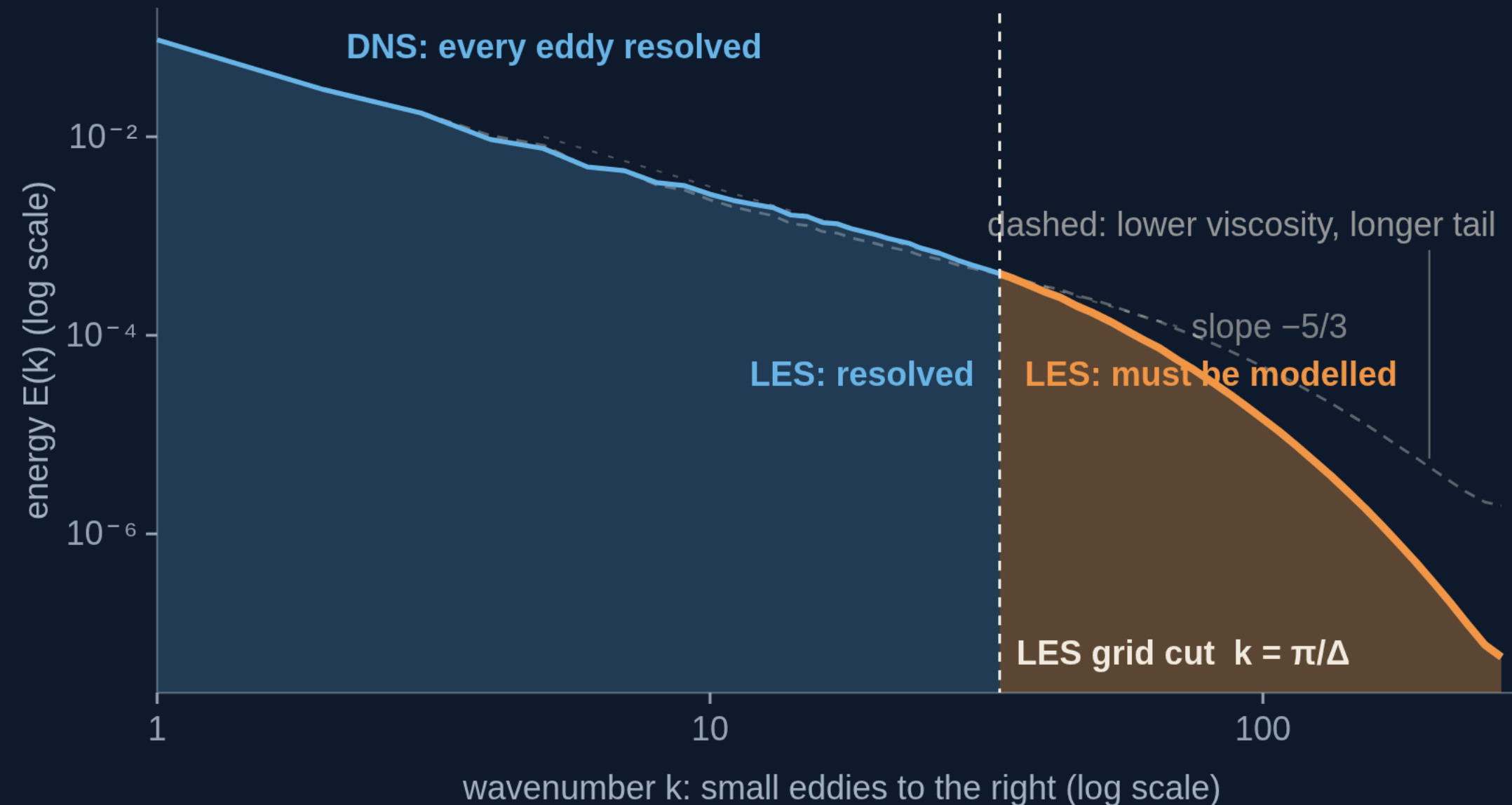
How much turbulence can a coarse grid know about?

A simulation on a coarse grid sees the flow only through that grid. What does that view determine, what can it not determine, and what can we do about the rest?



Taylor–Green vortex at $Re = 3000$ becoming turbulent (vortex structures)

Turbulence has too many scales



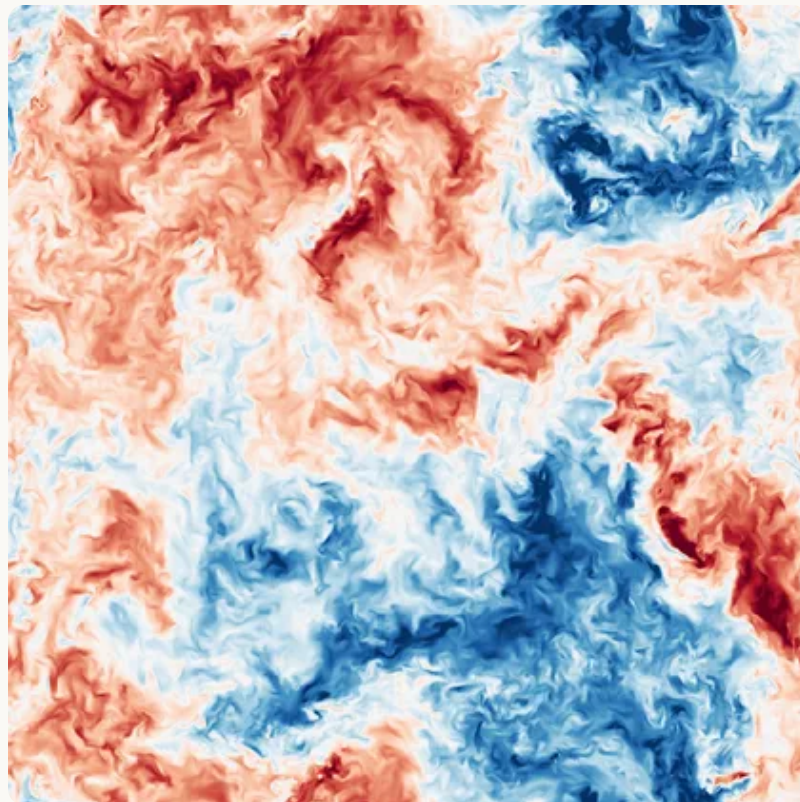
Reynolds number Re : how turbulent a flow is

$$N_{\text{grid}} \propto Re^{9/4}$$

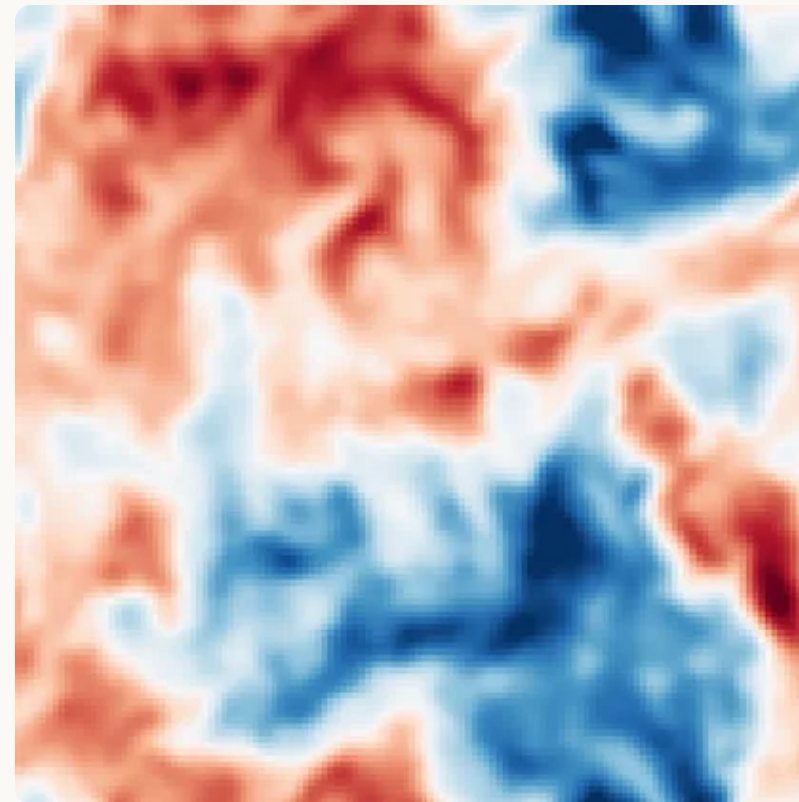
grid points to resolve every eddy: direct numerical simulation (DNS)

Large-eddy simulation (LES): a coarse grid, plus a model for what lies beyond the cut.

Coarse equations are not closed



DNS grid, 810^3 : every eddy



LES grid, 128^3 : the filtered flow $\bar{u}$

$$\partial_t \bar{u} + \nabla \cdot (\bar{u} \otimes \bar{u}) + \nabla \bar{p} - \nu \nabla^2 \bar{u} = -\nabla \cdot \tau$$

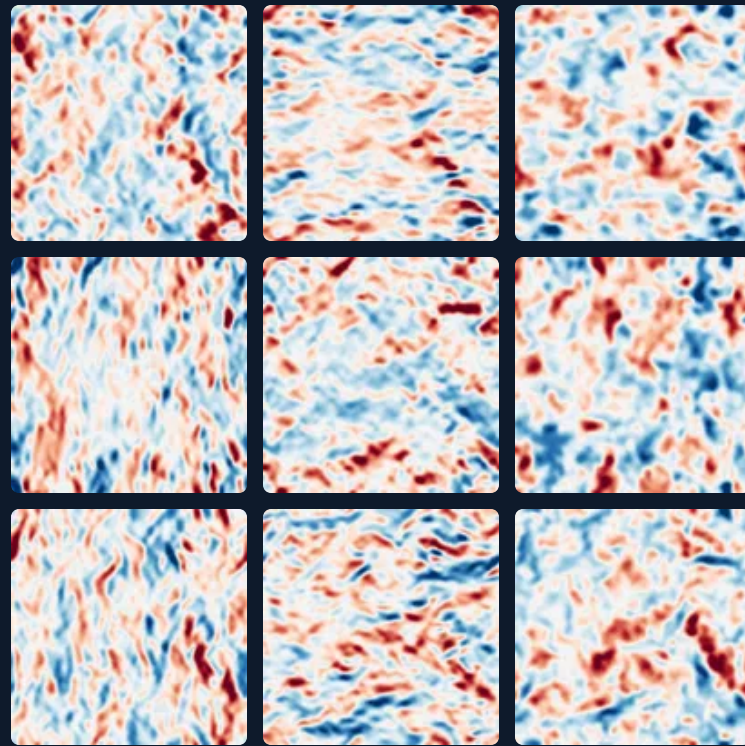
$$\tau = \overline{u \otimes u} - \bar{u} \otimes \bar{u}$$

The subfilter stress τ needs the small scales, which the coarse grid does not have.

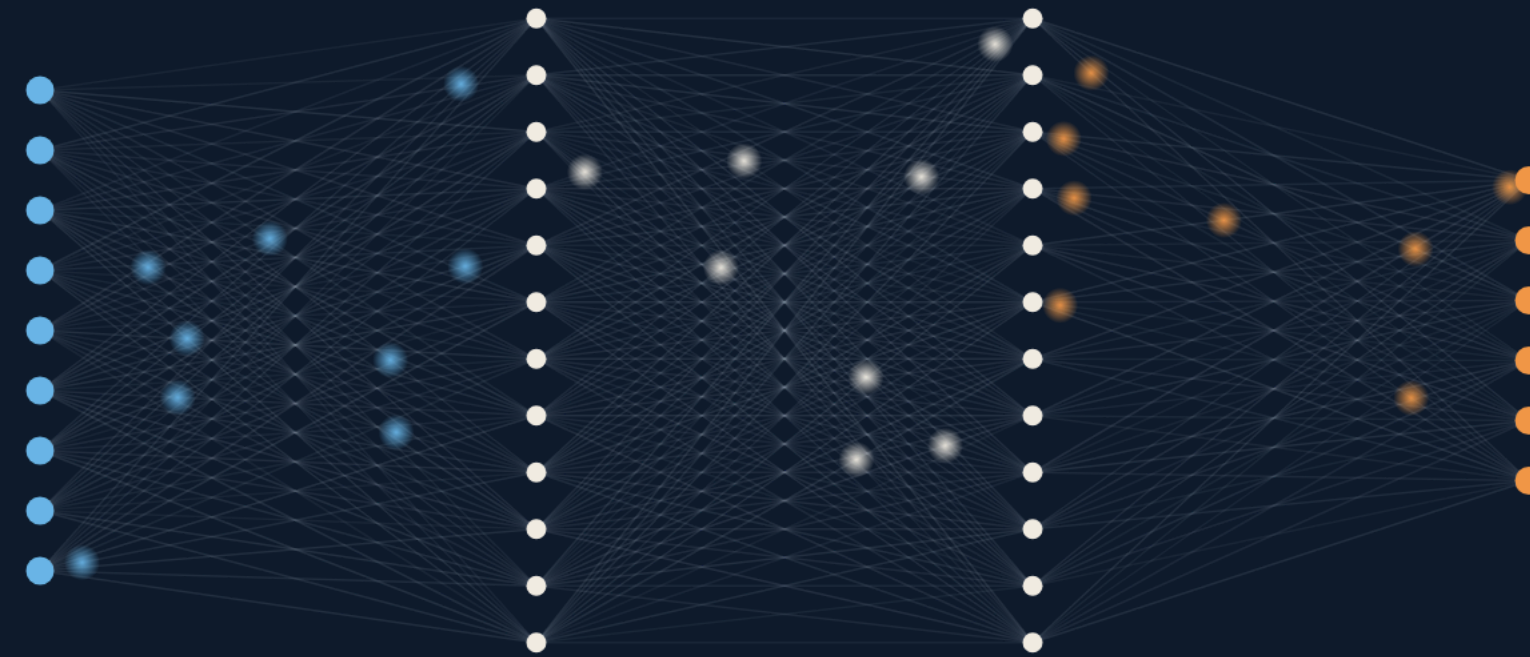
A closure model must supply τ from $\bar{u}$ alone.

II · LEARNED CLOSURES

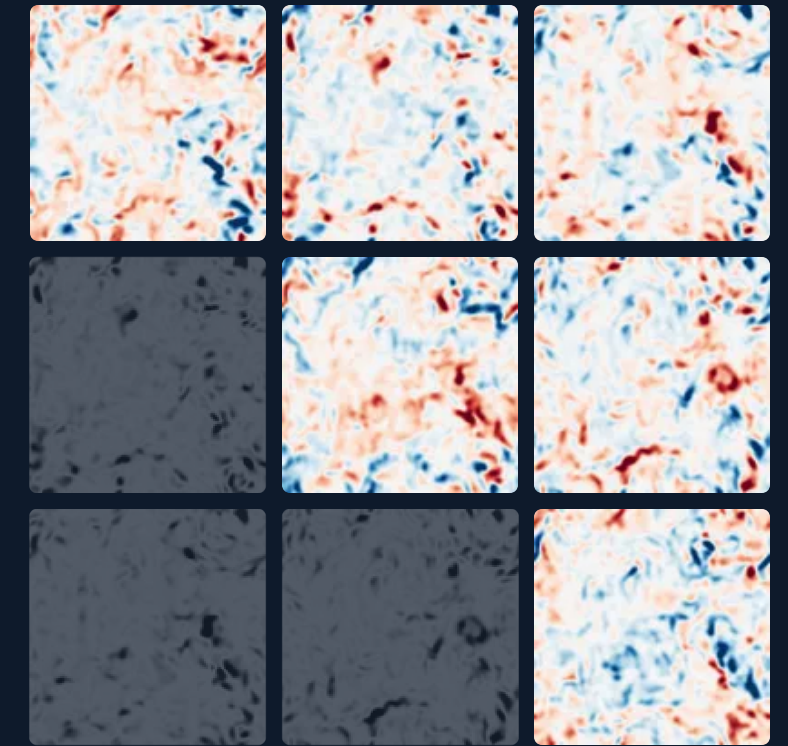
Learn the closure from simulation data



input: the resolved velocity
gradient $\bar{A}$, 9 components



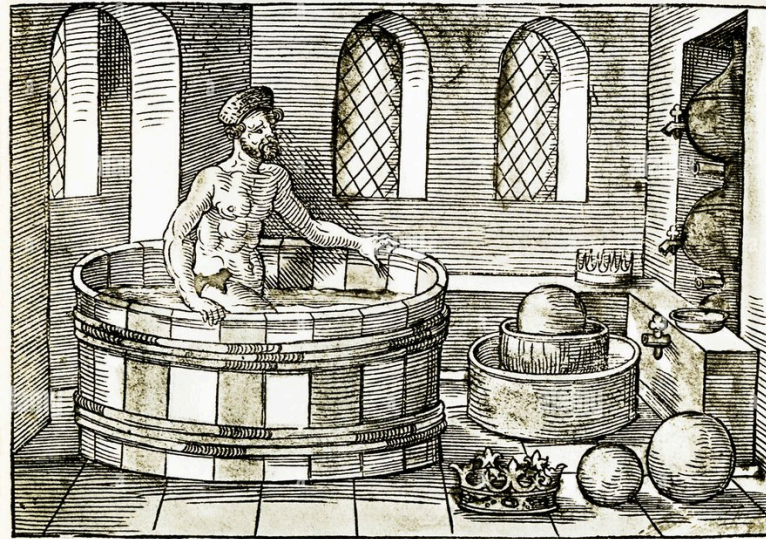
a small network, applied at every grid point



output: the stress τ , symmetric: 6
independent

$$\tau(x) \approx \mathcal{M}_\theta(\bar{A}(x)), \quad \bar{A} = \nabla \bar{u}$$

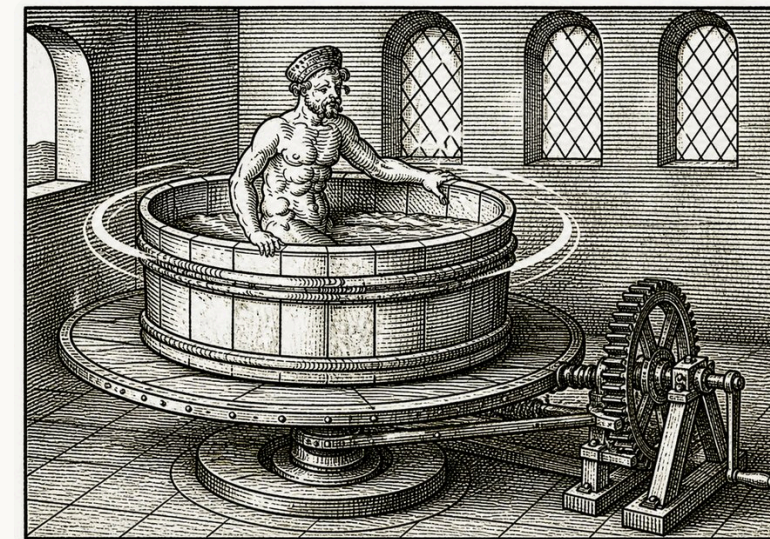
Symmetry: one flow, many views



the flow in the tub



move it at constant speed:
nothing changes



rotate it: the flow rotates along



mirror it: the flow is mirrored

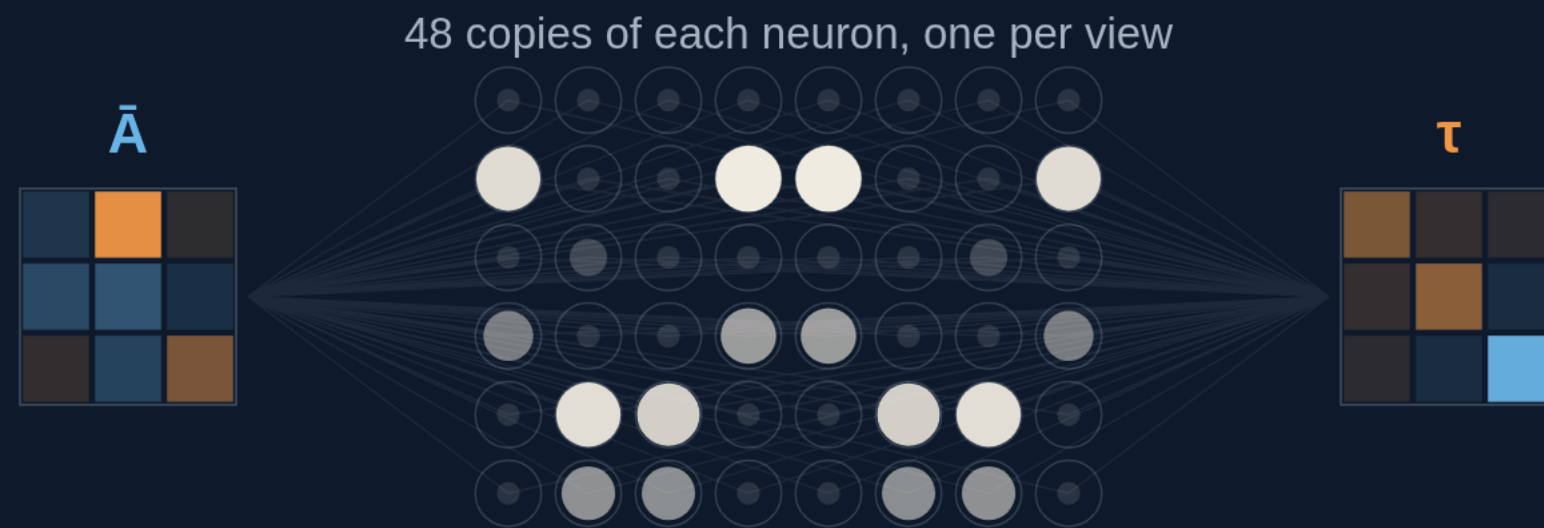
$$\mathcal{M}(Q\bar{A}Q^{\top}) = Q\mathcal{M}(\bar{A})Q^{\top} \quad \text{for all } Q \in \mathcal{O}_h$$

Build it in: share weights over the 48 (G-CNN), or use invariants and a tensor basis (TBNN).

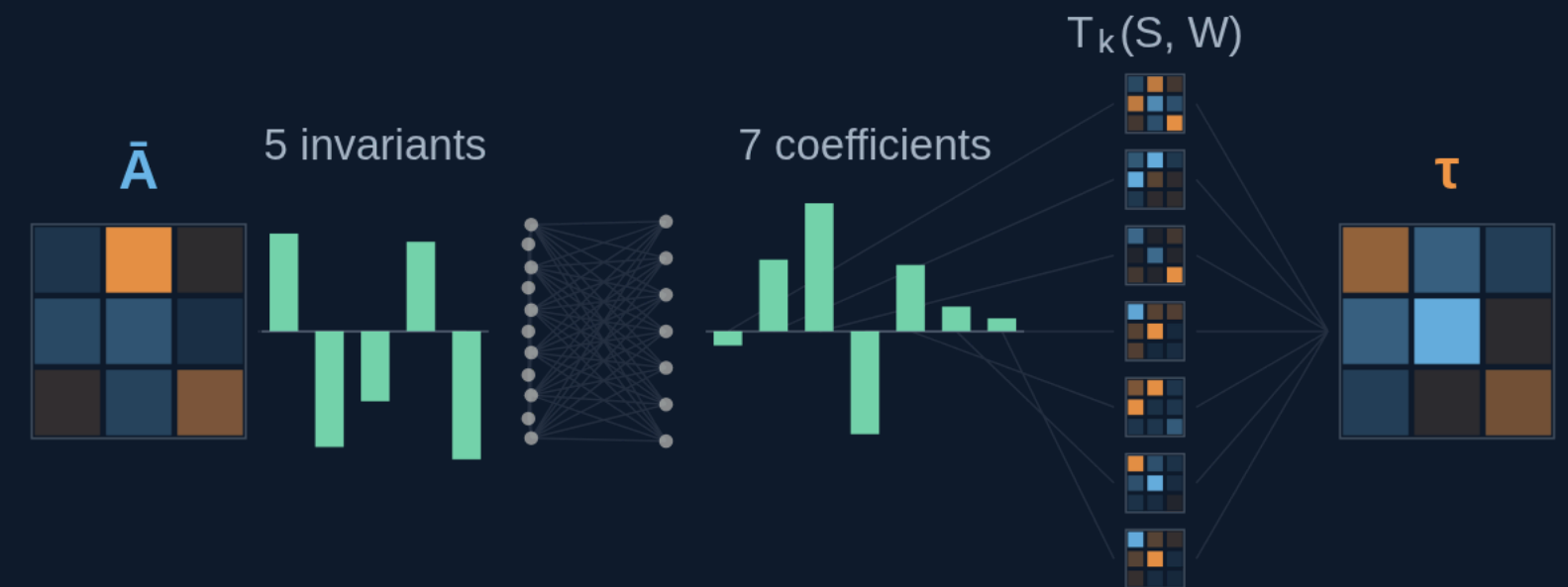
A cube grid keeps 48 rotations and reflections. A closure should respect all of them.

Two ways to build the symmetry in

G-CNN · share the weights



TBNN · invariants in, a tensor basis out



$$\zeta(g) = \varphi\left(\sum_{h \in \mathcal{O}_h} k(h^{-1}g) \xi(h) + b\right)$$

48 copies of every neuron share one kernel k . Turning the input only moves the copies around, so the output turns with it.

$$\mathcal{M}(\bar{A}) = \Delta^2 |\bar{A}|^2 \sum_{k=1}^7 \alpha_k(\lambda_1, \dots, \lambda_5) T_k(S, W)$$

The network sees only five invariants, which no turn changes. Seven fixed tensors built from S and W carry the turn.

II · LEARNED CLOSURES

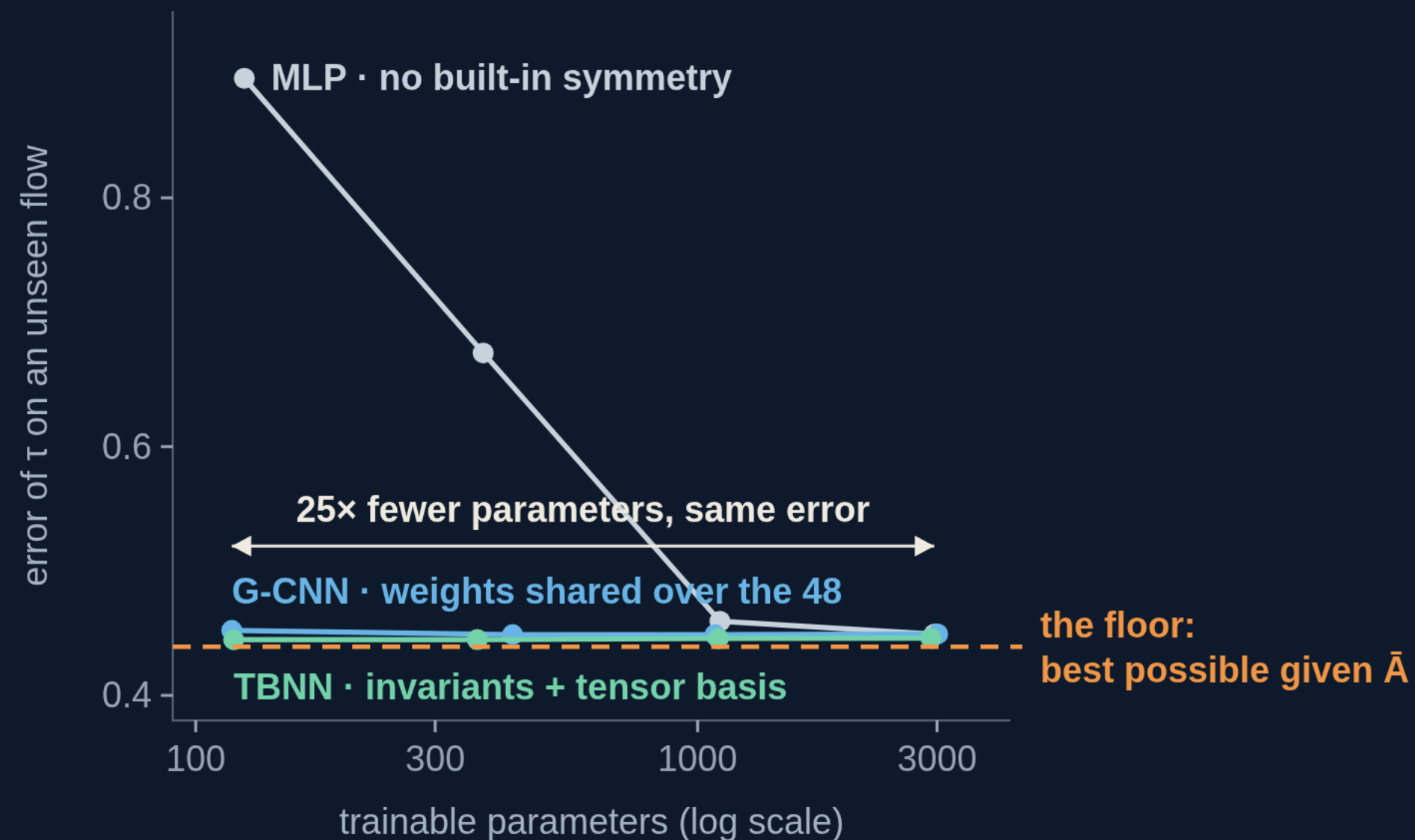
Every architecture hits the same floor

25×

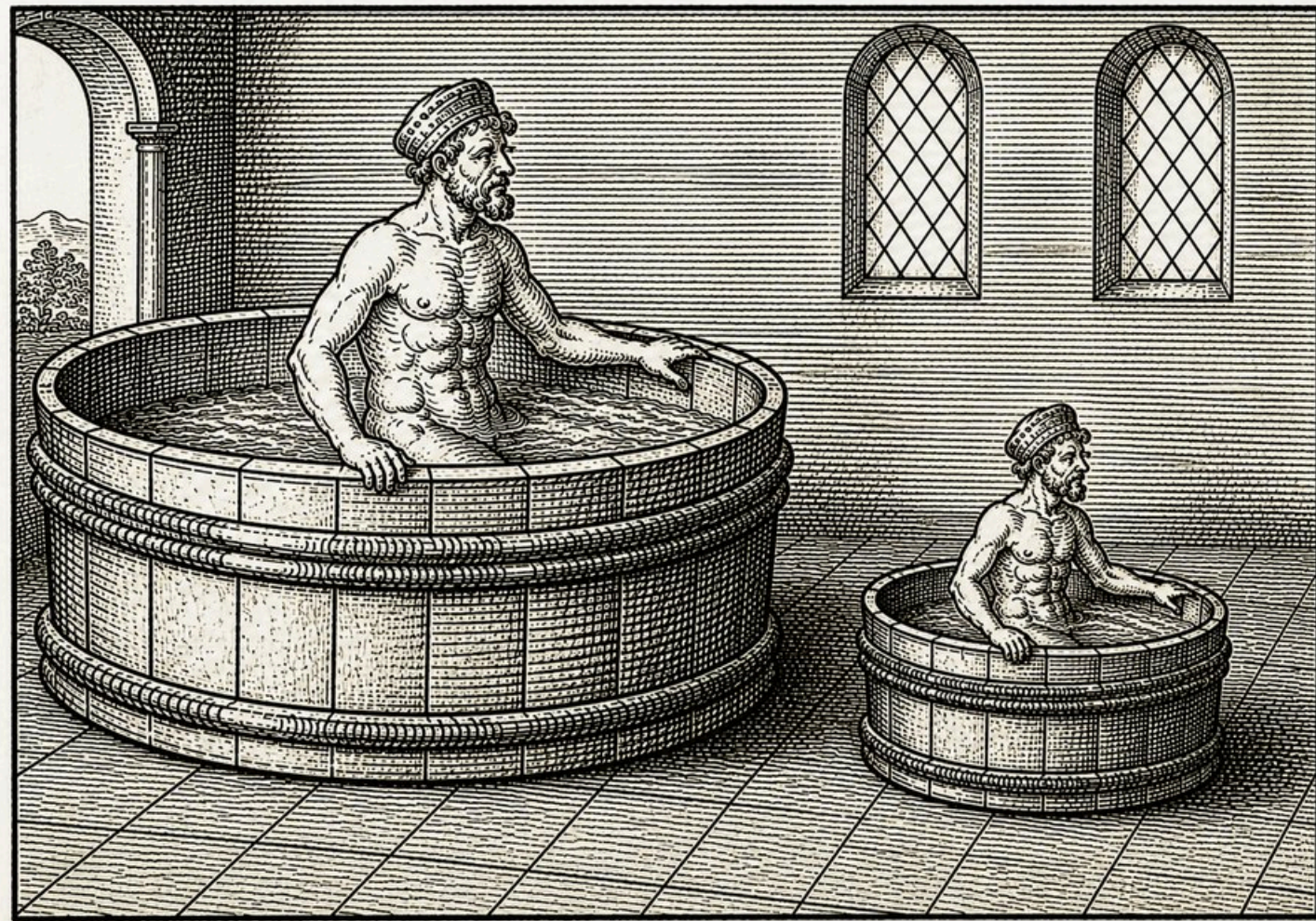
fewer parameters when the symmetry is built in

...but no network gets below the floor.

The floor: the error of the best possible function of $\bar{A}$, set by the input, not the architecture.



Blind to the Reynolds number



A closure that sees only $\bar{A}$ and Δ must, by dimensional analysis, take the form

$$\mathcal{M}(\bar{A}) = \Delta^2 |\bar{A}|^2 f(\bar{A}/|\bar{A}|)$$

so it cannot tell apart flows that differ only in viscosity.

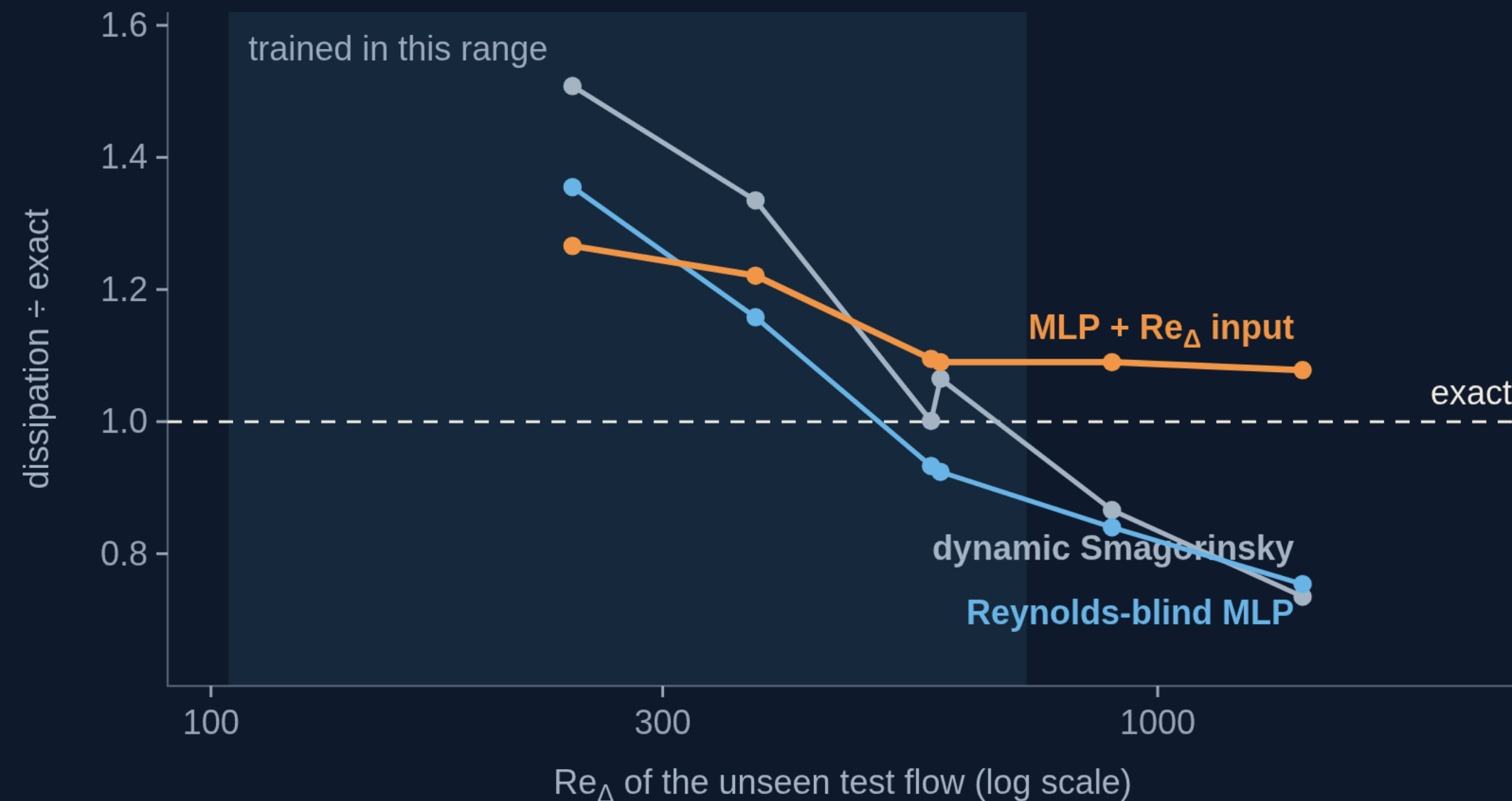
add the one dimensionless number the symmetry allows:

$$\mathcal{M}(\bar{A}) = \Delta^2 |\bar{A}|^2 f(\bar{A}/|\bar{A}|, \text{Re}_\Delta) \quad \text{Re}_\Delta = \frac{\Delta^2 \|\bar{A}\|}{\nu}$$

Symmetry hands the closure its missing input.

$$x \rightarrow ax, \quad t \rightarrow bt \quad \implies \quad \nu \rightarrow \frac{a^2}{b} \nu$$

Now the dissipation stays calibrated



Reynolds-blind: 1.36 → 0.75

drifts from too much to too little

With Re_Δ : 1.08–1.27

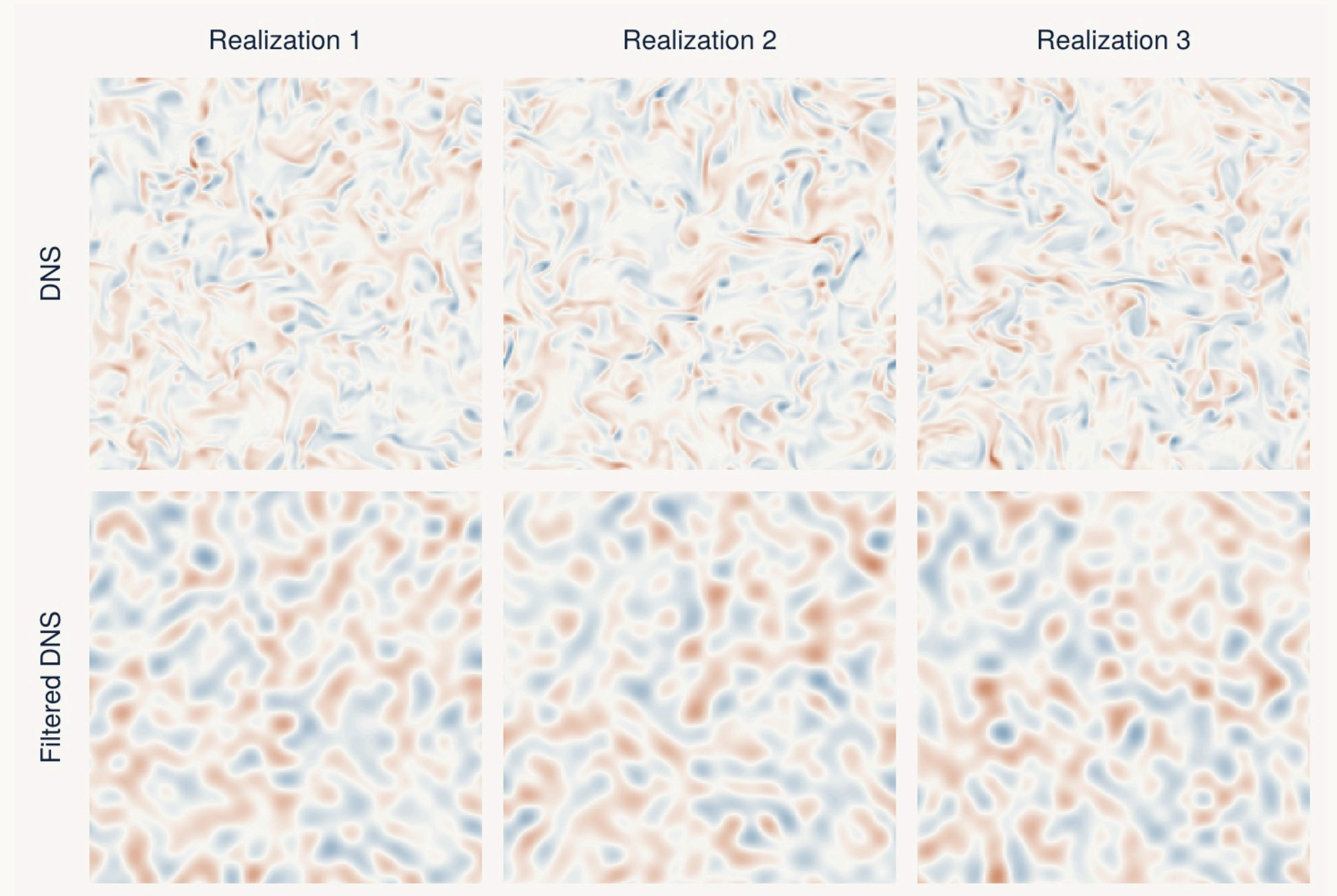
stays calibrated, also beyond the training range

III • BEYOND THE MEAN

One coarse state, many futures

Three simulations of 3D turbulence:
identical large scales, tiny differences in
the small eddies.

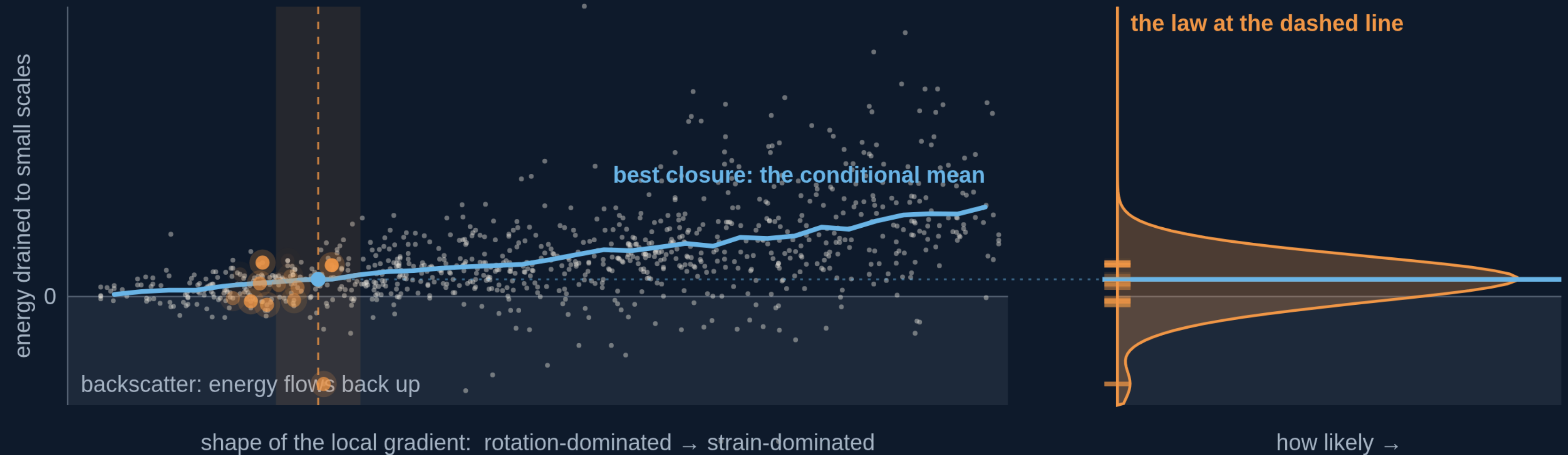
*Same resolved initial state.
Different futures.*



Taylor–Green vortex, $Re = 1600$, 256^3 grid. Top: vorticity. Bottom: filtered to what a coarse model sees.

III · BEYOND THE MEAN

The best closure is an average



$$\mathcal{M}^*(\bar{A}) = \mathbb{E}[\tau \mid \bar{A}] = \arg \min_{\mathcal{M}} \mathbb{E} \|\tau - \mathcal{M}(\bar{A})\|^2$$

Orange: the law of τ at the dashed line, and draws from it.

So measure the whole law

$$\mathbb{E}[|\tau|^2 \mid \bar{A}] = |\mathcal{M}^*(\bar{A})|^2 + \text{tr Cov}(\tau \mid \bar{A})$$

A mean closure keeps the blue term and drops the orange one: all the spread.

The object

$$p(\tau \mid \bar{A})$$

the law of the stress, given the resolved gradient

Made small by symmetry

$$5 \rightarrow 5$$

five invariants of $\bar{A}$ in, five stress components out

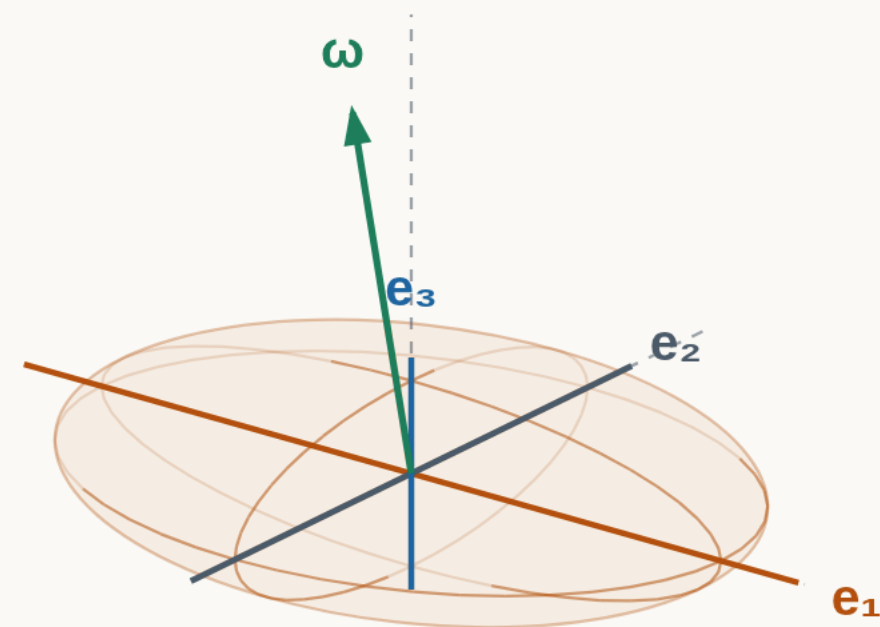
Measured on new data

13 runs

new simulations up to 1200^3 , scored on held-out data

Next: how much is left over, what shape it has, and whether it survives in a running LES.

Five numbers in, five out



$\bar{A}$ in its own axes

two strain rates + three vorticity

+0.62	-0.38	+0.17
+0.38	+0.15	+0.23
-0.17	-0.23	-0.77

$s_3 = -s_1 - s_2$; lower half = -upper

$$8 - 3 = 5$$

A traceless gradient has eight numbers. Three of them only say how it is turned: look along the strain's own axes and five remain.

Write τ in the same axes: its five components are the output.

So the law is a $5 \rightarrow 5$ conditional density.

Each axis is fixed only up to its sign: a vorticity convention picks one, and $\bar{A}$ and τ flip together.

III · BEYOND THE MEAN

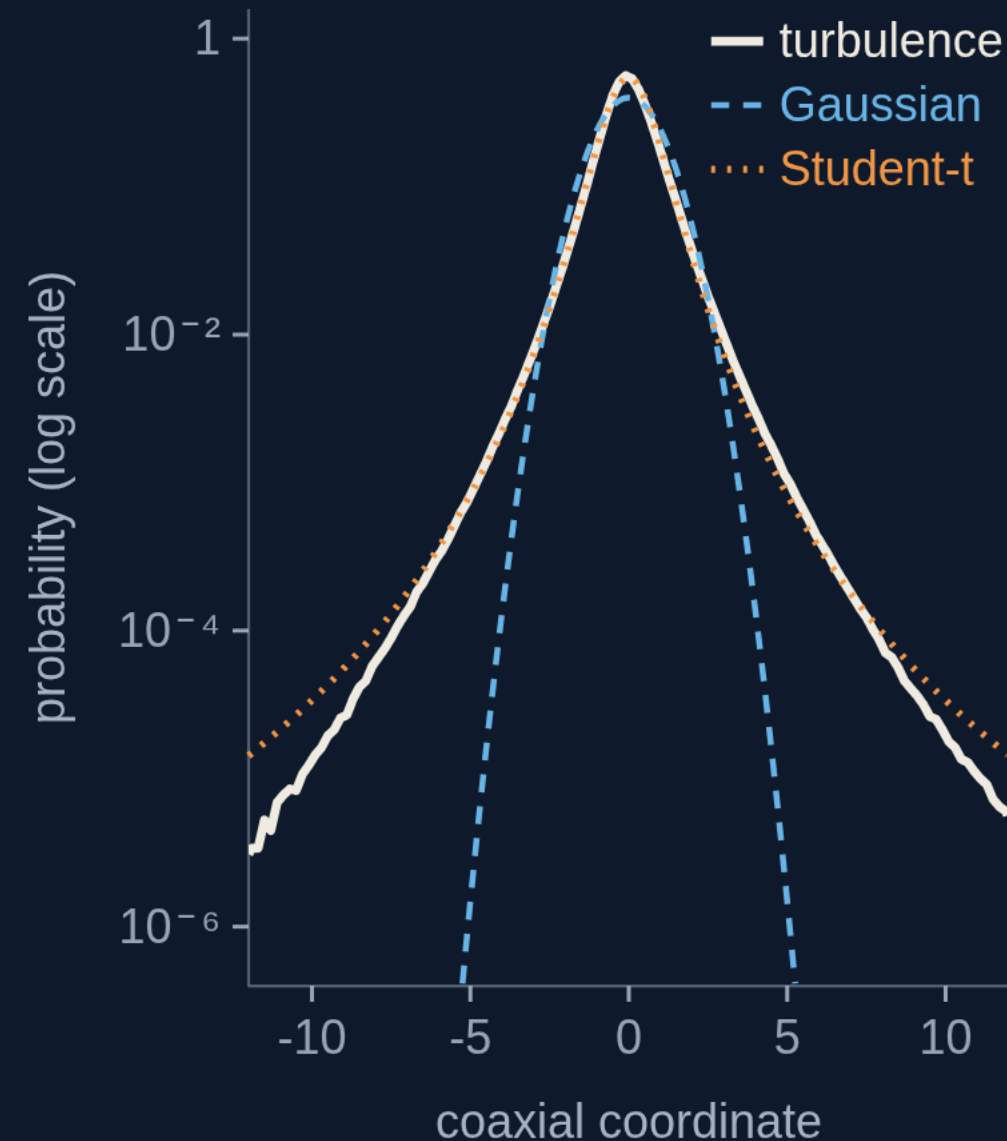
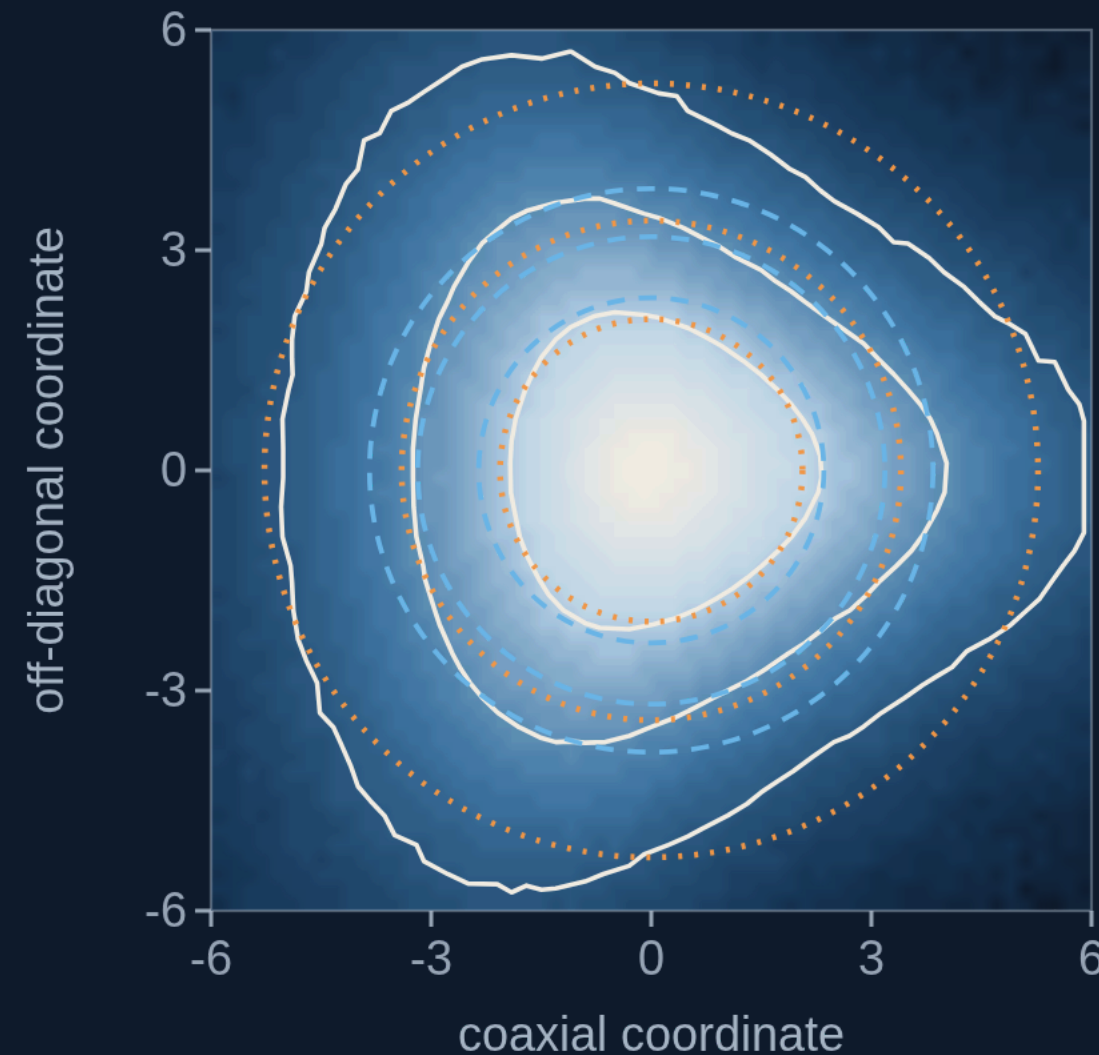
The mean explains most of the stress



That remainder is what a stochastic closure is for.

III · BEYOND THE MEAN

What remains is heavy-tailed



63–130

kurtosis, against 35 for a Gaussian

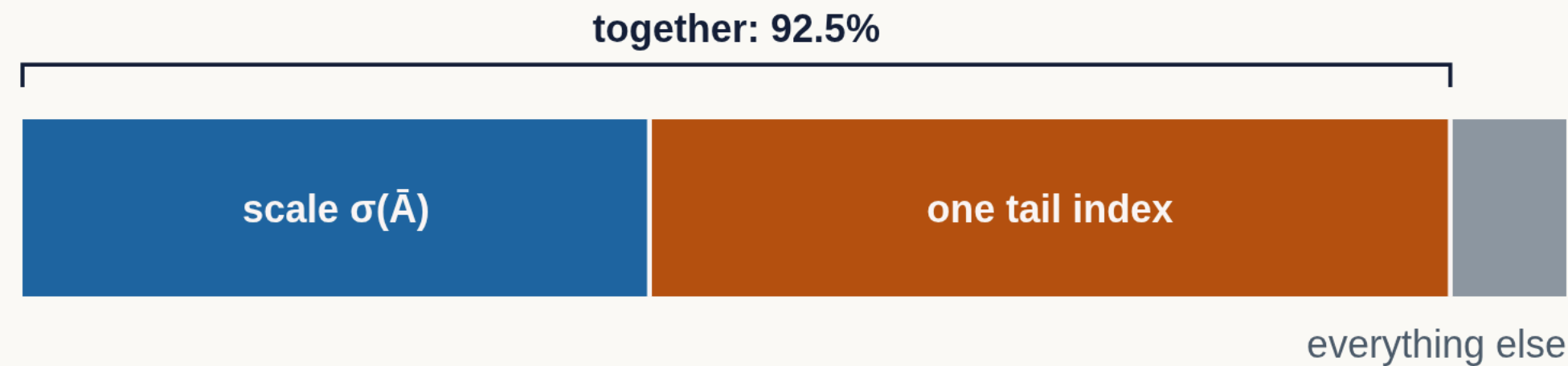
3.2–6.2

tail index of the Student-t fit; a Gaussian's is infinite

The long tail sits on the forward-transfer side: extreme stresses drain energy to the small scales.

III · BEYOND THE MEAN

...but two numbers carry most of it



● ○ ○ ○ ○ ○ ○ ○ ○ ○ ○
the eleven operating points (viscosity × filter width)

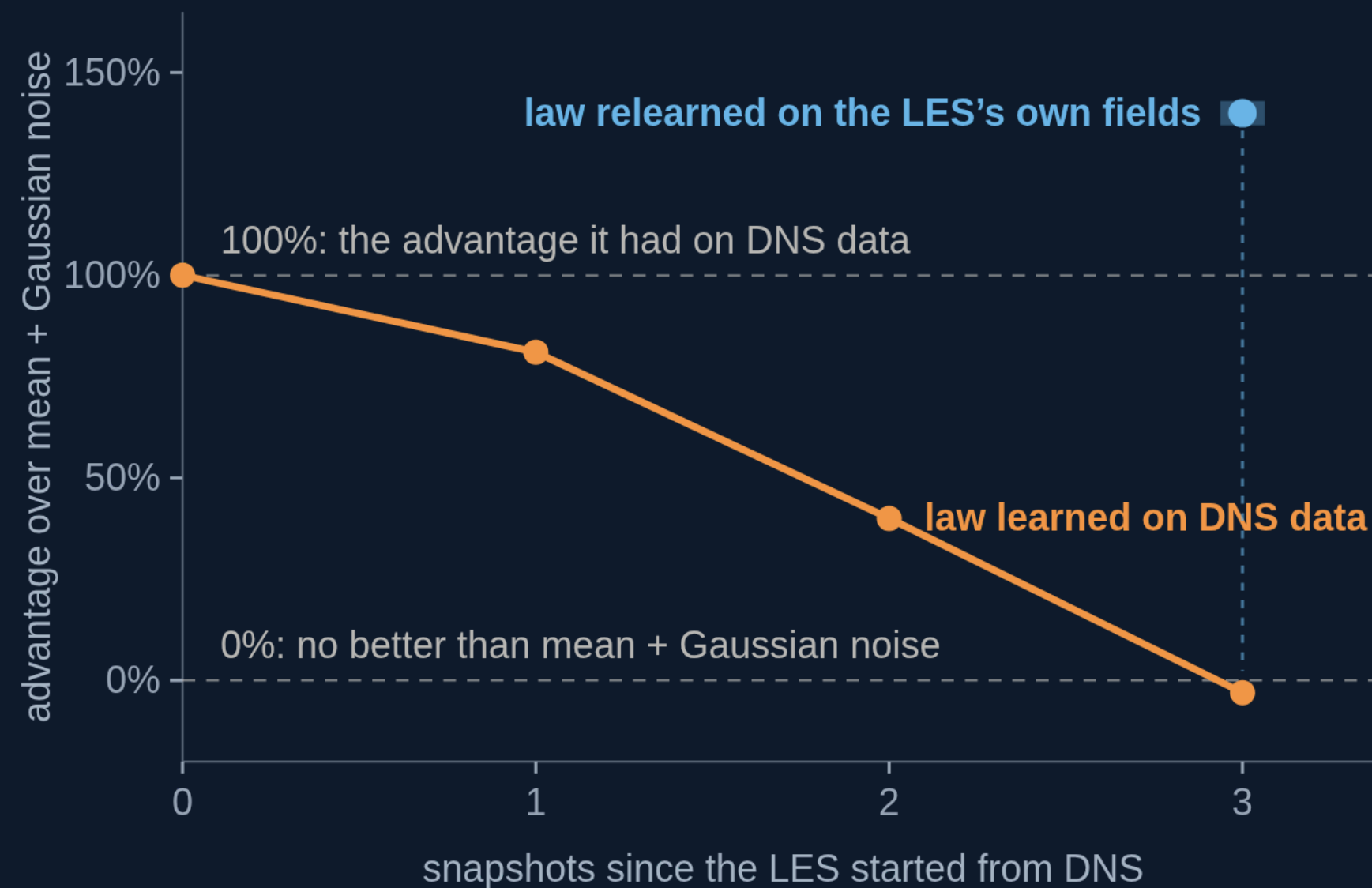
92–93%

of what the law gains over “mean plus constant Gaussian noise” comes from a scale $\sigma(\bar{A})$ and one tail index.

What the mean leaves out is low-order.

III · BEYOND THE MEAN

Learn the law where it is used



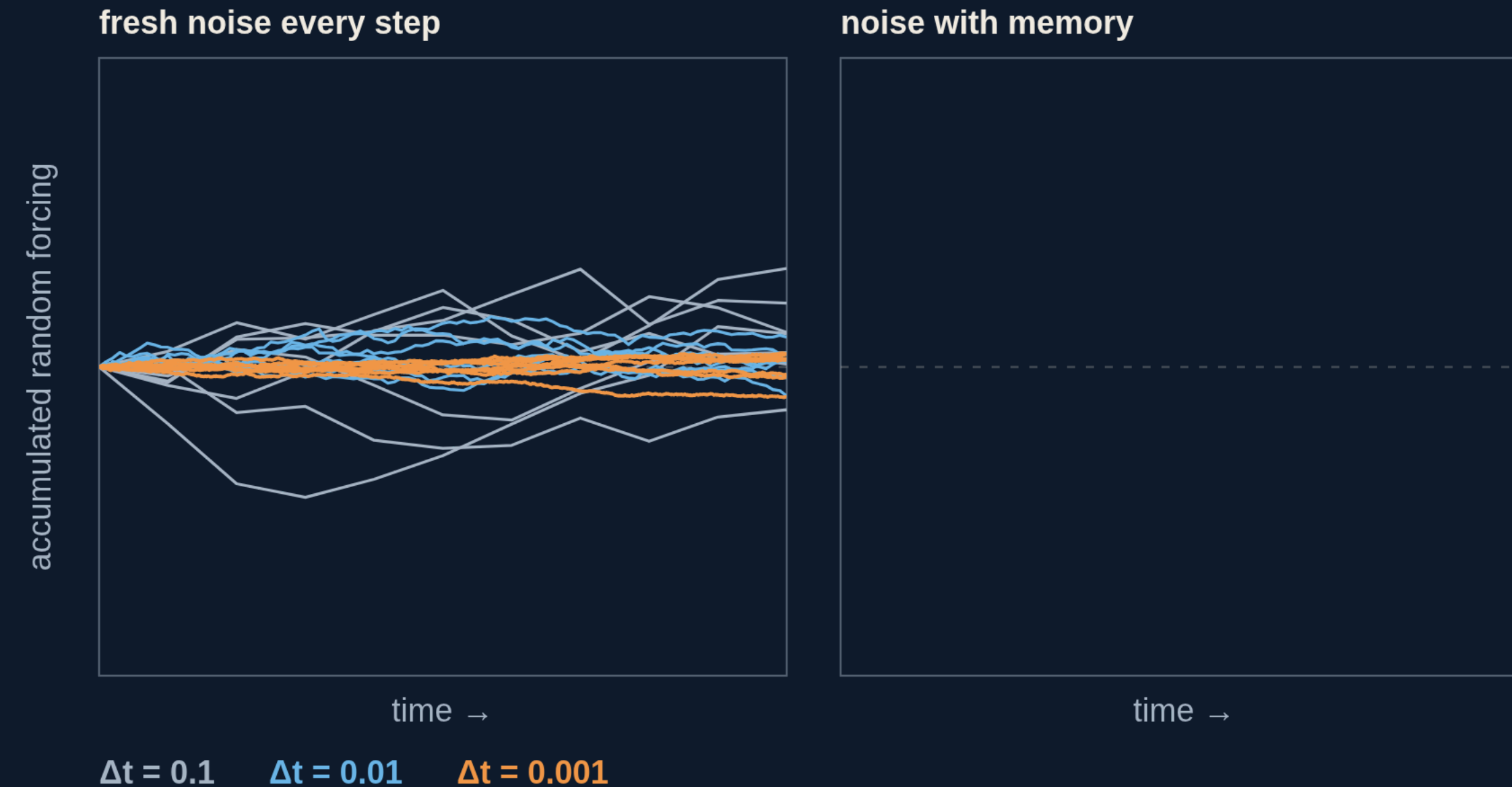
A law learned on DNS data loses its whole advantage over mean + Gaussian noise within three snapshots of a running LES.

Relearned on the LES's own fields, its advantage even grows: LES errors make the residual more heavy-tailed.

Above 100% means a bigger lead over the Gaussian reference, not a better prediction: absolute accuracy still falls.

III · BEYOND THE MEAN

A one-point law is not yet a closure



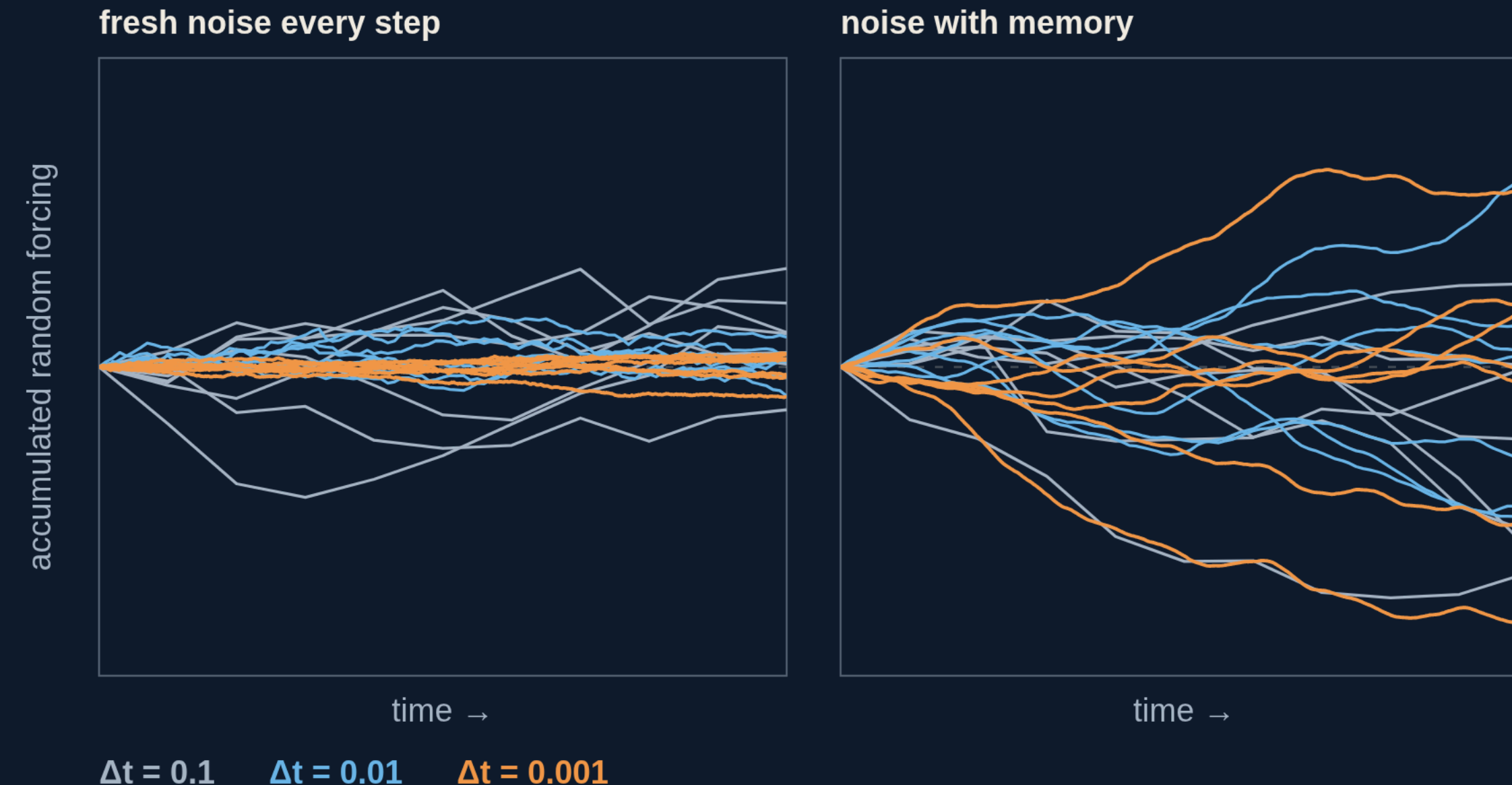
$$\text{Var} \left(\sum_{n=1}^{T/\Delta t} \Delta t \xi_n \right) = T \Delta t \sigma^2 \longrightarrow 0$$

Fresh noise at every step
averages away as $\Delta t \rightarrow 0$.

A stochastic closure without
memory is deterministic
in disguise.

III · BEYOND THE MEAN

A one-point law is not yet a closure



$$\text{Var} \left(\sum_{n=1}^{T/\Delta t} \Delta t \xi_n \right) = T \Delta t \sigma^2 \longrightarrow 0$$

Noise with memory keeps its spread, whatever Δt .

The measured residual has memory: it decorrelates in about one strain time. The one-point law does not fix it.

Next: stochastic LES on Snellius. The first stage has run; no verdict yet.

IV • HOW THIS WAS DONE

How was all this done?

Of the 442 commits in this project since July, *346 list an AI agent as co-author.*

An agent is a language model that acts

you >

project/

README.md	start here: what we believe now
literature/	327 paper notes
findings/	why our views changed
results/	what the data said
reviews/	adversarial reviews
scripts/	analysis code (Julia)
data/	simulation output, figures
paper/	the manuscript
decisions.md	what we committed to
gpu-solver/	the multi-GPU simulation code



tools

- Reference manager: PDFs
- Julia on a desktop GPU
- Supercomputer: 4× H100
- git: version history

An agent is a language model that acts

1 · Literature

you > Read the next papers; check each claim against its PDF.

project/

└ README.md	start here: what we believe now
└ literature/	327 paper notes
└ findings/	why our views changed
└ results/	what the data said
└ reviews/	adversarial reviews
└ scripts/	analysis code (Julia)
└ data/	simulation output, figures
└ paper/	the manuscript
└ decisions.md	what we committed to
└ gpu-solver/	the multi-GPU simulation code

agent

tools

Reference manager: PDFs

Julia on a desktop GPU

Supercomputer: 4× H100

git: version history

✓ new notes in literature/
each claim checked against its PDF

The prompt decides what it reads, runs and writes. Claims change only after my review.

An agent is a language model that acts

1 · Literature

2 · Check a result

you > Does the heavy-tail result survive the new simulations?

project/

└ README.md	start here: what we believe now
└ literature/	327 paper notes
└ findings/	why our views changed
└ results/	what the data said
└ reviews/	adversarial reviews
└ scripts/	analysis code (Julia)
└ data/	simulation output, figures
└ paper/	the manuscript
└ decisions.md	what we committed to
└ gpu-solver/	the multi-GPU simulation code



tools

Reference manager: PDFs
Julia on a desktop GPU
Supercomputer: 4× H100
git: version history

✓ **evidence added to results/**
claim change proposed for my review

The prompt decides what it reads, runs and writes. Claims change only after my review.

An agent is a language model that acts

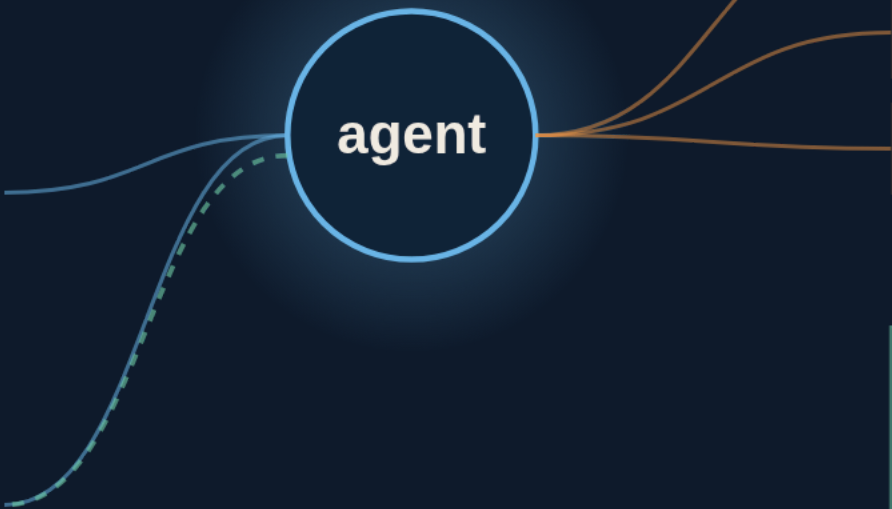
1 · Literature

2 · Check a result

3 · GPU port

you > Port the solver to four GPUs; test it against the one-GPU code.

project/	
├ README.md	start here: what we believe now
├ literature/	327 paper notes
├ findings/	why our views changed
├ results/	what the data said
├ reviews/	adversarial reviews
├ scripts/	analysis code (Julia)
├ data/	simulation output, figures
├ paper/	the manuscript
├ decisions.md	what we committed to
└ gpu-solver/	the multi-GPU simulation code



tools
Reference manager: PDFs
Julia on a desktop GPU
Supercomputer: 4× H100
git: version history

✓ 4 GPUs vs 1 GPU: agree to $2 \cdot 10^{-16}$
tests committed with the code

IV · HOW THIS WAS DONE

An agent is a language model that acts

1 · Literature

2 · Check a result

3 · GPU port

4 · Draft the paper

you > Draft the results section of the paper.



The prompt decides what it reads, runs and writes. Claims change only after my review.

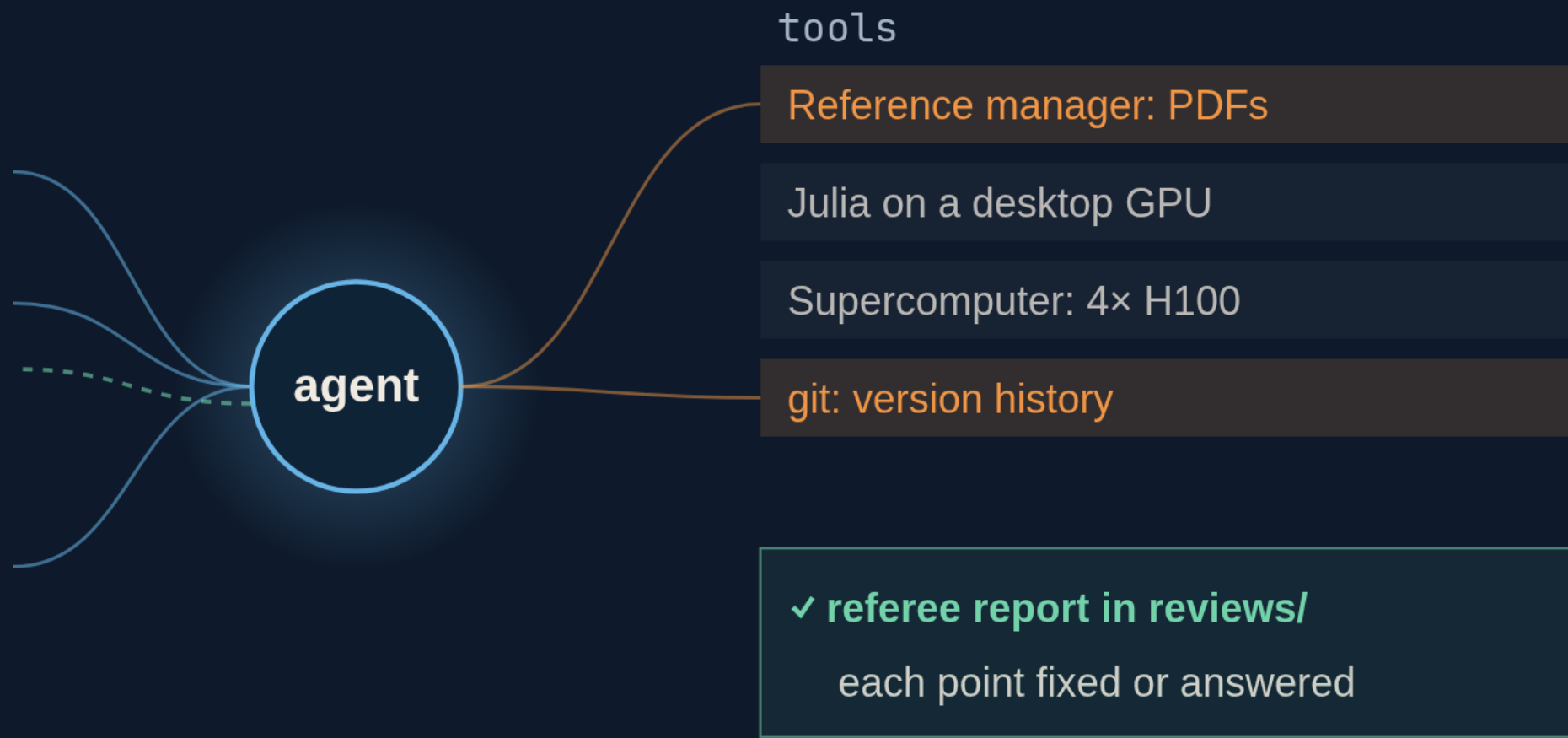
IV · HOW THIS WAS DONE

An agent is a language model that acts

- 1 · Literature
- 2 · Check a result
- 3 · GPU port
- 4 · Draft the paper
- 5 · Referee review

you > Read the paper as a critical referee: what would you reject, and why?

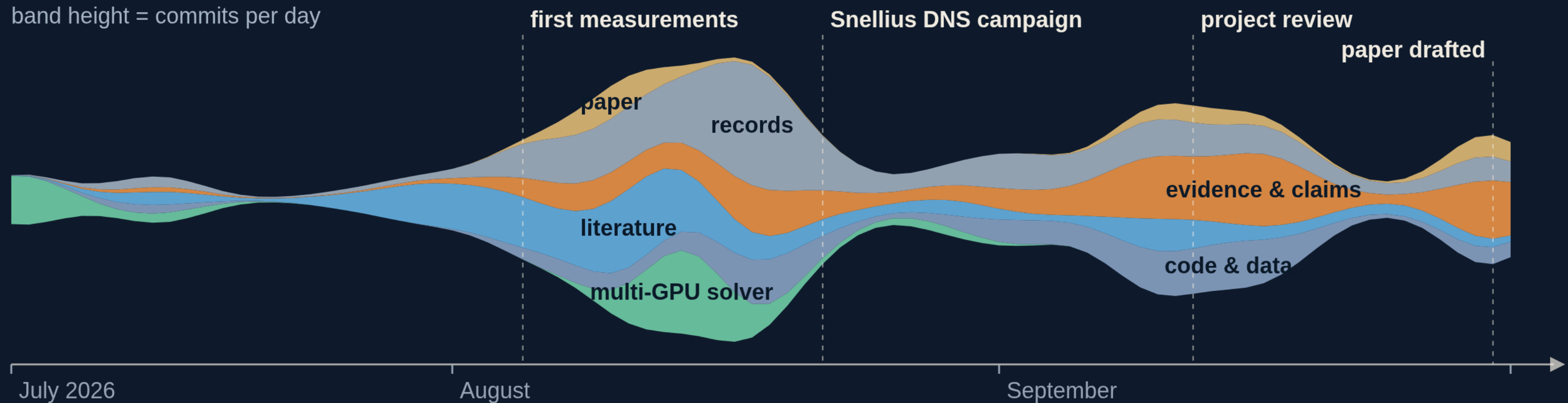
project/	
└ README.md	start here: what we believe now
└ literature/	327 paper notes
└ findings/	why our views changed
└ results/	what the data said
└ reviews/	adversarial reviews
└ scripts/	analysis code (Julia)
└ data/	simulation output, figures
└ paper/	the manuscript
└ decisions.md	what we committed to
└ gpu-solver/	the multi-GPU simulation code



The prompt decides what it reads, runs and writes. Claims change only after my review.

IV · HOW THIS WAS DONE

This project was built with agents



442

commits since July; 346 co-written by an agent

327

papers summarized, of 541 collected

185

analysis scripts in Julia

13

DNS runs on Snellius, up to 1200³

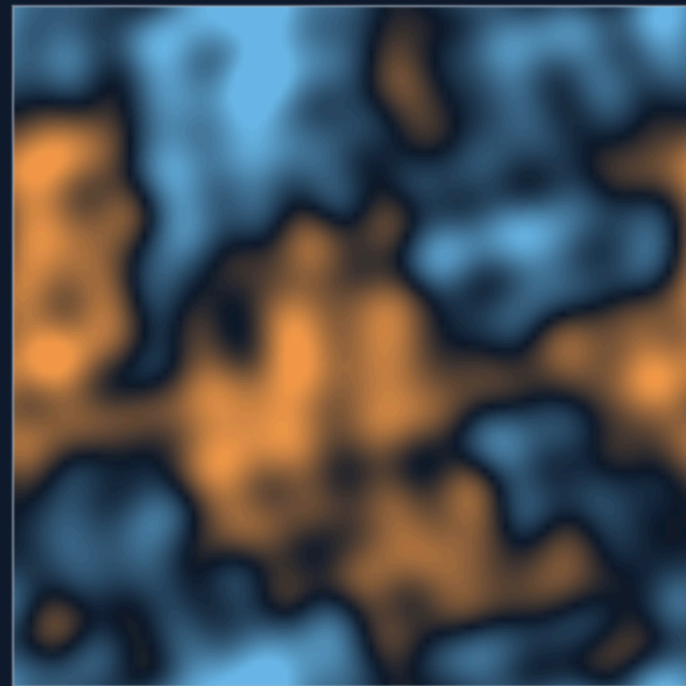
9

claim changes, merged after my review

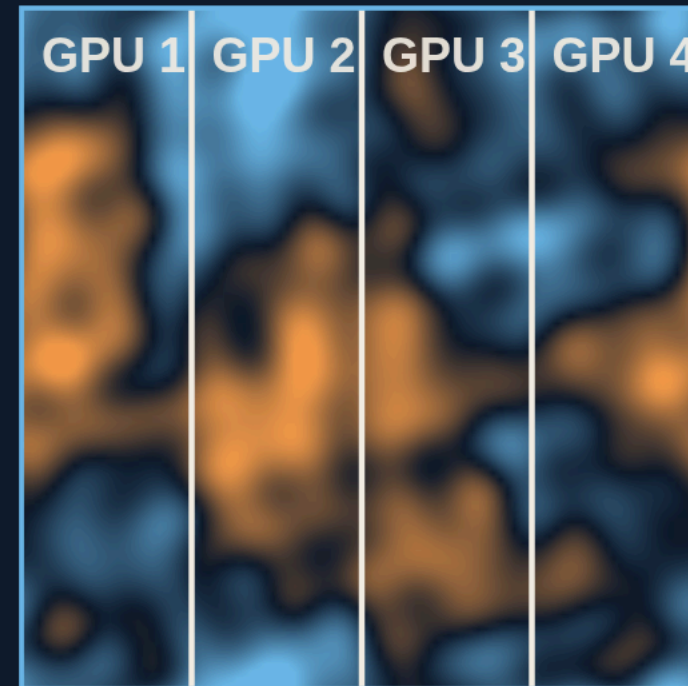
IV · HOW THIS WAS DONE

A GPU solver, built by agents

1 GPU · reference

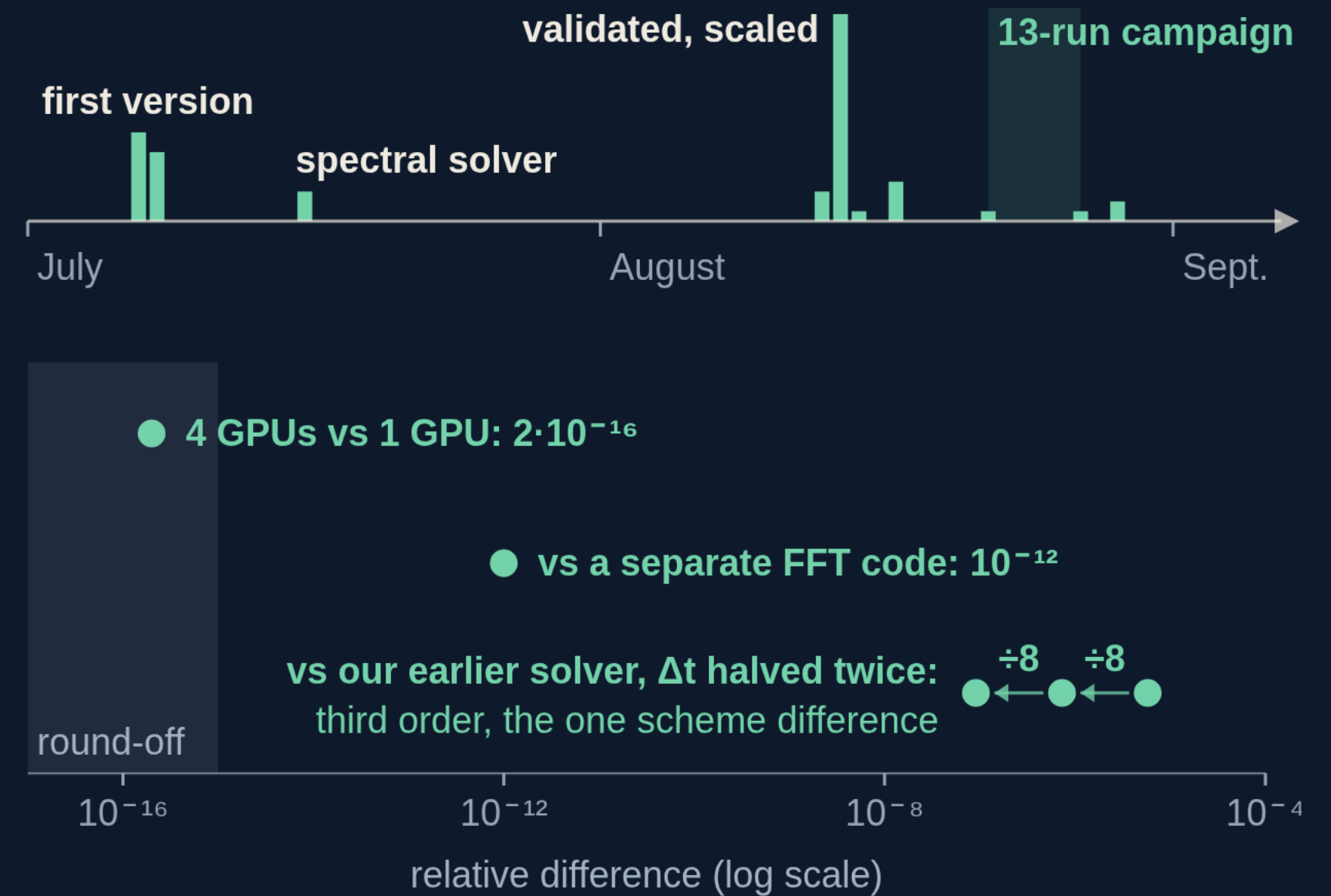


4 GPUs · split into slabs

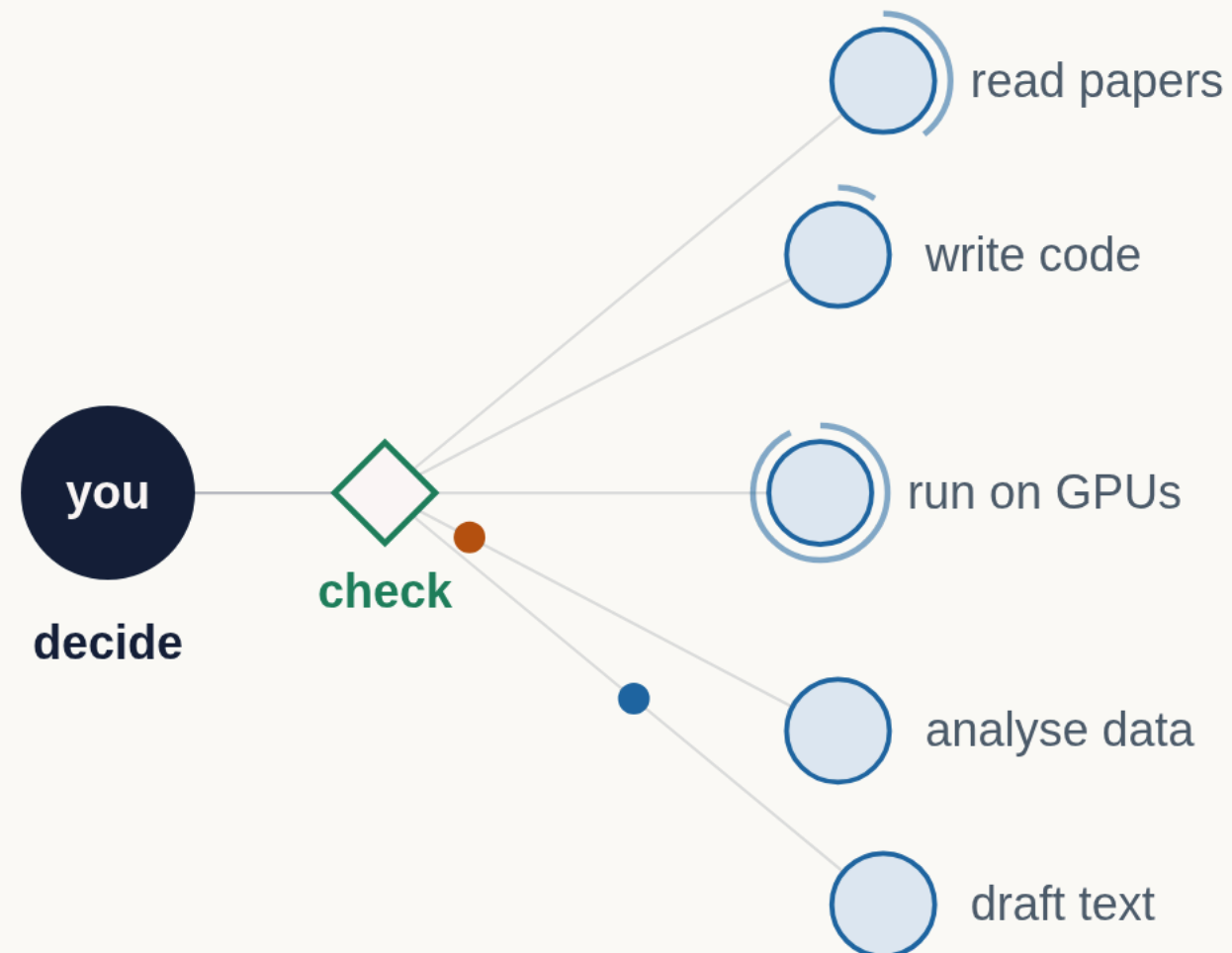


✓ 4 GPUs vs 1 GPU: agree to $2 \cdot 10^{-16}$

commits per day



Our job shifts to orchestrating



Coding → **specifying and verifying**

Reading → **curating and checking claims**

Writing → **deciding what we claim**

More power

×10

agents read, try ideas and
draft results in parallel

Nobody reads it all

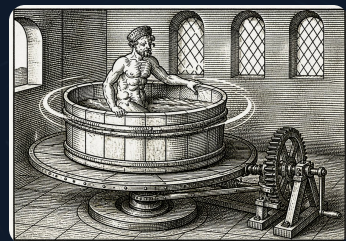
entry points

short summaries

depth: all the rest

*The bottleneck moves from producing
results to judging them.*

TO TAKE HOME



Symmetry makes closures $25\times$ smaller and hands them the Reynolds number.



The best closure is an average; what it misses is heavy-tailed but low-order.



With agents, our job becomes orchestrating and judging.

A coarse grid cannot know the small scales, but it can learn their law.

Thank you · arXiv:2603.05325



agdestein.github.io

Further reading

This talk

Agdestein & Sanderse (2026). Approaching the optimal closure: equivariance, inductive bias, and Reynolds-number generalization in data-driven LES. arXiv:2603.05325.

Agdestein, Verstappen & Sanderse (2026). Exact expressions for the unresolved stress in a finite-volume based large-eddy simulation. J. Comput. Phys. 556, 114810.

Agdestein & Sanderse (2025). Discretize first, filter next: learning divergence-consistent closure models for large-eddy simulation. J. Comput. Phys. 522, 113577.

In preparation: the one-point conditional law of the subfilter-scale stress given the resolved velocity gradient.

Background

Adrian (1990). Stochastic estimation of sub-grid scale motions. Appl. Mech. Rev. 43(5S), S214–S218.

Langford & Moser (1999). Optimal LES formulations for isotropic turbulence. J. Fluid Mech. 398, 321–346.

Zandonade, Langford & Moser (2004). Phys. Fluids 16(7), 2255–2271.

Parthipan, Christensen, Hosking & Wischik (2023). Geosci. Model Dev. 16(15), 4501–4519.

Pope (1983). Phys. Fluids 26, 3448–3450 · Thomson (1987). J. Fluid Mech. 180, 529–556.

Pope (2000). *Turbulent Flows*. Cambridge University Press.