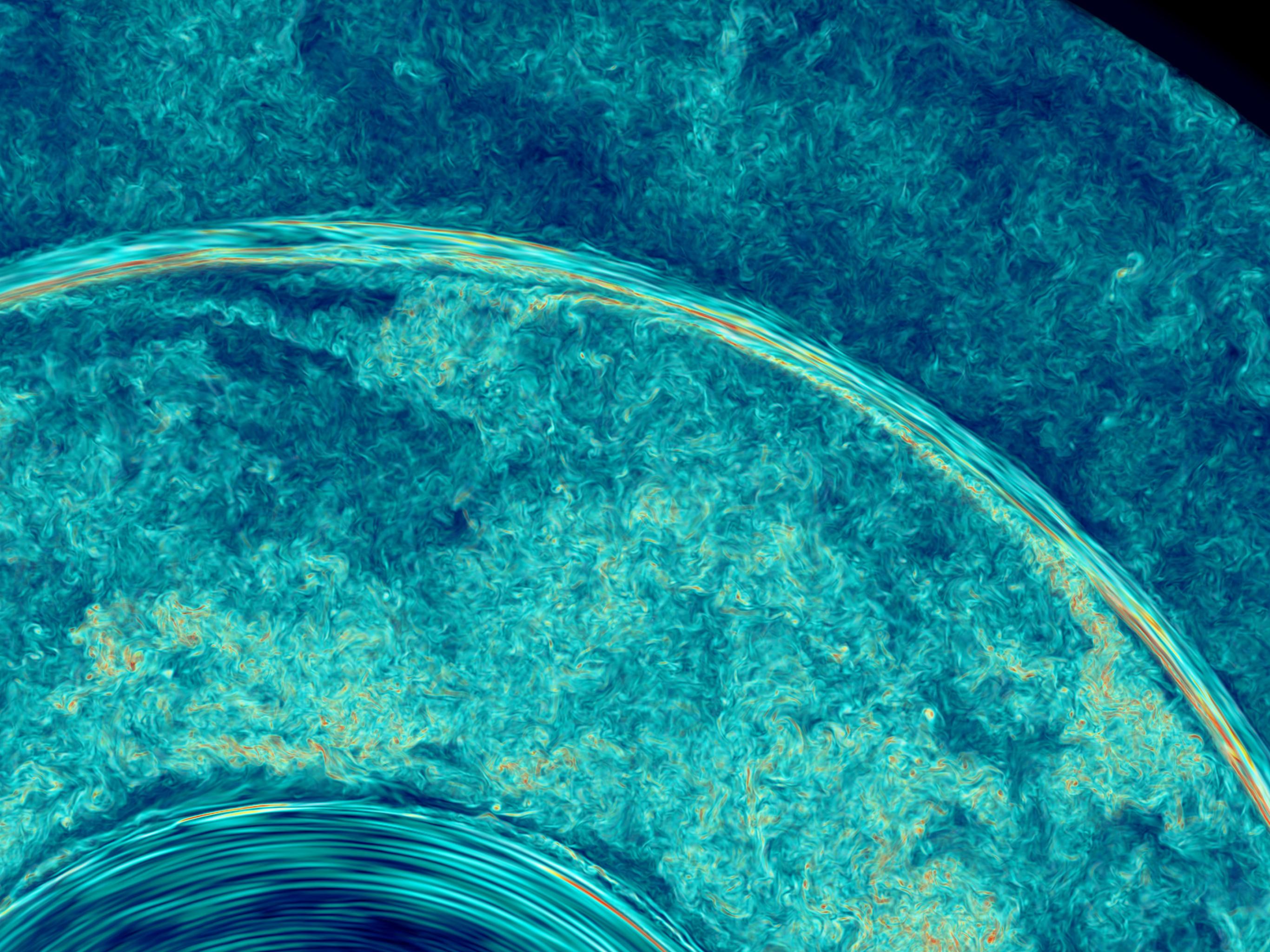


Case study: Large-scale 3D simulations of convection in a star

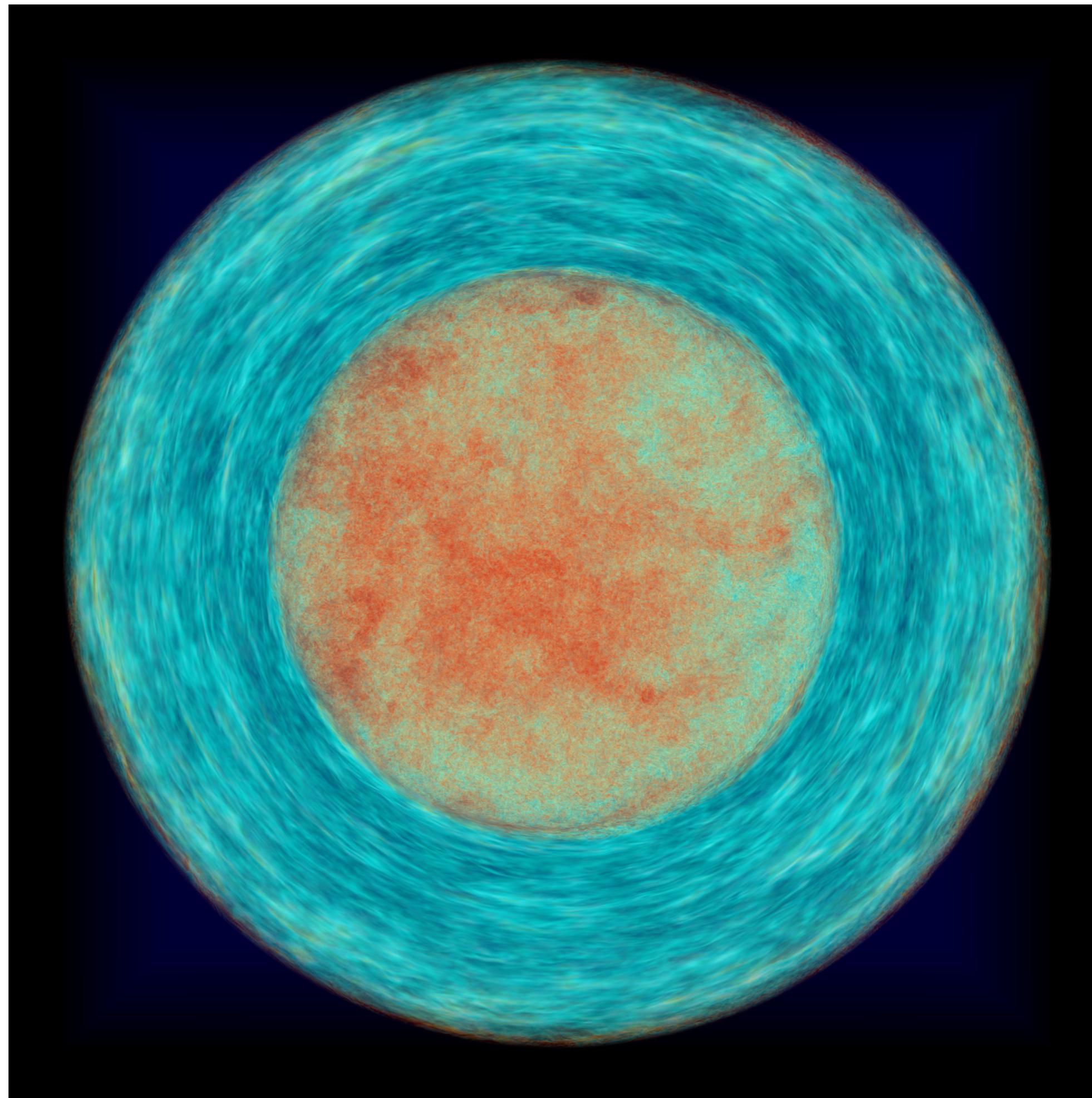
<https://www.ppmstar.org> virtual research environment



Volume rendering of
magnitude of vorticity
of a central slice of a
 4π full sphere
simulation of $25M_\odot$ H-
core burning
convection.

The highly turbulent
nature of stellar
convection is clear in
these simulations from
the irregular yet locally
isotropic distribution of
vorticity.

Simulation performed
with the PPMstar
code (Paul
Woodward, University
of Minnesota)



Large parallel runs on Niagara



[Wiki Main Page](#)

[Support](#)

[Getting started](#)

[Getting help](#)

[Running jobs](#)

[Known issues](#)

[System status](#)

[Resources](#)

Page

[Discussion](#)

Read

[View source](#)

<https://docs.computecanada.ca/wiki/Niagara>

Niagara

Other languages:

[English](#) • [français](#)

Availability: In production since April 2018

Login node: niagara.computecanada.ca

Globus endpoint: [computecanada#niagara](#)

Data mover nodes (rsync, scp, ...): [nia-dm2](#), [nia-dm2](#), see [Moving data](#)

System Status Page: <https://docs.scinet.utoronto.ca>

Niagara hardware specifications [\[edit\]](#)

- 1548 nodes, each with 40 Intel Skylake cores at 2.4GHz, for a total of 61,920 cores.
- 202 GB (188 GiB) of RAM per node.
- EDR Infiniband network in a so-called 'Dragonfly+' topology.
- 7PB of scratch, 2PB of project space (parallel filesystem: IBM Spectrum Scale, formerly known as GPFS).
- 256 TB burst buffer (Excelero + IBM Spectrum Scale).
- No local disks.
- No GPUs.
- Theoretical peak performance ("Rpeak") of 4.75 PF.
- Measured delivered performance ("Rmax") of 3.07 PF.
- 685 kW power consumption.

About effective large-scale parallel simulations

- Key challenge: processing on cores is fast but getting data there is slow
- Overlap of communication and computation
- Set aside team leaders to deal with communications while workers continue to work
- Compress messages into smaller number of large messages, requires finding balance, depends on hardware
- Processors get faster but the network is not getting faster at the same rate
- Dedicated code to do just this type of problem, not general purpose code
- $1.2^{10}=6.2 \rightarrow$ pick up each 10-20% effect, for example
- Beware of the different flop cost of different operations, there is a logical operation for addition or multiplication (1 flop each) but not for sin, cos or divide (14 flops each)
- Solve problems the way computers like them, often not how mathematicians would do it

References:

Woodward PR, Herwig F, Wetherbee T. *Comput. Sci. Eng.* 20(5):8–17
Woodward PR, Lin P-H, Mao H, Andrassy R, Herwig F. 2018. eprint
arXiv:1810.13416 (ASTRONUM conference paper)

Cyber challenges

- Data deluge naive approach: one dump 4 quantities $1536^3 \times 8$ byte x 4 quantities = 115GB - disk read (100MB/s) **20min**, network transport (20MB/s) **1.6hrs**
 - 500 time steps = 57.5TB - disk read **6.6d**, network transport **33d**
 - Large data sets require remote access and smart data management strategies
-
- Multi-scale, multi-physics, multi-method
 - Complex data interactions and collaborative cyber- research environments, data fusion
-
- Research data management - in reality means combining access to data, data-specific analytics tools and capability to execute analytic tools to data
 - Sharing work flows
 - Legacy software, tools, and workflows
 - Reproducibility of science

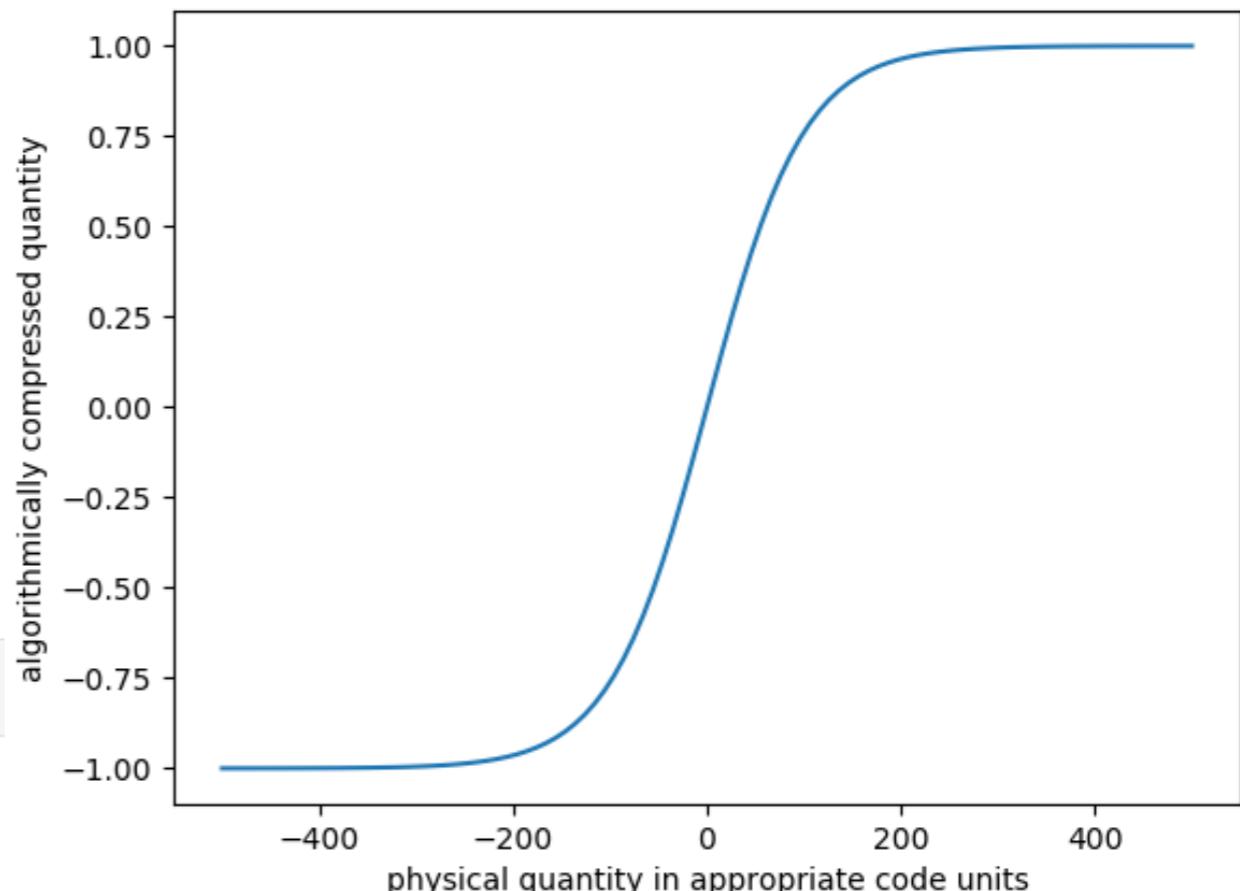
Data management for PPMstar code

- In-code generation of science-ready analysis data and images (this is a substantial departure from common practice)
 - instead of just writing out *raw* data we generate three types of algorithmically compressed file types
 - bobs - bricks of bytes algorithmically compressed
 - briquette data - spatially filtered (by factor 4 in each direction, 4 bytes, but 32 primary and derived quantities)
 - rprofs - radial profiles of lots of things
 - as well as one image for each of 10 quantities for each output dump
 - New feature: data compression (zip) takes place in code and data is written algorithmically and data compressed as well as filtered
- Example: one 768^3 -grid run with 1M time steps writes out 466 dumps
 - data held on project for immediate analysis (briquette & rprofs): 260GB
 - data on nearline: 3 restart dumps and bobs of 10 quantities (631GB)
 - instead of:

```
d = 466 *8*10* 768**3 /1.e12
print("Uncompressed data volume: 466 * 10 * 8 * 768**3 /1.e12 = {:.2f}TB".format(d))
```

Uncompressed data volume: 466 * 10 * 8 * 768**3 /1.e12 = 16.89TB

Example for algorithmic data compression



Sharing computational science with the community

Dynamics in a stellar convective layer and at its boundary: Comparison of five 3D hydrodynamics codes

R. Andrassy¹, J. Higl¹, H. Mao², M. Mocák³, D. G. Vlakrov⁴, W. D. Arnett⁵, I. Baraffe^{4, 6}, S. W. Campbell^{7, 8}, T. Constantino⁴, P. V. F. Edelmann⁹, T. Goffrey¹⁰, T. Guillet⁴, F. Herwig^{11, 12}, R. Hirschi^{3, 13}, L. Horst¹, G. Leidi^{1, 14}, C. Meakin^{5, 15}, J. Pratt¹⁶, F. Rizzuti³, F. K. Röpke^{1, 17}, and P. Woodward^{2, 12}

R. Andrassy et al.: Comparison of five 3D hydrodynamics codes

Verification: Do you solve the equations right?

Validation: Do you solve the right equation?

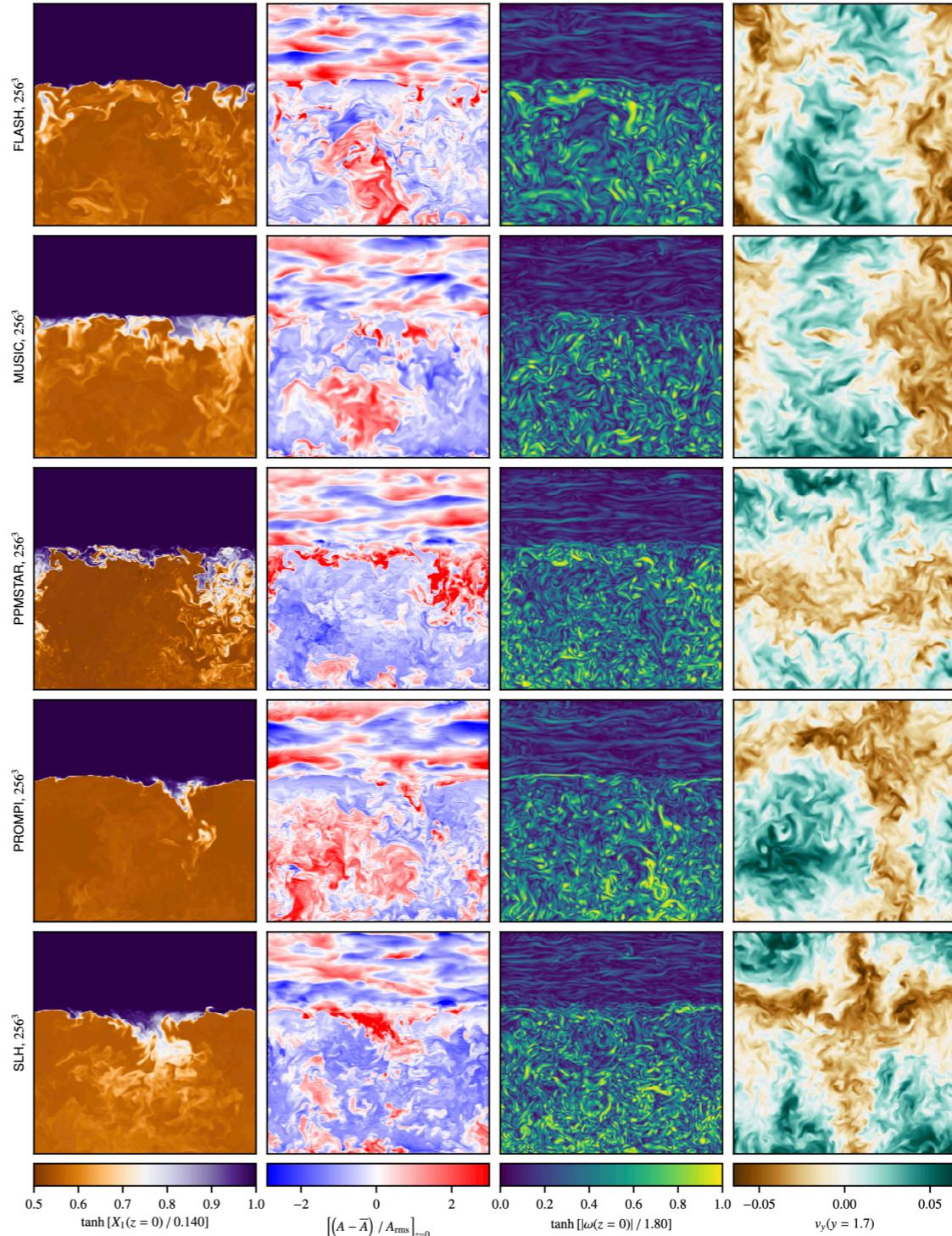


Fig. 2. Renderings of the flow field at $t = 1000$. The variables shown (from left to right) are the mass fraction X_1 , relative entropy fluctuations $(A - \bar{A})/A_{\text{rms}}$, magnitude of vorticity $|\omega|$, and the vertical component of velocity v_y . The first three variables correspond to slices through the simulations box in the $z=0$ plane and the last one in the $y=1.7$ plane. The non-linear scaling of X_1 and $|\omega|$ and the normalisation of entropy fluctuations by their horizontal rms spread A_{rms} (see the colour bars) are introduced for visualisation purposes.

PPMstar CoCoPy/1D2D/1D- (3)

ppmstar.org/coco/user/fherwig/d...

1D-profiles.ipynb

File Edit View Run Kernel Git Diagram Tabs Settings Help

Code git Python 3

Velocity field

```
[7]: res = 256
runs = np.append(run_lists[res], ['PPMSTAR-512', 'PROMPI-512'])

ifig=7; close(ifig);
figure(ifig, figsize=(width, 0.7*width), dpi=scrdpi)
for i, rid in enumerate(runs):
    plot(time[rid], v_rms_cl[rid], ls=ls[i], color=lc[i], label=run_lbls[rid])
    plot(time[rid], v_rms_sl[rid], ls=ls[i], color=lc[i])

    idx0 = np.argmin(np.abs(time[rid] - time0))
    mean_cl = np.mean(v_rms_cl[rid][idx0:])
    sigma_cl = np.mean((v_rms_cl[rid][idx0:] - mean_cl)**2)**0.5
    p = np.poly1d(np.polyfit(time[rid][idx0:], v_rms_cl[rid][idx0:], 1))
    #plot(time[rid], p(time[rid]), ls=ls[i], color=lc[i])

    change_cl = (p(time[rid][-1]) - p(time[rid][idx0]))
    mean_sl = np.mean(v_rms_sl[rid][idx0:])
    print('{:s}, {:.4f}, {:.1f}, {:.1f}, {:.4f}'.format(rid, mean_cl,
        100.*3.*sigma_cl/mean_cl, 100.*change_cl/mean_cl, mean_sl))
    axvline(x=time0, ls=':', color='k')
    legend(loc='center right', ncol=2)
    text(550, 0.0395, 'convective layer')
    text(550, 0.005, 'stable layer')
    xlim((0., 2e3))
    ylim((0., 0.0425))
    xlabel(r'$t$')
    ylabel(r'$\langle v \rangle_{rms}$')
    tight_layout()
    savefig(os.path.join(plot_dir, 'v_rms_evolution_{:d}.pdf'.format(res)), dpi=savedpi)

FLASH-256, 0.0347, 13.2, 4.1, 0.0088
MUSIC-256, 0.0334, 13.3, 4.4, 0.0090
PPMSTAR-256, 0.0342, 13.1, 1.3, 0.0093
PROMPI-256, 0.0343, 11.0, 1.5, 0.0096
SLH-256, 0.0343, 15.5, 11.1, 0.0088
PPMSTAR-512, 0.0354, 15.6, 5.8, 0.0094
PROMPI-512, 0.0349, 15.0, 16.9, 0.0094
```

Figure 7

PPMstar

3D Stellar Hydro Simulations [3D Panel Google Doc]

Research Hubs Movies About

access to different JupyterHub services that allow analytic and exploratory access to PPMstar outputs. We aim to provide the data and the notebooks required to make the figures in our Outreach Hub. If you want to just try out some notebooks use the Outreach Hub. If you want to do more in this platform please contact us and we can provide you enhanced storage, data access and star Collaborate hub.

Outreach, PPMstar Collaborate and CoCo hub you need to [register](#).

PPMstar Collaborate
Collaborate with PPMstar team
Persistent access to unpublished data, contact Falk Herwig or Paul Woodward to discuss project ideas. Register to access, see link above.

CoCo
Stellar hydro code comparison project
Sharing and exploring data from different hydro codes. Register to access, see link above.

velocity fluctuations in the convection zone ($y < y_{ub}(t) - 0.1$, upper set of curves) and the location $y_{ub}(t)$ of the convective boundary is tracked in time as shown in Fig. 9. The excluded from the analysis.