Massively parallel nonhydrostatic ocean modeling on GPUs

Ali Ramadhan

December 10, 2018

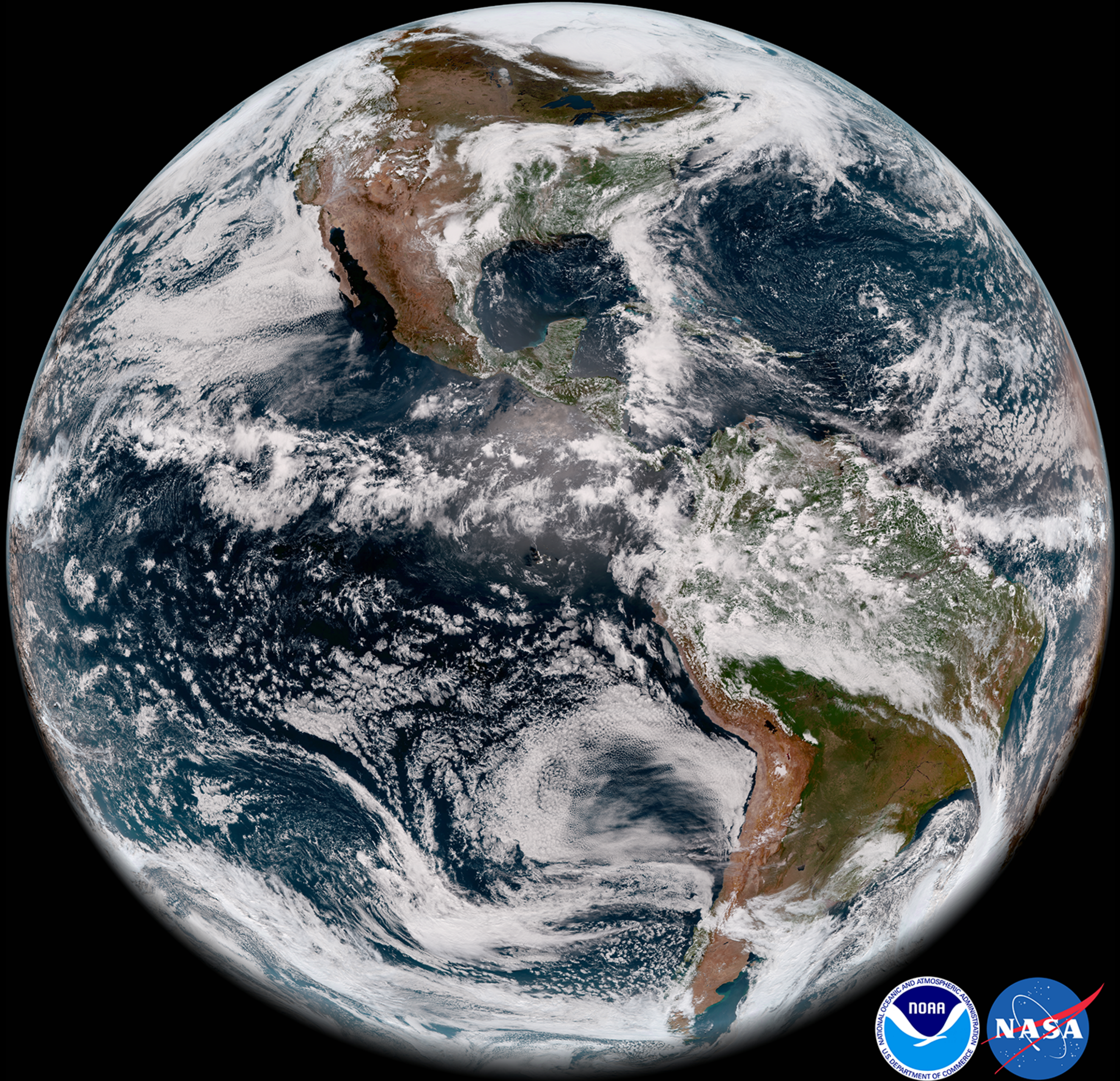
18.337/6.338 Modern Numerical Computing

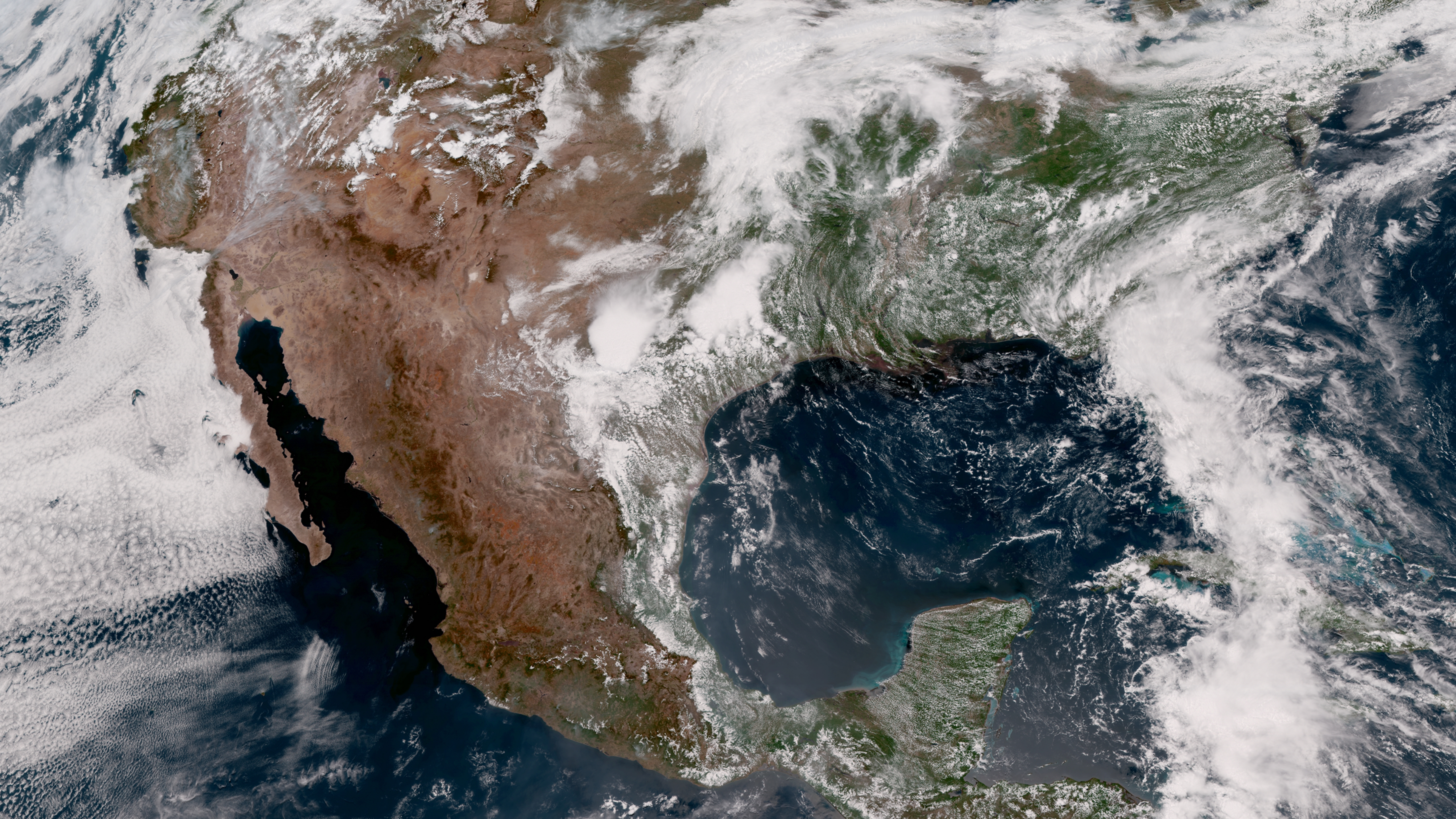
Massachusetts Institute of Technology

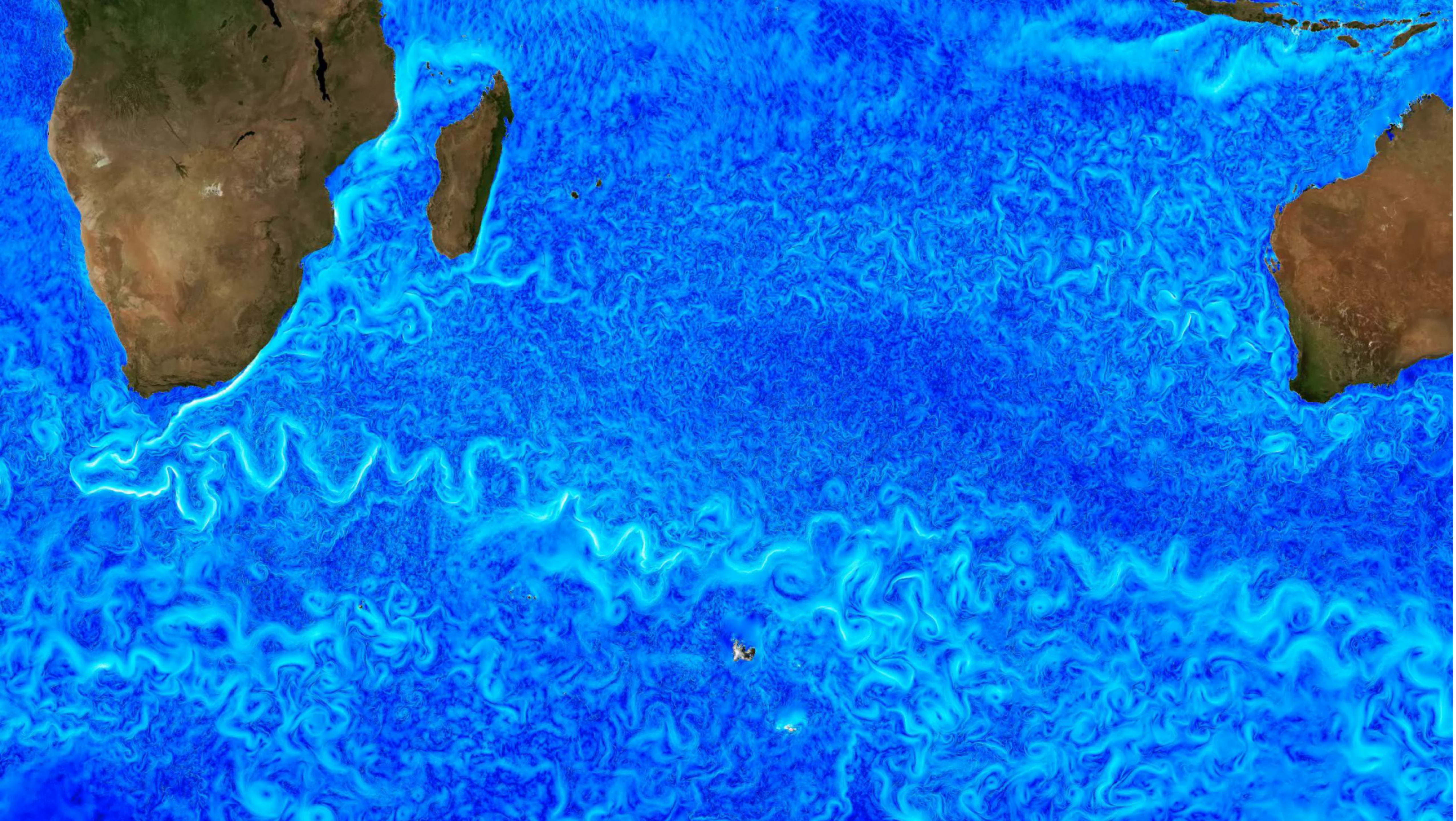
EAPS
Earth, Atmospheric and Planetary Sciences

Acknowledgements

- While I did write the code, I did not do this alone!
- **EAPS**: Chris Hill, Jean-Michel Campin, John Marshall, Greg Wagner, Mukund Gupta.
- **Julia Lab**: Valentin Churavy, Peter Ahrens, Shashi Gowda, Sung Woo Jeong, Ranjan Anantharaman, Alan Edelman.

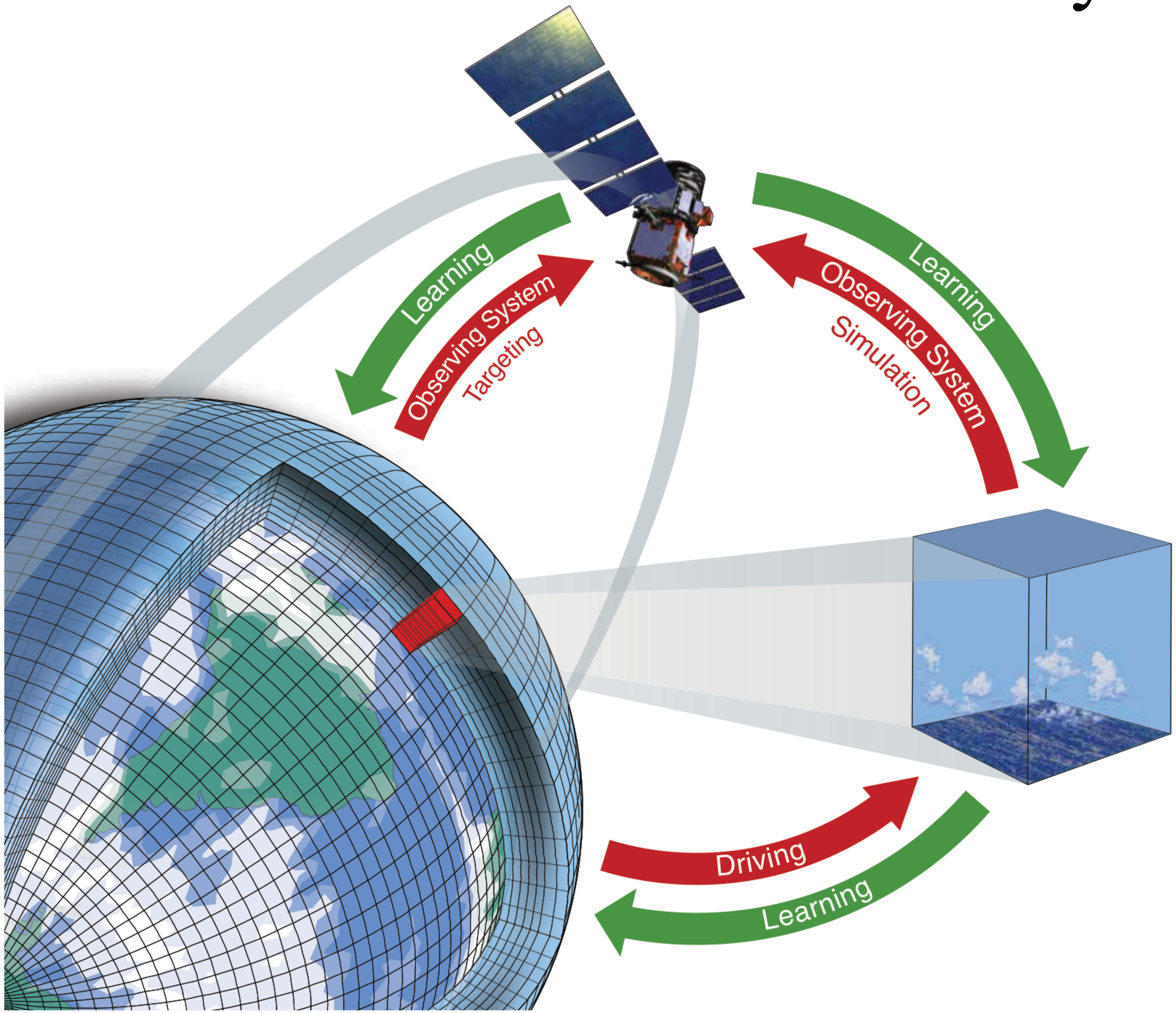

NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
NOAA
U.S. DEPARTMENT OF COMMERCE
NASA



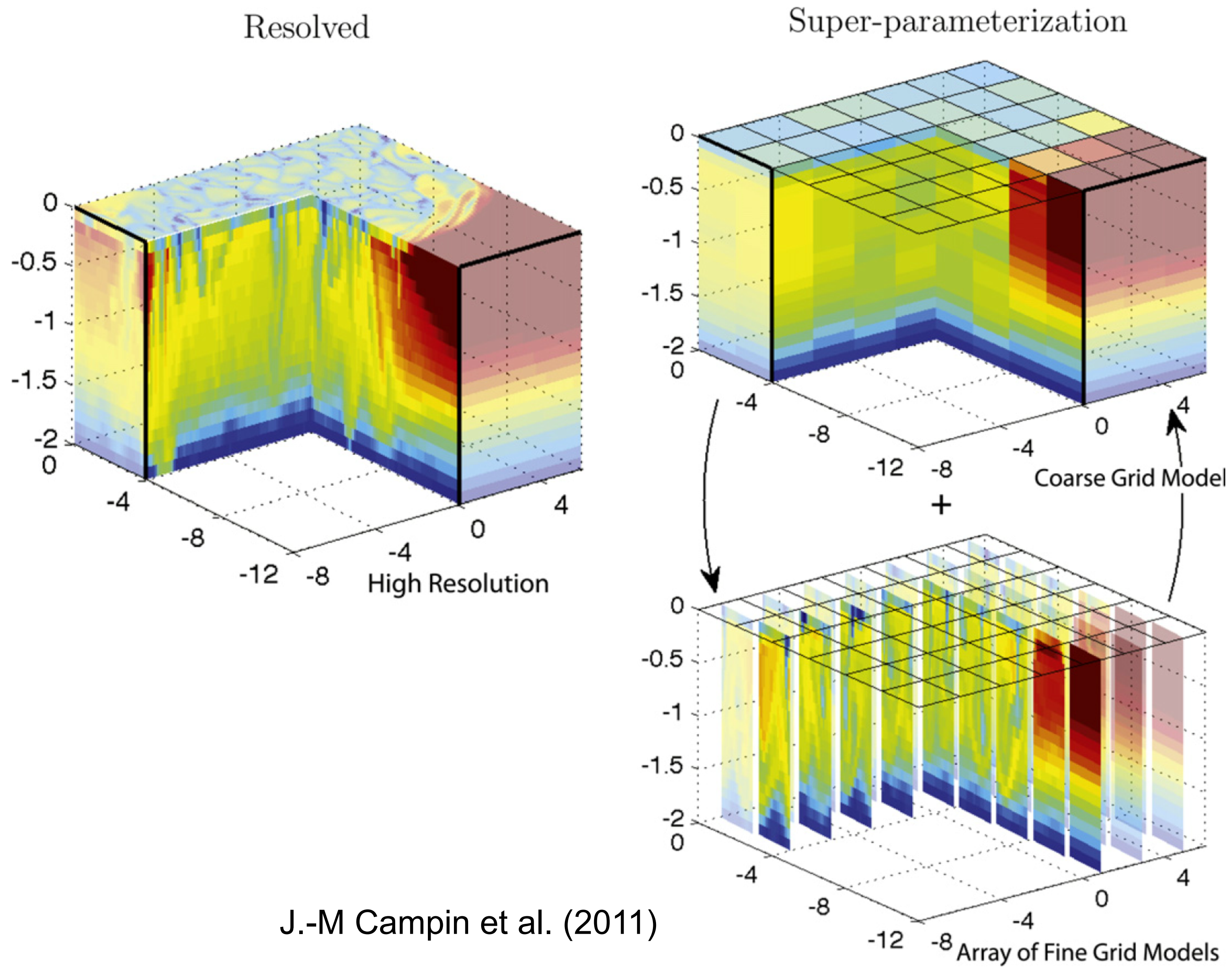



Aghulhas Rings

Data-driven Earth System Modeling



Super-parameterization in the MITGCM on GPUs



J.-M Campin et al. (2011)

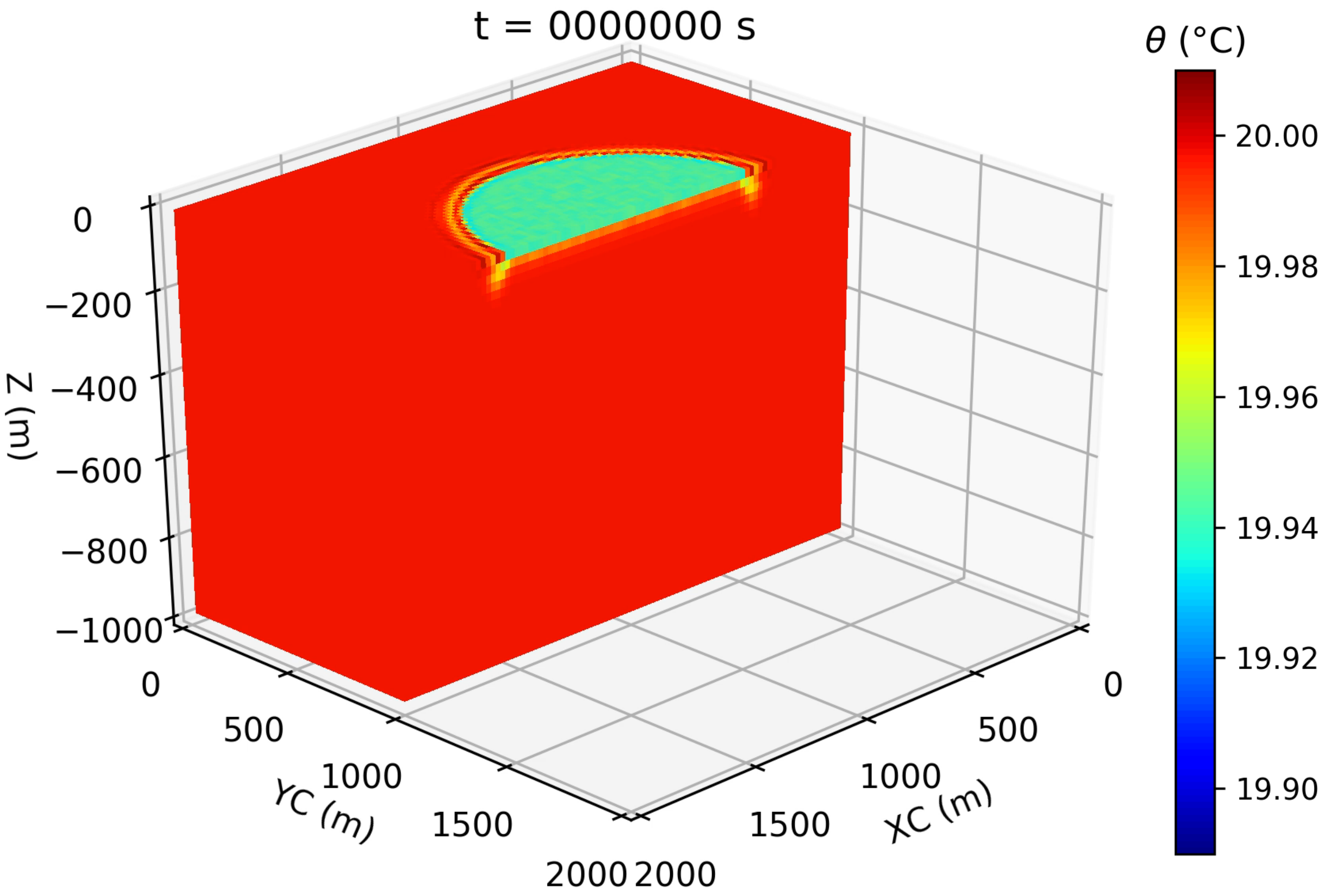
Non-hydrostatic to hydrostatic computation ratio:

HYD:SP:NH = 1:50:1000

Lofty goal:

HYD:SP:NH = 1:5:1000
(with 3D column models on GPUs)

Benchmarking experiment: deep convection



- Experiment: deep convection forced by a cold pool at the surface.

- Super-parameterization (fine model) precursor.

- N = 100×100×50 = 500,000 points
- 2×2×1 km domain
- Doubly periodic
- *f*-plane

Basic Hotspots Hotspots by CPU Usage viewpoint (change)

Collection Log | Analysis Target | Analysis Type | Summary | Bottom-up | Caller/Callee | Top-down Tree | Platform

Grouping: Call Stack

Single CPU run (CG3D)

Function Stack	CPU Time: Total	CPU Time: Self	Module	Function (Full)	Source File	Start Address
Total	100.0%	0s				
_start	100.0%	0s	mitgcmuv	_start		0x401e60
__libc_start_main	100.0%	0s	libc.so.6	__libc_start_main		0x21a40
main	100.0%	0s	mitgcmuv	main		0x401f50
main	100.0%	0s	mitgcmuv	main	main.f	0x5478b0
the_model_main	100.0%	0s	mitgcmuv	the_model_main	the_model_main.f	0x5fb990
the_main_loop	99.8%	0s	mitgcmuv	the_main_loop	the_main_loop.f	0x5fb7f0
main_do_loop	99.6%	0s	mitgcmuv	main_do_loop	main_do_loop.f	0x5de830
forward_step	99.5%	0s	mitgcmuv	forward_step	forward_step.f	0x5a8810
solve_for_pressure	77.2%	0.030s	mitgcmuv	solve_for_pressure	solve_for_pressure.f	0x5f3ee0
cg3d	75.3%	48.641s	mitgcmuv	cg3d	cg3d.f	0x57f490
cg2d	1.3%	0.770s	mitgcmuv	cg2d	cg2d.f	0x57b5c0
calc_div_ghat	0.4%	0.270s	mitgcmuv	calc_div_ghat	calc_div_ghat.f	0x568120
pre_cg3d	0.2%	0.140s	mitgcmuv	pre_cg3d	pre_cg3d.f	0x5e5910
dynamics	9.7%	0.050s	mitgcmuv	dynamics	dynamics.f	0x595b90
mom_fluxform	5.1%	1.640s	mitgcmuv	mom_fluxform	mom_fluxform.f	0x4bf420
calc_gw	1.8%	1.130s	mitgcmuv	calc_gw	calc_gw.f	0x56bb70
timestep	0.9%	0.300s	mitgcmuv	timestep	timestep.f	0x5fc220
diagnostics_fill	0.7%	0s	mitgcmuv	diagnostics_fill	diagnostics_fill.f	0x40dde0
calc_phi_hyd	0.4%	0.080s	mitgcmuv	calc_phi_hyd	calc_phi_hyd.f	0x570830
calc_viscosity	0.2%	0.140s	mitgcmuv	calc_viscosity	calc_viscosity.f	0x575d00
diags_phi_hyd	0.2%	0.110s	mitgcmuv	diags_phi_hyd	diags_phi_hyd.f	0x591510
__intel_avx_rep_memset	0.1%	0.090s	libintlc.so.5	__intel_avx_rep_memset		0x45d80
timestep_wvel	0.1%	0.080s	mitgcmuv	timestep_wvel	timestep_wvel.f	0x5ff930
__intel_avx_rep_memcpy	0.1%	0.060s	libintlc.so.5	__intel_avx_rep_memcpy		0x445c0
monitor	4.5%	0s	mitgcmuv	monitor	monitor.f	0x4d9b60
do_statevars_diags	3.4%	0s	mitgcmuv	do_statevars_diags	do_statevars_diags.f	0x5954f0
thermodynamics	2.8%	0.220s	mitgcmuv	thermodynamics	thermodynamics.f	0x5fbb50
momentum_correction_step	0.6%	0s	mitgcmuv	momentum_correction_step	momentum_correction_step.f	0x5de940
integr_continuity	0.4%	0s	mitgcmuv	integr_continuity	integr_continuity.f	0x5db2c0
do_oceanic_phys	0.3%	0.120s	mitgcmuv	do_oceanic_phys	do_oceanic_phys.f	0x5943c0
do_fields_blocking_exchanges	0.2%	0s	mitgcmuv	do_fields_blocking_exchanges	do_fields_blocking_exchanges.f	0x594270
do_write_pickup	0.1%	0s	mitgcmuv	do_write_pickup	do_write_pickup.f	0x595770
diagnostics_write	0.1%	0s	mitgcmuv	diagnostics_write	diagnostics_write.f	0x4459d0
do_the_model_io	0.1%	0s	mitgcmuv	do_the_model_io	do_the_model_io.f	0x5955e0
timer_start	0.0%	0s	mitgcmuv	timer_start	timers.f	0x559580
timer_stop	0.0%	0s	mitgcmuv	timer_stop	timers.f	0x55b9f0
timer_start	0.0%	0s	mitgcmuv	timer_start	timers.f	0x559580
initialise_varia	0.3%	0s	mitgcmuv	initialise_varia	initialise_varia.f	0x5d72c0
initialise_fixed	0.2%	0s	mitgcmuv	initialise_fixed	initialise_fixed.f	0x5d6e50
timer_printall	0.0%	0s	mitgcmuv	timer_printall	timers.f	0x557190

Top Hotspots

25 node MPI run

This section lists the most active functions in your application. Optimizing these hotspot functions typically results in improving overall application performance.

Function	Module	CPU Time
pmpi_recv_	libmpifort.so.12	176.132s
cg3d_	mitgcmuv	129.641s
pmpi_allreduce_	libmpifort.so.12	82.292s
pmpi_isend_	libmpifort.so.12	15.141s
exch_rl_send_put_x_	mitgcmuv	11.782s
[Others]		112.011s

CPU Usage Histogram

This histogram displays a percentage of the wall time the specific number of CPUs were running simultaneously. Spin and Overhead time adds to the Idle CPU usage value.