NIWA
Taihoro Nukurangi

Toward an adaptive GPU general inundation model

Cyprien Bosserelle, Emily Lane, Wolfgang Hayek, Richard Gorman

Climate, Freshwater & Ocean Science

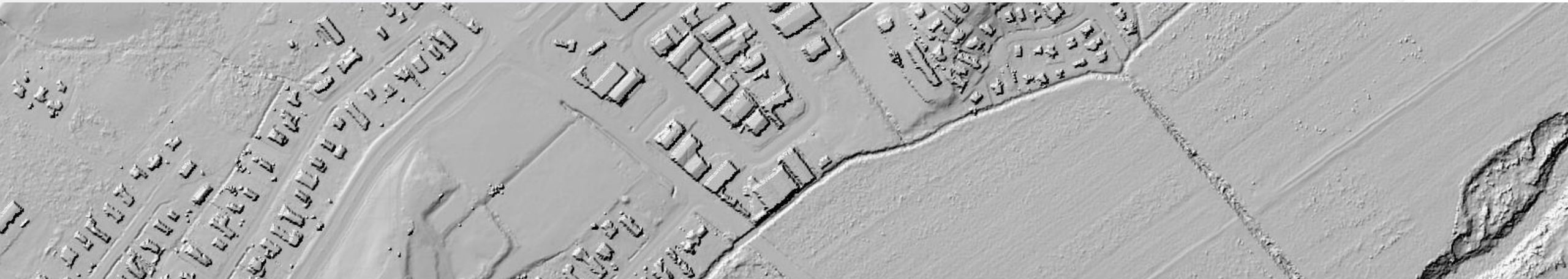

Jacuzzi
HOT TUBS ONTARIO

Models takes, like, forever!

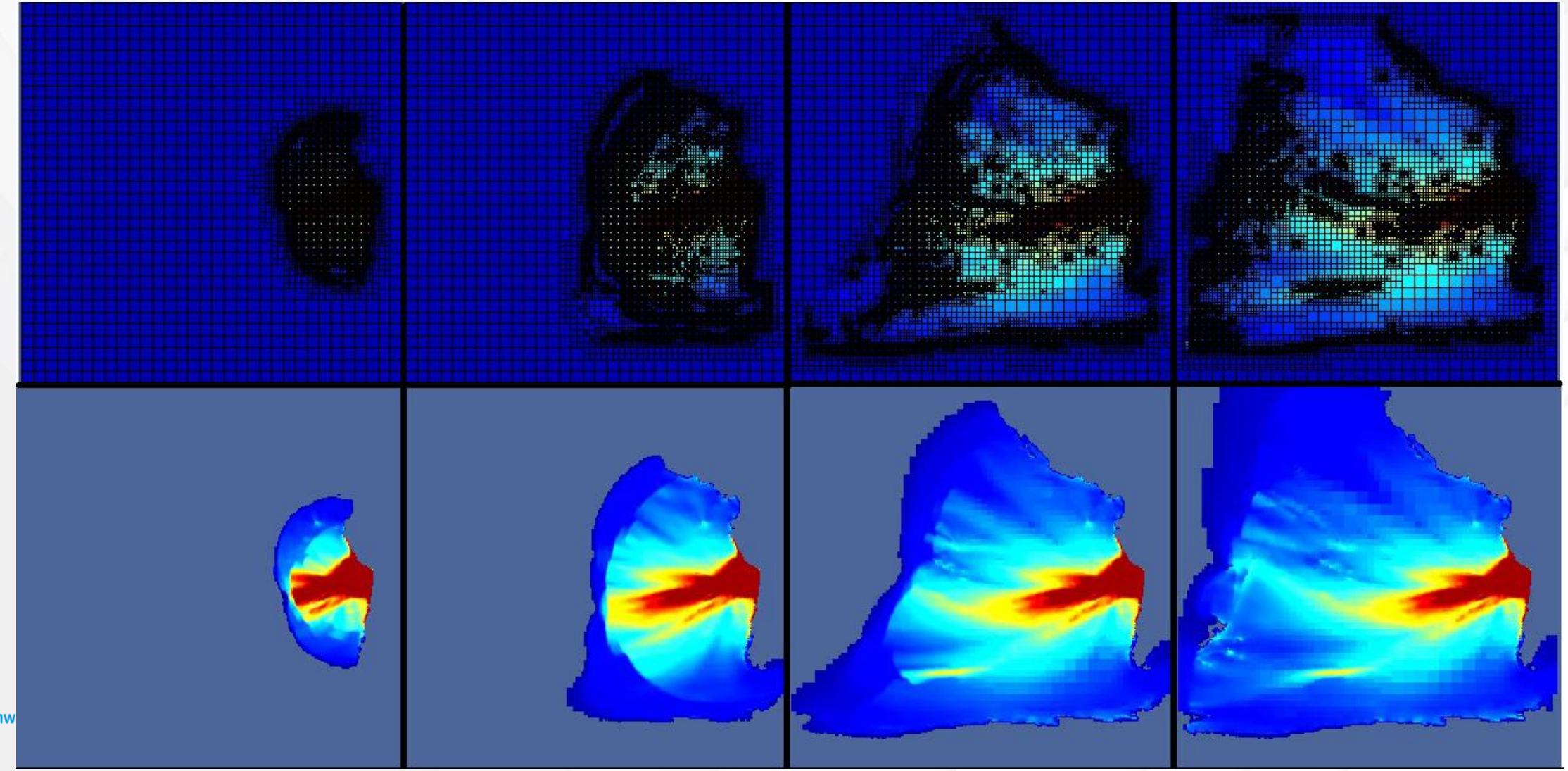
- Need to account for the whole flood/coastal plain
- LiDAR topography and bathymetry available at 1m resolution
- “Shock-capturing” schemes (explicit time stepping == small time steps) or NH/Boussinesq

If the model is too slow

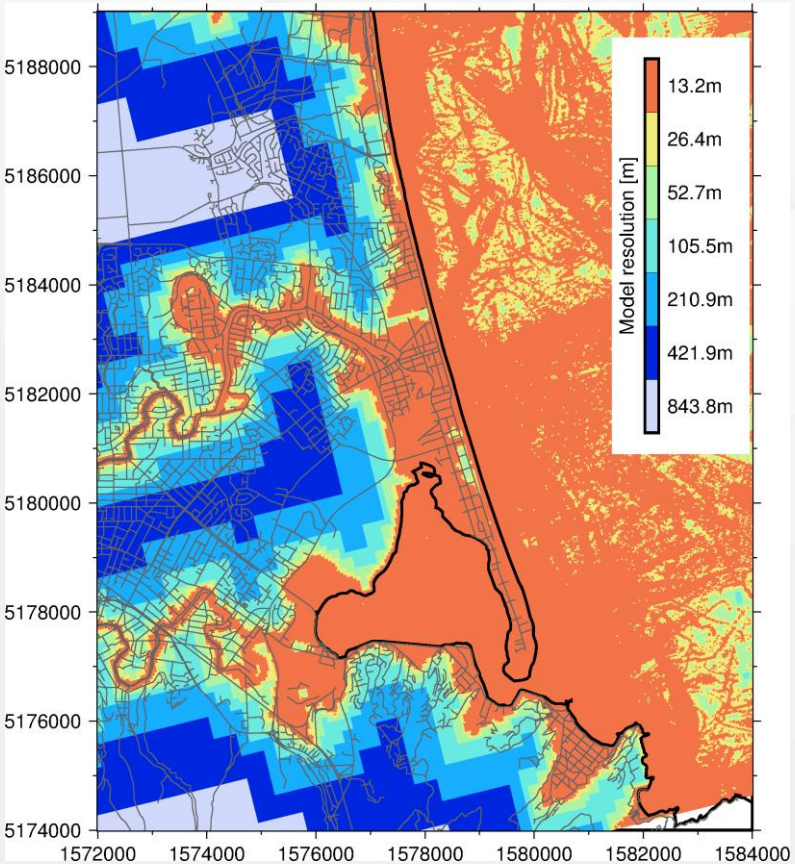
- No forecasting
- No probabilistic assessment
- No sensitivity analysis
- No gradient-descend calibration of the “nobs”

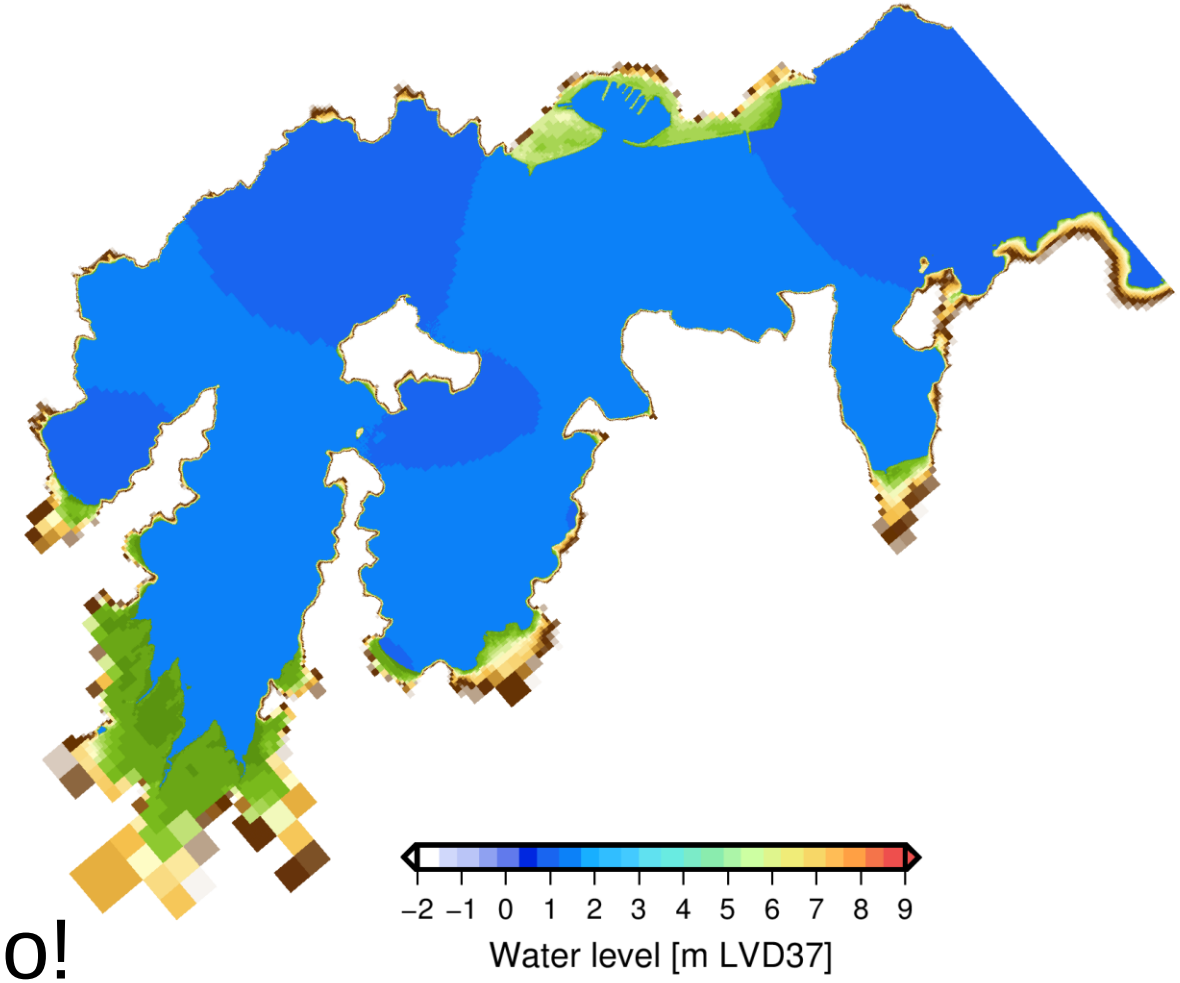


Basilisk: SWE with quadtree adaptive mesh refinement

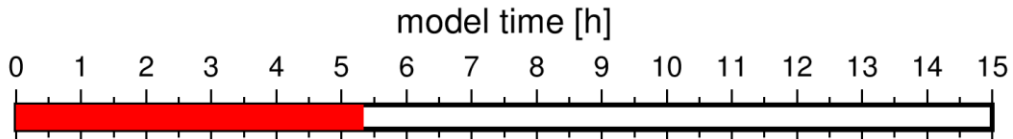


Basilisk: SWE with quadtree adaptive mesh refinement





Basilisk is frustratingly slow too!

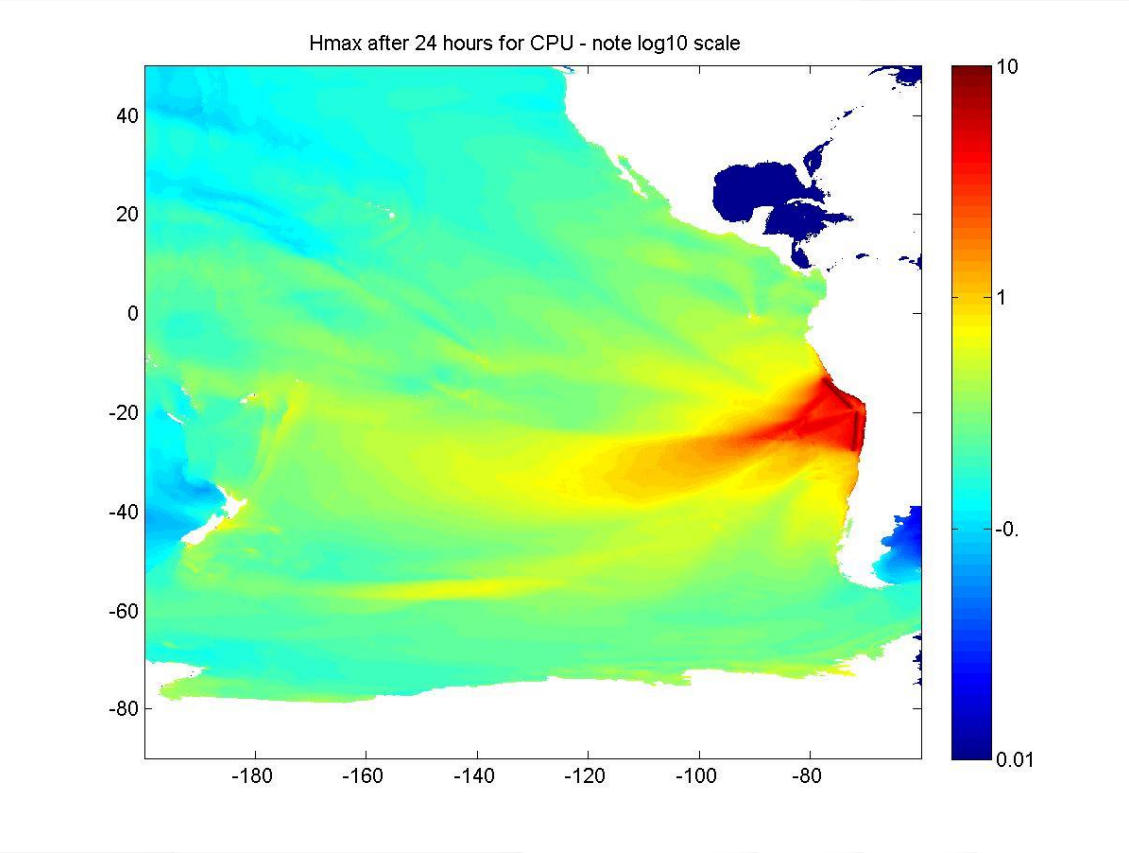


Making Basilisk run on GPU

Basilisk meets Nesi - a cross-species romance?

How

- Use openACC compiler
- Need to insert instruction to copy data and run part of the code on the GPU
- Only done for cartesian grid (not impleted in adaptive mode)
- Testing done on a transpacific tsunami propagation and a tsunami inundation in Efate



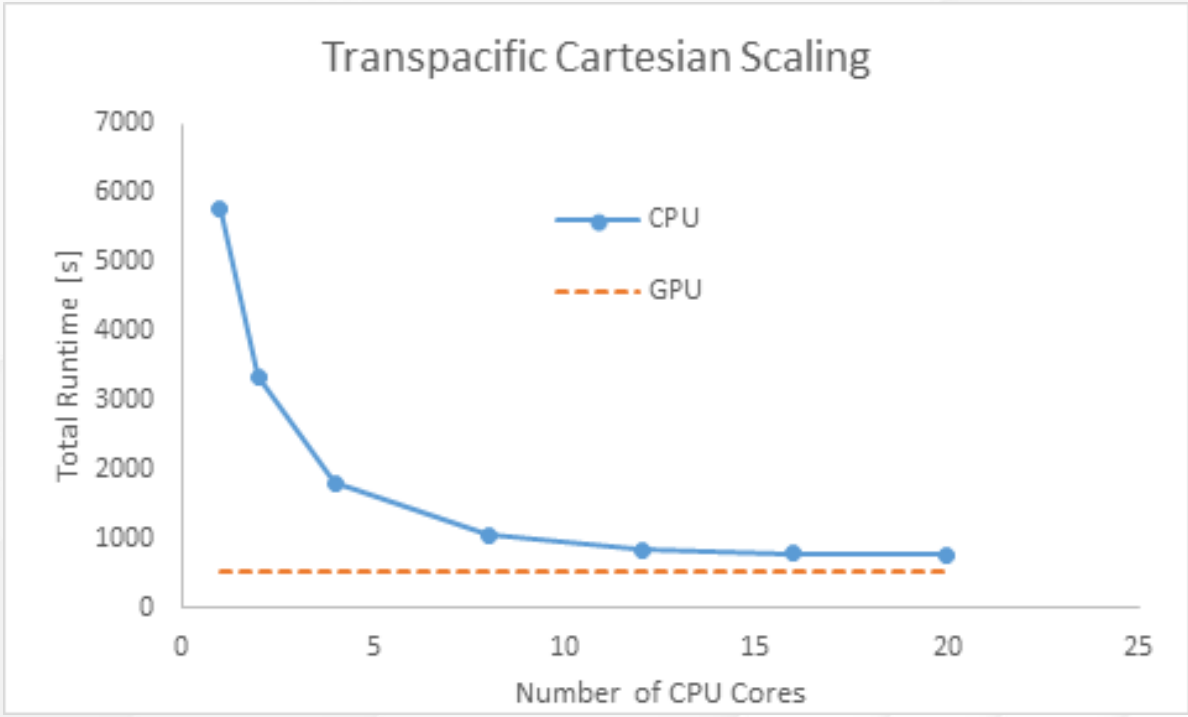
NIWA
Taihoro Nukurangi

Making Basilisk run on GPU

Basilisk meets Nesi - a cross-species romance?

Findings

- 4-6x Speed up on GPU (Tesla K40) compared with 4 core Haswell CPU
- Speed up increase with larger grid
- CPU code grid Adaptivity cost 40-60% of total runtime but can lead to speed up of 4-6 times



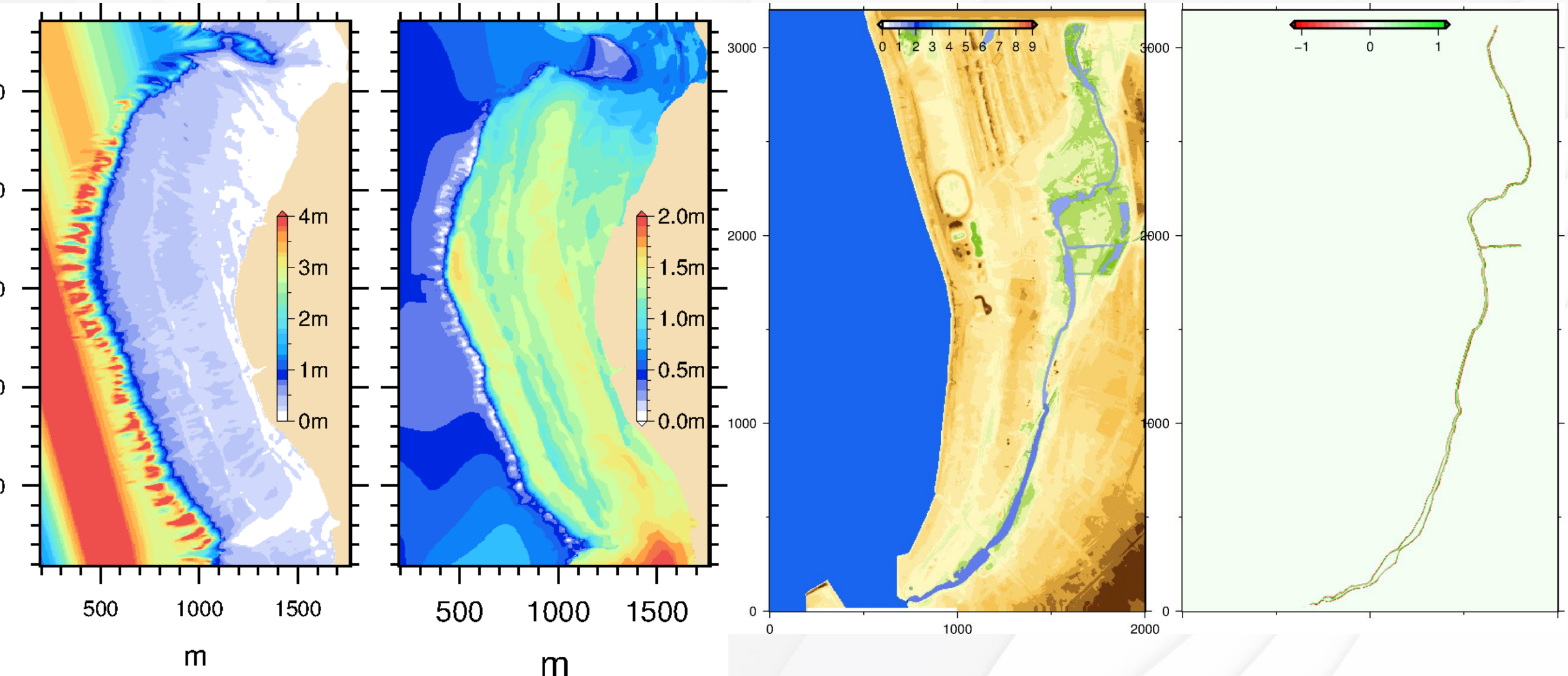
Limitation

- Limited functionality (e.g. no function pointers)
- Only the cartesian implementation so far
- Let the compiler optimize things

Good but pretty much a dead-end!

NIWA
Taihoro Nukurangi

I know ! Get Cyprien to do it



A Basilisk-like SWE solver for GPU

One Model to simulate all inundation related hazards:

- Tsunami
- Storm surge
- River
- Rainfall
- Waves

Open-source

Adaptive mesh (in development)

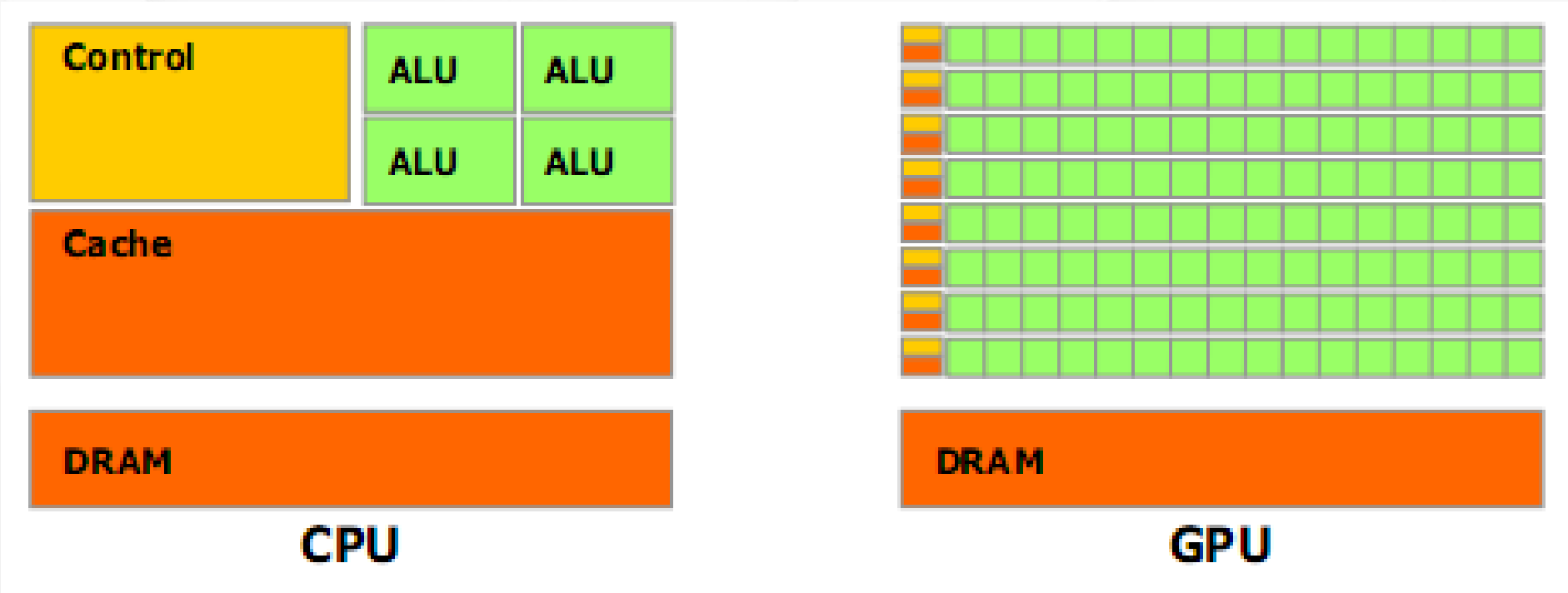
Dashing: Fast on the GPU

Lightweight : No frontend, minimal memory use

User friendly: easy to input forcing

Capable: concurrent hazards (e.g. storm surge + river + rainfall)

Memory model suitable for GPU



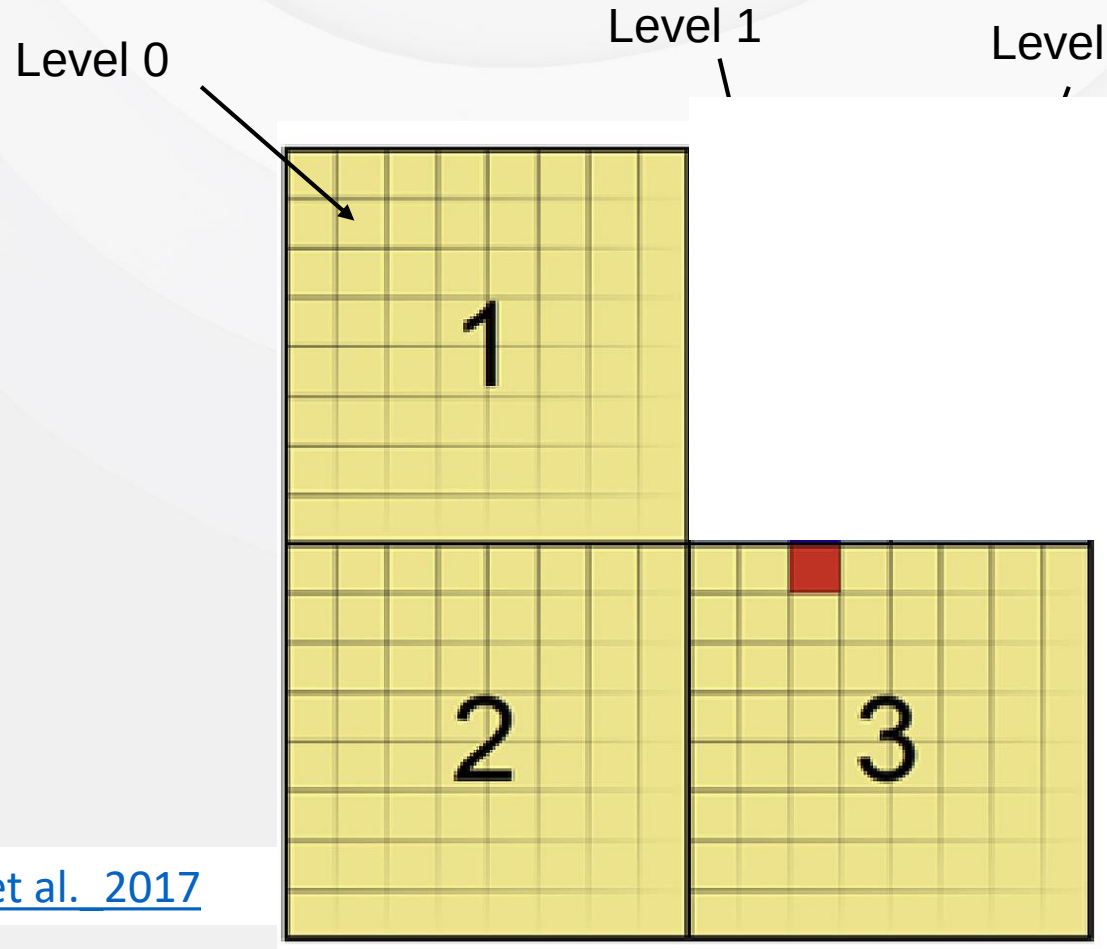
Few very fast, very independent cores

Many slow cores that can communicate fast but are interdependent

Block Uniform Quadtree (BUQ)

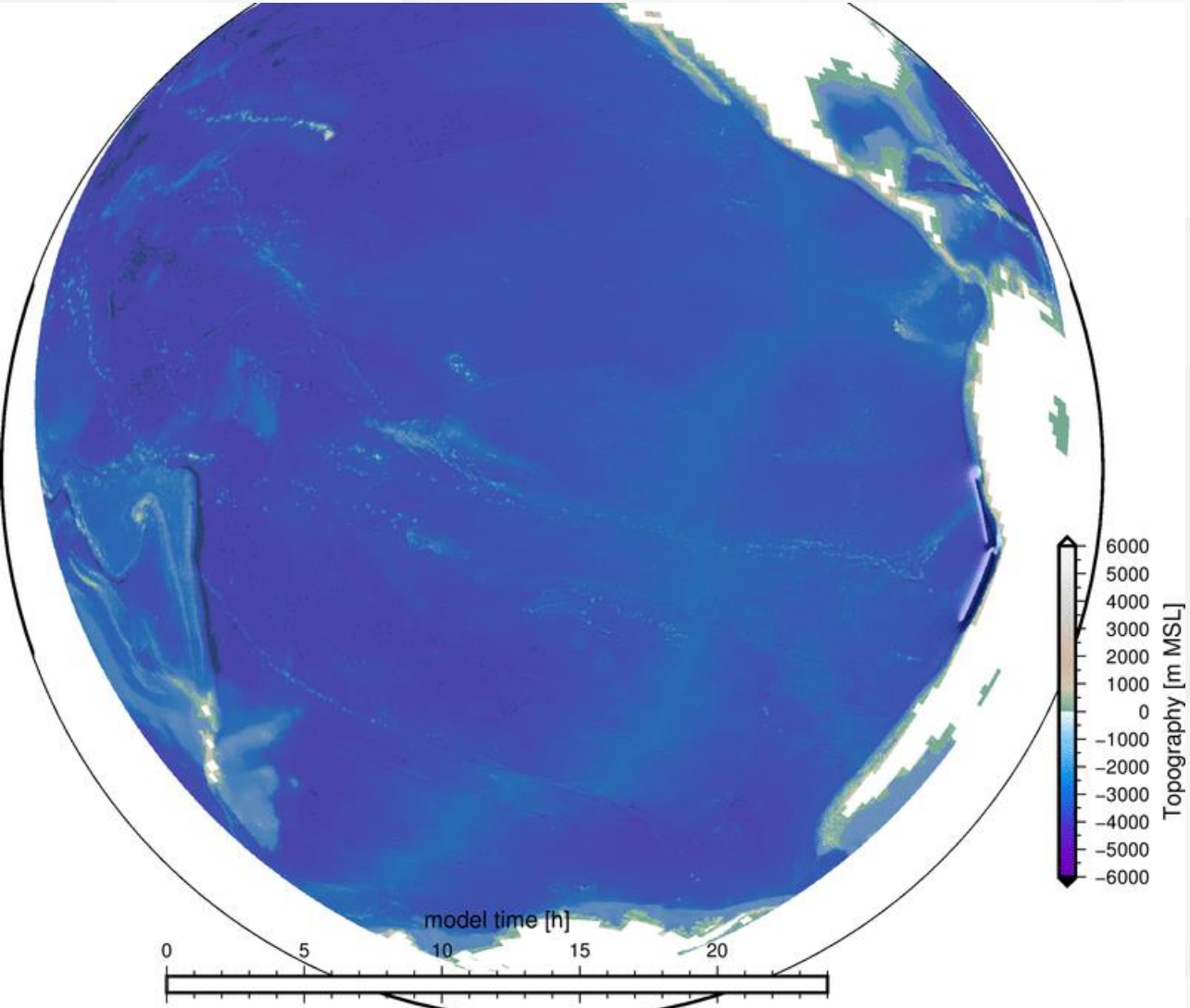
Memory model suitable for adaptive mesh refinement (AMR) on NVIDIA GPU

Mesh geographical layout



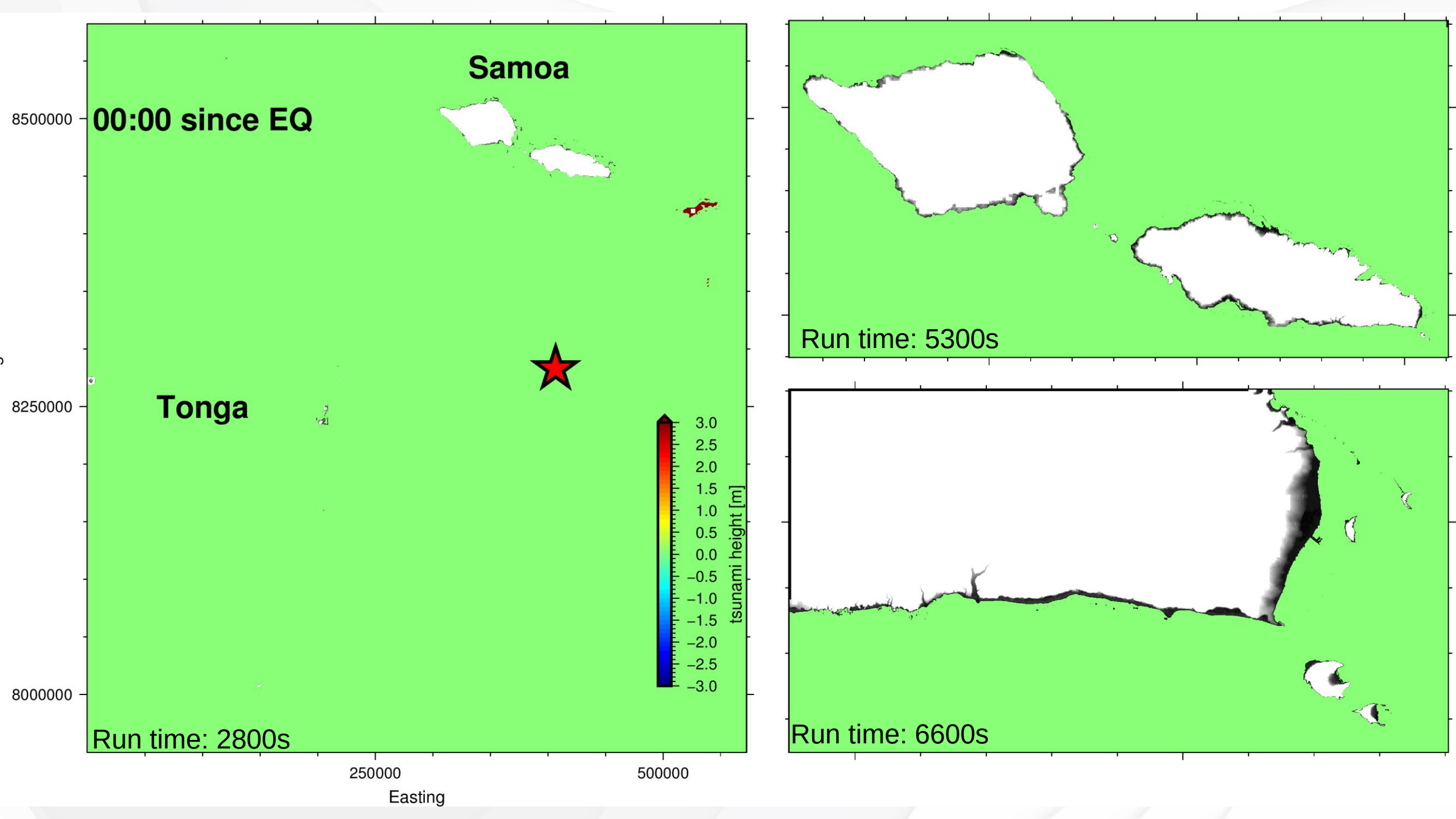
Vacondio _et al._2017

Tsunami propagation and inundation



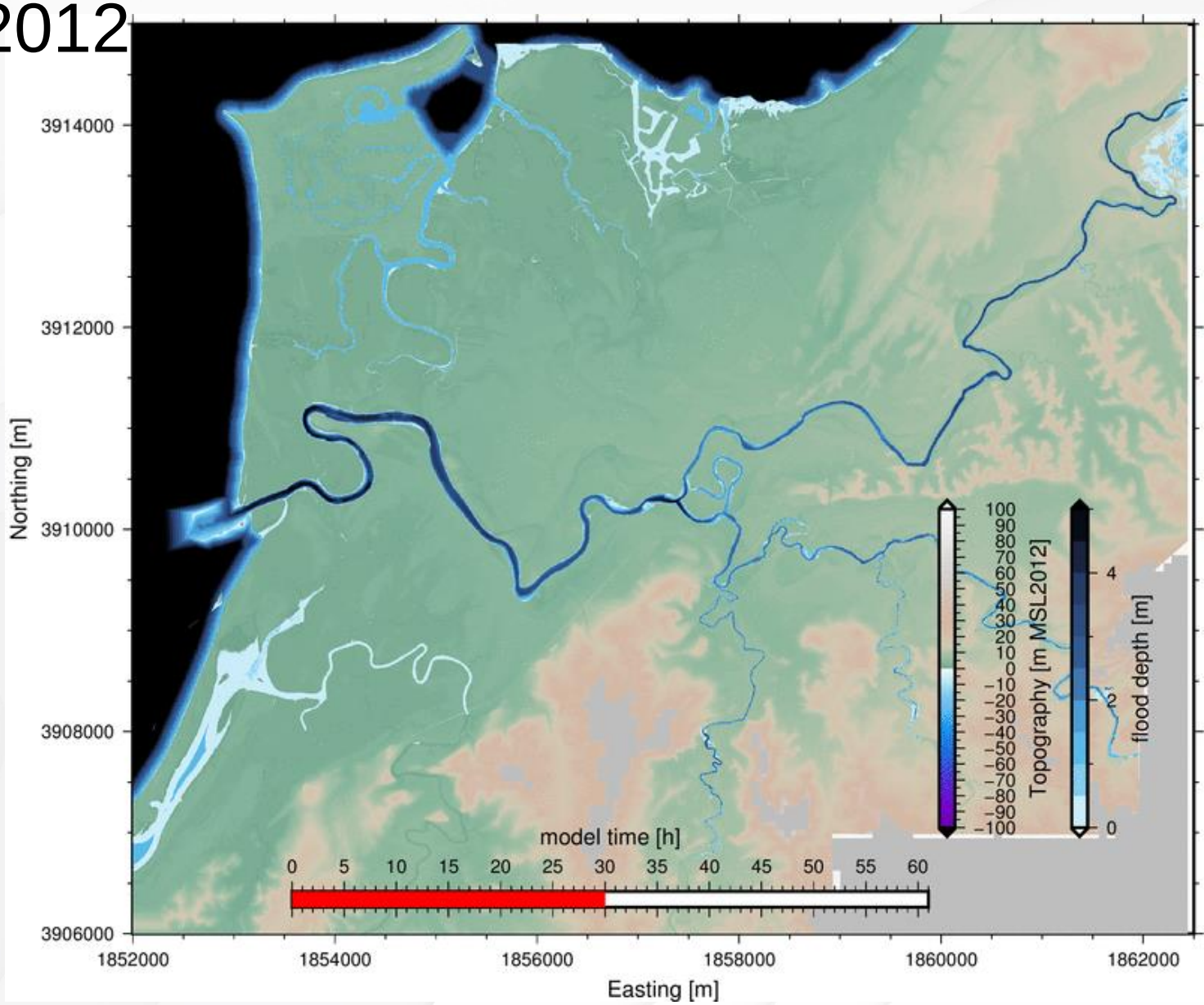
- Hypothetical scenario (1:2,500y ARI)
- ~7.6km res.
- Runtimes: 610s Tesla K40
  80s on Tesla P100
  Basilisk x4 Cores:11976s
  Basilisk GPU: 3700s

NIWA
Taihoro Nukurangi

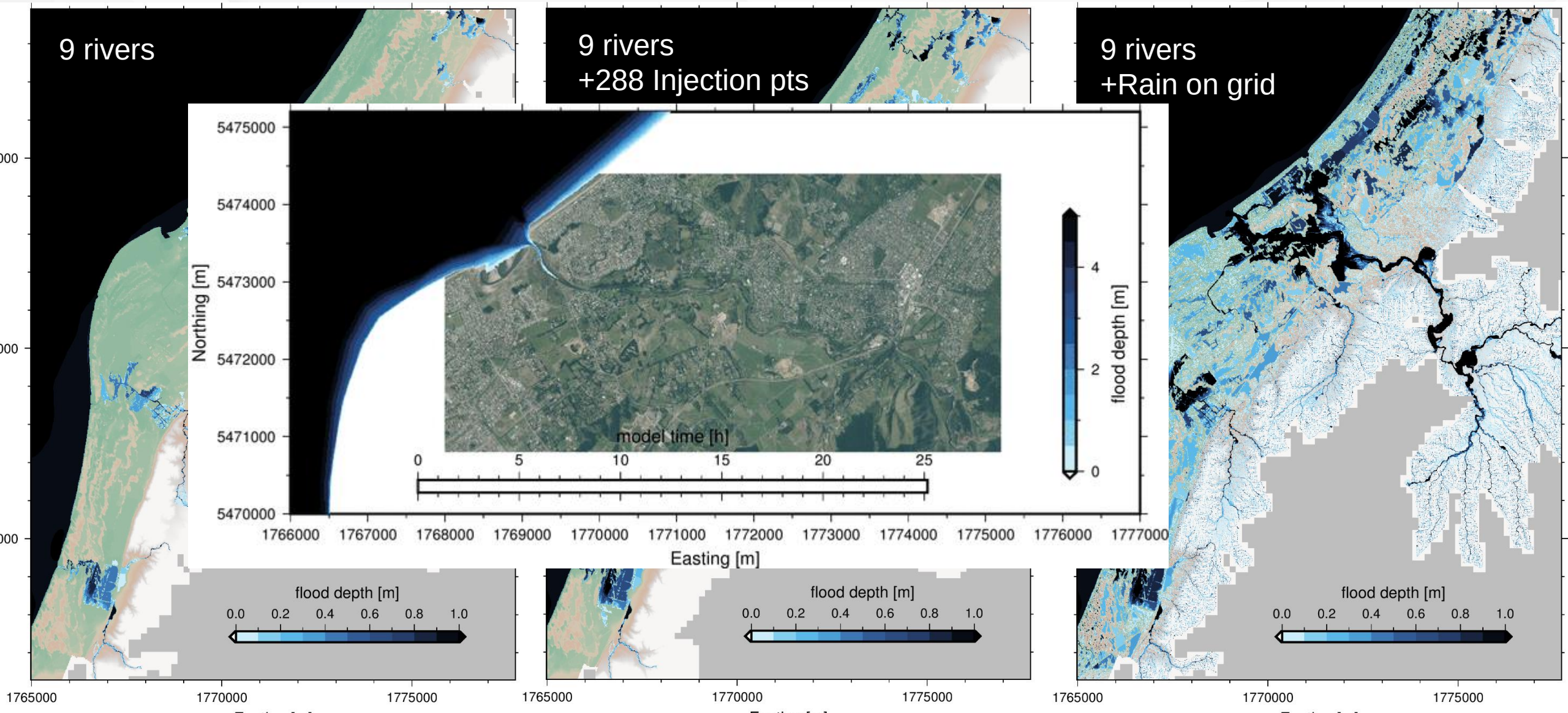

Samoa
00:00 since EQ
Tonga
Run time: 2800s
Run time: 5300s
Run time: 6600s
8500000
8250000
8000000
250000
500000
Easting
3.0
2.5
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
−3.0
tsunami height [m]

River inundation: Nadi 2012

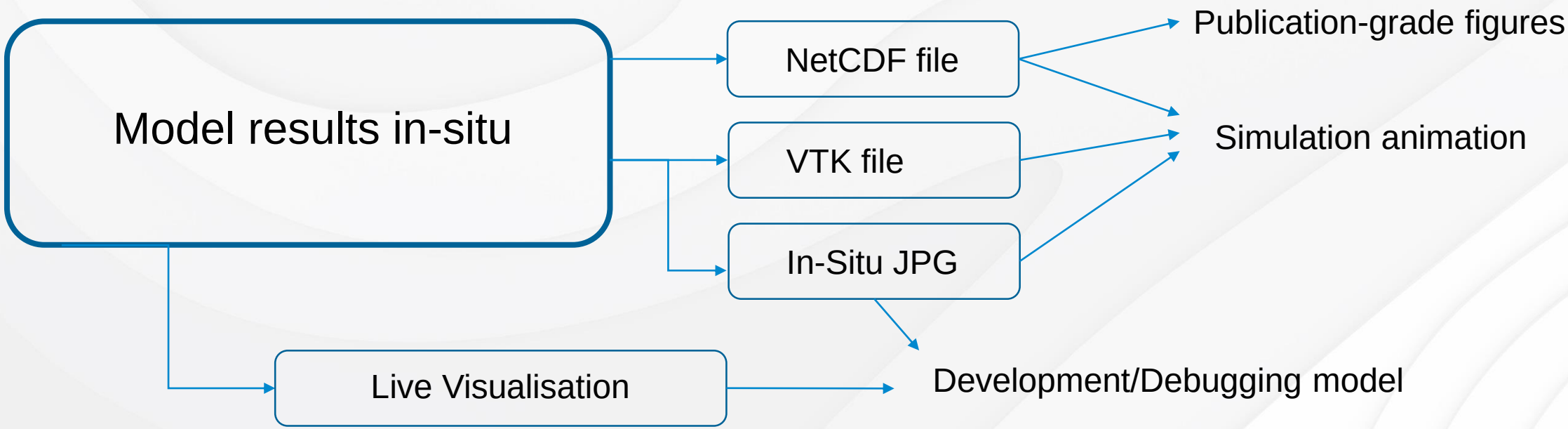
- 4 rivers + tide + storm surge
- 2048x2048; 5m res.
- Runtimes: ~3hrs

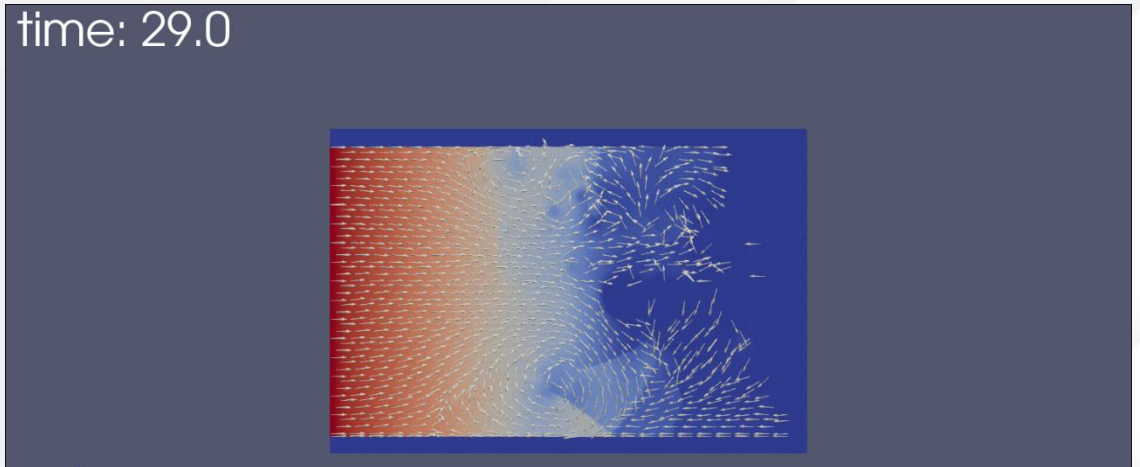


Tide+River+Rain inundation: Waikanae



BUQ visualisation: seeing the forest from the quadtree





NIWA
Taihoro Nukurangi

What's next

Have a go:

https://github.com/CyprienBosserelle/BG

7 Open ✓ 0 Closed Sort

1. Cartesian code for CPU Updated on Sep 17, 2018	First step. Need to improve and skim down this code to the bear bones.	···
2. Cartesian code on GPU Updated on Nov 28, 2018	Second step.	···
3. Add some user friendly I/O Updated on Apr 1	Cartesian code for GPU may be good for production if it has a decent user-friendly I/O. Netcdf outputs (with well described attributes) Netcdf bathy input asc bathy input (.md bathy input) text based input as in XBGPU Nice parameter sanity check	···
4. Add useful model physics Updated 6 days ago	Applies to both GPU and CPU models: _ Bnd conditions _ Wind, Atm forcing _ Bottom Frictions functions	···
5. non-adaptive Block uniform Cartesian Updated 1 minute ago	Implement the Block Uniform Quadtree on the cartesian grid	···
6. GPU/CPU Block Uniform Quad-tree Updated 6 days ago	Third step	···
7. GPU/CPU tree adaptive Updated on Jul 6, 2018	Implement the adaptivity of the Quad-tree based algorithm. It may be that the adaptivity is very costly to the GPU code but it needs to be tested.	···

Finish the multi-resolution BUQ (early 2020)

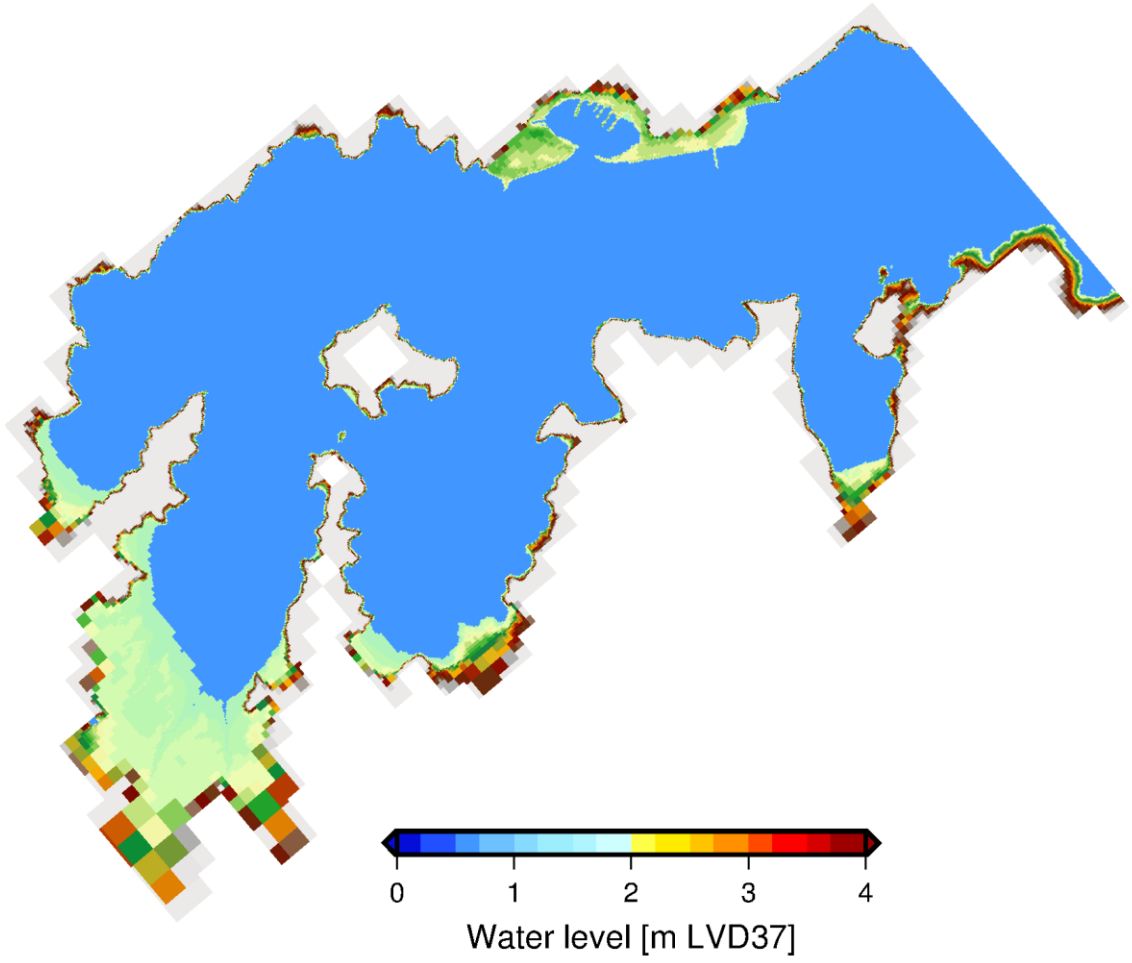
Finish the dynamic adaptive BUQ (mid-2020)

Waves next:

- 2G wave model?
- Boussinesq?

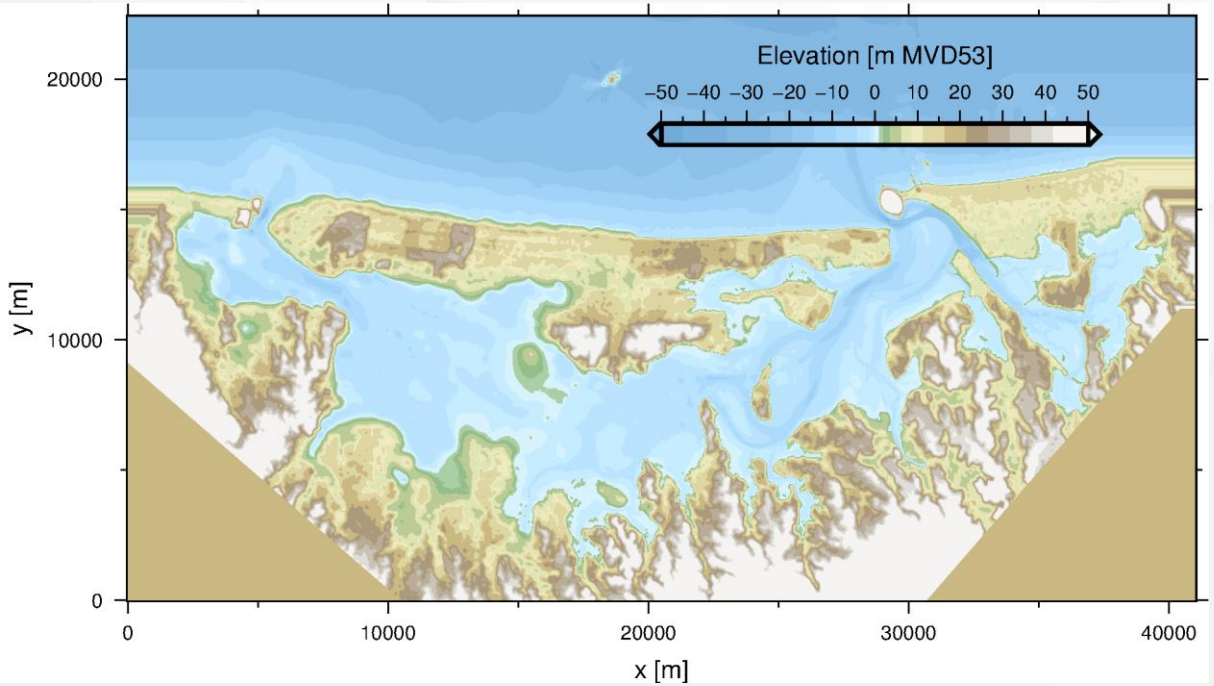
NIWA
Taihoro Nukurangi

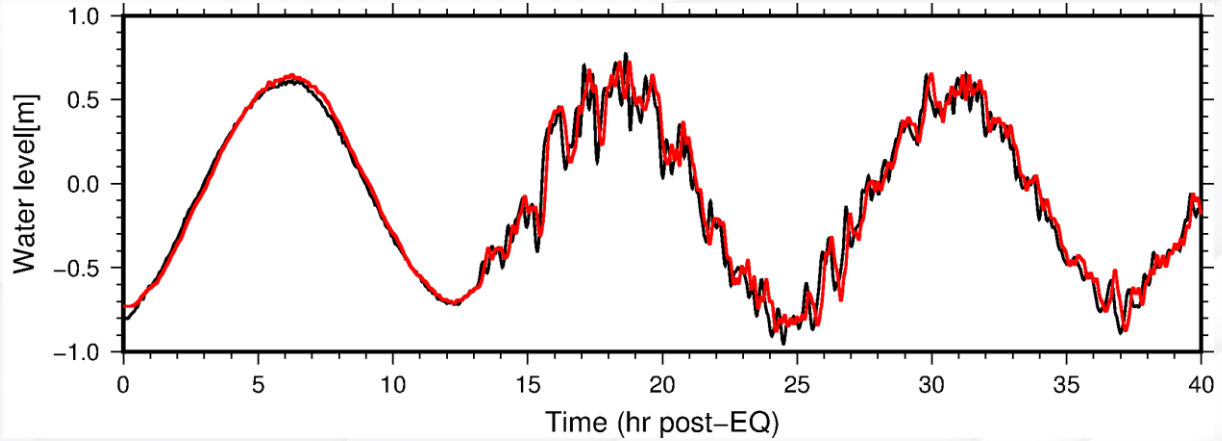
Good advection scheme to resolve eddies

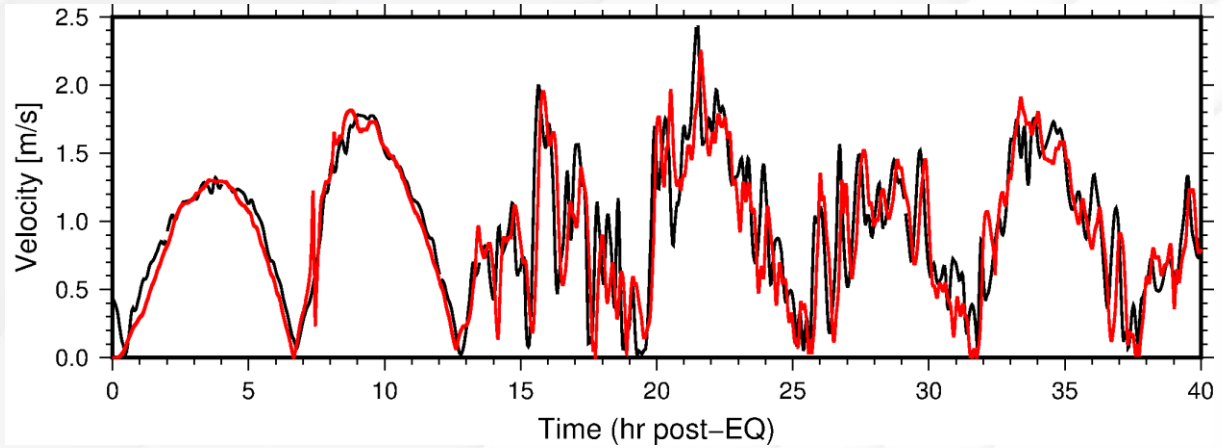


NIWA
Taihoro Nukurangi

Verification: Tauranga Hb Tohoku tsunami

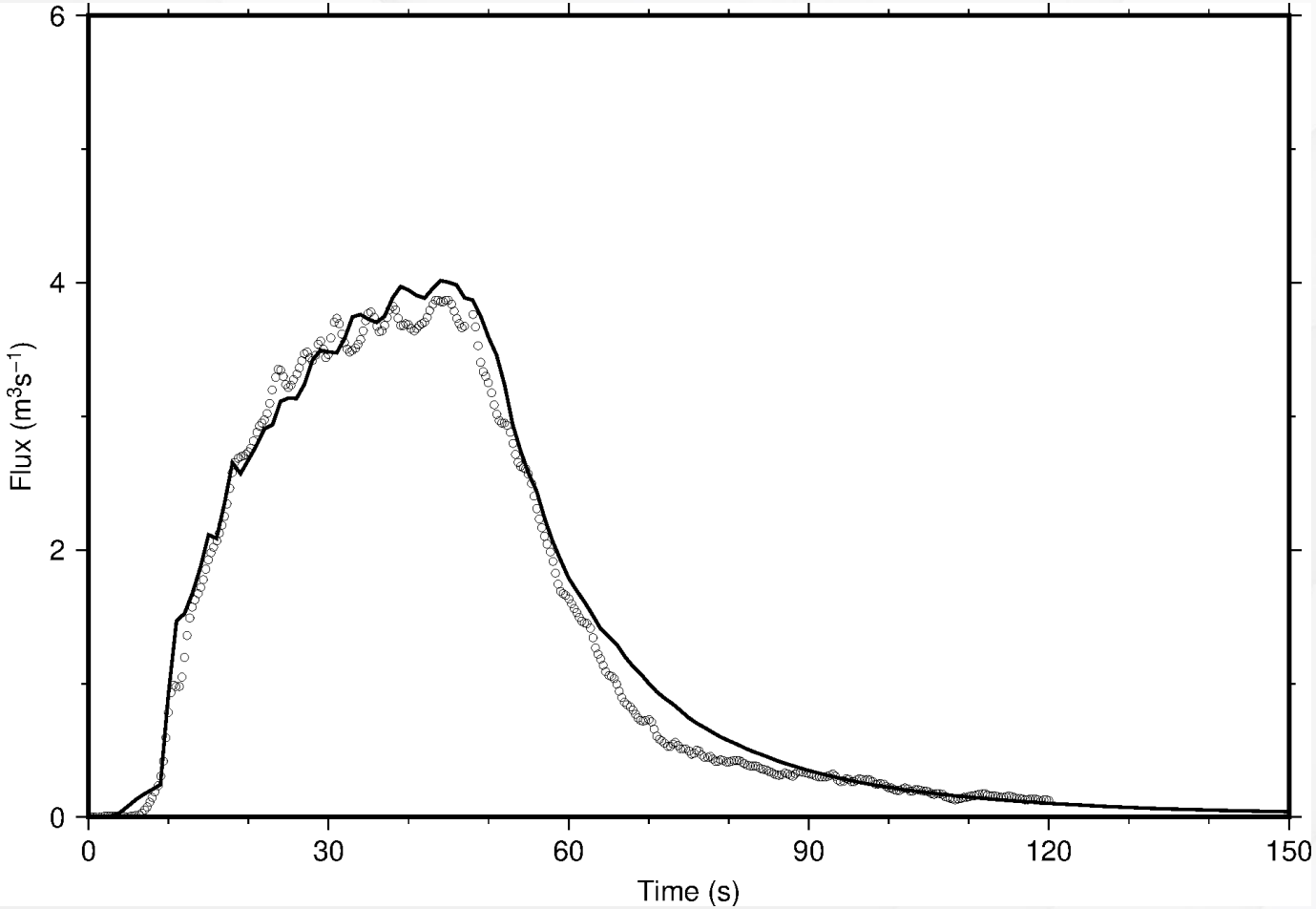


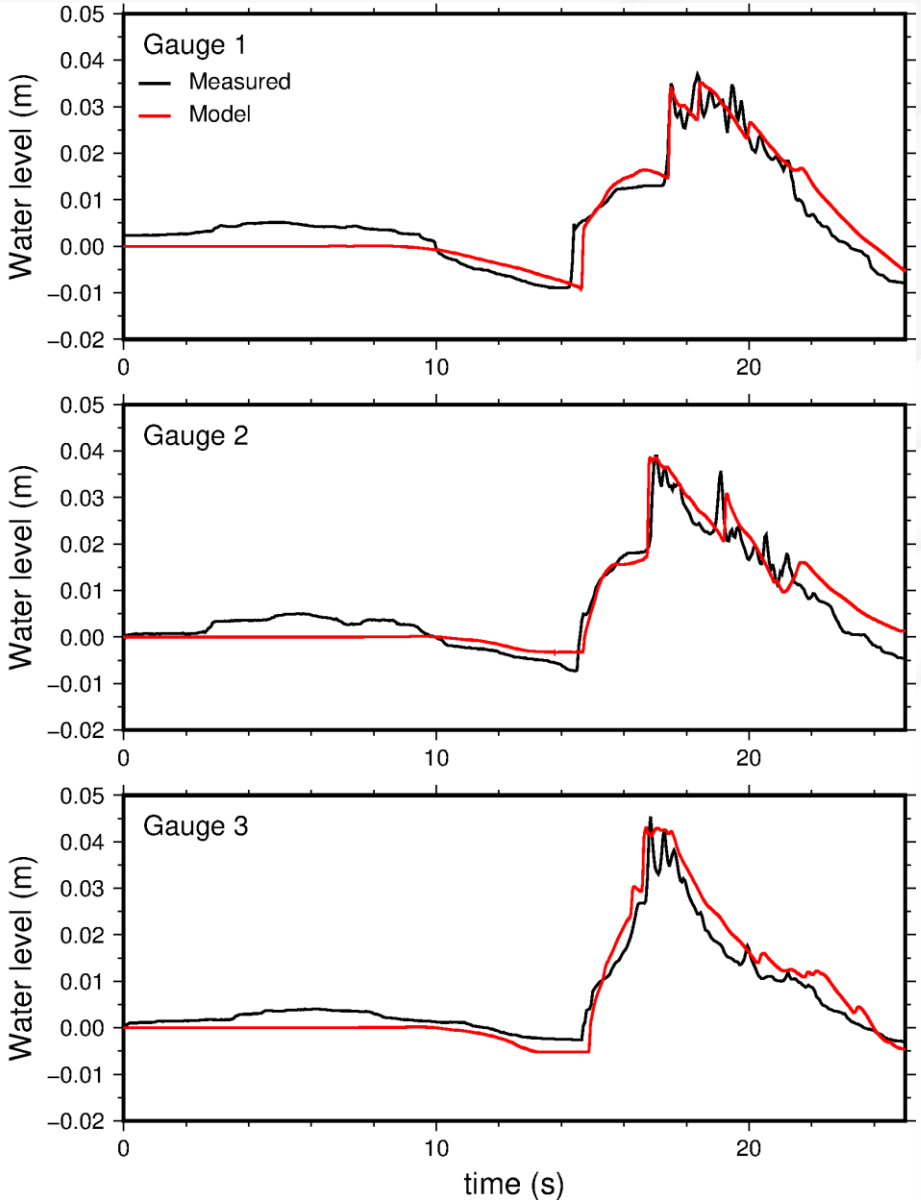




NIWA
Taihoro Nukurangi

Verification: ROG and Monai benchmark





Nga mihi

Cyprien.Bosserelle@niwa.co.nz