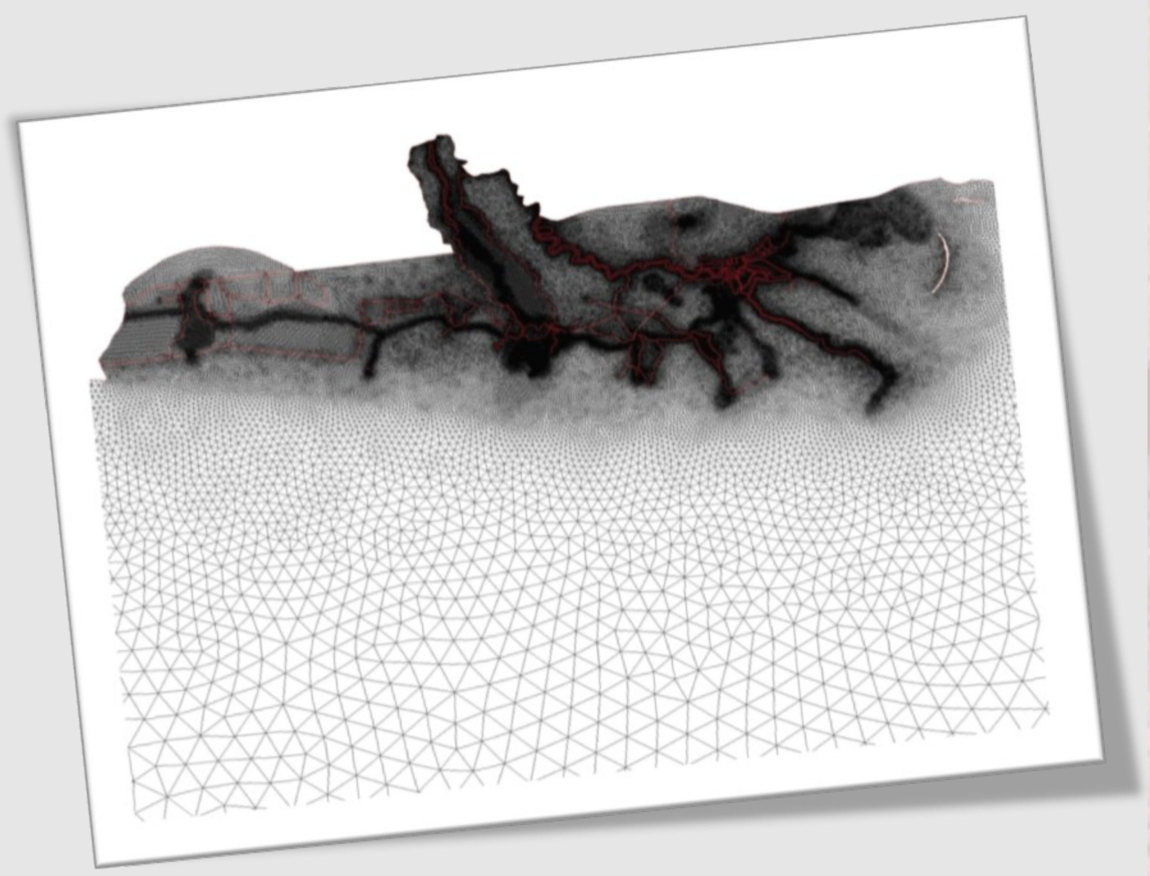
Towards Improving the Performance of ADCIRC Storm Surge Modeling Software

Nick Weidner
Department of Engineering and Computing
UNIVERSITY OF SOUTH CAROLINA

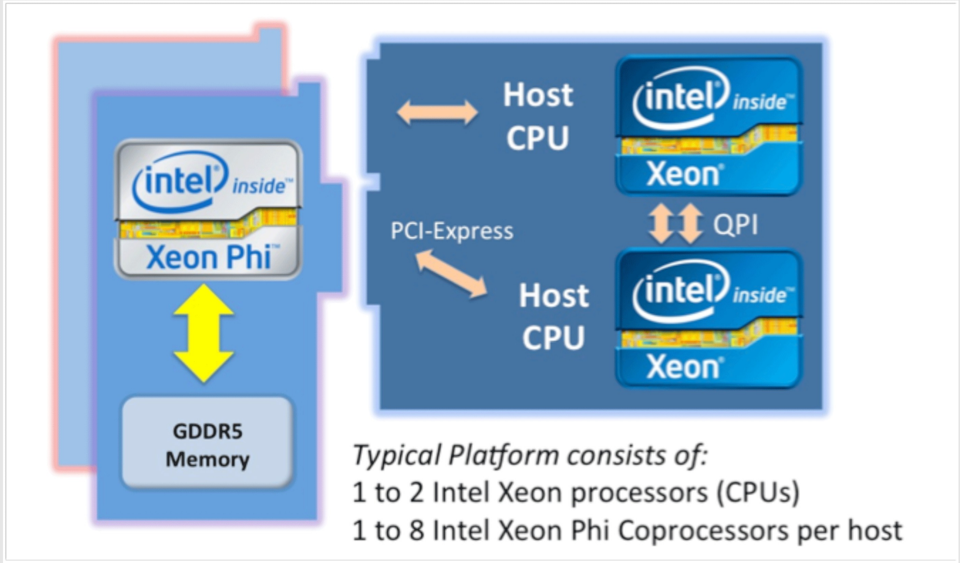
Background

Storm-Surge modeling is an incredibly demanding computational task that requires the use of powerful supercomputing resources. Modern day storm-surge modeling applications perform complex mathematical calculations on geographical areas using the Finite-Element mathematical method that represents the area using a grid of varying sized triangles. The smaller the triangle the better the resolution, but at the cost of more data points that take more time to compute. The figure bellow showcases a coast where the lighter sections cover open water, and the darker sections cover the actual coast where more data is computed.



Xeon Phi Co-Processor

• Research done on the Texas Advanced Copmuting Centers (TACC) Stampede Super Computer
•Each host processor is paired with an Intel Xeon Phi Co-Processor
•Host processors have 16 threads per
•Xeon-Phis have 240 per that are 2.5 times slower



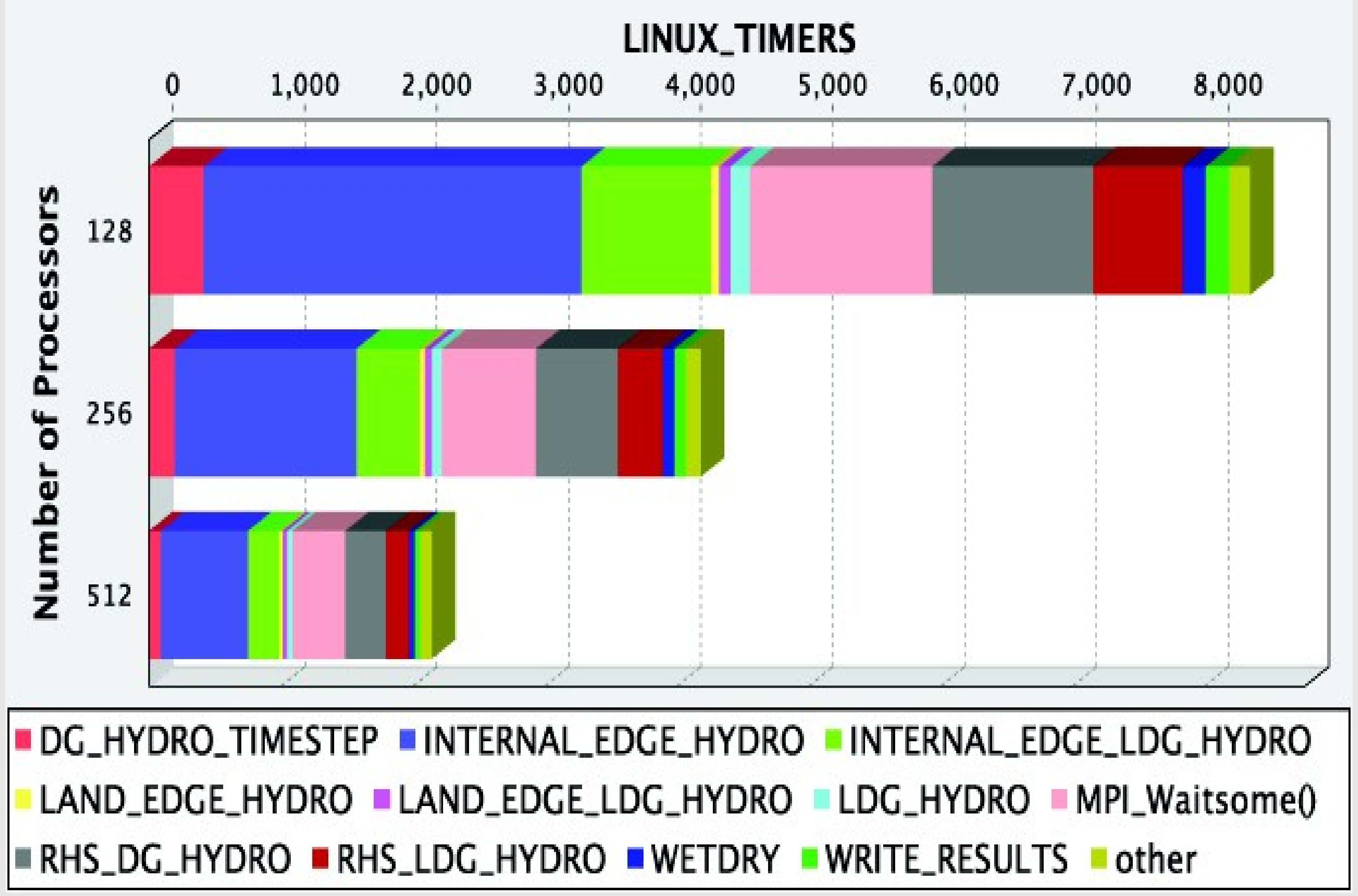
Typical Platform consists of:
1 to 2 Intel Xeon processors (CPUs)
1 to 8 Intel Xeon Phi Coprocessors per host

Objectives

•Identify computation "hotspots" on the host processor
•Implement OpenMP threading on expensive routines
•Increase speed by offloading threaded parallel work

Profiling

Figure below Showcases a profile run over 128, 256, and 512 cores. It is important to focus on the most expensive routines. The profiling used the Tuning and Analysis Utilities (TAU) at TACC



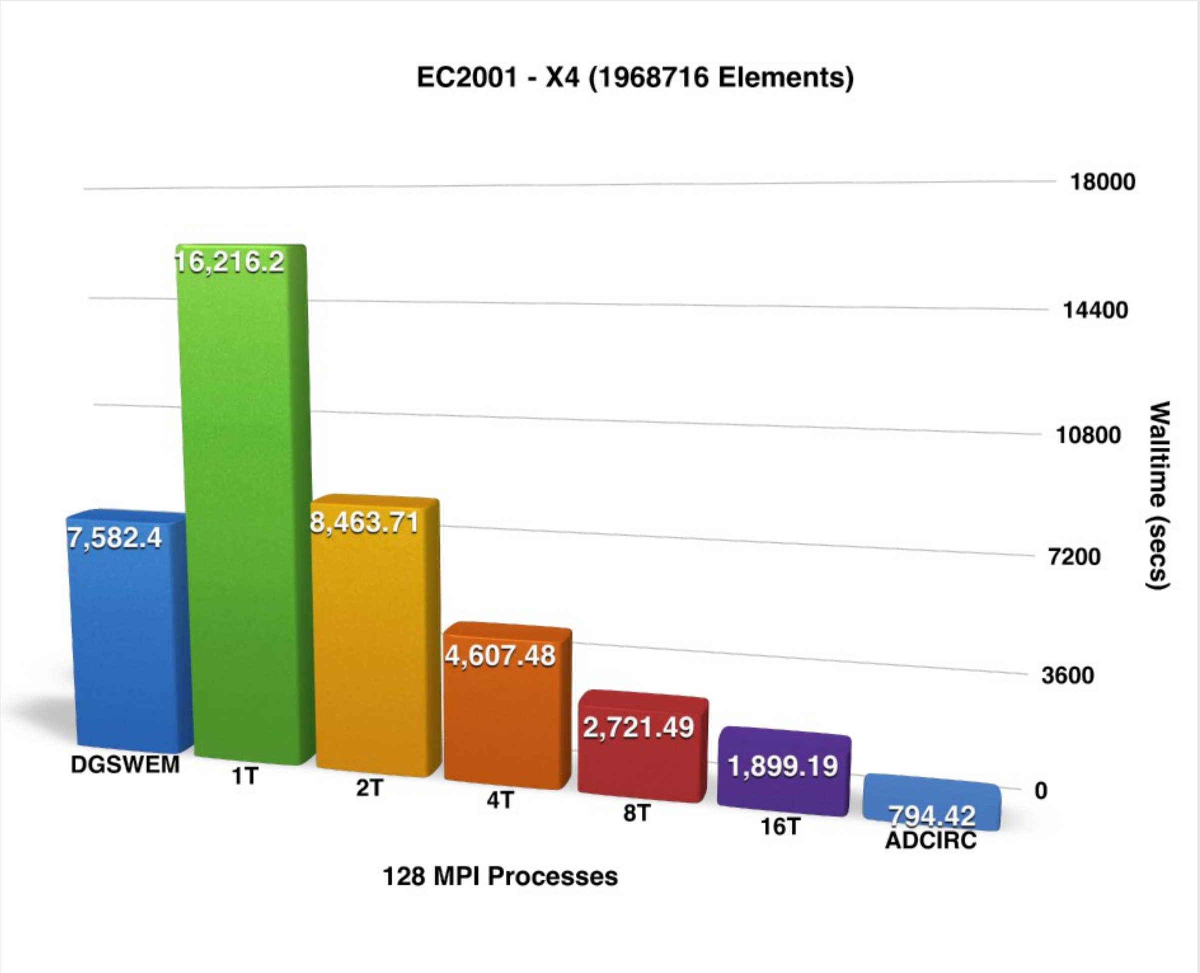
OpenMP & Offloading

OpenMP threading and offloading is achieved by wrapping directives around code regions. When execution arrives at these blocks, the work will be offloaded to the coprocessor. This requires identification of the global variables used which can result in extra computation time. The following is an example using OpenMP and offloading

```
!dir$ omp offload begin target(mic:0)
!$omp parallel do (VARIABLES)
   LOOP
!$omp end parallel do
!dir$ end offload
```

Multithreading

Originating from the Advanced Circulation (ADCIRC) storm surge model, DGSWEM uses a more accurate numerical model at the expense of speed. In order to increase DGSWEM's speed we try and take advantage of multithreading and the Xeon-Phis. The figure below showcases the introduction OpenMP and its performance increase up to the 16 thread level when compared to the original DGWEM and the old ADCIRC. The large spike in computation time at 1 thread is due to the OpenMP overhead and the need for atomic statements.



Current and Future Work

•Offloading the OpenMP code sections to the Xeon Phi
•Optimize code by directing the flow of necessary information between the co-processors and the host
•Compare the timings of Xeon-Phi offloading wit just OpenMP tests

Acknowledgments

•Tim Stitt, Jijun Tang, Steven Brus, Jarek Nabrzyski, Kallie O'Connell
Computational Hydraulics Lab, Center for Research Computing, University of Notre Dame
•National Science Foundation (NSF) Research Experience for Undergraduates (REU) program
•Magellan Apprentice Program