Addressing the scalability of Sea–ice model in CESM

John Dennis, Allison Baker (dennis,abaker@ucar.edu)  
National Center for Atmospheric Research  
Monika Luecke (m.luecke@grs-sim.de)  
German Research School for Simulation Science
High-resolution ASD simulation

- Opportunity to use large piece of newly installed “Yellowstone”
  - 1.5 Pflop peak, 72,288 cores
- Currently: 60+ years complete
- CAM5–SE (atmosphere model)
  - 28km resolution
  - Scalable spectral–element dynamical core
  - CAM5 physics
  - Fully prognostic aerosol (~50 tracers)
- POP (ocean model)
  - 11km resolution
  - 62 vertical levels
- CICE (sea–ice model)
  - 11km resolution
- CLM (land model)
  - 28km resolution
Computational Aspects of ASD simulation

General statistics:
- 2.0 SYPD
- 23,404 cores
- 1 TB of data generated per day
- 25.2 MCPU hours

Component configuration
- 11km Ocean model (6,124 cores)
- 11km Sea–ice model (16,295 cores)
- 28km Atmosphere (17,280 cores)
- 28km Land (900 cores)
- Coupler (10,800 cores)
Execution time for ASD simulation

- CAM (17280x1)
- POP (6124x1)
- CICE (16295x1)
- CLM (900x1)

11% total cost

Seconds per model day vs. # of cores

G8 Enabling Exascale Climate Simulation
Why are we not using more of Yellowstone?

- **System stability**
  - Significant performance loss (8x) in MPI_Allreduce on occasion (~0.5%)
  - Collective communication offload failures (33%)
- **Queue access for larger than 22k cores**
- **OS jitter sensitivity [CAM–SE, POP, CICE]**
- **Suboptimal CICE partitioning**
- **Suboptimal CPL scaling**
- **Infiniband routing table imbalances**
- **I/O overhead (6.4%)**
How to improve CICE partitioning?

Can we make the development of non-trivial partitioning algorithms easier?

(NCAR & GRS collaboration)
Sea-ice partitioning

land

open ocean

sea ice [prob > 0.5%]

G8 Enabling Exascale Climate Simulation
Goal: improve load-balancing for sea-ice model

How to assign blocks to cores?
- equalize amount of work per process
- minimize neighbors and communication volume

Code has flexible block structure:
- Small blocks
- Large blocks
- Multiple blocks per cores
- Eliminate land blocks
Large thin blocks?

Simple example

Advantages:
- match structure of load imbalance
- small number of neighbors

Disadvantages:
- poor surface to volume ratio
- limited parallelism
- limited land elimination
Variable number of small blocks

- Advantages:
  - allows land block elimination
  - potentially good surface to volume ratio
  - increases possible parallelism

- Disadvantages:
  - how to balance load?
  - non-contiguous blocks
  - number of neighbors

Better: (T. Craig)
Weighted space-filling curve

- **Need:** a cost model to estimate cost per process for a given partition

- **Issue:** the sea-ice code is complex (non-trivial to determine an analytical performance model)

**Strategy:**

- Assume a performance model based on code parameters and unknown coefficients
- Gather code timing data (e.g., Scalasca)
- Solve a L.S. problem to get cost model
- Integrate cost model into the code to guide partition choice
Total execution time for a sea-ice block depends on:

- $ocnj$ number of ocean points (in block $j$)
- $p_{\text{ice}j}$ number of sea-ice points (with a 0.5% probability of having ice)
- $bsx$, $bsy$ size of block in $x,y$ direction
- $\text{totVol}_j$ total communication volume
- $n\text{Neigh}_j$ number of neighbors*  (*not in model)

$$t_j = c1*ocnj + c2*p_{\text{ice}j} + c3*bsx*bsy + c4*\text{totVol}_j + c5*n\text{Neigh}_j$$
Sea–ice performance model

initial run => run statistics, timing => L.S => cost model

G8 Enabling Exascale Climate Simulation
Applying cost model

- Use the cost model to evaluate partitions during initialization

- Choose the “best” partition (iterative process)

  Remaining challenges:
  - estimate of communication may not be good enough
  - location of actual sea–ice is difficult for a “new” run
Applying cost model

![Graph showing average time for ICE component vs. number of cores]

- Default
- New
What went wrong with new CICE partitioning algorithm?
Decomposition: CICE on 3600 cores [Scalasca]

Bigger domains in open ocean

Small domains where sea-ice is present
EVP: Total time

65 sec

96 sec

38 sec
EVP: execution time

Flop intensive
EVP: MPI time

High communication cost
“Yellowstone” has enabled very large scale simulation capability at NCAR
  ◦ Laboratory for future larger simulations

Challenging issues still remain for large scale simulations

Leveraging G8 collaboration
  ◦ Scalasca timing for L.S.
  ◦ Scalasca topology visualization enables insights
Questions?

dennis@ucar.edu
Execution time for ASD on Yellowstone

Large variability in execution time

Execution time for non-I/O CESM day

Date

G8 Enabling Exascale Climate Simulation
Write bandwidth for ASD simulation on Yellowstone

Write Bandwidth for CESM I/O day [Yellowstone]

- 6.4% overhead
- 21% overhead
Advanced Scientific Discovery (ASD) project

- Opportunity to use a large piece of newly installed Yellowstone
- “Meso- to planetary-scale processes in a global ultra-high resolution model”
- R. Justin Small, Bailey, Bryan, Danabasogla, Holland, Jochum, Lawrence, Park, Peacock, Tomas, Tribbia, Dennis (NCAR), Saravanan (Texas A&M), Schneider (Hawaii), Kwon (WHOI)
- 47.1 M core hours
  - 25.2 M (2 months) [tuning]
  - 21.9 M (18 months)
Felix Wolf (f.wolf@fz-juelich.de)
Scalasca (JSC)
Research questions:

- How do you make non-trivial partitioning algorithm development easier? [Monika Luecke]
  - CICE partitioning algorithm: balance communication/computational imbalance
  - Simulated annealing for IFS model
- Can you identify scaling bottlenecks on large core counts through performance prediction? [Alexandru Calototolu]
Use traces to identify on-node performance problem
Identified 400 μsec OS jitter affect on Yellowstone
Identified subtle CPU resource issues -> code restructuring to enable better cache utilization
Acknowledgements

- NCAR:
  - D. Bailey
  - F. Bryan
  - T. Craig
  - B. Eaton
  - J. Edwards [IBM]
  - N. Hearn
  - K. Lindsay
  - N. Norton
  - M. Vertenstein
- COLA:
  - J. Kinter
  - C. Stan
- U. Miami
  - B. Kirtman
- U.C. Berkeley
  - W. Collins
  - K. Yelick (NERSC)
- U. Washington
  - C. Bitz
- NICS:
  - M. Fahey
  - P. Kovatch
- ANL:
  - R. Jacob
  - R. Loy
- LANL:
  - E. Hunke
  - P. Jones
  - M. Maltrud
- LLNL
  - D. Bader
  - D. Ivanova
  - J. McLean (Scripps)
  - A. Mirin
- ORNL:
  - P. Worley

- Grant Support:
  - DOE
    - DE–FC03–97ER62402 [SciDAC]
    - DE–PS02–07ER07–06 [SciDAC]
  - NSF
    - Cooperative Grant NSF01
    - OCI–0749206 [PetaApps]
    - CNS–0421498
    - CNS–0420873
    - CNS–0420985
- Computer Allocations:
  - TeraGrid TRAC @ NICS
  - DOE INCITE @ NERSC
  - LLNL Grand Challenge
- Thanks for Assistance:
  - Cray, NICS, and NERSC
  - and many more…
What is CESM?

- Consists of a set of 4->6 geo-components
  - ATM, OCN, LND, CICE, GLC, WAVE
  - Run on potentially different grids
  - Exchange boundary data with each other via coupler
    - hub and spoke architecture

- Large code base: >1.3M lines
  - Fortran 90 (mostly)
  - Developed over 20+ years
  - 200-300K lines are critically important
    - Communication, not computational kernels

- CESM is an interdisciplinary collaborative effort
  - DOE, NSF, NOAA, University Community
  - Applied Math, CS, software engineering, climate scientists
Space-filling curve
20 processes

Large domains @ low latitudes

Small domains @ high latitudes

G8 Enabling Exascale Climate Simulation
NWSC-Yellowstone is now operational

- **Compute Nodes**
  - Processor: 2.6 GHz Intel Sandy Bridge EP processors
  - Node: dual socket; 32 GB memory; 2 GB/core
  - 4,518 nodes, 72,288 cores total – **1.5 PFLOPs peak**
  - 144.6 TB total memory

- **High-Performance Interconnect**
  - Mellanox FDR InfiniBand full fat-tree
  - 13.6 GB/sec bidirectional bw/node
  - <2.5 usec latency (worst case)

- **Central File System**
  - 2012: 11 PB
  - 2014: 16.4 PB
  - Bandwidth: 90 GB/sec