Optimization of CAM with Finite-Volume Dynamical Core on the Cray X1

Arthur A. Mirin
Lawrence Livermore National Laboratory
and
Patrick H. Worley
Oak Ridge National Laboratory

FV/CAM uses a hybrid parallel model

- Multi-two-dimensional domain decomposition
  - latitude-vertical (yz) 2-D domain decomposition for most of dynamics
  - longitude-latitude (xy) 2-D domain decomposition for flux surface remapping and geopotential calc.
- Physics partitions horizontal domain into chunks
  - optionally equivalent to lon-lat decomposition
- Shared memory parallelism (*OpenMP*) largely in vertical, but also in latitude
- Decompositions connected by high-speed transposes using *Pilgrim* and *Mod_comm* libraries (*NASA/GSFC*)
  - *MPI* derived types
  - *MPI-2* one-sided communication

The Cray X1 combines massively parallel and vector computing capabilities

- Memory is globally addressable (NUMA)
- Each node has 4 multi-streaming processors (MSPs) and 16 GB memory
  - MSP peak speed is 12.8 GF (64-bit)
  - MSP has 2 MB cache
- Each MSP has 4 single-streaming processors (SSPs)
- Each SSP contains a superscalar processor and a two-pipe vector processor
- MPI tasks can be assigned to MSPs (default) or SSPs; OpenMP, SHMEM, Co-Array Fortran are supported
- Each node contains 32 communication network ports, each supporting 1.6 GB/s peak per direction

Improvements to flux surface remapping

- Original procedure had 4 levels of do-loops and many conditionals
- Determine interpolation levels in advance and pre-store, enabling main computation to be written in terms of relatively-clean vectorized loops
  - *main computation involves only 2 levels of do-loops and is almost conditional-free*
  - *order of loop indices for determining interpolation levels critical to performance*
- *Results in 10 to 20 fold improvement of remapping phase*

Improvements to main dynamical time advance

- *Logical tests (very often nested) are pervasive due to unwind differencing and accuracy choice*
  - conditionals associated with upwind differencing are not vector-friendly
- Evaluate conditionals in advance; use indirect indexing based on (multiple) conditionals
- Enhance vectorization/streaming possibilities by maintaining latitude loop throughout calling chain
- Add outer-loop streaming directives
- Results in 50-100% improvement of this phase
- Maintaining dual version of *tp_core* module since these changes degrade IBM performance

Polar filtering (*pft2d*) does not vectorize well

- Filtering uses vectorized FFTs, where the vectorization is with respect to the FFTs being performed in a single call of the FFT routine
- The number of such simultaneous FFTs is bounded by the number of latitude lines in a subdomain
- One can combine FFTs at different vertical levels, but this will gain only marginally, if at all
  - CAM climate scenarios have only 26 levels
  - the vertical domain is partitioned through domain decomposition and (possibly) OpenMP
  - a typical thread is responsible for only a few levels
- Libsci FFTs (which are faster) are presently not compatible with outer-loop streaming directives

Intermachine comparison for CAM-FV at 1°x1.25°x26L

Decomposition (lat-vert)	Processor count	Cray X1 Phoenix	IA64 Thunder	IBM PWR3 Frost
4x4	16	73	306	800
16x4	64	23	84	237
48x4	192	11	33	104

(seconds per simulated day)

Thunder shows the best scaling

Cray X1 shows the poorest scaling

Tuning parameters for Cray X1

- Superior performance (of physics) for pcols>=256
  - dependent on problem size and domain decomposition
  - IBM typically uses pcols~16
- Use of MPI derived types (versus contiguous buffers) significantly degrades performance
- Physics load-balancing (*phys_loadbalance*=2) is somewhat beneficial at low processor count but has little effect at high processor count
- Weak dependence on geopotential method

Planned improvements for Cray X1

- Optimize polar filtering
  - no plan on how to proceed at this time
- Optimize communication
  - test MPI-2 one-sided communication
  - implement *Shmem* constructs
    - already supported for border communication
    - not yet implemented for transposes
  - Add MPI_Alltoall communication option for transposes
  - add CoArray Fortran for border communication
- Test OpenMP
  - code presently fails with OpenMP
  - compatibility of multistreaming and OpenMP?

Acknowledgment

- This is LLNL Report UCRL-PRES-212875

- Work performed under the auspices of the U.S. Department of Energy by University of California Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48. (Mirin)

- Work sponsored by the Office of Mathematical, Information and Computational Sciences, Office of Science, U.S. Department of Energy under Contract No. DE-AC05-00OR22725 with UT-Batelle, LLC. (Worley)