SAVE PRINT

LA-UR-04-2991

Approved for public release; distribution is unlimited.

Title: Parallel Monte Carlo Transport

Author(s): Steven D. Nolen, X-3
Terry R. Adams, CCN-12

Submitted to: Workshop on Parallel Transport
Texas A&M University
April 30-May 1, 2004

SAVE PRINT CLEAR FORM

Los Alamos
NATIONAL LABORATORY

Abstract

Parallel Monte Carlo transport has been attempted and achieved with a variety of results over the past 20 years at Los Alamos. Recent efforts have focused on the advantages found by replicating the problem geometry on each of the processors and then allowing each to track some set of particles with the results accumulated as each processor finishes. MCNP is one such code that uses this approach. Unfortunately, this has a significant drawback as the problem geometry must completely fit on each of the processors. Previous work on the MC++ code was directed at following the particles across a geometry that was decomposed across the available processors. Although this allowed for arbitrarily sized problems, there was a significant amount of communications required to follow the particles. Recent work on MC++ has been to merge these two major parallel efforts and provide a hybrid. The results of this effort and it progress are the topic of this presentation.

Parallel Monte Carlo Transport at Los Alamos

Steve Nolen

Terry Adams

Parallel Issues for Monte Carlo

- Decomposition
  - Particles (increased memory)
  - Problem geometry (increased communication)
- Communication
  - Insuring mesh/particle information locality
  - Synchronization for particle population
  - Statistics and tallies

MC++ Transport

- Get neutrons from source description
- Remove source bias*
- Begin transport loop
- Collect statistics affecting next transport loop

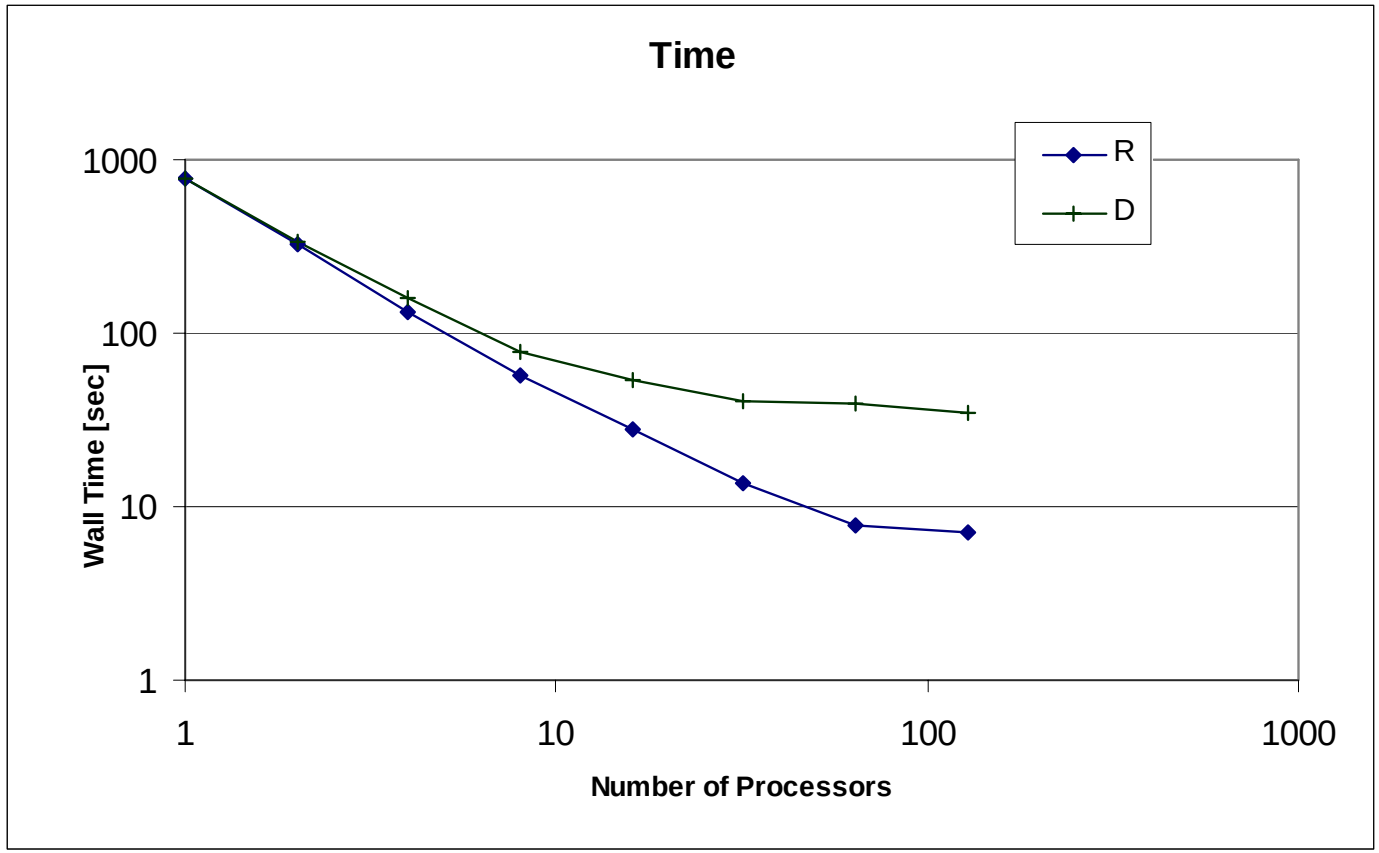
Transport Loop

- Get cross sections
- Find next event
  - Boundary collision with mesh
  - Material boundary collision
  - “Real” collision resulting in new particle state
- Push particles to new position (new host?)
- Update particle state
- Repeat until population is depleted

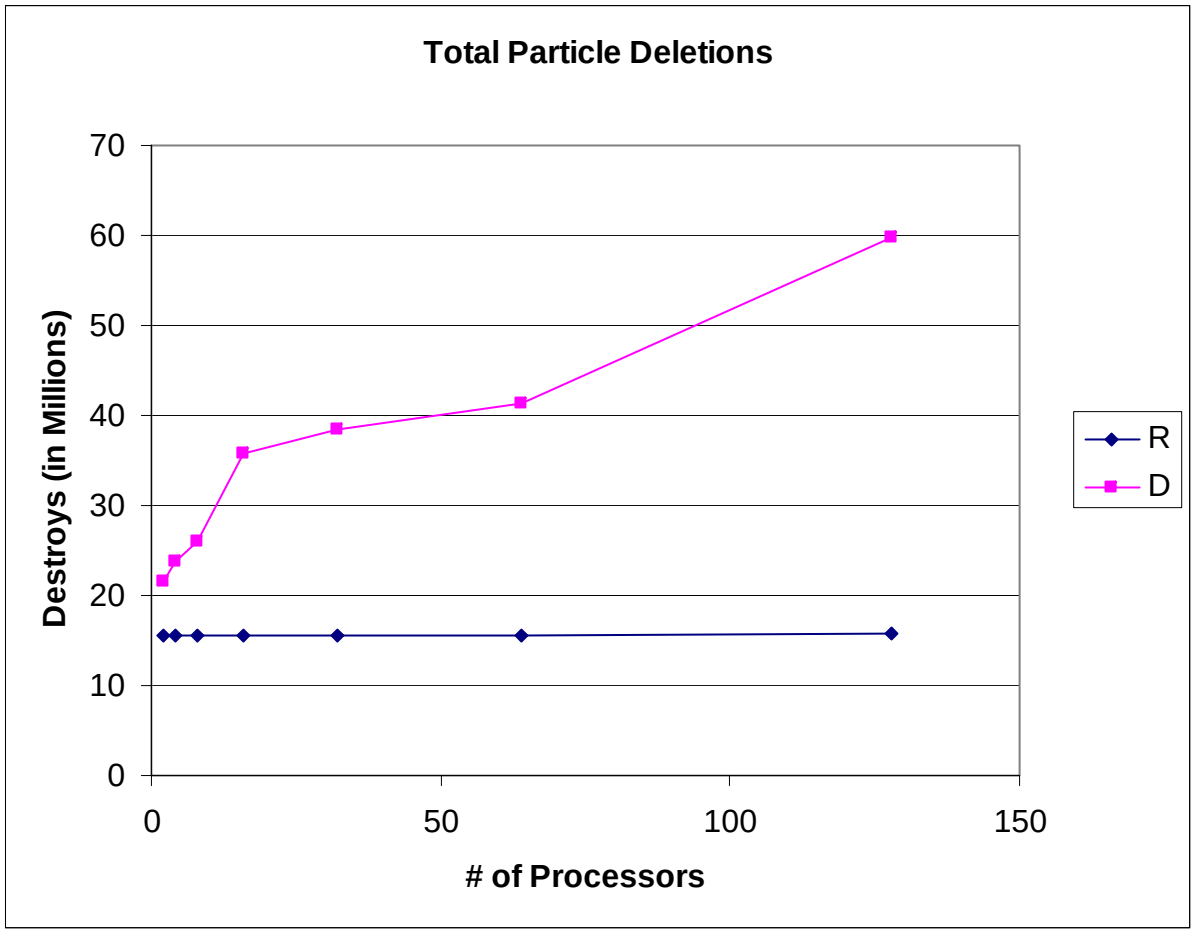
Decomposition vs. Full Replication

- Domain decomposed
  - Memory postulated to be limiting factor for large meshes
  - Synchronization intensive – particle swapping
  - Memory “thrashing” potential – particle swapping
  - Readily available for nonlinear applications
- The “trivially” parallel approach
  - Synchronization within transport loop removed
  - Bookkeeping costs for setup
  - Master processor approach of MCNP

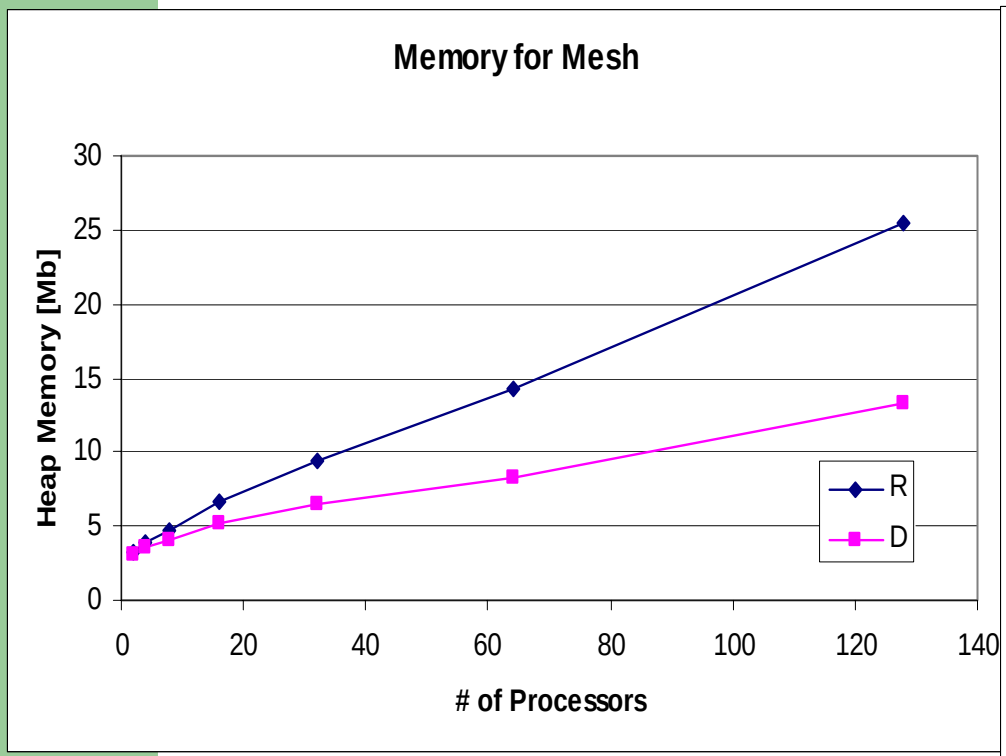
Full Replication vs. Decomposition Results

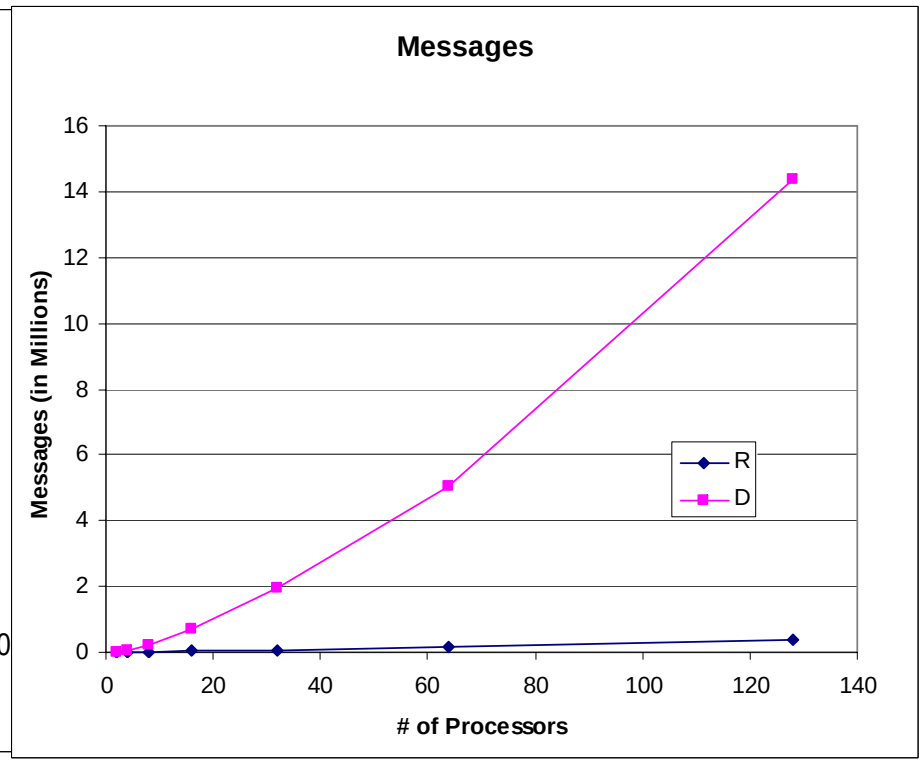


Full Replication vs. Decomposed Results (cont.)



Full Replication vs. Decomposed Results (cont.)

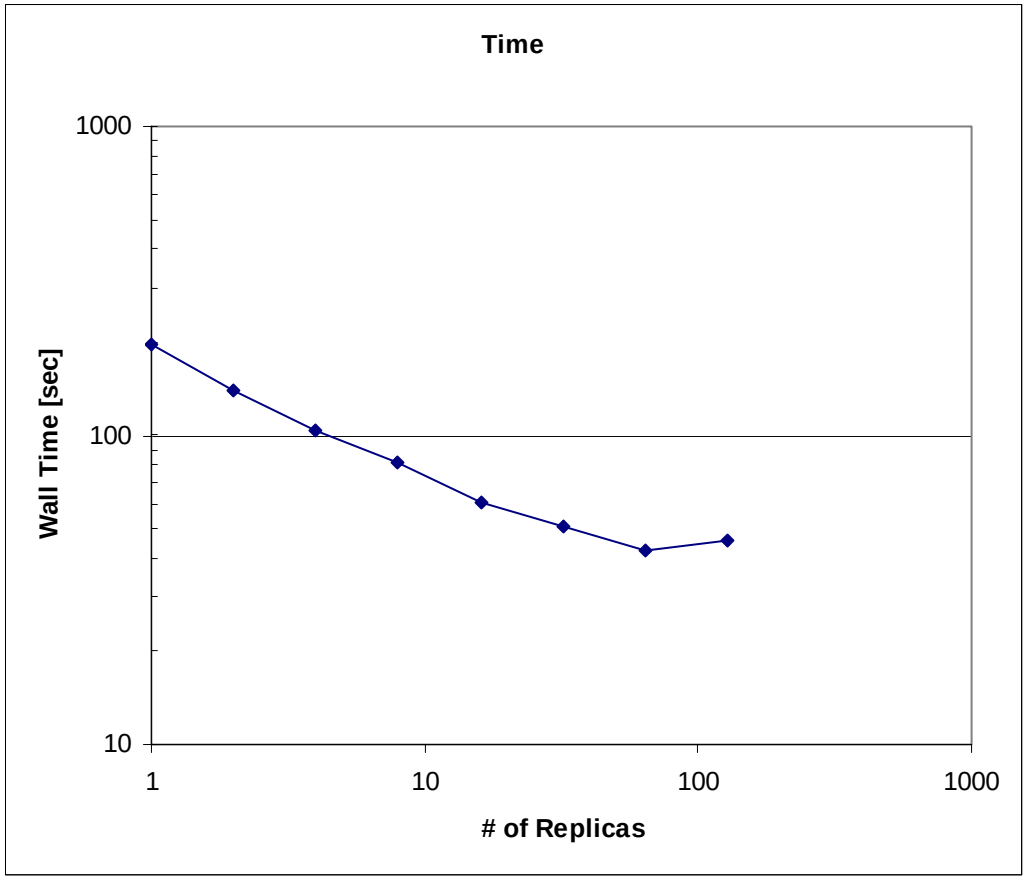




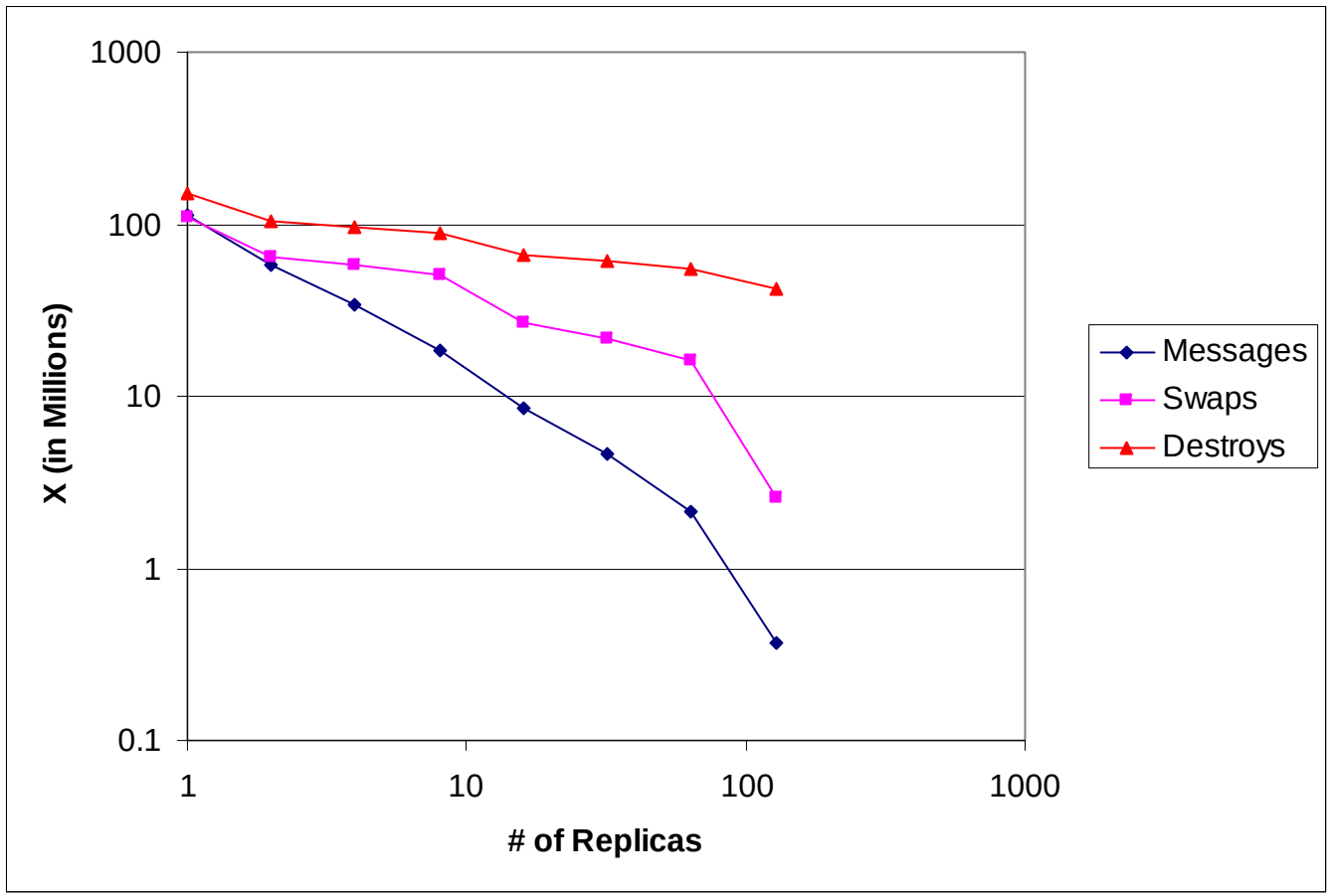
Partial vs Full Replication

- Complete problem description may not fit within available memory
- Memory availability/utilization – extremely difficult
- Synchronization can be limited to processors working with same particle/mesh group
- May be sensitive to placement of processor subgroups onto underlying hardware (off box)

Partial vs Full Replication Results



Partial vs Full Replication Results (cont.)



Conclusions

- Replication may be too valuable to completely ignore as Monte Carlo option
- Hybrid approach may be optimal solution combined with knowledge of problem
- Ideal replication may require a prohibitively detailed knowledge of target hardware
  - Memory capacity
  - Processor layout for communication

Thanks

- Jon Dahl
- Pooma team