A Communication-Optimal N-Body Algorithm for Direct Interactions

Michael Driscoll, Evangelos Georganas, Penporn Koanantakool, Edgar Solomonik, and Katherine Yelick

Motivation
- N-Body requires $n^2$ interactions = lots of communication
- Lower bounds from [Ballard, Demmel, Holtz, S. 2011a]
  $S = \Omega \left( \frac{n^2}{M} \right)$,  \hspace{0.5cm} W = \Omega \left( \frac{M^2}{M^2} \right)
- N-Body: at most $M^2$ force evaluations
  $S_{NB} = \Omega \left( \frac{E}{M^2} \right)$,  \hspace{0.5cm} W_{NB} = \Omega \left( \frac{E}{M} \right)
- M in the denominator = using extra memory can decrease lower bound.

No cutoff (all-pairs)

Naïve Algorithm
- $n$ particles, $p$ processors
  - local copy
  - replica buffers
  - $S_{naive} = \Theta \left( \frac{n^2}{p} \right) = \Theta \left( \frac{n^2}{p} \right)$
  - Each processor sends $p$ messages of size $n/p$
  - Communication-optimal.

CA-allpairs Algorithm
- $n$ particles
  - local copy
  - replica buffers
  - $S_{CA-allpairs} = \Theta \left( \frac{n}{p} \right) = \Theta \left( \frac{n}{p} \right)$
  - Replicate over $c$ layers
  - Shift teams particles along diagonal
  - For $p/c^2$ stages
  - Shift replicas by $c$ teams
  - Compute interactions
  - Reduce across column

Bounded: $l = \Theta \left( \frac{n}{c} \right)$

CA-allpairs (n, p), replication factor $c$ = $\Theta \left( \frac{n}{c} \right)$

Cutoff (1D)

Assumptions
- Particles are uniformly distributed.
- Cutoff distance $(r_c)$ spans multiple processor areas.
- Particles have $1D$ coordinates

Algorithm
- $m$ teams
  - local copy
  - replicas

Bounds
- Let $k$ be #interactions per particle.
  $k = \frac{m}{p/c^2}$ \hspace{0.5cm} n = \frac{mn}{p}$

$S_{CA-allpairs} = \Theta \left( \frac{n}{p} \right) = \Theta \left( \frac{n}{p} \right)$

Performance

Hopper: 24,576 cores, 196,608 particles
- Communication (Reduce) $\frac{n}{c}$
- Communication (Shift) $\frac{n}{c}$
- Computation $\frac{n}{c}$

Intrepid: 32,768 cores, 262,144 particles
- Communication (Reduce) $\frac{n}{c}$
- Communication (Shift) $\frac{n}{c}$
- Computation $\frac{n}{c}$

Conclusions
- Using extra memory $(c)$ copies reduces
  - Latency by a factor of $c^2$
  - Bandwidth by a factor of $c$.
- Communication avoidance decreases overall execution time for communication-bound problems.

Cutoff (2D)

Assumptions
- Particles are uniformly distributed.
- Cutoff distance $(r_c)$ spans multiple processor areas.
- Particles have $2D$ coordinates

Algorithm
- $p/c$ processor teams
  - $c$ layers
  - $c^2$ teams
- $S_{CA-allpairs} = \Theta \left( \frac{n}{c} \right)$

Performance

Hopper: 24,576 cores, 196,608 particles
- Communication (Reduce) $\frac{n}{c}$
- Communication (Shift) $\frac{n}{c}$
- Computation $\frac{n}{c}$

Intrepid: 32,768 cores, 262,144 particles
- Communication (Reduce) $\frac{n}{c}$
- Communication (Shift) $\frac{n}{c}$
- Computation $\frac{n}{c}$

- Observed up to $11.8 \times$ speedup over the non-CA version.
- Negligible benefits on compute-bound problems.
- Tunable replication factor $'c'$ in cases where cost of reduction comprises most of the communication cost.