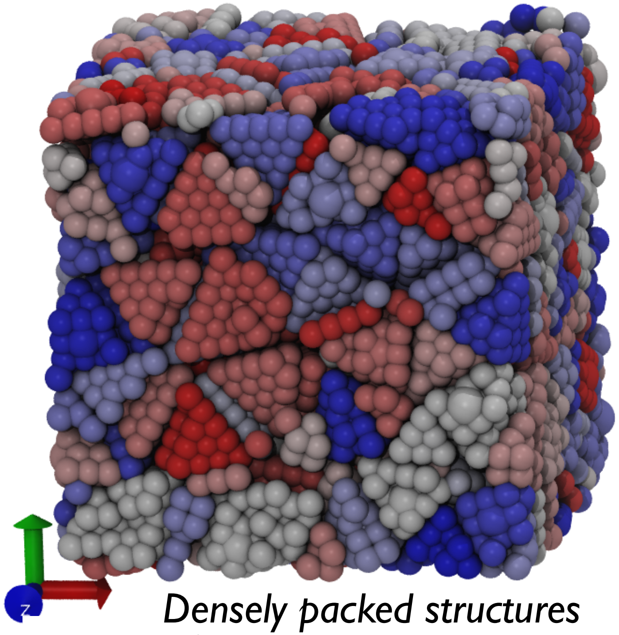
LAMMPS User Workshop, August, 2013

Alternative fix rigid for small particles

Trung Dac Nguyen[1], W. Michael Brown[1], Steve J. Plimpton[2]

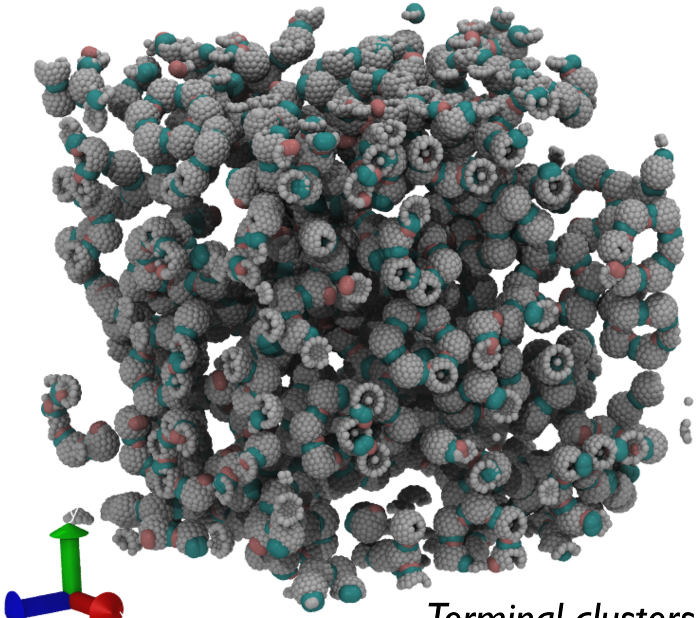
[1] National Center for Computational Sciences, Oak Ridge National Laboratory
[2] Sandia National Laboratory

Shaped nanoparticles



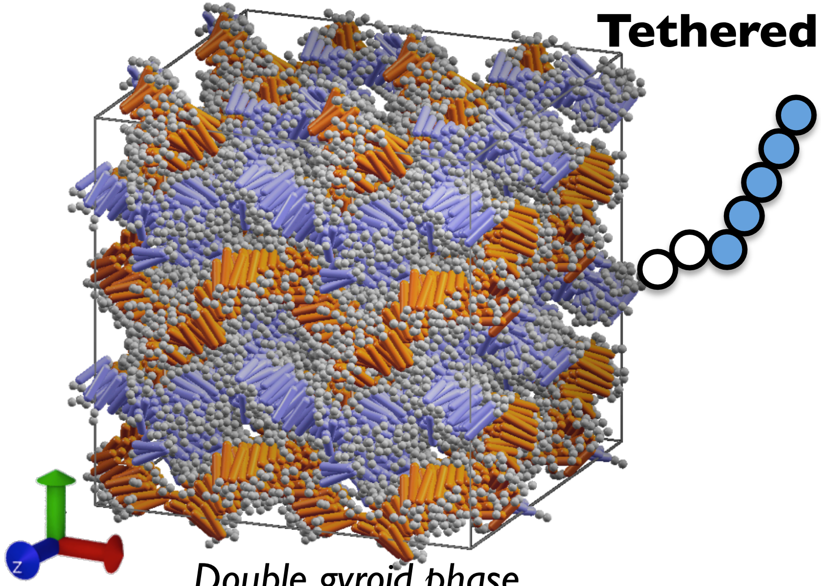
Densely packed structures

Patchy particles

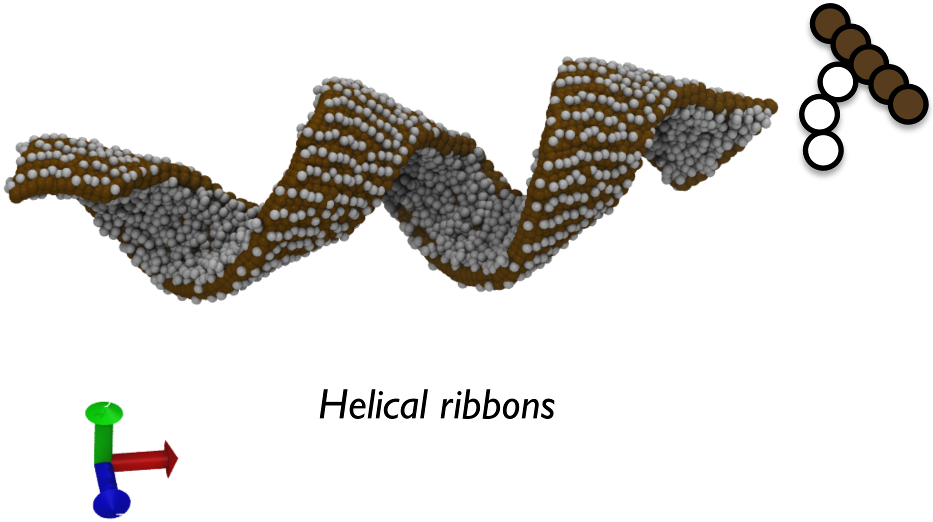


Terminal clusters

Tethered nanoparticles



Double gyroid phase



Helical ribbons

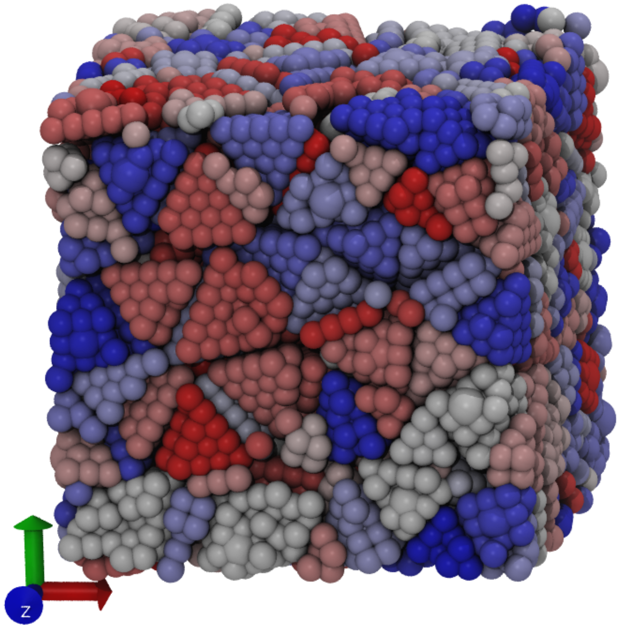
Zhang et al., *Nano Lett*, **2004**; Iacovella et al., *PRE*, **2007**; Nguyen et al. *Small*, **2009**

Rigid body models

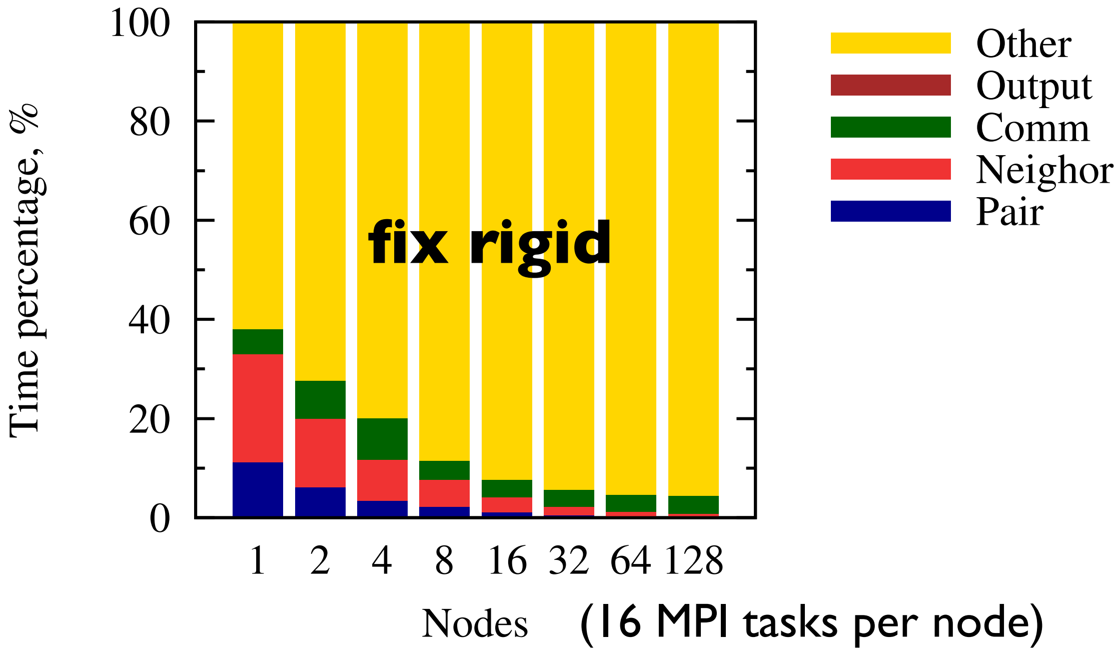
- Coarse-grained model of:
  - Molecules: CO_2, H_2O, liquid crystals
  - Nanoparticles, colloids and granular particles in different shapes other than spherical
- LAMMPS fix rigid
  - Rigid body dynamics using quaternions:
    - Compute body forces/torques from atom forces
    - Advance body translational/rotational degrees of freedom in NVE, NVT, NPT or NPH
    - Assign constituent atoms' position and velocity from body centers of mass and velocities

LAMMPS documentation and source code: **FixRigid**

Fix rigid overhead



N = 47M atoms (1.3M tetrahedra)



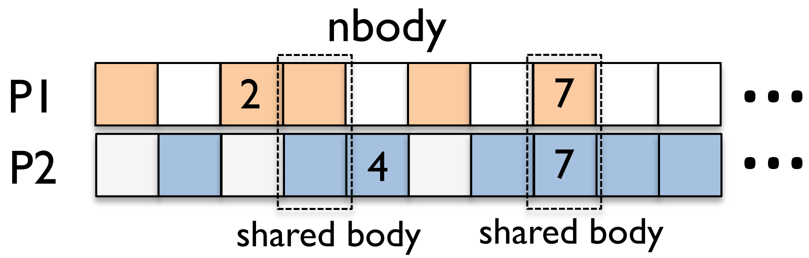
(16 MPI tasks per node)

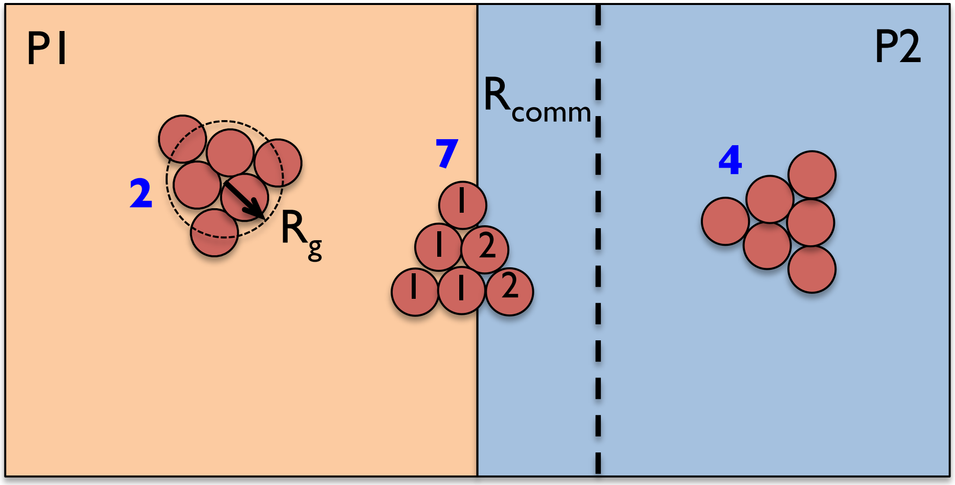
- Bottlenecks:
  - All-to-all communications for body forces and torques (MPI_Allreduce)
  - Replicas of body data updated by every MPI task:
    - Reset all the body forces and torques every time step
- **Small particles (or bodies)**:
  - Body dimensions ~ communication cutoff (by default, pair cutoff + skin)
    - Straddle only two sub-domains in each proc grid dimension

Local communication schemes for small bodies

- Fix rigid/small (by SJP, released 2/2013)
  - Bodies are treated like atoms (no replicated body data)
    - Only one proc updates a given body which is only owned by a local atom
  - Comm::forward_comm_variable_fix() **x 2**
    - Send/receive updated body info after initial_integrate() and final_integrate()
  - Comm::reverse_comm_fix() **x 1**
    - Tally body forces/torques to local bodies
  - Update local bodies and constituent local atoms
- Fix rigid/local (independent work since 2012)

fix rigid/local





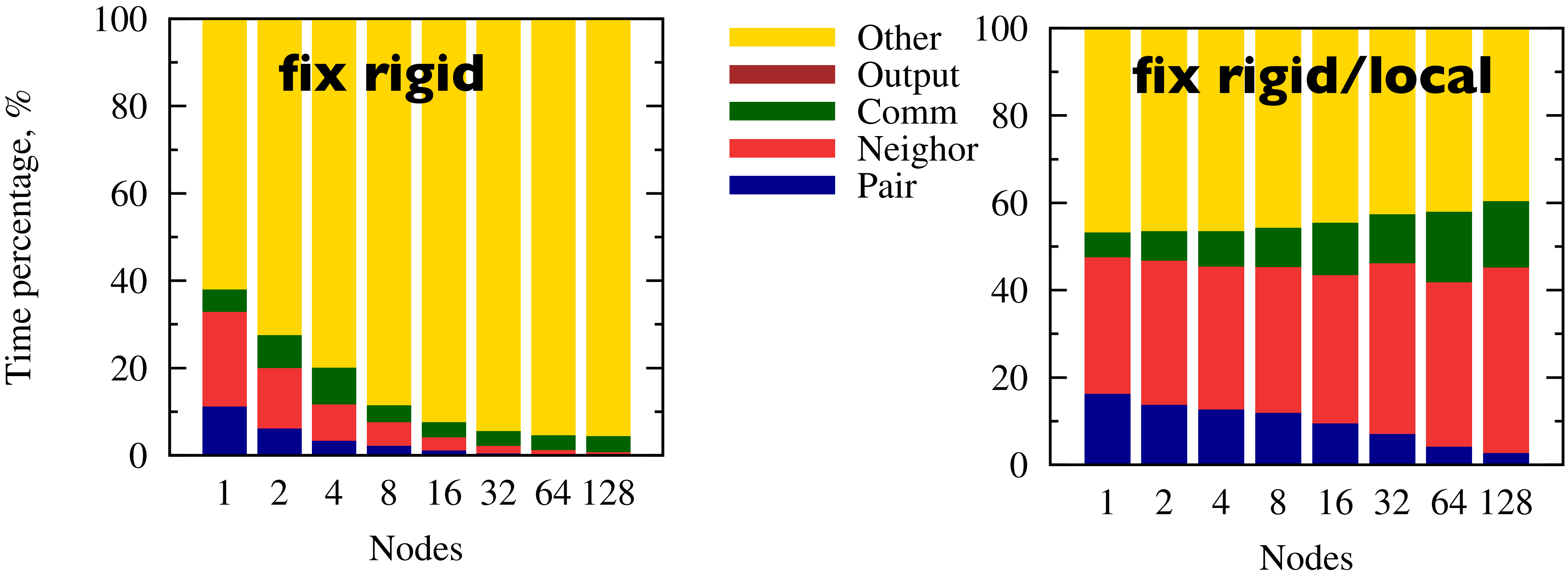
R_g = the maximum value of the radius of gyration of the rigid bodies in the system

- $R_{comm} > 2R_g$: all procs that share a body can acquire the full force/torque of the body via ghost atoms *via* **Comm::forward_comm_fix**()
- $R_g < R_{comm} < 2R_g$: at least one proc can acquire the full force/torque of the body via ghost atoms (need another call to **Comm::forward_comm_fix**() to broadcast)
- $R_{comm} < R_g$: dependent upon the processor grid and atom arrangement in bodies

FixRigidLocal implementation

- Derived from FixRigid
- Data structure:
  - Added an nbody array to avoid double count (unpacked_flag) array
  - Added an array of local and ghost bodies
- Member functions overridden from Fix
  - **pack_comm**() and **unpack_comm**(): *tally body forces/torques* via local and ghost atoms along with body indices (making sure no double count by toggling flags in the unpacked_flag array).
- Member functions overridden from FixRigid:
  - **pack_exchange**() and **unpack_exchange**(): *migrate bodies with atoms* (xcm, vcm, angmom, quat, imagebody, remapflag) in addition to body and displace
  - **pre_neighbor**(): update list of local and ghost bodies after exchange
  - **initial_integrate**(), **final_integrate**() and **deform**():
    - Reset forces and torques for local and ghost bodies before tallying (final_integrate())
    - Update only local bodies

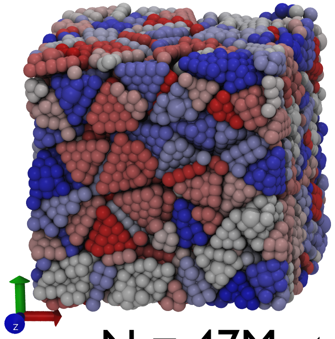
Rigid body integration overhead



Densely packed structure formed by 1.3M tetrahedra (i.e. 47M atoms),
Weeks-Chandler-Andersen potential (lj/cut w/ cutoff = $2^{1/6}\sigma$)

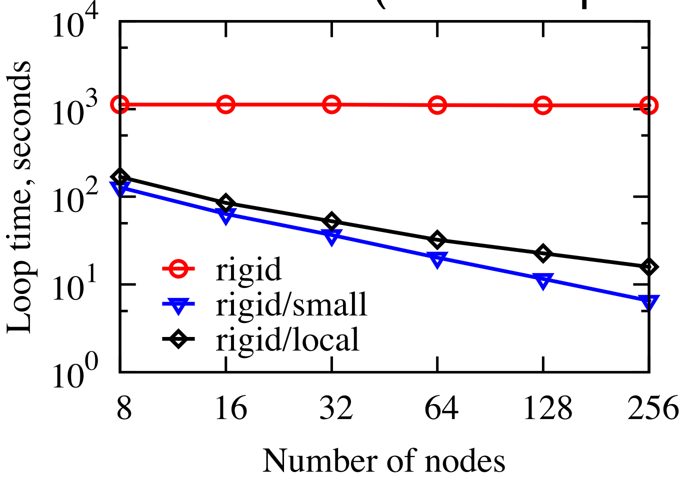
Benchmark results:

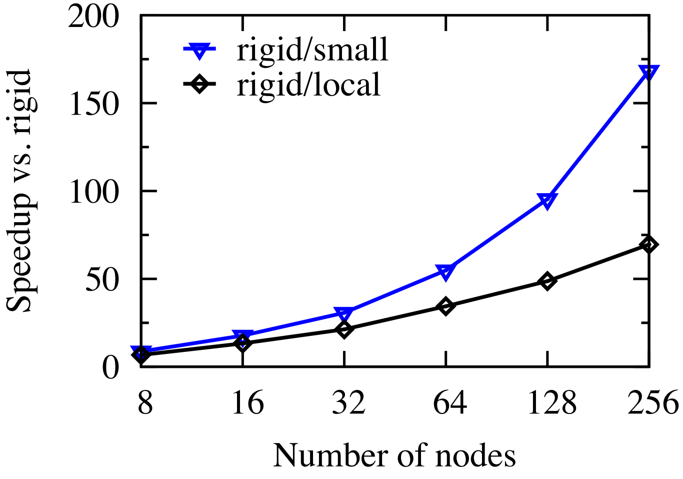
- *Cray Titan XK7, 16-core AMD Opteron 2.2 GHz CPUs, Gemini interconnect*
- *Strong scaling analysis (number of atoms fixed, increasing number of MPI tasks)*



Densely packed tetrahedra: lj/cut

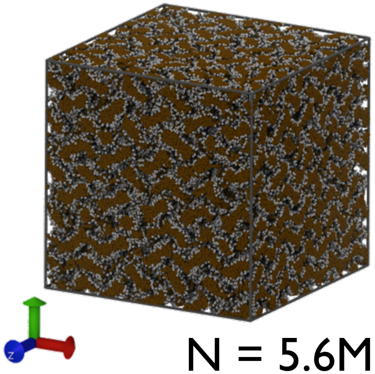
N = 47M atoms (35 atoms per body)





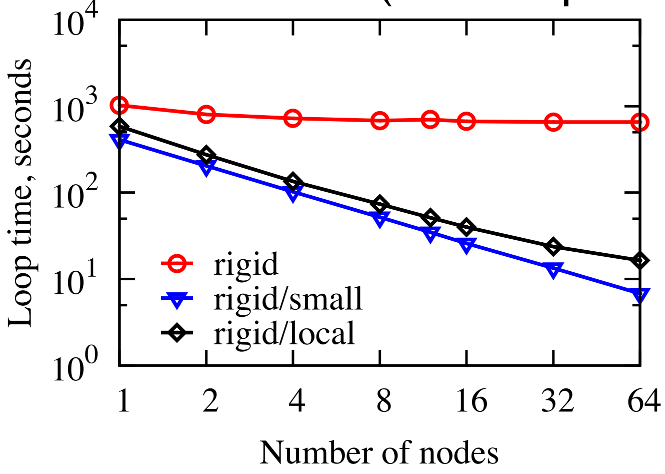
170x

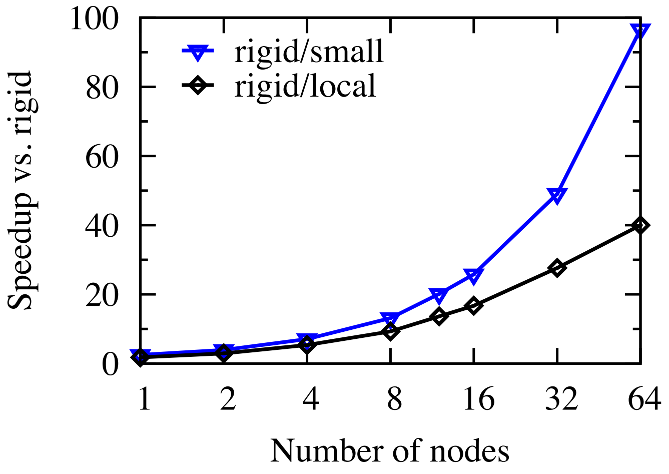
70x



Double gyroid by tethered nanorods: lj/cut

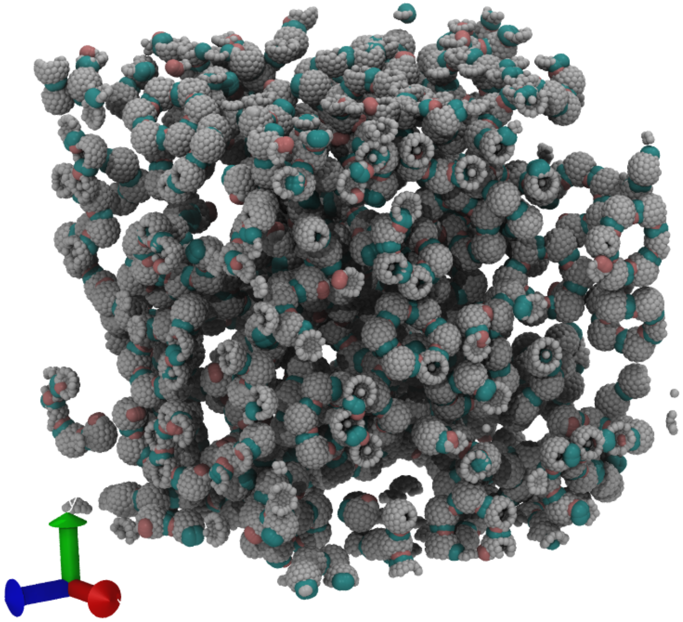
N = 5.6M atoms (7 atoms per body)





100x

40x

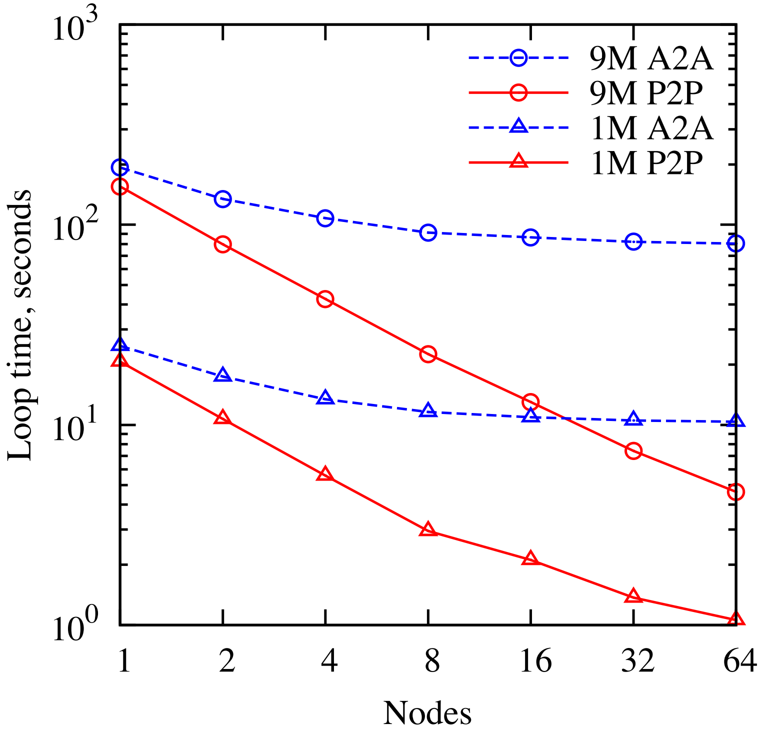


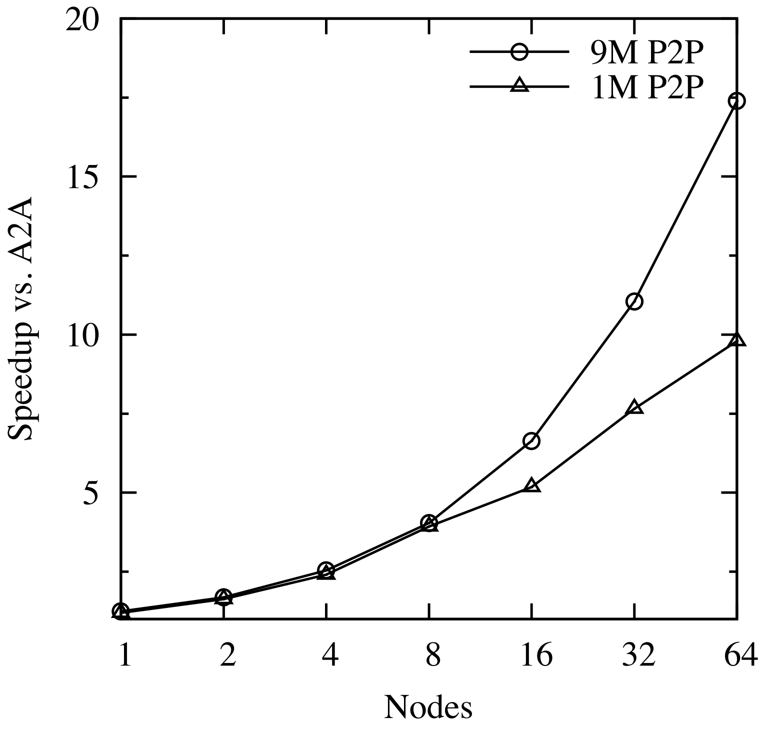
Rings formed by patchy particles: lj/cut

91 atoms per body

A2A = fix rigid

P2P = fix rigid/local





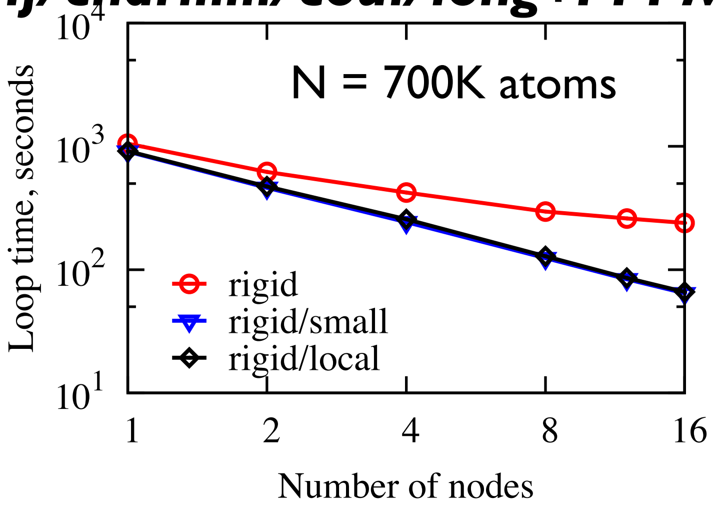
18x

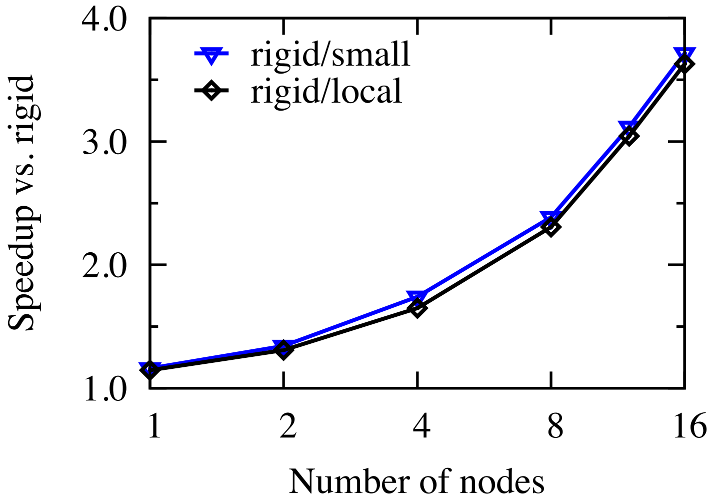
10x

Modest speedup

Systems where rigid body integration is not the primary bottleneck

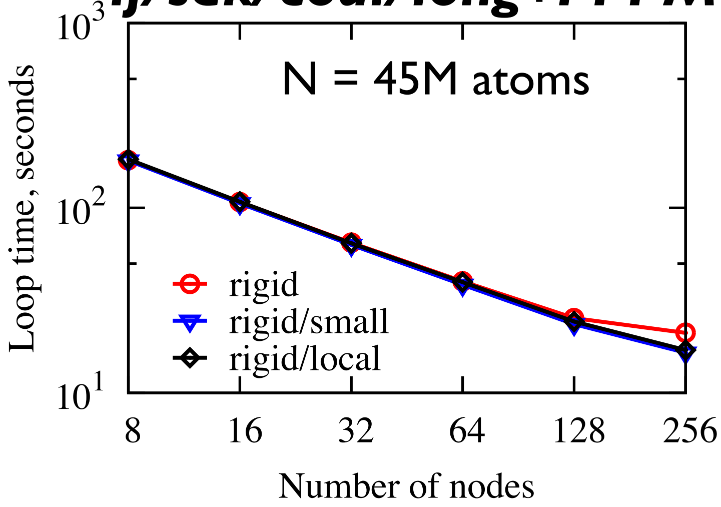
Water SPC/E: lj/charmm/coul/long+PPPM

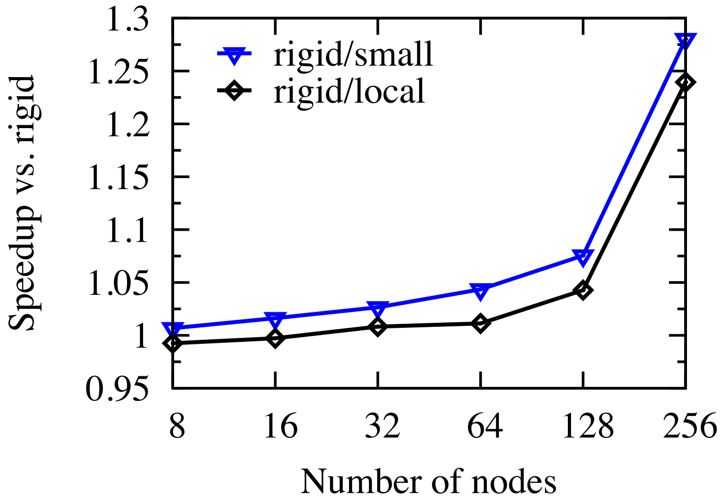




3.7x

Fullerene C60: lj/sdk/coul/long+PPPM





1.25x

Summary

- Rigid bodies of which dimensions are comparable to the communication cutoff should benefit from fix rigid/small and fix rigid/local
  - R_{comm}: communication single cutoff [value]
  - Further benefit from USER-OMP, GPU, USER-CUDA
- Rigid/local compared to rigid/small
  - Fewer local communications required: 1 (or 2) vs. 3
  - Replicated body data unresolved:
    - Conflict with spatial decomposition parallelism

Acknowledgements

LAMMPS with GPU Acceleration and Code Enhancements Available (Open Source):

- http://lammps.sandia.gov

Department of Energy, Office of Science under Contract No. DE-AC05-00OR22725

Thanks for attention-
Questions and comments?