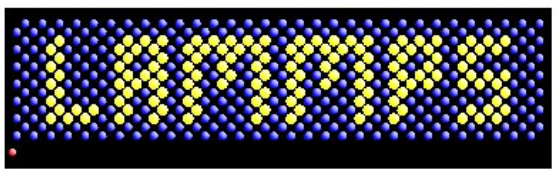
What's New in LAMMPS

Steve Plimpton
Sandia National Labs
sjplimp@sandia.gov

7th LAMMPS Workshop and Symposium
Virtual meeting – August 2021



Center for Computing Research

Sandia National Laboratories

Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

National Nuclear Security Administration

Two commands with significant enhancements

- Min_style fire for FIRE minimizer
  - Julien Guénolé (RWTH Aachen) &
    Erik Bitzek (FAU Erlangen-Nuernberg)
  - better convergence, added options
- Fix bond/react
  - Jake Gissinger (NASA Langley) talk: Thu late session
  - simplified templating via atom labels to describe reactions
  - plus other options and improvements

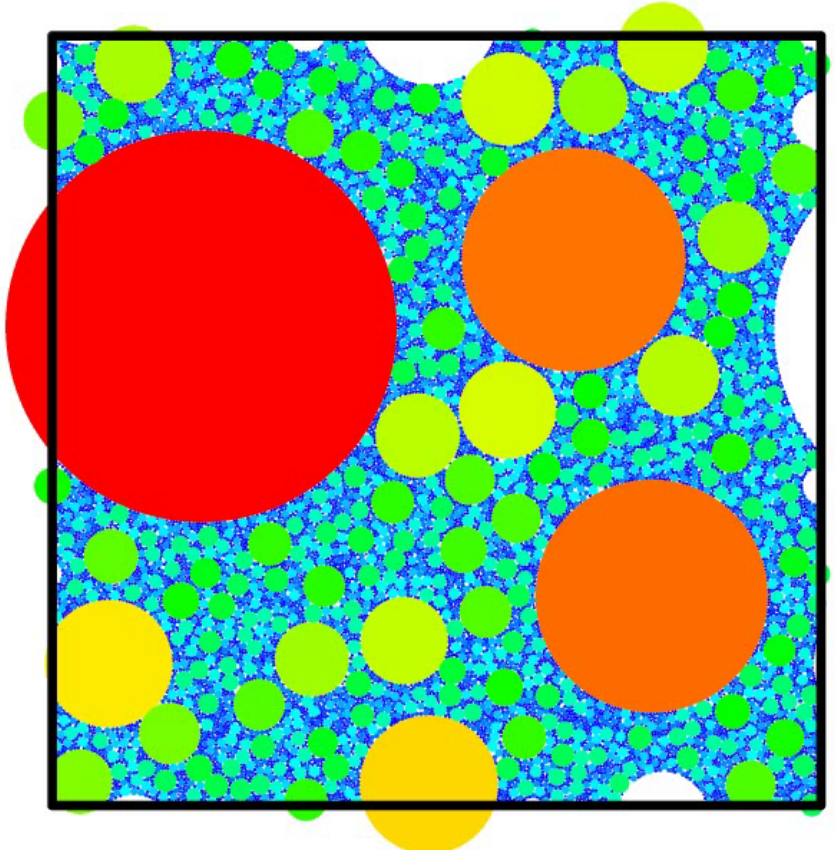
New commands

- Fix widom
  - Evangelos Voyiatzis (Royal DSM)
  - Widom insertions (Monte Carlo) of particles or molecules
- Fix numdiff and third_order command
  - Charlie Sievers (UC Davis)
  - forces via finite difference (debugging pair styles)
  - 3rd-order force constant tensor via FD
- Compute fabric
  - Joel Clemmer (Sandia)
  - pairwise fabric tensors
  - anisotropy/orientation of granular contacts

Improved multi neighboring for polydisperse systems

- Due to new algorithm from Kevin Stratford (U Edinburgh)
- Joel Clemmer (Sandia) helped implement in LAMMPS
- Old method:
  - one set of small-scale bins
  - different stencils for small/small, small/large, large/large
  - performance stalled at 10-20x polydispersity ratios
- New method:
  - separate binning for small and large particles
  - different stencils, similar to old method
  - optimized comm for small vs large ghost particles
  - good performance to 1000x or more polydispersity
  - timescale becomes the bottleneck, not neighbor finding

Shearing a 1000x polydisperse 2d granular system



Machine learning interatomic potential packages

- ML-SNAP, including new ChemSNAP variant
  - Aidan Thompson (Sandia) talk: Wed late session
  - Mitch Wood (Sandia) talk: Wed late session
  - Mary Alice Cusentino (Sandia) talk: Thu late session
- ML-RANN - rapid atomistic neural network potential
  - Christopher Barrett (MSU) talk: Thu early session
- ML-HDNNP = N2P2 lib for Behler-Parinello NN potentials
  - wrapper on external library
  - Andreas Singraber (U Vienna)
- ML-PACE - Atomic Cluster Expansion (ACE) potential
  - wrapper on external library
  - Yury Lysogorskiy, et al (ICAMS, Ruhr-Uni Bochum)
- SPIN + ML-SNAP for magnetic/elastic systems
  - Julien Tranchida (Sandia) talk: Thu late session

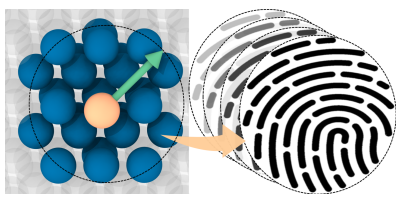
New ML potential framework via ML-IAP package

Aidan Thompson (Sandia) and Nick Lubbers (LANL)

- All MLIAPs perform 3 steps in MD context:
  1. compute descriptors from atoms
  2. compute model gradients with respect to descriptors
  3. compute forces from model gradients
- Step 2 is same for all Descriptors (Model provides it)
- Users can add new Descriptors and Models (steps 1 & 3)

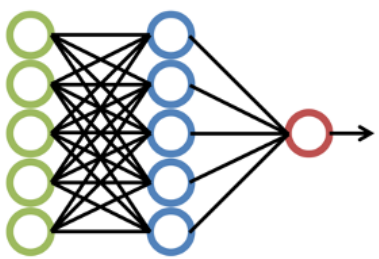
Descriptors:

2-body, 3-body, graph methods
moment tensors, SOAP, bispectrum, ACE



Energy Models:

linear & kernel-ridge regression, Gaussian process
non-linear opt, NNs & hierarchical NNs



New pair style MLIAP allows:

- mix-and-match any Descriptor with any Model
- call back to Python, link with PyTorch for GPUs

New packages

- DIELECTRIC
  - Trung Nguyen (Northwestern)
  - boundary element solvers for induced charge at interfaces
- BROWNIAN
  - Sam Cameron (U Bristol) and Stefan Paquay (Brandeis U)
  - Brownian dynamics for point, spherical, aspherical particles
  - fix propel/self for self-propelled particle models
- MESONT
  - Maxim Shugaev (U Virginia) and Philipp Kloza (U Cambridge)
  - coarse-grained carbon nanotube (CNT) potentials
- INTERLAYER
  - Wengen Ouyang (Tel Aviv U) & Jaap Kroes (Radboud U)
  - interatomic potentials between graphene layers
- BPM and RHEO (coming soon)
  - Joel Clemmer, Dan Bolintineanu, Thomas O'Connor (Sandia)
  - coarse-grained bonded particle models for brittle solids
  - reproducing hydrodynamics and elastic objects for liquids
  - Joel talk: Thu late session on both packages

New MDI package

- Wraps MDI code-coupling library
- Exchange data via MPI or sockets
- Authored by Taylor Barnes at MolSSI

New MDI package

- Wraps MDI code-coupling library
- Exchange data via MPI or sockets
- Authored by Taylor Barnes at MolSSI
- MDI uses client/server model $\Rightarrow$ driver/engine in MDI lingo
- Driver can use multiple engines and multiple instances of each
  - AIMD: MD using QM as engine
  - MC: MC using MD as engine
  - PRD or sampling or NEB: MD and/or QM code as engines
- MDI package in LAMMPS
  - support for LAMMPS as driver or engine
  - can be engine in 2 modes: stand-alone code or library plugin
  - enables coupling to variety of QM codes thru same interface
- Flexible, simple interface:
  - MPI-like data communication via MDI_Send() and MDI_Recv()
  - Codes in C/C++, Fortran, Python
  - LAMMPS can run on N procs, another code on M procs

Increased code coverage by accelerator packages

- OPENMP package for CPUs
  - 112 pair styles, 44 bond/angle/dihed/impro styles
  - 28 fixes, Ewald/PPPM/MSM
- INTEL package for CPUs
  - 18 pair, 10 bond/angle/dihed/impro, 7 fixes, PPPM
- GPU package for GPUs
  - 58 pair styles, 6 fixes, PPPM
- KOKKOS package for GPUs
  - 44 pair styles, 13 bond/angle/dihed/impro styles
  - 26 fixes, 3 computes, PPPM
  - Stan Moore talk: Wed late session

- Support for AMD and Intel GPUs
  - GPU package now has a HIP backend
  - KOKKOS package now has a HIP backend
  - GPU package OpenCL backend now works for Intel GPUs

Simplified adding of new atom styles – this replaces 500 LOC for atom_style charge

```
AtomVecCharge::AtomVecCharge(LAMMPS *lmp) :  AtomVec(lmp)
{
  molecular = Atom::ATOMIC;
  mass_type = PER_TYPE;
  atom->q_flag = 1;

  fields_grow = (char *) "q";
  fields_copy = (char *) "q";
  fields_comm = (char *) "";
  fields_comm_vel = (char *) "";
  fields_reverse = (char *) "";
  fields_border = (char *) "q";
  fields_border_vel = (char *) "q";
  fields_exchange = (char *) "q";
  fields_restart = (char *) "q";
  fields_create = (char *) "q";
  fields_data_atom = (char *) "id type q x";
  fields_data_vel = (char *) "id v";

  setup_fields();
}
```

General code and documentation improvements

- Code syntax and GitHub contributions
  - allowing more C++ style coding, auto clang-formatting
  - added utility classes & functions, e.g. for file parsing
  - extensive library of unit tests
  - more standardized code contribution process
  - Axel Kohlmeyer talk: Fri developer session
- Packages
  - STANDARD + USER variants ⇒ just packages
  - now ∼90 packages
- Manual
  - LaTeX equations now included directly in doc page source
  - large Programmer Guide section
  - auto-documented library interfaces
  - several expanded ways to use Python with LAMMPS
  - Richard Berger talk: Fri developer session
- LAMMPS website moved to Temple U:

  `https://www.lammps.org`

That's all

New LAMMPS overview paper in review, hopefully out soon:

Aidan Thompson (Sandia) first author + 14 co-authors

"LAMMPS - A flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales"

Thanks again for attending and participating in our workshop!