Shared Metadata and Data Formats for Big-Data Driven Materials Science: a NOMAD/FAIR-DI Workshop

Open questions and needs from hard and soft materials: discussion on molecular mechanics

Moderated by James Kermode

Warwick Centre for Predictive Modelling
School of Engineering, University of Warwick, United Kingdom

www.warwick.ac.uk/jrkermode
Metadata – using, storing and sharing

- **My background – using and reusing atomistic configurations**
  - Concurrent multiscale simulations (QM/MM)
  - Fitting interatomic potentials, uncertainty quantification
  - Experience of what’s needed for a minimal workable code-independent format

- **Drivers for sharing**
  - Reproducibility
  - Validation & verification – e.g. as part of continuous integration testing of software
  - Archiving of research data – may be required by funders, journals, or simply useful!
  - Sharing research data between collaborators (pre-publication) and wider (afterwards)

- **Barriers**
  - Technical – file formats, sharing mechanisms, access control
  - Social – cost/benefit, single or few configurations give small benefit, retain control

- **Consensus on distributed database solution**
  - Emerging consensus on computational framework – Python and/or REST APIs
  - Growing number of large datasets that have high value (e.g. OPTiMaDe)
  - Enable reuse of data by machine learning and data mining methods
  - Need for minimal, discoverable FAIR standard for exchanging data and metadata
Metadata Challenges for Force Field codes

- Representing initial conditions (atoms, cell)  Common to DFT and FF
- Defining output properties (quantities of interest) and uncertainties
- Representing sampling of phase space (thermostats, constraints, …)  Both – more pronounced in FF
- Large quantity of data produced
- Representing interatomic potentials (aka force fields)  FF specific
- Representing topology (atom types, bonds, angles)
Representing Potentials: Materials & Molecules

- Potentials used in computational materials science and in bimolecular simulation pose different metadata challenges
- Comp mat sci FFs typically reactive, fully many-body with finite cutoffs
- Functional forms may be analytic, tabulated or implicitly defined from data (e.g. ML potentials)
- Some potentials involve complex algorithms, eg. charge equilibration, polarisation
- Only truly defined by implementation?

- In bio sim, functional forms can be simpler (but not always, CG potentials also fully many-body; long-range electrostatic method must be carefully defined)
- More than one atom type for a given chemical species (e.g. C, Cα)
- Typically unreactive – but topology must be built first, often following complex rules
- Often hard to migrate simulations between codes?

Use case: dislocation glide in Ni-based superalloys

- MD simulation of dislocations in γ phase Ni
  - QoI: dislocation core splitting with associated uncertainty (cf. TEM experiments)
  - Aleatoric and epistemic uncertainties
  - Model uncertainty – both in parameters and functional form
  - Random microstructures, limited runtime
  - Algorithmic uncertainty in solvers
  - How much of this can/should be stored?
  - Complex simulation script ≈ workflow!

EAM, 5% Al, T = 300 K, 100 MPa shear stress

Reproducibility & Uncertainty Quantification

**QoIs:** vacancy formation and migration energies

Model + Test framework (cf. KIM)

https://github.com/libAtoms/silicon-testing-framework
https://doi.org/10.5281/zenodo.1250555
### Status of support for MD codes in NOMAD

<table>
<thead>
<tr>
<th>Code</th>
<th>Citations (2013-17)</th>
<th>Type</th>
<th>Search Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gaussian</td>
<td>19100</td>
<td>DFT</td>
<td>Frisch</td>
</tr>
<tr>
<td>VASP</td>
<td>17900</td>
<td>DFT</td>
<td>Kresse</td>
</tr>
<tr>
<td>Gromacs</td>
<td>11200</td>
<td>FF</td>
<td>Lindahl</td>
</tr>
<tr>
<td>LAMMPS</td>
<td>10300</td>
<td>FF</td>
<td>Plimpton</td>
</tr>
<tr>
<td>Amber</td>
<td>9440</td>
<td>FF</td>
<td>Kollman</td>
</tr>
<tr>
<td>NAMD</td>
<td>7110</td>
<td>FF</td>
<td>Schulten</td>
</tr>
<tr>
<td>GROMOS</td>
<td>7080</td>
<td>FF</td>
<td>Van Gunsteren</td>
</tr>
<tr>
<td>Quantum Espresso</td>
<td>6960</td>
<td>DFT</td>
<td>Giannozzi</td>
</tr>
<tr>
<td>ASE/ASAP</td>
<td>6650</td>
<td>FF</td>
<td>Jacobsen</td>
</tr>
<tr>
<td>CHARMM</td>
<td>6250</td>
<td>FF</td>
<td>Karplus</td>
</tr>
<tr>
<td>Discovery Studio</td>
<td>6240</td>
<td>DFT, FF</td>
<td>Accelyrs</td>
</tr>
<tr>
<td>GAMESS</td>
<td>5780</td>
<td>DFT</td>
<td>Gordon</td>
</tr>
<tr>
<td>WIEN2k</td>
<td>5570</td>
<td>DFT</td>
<td>Blaha</td>
</tr>
<tr>
<td>CASTEP</td>
<td>5330</td>
<td>DFT</td>
<td>Payne</td>
</tr>
<tr>
<td>Molpro</td>
<td>4440</td>
<td>DFT</td>
<td>Werner</td>
</tr>
</tbody>
</table>

- Conversion layers for the most popular MD codes have been implemented
- Remaining challenges including fully representing force fields, topology, sampling schemes, etc.
- Prototype interface between NOMAD & OpenKIM:
  - ✅ Enables extraction of reference data from NOMAD to KIM for validation
  - ❌ Matching KIM model IDs to NOMAD data – not implemented; currently model IDs can be output by LAMMPS, but not by other supported codes

---

Image courtesy of Berk Onat (Warwick)  
[www.nomad-coe.eu](http://www.nomad-coe.eu)
NOMAD Metadata for MD simulations

Hierarchy of metadata

- **section_run**
  - **section_frame_sequence**
    - **section_user_quantity_stats**
      - (time)
  - **section_sampling_method**
    - (ensemble type, sampling method, …)
  - **section_single_configuration_calculation**
    - **section_user_quantity**
      - (energy, pressure, temperature, …)
    - **section_single_configuration_to_system_ref**
      - (atom position, simulation cell, …)
    - **section_single_configuration_to_calculation_method_ref**
      - (force field parameters)
    - **section_single_configuration_to_constraint_ref**
      - (thermostat, barostat, fixes)
- **section_system**
  - **section_topology**
    - **section_atom_type**
    - **section_groups**
    - (topology)
  - **section_method**
    - **section_interaction**

Frame sequence to sampling ref

- **frame_sequence_to_sampling_ref**
  - one for each frame (usually)
  - one for each frame (rarely)
  - only one for all frames

Image courtesy of Adam Fekete (KCL)
## Metadata Challenges for Force Field codes

- Representing initial conditions (atoms, cell)  
  **Common to DFT and FF**

- Defining output properties (quantities of interest) and uncertainties

- Representing sampling of phase space (thermostats, constraints, …)  
  **Both – more pronounced in FF**

- Large quantity of data produced

- Representing interatomic potentials (aka force fields)

- Representing topology (atom types, bonds, angles)  
  **FF specific**
Questions for Discussion

1. How should we deal with the large quantity of data produced by MD simulations? Leave it up to users? Subsample trajectories? If so, by what metric?

2. How can interatomic potentials be represented? cf. OpenKIM (comp mat sci), OpenForceField and MolSSI (bio sim)

3. For molecular systems, do we need code independent representations of topology? (atom types, bonds, angles, dihedrals, what else?)

4. Can we represent sampling of phase space in a code independent way? (thermostats, constraints, algorithms, workflows, …)

5. Is containerisation of codes part of the solution (e.g. Docker, Singularity)? Is this just an excuse for bad dependency management by developers?

6. How precisely can/should we define output properties? Instantaneous vs. averaged? Are ontologies needed? (cf. KIM properties framework)

7. Do we need/want a single metadata/ontology approach to achieve critical mass?

8. Do workflows in MD calculations need special treatment?

9. How should changes to metadata be managed?

10. Anything else?