nap: A molecular dynamics package with parameter-optimization programs for classical and machine-learning potentials

Ryo KOBAYASHI[1]

1 Department of Physical Science and Engineering, Nagoya Institute of Technology, Gokiso, Showa, Nagoya 466-8555, Japan

Summary

The **nap** is a package for molecular dynamics (MD) simulation consisting of an MD program (**pmd**) that can perform large-scale simulation using a spatial-decomposition technique and two parameter-optimization programs: one for classical (CL) potentials (**fp.py**) and another for machine-learning (ML) potentials (**fitpot**). Since the numbers of parameters to be optimized are much different between CL and ML potentials, optimization approaches for them are also different; meta-heuristic global minimum-search algorithms for the CL potentials, in which the numbers of parameters are usually much less than one hundred, and gradient-based methods for the ML potentials. The parameters of CL potentials can be optimized to any target quantity that can be computed using the potentials since meta-heuristic methods do not require the derivatives of the quantity with respect to parameters. On the other hand, ML-potential parameters can be optimized to only energies, forces on atoms and stress components of reference systems, mainly because gradient-based methods require the derivatives of other quantities with respect to parameters, and the analytical derivatives and the coding of them are usually painful and sometimes impossible. Potentials can be used in combination with any other potential, such as pair and angular potentials, short-range and long-range potentials, CL and ML potentials. With using the **nap** package, users can perform MD simulation of solid-state materials with the choice of different levels of complexity (CL or ML) by creating interatomic potentials optimized to quantum-mechanical calculation data even if no potential is available.

Statement of need

MD simulation is widely used in many research fields such as materials science, chemistry, physics, etc., to study dynamics of atoms or molecules. In order to perform MD simulation of systems including large number of atoms, where quantum-mechanical calculations can not be used due to their computational cost, empirical interatomic potentials between species are required. And the results of MD simulation are strongly dependent on the property or accuracy of the potentials used in the simulation. Hence, there are a lot of CL potential models have been proposed such as Lennard-Jones (LJ) potential for van der Waals interaction, Coulombic potential for ionic interaction, Morse potential for covalent interactions (Morse, 1929), angular-dependent models for angles between covalent bonds, bond-order models for more complex systems, etc. Recently ML potentials have been also actively studied because they are usually more flexible and can reproduce reference data more precisely than CL potentials. Even though the potential is flexible or suitable to problems considered, the parameters in the

potential model still need to be optimized to well reproduce the properties or phenomena that are in focus.

There are already a lot of MD programs such as LAMMPS (Plimpton, 1995) and IMD (Stadler et al., 1997), and some of them can use both CL and ML potentials. And also there are some parameter-optimization programs that can produce parameter sets available in the other MD programs such as aenet (Artrith & Urban, 2016) for ML potentials and potfit (Brommer et al., 2015; Brommer & Gähler, 2007) for CL potentials. However, there is also a demand of combining ML potentials and simpler CL potentials, e.g., ML potential with Coulomb interactions (Morawietz & Behler, 2013) and ML potential with core-core repulsions (Wang et al., 2019), since creating an ML potential that covers very short-range and/or very long-range interactions is very inefficient. Thus it is beneficial if the programs of parameter-optimization for both CL and ML potentials are in one package and highly connected to one MD program and it will be more efficient than using several different programs to optimize parameters of CL and ML potentials.

Programs and functionalities

- **pmd**: A Fortran program to perform MD simulation. It can perform large-scale MD simulation efficiently by using linked-cell list (Allen & Tildesley, 2017) and spatial decomposition with MPI parallelization. Several widely-used interatomic potentials for solid-state materials are implemented, for example, pair potentials such as LJ, Morse, Coulomb and screened Coulomb; angular-dependent potentials such as Stillinger-Weber (Stillinger & Weber, 1985) and Tersoff (Tersoff, 1988); multi-body potentials such as Finnis-Sinclair (Finnis & Sinclair, 1984) and embeded-atom method (Daw & Baskes, 1984); machine-learning potentials such as linear-regression (Seko et al., 2014) and neural-network (NN) (Behler & Parrinello, 2007).
- **fp.py**: A Python program to optimize the parameters of classical potentials using meta-heuristic algorithms such as cuckoo search (Yang & Deb, 2009). Meta-heuristic algorithms need to perform MD simulations and to evaluate the quantities to be compared with target values of each individual representing a candidate parameter set. These jobs by individuals are performed as child processes of the program by calling a shell script that describes what to compute using given parameters. This program has a functionality of automatic update of the search range of parameters and it allows to optimize parameters efficiently and less dependent on the initial ranges of parameters (Kobayashi et al., 2020).
- **fitpot**: A Fortran program to optimize the parameters of ML potentials, such as linear regression and NN, using gradient-based methods (Kobayashi et al., 2017). It can use high-performance computer resources to evaluate energies, forces and stresses of a huge number of sample systems in parallel using the MPI library, which is sometimes crucial since in general ML models require a lot of sampling to avoid over-fitting and make them robust.

Acknowledgements

References

Allen, M. P., & Tildesley, D. J. (2017). *Computer simulation of liquids*. Oxford University Press. https://doi.org/10.1093/oso/9780198803195.001.0001

Artrith, N., & Urban, A. (2016). An implementation of artificial neural-network potentials for atomistic materials simulations: Performance for TiO2. *Comput. Mater. Sci.*, *114*(C), 135–150. https://doi.org/10.1016/j.commatsci.2015.11.047

Behler, J., & Parrinello, M. (2007). Generalized Neural-Network representation of High-Dimensional Potential-Energy surfaces. *Phys. Rev. Lett.*, *98*(14), 146401–146401. https://doi.org/10.1103/PhysRevLett.98.146401

Brommer, P., & Gähler, F. (2007). Potfit: Effective potentials from ab initio data. *Modell. Simul. Mater. Sci. Eng.*, *15*(3), 295–304. https://doi.org/10.1088/0965-0393/15/3/008

Brommer, P., Kiselev, A., Schopf, D., Beck, P., Roth, J., & Trebin, H.-R. (2015). Classical interaction potentials for diverse materials from ab initio data: A review of potfit. *Modell. Simul. Mater. Sci. Eng.*, *23*(7). https://doi.org/10.1088/0965-0393/23/7/074002

Daw, M. S., & Baskes, M. I. (1984). Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals. *Phys. Rev. B: Condens. Matter Mater. Phys.*, *29*(12), 6443–6453. https://doi.org/10.1103/PhysRevB.29.6443

Finnis, M. W., & Sinclair, J. E. (1984). A simple empirical n-body potential for transition metals. *Philos. Mag. A*, *50*(1), 45–55. https://doi.org/10.1080/01418618408244210

Kobayashi, R., Giofré, D., Junge, T., Ceriotti, M., & Curtin, W. A. (2017). Neural network potential for Al-Mg-Si alloys. *Physical Review Materials*, *1*(5), 53604–53611. https://doi.org/10.1103/PhysRevMaterials.1.053604

Kobayashi, R., Miyaji, Y., Nakano, K., & Nakayama, M. (2020). High-throughput production of force-fields for solid-state electrolyte materials. *APL Materials*, *8*(8), 081111. https://doi.org/10.1063/5.0015373

Morawietz, T., & Behler, J. (2013). A density-functional theory-based neural network potential for water clusters including van der Waals corrections. *J. Phys. Chem. A*, *117*(32), 7356–7366. https://doi.org/10.1021/jp401225b

Morse, P. M. (1929). Diatomic molecules according to the wave mechanics. II. Vibrational levels. *Phys. Rev.*, *34*(1), 57–64. https://doi.org/10.1103/PhysRev.34.57

Plimpton, S. (1995). Fast parallel algorithms for short-range molecular dynamics. *J. Comput. Phys.*, *117*(1), 1–19. https://doi.org/10.1006/jcph.1995.1039

Seko, A., Takahashi, A., Takahashi, A., Tanaka, I., & Tanaka, I. (2014). Sparse representation for a potential energy surface. *Phys. Rev. B: Condens. Matter Mater. Phys.*, *90*(2), 24101–24101. https://doi.org/10.1103/PhysRevB.90.024101

Stadler, J., Mikulla, R., & Trebin, H.-R. (1997). IMD: A software package for molecular dynamics studies on parallel computers. *Int. J. Mod. Phys. C*, *08*(05), 1131–1140. https://doi.org/10.1142/S0129183197000990

Stillinger, F. H., & Weber, T. A. (1985). Computer simulation of local order in condensed phases of silicon. *Phys. Rev. B: Condens. Matter Mater. Phys.*, *31*(8), 5262–5271. https://doi.org/10.1103/PhysRevB.31.5262

Tersoff, J. (1988). New empirical approach for the structure and energy of covalent systems. *Phys. Rev. B: Condens. Matter Mater. Phys.*, *37*(12), 6991–7000. https://doi.org/10.1103/PhysRevB.37.6991

Wang, H., Guo, X., Zhang, L., Wang, H., & Xue, J. (2019). Deep learning inter-atomic potential model for accurate irradiation damage simulations. *Appl. Phys. Lett.*, *114*(24). https://doi.org/10.1063/1.5098061

Yang, X.-S., & Deb, S. (2009). Cuckoo search via levy flights. *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*, 210–214. https://doi.org/10.1109/NABIC.2009.5393690