# OpenMD: A parallel molecular dynamics engine for complex systems and interfaces

Cody R. Drisko [1], **Hemanta Bhattarai** [2], **Christopher J. Fennell** [3], **Kelsey M. Stocker** [4], **Charles F. Vardeman II** [5], and **J. Daniel Gezelter** [1]¶

1 Department of Chemistry and Biochemistry, University of Notre Dame, Notre Dame, IN, United States **2** Department of Physics, Goshen College, Goshen, IN, United States **3** Department of Chemistry, Oklahoma State University, Stillwater, OK, United States **4** Department of Biochemistry, Chemistry, Environment, and Physics, Suffolk University, Boston, MA, United States **5** Center for Research Computing, University of Notre Dame, Notre Dame, IN, United States ¶ Corresponding author

Summary

Molecular dynamics (MD) simulations help bridge the gap between quantum mechanical calculations, which trade computational resources and system size for chemical accuracy, and continuum simulations, which utilize bulk and surface material properties to make predictions. MD provides the ability to study emergent properties and dynamics for relatively large systems ($\sim 10^6$ atoms) over a span of nano- to micro-seconds. MD has been used for simulations of complex systems such as liquids, proteins, membranes, nanoparticles, interfaces, and porous materials, and the approximations used in MD simulations allow us to reach experimentally-relevant time and length scales (Allen & Tildesley, 2017). A molecular dynamics engine leverages classical equations of motion to evolve atomic or molecular coordinates according to a well-defined potential energy function, known as a force field, which is a function of those atomic coordinates. There are a number of high quality molecular dynamics engines that specialize in materials (Thompson et al., 2022) or biomolecular (Brooks et al., 2009; Case et al., 2023) simulations or are models of computational efficiency (Abraham et al., 2015; Eastman et al., 2017). In this paper, we provide background on an open source molecular dynamics engine, OpenMD, which specializes in complex systems and interfaces and was just released into version 3.1.

OpenMD is capable of efficiently simulating a variety of complex systems using standard point-particle atoms, as well as atoms with orientational degrees of freedom (e.g. point multipoles and coarse-grained assemblies), and atoms with additional degrees of freedom (e.g. fluctuating charges). It provides a test-bed for new molecular simulation methodology, while being efficient and easy to use. Liquids, proteins, zeolites, lipids, inorganic nanomaterials, transition metals (bulk, flat interfaces, and nanoparticles), alloys, solid-liquid interfaces, and a wide array of other systems have all been simulated using this code. OpenMD works on parallel computers using the Message Passing Interface (MPI), and is packaged with trajectory analysis and utility programs that are easy to use, extend, and modify.

From the beginning, OpenMD has been an open source project and has been maintained in accordance with the FAIR (findable, accessible, interoperable, and reusable) principles for research software (Barker et al., 2022). It uses a meta-data language that is tightly integrated with input (.omd) and trajectory (.dump) files, providing a standardized, human-readable way to completely describe molecular systems and simulation parameters. All data files are stamped with the code revision that generated that data. This allows OpenMD simulations to be easily reproduced, modified, and reused by others. The `<MetaData>` section of these files also serve as a form of machine-actionable meta-data, clearly specifying the composition of the molecular system in a manner that promotes interoperability with other software tools.

Statement of need

OpenMD builds on the foundations of the Object-Oriented Parallel Simulation Engine (OOPSE) program (Meineke et al., 2005), rewritten in modern C++. It differs from other contemporary molecular dynamics engines in its focus on complex interfaces. A number of features unique to OpenMD facilitate the study of this problem space, including efficient non-equilibrium algorithms, non-periodic simulations, metal polarizability models, and real space electrostatics. The first such feature is Reverse Non-Equilibrium Molecular Dynamics (RNEMD), a family of algorithms which impose a non-physical flux on a system and use linear response theory to compute transport properties as the system approaches steady state. The goal of RNEMD methods is to calculate the relevant transport property (λ) that connects the flux (**J**) and driving force (∇X) according to the generalized equation,

$$\mathbf{J} = -\lambda \nabla X. \tag{1}$$

OpenMD is also capable of performing condensed phase simulations without the use of periodic boundary conditions. To do so, an external pressure and temperature bath is applied to atoms comprising the system's convex hull, rather than the interior region. This method, the Langevin Hull, allows for constant pressure, temperature, or isobaric-isothermal (NPT) simulations of explicitly non-periodic molecular systems. Other major developments are the inclusion of advanced real-space electrostatics for point multipoles, and polarizable force fields using fluctuating charges or fluctuating electron densities.

Reverse Non-Equilibrium Molecular Dynamics

RNEMD methods impose a non-physical (heat, momentum, or particle) flux between different regions of the simulation. In response, the system develops a temperature, velocity, or concentration gradient between the two regions, and the linear coefficient connecting applied flux and measured gradient is a transport property of the material. Since the amount of the applied flux is known exactly in RNEMD, and the measurement of gradients is generally straightforward, imposed-flux methods typically take shorter simulation times to obtain converged results for transport properties, when compared with equilibrium MD or forward-NEMD approaches. If an interface lies between the two regions, these methods can also provide *interfacial* transport coefficients by mapping any spatial discontinuities in concentration, velocity, or temperature with the applied flux (Drisko & Gezelter, 2024; Kuang & Gezelter, 2012; Stocker & Gezelter, 2014).

Non-equilibrium molecular dynamics is a well-developed area of research, and OpenMD supports many different RNEMD algorithms. The first is the original "swapping" approach by Müller-Plathe (Müller-Plathe, 1997, 1999). Here, the entire momentum vectors of two particles in separate slabs may be exchanged to generate a thermal flux. Occasionally, non-ideal Maxwell-Boltzmann distributions will develop in velocity profiles using this approach (Tenney & Maginn, 2010). OpenMD also introduces a number of new algorithms which extend the capabilities of RNEMD.

Rather than using momentum exchanges between individual particles in each region, the Non-Isotropic Velocity Scaling (NIVS) algorithm applies velocity scaling to all of the selected particles in both regions (Kuang & Gezelter, 2010). NIVS was shown to be very effective at computing thermal conductivities, but is not suitable for imposing a momentum flux or for computing shear viscosities. However, simultaneous velocity shearing and scaling (VSS) exchanges between the two regions remove all of these limitations (Kuang & Gezelter, 2012). The VSS-RNEMD method yields a simple set of equations which satisfy energy and linear momentum conservation constraints, while simultaneously imposing a desired flux between the two regions. The VSS approach is versatile in that it may be used to implement both thermal and shear transport either separately or simultaneously. OpenMD is also capable of leveraging the VSS method in *non-periodic simulations*, in which the regions have been generalized

to concentric regions of space ([Stocker & Gezelter, 2014](#)), allowing for simulations of heat transport away from nanostructures. In the following section, we explore the algorithm that makes non-periodic boundary simulations possible in OpenMD.

Another novel RNEMD algorithm allows for particle positions to be swapped between RNEMD regions resulting in an applied *particle flux*. The scaled particle flux (SPF) method accurately calculates diffusion coefficients while maintaining energy and linear momentum constraints, and can map the temperature dependence of diffusion when used in tandem with a thermal flux in VSS-RNEMD. SPF-RNEMD has also been applied to interfacial systems of nanoporous graphene in a molecular fluid. In this case, permeabilities were computed by imposing a molecular flux between regions on opposite sides of the membrane and measuring both the hydraulic and osmotic pressure that develops as a result of this flux ([Drisko & Gezelter, 2024](#)).

Langevin Hull

In many molecular simulations, systems have near-uniform compressibility, and OpenMD implements a range of integrators to sample the isothermal-isobaric (NPT) ensemble using the Nosé-Hoover-Andersen equations of motion. These integrators implement various forms of affine scaling to provide isotropic (NPTi) or fully-flexible (NPTf) scaling motions of the periodic box ([Andersen, 1980](#); [Hoover, 1986](#); [Sturgeon & Laird, 2000](#)). Additional constant pressure integrators use restricted affine scaling to enforce constant surface area (NPAT), constant surface tension (NPγT), or even orthorhombic box geometries (NPTxyz). For systems comprising materials of different compressibilities, such as a solvated nanoparticle, scaled coordinate transformations may cause numerical instability or poor volume sampling depending on the strength of the applied barostat. Users may also wish to represent systems without the explicit periodicity required by box-scaling constant pressure methods. For example, proteins may be in artificially crowded environments if periodic box simulations are required.

To address these problems, OpenMD implements a method called the Langevin Hull which samples the isothermal-isobaric (NPT) ensemble for non-periodic systems ([Vardeman et al., 2011](#)). The method, based on Langevin dynamics, couples an external bath at constant pressure, temperature, and effective solvent viscosity to the atomic sites on a dynamically-computed convex hull. This convex hull is defined as the set of facets that have no concave corners at an atomic site ([Edelsbrunner & Mücke, 1994](#)). The hull is computed using Delaunay triangulation between coplanar neighbors ([Delaunay, 1934](#); [Lee & Schachter, 1980](#)). These computations are performed by the external Qhull library ([Barber et al., 1996](#)), and are computed each time step, allowing molecules to move freely between the inner region and outer hull. Atoms in the interior evolve according to Newtonian mechanics. The equations of motion for sites on the hull,

$$m_i \dot{\mathbf{v}}_i(t) = -\nabla_i U + \mathbf{F}_i^{\text{ext}} \tag{2}$$

$$= -\nabla_i U + \sum_f \frac{1}{3} \left(-\hat{\mathbf{n}}_f P A_f - \Xi_f(t) \left(\frac{1}{3} \sum_{i=1} \mathbf{v}_i \right) + \mathbf{R}_f(t) \right)$$

include additional forces from the external bath. Each facet f on the convex hull has a contribution from a pressure bath acting in proportion to the facet's surface area and in the direction of the surface normal $(-\hat{\mathbf{n}}_f P A_f)$. The facets of the hull are also in contact with an implicit solvent which provides a drag on the velocity of the facet according to an approximate resistance tensor, $(-\Xi_f \mathbf{v}_f)$ and which also kicks the facet via a Gaussian random force $(\mathbf{R}_f)$.

Computation of a convex hull is $\mathcal{O}(N \log N)$ for sequential machines and remains the performance bottleneck for parallelization. In parallel, the global convex hull is computed using the union of sites from all local (processor-specific) hulls. Testing and validation for this method were carried out on three unique systems, a gold nanoparticle and an SPC/E water droplet (both with uniform compressibilities), and a gold nanoparticle solvated in SPC/E water (non-uniform compressibility), shown in Fig. [1](#). This method was shown to work well across

all test systems (Vardeman et al., 2011), and remains the preferred method of simulating nanoparticles in the isothermal-isobaric (NPT) ensemble with OpenMD.
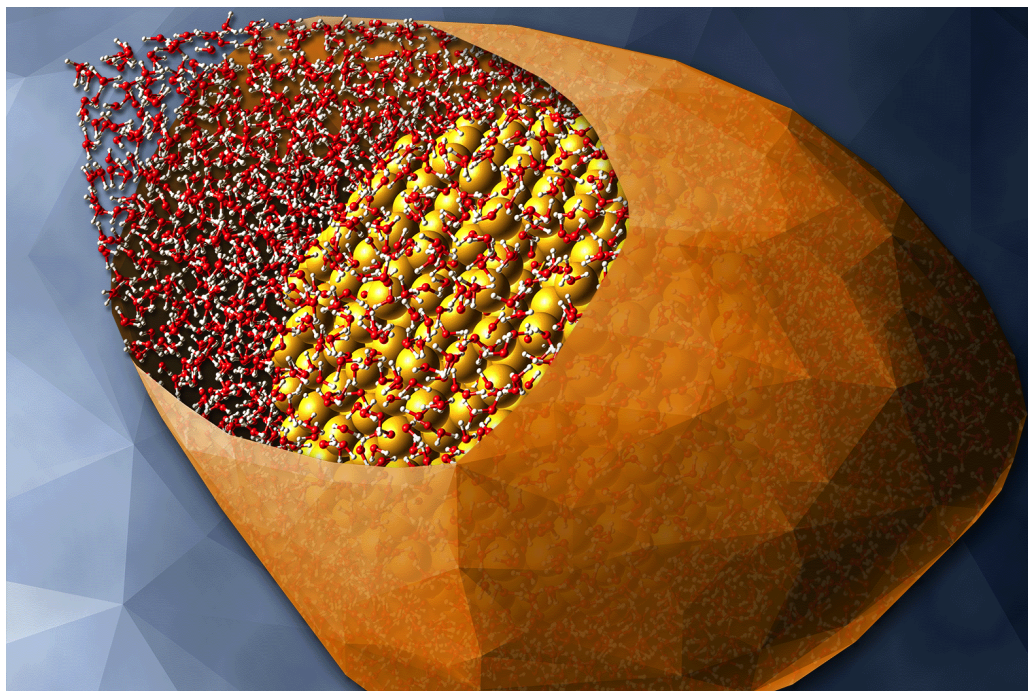

Figure 1: A Langevin Hull surrounds explicit water solvating a gold nanoparticle. The Langevin Hull imposes an external pressure and temperature bath and maintains isobaric-isothermal conditions without periodic boundaries. (Image created with the help of Dr. Kristina Davis from the Notre Dame Center for Research Computing.)

Real Space Electrostatics

Electrostatic interactions are one of the most important intramolecular forces and are present in all but the most basic of molecular simulations. These interactions are also long ranged, and are typically the most computationally expensive. As a result, significant effort has gone into to balancing the accuracy and efficiency of these calculations. OpenMD implements a number of techniques which are generally classified according to how solvent molecules are incorporated into the systems of interest. Implicit methods, which exclude solvent molecules from the simulation, offer computational efficiency at the cost of accuracy. One example would be the use of a reaction field (Onsager, 1936) for electrostatics coupled with Langevin dynamics to include the hydrodynamic effects of the solvent. Explicit methods which include all solvent molecules directly are the most widely used with OpenMD. Explicit electrostatic methods can further be classified as either *Real Space* or *Ewald*-based methods.

The default electrostatics summation method used in OpenMD is a real space, damped-shifted force (DSF) model (Fennell & Gezelter, 2006) which extends and combines the standard shifted potentials of Wolf et al. (1999) and the damped potentials of Zahn et al. (2002). The potential due to the damped-shifted force has the form:

$$U_{\text{Coulomb}} = \frac{1}{4\pi\epsilon_0} \left[\sum_i \sum_{j>i} U_{\text{DSF}}(q_i, q_j, r_{ij}) + \sum_i U_i^{\text{self}}(q_i) \right] \tag{3}$$

where the damped shifted force potential,

$$U_{\text{DSF}}(q_i, q_j, r_{ij}) = \begin{cases} q_i q_j \left[f(r_{ij}) - f(R_c) - f'(R_c)(r_{ij} - R_c) \right], & r_{ij} \leq R_c \\ 0, & r_{ij} > R_c \end{cases} \tag{4}$$

Drisko et al. (2024). OpenMD: A parallel molecular dynamics engine for complex systems and interfaces. *Journal of Open Source Software*, *9*(103), 7004. https://doi.org/10.21105/joss.07004

cuts off smoothly as $r_{ij} \to R_c$, and the Coulombic kernel is damped using a complementary error function, $f(r) = \frac{\text{erfc}(\alpha r)}{r}$. The shifted potential term can be thought of as the interactions of charges with neutralizing charges on the surface of the cutoff sphere. The damping parameter (α) can be specified directly in OpenMD, or is set by default from the cutoff radius, $\alpha = 0.425 - 0.02 R_c$, where the code switches to an undamped kernel for $R_c > 21.25$ Å. The self potential represents the interaction of each charge with its own neutralizing charge on the cutoff sphere, and this term resembles the self-interaction in the Ewald sum,

$$U_i^{\text{self}}(q_i) = -\left(\frac{\text{erfc}(\alpha R_c)}{R_c} + \frac{\alpha}{\pi^{1/2}} \right) q_i^2. \tag{5}$$

DSF offers an attractive compromise between the computational efficiency of *Real Space* methods $(\mathcal{O}(N))$ and the accuracy of the full Ewald sum (Fennell & Gezelter, 2006). The DSF method has also been extended for use with point dipoles and quadrupoles as *Gradient Shifted* and *Taylor Shifted* potentials (Lamichhane, Gezelter, et al., 2014) and has been validated for potentials and atomic forces (Lamichhane, Newman, et al., 2014), as well as dielectric properties (Lamichhane et al., 2016), against the full Ewald sum. Note that the Ewald method was extended to point multipoles up to quadrupolar order (Smith, 1982, 1998), and this has also been implemented in OpenMD. The *Shifted Force* potential generalizes most directly as the *Gradient Shifted* potential for multipoles, and these are the default electrostatic summation methods in OpenMD.

Fluctuating Charges and Densities

One way to include the effects of polarizability in molecular simulations is to use electronegativity equalization (Rappé & Goddard, 1991) or fluctuating charges on atomic sites (Rick et al., 1994). OpenMD makes it relatively easy to add extended variables (e.g. charges) on sites to support these methods. In general, the equations of motion are derived from an extended Lagrangian,

$$\mathcal{L} = \sum_{i=1}^{N} \left[\frac{1}{2} m_i \dot{\mathbf{r}}_i^2 + \frac{1}{2} M_q \dot{q}_i^2 \right] - U(\{\mathbf{r}\}, \{q\}) - \lambda \left(\sum_{i=1}^{N} q_i - Q \right) \tag{6}$$

where the potential depends on both atomic coordinates and the dynamic fluctuating charges on each site. The final term in Eq. 6 constrains the total charge on the system to a fixed value, and M_q is a fictitious charge mass that governs the speed of propagation of the extended variables.

A relatively new model for simulating bulk metals, the density readjusting embedded atom method (DR-EAM), has also been implemented (Bhattarai et al., 2019). DR-EAM allows fluctuating densities within the framework of the Embedded Atom Method (EAM) (Daw & Baskes, 1984), by adding an additional degree of freedom, the charge for each atomic site. The total configurational potential energy, U, as a function of both instantaneous positions, $\{\mathbf{r}\}$, and partial charges $\{q\}$:

$$U_{\text{DR-EAM}}(\{\mathbf{r}\}, \{q\}) = \sum_i F_i[\bar{\rho}_i] + \frac{1}{2} \sum_i \sum_{j \neq i} \phi_{ij}(r_{ij}, q_i, q_j)$$
$$+ \frac{1}{2} \sum_i \sum_{j \neq i} q_i q_j J(r_{ij}) + \sum_i U_i^{\text{self}}(q_i) \tag{7}$$

where the cost of embedding atom i in a total valence density of $\bar{\rho}_i$ is computed using the embedding functional, $F_i[\bar{\rho}_i]$. ϕ_{ij} is the pair potential between atoms i and j, and $J(r_{ij})$ is the Coulomb integral (which can be computed using the DSF approximation above). Lastly, U_{self} is an additional self potential, modeled as a sixth-order polynomial parameterized by

electron affinities and ionization potentials for a wide range of metals,

$$U_i^{\mathrm{self}}(q_i) = \sum_{n=1}^{6} a_n q_i^n. \tag{8}$$

The contribution to the local density at site i depends on the instantaneous partial charges on all other atoms,

$$\bar{\rho}_i = \sum_{j \neq i} \left(1 - \frac{q_j}{N_j} \right) f_j(r_{ij}) \tag{9}$$

with N_j as the valency count for atom j. Modifications to the pair potential used in DR-EAM are also supported.

DR-EAM was shown to perform well for bulk metals, metal surfaces, and alloys; and most importantly, it retains the strengths of the unmodified EAM in modeling bulk elastic constants and surface energies (Bhattarai et al., 2019). DR-EAM has similar performance to the unmodified EAM in that both approaches require a double-pass through the force loop, once to compute local densities and again to compute forces and energies. We note that the infrastructure required to implement DR-EAM is a superset of what is required for common *fluc-q* potentials like the TIP4P-FQ model for water (Rick et al., 1994).

Acknowledgements

References

Abraham, M. J., Murtola, T., Schulz, R., Páll, S., Smith, J. C., Hess, B., & Lindahl, E. (2015). GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX*, *1*, 19–25. https://doi.org/10.1016/j.softx.2015.06.001

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

Andersen, H. C. (1980). Molecular dynamics simulations at constant pressure and/or temperature. *The Journal of Chemical Physics*, *72*(4), 2384–2393. https://doi.org/10.1063/1.439486

Barber, C. B., Dobkin, D. P., & Huhdanpaa, H. (1996). The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, *22*(4), 469–483. https://doi.org/10.1145/235815.235821

Barker, M., Chue Hong, N. P., Katz, D. S., Lamprecht, A.-L., Martinez-Ortiz, C., Psomopoulos, F., Harrow, J., Castro, L. J., Gruenpeter, M., Martinez, P. A., & Honeyman, T. (2022). Introducing the FAIR principles for research software. *Scientific Data*, *9*(1), 622. https://doi.org/10.1038/s41597-022-01710-x

Bhattarai, H., Newman, K. E., & Gezelter, J. D. (2019). Polarizable potentials for metals: The density readjusting embedded atom method (DR-EAM). *Physical Review B*, *99*(9), 94106. https://doi.org/10.1103/PhysRevB.99.094106

Brooks, B. R., Brooks III, C. L., Mackerell Jr., A. D., Nilsson, L., Petrella, R. J., Roux, B., Won, Y., Archontis, G., Bartels, C., Boresch, S., Caflisch, A., Caves, L., Cui, Q., Dinner, A. R., Feig, M., Fischer, S., Gao, J., Hodoscek, M., Im, W., … Karplus, M. (2009). CHARMM: The biomolecular simulation program. *Journal of Computational Chemistry*, *30*(10), 1545–1614. https://doi.org/10.1002/jcc.21287

Case, D. A., Aktulga, H. M., Belfon, K., Cerutti, D. S., Cisneros, G. A., Cruzeiro, V. W. D., Forouzesh, N., Giese, T. J., Götz, A. W., Gohlke, H., Izadi, S., Kasavajhala, K., Kaymak, M. C., King, E., Kurtzman, T., Lee, T.-S., Li, P., Liu, J., Luchko, T., … Merz, K. M. Jr. (2023). AmberTools. *Journal of Chemical Information and Modeling*, *63*(20), 6183–6191. https://doi.org/10.1021/acs.jcim.3c01153

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

Delaunay, B. (1934). Sur la sphère vide: A la mémoire de Georges Voronoï. *Bulletin of the Academy of Sciences of the USSR VII: Classe Des Sciences Mathématiques Et Naturelles*, 793–800. http://mi.mathnet.ru/im4937

Drisko, C. R., & Gezelter, J. D. (2024). A reverse nonequilibrium molecular dynamics algorithm for coupled mass and heat transport in mixtures. *Journal of Chemical Theory and Computation*, *20*(12), 4986–4997. https://doi.org/10.1021/acs.jctc.4c00182

Eastman, P., Swails, J., Chodera, J. D., McGibbon, R. T., Zhao, Y., Beauchamp, K. A., Wang, L.-P., Simmonett, A. C., Harrigan, M. P., Stern, C. D., Wiewiora, R. P., Brooks, B. R., & Pande, V. S. (2017). OpenMM 7: Rapid development of high performance algorithms for molecular dynamics. *PLoS Computational Biology*, *13*(7), 1–17. https://doi.org/10.1371/journal.pcbi.1005659

Edelsbrunner, H., & Mücke, E. P. (1994). Three-dimensional alpha shapes. *ACM Transactions on Graphics*, *13*(1), 43–72. https://doi.org/10.1145/174462.156635

Fennell, C. J., & Gezelter, J. D. (2006). Is the Ewald summation still necessary? Pairwise alternatives to the accepted standard for long-range electrostatics. *The Journal of Chemical Physics*, *124*(23), 234104. https://doi.org/10.1063/1.2206581

Hoover, W. G. (1986). Constant-pressure equations of motion. *Physical Review A*, *34*(3), 2499–2500. https://doi.org/10.1103/physreva.34.2499

Kuang, S., & Gezelter, J. D. (2010). A gentler approach to RNEMD: Nonisotropic velocity scaling for computing thermal conductivity and shear viscosity. *The Journal of Chemical Physics*, *133*(16), 164101. https://doi.org/10.1063/1.3499947

Kuang, S., & Gezelter, J. D. (2012). Velocity shearing and scaling RNEMD: A minimally perturbing method for simulating temperature and momentum gradients. *Molecular Physics*, *110*(9-10), 691–701. https://doi.org/10.1080/00268976.2012.680512

Lamichhane, M., Gezelter, J. D., & Newman, K. E. (2014). Real space electrostatics for multipoles. I. Development of methods. *The Journal of Chemical Physics*, *141*(13), 134109. https://doi.org/10.1063/1.4896627

Lamichhane, M., Newman, K. E., & Gezelter, J. D. (2014). Real space electrostatics for multipoles. II. Comparisons with the Ewald sum. *The Journal of Chemical Physics*, *141*(13), 134110. https://doi.org/10.1063/1.4896628

Lamichhane, M., Parsons, T., Newman, K. E., & Gezelter, J. D. (2016). Real space electrostatics for multipoles. III. Dielectric properties. *The Journal of Chemical Physics*, *145*(7), 74108. https://doi.org/10.1063/1.4960957

Lee, D. T., & Schachter, B. J. (1980). Two algorithms for constructing a Delaunay triangulation. *International Journal of Computer & Information Sciences*, *9*(3), 219–242. https://doi.

org/10.1007/BF00977785

Meineke, M. A., Vardeman II, C. F., Lin, T., Fennell, C. J., & Gezelter, J. D. (2005). OOPSE: An object-oriented parallel simulation engine for molecular dynamics. *Journal of Computational Chemistry*, *26*(3), 252–271. https://doi.org/10.1002/jcc.20161

Müller-Plathe, F. (1997). A simple nonequilibrium molecular dynamics method for calculating the thermal conductivity. *The Journal of Chemical Physics*, *106*(14), 6082–6085. https://doi.org/10.1063/1.473271

Müller-Plathe, F. (1999). Reversing the perturbation in nonequilibrium molecular dynamics: An easy way to calculate the shear viscosity of fluids. *Physical Review E*, *59*(5), 4894–4898. https://doi.org/10.1103/PhysRevE.59.4894

Onsager, L. (1936). Electric moments of molecules in liquids. *Journal of the American Chemical Society*, *58*(8), 1486–1493. https://doi.org/10.1021/ja01299a050

Rappé, A. K., & Goddard, W. A. I. (1991). Charge equilibration for molecular dynamics simulations. *The Journal of Physical Chemistry*, *95*(8), 3358–3363. https://doi.org/10.1021/j100161a070

Rick, S. W., Stuart, S. J., & Berne, B. J. (1994). Dynamical fluctuating charge force fields: Application to liquid water. *The Journal of Chemical Physics*, *101*(7), 6141–6156. https://doi.org/10.1063/1.468398

Smith, W. (1982). Point multipoles in the Ewald summation. *CCP5 Information Quarterly*, *4*, 13–25. https://www.ccp5.ac.uk/newsletter-1982

Smith, W. (1998). Point multipoles in the Ewald summation (revisited). *CCP5 Information Quarterly*, *46*, 15–25. https://www.ccp5.ac.uk/newsletter-1998

Stocker, K. M., & Gezelter, J. D. (2014). A method for creating thermal and angular momentum fluxes in nonperiodic simulations. *Journal of Chemical Theory and Computation*, *10*(5), 1878–1886. https://doi.org/10.1021/ct500221u

Sturgeon, J. B., & Laird, B. B. (2000). Symplectic algorithm for constant-pressure molecular dynamics using a Nosé-Poincaré thermostat. *The Journal of Chemical Physics*, *112*(8), 3474–3482. https://doi.org/10.1063/1.480502

Tenney, C. M., & Maginn, E. J. (2010). Limitations and recommendations for the calculation of shear viscosity using reverse nonequilibrium molecular dynamics. *The Journal of Chemical Physics*, *132*(1), 14103. https://doi.org/10.1063/1.3276454

Thompson, A. P., Aktulga, H. M., Berger, R., Bolintineanu, D. S., Brown, W. M., Crozier, P. S., in 't Veld, P. J., Kohlmeyer, A., Moore, S. G., Nguyen, T. D., Shan, R., Stevens, M. J., Tranchida, J., Trott, C., & Plimpton, S. J. (2022). LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Computer Physics Communications*, *271*, 108171. https://doi.org/10.1016/j.cpc.2021.108171

Vardeman, C. F. I., Stocker, K. M., & Gezelter, J. D. (2011). The Langevin Hull: Constant pressure and temperature dynamics for nonperiodic systems. *Journal of Chemical Theory and Computation*, *7*(4), 834–842. https://doi.org/10.1021/ct100670m

Wolf, D., Keblinski, P., Phillpot, S. R., & Eggebrecht, J. (1999). Exact method for the simulation of Coulombic systems by spherically truncated, pairwise r^{-1} summation. *The Journal of Chemical Physics*, *110*(17), 8254–8282. https://doi.org/10.1063/1.478738

Zahn, D., Schilling, B., & Kast, S. M. (2002). Enhancement of the Wolf damped Coulomb potential: Static, dynamic, and dielectric properties of liquid water from molecular simulation. *The Journal of Physical Chemistry B*, *106*(41), 10725–10732. https://doi.org/10.1021/jp025949h