

Recent Advances for Improving the Accuracy, Transferability and Efficiency of Reactive Force Fields

Itai Leven,^{†,‡} Hongxia Hao,^{†,‡} Songchen Tan,[¶] Xingyi Guan,^{†,‡} Katheryn A.
Penrod,[§] Dooman Akbarian,[§] Benjamin Evangelisti,[§] Md Jamil Hossain,[§] Md
Mahbubul Islam,^{||} Jason Koski,[⊥] Stan Moore,[⊥] Hasan Metin Aktulga,[#] Adri
van Duin,[§] and Teresa Head-Gordon^{*,†,‡,@}

[†]*Pitzer Center for Theoretical Chemistry, Department of Chemistry, University of
California, Berkeley 94720*

[‡]*Chemical Sciences Division, Lawrence Berkeley National Laboratory*

[¶]*College of Chemistry and Molecular Engineering, Peking University, China*

[§]*Department of Mechanical Engineering, Chemical Engineering, Engineering Science and
Mechanics, Chemistry, Materials Science and Engineering Penn State University 240
Research East University Park PA 16802*

^{||}*Department of Mechanical Engineering, Wayne State University, Detroit, Michigan
48202, United States*

[⊥]*Sandia National Laboratories Albuquerque, NM 87185-1315*

[#]*Department of Computer Science and Engineering, Michigan State University, East
Lansing, Michigan 48824*

[@]*Departments of Bioengineering and Chemical and Biomolecular Engineering, University
of California, Berkeley, California 94720*

E-mail: thg@berkeley.edu

Abstract

Reactive force fields provide an affordable model for simulating chemical reactions at a fraction of the cost of quantum mechanical approaches. However classically accounting for chemical reactivity often comes at the expense of accuracy and transferability, while computational cost is still large relative to non-reactive force fields. In this Perspective we summarize recent efforts for improving the performance of reactive force fields in these three areas with a focus on the ReaxFF theoretical model. To improve accuracy we describe recent reformulations of charge equilibration schemes to overcome unphysical long-range charge transfer, new ReaxFF models that account for explicit electrons, and corrections for energy conservation issues of the ReaxFF model. To enhance transferability we also highlight new advances to include explicit treatment of electrons in the ReaxFF and hybrid non-reactive/reactive simulations that make it possible to model charge transfer, redox chemistry, and large systems such as reverse micelles within the framework of a reactive force field. To address the computational cost we review recent work in extended Lagrangian schemes and matrix preconditioners for accelerating the charge equilibration method component of ReaxFF and improvements in its software performance in LAMMPS.

Introduction

Reactive force fields offer an efficient model for simulating chemical reactions within the framework of a classical description, enabling the simulation of time and length scales which are orders of magnitude higher than purely quantum mechanical approaches. One of the more popular reactive force fields is ReaxFF, which employs a distance-dependent formalism to bond-order changes for chemical reactivity within a fully classical model.[11] ReaxFF has been applied to a wide range of research fields including surface science[36, 43, 45], energetic materials[14, 28, 41], electrolysis[31, 42, 56], and biochemistry.[23, 29, 44]

A key feature of ReaxFF is its ability to account for charge rearrangement as bonds

are broken and formed. This has been historically achieved through the electronegativity equalization method (EEM) originally proposed by Mortier et al. over three decades ago[3, 4] and later refined by Rappe et al and termed the charge equilibration method (CEM) or charge equalization (QEq)).[7]. However, the simplifications associated with classically accounting for the electronic degrees of freedom using EEM in ReaxFF has notable shortcomings in some applications[30, 53] including unphysical long-range charge transfer, non-integer molecular charge at large separations, lack of out-of-plane polarization, lack of energy conservation, poor parameterization, and lack of transferability. In addition, while ReaxFF offers an affordable method with respect to quantum chemistry, it is still substantially more expensive than non-reactive force fields and therefore increasing its efficiency is highly desirable.[21, 22, 58] This computational cost arises in part from the EEM methods that require solving a set of linear equations self consistently, in which previous work has shown that EEM can account for $\sim 80\%$ of the computational time for ReaxFF applied to large systems and tight convergence tolerance.[22, 51]

Here we consider recent advancements that have been developed in order to address some of these issues. Methods such as the split-charge equilibration (SQE)[17] and the atom condensed Kohn Sham DFT approximated to second order (ACKS2)[24] were designed to impede unphysical long-range charge transfer, and new reactive force field models that account for electrons and electron density more explicitly have improved overall accuracy, including the the coarse-grained electron model (C-GeM)[51, 57] and explicit electron ReaxFF (eReaxFF) model[32] which are also designed to be more transferable. We illustrate this progress in applications including Ag nanoclusters[54], hydrocarbons,[32], lithium ion batteries[33], electrical breakdown of polyethylene[63], and water in different condensed phase environments.[51, 57] We also consider the benefit derived from hybrid models in which C-GeM is embedded in a classical electrostatic force field to describe proton hopping mechanisms in the water pool of reverse micelles. Finally, we review the recent efforts that have been devoted to reducing the computational cost of EEM, including matrix precondition-

ers[21, 46, 58] and extended Lagrangian schemes[51, 61], largely which have fully eliminated the cost of the SCF altogether, as well as better optimized performance of these methods in community codes such as LAMMPS.

Bond Order Reactive Force Fields

Bond order reactive force fields are targeted toward describing electronic rearrangements involving chemical reactivity within classical models. Some of the early incarnations of reactive force fields include the Tersoff[5], REBO[6], and AIREBO[13] potentials that were developed to describe the different phases of hydrocarbons within one theoretical framework. Subsequently models such as the ReaxFF[11] and COMB[18] potentials have expanded the chemical space for reactivity beyond just hydrocarbons, and advanced their applicability by combining the bond order terms with charge equilibration methods[4, 7] to describe the charge rearrangement as bonds break and form. Here we review the ReaxFF method specifically to illustrate the ideas behind bond order reactive force fields.

In ReaxFF, the total energy is decomposed into various partial energy contributions, similar to non-reactive force fields. Unlike the non-reactive force field counterparts which use fixed bond connectivity, ReaxFF computes bond orders as a function of interatomic distance. These bond orders are then incorporated into flexible bonded terms that contribute to the total ReaxFF energy and forces. In addition, ReaxFF computes non-bonded interactions using shielded Coulomb and van der Waals (vdW) potentials.

More specifically, the partial energy contributions to the ReaxFF total energy are as follows:

$$E_{ReaxFF} = E_{bond} + E_{over} + E_{under} + E_{lp} + E_{val} + E_{tors} + E_{vdW} + E_{Coul} + E_{Specific} \quad (1)$$

The first six terms on the right-hand side of Eq. 1 correspond to the flexible bond-order-dependent energy terms. The first of these terms, E_{bond} , captures the energy from the

breaking and formation of chemical bonds in the system. To compute the second, third, and fourth terms, the ReaxFF bond orders are compared to the valency expected based on atom type. The second term, E_{over} , is an energetic penalty applied to overcoordinated atoms. Analogously, E_{under} , is an energetic stabilization applied to undercoordinated atoms. The fourth term, E_{lp} , represents the energy from lone-pair electrons. The fifth and sixth terms, E_{val} and E_{tors} , describe the three-body valence angle strain and the four-body torsional angle strain, respectively. These bonded terms are followed by the non-bonded vdW and Coulombic contributions (E_{VDW} and $E_{Coulomb}$), which are computed for all atom pairs regardless of connectivity. The final term represents a force-field-specific parameter used to account for additional energy contributions for a particular system of interest (i.e. conjugation, hydrogen bonding, C2 corrections, system-specific torsions). We review the functional forms of some of these partial energy contributions, and they are also described in more detail in previous publications.[11, 19, 38]

In ReaxFF, the bond order is calculated from interatomic distances using the following continuous equation:

$$BO_{ij} = BO_{ij}^{\sigma} + BO_{ij}^{\pi} + BO_{ij}^{\pi\pi} = \exp[p_{bo1}(\frac{r_{ij}}{r_0^{\sigma}})^{p_{bo2}}] + \exp[p_{bo3}(\frac{r_{ij}}{r_0^{\pi}})^{p_{bo4}}] + \exp[p_{bo5}(\frac{r_{ij}}{r_0^{\pi\pi}})^{p_{bo6}}] \quad (2)$$

where BO_{ij} corresponds to the bond order between atoms i and j , r_{ij} is the interatomic distance between those atoms, r_0 terms represent equilibrium bond lengths for each bond type (sigma, pi, double pi), and p_{bo} 's are empirical fitting parameters. Because Eq. 2 is continuous, it allows for smooth transitions between sigma, pi, and double pi bond character while yielding a differentiable potential energy surface (PES) that allows for the computation of interatomic forces. When uncorrected, however, this equation often overestimates bond orders between non-bonded neighbors and produces unphysical coordination numbers for carbon atoms. To address these issues, a local overcoordination correction is performed on all 1-3 nearest neighbor pairs and over/under-coordination terms (E_{over} and E_{under}) are

included in the total energy equation (Eq. 1). Over/under-coordination is determined by comparing the bond order to the valency of the atoms involved in the bond. If the corrected bond order exceeds the valency of the atom, an energetic penalty for overcoordination is incurred. If the valency is larger, the system is stabilized by the E_{under} term.

The ReaxFF workflow begins with the user-defined atomic positions and atom type designations (i.e. C, H, O). From these atomic positions, E_{VDW} is calculated between all atom pairs. The Morse potential utilized for calculation of this term is shielded to prevent excessive repulsions between bonded atoms or next-nearest neighbors. Using the EEM,[3] the partial charge for each atom is then computed based on atomic positions, electronegativity, and hardness (and described more fulsomely in the next Section). Once partial charges have been determined, $E_{Coulomb}$ can be calculated using a shielded Coulomb potential computed between all atom pairs. The remaining partial energy contributions are bond-order-dependent. Bond orders are computed for each atom using Eq. 2 and are then corrected for local overcoordination. The corrected bond orders are used to determine E_{bond} , E_{over} , E_{under} , and E_{lp} . From these corrected bond orders, the valence and torsion angles are enumerated, and their respective energetic contributions (E_{val} and E_{tors}) are calculated. Finally, these partial energy contributions are summed (along with any system-specific terms included in $E_{specific}$) and the total ReaxFF energy and forces are complete for the current time step iteration.[11, 37]

In regards to transferability, when using a ReaxFF force field, careful consideration of how the force field was trained is needed to assess its viability for the user’s application. No ReaxFF parameterization, which typically utilizes QM data at the Density Functional Theory (DFT) level, should be used as a "black box" for their application just because it incorporates interactions involving atomic elements of interest. To illustrate, the standard LDA/GGA density functional approximation are not sufficient to calculate the heat of formation of a metal oxide. This is strikingly apparent in nickel oxide where the GGA will predict that Ni_2O_3 (the Ni^{3+} oxidation state of nickel) is more stable than NiO (the Ni^{2+} ox-

idation state of nickel), which is not observed experimentally. Furthermore, the LDA/GGA approximation underpredicts both the band gap and magnetic moment of NiO. In this example, a ReaxFF parameterization that is not fit to additional data such as surface energy or heat of formation would not be a good candidate for an application relying on stable crystal structures.

In summary, careful examination needs to be taken of how the ReaxFF force field was parameterized before using it in an application of choice. A good resource for such examination is the publication that describes the development and intended applications of the force field and using good chemical judgement. Later we address some recent efforts to improve transferability to overcome these limitations.

Models of Charge Distributions in Reactive Force Fields

Electronegativity Equalization Method. The EEM builds on the early works of Gasteiger and Marselli which based on the principles of electronegativity equalization, formulated an iterative method for evaluating partial charges in molecules.[1] The EEM method developed by Mortier *et al.*, is theoretically justified from DFT and adopts the Parr and Pearson’s definitions of atomic electronegativity and hardness[2]. [3, 4] Qeq (or CEM) is a variation of the EEM method developed by Rappe and Goddard which provides a definition for the atomic electronegativity and hardness parameters and introduces electrostatic screening through Slater type charge densities. In the context of the ReaxFF model the EEM formulation probably best fits the description of the method used as the parameters are obtained empirically and it doesn’t use Slater type charge densities. Throughout this paper we refer to EEM as the general class of charge distribution algorithms that encompasses variants such as CEM and Qeq.

EEM is based on the second order expansion of the electrostatic energy with respect to

atomic charge.

$$E(q_1, q_2, \dots, q_N) = \sum_{i=1}^N (E_i(0) + \chi_i^0 q_i + \frac{1}{2} \sum_{j=1}^N H_{ij} q_i q_j) \quad (3)$$

$$H_{ij} = J_{ii}^0 \delta_{ij} + J_{ij} (1 - \delta_{ij}) \quad (4)$$

where N is the total number of atoms in the system, $E_i(0)$ is the charge reference point at $q=0$ for atom i , $\chi_i^0 = (\frac{\partial E_i}{\partial Q_i})$ is the electronegativity of atom i , q_i and q_j are the partial charges of atoms i and j respectively, $J_{ii}^0 = \frac{\partial^2 E_i}{\partial Q_i^2}$ is the atomic hardness of atom i , $J_{ij} = \frac{1.0}{[r_{ij}^3 + (\frac{1}{\gamma_{ij}})^3]^{\frac{1}{3}}}$ is a shielded Coulomb potential where γ_{ij} is an electrostatic shielding parameter. It essentially poses the charge distribution problem as an optimization problem where the objective is to minimize the electrostatic energy under the net system charge constraint.

To solve for the partial charges in the stated optimization problem, we use the derivative of Eq(3) with respect to atomic charge:

$$\chi_i(q_1, q_2, \dots, q_N) = \frac{\partial E}{\partial q_i} = \chi_i^0 + J_{ii}^0 q_i + \sum_{j \neq i}^N (J_{ij} q_j) \quad (5)$$

and solve it under the constraint of charge conservation:

$$\sum_{i=1}^N q_i = 0 \quad (6)$$

A Lagrange multiplier μ is utilized to minimize Eq. (3) under the constraint of charge neutrality, yielding:

$$\mu = \frac{\sum_{i=1}^N \sum_{j=1}^N H_{ij}^{-1} (-\chi_j^0)}{\sum_{i=1}^N \sum_{j=1}^N H_{ij}^{-1} (1_j)} = \frac{\sum_{i=1}^N q_i^s}{\sum_{i=1}^N q_i^t} \quad (7)$$

Consequently, two sets of linear equation are solved independently:

$$\sum_{i=1}^N \sum_{j=1}^N H_{ij} (q_i^s) = \sum_{i=1}^N -\chi_i^0$$

$$\sum_{i=1}^N \sum_{j=1}^N H_{ij}(q_i^t) = \sum_{i=1}^N 1_i. \quad (8)$$

Finally, the charge of atom i is determined through the relation:

$$q_i = q_{i,SCF}^s + \mu q_{i,SCF}^t \quad (9)$$

The EEM method has been used for close to 3 decades within reactive force fields, but has significant shortcomings such as unphysical long-range charge transfer, non-linear size dependent polarizability, and non-integer molecular charge. Therefore we review three recent theoretical improvements that are starting to displace the EEM method in ReaxFF.

New alternatives to EEM. Since EEM’s development in the mid 80’s there have been several attempts to remedy EEM’s deficiencies with regards to its unphysical long-range charge transfer and metallic polarizability. Some of the most notable methods are the atom-atom charge transfer method (AACT)[9], split charge equilibration (SQE)[17], and charge transfer with current polarization equalization (QTPIE)[16]. However unlike the original EEM method, these earlier solutions are based on physical intuition as apposed to being derived from quantum chemistry.

The atom-condensed Kohn-Sham DFT to second order (ACKS2)[24] stands out from these previous efforts as it provides an extension of the original EEM method within an atom-in-molecule derivation from Kohn-Sham DFT. Verstraelen and coworkers revealed that EEM was missing non-local contributions of the kinetic energy which was essential for eliminating the unphysical long-range charge transfer. In their derivation, the second order expansion of the kinetic energy introduces the Kohn-Sham linear response matrix (X_{ij}) which can be interpreted as a measure of electron delocalization. Unfortunately, the exact relation between the Kohn-Sham linear response matrix and atomic coordinates is not trivial and simplified approximations are required. For practical purposes the Kohn-Sham linear response matrix between two fragments is defined to exponentially decay to zero at large inter-atomic

separations impeding long-range charge transfer.

$$E_{ACKS2} = \underset{\vec{q}}{\operatorname{argmin}} \left[\sum_{i=1}^N \chi_i^0 q_i + \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N H_{ij} q_i q_j + \underset{\vec{\mu}}{\operatorname{argmax}} \left[\sum_{i=1}^N \mu_i (q_i - q_i^0) + \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N X_{ij} \mu_i \mu_j \right] \right] \quad (10)$$

where μ_i is the atomic Kohn-Sham potential of atom i and X_{ij} is a distance dependent linear response matrix element which determines to what extent atoms i and j transfer charge. In Fig. 1 it is shown that ACKS2 predicts that the charge and charge transfer energy in hydrogen fluoride approach zero at the limit of large separation.

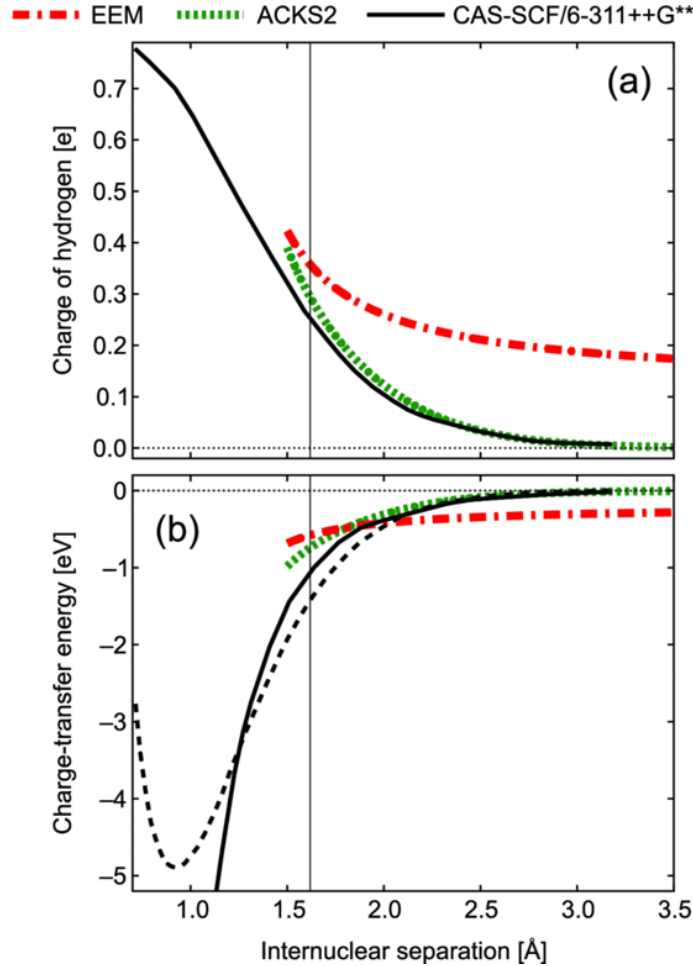


Figure 1: ACKS2 performance for hydrogen fluoride dissociation: (a) Charge of hydrogen as a function of hydrogen-fluoride distance. (b) Charge transfer energy as a function of a hydrogen-fluoride distance. Comparing EEM (red), ACKS2 (green) and CAS-SCF/6-311++G**.

In regards computational expense the ACKS2 model requires solving a $(2N+2) \times (2N+2)$ matrix, and comparisons between the computational performance of EEM and ACKS2 is presented below.

Toward more explicit representations of electrons. The ReaxFF/EEM approach models electrons implicitly in the bonded interactions, which has proven insufficient for several applications.[32] For example, redox reactions modeled with standard ReaxFF impose charge constraints which are problematic for describing charge transfer in reactive environments.[33] ReaxFF also is not able to reproduce electron affinities (EAs) and ionization potentials (IPs) for most chemical species. An accurate description of these electron transport phenomena requires an explicit treatment of electronic degrees of freedom.

The eReaxFF Model. eReaxFF expands upon the standard ReaxFF formalism by incorporating a pseudoclassical explicit electron/hole scheme.[32] The general expression for the eReaxFF total energy is:

$$E_{eReaxFF} = E_{ReaxFF} + E_{nucl-elec} + E_{elec} \quad (11)$$

where partial energy contributions include all standard ReaxFF terms (see Eq. 1) with additional energy terms to account for electron-nucleus interactions ($E_{nucl-elec}$) and electron-electron interactions (E_{elec}). All many-body bonded and nonbonded interaction terms in ReaxFF are retained with modifications made to the functional forms of E_{over} , E_{under} , and E_{lp} to incorporate the explicit electrons/holes. Note that while eReaxFF can mathematically describe holes as well as electrons, so far only eReaxFF descriptions using explicit electrons have been reported as such we will limit the further discussion to electrons.

In eReaxFF, nuclei are treated as point charges and electrons as Gaussian wavefunctions of the form $\psi \propto \exp(-\alpha(r - r')^2)$. These Gaussian wavefunctions interact with the nuclei

through the following pairwise interaction:

$$E_{nucl(i)-elec(j)} = -1/(4\pi\epsilon_0)\beta \sum_{i,j} \frac{Z_i}{R_{ij}} erf(\sqrt{2\alpha}R_{ij}) \quad (12)$$

where Z is the nuclear charge, R_{ij} is the distance between the electron and nucleus, α and β are constants that depend on the atom type. The electron is represented as an additional particle bearing a -1 charge; electron particles interact with all other atomic charges using the same shielded Coulomb interaction as in standard ReaxFF. The number of electrons associated with any single atom is determined by:

$$n_{el} = exp(-p_{val} \cdot R_{ij}^2) \quad (13)$$

where R_{ij} is the distance between the atom-center and the electron and P_{val} is a Gaussian exponent. This enables the electron particle to hop between atoms, thus allowing electron delocalization effects to be modeled by eReaxFF.

One of the biggest drawbacks of the standard ReaxFF formalism is that the bonded and non-bonded interactions are calculated independently. That is, the valency and number of lone-pair (valence) electrons for each atom type are independent of the partial atomic charges. This decoupling of the bond order and atomic charges can give rise to unphysical behavior of ions. For example, based on its atom type, standard ReaxFF would assign a hydrogen cation (H^+) a valency of one, despite its positive charge. By introducing variable valency and coupling the number of lone-pair electrons of an atom to the proximity of explicit electrons, eReaxFF is capable of modeling charge transfer in reactive systems more accurately than ReaxFF. As shown in the flowchart in Fig. 2a, the eReaxFF method retains the basic framework of ReaxFF.

As in the original ReaxFF, the user provides atomic positions and atom type designations (i.e. C, H, O). Additionally, the user defines explicit electrons as “atoms” with their own sets of Cartesian coordinates. In all current implementations, electrons are assigned a mass

of 1 amu in order to facilitate femtosecond-range simulation timesteps. Interactions between electrons and atomic nuclei ($E_{nucl-elec}$) are calculated using Eq. 12. Partial charges in eReaxFF are computed using the ACKS2 method rather than the EEM method employed by standard ReaxFF (see Section 3). Using a shielded Coulomb potential, the electrostatic energy contribution ($E_{Coulomb}$) is computed from the ACKS2 partial charges. All non-bonded interactions have now been accounted for, leaving only bond-order-dependent terms. The bond orders are calculated in the same manner as described above for the original ReaxFF.

The fundamental difference between standard ReaxFF and eReaxFF is the modification of an atom’s valency based on its proximity to an explicit electron. First, the electron occupancy is determined using Eq. 13, and the valency of any affected atoms are modified accordingly. For example, consider a neutral oxygen atom with its six valence electrons and valency of two. Should that oxygen atom gain a seventh valence electron, it may only form one bond, and its valency is reduced to one. This decreased valency results in a larger overcoordination penalty and reduces the bond order associated with the atom. These electron-corrected bond orders are used to compute the energetic contributions from the chemical bonds (E_{bond}), the overcoordination penalty (E_{over}), and the lone pair electrons (E_{lp}). In contrast to ReaxFF, this scheme has enabled the eReaxFF method to counteract the formation of unphysical bonds such as H^+-H^+ due to the higher overcoordination penalty stemming from the loss of valence electron from the H atom. Finally, the partial energy contributions are summed (along with any system-specific terms in $E_{specific}$) and the total eReaxFF energy is reported for the iteration.

The eReaxFF method has been successfully employed to study electron transfer in hydrocarbons[32], solid electrolyte interfaces[33], Ag-metal systems[54], and recently in electrical breakdown in polyethylene (PE)[63]. Islam et al. performed eReaxFF MD simulations on two representative hydrocarbon radicals, $C_{12}H_{19}$ and $C_{14}H_{23}$, at different temperatures, finding that longer aliphatic chain length slows down electron transfer (Fig. 2b).[32] Additionally, they found that an increase in temperature accelerates electron transfer, but decreases

electron localization around the junction. eReaxFF was also employed to study the major reduction reaction pathways of ethylene carbonate (EC) in Li-ion batteries.[33] In this study, eReaxFF properly captured the electron transfer event from Li to EC (forming EC⁻/Li⁺),

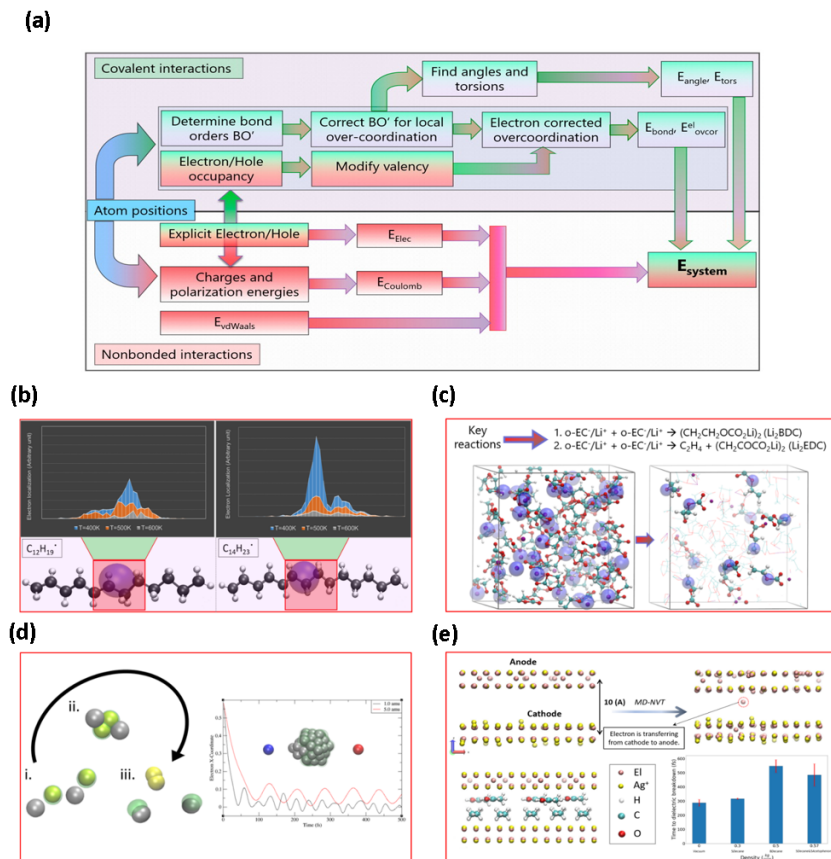


Figure 2: Applications of the eReaxFF: (a) The eReaxFF workflow.[32] (b) Time-averaged electron localization near the contact point of aliphatic and conjugated chains of C₁₂H₁₉ (left) and C₁₄H₂₃ (right) at different temperatures; violet spheres represent the electrons.[32] (c) Snapshot of eReaxFF MD at t = 0 ps (left) and t = 25 ps (right) with produced o-EC/Li⁺ radicals highlighted; Li ions and EC molecules not involved in the electron-transfer events are shown as lines.[33] (d) Reduction mechanism of 2AgCl at elevated temperatures (left), where two sets of AgCl pairs align with one another (i) before Cl (yellow spheres) aggregates to form Cl₂ (ii) while excess electrons (green spheres) are transferred from Cl₂ to Ag⁺ ions (silver spheres), yielding 2 Ag⁰ atoms (iii); plasmonic oscillations (right) of initially-polarized Ag₅₅ between positive (red sphere) and negative (blue sphere) charges at T = 300 K.[54] (e) TDDB of PE (top), where TDDB is taken as the time of first electron transfer from anode to cathode; addition of 5 decane and 5 acetophenone molecules into vacuum between electrodes (bottom left); effect of increased PE density and presence of acetophenone on TDDB (bottom right).[63]

the subsequent C-O bond cleavage (forming o-EC-/Li+ radical), and radical termination reactions (forming dilithium butyl/ethyl dicarbonate) (Fig. 2c).[33] eReaxFF has also been extended to reproduce isomerization energies, EAs, and IEs of small, neutral and charged Ag nanoclusters.[54] This Ag-metal force field introduces a quasi-Drude atomistic description into eReaxFF with every neutral atom being represented by a paired cation and electron (i.e. a neutral Ag atom becomes an Ag+ cation and an explicit electron). With this additional electronic degree of freedom, it becomes possible to model the reduction of Ag^+ to Ag^0 by Cl_2 and the resulting aggregation and growth of the nascent Ag cluster. One novel functionality introduced by the quasi-Drude model is the ability to model plasmonic motion by tracking the movements of the explicit electrons throughout a simulation (Fig. 2d).[54] In another recent study, eReaxFF was used to simulate the electrical breakdown of polyethylene (PE), exploring the roles of byproducts and processing parameters such as density on the time to dielectric breakdown (TDDB) of PE (Fig. 2e).[63] These simulations indicate that an increased PE density enhances the TDDB, while adding species with positive EA (i.e. acetophenone byproducts) reduces the TDDB.

The C-GeM Model. C-GeM is an alternative explicit electron density method recently proposed by Leven and Head-Gordon for treating electrostatics in molecular simulations.[50] In C-GeM, atoms are represented by a positive charged core and a negative charged shell treated as Gaussian charges. In the SCF regime, shells are treated as massless particles which are instantaneously positioned according to the field generated by the cores. The interaction between the cores and shells comprise an electrostatic term:

$$E_{ij}^{elec}(r_{ij}) = \frac{(q_i \cdot q_j)}{r_{ij}} erf\left(\sqrt{\frac{(\alpha_i \cdot \alpha_j)}{(\alpha_i + \alpha_j)}} r_{ij}\right) \quad (14)$$

$$\alpha_i = \frac{\gamma}{(2 * R_i^2)}$$

Where r_{ij} is the distance between core i and shell j , q_i and q_j are the core and shell charges,

R_i and R_j are the core and shell radii and γ is a global parameter, the Gaussian term is given by:

$$E_{ij}^{Gauss}(r_{ij}) = A_i e^{(-\gamma_i r_{ij}^2)} \quad (15)$$

$$\gamma_i = \frac{\omega}{(2 * R_i)}$$

Where A_i is the magnitude of the Gaussian interaction determined by the electronegativity of core i and ω is a global parameter. In C-GeM the sum of the electrostatic and Gaussian terms at complete overlap is defined as the electronegativity of a given atom type as shown in the following expression:

$$\chi_i = E_{ij}^{Gauss}(r_{ij} = 0) + E_{ij}^{elec}(r_{ij} = 0) \quad (16)$$

Where χ_i is the electronegativity of atom i , and $E_{ij}^{Gauss}(r_{ij} = 0)$ and $E_{ij}^{elec}(r_{ij} = 0)$ are the Gaussian and electrostatics components, respectively, at a core-shell distance of $r_{ij} = 0$. The total energy of the system is then given by the sum over all cores and shells:

$$E_{C-GeM} = \sum_i^n \sum_{j<i}^n (E_{ij}^{elec}(r_{ij}) + E_{ij}^{Gauss}(r_{ij})) \quad (17)$$

Here E_{C-GeM} is the total C-GeM energy, n is the number of core+shells in the system and r_{ij} is the distance between particles i and j . Accordingly, in the SCF regime the position of the shells are found by minimizing the following equation:

$$\frac{\delta E_{C-GeM}}{\delta r_{shell}} = 0 \quad (18)$$

C-GeM was first developed to reproduce the EPS of molecules containing C, H, O and Cl elements as demonstrated in Fig. 3a. The model showed good accuracy with an average absolute mean deviation from the DFT EPS of 0.11eV compared to 0.21eV of the EEM and

0.06eV of iterative Hirschfeld partial charges. One of the significant drawbacks of the EEM method is its inability to dissociate molecules to integer charged fragments and unphysical long range charge transfer. In Ref[[50]], it was demonstrated using ab-initio MD trajectories of HCl dissociation in solvated water that C-GeM can account for the dissociation of hydrogen chloride to the ionic products Cl^- and H_3O^+ in solution.

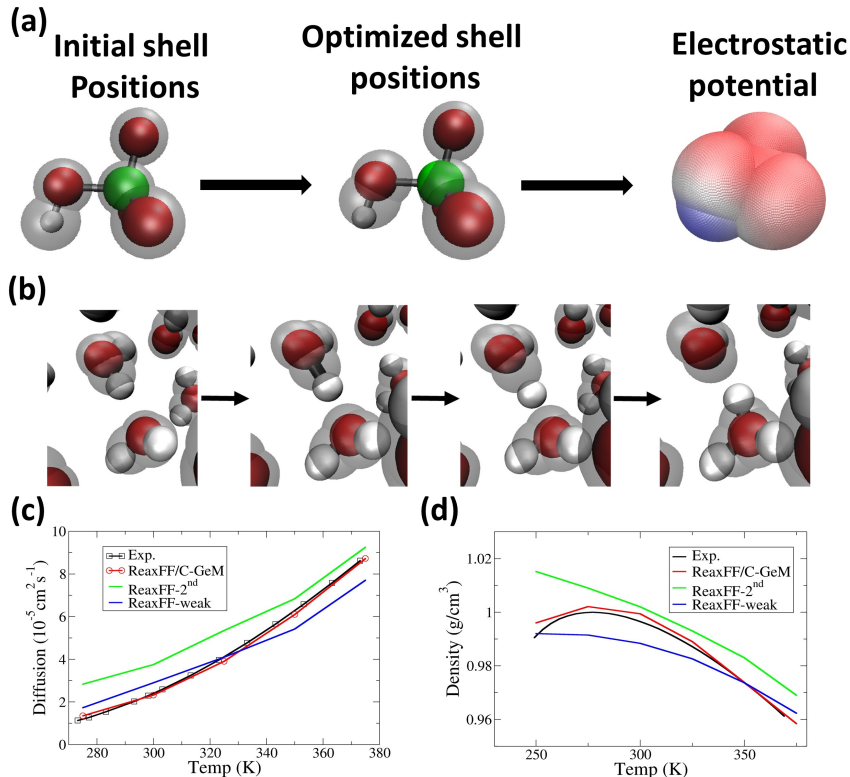


Figure 3: Integration of C-GeM with the ReaxFF: (a) Schematic presentation showing the basic principle behind C-GeM, starting with an initial guess for the shell positions the shells are optimized which are then used to generate the electrostatic potential surface of the molecule. (b) Trajectories from a constrained dynamics simulation showing autodissociation of water to hydronium and hydroxide products using the ReaxFF/C-GeM model. (c) Diffusion of water as a function of temperature comparing the ReaxFF/C-GeM (red) with ReaxFF-2nd (green), ReaxFF-weak (blue) and experiment (black). (d) Density of water as a function of temperature comparing the ReaxFF/C-GeM (red) with ReaxFF-2nd (green), ReaxFF-weak (blue) and experiment (black).

Recently, Leven et al. have developed a reactive water model by integrating C-GeM within the framework of the ReaxFF. One of the advantages in using C-GeM is its ability to correctly assign charges in scenarios involving ionic molecules. Fig.3b. shows constrained

dynamics trajectories of water auto-dissociation using the ReaxFF/C-GeM model presenting the dissociation of water to hydronium and hydroxide ions. In addition, bulk water properties were compared to two recent ReaxFF water force fields, showing superior performance of the ReaxF/C-GeM model. Fig. 3c plots the diffusion as a function of temperature showing great agreement between the ReaxFF/C-GeM model and experiments. Figure 6d plots the density as a function of temperature showing that the ReaxFF/C-GeM model can reproduce the maximum density of water.

Electrostatic Embedding using ReaxFF/C-GeM. The Head-Gordon group has recently advanced a ReaxFF/C-GeM model by performing electrostatic embedding using non-reactive force fields, similar to that of QM/MM methodologies. Such an approach will benefit the high efficiency and reasonable accuracy of non-reactive force fields at long-ranged while still being able to account for chemical reactivity in a local region of the system. Similar to QM/MM methods, atoms are split into three categories: (i) ReaxFF/C-GeM atoms, which include all atoms in the chemically reactive region, (ii) the MM atoms, which include all remaining atoms and is handled exclusively by the non-reactive force field, and (iii) the van der Waals and electrostatic non-bonded interactions between the ReaxFF/C-GeM region and the MM region.

The ReaxFF/C-GeM hybrid model has recently been applied to the study of proton hopping in confined acidic nanometer-sized reverse micelles[62], which is an important model system to help our understanding of charge transport in a large range of chemical and biological processes[39], especially for environments under confinement and at interfaces[20, 40]. The reverse micelle spontaneously form by self-assembly in which the amphiphilic nature of surfactants in a nonpolar solvent organizes to encapsulate a water pool as depicted in Fig. 4. The water pool is encapsuled by the Igepal CO-520 surfactants (shown in dark green) with the hydrophilic head-group and the hydrophobic group is headed to the cyclohexane solvent (light green). The reverse micelles were first equilibrated with classical MD simulation using the general Amber force field (GAFF)[15] to account for the cyclohexane and surfactant

molecules with AM1-BCC[10] charges obtained by ANTECHAMBER[12], and the TIP3P water was used for the equilibration of the water pool. Then the water pool was replaced with the ReaxFF/C-GeM water model and further equilibrated. Then various concentrations of HCl are simulated by replacing waters with ions to reproduce experimental acid concentrations.

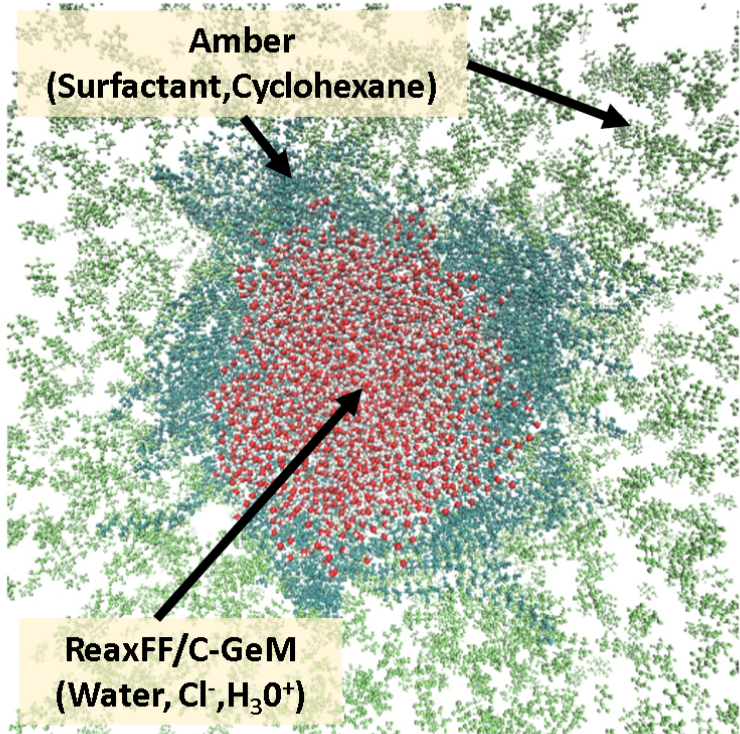


Figure 4: Schematic presentation of modeling a reverse micelle using a hybrid ReaxFF/C-GeM+Amber hybrid model: Snapshot of the reverse micelle with diameter=5.96nm, the water pool is treated with the Reaxff/C-GeM model (oxygens=red and hydrogens=white), surfactants (dark green) and cyclohexane solvent (light green) were treated with the amber force field.

From the ReaxFF/C-GeM reactive molecular dynamics simulations, which were able to reproduce the Tera Hertz and dielectric relaxation data on the exact same systems, we found a surprising change in the proton hopping mechanism as hydronium ion concentration increases. More specifically we find a switch in the dominant Grotthus proton hopping mechanism for small reverse micelles to one in which localized oscillatory hopping emerges

as the dominant mechanism at a critical micelle size also seen experimentally. This change stems from the accumulation of hydronium and chloride ions at the micelle interface which creates a short-circuited hydrogen-bonding network for proton hops. I.e. as the micelle size and/or acid concentration increases, the interfacial water network induces a “traffic jam” that favors the localized oscillatory hopping mechanism.

Software Advances for Reactive Force Fields

The code for the ReaxFF method was originally implemented by van Duin as a sequential Fortran based program that still serves as the reference implementation in regards to accuracy of energy and force calculation. ReaxFF is also available in the open-source Purdue Reactive Molecular Dynamics (PuReMD) software [27] in both distributed memory parallel and multi-GPU parallel versions [21, 34]. In addition to its standalone version, PuReMD has been integrated into the open-source Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) as the USER-ReaxC package. Later, a hybrid OpenMP/MPI parallel version of this package optimized for multi-core architectures, the USER-ReaxC-OMP package, was released in LAMMPS [47]. The ReaxFF method is also available in other important simulation environments, including the Amsterdam Density Functional (ADF) program and Materials Studio released under license by Biovia (<https://www.3dsbiovia.com>), as well as Nanohub (<http://www.nanohub.org>).

ReaxFF Parallel Implementation in LAMMPS. LAMMPS uses the highly scalable domain-decomposition method, along with MPI, for large distributed-memory parallel computations [8]. The ReaxFF capability in LAMMPS (USER-REAXC) is originally based on PuReMD [21], as is the QEq charge equilibration method in LAMMPS. Traditionally, the QEq charge equilibration method has been used with ReaxFF. However, QEq has been shown to be problematic for isolated atoms or molecules, where the ACKS2 method should be used instead [24]. A fully MPI-enabled ACKS2 method has been implemented in LAMMPS,

based on a serial PuReMD implementation [59]. LAMMPS uses the Kokkos library [25] on top of MPI, for performance portability. Kokkos abstractions allow a single c++ kernel to run on different hardware, i.e. GPUs and multi-core CPUs. LAMMPS also includes a native OpenMP version of ReaxFF in the USER-OMP package [46].

When running on the GPU, it is important to use the “full” neighbor-list option for QEq for performance, which eliminates the need for thread-level atomics, at the cost of redundant computation in the sparse-matvec. In order to investigate the performance of ReaxFF in LAMMPS, a water system was run on Intel Skylake (Xeon 6140) and NVIDIA V100-16GB hardware. Both the QEq and ACKS2 charge equilibration methods were used. Fig. 5a shows performance vs system size on a single node or GPU. CPU performance is relatively flat vs system size, while GPU performance is better for larger systems. Fig. 5b shows performance vs node count when weak scaling 96k atoms per node. GPU performance is better for larger node counts, while CPU performance is relatively flat.

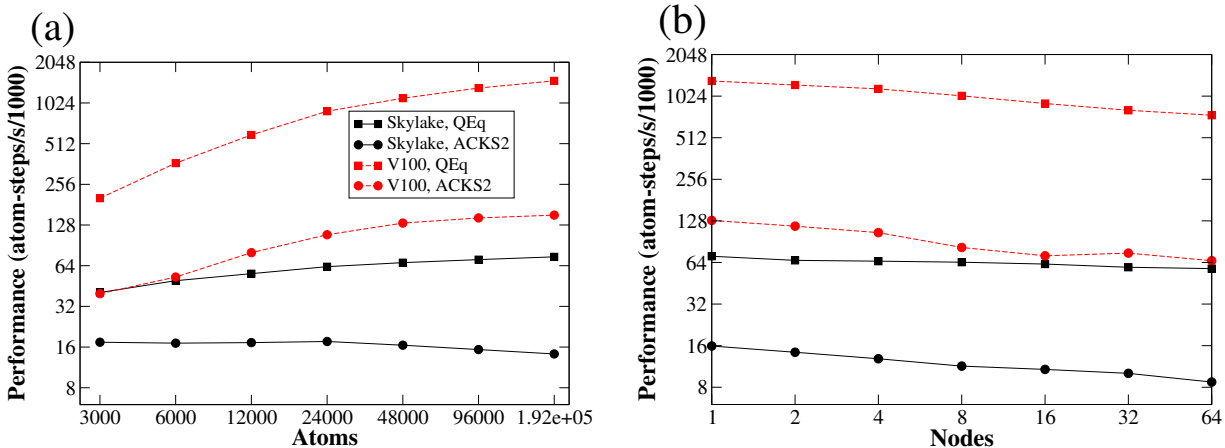


Figure 5: LAMMPS overall performance (thousand atom-timesteps per second) for a ReaxFF water system vs. (a) system size, running on a single CPU node or GPU and (b) node count, weak scaling 96k atoms per node. Black curves represent Intel Skylake CPU hardware, and blue curves represent NVIDIA V100 GPU hardware. Squares represent the QEq method, while circles represent the ACKS2 method. Higher is better, and ideal scaling is a horizontal line.

Energy Conservation in ReaxFF. A known flaw of the ReaxFF version in LAMMPS is its lack of appropriate energy conservation in the NVE ensemble. While thermostats are usually applied in applications of MD which can tolerate small energy drifts, lack of

energy conservation implies a defective PES and imperfect forces acting on the atoms. A recent paper by David Furman and David Wales identified the origins of energy drifts due to discontinuities in the PES which result from distance cutoffs used in the bonded interaction terms of ReaxFF.[49] In order to repair the flaws in the PES they applied a 7th degree polynomial tapering function to the bonder order and bonded energy terms. A high degree polynomial tapering function was utilized since it satisfies the requirement of a smooth first and second order derivatives at the cutoff distances and low computational overhead.

Figs. 6a and 6b compare the energy drift in the NVE ensemble with and without the tapered bond energy terms of the Trpzip2 peptide and TNT crystal systems, respectively. It is evident that the tapered PES successfully reduces the energy drifts by more than an order of magnitude with respect to the original ReaxFF PES. However, one should be cautious when applying the tapering function correction as the modification in the PES can alter systems properties and therefore a reparameterization of the force field might be necessary. For example, the density of the TNT crystal changed from $1.59\text{g}/\text{cm}^3$ in the original ReaxFF to $1.69\text{g}/\text{cm}^3$ in the tapered corrected PES. It is worth mentioning a subsequent paper from the same group which proposed modifications of various ReaxFF energy terms with an aim to further improve the stability and accuracy of ReaxFF.[55]

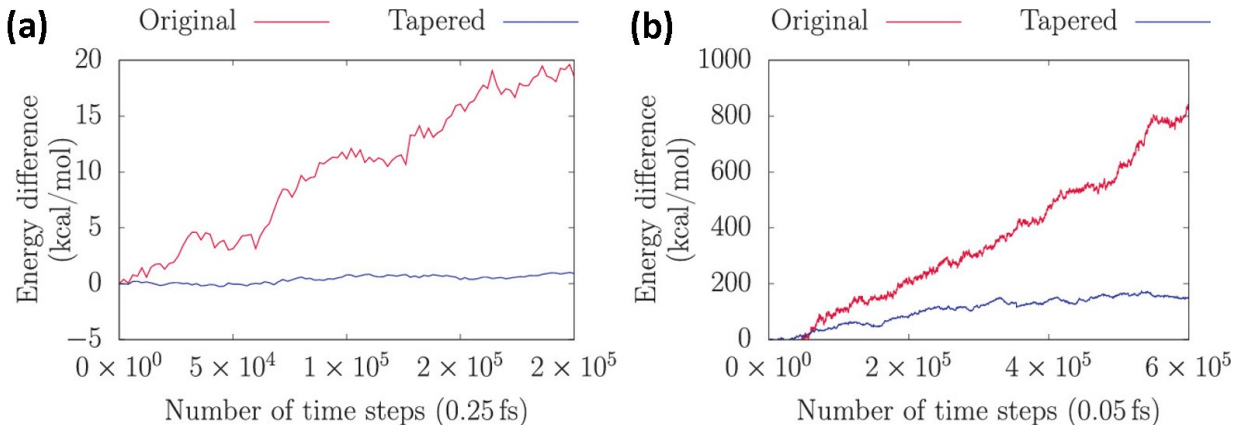


Figure 6: Total energy conservation during NVE dynamics for original (red line) and tapered (blue line) ReaxFF: (a) Trpzip2 peptide; (b) TNT single crystal

Hybrid models in LAMMPS. It is relatively straightforward to conduct a ReaxFF/MM simulation in the LAMMPS code. Along these lines, a hybrid ReaxFF/AMBER MD tool, which introduces bond breaking and formation capabilities within the AMBER MD software package, was recently introduced [60]. This tool enables the study of local reactive events in large systems at a fraction of the computational costs of QM/MM models. For this ReaxFF/AMBER MD integration, AMBER is the simulation driver. AMBER categorizes atoms into the core ReaxFF, transition (ReaxFF/MM) and AMBER atom groups, it sends all relevant information for core ReaxFF and ReaxFF/MM atoms to the ReaxFF program. The ReaxFF/AMBER tool currently uses the external model interface, i.e., the necessary data transfers between ReaxFF and AMBER are performed using file-based data exchange (a full fledged software integration is in the works). After AMBER writes the data exchange files, it launches the ReaxFF program as an external binary. The ReaxFF program then runs a zero-step non-periodic simulation to calculate the dynamic charges, energies and forces on the ReaxFF and ReaxFF/MM atoms, and writes this information back into another file for AMBER to read.

In implementing the interface between the ReaxFF and AMBER programs, the following procedure is currently adopted by the ReaxFF/AMBER software:

- Dynamic charges on ReaxFF atoms are calculated under the influence of ReaxFF/MM atoms with static charges using CEM,
- All interactions between ReaxFF-ReaxFF atom pairs are calculated without any modifications,
- Electrostatic interactions between ReaxFF (w/dynamic charges)-ReaxFF/MM (w/static charges) atom pairs are calculated by ReaxFF,
- van der Waals interactions between ReaxFF-ReaxFF/MM atom pairs are handled by AMBER (e.g., using a Lennard Jones potential),

- Interactions between MM-ReaxFF/MM and MM-MM pairs are handled by AMBER as usual.

It should be noted that there are some limitations of the current ReaxFF/AMBER tool. Currently, only systems with non-covalent bonds between ReaxFF and ReaxFF/MM regions can be studied. Also, only shared memory parallelism can be leveraged for the ReaxFF region. Nevertheless, the current implementation serves as a proof-of-concept on the feasibility and advantages of the hybrid approach[60], and proved robust for our implementation of the ReaxFF/C-GeM/Amber model to consider proton hopping mechanisms in acidic nanopools of reverse micelles[62].

Methods for Accelerating Reactive Force Fields

Preconditioners generally transform the set of linear equations for improving their spectral properties. A recent study compared the performance of several preconditioners on the convergence of EEM and demonstrated that a significant reduction in computational time can be achieved by applying incomplete factorization and sparse approximate inverse (SAI) preconditioners.[58] Another recent addition is the development of extended Lagrangian schemes that have been demonstrated to reduce the computational cost of EEM significantly. The inertial extended Lagrangian/self-consistent scheme method (iEL-SCF) reduces the number of SCF iterations by allowing sufficient energy conservation with a high convergence tolerance.[51] An additional extended Lagrangian method is the stochastic constrained extended Lagrangian molecular dynamics (SC-XLMD) which eliminates the requirement for SCF altogether by propagating the charges and chemical potential along with the atomic degrees of freedom and applying holonomic constraint to ensure charge conservation.[61] Here we review these new algorithms for improving reactive force field performance with focus on ReaxFF specifically.

Preconditioned Solvers. Solution of large sparse linear systems required for charge

equilibration models constitutes a significant bottleneck against scalability of ReaxFF simulations. More precisely, the sparse linear solves may start accounting for a significant portion of the execution time in large runs due to communication overheads. O’Hearn et al. have addressed this issue in a series of publications. First, they presented a number of incomplete LU (ILU) based preconditioning techniques and their parallelization on shared memory architectures [35]. They have shown that these techniques significantly accelerate the Krylov subspace solvers for the QEq, EEM and ACKS2 charge models (from tens of iterations per step to as low as a few iterations).

However, ILU based preconditioners are inherently sequential, and therefore they do not lend themselves for scalable distributed memory parallelization. For this reason, they next investigated sparse approximate inverse (SAI) preconditioning techniques, as those can easily be scaled to large systems [58]. SAI preconditioners are generally less effective than ILU techniques. However, they were able to identify sparsity patterns in SAI that allow QEq, EEM and ACKS2 solvers to convergence at about the same rate as ILU based techniques. In fact, their experiments on shared memory systems have shown that SAI solvers outperform ILU ones as a result of their better parallelizability properties.

Recently, they have developed efficient distributed memory parallel implementations of the SAI preconditioning based solvers mentioned above [52]. The basic CG solver used in parallel ReaxFF implementations utilizes the simple Jacobi (diagonal) preconditioner and requires tens of iterations per step for convergence. The newly developed distributed memory parallel SAI implementation leverages problem-specific characteristics of ReaxFF for high efficiency and scalability. The resulting solver significantly outperforms the Jacobi preconditioning based solvers in terms of rate of convergence and execution time as demonstrated by numerical tests (see below). To ensure high scalability, the new solver implementation utilizes a communication-hiding conjugate gradient scheme, specifically the pipelined CG (PIPECG) algorithm that reduces the number of global communications to only one non-blocking global reduction per iteration that can actually be overlapped with the preconditioner application

and sparse matrix vector multiplication (SpMV) operation at the expense of some increase in vector operation costs [26].

For numerical tests, two molecular systems, which were comprised of bulk water (H_2O) and amorphous silica (SiO_2), were selected. These performance experiments were performed on the Cori supercomputer at the National Energy Research Scientific Computing Center (NERSC). In Table 1, we present the mean solver iterations for the Jacobi and SAI preconditioned QEq solvers at 10^{-6} and 10^{-10} tolerances for the 864K atom silica and 837K atom water systems using 512 processes. SAI preconditioning significantly improves the convergence rate of QEq compared to Jacobi preconditioning, yielding about 3.5 and 6 times faster convergence rates at 10^{-6} tolerance for silica and water, respectively. The margin between SAI and Jacobi preconditioning gets larger as we go to 10^{-10} tolerance threshold both for silica and water systems. Other tests (not shown here) indicate that this trend continues as we move to even smaller tolerances, *e.g.*, 10^{-14} .

Table 1: Mean solver iterations for the Jacobi and SAI preconditioned QEq solvers.

			Iters.
Dataset	Tol.	Solver	
Silica	10^{-6}	CG+Jacobi	11.8
		PIPECG+SAI	1.9
	10^{-10}	CG+Jacobi	39.2
		PIPECG+SAI	5.8
Water	10^{-6}	CG+Jacobi	9.5
		PIPECG+SAI	2.7
	10^{-10}	CG+Jacobi	38.4
		PIPECG+SAI	10.2

In Fig. 7, the strong scaling plots for mean QEq time of preconditioned solvers for the silica and water systems are shown. Figures from left to right within a row show results at convergence tolerance levels of 10^{-6} and 10^{-10} . There are four different solver variants, starting with the basic CG+Jacobi solver and progressively advancing to the PIPECG+SAI solver. As expected, the fully optimized PIPECG+SAI version exhibits the best overall performance in large core counts. At $1e6$ tolerances speedups up to 2.4x, and at $1e10$

tolerances speedups up to 4.6x are observed from PIPECG+SAI over the original QEq solver in PuReMD.

The distributed memory parallelization work described above focused on the QEq model. While, O’Hearn et al. have developed an effective GMRES based solver for ACKS2 in [58], unfortunately GMRES is not a nicely parallelizable solver, and CG does not work for ACKS2 due to the indefinite nature of the matrices. However, it is possible to extend these preconditioning techniques to the ACKS2 model using a BiCGStab solver.

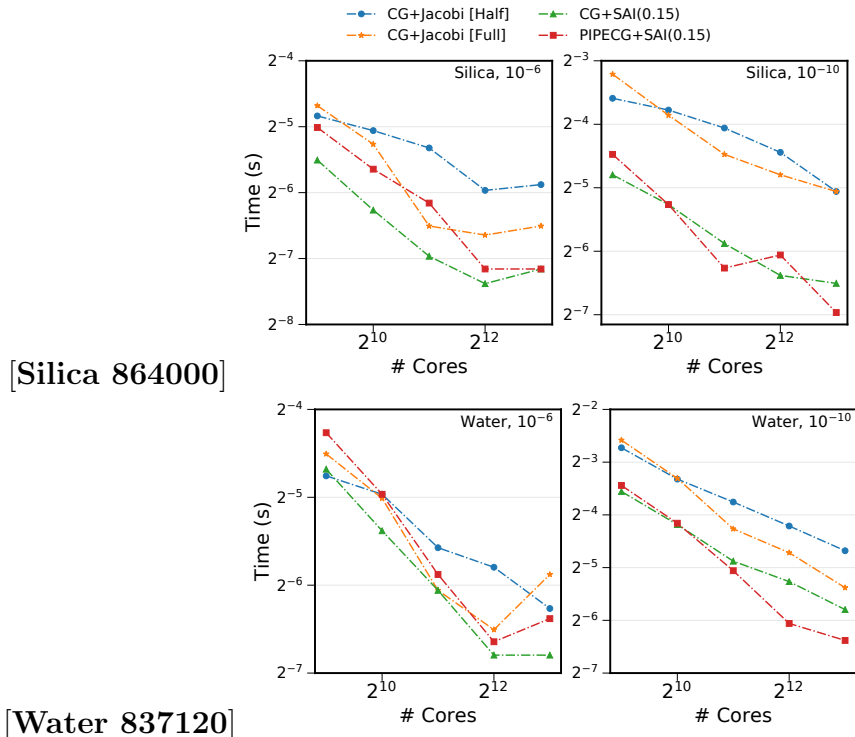


Figure 7: Strong scaling plots for mean QEq time of four preconditioned solvers for the silica and water systems. Figures from left to right within a row show results at convergence tolerance levels of 10^{-6} and 10^{-10} .

Extended Lagrangian Methods for Solving Charge Equilibration. EEM is an essential component of ReaxFF, allowing for charge rearrangement as bonds break and form, however, the self consistent solution of two sets of linear equations at each timestep is associated with a high computational cost. Therefore, numerous efforts are being made to decrease the computational cost of EEM which will allow applying ReaxFF on larger scales.

Recently, Leven et al. applied the inertial extended Lagrangian self-consistent field method (iEL-SCF), originally developed for solving induced polarization, for solving EEM.[50] In the iEL-SCF method, auxiliary charges are introduced and propagated with respect to a harmonic potential around the real charges which serve as an initial guess for the CG SCF solution of the real charges. As shown in Eq. (8), determining the partial charge in EEM requires solving two separate sets of linear equations. Therefore, the extended Lagrangian of the system is written as follows:

$$\begin{aligned} \mathcal{L}_{\text{hybrid}}^{\text{charge}} = & \frac{1}{2} \sum_{i=1}^N m_i \dot{r}_i^2 + \frac{1}{2} \sum_{i=1}^N m_s (\dot{q}_{i,aux}^s)^2 + \frac{1}{2} \sum_{i=1}^N m_t (\dot{q}_{i,aux}^t)^2 - U(\vec{r}^N, q^N) \\ & - \frac{1}{2} \omega^2 \sum_{i=1}^N m_s (q_{i,aux}^s - q_{i,SCF}^s)^2 - \frac{1}{2} \omega^2 \sum_{i=1}^N m_t (q_{i,aux}^t - q_{i,SCF}^t)^2 \end{aligned} \quad (19)$$

Where r_i is the position vector of atom i , U is the ReaxFF potential energy, the auxiliary charges, $q_{i,aux}^s$ and $q_{i,aux}^t$, evolve in time subject to a harmonic potential around the real charges $q_{i,SCF}^s$ and $q_{i,SCF}^t$, and m_i , m_s and m_t are the masses of atom i , s 's and t 's respectively. The above extended Lagrangian yields unstable dynamics due to accumulation of numerical error from insufficient convergence of the SCF which manifests as increased kinetic energy of the auxiliary charges. Therefore, we apply Berendsen thermostats to the auxiliary charges which stabilize their dynamics. In Fig. 8a we plot the energy as a function of time for the $FeHO_3$ system, comparing the original CG method with iEL-SCF using various convergence thresholds. It is evident that the iEL-SCF allows stable dynamics even at high convergence tolerance thresholds of $5e^{-4}$ for the q_{aux}^s and e^{-4} for the q_{aux}^t . Furthermore it is shown in Fig. 8b that the number of SCF cycles with iEL-SCF can be reduced to around a third of the number of SCF cycles using CG at 10^{-8} .

While the number of iterations to solve the EEM equations can be largely reduced by iEL-SCF methods, it would be more desirable to eliminate iterations altogether by incorporating Stochastic-XLMD-like approaches. In this direction, Tan *et al* [61] introduced the SC-

XLMD (Stochastic Constrained XLMD) approach, which is a re-formulation of a previous iteration-free scheme, Stochastic-XLMD.[48] Since Stochastic-XLMD is originally designed for polarizable models, the key ingredient here is employing the constraint of the invariance of the total charge. Therefore, both the charges $\mathbf{q} = (q_1, \dots, q_N)$ and the chemical potential μ are considered latent positions $\mathbf{l} = (\mathbf{q}, \mu)$, with latent momenta $\mathbf{p}_l = (\mathbf{p}_q, p_\mu)$ and latent mass $\mathbf{M}_l = m_q \mathbf{I}_n, m_\mu$.

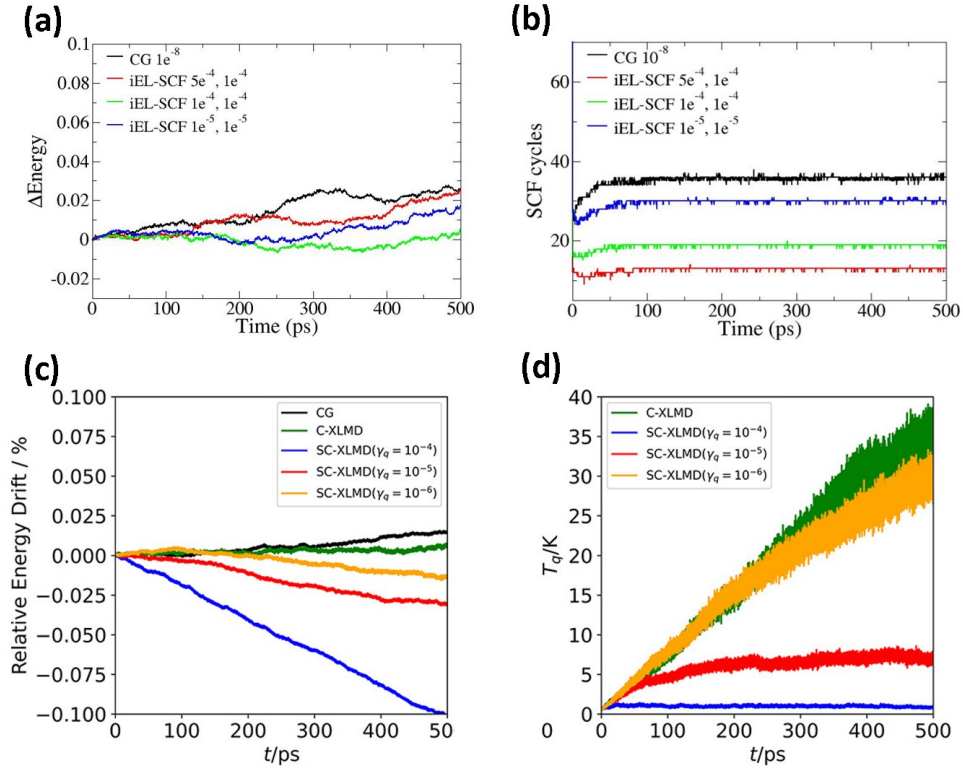


Figure 8: iEL-SCF and sc-XLMD extended Lagrangian methods for accelerating EEM. Top: Comparing the energy conservation and SCF-cycles of CG-SCF at e^{-8} level of convergence with iEL-SCF using various levels of convergence tolerance of q_{SCF}^s (right) and q_{SCF}^t (left) for the $FeOH_3$ system (a) Energy conservation (b) SCF cycles Bottom: Comparison of methods for energy conservation and latent charge temperature of the SC-XLMD simulation for bulk water. (c) Total relative energy drift in percentage units as a function of time for CG (1010 convergence criteria), an extended Lagrangian with no latent variable thermostating (C-XLMD), and when using SC-XLMD with various values of the latent thermostat coupling parameter γ_q . The absolute value of energy drift rates in percent ns1 are 0.03 (CG), 0.01 (C-XLMD), 0.22 (SC-XLMD with $\gamma_q = 10^4$), 0.05 ($\gamma_q = 10^5$), and 0.03 ($\gamma_q = 10^6$). (d) The corresponding latent charge temperature as a function of time for the above methods except CG.

The atomic and latent variables evolve according to the following extended Hamiltonian:

$$H_{\text{ext}}(\mathbf{r}, \mathbf{p}, \mathbf{l}, \mathbf{p}_l) = \frac{1}{2} \mathbf{p}^T \mathbf{M}^{-1} \mathbf{p} + \frac{1}{2} \mathbf{p}_l^T \mathbf{M}_l^{-1} \mathbf{p}_l + U(r) + \frac{1}{2} \mathbf{l}^T \begin{pmatrix} \mathbf{J}(\mathbf{r}) & -1 \\ -1 & 0 \end{pmatrix} - \begin{pmatrix} -\chi \\ 0 \end{pmatrix}^T \mathbf{l} \quad (20)$$

Similar to the iEL-SCF method, the latent variables are further thermostated with a Langevin thermostat. The appropriate thermostat coupling constant, γ , however, should be determined from preliminary tests, where the thermostating can both control the latent temperature and obtain sufficient energy conservation. Fig. 8c and 8d present the effect of the thermostat coupling parameter on the energy conservation and temperature of the latent variables respectively. It is shown that while increasing the thermostat coupling parameter results in poorer energy conservation Fig. 8c it allows to stabilize the temperature of the latent variables Fig. 8d. Therefore a small thermostat coupling should be used.

Conclusion and Future Directions

To conclude we have summarized recent efforts to improve the accuracy, transferability, and the efficiency of bond order reactive force fields such as ReaxFF. The C-GeM approach offers many advantages over reactive force field CEM: it can rapidly predict the electrostatic potential surfaces of molecules, molecules dissociate into integer charge fragments so that redox reactions are easily described, there is no unphysical long-range charge transfer, it can account for out-of-plane polarization, and charges are not required to be centered on atoms, thereby accounting for electrostatic features such as sigma holes. A number of improving the electrostatic description in ReaxFF have been proposed. The ACKS2 is a good alternative for replacing EEM by fixing the well-known unphysical long-range charge transfer and metallic polarizability associated with EEM. However, the more advanced ACKS2 method comes at the expense of substantial computational cost. Two extended Lagrangian schemes have been developed for solving charge equilibration. The iEL-SCF and the SC-XLMD. The iEL-SCF

had been shown to substantially reduce the number of SCF iterations by allowing stable dynamics at higher convergence tolerance thresholds. The SC-XLMD method eliminates the requirement for solving a self-consistent solution of charge equilibration. In addition, SAI preconditioners have shown to provide a significant improvement with respect to the Jacobi preconditioner which is currently utilized in LAMMPS. All of these methods have shown to substantially reduce the computational cost of charge equilibration, however there is still no straight answer as to which method is preferred. The SC-XLMD probably provides the highest increase in performance due to elimination of the SCF process, however this comes at the expense of having to tune the thermostat coupling parameter to ensure proper stability. The SAI preconditioners and the iEL-SCF are showing similar numbers in the reduction of the number of SCF cycles. Perhaps future work can assess whether combining iEL-SCF with SAI preconditioners can further enhance the efficiency of the charge equilibration methods. Perhaps future work should focus on developing extended Lagrangian schemes and matrix preconditioners for accelerating the performance of ACKS2. C-GeM has also been shown to provide an alternative for EEM, showing exceptional production of bulk water properties. One of the main advantages of C-GeM is that it not only remedies the shortcomings of EEM but can also describe electron transfer processes and dissociation to ionic products by utilizing explicit electrons. Current research is focused on extending C-GeM for bio-molecules and drug like compounds showing that atom-typing significantly enhance the accuracy C-GeM. Future research will work on implementing the atom-typed C-GeM in non-reactive force fields which will benefit from high accuracy and a unified framework for treating permanent electrostatics, polarization and charge transfer. Another method for accounting for redox chemistry is the eReaxFF, which replaces EEM with ACKS2 for the electrostatic potential and utilizes explicit electrons which enables to describe the different oxidation states of atoms. A unique feature of eReaxFF is that it incorporates modified bond order terms which couple the charge state of an atom with its chemical activity. Recent applications of the eReaxFF have demonstrated that paring every atom with an electron in a “quasi-drude” type

model is showing promising results. Current research is working to extend the applications of the "quasi-drude" eReaxFF to other systems such as water and graphite. Perhaps future research directions can combine C-GeM with eReaxFF, thus eliminating the requirement for solving the ACKS2. Recent methods for combining the ReaxFF with non-reactive force fields have also been demonstrated, benefiting from the efficiency of non-reactive force fields and accounting for reactivity only in a local region. To date the hybrid approach has only been applied for systems where the non-reactive and reactive region interact through non-bonded interactions. Future research should extend the hybrid approach to include also a covalent bonded interface between the reactive and non-reactive regions. To conclude, ReaxFF is one of few methods available for chemists to computationally account for chemical reactivity in large scale simulations. Therefore, ReaxFF is likely to keep playing a significant role in the computational study of complex reactive systems and we expect to see further developments for improving the ReaxFF.

Acknowledgments

This work was supported by the National Science Foundation under grant CHE-1955643 (I.L. and T.H-G.). We also acknowledge the CPIMS program by the Director, Office of Science, Office of Basic Energy Sciences, Chemical Sciences Division of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231 for the reverse micelle study (H.H., I. L., T.H-G.). S. T. thanks the University of California Education Aboard Program (UCEAP) for financial and visa support. M.A. was supported in part by a Strategic Partnership Grant from the Michigan State University Foundation, an NSF CDS&E grant (award number 1807622) and an NIH grant (award number GM130641). This work used computational resources provided by the Institute for Cyber-Enabled Research at Michigan State University, and the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility operated under Contract No. DE-AC02-05CH11231. A.C.T.v.D.,

K.A.P., D.A. B.E. and Md.J.H. were supported by NSF CDS&E grant #1807622, the U.S. Army Research Laboratory through the Collaborative Research Alliance for Multi-Scale Multidisciplinary Modeling of Electronic Materials (MSME) under Cooperative Agreement No. W911NF-12-2-0023 and through NSF NRT grant #DGE-1449785.

References

- (1) Gasteiger, J.; Marsili, M. *Tetrahedron Letters* **1978**, *19*, 3181–3184.
- (2) Parr, R. G.; Pearson, R. G. *Journal of the American Chemical Society* **1983**, *105*, 7512–7516.
- (3) Mortier, W. J.; Van Genechten, K.; Gasteiger, J. *Journal of the American Chemical Society* **1985**, *107*, 829–835.
- (4) Mortier, W. J.; Ghosh, S. K.; Shankar, S. *Journal of the American Chemical Society* **1986**, *108*, 4315–4320.
- (5) Tersoff, J. *Physical Review B* **1988**, *37*, 6991–7000.
- (6) Brenner, D. W. *Physical Review B* **1990**, *42*, 9458–9471.
- (7) Rappé, A. K.; Goddard, W. A. *Journal of Physical Chemistry* **1991**, *95*, 3358–3363.
- (8) Plimpton, S. J. *Comput. Phys.* **1995**, *117*, 1–19.
- (9) Chelli, R.; Procacci, P.; Righini, R.; Califano, S. *The Journal of Chemical Physics* **1999**, *111*, 8569–8575.
- (10) Jakalian, A.; Bush, B. L.; Jack, D. B.; Bayly, C. I. *J. Comput. Chem.* **2000**, *21*, 132–146.
- (11) Van Duin, A. C. T.; Dasgupta, S.; Lorant, F.; Goddard, W. A. *The Journal of Physical Chemistry A* **2001**, *105*, 9396–9409.
- (12) Wang, J.; Wang, W.; Kollman, P. A.; Case, D. A. *J. Am. Chem. Soc* **2001**, *222*, U403.

- (13) Brenner, D. W.; Shenderova, O. A.; Harrison, J. A.; Stuart, S. J.; Ni, B.; Sinnott, S. B. *Journal of Physics-Condensed Matter* **2002**, *14*, 783–802.
- (14) Strachan, A.; Kober, E. M.; Van Duin, A. C.; Oxgaard, J.; Goddard, W. A. *Journal of Chemical Physics* **2005**, *122*, 054502.
- (15) Wang, J.; Wang, W.; Kollman, P. A.; Case, D. A. *J. Mol. Graph. Model.* **2006**, *25*, 247–260.
- (16) Chen, J.; Martínez, T. J. *Chemical Physics Letters* **2007**, *438*, 315–320.
- (17) Mathieu, D. *Journal of Chemical Physics* **2007**, *127*, DOI: 10.1063/1.2803060.
- (18) Yu, J.; Sinnott, S. B.; Phillpot, S. R. *Physical Review B* **2007**, *75*, DOI: 10.1103/PhysRevB.75.085311.
- (19) Chenoweth, K.; van Duin, A. C. T.; Goddard, W. A. *The Journal of Physical Chemistry A* **2008**, *112*, PMID: 18197648, 1040–1053.
- (20) McNeill, V.; Yatawelli, R.; Thornton, J.; Stipe, C.; Landgrebe, O. *Atmospheric Chemistry & Physics* **2008**, *8*.
- (21) Aktulga, H.; Fogarty, J.; Pandit, S.; Grama, A. *Parallel Computing* **2012**, *38*, 245–259.
- (22) Aktulga, H. M.; Pandit, S. A.; van Duin, A. C. T.; Grama, A. Y. *SIAM Journal on Scientific Computing* **2012**, *34*, C1–C23.
- (23) Monti, S.; Corozzi, A.; Fristrup, P.; Joshi, K. L.; Shin, Y. K.; Oelschlaeger, P.; Van Duin, A. C.; Barone, V. *Physical Chemistry Chemical Physics* **2013**, *15*, 15062–15077.
- (24) Verstraelen, T.; Ayers, P. W.; Van Speybroeck, V.; Waroquier, M. *Journal of Chemical Physics* **2013**, *138*, DOI: 10.1063/1.4791569.
- (25) Edwards, H. C.; Trott, C. R.; Sunderland, D. *Journal of Parallel and Distributed Computing* **2014**, *74*, Domain-Specific Languages and High-Level Frameworks for High-Performance Computing, 3202–3216.
- (26) Ghysels, P.; Vanroose, W. *Parallel Computing* **2014**, *40*, 224–238.

- (27) Grama, A.; Aktulga, H. M.; Kylasa, S. B. PuReMD, Purdue Reactive Molecular Dynamics package, <https://www.cs.purdue.edu/puremd>, Accessed on June 8, 2016, 2014.
- (28) Shan, T. R.; Van Duin, A. C.; Thompson, A. P. *Journal of Physical Chemistry A* **2014**, *118*, 1469–1478.
- (29) Verlack, C. C.; Neyts, E. C.; Jacob, T.; Fantauzzi, D.; Golkaram, M.; Shin, Y. K.; Van Duin, A. C.; Bogaerts, A. *New Journal of Physics* **2015**, *17*, 103005.
- (30) Boes, J. R.; Groenenboom, M. C.; Keith, J. A.; Kitchin, J. R. *International Journal of Quantum Chemistry* **2016**, *116*, 979–987.
- (31) Hjertenæs, E.; Nguyen, A. Q.; Koch, H. *Phys. Chem. Chem. Phys.* **2016**, *18*, 31431–31440.
- (32) Islam, M. M.; Kolesov, G.; Verstraelen, T.; Kaxiras, E.; Van Duin, A. C. *Journal of Chemical Theory and Computation* **2016**, *12*, 3463–3472.
- (33) Islam, M. M.; van Duin, A. C. T. *The Journal of Physical Chemistry C* **2016**, *120*, 27128–27134.
- (34) Kylasa, S. B.; Aktulga, H. M.; Grama, A. Y. *IEEE Transactions on Parallel and Distributed Systems* **2016**, *28*, 202–214.
- (35) O’Hearn, K. A.; Aktulga, H. M. In *2016 7th Workshop on Latest Advances in Scalable Algorithms for Large-Scale Systems (ScalA)*, 2016, pp 9–16.
- (36) Psfogiannakis, G.; Van Duin, A. C. *Surface Science* **2016**, *646*, 253–260.
- (37) Senftle, T. P.; Hong, S.; Islam, M. M.; Kylasa, S. B.; Zheng, Y.; Shin, Y. K.; Junkermeier, C.; Engel-Herbert, R.; Janik, M. J.; Aktulga, H. M.; Verstraelen, T.; Grama, A.; van Duin, A. C. T. *npj Computational Materials* **2016**, *2*, 15011.
- (38) Ashraf, C.; van Duin, A. C. *The Journal of Physical Chemistry A* **2017**, *121*, PMID: 28072539, 1051–1068.

- (39) Giberti, F.; Hassanali, A. A. *The Journal of Chemical Physics* **2017**, *146*, 244703.
- (40) Qiu, Y.; Odendahl, N.; Hudait, A.; Mason, R.; Bertram, A. K.; Paesani, F.; DeMott, P. J.; Molinero, V. *Journal of the American Chemical Society* **2017**, *139*, 3052–3064.
- (41) Wang, N.; Peng, J.; Pang, A.; He, T.; Du, F.; Jaramillo-Botero, A. *Journal of Physical Chemistry C* **2017**, *121*, 14597–14610.
- (42) Yun, K. S.; Pai, S. J.; Yeo, B. C.; Lee, K. R.; Kim, S. J.; Han, S. S. *Journal of Physical Chemistry Letters* **2017**, *8*, 2812–2818.
- (43) Lu, K.; He, Y.; Huo, C. F.; Guo, W. P.; Peng, Q.; Yang, Y.; Li, Y. W.; Wen, X. D. *Journal of Physical Chemistry C* **2018**, *122*, 27582–27589.
- (44) Trnka, T.; Tvarovska, I.; Kovca, J. *Journal of Chemical Theory and Computation* **2018**, *14*, 291–302.
- (45) Akbarian, D.; Yilmaz, D. E.; Cao, Y.; Ganesh, P.; Dabo, I.; Munro, J.; Van Ginhoven, R.; Van Duin, A. C. *Physical Chemistry Chemical Physics* **2019**, *21*, 18240–18249.
- (46) Aktulga, H. M.; Knight, C.; Coffman, P.; O’Hearn, K. A.; Shan, T.-R.; Jiang, W. *The International Journal of High Performance Computing Applications* **2019**, *33*, 304–321.
- (47) Aktulga, H. M.; Knight, C.; Coffman, P.; O’Hearn, K. A.; Shan, T.-R.; Jiang, W. *The International Journal of High Performance Computing Applications* **2019**, *33*, 304–321.
- (48) An, D.; Y. Cheng, S.; Head-Gordon, T.; Jianfeng Lu, L. L. <https://arxiv.org/abs/1904.12082> **2019**.
- (49) Furman, D.; Wales, D. J. *The Journal of Physical Chemistry Letters* **2019**, *10*, PMID: 31682448, 7215–7223.
- (50) Leven, I.; Head-Gordon, T. *Journal of Physical Chemistry Letters* **2019**, *10*, 6820–6826.

- (51) Leven, I.; Head-Gordon, T. *Physical Chemistry Chemical Physics* **2019**, *21*, 18652–18659.
- (52) O’Hearn, K. A.; Alperen, A.; Aktulga, H. M. In *Proceedings of the ACM International Conference on Supercomputing*, 2019, pp 150–159.
- (53) Bertels, L. W.; Newcomb, L. B.; Alaghemandi, M.; Green, J. R.; Head-Gordon, M. *Journal of Physical Chemistry A* **2020**, *124*, 5631–5645.
- (54) Evangelisti, B.; Fichthorn, K. A.; Van Duin, A. C. T. *J. Chem. Phys* **2020**, *153*, 104106.
- (55) Furman, D.; Wales, D. J. *The Journal of Chemical Physics* **2020**, *153*, 021102.
- (56) Hossain, M. J.; Pawar, G.; Liaw, B.; Gering, K. L.; Dufek, E. J.; Van Duin, A. C. *Journal of Chemical Physics* **2020**, *152*, 184301.
- (57) Leven, I.; Hao, H.; Das, A. K.; Head-Gordon, T. *The Journal of Physical Chemistry Letters* **2020**, *11*, 9240–9247.
- (58) O’Hearn, K. A.; Alperen, A.; Aktulga, H. M. *SIAM Journal on Scientific Computing* **2020**, *42*, C1–C22.
- (59) O’Hearn, K. A.; Swift, M. W.; Liu, J.; Magoulas, I.; Piecuch, P.; van Duin, A. C.; Aktulga, H. M.; Qi, Y. *The Journal of chemical physics* **2020**, *153*, 084107.
- (60) Rahnamoun, A.; Kaymak, M. C.; Manathunga, M.; Götz, A. W.; Van Duin, A. C.; Merz Jr, K. M.; Aktulga, H. M. *Journal of Chemical Theory and Computation* **2020**, *16*, 7645–7654.
- (61) Tan, S.; Leven, I.; An, D.; Lin, L.; Head-Gordon, T. *Journal of Chemical Theory and Computation* **2020**, *16*, 5991–5998.
- (62) Adams, E.; Hao, H.; Leven, I.; Rüttermann, M.; Wirtz, H.; Havenith, M.; Head-Gordon, T. *submitted* **2021**, *xx*.

- (63) Akbarian, D.; Ganeshan, K.; Woodward, W. H. H.; Moore, J.; van Duin, A. C. T. *The Journal of Chemical Physics* **2021**, *154*, 024904.