



Perspective on coarse-graining, cognitive load, and materials simulation

Eric Jankowski^{a,*}, Neale Ellyson^a, Jenny W. Fothergill^a, Michael M. Henry^a,
Mitchell H. Leibowitz^a, Evan D. Miller^a, Mone't Alberts^a, Samantha Chesser^b, Jaime D. Guevara^c,
Chris D. Jones^a, Mia Klopfenstein^a, Kendra K. Noneman^a, Rachel Singleton^d,
Ramon A. Uriarte-Mendoza^e, Stephen Thomas^f, Carla E. Estridge^g, Matthew L. Jones^a



^a Micron School of Materials Science and Engineering, Boise State University, 1910 University Dr, Boise, ID 83725, United States

^b Department of Physics and Mathematics, Kansas Wesleyan University, Salina, KS 67401, United States

^c Department of Computer Science, Boise State University, Boise, ID 83725, United States

^d Department of Mathematics, Boise State University, Boise, ID 83725, United States

^e Department of Chemistry and Biochemistry Northern Arizona University, Flagstaff, AZ 86011, United States

^f Department of Electrical Engineering and Computer Science, The University of Tennessee at Knoxville, Knoxville, TN 865974, United States

^g The Boeing Company, St. Louis, MO 63134, United States

ARTICLE INFO

Keywords:
Coarse-grained models
Cognitive load
Molecular dynamics
Organic photovoltaics
Epoxy
Thermosets
Teaching
Software
Thermodynamics

ABSTRACT

The predictive capabilities of computational materials science today derive from overlapping advances in simulation tools, modeling techniques, and best practices. We outline this ecosystem of molecular simulations by explaining how important contributions in each of these areas have fed into each other. The combined output of these tools, techniques, and practices is the ability for researchers to advance understanding by efficiently combining simple models with powerful software. As specific examples, we show how the prediction of organic photovoltaic morphologies have improved by orders of magnitude over the last decade, and how the processing of reacting epoxy thermosets can now be investigated with million-particle models. We discuss these two materials systems and the training of materials simulators through the lens of cognitive load theory.

For students, the broad view of ecosystem components should facilitate understanding how the key parts relate to each other first, followed by targeted exploration. In this way, the paper is organized in loose analogy to a coarse-grained model: The main components provide basic framing and accelerated sampling from which deeper research is better contextualized. For mentors, this paper is organized to provide a snapshot in time of the current simulation ecosystem and an on-ramp for simulation experts into the literature on pedagogical practice.

1. A vibrant ecosystem

This perspective describes four issues in computational materials, the vibrant ecosystem in which they are being solved (Fig. 1) and a review of recent advances and best practices for studying materials self-assembly. A central theme of this work is the use of simplified models [1] to provide accessible on-ramps for deeper investigation. The four issues are as follows:

1. Understanding materials behavior through computer simulation
2. Reproducibility of research
3. Accessibility of materials simulation tools
4. Demand for computationally literate researchers

These issues overlap: Reproducible results better advance

understanding of materials. Accessible tools facilitate reproducibility. Students with molecular simulation expertise have transferable, in-demand skills. By discussing these issues in the context of the molecular simulation ecosystem, we show how components of the ecosystem are related and are advancing materials research.

The problems of research reproducibility and demand for computationally literate researchers are broad, encompassing more than the molecular simulation community. In 2016, 52% of researchers agreed there is a "crisis" of reproducibility [2] and more than 600,000 high-paying tech jobs went unfilled in the US [3]. Who will fill these jobs and who will ensure research is reproducible? One candidate population is the pool of XSEDE [4] supercomputer users. These researchers (2,186 undergraduate and 8,409 graduate students in 2017) use nationally-available high performance computing (HPC) facilities to perform scientific research [5] and develop expertise with automating repeatable

* Corresponding author.

E-mail address: ericjankowski@boisestate.edu (E. Jankowski).

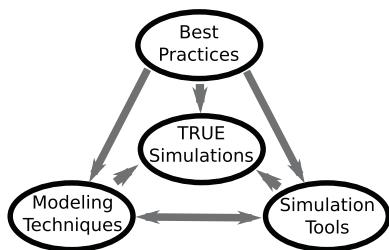


Fig. 1. Molecular simulations are becoming more informative and reproducible due to overlapping advances in modeling techniques and simulation tools through best practices in teaching, sharing, and software development.

tasks, managing software stacks, using parallel hardware, and writing software to extract understanding from data. Such computational researchers have the opportunity to demonstrate leadership with reproducibility because the entire research apparatus of one user, including the hardware, software, and pseudorandom number generator seeds used to perform a computation can be replicated exactly by another user—luxuries that are generally not available to non-computational research. However, the fact that only 0.011% of the 19.8 million US undergraduates in 2017 were XSEDE users gives a sense for how rare such leaders might be and the gaps that exist in training computationally literate scientists. Researchers themselves are aware of the gaps: 60% of those surveyed in 2015 reported computational training as their greatest need [6]. In part, this is due to increased data ubiquity and the associated data science and HPC skills needed to manage it [7].

Because computational materials researchers develop XSEDE-user skills, understanding the computational materials ecosystem of tools, techniques, and practices can inform modern workforce training more broadly. We aim for materials simulations that are *transferable, reproducible, usable, and extensible (TRUE)* [8]. In this work we describe best practices and computational tools that enable TRUE simulations. These practices and tools help researchers waste less time, enhance research reproducibility, and prepare them for in-demand technical roles.

“Best practices” refers to the use of open software, software engineering practices, and pedagogy for teaching computing generally and molecular simulations specifically. These practices have both a) enabled the creation of open source tools used broadly by the molecular simulation community, and b) been used within the molecular simulation community to advance simulation tools and modeling techniques (Fig. 1). The “simulation tools” discussed are used primarily to perform molecular dynamics simulations using pairwise potentials to model the interactions between simulation elements. Each of the main simulation engines is a significant feat of software engineering towards meeting their users’ demands of application-specificity and performance. “Modeling techniques” refers to the algorithms used within simulation engines, interaction potentials (force fields), statistical sampling techniques, and theory. Explaining how practices, techniques, and engines are connected to each other is important because each area in isolation has near-infinite depth that can hinder accessibility to new researchers.

2. Timescale problems

Molecular simulations predict the structure and properties of materials using computer implementations of physics-based descriptions of matter. The recent review article by Braun et al. provides a comprehensive overview of the components and considerations for molecular dynamics (MD) simulations [9], which we also focus on here. MD simulations suffer from two *scaling* timescale problems: The more atoms needed to represent a system, (1) the more calculation time is required to generate the next configuration, and (2) the more configurations need to be sampled before equilibrium is achieved. In other words, it takes a lot more time to simulate larger systems. These timescale

problems derive from algorithmic scaling of calculating interactions between N simulation elements (often atoms) and because larger systems have more configurations (microstates) [10,11]. Graphics processing units (GPUs) represent a major advance in computing hardware for ameliorating these scaling problems, and we recommend the 2010 review by Stone et al. as a starting point [12]. Because of the performance benefits of GPUs, all open-source MD packages now offer GPU support [13–17].

We introduce here the *training* timescale problem that MD and MC simulation techniques also suffer from: Researchers spend more time making and fixing modeling errors as the number of software dependencies and scientific topics needed for the model increases, especially if any of them are new to the researcher. The importance of the training timescale problem explains the growing efforts around training computational researchers [18]. Because scaling and training problems are obstacles to performing TRUE simulations, it is important for researchers to be mindful of tradeoffs between them when making modeling choices.

3. Best practices and cognitive load

Evidence-based instructional practices are being applied within communities of scientific software developers to create tools and training materials that feed back into these communities. Ambrose et al. provides a comprehensive review of the science of teaching, and is an accessible introduction to research around *cognitive load* that we focus on here [19]. The basic idea of cognitive load is that the mental faculties of learners are finite, and their performance on a task (e.g., testing a new MD package) is hindered when they are asked to do more than one thing at a time [20]. The lens of cognitive load provides an accessible introduction to the research around stereotype threat and inclusivity, major barriers to participation of historically underrepresented groups in science, technology, engineering, and mathematics [21–23]. Reduction of cognitive load is a principle of course design [24–26,6,27,28], human computer interactions [29], model-based computing [30,31], and efforts to make academic writing more accessible [32]. In particular, Software Carpentry, Data Carpentry, and Library Carpentry (The Carpentries) are community-driven projects that apply the science of teaching (especially cognitive load reduction) to empower individuals to use computing in support of their professions [27,18,26,6,28]. We focus on cognitive load because of its centrality to tool accessibility and inclusive research communities.

For a sense of the ubiquity of cognitive overload in materials simulation, consider a novice simulator investigating how metal nanoparticles sinter on a surface during additive manufacturing in an atmosphere with alkanes. They begin with an xml file and discover they need to use a command-prompt to get it “in” to their lab’s simulation engine. They review the literature to find dozens of seemingly appropriate forcefields with different parameterizations and functional forms [33]. After selecting the embedded atom model [34] to represent the metal atoms, they find difficulty choosing a forcefield for the atmosphere from MM4 [35], OPLS-AA [36], GAFF [37], COMPASS [38], and Trappe [39], all with different models for the same compounds—how can this be? They consider the importance of charges, leading to Ewald summation [40] and polarizable force fields [41]. They begin to despair and wonder if compiling a density functional theory package will be faster. It isn’t. Now with six unresolved lines of questioning and a command-prompt that beeps at the letter ‘f’, the simulator feels like they’re moving backward.

Beeping prompts and sad students are finding help in the modern pedagogy summarized above, facilitated by the development of open-source tools. It came as a surprise to many that large, decentralized software projects could be successful despite lack of private return [42]. However, people enjoy helping each other online, both for the enjoyment of sharing their experiences and building professional reputation, and researchers further derive utility from software that helps with

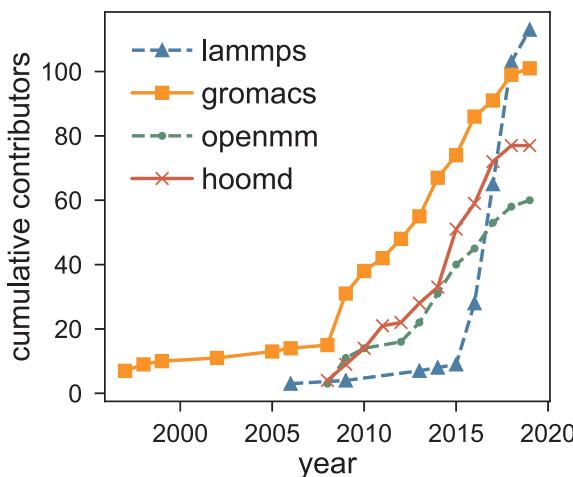


Fig. 2. Number of unique authors of four popular MD simulation engines over the last two decades. The increased growth around 2010 coincides with maturation of GPU technologies for MD and growth in Software Carpentry efforts. Numbers are approximate, as a few authors in each community may be double-counted if they commit with multiple pseudonyms.

research [43]. This is apparent in the development logs of Carpentries lessons: As one example, there are 1,942 commits from about 230 individuals since 2013, all attempting to make a better lesson for teaching the basics of version control (Software Carpentry commit log). Open-source, GPU-accelerated MD engines have experienced growth in community development over the same time frame (Fig. 2). This is not to say any one of the Carpentries, GPUs, or GitHub explains the recent growth in open source science software, but instead emphasizes that coincident contributions to pedagogical practices, hardware advances, and online development communities are important in understanding this ecosystem.

Simulators are exchanging information beyond individual packages, now sharing teaching material under version control (David Kofke's molecular simulation course), pre-packaged virtual machines for workshops [44], and new journals for living documents of best practices, tutorials, and perpetual reviews. A number of organizations have grown around the support of sustainable software development for science and the work of Katz et al. provides a broad overview [45]. A list of tools for molecular simulation is included in Table 1, and The OpenScience Project catalogs hundreds of open projects across disciplines.

Table 1
Open source software helpful for materials simulations.

Package Name	Description
Diffractometer [46,47]	Python code for generating scattering patterns from MD snapshots
oyer [48]	Python package for atom-typing
freud [49]	Python exposure of C++ analysis: RDF, order parameters, correlation functions
mBuild [50]	Python package for system initialization with reusable, hierarchical components enabling complex initialization from simple building blocks
MDAnalysis [51]	Python package for MD trajectory analysis supporting many file formats
MDTraj [52]	Python package for analyzing and extracting information from MD trajectories
MorphCT [46,53]	Python package for obtaining and aggregating charge transport properties from MD snapshots
OVITO [54]	Python exposure of C++ analysis: Visualization, structure determination
packmol [55]	Library for initializing configurations of simulation elements
physical_validation [56]	Python package for performing thermodynamic consistency checks
Planckton [57]	Python package for initializing and executing HOOMD-Blue (<code>hoomd</code>) simulations
PLUMED [58]	Software for advanced sampling, using collective variables
pyLAMMPS [59]	Python package used to manage LAMMPS output
Rhaco [60]	Python package for initializing and simulating molecules and atoms at surfaces
SSAGES [61]	Use collective variables and advanced sampling methods with amny engines
signac-flow [62,63]	Python package for automating workflows including HPC schedulers and job submission
signac [62,63]	Python package used to manage multi-dimensional data spaces and general workflows at scale
VMD [64]	Interactive and scriptable visualize and analysis of simulations
VOTCA [65]	Package for automating coarse-graining and charge transport calculations

Beyond tools designed for molecular simulation, there are important categories of tools for lowering the cognitive load of software development and fostering collaboration. GitHub, Bitbucket, and Gitlab are the three largest platforms [66] for collaborating on code repositories, and offer extra useful features at no cost for academic use. Messaging products Slack and Gitter are now popular for their integration with hosted repositories [67] and lower the barrier to entry for discussing issues and getting help. To lower the cognitive load of getting someone else's code to run, myBinder [68] enables users to launch Jupyter notebooks supporting multiple languages with pre-made environments for the code in question. As examples, the MoSDef tutorials [69] use myBinder to spin up a Jupyter notebooks enabling users to begin tutorials without touching the software environment on their own computer. For solving the software stack problem on HPC clusters, singularity [70] enables users to deploy portable "containers" from open-source Dockerfiles [71] across multiple clusters, works with NVIDIA GPUs, and on many XSEDE resources.

Recurring themes in recommended readings [72,27,73] around best practices and considerations for community-driven scientific software development include:

1. take into account cognitive load
2. use version control
3. automate repetitive tasks
4. collaborate on and share open code
5. write code in the highest-level language possible
6. software development is a fundamental literacy for engineers and researchers.

We also recommend Ref. [51] as an example of a clearly described scientific software package (MDAnalysis) with relevance to materials simulation.

4. Modeling techniques

We now return to the original problem of advancing understanding of materials from molecular simulations and describe techniques for extracting more information from each step of an MD trajectory. Both coarse-grained models and advanced sampling help with scaling time-scale problems by focusing on the key features of the phenomenon of interest, spending less effort on irrelevant details. In practice, implementing these techniques can lead to increased cognitive load if we are unaware of available infrastructure, so we organize key sources for learning more.

Briefly, by representing a collection of atoms with “coarse” simulation elements, significantly longer timescales are accessible because a) less computation is needed to compute the next configuration, and b) dynamics are accelerated because the underlying energy landscape is smoothed. Simplified models of polymers are among the first systems studied with molecular simulations [74,75] and the literature around coarse-graining is now extensive. For a polymer focus, see the recent perspective by Gartner and Jayaraman [76]. For biomolecules and protein folding there are many good sources including the reviews of Voth [77], Clementi [78], Elber [79], Klepeis [80], Kamerlin [81], and Kmiecik [82]. The MARTINI model stands out as a broadly successful coarse biomolecular model [83]. The specific problem of virus capsid self-assembly is reviewed in Perlmutter et al. [84]. For multiscale modeling, coarse-grained potentials can be derived by matching structure [85] or forces [86,87], and the relative entropy framework of Shell and Chaimovich [88,89] provides a measure of information loss through coarse-graining.

The calculation of free energy differences, rare events, and alternative approaches to sampling long dynamics can be accomplished by applying statistical mechanics to simulated trajectories. SSAGES [61] provides a comprehensive overview of advanced sampling techniques as well as open software for deploying them. Markov state models (MSM) [90] are statistical tools for describing the coarse dynamics of MD trajectories and provide a way of aggregating information from multiple short runs. MSMs themselves are a coarse-graining approach that has benefited from and contributed to the molecular simulation ecosystem. Machine learning approaches provide opportunities for extracting collective variables, trends, and patterns from materials simulations, and the review by Ferguson [91] provides a current, comprehensive view.

5. Organic photovoltaic structure and performance

In this section we review key topics in simulations of organic photovoltaics (OPVs) and describe our recent work in this context. OPVs convert photons into electrical current and engineering their structure to improve performance is an active area of research. For a more detailed picture of why OPVs are a promising technology for sustainable energy generation, start with [92,93]. OPV performance is strongly dependent on the morphology, and Refs. [94,95] provide overviews of the key factors governing charge generation, separation, and transport. A review summarizing computational OPV morphology prediction at different length-scales is presented in Ref. [96]. Computational predictions of OPV morphologies are used as inputs into charge transport simulations that link OPV structure to metrics determining their efficiency. Refs. [95,97] explain charge generation and transport, while [98] explains how these properties can be simulated with kinetic Monte Carlo algorithms. We summarize recent morphology and charge transport predictions of the benchmark OPV material poly(3-hexylthiophene) (P3HT) in Table 2. Combining hardware, software, and coarse-graining advances, routine simulations of P3HT have improved by roughly four orders of magnitude over the last decade (0.6 monomer- μ s in 2010 vs. 6900 monomer- μ s in 2018).

A challenge to making efficient OPVs is determining which combinations of photoactive compounds and thermodynamic conditions (temperature, pressure, concentrations) will result in a favorable morphology—a task well-suited to MD simulation. Probing this vast data space requires organizing ensembles of simulations, distributing them on high-performance computing clusters, retrieving the data, and then distilling the data into understandable chunks.

In our recent work [111,112,11,109], we use HOOMD-Blue (`hoomd`) to predict morphologies of perylene, perylothiophene, P3HT, and poly(benzodithiophene-thienopyrrolo-dione) (BDT-TPD) oligomers using simplified (united atom) models. Neglecting partial charges and treating conjugated systems as rigid are two assumptions that lower cognitive load associated with force fields, avoiding the first-principles

calculation of unknown charge, dihedral, and angle parameterizations missing from OPLS-UA or GAFF. These simplifications helped with both training and scaling timescale problems, resulting in morphology predictions in agreement with experiments [111,112,11] (see Figs. 3 and 4).

An example of developing transferable skills to deal with combinatorial explosion occurred in this P3HT work: Only 14 temperatures, 6 solvent strengths, and 5 densities equates to 420 unique simulations. In each of these 420 cases, we aim to understand how the proximity and orientation of thiophene rings correlates with charge transport. A single structural descriptor is applied to each case, and a “phase-diagrams” is constructed for each density, providing a handful of figures summarizing large data spaces (Fig. 5).

Testing transferability, reproducibility, usability, and extendibility of OPV modeling techniques is an exciting area of future work. For P3HT in particular, the number of models, simulation engines, and sampling schemes used makes it a good candidate for evaluating TRUEness. With a myriad of open approaches for coarse-graining, there is no fundamental reason why multiscale efforts validated by different groups could not be a test-bed for testing reproducibility these scales. Similarly, the availability of various charge-transport calculation approaches provides for opportunities to reproduce predictions of how charge transport depends on morphology and chemistry. Such cross-validation teams would help accelerate improvements around charge transfer calculations themselves, where opportunities exist to improve understanding of this broadly applicable phenomenon.

6. Predicting crosslinking dynamics

In this section we review computational approaches to predicting the crosslinked networks of toughened thermosets and discuss our recent work in this context. Thermosets are strong, low-density materials formed by the covalent bonding of liquid precursors into a 3D network that can be made less brittle through the introduction of a thermoplastic “toughener”. In the fabrication of composite materials made from carbon fibers impregnated with toughened thermosets the network is “cured” through the heating and cooling of a part over time. The temperature history experienced by the part during curing influences the rates of diffusion and reaction, and therefore its resulting nanostructure and residual stresses. As the thermoset precursors crosslink, there is an entropic driving force for the phase-separation of the thermoplastic [113], which complicates nanostructure evolution. For a review of the key concepts in modeling thermosets (cure fraction, gelation, and glass transition temperature) see Li and Strachan [114]. The challenge focused on here is using molecular simulations to predict how thermoset formulation, toughener chemistry, and temperature history determine the cured nanostructure.

The central problems are those of scaling and training timescales, plus the fact that reacting systems are not in equilibrium. On the sampling side, the slow dynamics of gelling, glassy thermosets make relaxation intractably long even for small systems. Further, validating simulations against experimental systems with 1 nm-100 nm phase-separated length scales demands large simulated volumes. On the training side the main tensions are between faithful representation of reaction kinetics, coarse models that enable access to long timescales, and implementing these simultaneously. The knowledge that the equilibrium integration schemes available in `hoomd` and `lammps` are in conflict with the exothermic formation of bonds, and that using ReaxFF [115] won’t permit sufficient volumes to be accessed is liberating: It allows the question to be reframed as “How predictive of nanostructure can a simplified model of crosslinking thermosets be?”.

To advance towards the goal of large, fast, predictive thermoset simulations we develop `epoxy` as detailed in Ref. [116]. This was the first project in which we employed continuous integration into model development. Sanity checks built around initialization of the tougheners and the generation of trajectories with and without the reaction

Table 2

Overview of recent computational studies of P3HT, including method (MD or MC), resolution (AA – all-atom, UA – united-atom, 3CG – coarse-grained with 3 simulation elements per repeat unit, 1CG – coarse-grained with one element per repeat unit), approximate number of repeat units simulated, simulation time, computational effort (estimated largest product of Repeats Units \times Time), and if the structures were used for charge-transport calculations. The work of Carrillo et al. [99] is a notable outlier, having successfully combined coarse models of millions of repeat units with development access to the then-most-powerful supercomputer on the planet. *Explicit numbers were not provided in the report, but are estimated here.

Year	Study	Method	Model	Repeat Units	Simulation Time	Effort (μs)	CT
2010	Moreno [100]	MD	AA	300	2 ns	6.0×10^{-1}	No
2010	Huang [101]	MD	AA	720	5–35 ns	2.5×10^1	No
2010	Huang [101]	MD	3CG	36864	10 ns	3.7×10^2	No
2011	Lee [102]	MD	1CG	X	10 ns	X	No
2013	Bhatta [103]	MD	AA	320–1280	5 ns	6.4×10^0	No
2013	D'Avino [104]	MD	UA	1600	60 ns	9.6×10^0	Yes
2013	Alexiadis [105]	MD	AA	2700*	20–45 ns	1.2×10^2	No
2013	Jankowski [106]	MD	3CG	2250–3750	1.7 μs	6.4×10^3	No
2013	Carrillo [99]	MD	1CG	3.2×10^6	400 ns	1.3×10^6	No
2016	Jones [107]	MD	3CG	4600–17000*	8 ns	1.4×10^2	Yes
2016	Scherer [108]	MD	3CG	8000	80 ns	6.4×10^2	No
2017	Jones [46]	MD	3CG	2250–3750	1.7 μs	6.4×10^3	Yes
2018	Miller [11,109]	MD	UA	1500–15000	0.3–3 μs	6.9×10^3	Yes
2019	Greco [110]	MC	1CG	8000–16000	N/A	N/A	Yes

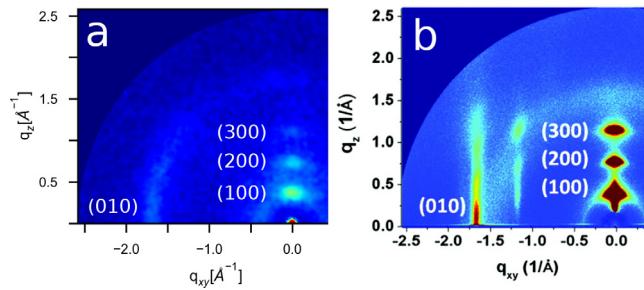


Fig. 3. a) Simulated and b) experimental grazing incident X-ray scattering of P3HT show near identical features and wavenumbers along the (010) and (100) planes. The agreement indicates the same structures are being probed in both cases. Figure adapted with permission from Ref. [11].

algorithms implemented allowed the submission of large ensembles of jobs to multiple clusters with confidence. This further allowed the research team to quickly progress through a series of models in support of the science question:

- DPD models are good for simplicity and performance, but not for representing entangled glasses
- LJ potentials and bond constraints can be parameterized to model entangled glasses
- Angle constraints are needed here for T_g measurements to fit the DiBenedetto expression
- In some cases, million-particle systems are needed to capture the microphase separated morphologies (Fig. 6)
- Bond-forming models can be made with `hoomd` plugins and calibrated against reaction kinetic models with and without heats of reaction

The results of our approach are summarized with recent simulations in Table 3. The two distinguishing features of our recent work ([116]) are (1) the ability to investigate structural evolution while the model epoxies cure and (2) ability to do so for million-particle volumes in a few days or weeks on a single GPU.

Despite the simplifications, our coarse simulations match experimental reaction dynamics and glass transition temperatures [124]. Further, because we can vary temperature over the course of these reacting systems we can for the first time use MD to investigate how nanostructure depends upon temperature history during curing. Here

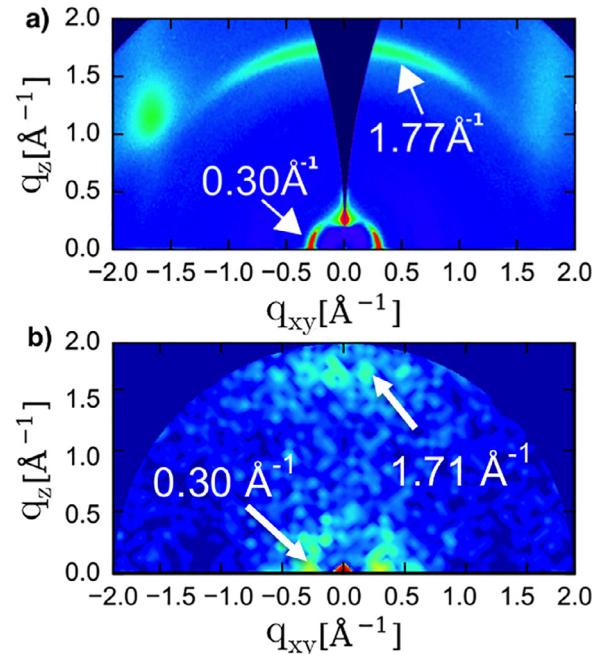


Fig. 4. a) Experimental and b) simulated grazing incident X-ray scattering of BDT-TPD showing agreement. The agreement validates our simplified model. Reprinted with permission from [112]. Copyright 2017 American Chemical Society.

we present new results (Fig. 7 and Fig. 8) summarizing the evolution of structure in two types of curing simulations where the primary activation energy is 2.1 dimensionless energy units, the secondary $E_A = 2.52$, $N = 400000$, $L = 73.7\text{nm}$ and $dt = 0.01$, using the Lennard-Jones parameters from Table 5.1 and the fiducial simulation parameters from Table 5.2 of [124]. Specifically, the ratio of coarse amine, epoxy, and toughener (A, B, and C) considered here is 1:2:2, with $\epsilon_{AA} = 0.9216$, $\epsilon_{BB} = 1.0$, $\epsilon_{CC} = 0.8840$, $\epsilon_{AB,AC,BC}$ are obtained using the Lorentz-Berthelot mixing rule (e.g. $\epsilon_{AB} = \sqrt{\epsilon_{AA}\epsilon_{BB}}$), harmonic bond $r_0 = 1.0$ and $k = 100\frac{\epsilon}{r_0^2}$. The Langevin thermostat with drag parameter $\gamma = 4.5$ is used to advance simulation trajectories. Simulations were performed on NVIDIA K40 cards using `hoomd` 2.2.1 (commit hash f664aebedf55e44f10cdd6d5edc3a090f1bca713b),

In both figures, the wavenumber associated with microphase

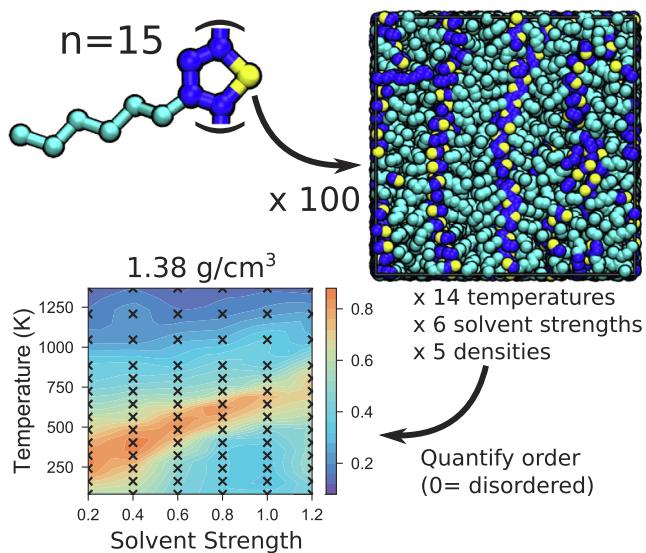


Fig. 5. 100 P3HT chains of 15 repeat units are represented as three simulation species: Yellow (sulfur), blue (aromatic carbon), and cyan (aliphatic carbon). Temperature, solvent strength (specified as a scaling of the Lennard-Jones well depth) and density, determine thermodynamic self-assembly of 420 unique structures. The self-assembled structures are quantified for ordering on the interval $[0, 1]$ – where 0 is completely disordered and 1 is completely ordered, according to the clustering of neighboring repeat units based on relative distances and orientations. Ordering is then summarized into “phase-diagrams” depicting the order as a function of these three variables; demonstrating how hundreds of simulations can be distilled to a few, quickly interpretable figures.

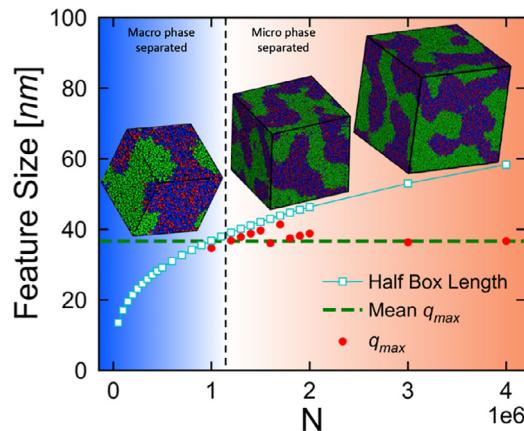


Fig. 6. The ability to perform curing simulations of million-particle toughened thermoset models enables the identification of sufficient box sizes. Here, divergence of the low-wavenumber structure factor is used to identify macrophase separation, and for small volumes (blue, left) of this CG model the morphologies appear macrophase separated. Large volumes (orange, right) of the same model show a local maximum in the structure factor (q_{max}) indicating microphase separation is observable when the length-scales of separation are smaller than half the simulation box length. Here, 1.2×10^6 particles (35 nm boxes) are needed, reinforcing the importance of fast, large simulations for studying toughened epoxy thermosets.

separation is plotted vs. time, with the simulation temperature (red) overlaid, along with the system’s glass transition temperature (purple) which is a function of the degree-of-cure. In [Fig. 7](#), the sample being cured at 0.8 kT (above T_g) is quenched to 0.5 kT (below T_g) before gelation, and in [Fig. 8](#) the quench occurs after the onset of gelation. In both cases, the morphologies achieve the same degree of cure ($\alpha = 0.78$), and the standard error of five independent simulations are plotted with the grey error bars. We observe that curing post-gelation narrows the variance in cured structure, and these results demonstrate

Table 3

Overview of recent reacting epoxy models, sorted by system size. Most efforts do not capture dynamic bonding during MD integration nor validate glass transition (T_g) as a function of cure fraction α (DiBenedetto expression).

Model (Chemistry)	Bond Probability	System Size	$T_g(\alpha)$ Validation
CGMD [117] EP/CA	Arbitrary	5.0×10^3	No
AAMD [118] (DGEBA/DETA)	Arbitrary	5.9×10^3	No
CGMD/AAMD [119] (DGEBA/DETA)	1	2.2×10^4	No
AAMD [120] (DGEBA/33DDS)	1	6.9×10^4	No
AAMD [121] (DGEBA/44DDS)	1	9.8×10^4	No
DPD [122] (DGEBA/DETA)	1	1.1×10^5	No
DPD [123] (DDS/RA/SA)	0.001	2.5×10^5	No
CGMD/DPD [124] (DGEBA/44DDS/PES)	$\sim \exp(\frac{E_a}{k_B T})$	4.0×10^6	Yes

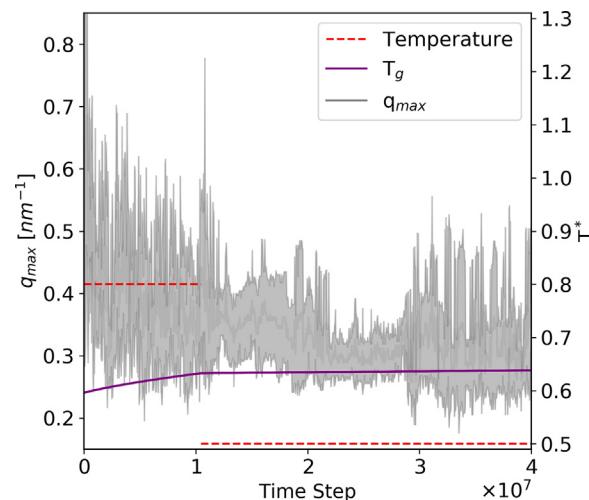


Fig. 7. Time evolution of the dominant length scale measured by the toughener-toughener structure factor for toughened reacting epoxy thermosets quenched below T_g (solid line) before gelation at time step 10,480,000. Curing temperature is shown by the dotted line. Error bars represent standard error from five independent simulations.

the importance of temperature history on cured morphology.

The present example combining simplified models with continuous integration—two of the best practices from [section 3](#)—demonstrates improved understanding of materials behavior. Because the code is open, the data is available, and because experimentation in this area is active, we identify epoxy thermosets as an area where we expect development around TRUE simulations to accelerate. Community validation of morphology predictions from simulations with varied temperature histories offers opportunity to increase the industrial impact of molecular simulations.

7. Training new simulators

In this section we describe several examples of on-boarding students to new projects wherein open tools (namely `hoomd`, `mBuild`, `foyer`, `signac`) and Software Carpentry pedagogy are used to reduce cognitive load and aid reproducibility. In addition to using the aforementioned tools directly in python scripts, we develop two packages, `Rhaco` [60] and `Planckton` [57], that combine these tools to accomplish common tasks for specific systems. `Rhaco` facilitates the initialization and

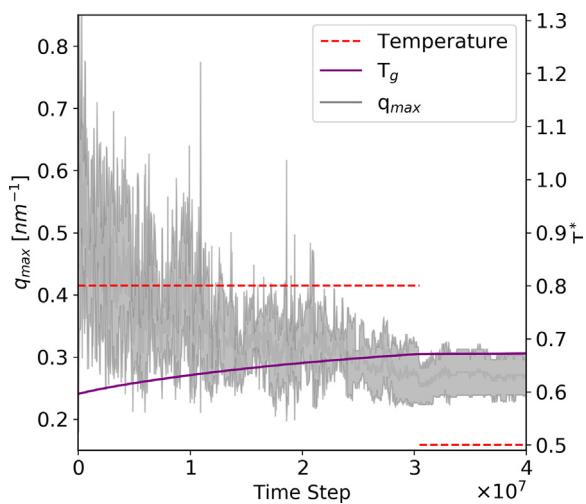


Fig. 8. Time evolution of the dominant length scale measured by the toughener-toughener structure factor for toughened reacting epoxy thermosets quenched below T_g (solid line) after gelation at time step 30,480,000. Curing temperature is shown by the dotted line. Error bars represent standard error from five independent simulations.

simulation of matter near surfaces and Planckton provides infrastructure for coupling MD simulations of OPVs to charge transport simulations (Table 1). Examples of the diverse surface systems investigated with Rhaco are summarized in Fig. 9 and have enabled broad testing of forcefield compatibilities and qualitative investigation of surface phenomena while teaching students from multiple disciplines. In Fig. 10 we summarize DNA, fullerene, OPV material, asphaltene, and patchy particle models developed by new students leveraging mBuild and hoomd. These examples are representative of students being able to initialize and debug models in weeks. In the case of Fig. 10a, quickly moving past the initialization step of building coarse-grained DNA identified issues with our implementation of the Knotts model. In Fig. 10b, the minimal physics for polar fullerene oxides solubilizing C_{60} was identified by trying seven coarse models. In Fig. 10c, initializing the complex truxene (a new candidate molecule for OPVs) with foyer immediately identified needed dihedral constraints. The asphaltenes

components in Fig. 10d can be tuned for molecular weights and number of chains and was accomplished by an REU team. The patchy trapezoids in Fig. 10e permit quick testing of how patch size and shape influences assembly propensity.

By combining tools for particular applications, we create “higher-level” languages for describing the system initializations in these examples. This enables the concepts of the models to be probed faster, lowering the load associated with initialization and parameterization, and the management of conversion between units and dimensionless quantities used in hoomd. Further, the code development benefits are bidirectional: By engaging with developer communities, students broaden their network of support and provide feedback that informs tool development. All of the repositories mentioned here (hoomd, mBuild, foyer, signac, and Software Carpentry’s instructor training) have now merged student pull requests originating from missing features or bugs encountered en route to the above work. These examples dispel the notion that aiming for simplicity necessarily creates “black-boxes” that limit student understanding of details, or the notion that higher-level languages necessarily create dependencies on established techniques. Rather, aiming for low cognitive load can focus development of new techniques and features with the best payoff for the researcher. Students anecdotally report increased interest and confidence, and this area represents a potential opportunity for future studies of professional identity, retention, and long-term career outcomes [126].

Related to the professional identities of materials simulators is the idea that certain scientist roles might enhance TRUE simulations. For example, should training prioritize the development of tool developers and tool users separately, or jack-of-all-trades scientists that can do everything? We find guidance to answering this question from our experiences with distributed code development: It is impossible for any individual to master everything, so leveraging the knowledge embedded in diverse development communities should be prioritized. Prioritizing diverse development communities highlights the need for the individuals in these communities to communicate and collaborate. The existence of and adherence to community codes of conduct (see The Carpentries at <https://bit.ly/2GR6pKs>) helps to ensure communities practice inclusivity and fosters collaboration. Within such communities we do not know if the prioritization of particular roles is yet warranted to enhance the realization of TRUE simulations, but we hold the optimistic opinion that communities offer the opportunity for the

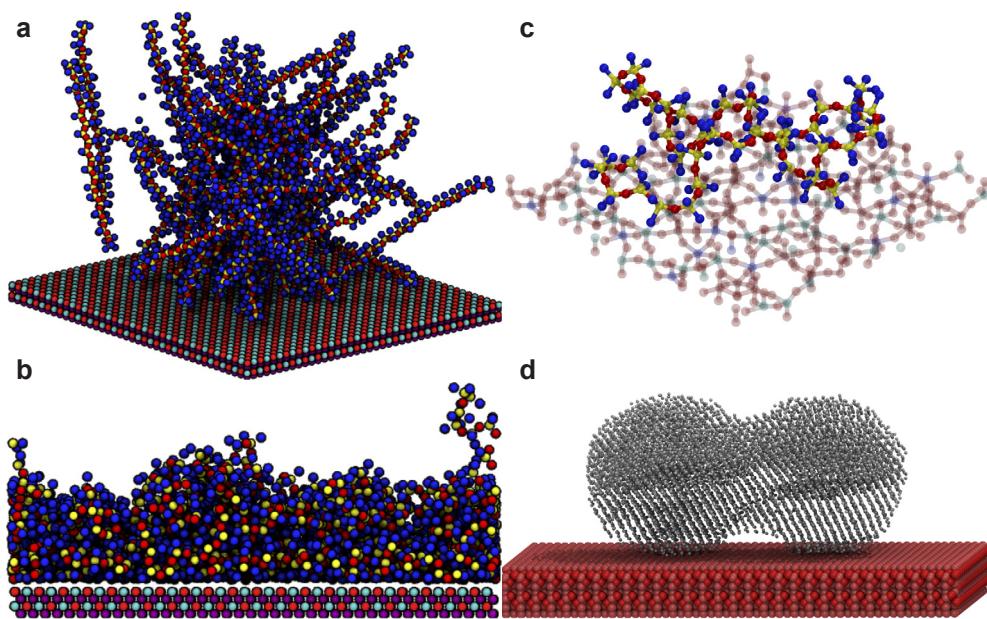
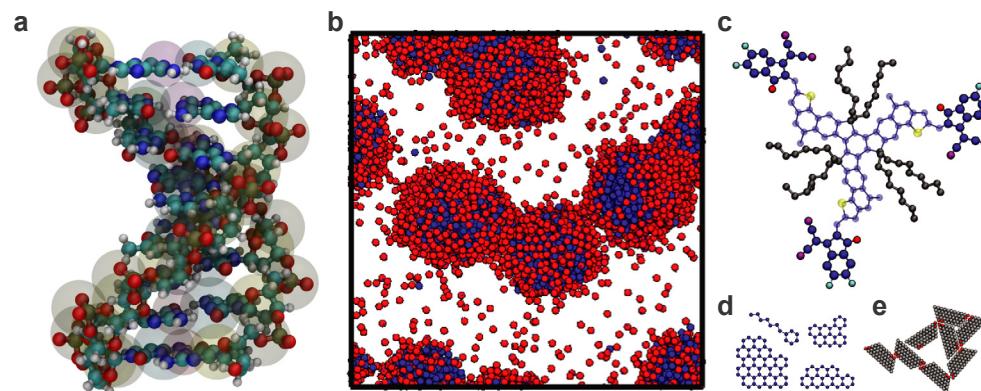


Fig. 9. a) PDMS chains initialized over NiMnGa, b) configuration of PDMS on NiMnGa by combining UFF and an OPLS-UA-derived potential, c) PDMS initialized on a M1 surface, d) Sintering silver nanoparticles on a corundum surface combining EAM and UFF.



ensemble of individual interests and strengths to overlap in a way that makes any individual's shortcomings irrelevant.

8. Outlook

This instant in time represents a period of major change in the materials simulation ecosystem. Distributed coding communities have appeared, grown (Fig. 2), and disrupted the development practices and availability of simulation tools, continuing progress towards TRUE simulations since the field's origins [127,128]. In this perspective we make the case for thoughtful training to take a central role in enhancing research reproducibility while simultaneously training researchers for broadly-needed technical roles. We also make the case that *en route* to TRUE simulations, these simulations should begin as less "true": Lowering cognitive load by sacrificing completeness now is made up for by increased efficiency and correctness later. However, TRUE simulations are not yet the norm. Materials simulators are in a position of opportunity and responsibility: We can demonstrate how reproducible science can be performed through increased engagement with our peers in the coming years. If the current momentum around communication and collaboration wanes, however; if the community becomes less inclusive rather than more, we may expect the amount of beeping prompts to increase. We take a more optimistic view: There has never been a better time to be a molecular simulator because of how active the community is with helping train its members to do transferable, reproducible, usable, extensible science.

Declaration of Competing Interest

The authors declare that they have no conflicts of interest to this work.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grants Nos. 1835593, 1653954, 1658076, and 1229709. This work used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation Grant No. ACI-1053575 [4]. This research is part of the Blue Waters sustained-petascale computing project, which is supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) and the state of Illinois. The authors gratefully acknowledge sponsorship through the Laboratory Directed Research and Development (LDRD) program of the Idaho National Laboratory. This material is based upon work supported by The Boeing Company under contract BRT-LO118-0015. We acknowledge high-performance computing support of the R2 compute cluster (DOI: 10.18122/B2S41H) provided by Boise State University's Research Computing Department. Blue Waters is a joint effort of the University of Illinois at Urbana-

Fig. 10. a) Coarse-grained DNA initialized directly from sequence of nucleobases using the model from [125], b) Micelle self-assembly from a coarse-grained model of fullerene and their oxides, c) truxene molecule with electroactive components core and three functional groups, d) examples of programmatically-generated asphaltene components, e) 2D patchy particles designed to self-assemble terminal structures.

Champaign and its National Center for Supercomputing Applications. KN, BA, MA, JG and EJ thank the Shodor Education Foundation and the Blue Waters Student Internship Program for support of this work. The raw and processed data required to reproduce these findings are available to download from <https://doi.org/10.5281/zenodo.2619585>.

References

- [1] S.E. Page, *The Model Thinker: What You Need to Know to Make Data Work for You*, Hachette, UK, 2018.
- [2] M. Baker, 1,500 scientists lift the lid on reproducibility, *Nature* 533 (7604) (2016) 452–454, <https://doi.org/10.1038/533452a> URL: <http://www.nature.com/doifinder/10.1038/533452a>.
- [3] M. Smith, Computer Science for All (2016).<https://www.whitehouse.gov/blog/2016/01/30/computer-science-all>.
- [4] J. Towns, T. Cockerill, M. Dahan, I. Foster, K. Gaither, A. Grimshaw, V. Hazlewood, S. Lathrop, D. Lifka, G.D. Peterson, R. Roskies, J.R. Scott, N. Wilkins-Diehr, XSEDE: accelerating scientific discovery, *Comput. Sci. Eng.* 16 (5) (2014) 62–74, <https://doi.org/10.1109/MCSE.2014.80> URL: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6866038>.
- [5] J. Wernert, L. Rivera, 2018 eXtreme Science and Engineering Discovery Environment (XSEDE) Annual User Satisfaction Survey Evaluation Report Prepared for: John Towns Prepared by Lizzanne DeStefano Table of Contents (September).
- [6] T.K. Teal, K.A. Cranston, H. Lapp, E.P. White, G. Wilson, K. Ram, A. Pawlik, Data carpentry: workshops to increase data literacy for researchers, *Int. J. Digit. Curation* 10 (2015) 135–143, <https://doi.org/10.2218/ijdc.v10i1.351> URL: <http://www.ijdc.net/index.php/ijdc/article/view/351>.
- [7] A.S. Vasilevich, A. Carlier, J. de Boer, S. Singh, How not to drown in data: a guide for biomaterial engineers, *Trends Biotechnol.* 35 (8) (2017) 743–755, <https://doi.org/10.1016/j.tibtech.2017.05.007> URL: <https://doi.org/10.1016/j.tibtech.2017.05.007> <https://linkinghub.elsevier.com/retrieve/pii/S0167779917301270>.
- [8] P.T. Cummings, Computational screening of soft materials systems with application to nano-lubrication systems, *Telluride Sci. Res. Cent. Work. Mol. Eng. soft matter Spanning small Mol. to Macromol.*, Telluride, CO (2017), <https://doi.org/10.6084/m9.figshare.89666312> [https://figshare.com/articles/Telluride_\(Meeting_2017-06-23-2017-final\).pdf/89666312](https://figshare.com/articles/Telluride_(Meeting_2017-06-23-2017-final).pdf/89666312).
- [9] E. Braun, J. Gilmer, H.B. Mayes, D.L. Mobley, J.I. Monroe, S. Prasad, D.M. Zuckerman, Best practices for foundations in molecular simulations [Article v1.0], *Living J. Comput. Mol. Sci.* 1 (1) (2018), <https://doi.org/10.33011/livcoms.1.1.5957> URL: <https://bit.ly/2HPkcm>.
- [10] D. Frenkel, Simulations: the dark side, *Eur. Phys. J. Plus* 128 (1) (2013) 10, <https://doi.org/10.1140/epjp/i2013-13010-8> URL: <http://link.springer.com/10.1140/epjp/i2013-13010-8>.
- [11] E.D. Miller, M.L. Jones, M.M. Henry, P. Chery, K. Miller, E. Jankowski, Optimization and validation of efficient models for predicting polythiophene self-assembly, *Polymers (Basel)* 10 (12) (2018) 1305, <https://doi.org/10.3390/polym10121305> URL: <http://www.mdpi.com/2073-4360/10/12/1305>.
- [12] J.E. Stone, D.J. Hardy, I.S. Ufimtsev, K. Schulten, GPU-accelerated molecular modeling coming of age, *J. Mol. Graph. Model.* 29 (2) (2010) 116–125, <https://doi.org/10.1016/j.jmgm.2010.06.010> URL: <https://doi.org/10.1016/j.jmgm.2010.06.010> <https://linkinghub.elsevier.com/retrieve/pii/S109326310000914>.
- [13] J.A. Anderson, C.D. Lorenz, A. Traverset, General purpose molecular dynamics simulations fully implemented on graphics processing units, *J. Comput. Phys.* 227 (10) (2008) 5342–5359, <https://doi.org/10.1016/j.jcp.2008.01.047> URL: <http://www.sciencedirect.com/science/article/pii/S0021999108000818> <https://linkinghub.elsevier.com/retrieve/pii/S0021999108000818>.
- [14] C.R. Trott, LammmpsCuda-a new GPU accelerated molecular dynamics simulations package and its application to ion-conducting glasses, *Technische Universität Ilmenau* (2011) (Ph.D. thesis).
- [15] J.A. Anderson, S.C. Glotzer, The development and expansion of HOOMD-blue through six years of GPU proliferation, *arXiv* 1308.5587. <http://arxiv.org/abs/1308.5587>

arxiv.org/abs/1308.5587.

[16] J. Glaser, T.D. Nguyen, J.A. Anderson, P. Lui, F. Spiga, J.A. Millan, D.C. Morse, S.C. Glotzer, Strong scaling of general-purpose molecular dynamics simulations on GPUs, *Comput. Phys. Commun.* 192 (2014) 97–107, <https://doi.org/10.1016/j.cpc.2015.02.028> arXiv:1412.3387, <http://arxiv.org/abs/1412.3387> <http://linkinghub.elsevier.com/retrieve/pii/S0010465515000867>.

[17] M. Zheng, X. Li, L. Guo, Algorithms of GPU-enabled reactive force field (ReaxFF) molecular dynamics, *J. Mol. Graph. Model.* 41C (2010) (2013) 1–11, <https://doi.org/10.1016/j.jmgm.2013.02.001> URL: <http://www.ncbi.nlm.nih.gov/pubmed/23454611>.

[18] G. Wilson, Software Carpentry: lessons learned, *F1000Research* 3 (2014) 62, <https://doi.org/10.12688/f1000research.3-62.v1> arXiv:1307.5448, <http://f1000research.com/articles/3-62/v1>.

[19] S.A. Ambrose, M.W. Bridges, M. DiPietro, M.C. Lovett, M.K. Norman, *How Learning Works: Seven research-based principles for smart teaching* vol. 48, John Wiley & Sons, 2010, <https://doi.org/10.1002/mop.21454> URL: <https://firstliteracy.org/wp-content/uploads/2015/07/How-Learning-Works.pdf>.

[20] D. Kahneman, *Attention and effort*, Citeseer, 1973.

[21] S. Cheryan, V.C. Plaut, P.G. Davies, C.M. Steele, Ambient belonging: how stereotypical cues impact gender participation in computer science, *J. Pers. Soc. Psychol.* 97 (6) (2009) 1045.

[22] J.R. Shapiro, A.M. Williams, The Role of stereotype threats in undermining girls' and women's performance and interest in STEM fields, *Sex Roles* 66 (3–4) (2012) 175–183, <https://doi.org/10.1007/s11199-011-0051-0> URL: <http://link.springer.com/10.1007/s11199-011-0051-0>.

[23] J.N. Schinske, H. Perkins, A. Snyder, M. Wyer, Scientist spotlight homework assignments shift students' stereotypes of scientists and enhance science identity in a diverse introductory science class, *CBE Life Sci. Educ.* 15 (3) (2016) 1–18, <https://doi.org/10.1187/cbe.16-01-0002>.

[24] A.C. Bart, A. Sarver, M. Friend, L. Cox II, PythonSneks: an open-source, instructionally-designed introductory curriculum with action-design research, *Proc. 50th ACM Tech. Symp. Comput. Sci. Educ. – SIGCSE 19* (2019) 307–313, <https://doi.org/10.1145/3287324.3287428> URL: <http://dl.acm.org/citation.cfm?doid=3287324.3287428>.

[25] M.P. Cook, Visual representations in science education: The influence of prior knowledge and cognitive load theory on instructional design principles, *Sci. Educ.* 90 (6) (2006) 1073–1091, <https://doi.org/10.1002/sce.20164> URL: <http://onlinelibrary.wiley.com/doi/10.1002/sce.20164/abstract> <http://doi.wiley.com/10.1002/sce.20164>.

[26] A. Simperler, G. Wilson, Software Carpentry get more done in less time, *Arxiv Prepr. arXiv:1506.02575*. <http://arxiv.org/abs/1506.02575>.

[27] G. Wilson, D.A. Aruliah, C.T. Brown, N.P. Chue Hong, M. Davis, R.T. Guy, S.H.D. Haddock, K.D. Huff, I.M. Mitchell, M.D. Plumley, B. Waugh, E.P. White, P. Wilson, N.P.C. Hong, Best Practices for Scientific Computing, *PLoS Biol.* 340 (2014), <https://doi.org/10.1371/journal.pbio.1001745> e1001745, arXiv:1210.0530v1, <http://arxiv.org/abs/1210.0530> <http://www.ncbi.nlm.nih.gov/articlerender.fcgi?artid=3886731&tool=pmcentrez&rendertype=abstract>.

[28] B.M. Duckles, Value of Software Carpentry to Instructors Report, Tech. Rep. January, Foundation, Software Carpentry Foundation, Data Carpentry, Portland, OR (2016).

[29] N. Hollender, C. Hofmann, M. Deneke, B. Schmitz, Integrating cognitive load theory and concepts of human-computer interaction, *Comput. Human Behav.* 26 (6) (2010) 1278–1288, <https://doi.org/10.1016/j.chb.2010.05.031> URL: <https://doi.org/10.1016/j.chb.2010.05.031>.

[30] S.A. Bohner, S. Mohan, Model-based engineering of software: Three productivity perspectives, *Proc. – 33rd Annu. IEEE Softw. Eng. Work. SEW-33* 2009 (2010) 35–44, <https://doi.org/10.1109/SEW.2009.19>.

[31] G. Varga, A. Ledeczi, C.R. Iacovella, P.T. Cummings, J. Sallai, A. Ledeczi, C.R. Iacovella, C. McCabe, P.T. Cummings, Enabling Cross-Domain Collaboration in Molecular Dynamics Workflows, *Fourth Int. Conf. Adv. Collab. Networks, Syst. Appl. (COLLA 2014)* (2014) 41–47.

[32] D.M. Oppenheimer, Consequences of erudite vernacular utilized irrespective of necessity: problems with using long words needlessly, *Appl. Cogn. Psychol.* 20 (2) (2006) 139–156, <https://doi.org/10.1002/acp.1178> URL: <http://doi.wiley.com/10.1002/acp.1178>.

[33] J.W. Ponder, D.A. Case, Protein Simulations, *Adv. Protein Chem.* 66 (2003) 27–85, [https://doi.org/10.1016/S0065-3233\(03\)66002-X](https://doi.org/10.1016/S0065-3233(03)66002-X) URL: <http://www.sciencedirect.com/science/article/pii/S006532330366002X>.

[34] M.S. Daw, M.I. Baskes, Semiempirical, quantum mechanical calculation of hydrogen embrittlement in metals, *Phys. Rev. Lett.* 50 (17) (1983) 1285–1288, <https://doi.org/10.1103/PhysRevLett.50.1285>.

[35] N.L. Allinger, K. Chen, J.H. Lii, An improved force field (MM4) for saturated hydrocarbons, *J. Comput. Chem.* 17 (5–6) (1996) 642–668, [https://doi.org/10.1002/\(SICI\)1096-987X\(199604\)17:5/6<642::AID-JCC6>3.0.CO;2-U](https://doi.org/10.1002/(SICI)1096-987X(199604)17:5/6<642::AID-JCC6>3.0.CO;2-U).

[36] W.L. Jorgensen, J. Tirado-Rives, The OPLS [optimized potentials for liquid simulations] potential functions for proteins, energy minimizations for crystals of cyclic peptides and crambin, *J. Am. Chem. Soc.* 110 (6) (1988) 1657–1666, <https://doi.org/10.1021/ja00214a001> URL: <http://pubs.acs.org/doi/abs/10.1021/ja00214a001>.

[37] J. Wang, R.M. Wolf, J.W. Caldwell, P.A. Kollman, D.A. Case, Development and testing of a general Amber force field, *J. Comput. Chem.* 25 (9) (2004) 1157–1174, <https://doi.org/10.1002/jcc.20025> arXiv:2024.

[38] H. Sun, COMPASS: an ab initio force-field optimized for condensed-phase applications—overview with details on alkane and benzene compounds, *J. Phys. Chem. B* 102 (38) (2002) 7338–7364.

[39] S.J. Keasler, S.M. Charan, C.D. Wick, I.G. Economou, J.I. Siepmann, Transferable potentials for phase equilibria-united atom description of five- and six-membered cyclic alkanes and ethers, *J. Phys. Chem. B* 116 (36) (2012) 11234–11246, <https://doi.org/10.1021/jp302975c>.

[40] D.N. LeBard, B.G. Levine, P. Mertmann, S.A. Barr, A. Jusufi, S. Sanders, M.L. Klein, A.Z. Panagiotopoulos, A. Barr, A. Jusufi, S. Sanders, L. Klein, Self-assembly of coarse-grained ionic surfactants accelerated by graphics processing units, *Soft Matter* 8 (8) (2012) 2385–2397, <https://doi.org/10.1039/C1SM06787G> URL: <http://xlink.rsc.org/?DOI=C1SM06787G> <http://xlink.rsc.org/?DOI=c1sm06787g>.

[41] T.A. Halgren, W. Damm, Polarizable force fields, *Curr. Opin. Struct. Biol.* 11 (2) (2001) 236–242, [https://doi.org/10.1016/S0959-440X\(00\)00196-2](https://doi.org/10.1016/S0959-440X(00)00196-2) URL: http://ac.els-cdn.com/S0959440X00001962/s2.0-S0959440X00001962-main.pdf?_tid=3e24fd1e-415c-11e7-bada-00000aabb35e&acdnat=1495725195_b9b23eab11032341bb59fde8955297.

[42] E. von Hippel, G. von Krogh, Open source software and the "Private-Collective" innovation model: issues for organization science, *Organ. Sci.* 14 (2) (2003) 209–223, <https://doi.org/10.1287/orsc.14.2.209.14992> URL: <http://pubsonline.informs.org/doi/abs/10.1287/orsc.14.2.209.14992>.

[43] M.M. Wasko, S. Faraj, Why should i share? Examining social capital and knowledge contribution in electronic networks of practice, *MIS Q.* 29 (1) (2005) 35, <https://doi.org/10.2307/25148667> URL: <http://www.jstor.org/stable/25148667> <https://www.jstor.org/stable/10.2307/25148667>.

[44] E. Jankowski, AIChE 2018 CoMSEF Hands-On Workshop Molecular Modeling Toolbox, 2018. https://doi.org/10.18122/cme_lab/5/boisestate.

[45] D.S. Katz, L.C. McInnes, D.E. Bernholdt, A.C. Mayes, N.P.C. Hong, J. Duckles, S. Gesing, M.A. Heroux, S. Hettrick, R.C. Jimenez, M. Pierce, B. Weaver, N. Wilkins-Diehr, Community organizations: changing the culture in which research software is developed and sustained, *Comput. Sci. Eng.* 21 (2) (2019) 8–24, <https://doi.org/10.1109/MCSE.2018.2883051> URL: <https://ieeexplore.ieee.org/document/8565942>.

[46] M.L. Jones, E. Jankowski, Computationally connecting organic photovoltaic performance to atomistic arrangements and bulk morphology, *Mol. Simul.* 43 (10–11) (2017) 756–773, <https://doi.org/10.1080/08927022.2017.1296958> URL: <https://www.tandfonline.com/doi/full/10.1080/08927022.2017.1296958> <https://doi.org/10.1080/08927022.2017.1296958>.

[47] E.D. Miller, M.M. Henry, E. Jankowski, Diffractometer (aug 2018). <https://doi.org/10.5281/zenodo.1340716> <https://zenodo.org/record/1340716>.

[48] C. Klein, A.Z. Summers, M.W. Thompson, J. Gilmer, C. McCabe, P.T. Cummings, J. Sallai, C.R. Iacovella, Formalizing atom-typing and the dissemination of force fields with foyer, *Comput. Mater. Sci.* 167 (May) (2018) 215–227, <https://doi.org/10.1016/j.commatsci.2019.05.026> arXiv:1812.06779, <http://arxiv.org/abs/1812.06779>.

[49] E.S. Harper, M. Spellings, J. Anderson, S.C. Glotzer, harperic/freud: Zenodo DOI release (nov 2016). <https://doi.org/10.5281/ZENODO.166564> <https://zenodo.org/record/166564>.

[50] C. Klein, J. Sallai, T.J. Jones, C.R. Iacovella, C. McCabe, P.T. Cummings, A hierarchical, component based approach to screening properties of soft matter, *Found. Mol. Model. Simul.* <https://doi.org/10.1007/978-981-10-1128-3-5>.

[51] R. Gowers, M. Linke, J. Barnoud, T. Reddy, M. Melo, S. Seyler, J. Domański, D. Dotson, S. Buchoux, I. Kenney, O. Beckstein, MDAnalysis: a python package for the rapid analysis of molecular dynamics simulations, in: *Proc. 15th Python Sci. Conf.*, 2016, pp. 98–105. <https://doi.org/10.25080/Majora-629e541a-00e> <https://conference.scipy.org/proceedings/scipy2016/oliver.beckstein.html>.

[52] R.T. McGibbon, K.A. Beauchamp, M.P. Harrigan, C. Klein, J.M. Swails, C.X. Hernández, C.R. Schwantes, L.P. Wang, T.J. Lane, V.S. Pande, MDTraj: a modern open library for the analysis of molecular dynamics trajectories, *Biophys. J.* 109 (8) (2015) 1528–1532, <https://doi.org/10.1016/j.bpj.2015.08.015>.

[53] M.L. Jones, MorphCT – DOI: 10.5281/zenodo.1243843 (2018). <https://doi.org/10.5281/zenodo.1243843>.

[54] A. Stukowski, Visualization and analysis of atomistic simulation data with OVITO—the Open Visualization Tool, *Model. Simul. Mater. Sci. Eng.* 18 (1) (2009), <https://doi.org/10.1088/0956-0393/18/1/015012>.

[55] L. Martínez, R. Andrade, E.G. Birgin, J.M. Martínez, PACKMOL: a package for building initial configurations for molecular dynamics simulations, *J. Comput. Chem.* 30 (13) (2009) 2157–2164, <https://doi.org/10.1002/jcc.21224> URL: <http://repositorio.uta.edu.ec/bitstream/123456789/27787/1/AL674.pdf> <http://doi.wiley.com/10.1002/jcc.21224>.

[56] P.T. Merz, M.R. Shirts, Testing for physical validity in molecular simulations, *PLoS One* 13 (9) (2018) 1–21, <https://doi.org/10.1371/journal.pone.0202764> URL: <https://doi.org/10.1371/journal.pone.0202764>.

[57] E.D. Miller, M.M. Henry, M.L. Jones, E. Jankowski, Planckton (mar 2019). <https://doi.org/10.5281/zenodo.2616728>.

[58] G.A. Tribello, M. Bonomi, D. Branduardi, C. Camilloni, G. Bussi, PLUMED 2: new feathers for an old bird, *Comput. Phys. Commun.* 185 (2) (2014) 604–613, <https://doi.org/10.1016/j.cpc.2013.09.018> URL: <https://linkinghub.elsevier.com/retrieve/pii/S0010465513003196>.

[59] M.T. Humbert, Y. Zhang, E.J. Maginn, PyLAT: Python LAMMPS Analysis Tools, *J. Chem. Inf. Model. acs.jcim.9b00066* (2019), <https://doi.org/10.1021/acs.jcim.9b00066> <http://pubs.acs.org/doi/10.1021/acs.jcim.9b00066>.

[60] M.L. Jones, Rhaco – DOI: 10.5281/zenodo.1308187 (2018). <https://doi.org/10.5281/zenodo.1308187>.

[61] H. Sidky, Y.J. Colón, J. Helfferich, B.J. Sikora, C. Bezirk, W. Chu, F. Giberti, A.Z. Guo, X. Jiang, J. Lequieu, J. Li, J. Möller, M.J. Quevillon, M. Rahimi, H. Ramezani-Dakhel, V.S. Rathee, D.R. Reid, E. Sevgen, V. Thapar, M.A. Webb, J.K. Whitmer, J.J. De Pablo, SSAGES: software suite for advanced general ensemble simulations, *J. Chem. Phys.* 148 (4) (2018), <https://doi.org/10.1063/1.4993111>.

5008853.

[62] C.S. Adorf, P.M. Dodd, V. Ramasubramani, S.C. Glotzer, Simple data and workflow management with the signac framework, *Comput. Mater. Sci.* 146 (C) (2018) 220–229, <https://doi.org/10.1016/j.commatsci.2018.01.035> URL: <https://doi.org/10.1016/j.commatsci.2018.01.035>.

[63] C.S. Adorf, V. Ramasubramani, B.D. Dice, M.M. Henry, P.M. Dodd, S.C. Glotzer, glotzerlab/signac (feb 2019).<https://doi.org/10.5281/zenodo.2581327>.

[64] W. Humphrey, A. Dalke, K. Schulten, VMD: visual molecular dynamics, *J. Mol. Graph.* 14 (1) (1996) 33–38, [https://doi.org/10.1016/0263-7855\(96\)00018-5](https://doi.org/10.1016/0263-7855(96)00018-5) URL: <http://www.sciencedirect.com/science/article/pii/0263785596000185> <http://linkinghub.elsevier.com/retrieve/pii/0263785596000185>.

[65] V. Rühle, C. Junghans, A. Lukyanov, K. Kremer, D. Andrienko, Versatile object-oriented toolkit for coarse-graining applications, *J. Chem. Theory Comput.* 5 (12) (2009) 3211–3223, <https://doi.org/10.1021/ct900369w> URL: <http://pubs.acs.org/doi/abs/10.1021/ct900369w>.

[66] M.J. Foley, GitHub to give users of its free plan access to unlimited private repositories, 2019.<https://zd.net/2VafryH>.

[67] V. Käfer, D. Graziotin, I. Bogicevic, S. Wagner, J. Ramadani, Poster: communication in open-source projects—end of the e-mail era? *IEEE/ACM 40th Int. Conf. Softw. Eng. Companion* 2018 (2018) 242–243.

[68] P. Jupyter, M. Bussonnier, J. Forde, J. Freeman, B. Granger, T. Head, C. Holdgraf, K. Kelley, G. Nalvarthe, A. Osheroft, M. Pacer, Y. Panda, F. Perez, B. Ragan-Kelley, C. Willing, Binder 2.0 – Reproducible, interactive, sharable environments for science at scale, in: F. Akici, D. Lippa, D. Niederhut, M. Pacer (Eds.), *Proceedings 17th Python Science Conference*, 2018, pp. 113–120.<https://doi.org/10.25080/Majora-4af1f417-011>.

[69] A.Z. Summers, Chrisiacovella, J. Gilmer, mosdef-hub/mosdef_tutorials: MoSDeF Tutorials 0.0.0 (mar 2019).<https://doi.org/10.5281/zenodo.2617384>.

[70] G.M. Kurtzer, V. Sochat, M.W. Bauer, Singularity: scientific containers for mobility of compute, *PLoS One* 12 (5) (2017) 1–20, <https://doi.org/10.1371/journal.pone.0177459>.

[71] D. Merkel, Docker lightweight linux containers for consistent development and deployment, *Linux J.* (2014) 239,<http://dl.acm.org/citation.cfm?id=2600239.2600241>.

[72] T. Crick, B.A. Hall, S. Ishaq, Reproducibility in Research: Systems, Infrastructure, Culture, *arXiv* (2015) 1–12, arXiv:1503.02388.<http://arxiv.org/abs/1503.02388>.

[73] D.S. Katz, K.E. Niemeyer, S. Gesing, L. Hwang, W. Bangerth, S. Hettrick, R. Idaszak, J. Salac, N.C. Hong, S.N. Corrales, A. Allen, R.S. Geiger, J. Miller, E. Chen, A. Dubey, P. Lago, Report on the Fourth Workshop on Sustainable Software for Science: Practice and Experiences (WSSSPE4), *arXiv* 1 (1) (2017) 0–3, arXiv:1705.02607,<https://doi.org/10.5334/jors.184>.<http://arxiv.org/abs/1705.02607>.

[74] D. Richter, A. Baumgärtner, K. Binder, B. Ewen, J.B. Hayter, Dynamics of collective fluctuations and brownian motion in polymer melts, *Phys. Rev. Lett.* 47 (2) (1981) 109–113, <https://doi.org/10.1103/PhysRevLett.47.109>.

[75] K. Kremer, G.S. Grest, K. Kremer, I. Festkorperforschung, F. Julich, W. Germanj, I. Physik, U. Mainz, D. Mainz, W. Germany, G.S. Grest, Dynamics of entangled linear polymer melts: a moleculardynamics simulation dynamics of entangled linear polymer melts: a molecular-dynamics simulation, *J. Chem. Phys.* 5057 (2013), <https://doi.org/10.1063/1.458541>.

[76] T.E. Gartner, A. Jayaraman, Modeling and simulations of polymers: a roadmap, *Macromolecules* 52 (2019) 755–786, <https://doi.org/10.1021/acs.macromol.8b01836>.

[77] G. Voth (Ed.), *Coarse-Graining of Condensed Phase and Biomolecular Systems*, CRC Press, 2008, , <https://doi.org/10.1201/9781420059564> URL: <https://www.taylorfrancis.com/books/9781420059564>.

[78] C. Clementi, Coarse-grained models of protein folding: toy models or predictive tools? *Curr. Opin. Struct. Biol.* 18 (1) (2008) 10–15, <https://doi.org/10.1016/j.sbi.2007.10.005>.

[79] R. Elber, Long-timescale simulation methods, *Curr. Opin. Struct. Biol.* 15 (2) (2005) 151–156, <https://doi.org/10.1016/j.sbi.2005.02.004>.

[80] J.L. Klepeis, K. Lindorff-Larsen, R.O. Dror, D.E. Shaw, Long-timescale molecular dynamics simulations of protein structure and function, *Curr. Opin. Struct. Biol.* 19 (2) (2009) 120–127, <https://doi.org/10.1016/j.sbi.2009.03.004>.

[81] S.C. Kamerlin, S. Vicosas, A. Dryga, A. Warshel, Coarse-grained (multiscale) simulations in studies of biophysical and chemical systems, *Annu. Rev. Phys. Chem.* 62 (1) (2011) 41–64, <https://doi.org/10.1146/annurev-physchem-032210-103335>.

[82] S. Kmiecik, D. Gront, M. Kolinski, L. Witeska, A.E. Dawid, A. Kolinski, Coarse-grained protein models and their applications, *Chem. Rev.* 116 (14) (2016) 7898–7936, <https://doi.org/10.1021/acs.chemrev.6b00163>.

[83] S.J. Marrink, D.P. Tieleman, Perspective on the martini model, *Chem. Soc. Rev.* 42 (16) (2013) 6801–6822, <https://doi.org/10.1039/c3cs60093a> URL: <http://www.ncbi.nlm.nih.gov/pubmed/23708257>.

[84] J.D. Perlmuter, M.F. Hagan, Mechanisms of virus assembly, *Annu. Rev. Phys. Chem.* 66 (2014) 217–239, <https://doi.org/10.1146/annurev-physchem-040214-121637>.

[85] T.C. Moore, C.R. Iacobella, C. McCabe, Derivation of coarse-grained potentials via multistate iterative Boltzmann inversion, *J. Chem. Phys.* 140 (22) (2014) 224104, <https://doi.org/10.1063/1.4880555> arXiv:1410.1853,<http://aip.scitation.org/doi/10.1063/1.4880555> <http://www.ncbi.nlm.nih.gov/pubmed/24929371>.

[86] L. Lu, J.F. Dama, G.A. Voth, Fitting coarse-grained distribution functions through an iterative force-matching method, *J. Chem. Phys.* 139 (2013), <https://doi.org/10.1063/1.4811667>.

[87] L.-P. Wang, T.J. Martinez, V.S. Pande, Building force fields: an automatic, systematic, and reproducible approach, *J. Phys. Chem. Lett.* 5 (11) (2014) 1885–1891, <https://doi.org/10.1021/jz500737m> URL: <http://pubs.acs.org/doi/10.1021/jz500737m>.

[88] M.S. Shell, The relative entropy is fundamental to multiscale and inverse thermodynamic problems, *J. Chem. Phys.* 129 (14) (2008), <https://doi.org/10.1063/1.2992060>.

[89] A. Chaimovich, M.S. Shell, Relative entropy as a universal metric for multiscale errors, *Phys. Rev. E – Stat. Nonlinear, Soft Matter Phys.* 81 (6) (2010) 1–4, <https://doi.org/10.1103/PhysRevE.81.060104>.

[90] B.E. Husic, V.S. Pande, Markov state models: from an art to a science, *J. Am. Chem. Soc.* 140 (7) (2018) 2386–2396, <https://doi.org/10.1021/jacs.7b12191> URL: <http://pubs.acs.org/doi/10.1021/jacs.7b12191>.

[91] A.L. Ferguson, Machine learning and data science in soft materials engineering, *J. Phys. Condens. Matter* 30 (4) (2018) 043002, , <https://doi.org/10.1088/1361-648X/aa98bd> URL: <http://stacks.iop.org/0964-1726/26/i=10/a=105037?key=crossref.7455f3b928620d7b120b4c9d6c542f4> <http://stacks.iop.org/0953-8984/30/i=4/a=043002?key=crossref.8eb5b295cc3886eda810456a0c743d4>.

[92] N. Espinosa, M. Hösel, D. Angmo, F.C. Krebs, Solar cells with one-day energy payback for the factories of the future, *Energy Environ. Sci.* 5 (1) (2012) 5117, <https://doi.org/10.1039/c1ee02728j> URL: <http://xlink.rsc.org/?DOI=C1EE02728J>.

[93] K.A. Mazzio, C.K. Luscombe, The future of organic photovoltaics, *Chem. Soc. Rev.* 44 (1) (2015) 78–90, <https://doi.org/10.1039/C4CS00227J> URL: <http://xlink.rsc.org/?DOI=C4CS00227J>.

[94] K. Vandewal, S. Himmelberger, A. Salleo, Structural factors that affect the performance of organic bulk heterojunction solar cells, *Macromolecules* 46 (2013) 6379–6387, <https://doi.org/10.1021/ma400924b>.

[95] T.M. Clarke, J.R. Durrant, Charge photogeneration in organic solar cells, *Chem. Rev.* 110 (11) (2010) 6736–6767, <https://doi.org/10.1021/cr900271s> URL: <http://pubs.acs.org/doi/abs/10.1021/cr900271s>.

[96] T.F. Harrelson, A.J. Moulé, R. Faller, Modeling organic electronic materials: bridging length and time scales, *Mol. Simul.* 43 (10–11) (2017) 730–742, <https://doi.org/10.1080/08927022.2016.1273526>.

[97] C. Groves, Developing understanding of organic photovoltaic devices: kinetic Monte Carlo models of geminate and non-geminate recombination, charge transport and charge extraction, *Energy Environ. Sci.* 6 (11) (2013) 3202–3217, <https://doi.org/10.1039/c3ee41621f>.

[98] C. Groves, Simulating charge transport in organic semiconductors and devices: a review, *Reports Prog. Phys.* 80 (2) (2017) 026502, , <https://doi.org/10.1088/1361-6633/80/2/026502> URL: <http://stacks.iop.org/0034-4885/80/i=2/a=026502?key=crossref.44d1e9a057340b9d03e5e225c0a84eab>.

[99] J.-M.Y. Carrillo, R. Kumar, M. Goswami, B.G. Sumpter, W.M. Brown, New insights into the dynamics and morphology of P3HT:PCBM active layers in bulk heterojunctions, *Phys. Chem. Chem. Phys.* 15 (41) (2013) 17873, <https://doi.org/10.1039/c3cp53271b> URL: <http://xlink.rsc.org/?DOI=c3cp53271b> <http://pubs.rsc.org/en/Content/ArticleLanding/2013/CP/c3cp53271b> <http://www.ncbi.nlm.nih.gov/pubmed/24056906>.

[100] M. Moreno, M. Casalegno, G. Raos, S.V. Meille, R. Po, Molecular modeling of crystalline alkylthiophene oligomers and polymers, *J. Phys. Chem. B* 114 (4) (2010) 1591–1602, <https://doi.org/10.1021/jp9106124> URL: <http://www.ncbi.nlm.nih.gov/pubmed/20058895>.

[101] D.M. Huang, R. Faller, K. Do, A.J. Moulé, Coarse-grained computer simulations of polymer/fullerene bulk heterojunctions for organic photovoltaic applications, *J. Chem. Theory Comput.* 6 (2) (2010) 526–537, <https://doi.org/10.1021/ct900496t> URL: <http://pubs.acs.org/doi/abs/10.1021/ct900496t> <https://doi.org/10.1021/ct900496t>.

[102] C.-K. Lee, C.-W. Pao, C.-W. Chu, Multiscale molecular simulations of the nanoscale morphologies of P3HT:PCBM blends for bulk heterojunction organic photovoltaic cells, *Energy Environ. Sci.* 4 (10) (2011) 4124–4132, <https://doi.org/10.1039/c1ee01508g> URL: <http://xlink.rsc.org/?DOI=c1ee01508g>.

[103] R.S. Bhatta, Y.Y. Yimer, D.S. Perry, M. Tsige, Improved force field for molecular modeling of poly(3-hexylthiophene), *J. Phys. Chem. B* 117 (34) (2013) 10035–10045, <https://doi.org/10.1021/jp404629a> URL: <http://www.ncbi.nlm.nih.gov/pubmed/23899343> <http://pubs.acs.org/doi/abs/10.1021/jp404629a>.

[104] G. D’Avino, S. Mothy, L. Muccioli, C. Zannoni, L. Wang, J.J. Cornil, D. Beljonne, F. Castel, G.D. Avino, S. Mothy, L. Muccioli, C. Zannoni, L. Wang, J.J. Cornil, D. Beljonne, F. Castel, Energetics of electron-hole separation at P3HT/PCBM heterojunctions, *J. Phys. Chem. C* 117 (25) (2013) 12981–12990, <https://doi.org/10.1021/jp402957g>.

[105] O. Alexiadis, V.G. Mavrantzas, All-atom molecular dynamics simulation of temperature effects on the structural, thermodynamic, and packing properties of the pure amorphous and pure crystalline phases of regioregular P3HT, *Macromolecules* 46 (6) (2013) 2450–2467, <https://doi.org/10.1021/ma302211g> URL: <http://pubs.acs.org/doi/abs/10.1021/ma302211g>.

[106] E. Jankowski, H.S. Marsh, A. Jayaraman, Computationally linking molecular features of conjugated polymers and fullerene derivatives to bulk heterojunction morphology, *Macromolecules* 46 (14) (2013) 5775–5785, <https://doi.org/10.1021/ma400724e> URL: <http://pubs.acs.org/doi/abs/10.1021/ma400724e>.

[107] M.L. Jones, D.M. Huang, B. Chakrabarti, C. Groves, Relating molecular morphology to charge mobility in semicrystalline conjugated polymers, *J. Phys. Chem. C* 120 (8) (2016) 4240–4250.

[108] C. Scherer, D. Andrienko, Comparison of systematic coarse-graining strategies for soluble conjugated polymers, *Eur. Phys. J. Spec. Top.* 225 (8–9) (2016) 1441–1461, <https://doi.org/10.1140/epjst/e2016-60154-9>.

[109] E.D. Miller, M.L. Jones, E. Jankowski, Tying together multiscale calculations for charge transport in P3HT: structural descriptors, morphology, and tie-chains, *Polymers (Basel)* 10 (12) (2018) 1358, <https://doi.org/10.3390/polym10121358>

URL: <http://www.mdpi.com/2073-4360/10/12/1358>.

[110] C. Greco, A. Melnyk, K. Kremer, D. Andrienko, K.C. Daoulas, Generic model for lamellar self-assembly in conjugated polymers: linking mesoscopic morphology and charge transport in P3HT, *Macromolecules* 52 (2019) 968–981, <https://doi.org/10.1021/acs.macromol.8b01863>.

[111] E.D. Miller, M.L. Jones, E. Jankowski, Enhanced computational sampling of perylene and perylothiophene packing with rigid-body models, *ACS Omega* 2 (1) (2017) 353–362, <https://doi.org/10.1021/acsomega.6b00371> URL: <http://pubs.acs.org/doi/abs/10.1021/acsomega.6b00371> <http://pubs.acs.org/doi/10.1021/acsomega.6b00371>.

[112] M.M. Henry, M.L. Jones, S.D. Oosterhout, W.A. Braunecker, T.W. Kemper, R.E. Larsen, N. Kopidakis, M.F. Toney, D.C. Olson, E. Jankowski, Simplified models for accelerated structural prediction of conjugated semiconducting polymers, *J. Phys. Chem. C* 121 (47) (2017) 26528–26538, <https://doi.org/10.1021/acs.jpcc.7b09701> URL: <http://pubs.acs.org/doi/abs/10.1021/acs.jpcc.7b09701>.

[113] J.L. Chen, F.C. Chang, Phase separation process in poly(-caprolactone)-epoxy blends, *Macromolecules* 32 (2) (1999) 5348–5356, <https://doi.org/10.1021/ma981819o> URL: <http://pubs.acs.org/doi/pdf/10.1021/ma981819o>.

[114] C. Li, A. Strachan, Molecular scale simulations on thermoset polymers: a review, *J. Polym. Sci. Part B Polym. Phys.* 53 (2) (2015) 103–122, <https://doi.org/10.1002/polb.23489> URL: <http://doi.wiley.com/10.1002/polb.23489>.

[115] A.C.T. van Duin, S. Dasgupta, F. Lorant, W.A. Goddard, ReaxFF: a reactive force field for hydrocarbons, *J. Phys. Chem. A* 105 (41) (2001) 9396–9409, <https://doi.org/10.1021/jp004368u> URL: <http://pubs.acs.org/doi/abs/10.1021/jp004368u>.

[116] S. Thomas, M. Alberts, M.M. Henry, C.E. Estridge, E. Jankowski, Routine million-particle simulations of epoxy curing with dissipative particle dynamics, *J. Theor. Comput. Chem.* 17 (03) (2018) 1840005, <https://doi.org/10.1142/S0219633618400059> URL: <https://www.worldscientific.com/doi/abs/10.1142/S0219633618400059>.

[117] P.V. Komarov, C. Yu-tsung, C. Shih-ming, P.G. Khalatur, P. Reineker, Highly cross-linked epoxy resins: an atomistic molecular dynamics simulation combined with a mapping/reverse mapping procedure, *Macromolecules* 40 (22) (2007) 8104–8113, <https://doi.org/10.1021/ma070702+>.

[118] J.R. Gissinger, B.D. Jensen, K.E. Wise, Modeling chemical reactions in classical molecular dynamics simulations, *Polymer (Guildf)* 128 (2017) 211–217, <https://doi.org/10.1016/j.polymer.2017.09.038>.

[119] M. Langloeth, T. Sugii, M.C. Böhm, F. Müller-Plathe, The glass transition in cured epoxy thermosets: a comparative molecular dynamics study in coarse-grained and atomistic resolution, *J. Chem. Phys.* 143 (24) (2015) 243158, , <https://doi.org/10.1063/1.4937627> URL: <http://scitation.aip.org/content/aip/journal/jcp/143/24/1063/1.4937627>.

[120] C. Li, G.A. Medvedev, E.W. Lee, J. Kim, J.M. Caruthers, A. Strachan, Molecular dynamics simulations and experimental studies of the thermomechanical response of an epoxy thermoset polymer, *Polym. (United Kingdom)* 53 (19) (2012) 4222–4230, <https://doi.org/10.1016/j.polymer.2012.07.026>.

[121] K.S. Khare, F.R. Phelan, Quantitative comparison of atomistic simulations with experiment for a cross-linked epoxy: a specific volume-cooling rate analysis, *Macromolecules* 51 (2) (2018) 564–575, <https://doi.org/10.1021/acs.macromol.7b01303>.

[122] G. Kacar, E.A.J.F. Peters, G. De With, Multi-scale simulations for predicting material properties of a cross-linked polymer, *Comput. Mater. Sci.* 102 (2015) 68–77, <https://doi.org/10.1016/j.commatsci.2015.02.021>.

[123] H. Liu, M. Li, Z.-Y. Lu, Z.-G. Zhang, C.-C. Sun, T. Cui, Multiscale simulation study on the curing reaction and the network structure in a typical epoxy system, *Macromolecules* 44 (21) (2011) 8650–8660, <https://doi.org/10.1021/ma201390k> URL: <http://pubs.acs.org/doi/abs/10.1021/ma201390k>.

[124] S. Thomas, New Methods for Understanding and Controlling the Self-Assembly of Reacting Systems Using Coarse-Grained Molecular Dynamics, Boise State University, 2018 (Ph.D. thesis).

[125] T.A. Knotts, N. Rathore, D.C. Schwartz, J.J. de Pablo, A coarse grain model for DNA, *J. Chem. Phys.* 126 (8) (2007) 084901, , <https://doi.org/10.1063/1.2431804> URL: <http://aip.scitation.org/doi/10.1063/1.2431804>.

[126] H. Ibarra, Provisional selves: experimenting with image and identity in professional adaptation, *Adm. Sci. Q.* 44 (4) (1999) 764, <https://doi.org/10.2307/2667055> URL: <https://www.jstor.org/stable/2667055?origin=crossref>.

[127] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, Equation of state by fast computing machines, *J. Chem. Phys.* 21 (1953) 1087–1092, <https://doi.org/10.1063/1.1699114> URL: <http://link.aip.org/link/?JCP/21/1087/1>.

[128] B. Alder, T. Wainwright, Phase transition for a hard sphere system, *J. Chem. Phys.* 27 (1957) (1957) 1208, <https://doi.org/10.1063/1.1743957>.