

MIT Open Access Articles

A cloud platform for sharing and automated analysis of raw data from high throughput polymer MD simulations

The MIT Faculty has made this article openly available. **Please share** how this access benefits you. Your story matters.

Citation: Tian Xie, Ha-Kyung Kwon, Daniel Schweigert, Sheng Gong, Arthur France-Lanord, Arash Khajeh, Emily Crabb, Michael Puzon, Chris Fajardo, Will Powelson, Yang Shao-Horn, Jeffrey C. Grossman; A cloud platform for sharing and automated analysis of raw data from high throughput polymer MD simulations. APL Mach. Learn. 1 December 2023; 1 (4): 046108.

As Published: 10.1063/5.0160937

Publisher: AIP Publishing

Persistent URL: <https://hdl.handle.net/1721.1/154280>

Version: Final published version: final published article, as it appeared in a journal, conference proceedings, or other formally published context

Terms of use: Creative Commons Attribution



RESEARCH ARTICLE | NOVEMBER 17 2023

A cloud platform for sharing and automated analysis of raw data from high throughput polymer MD simulations

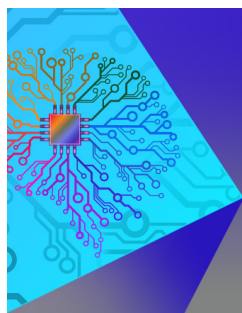
Special Collection: [2023 Papers with Best Practices in Data Sharing and Comprehensive Background Review](#)

Tian Xie   ; Ha-Kyung Kwon   ; Daniel Schweigert   ; Sheng Gong  ; Arthur France-Lanord  ; Arash Khajeh  ; Emily Crabb  ; Michael Puzon  ; Chris Fajardo  ; Will Powelson  ; Yang Shao-Horn   ; Jeffrey C. Grossman  



APL Mach. Learn. 1, 046108 (2023)

<https://doi.org/10.1063/5.0160937>



APL Machine Learning

Special Topic: Neuromorphic Technologies for Novel Hardware AI

Submit Today

A cloud platform for sharing and automated analysis of raw data from high throughput polymer MD simulations

Cite as: APL Mach. Learn. 1, 046108 (2023); doi: 10.1063/5.0160937

Submitted: 6 June 2023 • Accepted: 20 October 2023 •

Published Online: 17 November 2023















View Online



Export Citation



CrossMark

Tian Xie,^{1,a)}  Ha-Kyung Kwon,^{2,a)}  Daniel Schweigert,^{2,a)}  Sheng Gong,¹  Arthur France-Lanord^{1,3} 
Arash Khajeh,²  Emily Crabb,¹  Michael Puzon,²  Chris Fajardo²  Will Powelson,² 
Yang Shao-Horn^{1,4,a)}  and Jeffrey C. Grossman^{1,a)} 

AFFILIATIONS

¹ Department of Materials Science and Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA

² Toyota Research Institute, 4440 El Camino Real, Los Altos, California 94022, USA

³ Sorbonne Université, Institut des Sciences du Calcul et des Données, ISCD, F-75005 Paris, France

⁴ Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA

^{a)} Authors to whom correspondence should be addressed: tianxie@microsoft.com; ha-kyung.kwon@tri.global; daniel.schweigert@tri.global; shaohorn@mit.edu; and jcg@mit.edu

ABSTRACT

Open material databases storing thousands of material structures and their properties have become the cornerstone of modern computational materials science. Yet, the raw simulation outputs are generally not shared due to their huge size. In this work, we describe a cloud-based platform to enable fast post-processing of the trajectories and to facilitate sharing of the raw data. As an initial demonstration, our database includes 6286 molecular dynamics trajectories for amorphous polymer electrolytes (5.7 terabytes of data). We create a public analysis library at https://github.com/TRI-AMDD/http_md to extract ion transport properties from the raw data using expert-designed functions and machine learning models. The analysis is run automatically on the cloud, and the results are uploaded onto an open database. Our platform encourages users to contribute both new trajectory data and analysis functions via public interfaces. Finally, we create a front-end user interface at <https://www.htpmd.matr.io/> for browsing and visualization of our data. We envision the platform to be a new way of sharing raw data and new insights for the materials science community.

© 2023 Author(s). All article content, except where otherwise noted, is licensed under a Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>). <https://doi.org/10.1063/5.0160937>

I. INTRODUCTION

In the past decade, the rapid development and application of computational theory, methodology, and infrastructure for high throughput materials discovery have generated huge amounts of data in the computational materials science community.^{1–4} Open databases, such as the Materials Project,⁵ AFLOW,⁶ and Materials Cloud,⁷ store millions of material structures and computed properties, spanning inorganic crystals, metal organic frameworks, and many other types of materials. In addition, open source software, such as pymatgen,⁸ atomate,⁹ FireWorks,¹⁰ and RDKit,¹¹ have streamlined the analysis and visualization of materials data,

significantly simplifying tasks such as computing effective mass from band structures,^{12–14} calculating Li-ion conductivity from molecular dynamics (MD) trajectories,^{15–18} and rendering chemical structures.^{19–21} In the biophysics community, tools such as GPCRmd,²² BIGNASIM,²³ Cyclo-lib,²⁴ and Dymeomics²⁵ have enabled interactive analysis and visualization of MD data of proteins and small molecules.

Despite a push toward open science, a significant portion of computational materials data has not been shared publicly^{26–28}—the raw outputs from the simulations, such as the trajectories from MD simulations and the charge densities from density functional theory (DFT) calculations. The raw data can provide valuable information

about how the simulations were run and analyzed. Sharing of raw data can enable full transparency and reproducibility of the simulation data and accompanying analyzed results. Additionally, as new analysis methods are developed that more appropriately describe a physical phenomenon, these can be re-run on raw simulation data to extract new insights without re-running the simulations. However, the raw data for a single calculation can require gigabytes of storage, easily accumulating to terabytes for a high throughput screening project. Due to the high cost of data storage and transfer, most open databases only store key properties extracted from the raw data^{28,29} while leaving the raw data in the local storage of large supercomputer centers where it is often left unattended or deleted after a period of time. Very recently, the Materials Project⁵ has started to provide charge density distributions from DFT to users. However, the transfer of charge density data is not automated as users still need to communicate with the provider for access.

In this work, we provide a cloud-based platform to facilitate the sharing of raw data from high throughput materials screening. Our platform includes three components, as illustrated in Fig. 1: (1) cloud storage on Amazon Web Services (AWS) that stores raw data from simulations; (2) an open codebase on GitHub that analyzes raw data and extracts key properties; and (3) a graphical interface that allows users to interact with and visualize analyzed properties. Users can access extracted properties like in other open databases, and they can also develop new analysis functions to extract new properties from the raw data via the open codebase. Our platform eliminates the high cost of transferring terabytes of raw data by running the analysis in the cloud but still allows the user to analyze raw data based on their needs. Finally, it also aims to create a standard data format and analysis software ecosystem for MD trajectories that can eventually be expanded to include other raw outputs from other simulation methods, such as DFT. We

demonstrate the effectiveness of this platform by creating an open database of MD trajectories for amorphous polymer electrolytes generated using the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) program,^{30,31} which includes 6286 trajectories and 5.7 terabytes of data.

II. SOFTWARE INFRASTRUCTURE AND DATABASE

Our software infrastructure is hosted on Amazon Web Services (AWS)³² and utilizes its serverless cloud services for data processing, analysis, storage, and flow management, as shown in Fig. 2.

For processing, each raw trajectory data are expected to have the complete trajectory (in the “custom” LAMMPS trajectory format³³) as well as a metadata file (in json format) that describes the input parameters of the data (SMILES, temperature, molality, length of simulations, force field, and ion types). We use [Cu] and [Au] as special atoms to label the two polymerization points in the SMILES string of polymers. When new trajectory data are uploaded to the platform, they are archived in an AWS Simple Storage Service (S3) bucket, which is a scalable cloud file system. The creation of the new data files in the bucket triggers an upload event, which is picked up by a serverless compute service AWS Lambda. The Lambda instance verifies the completeness of the trajectory data to ensure that all files needed for the analysis are present. Afterward, the Lambda instance initiates a workflow execution in a AWS StepFunction graph.

As part of the StepFunction workflow, containerized AWS Elastic Container Service (ECS) tasks are run to analyze the raw trajectory data and store the results. Analyzed properties, such as ionic conductivity, diffusion coefficients of cations, anions, and polymer chains, and transference number, as well as metadata—SMILES, molality, temperature, length of simulations, force field, and ion types—are stored in an AWS Relational Database Services (RDS)

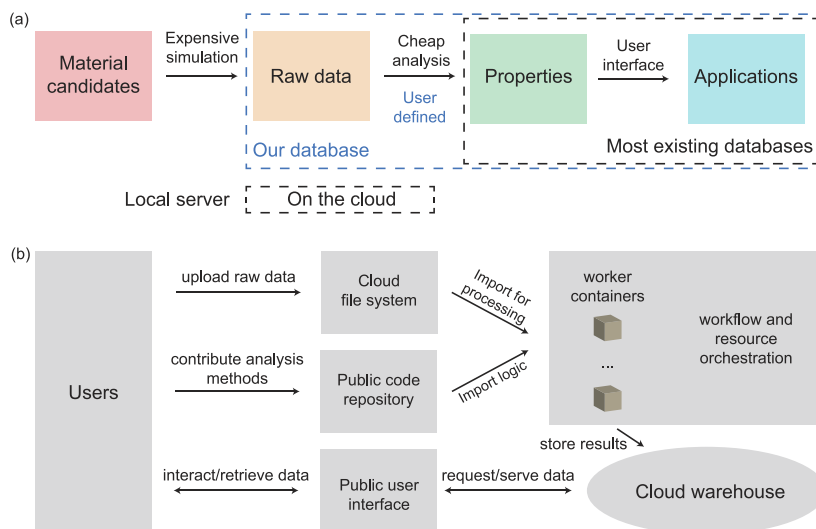


FIG. 1. Illustration of the cloud based database for raw data sharing. (a) A typical workflow for computational material data generation. Our database enables the sharing of raw simulation data and flexible user-defined analysis methods to obtain properties, while most existing databases have a fixed set of pre-computed properties. (b) An overview of the software components in the platform. The platform consists of a cloud file system containing raw data, a public code repository for analysis functions, and a user interface for interacting with the data.

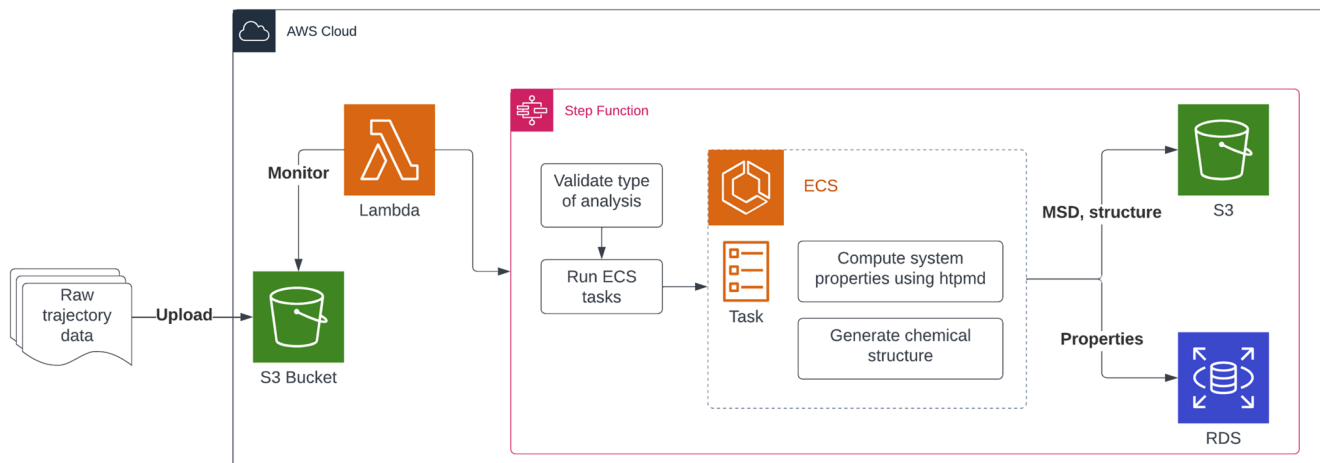


FIG. 2. Detailed software workflow showing the backend processes. Final results stored in S3 bucket and RDS are accessed by the frontend UI via an API stored in a AWS Lambda instance.

postgreSQL database. Other types of data, such as the mean squared displacement (MSD) time series for the ions, as well as the final structure file (.cif) and an image of the monomer structure (.png) are uploaded to an S3 bucket and their URLs are stored in the database.

The specific analysis steps that are run as ECS tasks are specified in the public htpmd GitHub repository (https://github.com/TRI-AMDD/htp_md). The repository contains analysis code suited for polymer electrolytes based on a LiTFSI salt and allows extracting

property results, such as Li-ion conductivity, diffusion coefficients of Li^+ , TFSI^- , and polymer chains, and transfer number. In addition, the code generates average mean squared displacement (MSD) time series for Li^+ and TFSI^- ions, as well as the final structure of the simulation box, which can also be retrieved from the UI. In addition, pre-trained machine learning models are applied to the existing trajectory data to provide predictions of a subset of the properties. More details are found in Sec. IV C. While there is a natural discrepancy between predicted and computed properties due to the variance in

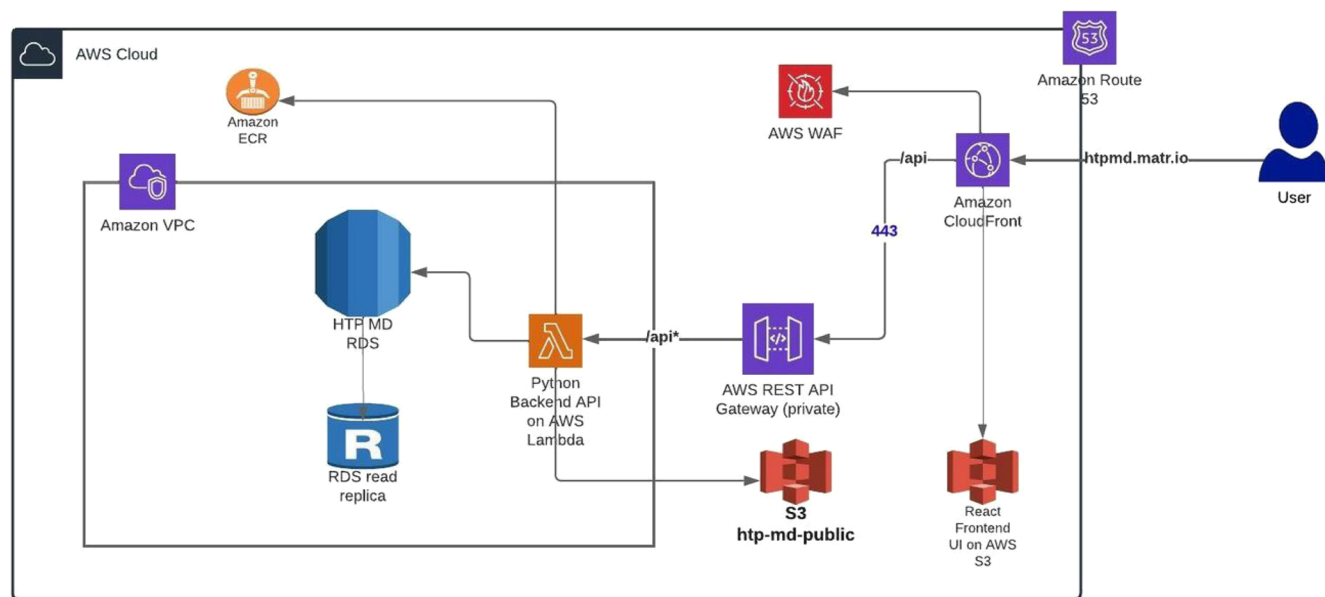


FIG. 3. Workflow for retrieving data from the backend and displaying it on the frontend.



FIG. 4. Web graphical user interface (GUI) to browse database for Simulation Data (a) and Prediction Data (b).

the machine learning models, predictions can be useful for making estimates of a property without having to run the full length of the simulation.

Members of the research community are encouraged to provide their specific analyses or prediction models to the GitHub codebase via a pull request from a fork. Upon review and merging of new code, the application is containerized using Docker and provided to ECS to be run as an automated task in the workflow.

III. FRONTEND UI

The platform provides a publicly accessible graphical user interface (GUI) that allows browsing through the available data. This web app is hosted at www.htpmd.matr.io and utilizes the React framework³⁴ for the frontend and Python for the backend. The frontend communicates with the backend using RESTful Application Programming Interface (API) calls, as described in Fig. 3. Plots are drawn using Plotly JS, which allows graphs to be zoomed in and exported to png.

On load of the application, the frontend makes an API call to fetch all trajectories available. This is used to populate the trajectories table and generate scatter plots of properties of interest. When filters are changed on the left panel, the cached data are filtered based on the user's selected filters.

A. Simulation data

Frontend UI displays data in two available tabs. The default tab, Simulation Data, displays all data related to the MD simulations of polymers, such as extracted properties and simulation input parameters for individual trajectories. Properties such as ionic conductivity, diffusion coefficients of Li^+ , TFSI⁻, and polymer chains, and the transference number are extracted using analysis functions in the htpmd github repository.

This tab shows aggregate data in two ways: (1) scatter plots of transport properties (Li-ion conductivity, Li ion diffusion coef-

ficient, TFSI ion diffusion coefficient, polymer chain diffusion coefficient, and transference number) as a function of molality, monomer molecular weight, or degree of polymerization and (2) a table overview of all trajectories (named Sample) and their properties. By default, analyzed properties of all available MD trajectories are shown in the graphs and tables. A filter on the side bar allows the user to down-select data by material group, cation/anion types, as well as temperature and molality ranges. In addition, the user can filter by the range of the property of interest.

When a trajectory is selected via table row click or a plot data point click, additional API calls are made to fetch data specific to a single trajectory. These calls fetch monomer SMILES string, chemical structure, the conditions of simulations, and the Li^+ and TFSI⁻ MSD time series. If the user clicks on the "Download Raw Data" button, a pre-signed S3 url is opened for the user to download a zip file of the trajectory's raw data.

All table and graphs have a download button that allows the user to retrieve the displayed data as comma-separated values (csv) data files.

B. Prediction data

Switching to the Prediction Data tab allows the user to view aggregate and trajectory-specific prediction data made using pre-loaded ML models, as shown in Fig. 4(b). The table shows ionic conductivity, diffusion coefficients of ions and polymers, and transference number that have been extracted from MD simulations, compared against predictions using RF and graph neural network (GNN) models (details provided in Sec. IV C).

Scattered plots show parity plots for conductivity and ion diffusion coefficients for predicted data against MD simulation data. The user can select a specific trajectory by clicking on a data point in the plots or by selecting a row in the tables. As with the Simulation Data tab, any aggregate data can be downloaded using the download button below the table or graphs.

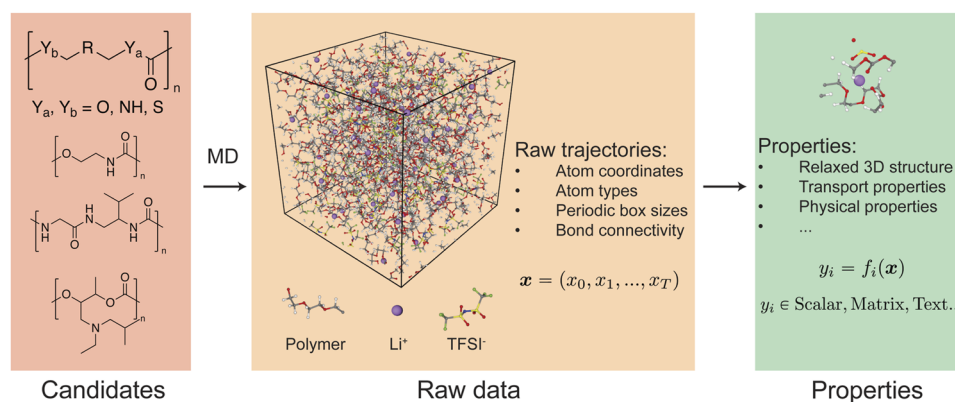


FIG. 5. Database of polymer electrolytes, containing 6057 unique polymers and their raw trajectories. Raw trajectories contain coordinates and types of atoms, as well as periodic box sizes and bond connectivity. Using the htpmd github repository, multiple properties including relaxed 3D structures, transport properties, and physical properties are automatically extracted from the raw trajectories.

TABLE I. Properties computed and associated methods.

Property	Symbol and units	Type	Method and comments
Molality	m (mol/kg)	Scalar	Number of moles of ion pairs divided by the total polymer mass
Structure		String	Written out in the CIF file format ⁴²
Atomic displacement	$\delta(\mathbf{X})$ [Å]	Scalar	Can output either the mean or the maximum displacement along the trajectory
Mean squared displacement	$\text{MSD}(t)$ (Å ²)	Vector	$\text{MSD}(t) = \langle \mathbf{X}(t) - \mathbf{X}(0) ^2 \rangle$ the average is performed on all atoms of the same species and can be switched on for time origins
Ion diffusivity	D (cm ² /s)	Scalar	$D = \frac{1}{6} \frac{d\text{MSD}(t)}{dt}$
Polymer diffusivity	D (cm ² /s)	Scalar	Defined as the average of electronegative sites (N, S, O)
Ionic conductivity	σ (S/cm)	Scalar	Nernst–Einstein or cluster Nernst–Einstein approximation ³⁶
Cation transference number	t^+	Scalar	Nernst–Einstein or cluster Nernst–Einstein approximation ³⁶
Polymerization degree	p	Scalar	Degree of polymerization
Density	ρ (g/cm ³)	Scalar	Density of the system

IV. POLYMER DATABASE AND CONTENT

A. Overview of the polymer database

As a demonstration of the platform, we upload the raw trajectories generated by a previous study³⁵ that uses molecular dynamics (MD) to screen polymer electrolytes for Li-ion battery applications. The database contains 6057 unique polymers that share the same structure template in Fig. 5, which can be synthesized through a condensation polymerization route detailed in Ref. 35. The initial 3D structures of the polymer electrolytes are generated by inserting 1.5 mol lithium bis(trifluoromethanesulfonyl)imide (LiTFSI) salt per kilogram of polymer (equivalent to 50 pairs of LiTFSI ions) into a mixture of polymer chains and performing a 5 ns MD equilibration at 353 K. Currently, the database contains 6152 MD trajectories for 5 ns simulations of polymer electrolytes at 353 K, recorded every 2 ps. This database is more comprehensive than the previous study,³⁵ in which only 900 of these polymers were simulated with MD, the rest screened with ML property predictors. In addition, the

database also contains 134 MD trajectories for 50 ns simulations of polymers, recorded every 2 ps, which provides better converged transport properties, such as diffusion coefficients and Li-ion conductivity. The force fields and simulation protocols follow previous work.³⁵

B. Properties computed and associated methods

Several properties are computed by default, mostly related to ion transport. Current methods rely on the identification of ion clusters to calculate transport properties, which are shown in Fig. 5. An extensive list is given in Table I, with the associated methods.

We also provide two ways of calculating the ionic conductivity and cation transference number: the standard Nernst–Einstein approximation and the cluster Nernst–Einstein approximation.³⁶ These methods make assumptions about the interaction strength between ions in the polymer–salt system and are appropriate for different ranges of salt concentrations. For instance, the cluster

TABLE II. Performance of different machine learning models on Li transport properties. Errors are based on the test set, and error bars are standard deviations of predictions from the four models trained in the fourfold cross-validation.

Property	RF + molecular features		GNN + molecule structures	
	MAE	R2	MAE	R2
σ (S/cm)	0.120 ± 0.003	0.532 ± 0.024	0.115 ± 0.002	0.573 ± 0.014
D_{Li^+} (cm ² /s)	0.117 ± 0.002	0.508 ± 0.012	0.115 ± 0.000	0.492 ± 0.005
D_{TFSI^-} (cm ² /s)	0.103 ± 0.002	0.633 ± 0.017	0.100 ± 0.001	0.650 ± 0.010
D_{chain} (cm ² /s)	0.088 ± 0.002	0.820 ± 0.005	0.082 ± 0.001	0.832 ± 0.002
t^+	0.158 ± 0.000	0.502 ± 0.002	0.159 ± 0.001	0.491 ± 0.004

Nernst–Einstein approximation to ionic conductivity was shown recently³⁷ to hold quantitatively for LiTFSI-tetraglyme up to intermediate concentration ($r = 0.10$ Li/EO) and qualitatively up to the highest reported concentration ($r = 0.24$ Li/EO). Eventually, we will implement solvent reference frame based methods,^{38,39} which were recently shown to be crucial in capturing the correct transference numbers.⁴⁰ This highlights the need for sharing of raw trajectory data: as our understanding of behaviors in more complex systems increases, so do our methods of extracting properties. Raw trajectory data not only make it very easy to follow the provenance of extracted properties (assumptions made, methods used, and how the simulations were performed) but also make it extremely easy to compare the applicability of different analysis methods and reanalyze as necessary.

In the future, we expect to further extend the analysis methods to calculate more properties, such as dielectric constant based on linear response theory⁴¹ and viscosity based on the Einstein–Helfand method or the Green–Kubo method.

C. ML predictions

The dataset enables investigating through machine learning the transport properties of polymer electrolytes. In this work, we use two baseline machine learning models to learn the transport properties: (1) human-engineered descriptors + random forest model and (2) graph neural networks (GNNs). Twenty percent of the data points are randomly reserved at the very beginning as the test set, and the remaining data points are used to train and tune the hyperparameters via a fourfold cross-validation. The human-engineered descriptors are generated using the package Mordred.⁴³ The random forest models are built using scikit-learn.⁴⁴ We adopt the GNN architecture used in our prior work³⁵ that builds Crystal Graph Convolutional Neural Networks (CGCNN)⁴⁵ on top of polymer graphs. Note that in this work, both models are based on 2D molecular information of the monomer; machine learning models making use of the 3D structure of polymers will be left for future work.

In Table II, we show the performance of the machine learning models on five transport properties: Li-ion conductivity, Li^+ , TFSI⁻, polymer chain diffusion coefficients, and transference number. We can see that the deep representation learning model (GNN) performs slightly better than the random forest model based on human-engineered descriptors for all transport properties, except for the transference number. In addition, except for the polymer chain diffusion coefficient, the R^2 scores of prediction of transport properties from both machine learning models are lower than 0.8, which indicates the limitations of current two models. Note that the machine learning prediction of ion transport properties of solid polymer electrolytes is not the main focus of this study and is merely included to demonstrate the usefulness of our database. More information on improving the prediction performance of machine learning models using features extracted from 3D structures and dynamics of the systems has been discussed in detail somewhere else.⁴⁶

V. USER SCENARIO

We envision two broad user scenarios for the platform: (1) a user whose primary goal is to explore and visualize existing data (raw

data and analyzed properties) and (2) a user who wishes to develop and contribute new analysis functions and ML prediction models to derive additional insights from existing data or use existing analysis functions and ML prediction models on private data and potentially contribute new data to our platform. We outline recommended workflows for each user scenario.

A. Visualization and exploration of data

A user wishing to explore the data can access the platform at <https://www.htpmd.matr.io/>. All trajectory data are loaded at once; however, the user can down-select the data using the selection panel on the left, filtering by components, simulation, material conditions (molality, monomer molecular weight, degree of polymerization, force field, time step, temperature, and simulation length), and the desired range of analyzed properties. Most trajectories also have additional data (chemical structure and mean squared displacement time series) and can be filtered by whether the data are available.

Filtered data are displayed in tabular and graphical formats. The data table lists simulation conditions and analyzed properties for each trajectory ID and can be sorted by ascending/descending value. Aggregate view shows one plot with selectable x- and y-axes, where molality, monomer molecular weight, or degree of polymerization can be plotted on the x-axis and Li-ion conductivity and diffusion coefficients of Li^+ and TFSI⁻ and polymer chains and transference number can be selected as the y-axis (Fig. 6). Users can hover over each data point for trajectory ID information or zoom into parts of each plot using click-and-drag.

More detailed information on a single trajectory can be displayed by selecting a specific trajectory (in the table or on the graph), as shown in Fig. 4. The sample view displays MSD time series, chemical structure, and simulation conditions for the selected trajectory.

Trajectory-specific data in sample view or aggregate information in aggregate view can be downloaded by clicking on the Download button. This information is downloaded as a csv file.

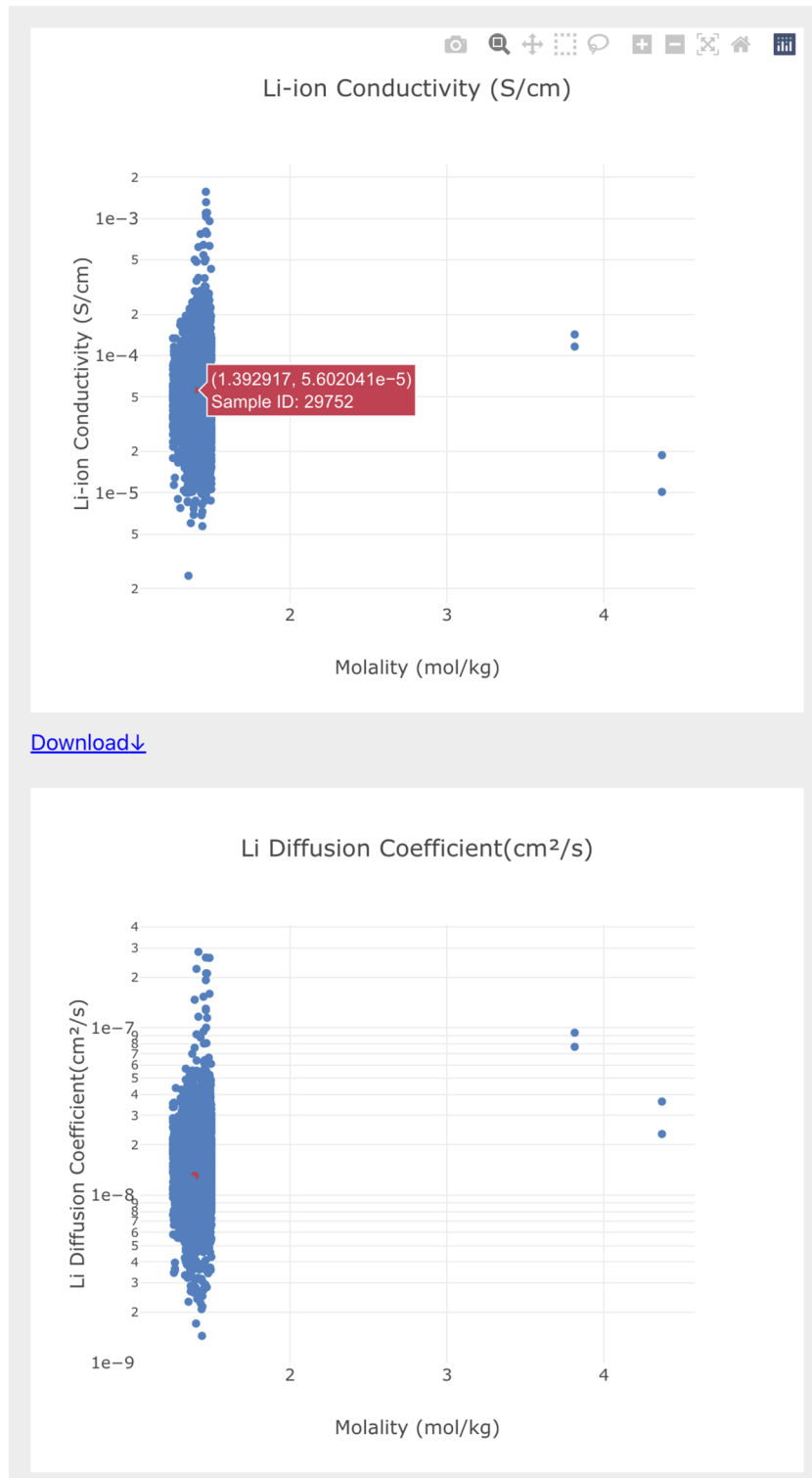
B. Community contribution of new analysis methods and data

Some users may wish to run the analysis module locally in order to develop new analysis functions, train new ML prediction models, or try existing analysis functions on private data. This can enable new insights from existing data. For these users, the best starting point is at the public github repository at https://github.com/TRI-AMDD/htp_md.

The repository provides test data for three test systems (an aqueous NaCl electrolyte and two LiTFSI polymer electrolytes). Additional data can be downloaded from the database via the UI. Newly developed analysis functions can be tested on the test data, as well as any data downloaded from the database [see Fig. 7(a)].

In order to merge the new analysis function or method, the user must open a pull request containing the source code in `function.py` and an accompanying test in `function_test.py`, as well as test data and results (if different from provided test data). The format of the code should follow the provided template. Contributed code will be reviewed by the repository maintenance team.

Once contributed code is reviewed and merged, htpmd version number will be updated, and the pipeline will run the latest version



[Download](#) ↓

FIG. 6. Aggregate data plots showing Li-ion conductivity and Li^+ diffusion coefficient of all available trajectory data. Numeric values and trajectory ID can be shown when hovering on a specific data point. Clicking on a data point highlights all corresponding points in subsequent graphs, as well as on the table. This also enables trajectory sample view, which gives specific information about the polymer, simulation conditions, and property values.

25 April 2024 13:34:03

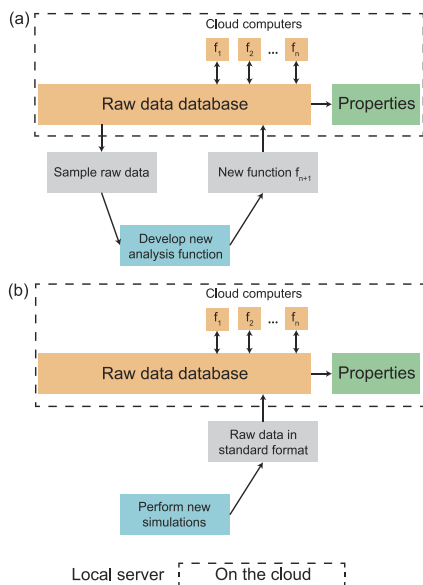


FIG. 7. Schematics for contributing new methods and data. (a) Analysis method contribution. User can download sample raw data from the database and develop new analysis function locally. New functions will be scaled to the entire database after merging. (b) Data contribution. User can contribute new trajectories to the database by following the standard format. All existing analysis functions will be used to analyze the new data.

of analysis functions on existing data. Latest versions of extracted properties will be available on the webapp.

Alternatively, some users may wish to contribute new data that were locally generated [Fig. 7(b)]. Users are not required but are encouraged to provide information on the compiler and version of LAMMPS used.

VI. DISCUSSION

The htpmd database enables researchers to harvest insights from molecular dynamics simulations of thousands of polymer-salt systems. We will be adding trajectories and property data as we simulate more polymer-salt systems to identify their respective ion transport properties. We encourage researchers in the community to make use of the presented data, methods, and models for their own investigations and for use as benchmarks. We also welcome any contributions to this database. If you would like to add your data, methods, or machine learning models to this platform, contact us for details. In future updates of the database web portal, we envision adding functionality that will enable direct upload and automatic verification of new data from MD simulations, experiments, and literature.

In order to make material simulations a meaningful tool in the pursuit of accelerated materials discovery, it is necessary to establish the validity and accuracy of material property data resulting from such simulations. Previous work examined the alignment of MD simulation results with experimentally determined transport properties, such as Li^+ conductivity.^{47–49} To further complete the picture, we envision future additional features of the platform, which

allow a direct comparison of simulation data to experimental results published in the literature. This will enable researchers to further explore the conditions for validity, possible limitations, and future improvements to the simulation methodologies.

A separate effort that is currently under way is the development of high throughput experimental screening methods for measuring ionic conductivity in solid polymer systems. One of its uses will be to validate computational results via synthesis and characterization of previously simulated systems. The experimental data generated through the screening can be further incorporated into our database.

To close, we emphasize that our suite of analysis functions can make the data easily shareable, even those that already exist in literature. For example, if all published studies of MD simulations of polymer-salt electrolytes (from January–March of 2023) shared raw trajectory data, it would enable open sharing of roughly 130 GB of additional data. If every study from the last decade shared their raw trajectory data, it would double our database, increasing it to roughly 11 terabytes.

While the current available analysis functions and the front-end UI are specific to the transport properties in polymer-salt systems, the cloud-based platform and its infrastructure can be easily extended into other types of simulation approaches and material properties. The backend workflow of our platform only requires a raw data format and a list of analysis functions designed for the format. This means that with the addition of specific analysis functions and raw trajectory format, we could expand our platform to other use cases, such as the extraction of rheological and mechanical properties from MD simulations of large polymer systems at different strain rates.

ACKNOWLEDGMENTS

This work was supported by the Toyota Research Institute. Computational support was provided by the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231, and the Extreme Science and Engineering Discovery Environment, supported by National Science Foundation Grant No. ACI-1053575.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Tian Xie, Ha-Kyung Kwon, and Daniel Schweigert contributed equally to this work.

T.X., H.-K.K., D.S., J.C.G., and Y.S.-H. conceived the idea. H.-K.K., D.S., and T.X. led the development of the platform. T.X., S.G., A.F.-L., and E.C. generated the data and developed the analysis functions. H.-K.K., D.S., A.K., M.P., C.F., and W.P. developed the software infrastructure on AWS, including both frontend and backend. All authors (T.X., H.-K.K., D.S., S.G., A.F.-L., A.K., E.C., M.P.,

C.F., W.P., Y.S.-H., and J.C.G.) contributed to the writing of the paper.

Tian Xie: Conceptualization (equal); Formal analysis (equal); Methodology (equal); Software (equal); Writing – original draft (equal); Writing – review & editing (equal). **Ha-Kyung Kwon:** Conceptualization (equal); Methodology (equal); Software (equal); Visualization (equal); Writing – original draft (equal); Writing – review & editing (equal). **Daniel Schweigert:** Conceptualization (equal); Methodology (equal); Software (equal); Visualization (equal); Writing – original draft (equal); Writing – review & editing (equal). **Sheng Gong:** Formal analysis (equal); Writing – original draft (equal); Writing – review & editing (equal). **Arthur France-Lanord:** Formal analysis (equal); Writing – original draft (equal); Writing – review & editing (equal). **Arash Khajeh:** Software (equal); Visualization (equal); Writing – original draft (equal); Writing – review & editing (equal). **Emily Crabb:** Formal analysis (equal); Writing – original draft (equal); Writing – review & editing (equal). **Michael Puzon:** Software (equal); Visualization (equal); Writing – original draft (equal); Writing – review & editing (equal). **Chris Fajardo:** Software (equal); Visualization (equal); Writing – original draft (equal); Writing – review & editing (equal). **Will Powelson:** Software (equal); Writing – original draft (equal); Writing – review & editing (equal). **Yang Shao-Horn:** Conceptualization (equal); Writing – original draft (equal); Writing – review & editing (equal). **Jeffrey C. Grossman:** Conceptualization (equal); Writing – original draft (equal); Writing – review & editing (equal).

DATA AVAILABILITY

All data are available at <https://www.htpmd.matr.io>⁵⁰ and code is available at https://github.com/TRI-AMDD/htp_md.⁵¹

REFERENCES

- C. Suh, C. Fare, J. A. Warren, and E. O. Pyzer-Knapp, “Evolving the materials genome: How machine learning is fueling the next generation of materials discovery,” *Annu. Rev. Mater. Res.* **50**, 1 (2020).
- Y. Liu, T. Zhao, W. Ju, and S. Shi, “Materials discovery and design using machine learning,” *J. Mater. Res.* **3**, 159 (2017), part of the Special Issue: High-Throughput Experimental and Modeling Research Toward Advanced Batteries.
- R. Pollice, G. dos Passos Gomes, M. Aldeghi, R. J. Hickman, M. Krenn, C. Lavigne, M. Lindner-D’Addario, A. Nigam, C. T. Ser, Z. Yao, and A. Aspuru-Guzik, “Data-driven strategies for accelerated materials design,” *Acc. Chem. Res.* **54**, 849 (2021).
- A. Nandy, C. Duan, and H. J. Kulik, “Audacity of huge: Overcoming challenges of data scarcity and data quality for machine learning in computational materials discovery,” *Curr. Opin. Chem. Eng.* **36**, 100778 (2022).
- A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, and K. A. Persson, “Commentary: The materials project: A materials genome approach to accelerating materials innovation,” *APL Mater.* **1**, 011002 (2013).
- S. Curtarolo, W. Setyawan, G. L. Hart, M. Jahnatek, R. V. Chepulskii, R. H. Taylor, S. Wang, J. Xue, K. Yang, O. Levy, M. J. Mehl, H. T. Stokes, D. O. Demchenko, and D. Morgan, “AFLOW: An automatic framework for high-throughput materials discovery,” *Comput. Mater. Sci.* **58**, 218 (2012).
- L. Talriz, S. Kumbhar, E. Passaro, A. V. Yakutovich, V. Granata, F. Gargiulo, M. Borelli, M. Uhrin, S. P. Huber, S. Zoupanos *et al.*, “Materials cloud, a platform for open computational science,” *Sci. Data* **7**, 299 (2020).
- S. P. Ong, W. D. Richards, A. Jain, G. Hautier, M. Kocher, S. Cholia, D. Gunter, V. L. Chevrier, K. A. Persson, and G. Ceder, “Python materials genomics (pymatgen): A robust, open-source python library for materials analysis,” *Comput. Mater. Sci.* **68**, 314 (2013).
- K. Mathew, J. H. Montoya, A. Faghaninia, S. Dwarakanath, M. Aykol, H. Tang, I. h. Chu, T. Smidt, B. Bocklund, M. Horton, J. Dagdelen, B. Wood, Z.-K. Liu, J. Neaton, S. P. Ong, K. Persson, and A. Jain, “Atomate: A high-level interface to generate, execute, and analyze computational materials science workflows,” *Comput. Mater. Sci.* **139**, 140 (2017).
- A. Jain, S. P. Ong, W. Chen, B. Medasani, X. Qu, M. Kocher, M. Brafman, G. Petretto, G.-M. Rignanese, G. Hautier, D. Gunter, and K. A. Persson, “Fireworks: A dynamic workflow system designed for high-throughput applications,” *Concurrency Comput.: Pract. Exper.* **27**, 5037 (2015).
- G. Landrum, Rdkit: Open-Source Cheminformatics, 2013, <http://www.rdkit.org>.
- R. Woods-Robinson, D. Broberg, A. Faghaninia, A. Jain, S. S. Dwarakanath, and K. A. Persson, “Assessing high-throughput descriptors for prediction of transparent conductors,” *Chem. Mater.* **30**, 8375 (2018).
- G. Hautier, A. Miglio, G. Ceder, G.-M. Rignanese, and X. Gonze, “Identification and design principles of low hole effective mass *p*-type transparent conducting oxides,” *Nat. Commun.* **4**, 2292 (2013).
- V.-A. Ha, F. Ricci, G.-M. Rignanese, and G. Hautier, “Structural design principles for low hole effective mass *s*-orbital-based *p*-type oxides,” *J. Mater. Chem. C* **5**, 5772 (2017).
- E. Sivonxay, M. Aykol, and K. A. Persson, “The lithiation process and Li diffusion in amorphous SiO₂ and Si from first-principles,” *Electrochim. Acta* **331**, 135344 (2020).
- J. Cheng, E. Sivonxay, and K. A. Persson, “Evaluation of amorphous oxide coatings for high-voltage Li-ion battery applications using a first-principles framework,” *ACS Appl. Mater. Interfaces* **12**, 35748 (2020).
- J. Qi, S. Banerjee, Y. Zuo, C. Chen, Z. Zhu, M. Holekevi Chandrappa, X. Li, and S. Ong, “Bridging the gap between simulated and experimental ionic conductivities in lithium superionic conductors,” *Mater. Today Phys.* **21**, 100463 (2021).
- B. Zhang, Z. He, J. Zhong, L. Yang, Z. Lin, and F. Pan, “Balancing stability and Li-ion conductivity of Li₁₀SiP₂O₁₂ for solid-state electrolytes with the assistance of a body-centered cubic oxygen framework,” *J. Mater. Chem. A* **9**, 22952 (2021).
- D. Flam-Shepherd, T. C. Wu, and A. Aspuru-Guzik, “MPGVAE: Improved generation of small organic molecules using message passing neural nets,” *Mach. Learn.: Sci. Technol.* **2**, 045010 (2021).
- J. Jiménez-Luna, A. Cuzzolin, G. Bolcato, M. Sturlese, and S. Moro, “A deep-learning approach toward rational molecular docking protocol selection,” *Molecules* **25**, 2487 (2020).
- R. J. Hall, C. W. Murray, and M. L. Verdonk, “The fragment network: A chemistry recommendation engine built using a graph database,” *J. Med. Chem.* **60**, 6440 (2017).
- I. Rodríguez-Espigares, M. Torrens-Fontanals, J. K. Tiemann, D. Aranda-García, J. M. Ramírez-Anguita, T. M. Stepniewski, N. Worp, A. Varela-Rial, A. Morales-Pastor, B. Medel-Lacruz *et al.*, “GPCRmd uncovers the dynamics of the 3D-GPCRome,” *Nat. Methods* **17**, 777 (2020).
- A. Hospital, P. Andrio, C. Cugnasco, L. Codo, Y. Becerra, P. D. Dans, F. Battistini, J. Torres, R. Goñi, M. Orozco, and J. L. Gelpi, “BIGNASim: A NoSQL database structure and analysis portal for nucleic acids simulation data,” *Nucleic Acids Res.* **44**, D272 (2015).
- E. Mixcoha, R. Rosende, R. García-Fandino, and Á. Piñeiro, “Cyclo-lib: A database of computational molecular dynamics simulations of cyclodextrins,” *Bioinformatics* **32**, 3371 (2016).
- M. W. van der Kamp, R. D. Schaeffer, A. L. Jonsson, A. D. Scouras, A. M. Simms, R. D. Toofanny, N. C. Benson, P. C. Anderson, E. D. Merkle, S. Rysavy, D. Bromley, D. A. Beck, and V. Daggett, “Dynamomics: A comprehensive database of protein dynamics,” *Structure* **18**, 423 (2010).
- F.-X. Coudert, “Materials databases: The need for open, interoperable databases with standardized data and rich metadata,” *Adv. Theory Simul.* **2**, 1900131 (2019).

- ²⁷C. H. Ward, J. A. Warren, and R. J. Hanisch, "Making materials science and engineering data more valuable research products," *Integr. Mater. Manuf. Innovation* **3**, 292 (2014).
- ²⁸L. Himanen, A. Geurts, A. S. Foster, and P. Rinke, "Data-driven materials science: Status, challenges, and perspectives," *Adv. Sci.* **6**, 1900808 (2019).
- ²⁹S. R. Kalidindi and M. De Graef, "Materials data science: Current status and future outlook," *Annu. Rev. Mater. Res.* **45**, 171 (2015).
- ³⁰S. Plimpton, "Fast parallel algorithms for short-range molecular dynamics," *J. Comput. Phys.* **117**, 1 (1995).
- ³¹A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. in't Veld, A. Kohlmeyer, S. G. Moore, T. D. Nguyen *et al.*, "LAMMPS—A flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales," *Comput. Phys. Commun.* **271**, 108171 (2022).
- ³²S. Mathew and J. Varia, Overview of amazon web services, Amazon Whitepapers, 2014.
- ³³See <https://docs.lammps.org/dump.html> for LAMMPS Manual: Dump Command.
- ³⁴J. Walke, React—A javascript library for building user interfaces, 2022, <https://reactjs.org/>.
- ³⁵T. Xie, A. France-Lanord, Y. Wang, J. Lopez, M. Stolberg, M. Hill, G. M. Lev-erick, R. Gomez-Bombarelli, J. A. Johnson, Y. Shao-Horn, and J. C. Grossman, "Accelerating amorphous polymer electrolyte screening by learning to reduce errors in molecular dynamics simulated properties," *Nat. Commun.* **13**, 3415 (2022).
- ³⁶A. France-Lanord and J. C. Grossman, "Correlations from ion pairing and the Nernst-Einstein equation," *Phys. Rev. Lett.* **122**, 136001 (2019).
- ³⁷C. Fang, A. Mistry, V. Srinivasan, N. P. Balsara, and R. Wang, "Elucidating the molecular origins of the transference number in battery electrolytes using computer simulations," *JACS Au* **3**, 306 (2023).
- ³⁸D. R. Wheeler and J. Newman, "Molecular dynamics simulations of multi-component diffusion. 1. Equilibrium method," *J. Phys. Chem. B* **108**, 18353 (2004).
- ³⁹K. D. Fong, H. K. Bergstrom, B. D. McCloskey, and K. K. Mandadapu, "Transport phenomena in electrolyte solutions: Nonequilibrium thermodynamics and statistical mechanics," *AIChE J.* **66**, e17091 (2020).
- ⁴⁰Y. Shao, H. Gudla, D. Brandell, and C. Zhang, "Transference number in polymer electrolytes: Mind the reference-frame gap," *J. Am. Chem. Soc.* **144**, 7583 (2022).
- ⁴¹I. Leontyev and A. Stuchebrukhov, "Accounting for electronic polarization in non-polarizable force fields," *Phys. Chem. Chem. Phys.* **13**, 2613 (2011).
- ⁴²I. D. Brown and B. McMahon, "CIF: The computer language of crystallography," *Acta Crystallogr., Sect. B* **58**, 317 (2002).
- ⁴³H. Moriwaki, Y.-S. Tian, N. Kawashita, and T. Takagi, "Mordred: A molecular descriptor calculator," *J. Cheminf.* **10**, 4 (2018).
- ⁴⁴F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.* **12**, 2825 (2011).
- ⁴⁵T. Xie and J. C. Grossman, "Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties," *Phys. Rev. Lett.* **120**, 145301 (2018).
- ⁴⁶A. Khajeh, D. Schweigert, S. Torrisi, L. Hung, B. Storey, and H.-K. Kwon, "Early prediction of ion transport properties in solid polymer electrolytes using machine learning and system behavior-based descriptors of molecular dynamics simulations," *Macromolecules* **56**, 4787 (2022).
- ⁴⁷Z. Li, G. D. Smith, and D. Bedrov, "Li⁺ solvation and transport properties in ionic liquid/lithium salt mixtures: A molecular dynamics simulation study," *J. Phys. Chem. B* **116**, 12801 (2012).
- ⁴⁸P. Kubisiak and A. Eilmes, "Molecular dynamics simulations of ionic liquid based electrolytes for Na-ion batteries: Effects of force field," *J. Phys. Chem. B* **121**, 9957 (2017).
- ⁴⁹H. Gudla, C. Zhang, and D. Brandell, "Effects of solvent polarity on Li-ion diffusion in polymer electrolytes: An all-atom molecular dynamics study with charge scaling," *J. Phys. Chem. B* **124**, 8124 (2020).
- ⁵⁰See <https://www.htpmd.matr.io> for htpmd web app.
- ⁵¹See https://github.com/tri-amdd/htp_md for htpmd source code.