

AA2UA

Converting all-atom models into their united atom coarse grained counterparts for use in LAMMPS

Assaf, Eli I.; Liu, Xueyan; Erkens, Sandra

DOI

[10.1016/j.simpa.2024.100686](https://doi.org/10.1016/j.simpa.2024.100686)

Publication date

2024

Document Version

Final published version

Published in

Software Impacts

Citation (APA)

Assaf, E. I., Liu, X., & Erkens, S. (2024). AA2UA: Converting all-atom models into their united atom coarse grained counterparts for use in LAMMPS. *Software Impacts*, 21, Article 100686. <https://doi.org/10.1016/j.simpa.2024.100686>

Important note

To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.



Original software publication

AA2UA: Converting all-atom models into their united atom coarse grained counterparts for use in LAMMPS

Eli I. Assaf^{a,*}, Xueyan Liu^a, Sandra Erkens^{a,b}^a Delft University of Technology, Delft, The Netherlands^b Ministry of Infrastructure and Water Management (Rijkswaterstaat), The Netherlands

ARTICLE INFO

Keywords:

Molecular Dynamics
 United Atom
 Coarse-grained
 LAMMPS

ABSTRACT

Atomistic simulations are crucial for understanding material properties at the molecular level but are limited by high computational costs, especially for large, complex systems like bituminous materials. Our team developed a Force-matched United Atom (UA) Coarse Graining (CG) force field to enhance computational efficiency while retaining atomic detail. However, converting all-atom models to CG models is complex, requiring detailed atom-to-bead mapping and compatibility with molecular dynamics (MD) engines like LAMMPS. To address this, we introduce AA2UA, an open-source software that simplifies the conversion of PDB files into LAMMPS-readable structure topology files, facilitating broader use of the developed UA force field.

Code metadata

Current code version

Permanent link to code/repository used for this code version

Permanent link to reproducible capsule

Legal code license

Code versioning system used

Software code languages, tools and services used

Compilation requirements, operating environments and dependencies

If available, link to developer documentation/manual

Support email for questions

1.0.0

<https://github.com/SoftwareImpacts/SIMPAC-2024-136><https://codeocean.com/capsule/3653460/tree/v1>

GNU General Public License (GPL)

None

Python 3.12

Python 3.7+, Rdkit, Numpy

<https://github.com/eli-ams/AA2UA>e.i.assaf@tudelft.nl

1. Introduction

Atomistic simulations are integral in elucidating the intricate properties of materials at the molecular level, pivotal for the design and development of novel materials. Despite their importance, these simulations are often hindered by high computational costs, limiting their applicability to relatively small systems, typically within the nanoscale domain [1,2]. This limitation is particularly evident in the study of bituminous materials, which exhibit complex intermolecular interactions and structural morphologies extending into the microscale [3,4]. To adequately capture the fundamental mechanisms governing the mechanical and rheological behaviors of bitumen, it is necessary to perform analyses across a wider range of scales [5].

Coarse Graining (CG) techniques offer a promising solution by simplifying atomistic models, representing groups of atoms with simplified particles or “beads” [6]. This approach significantly enhances computational efficiency, enabling the exploration of larger systems and longer time scales [7]. However, the available CG methods are not specifically tailored to capture the unique behaviors of bituminous materials, which pose distinct challenges due to their complex chemical compositions and interactions. In response to these challenges, a Force-matched United Atom (UA) CG force field has been developed by our team to provide a balanced approach, grouping hydrogen atoms with their respective parent atoms, thus preserving a considerable level of atomic detail while enhancing computational performance a hundred-fold [8].

While the force field simplification rules and input parameters are available in the literature, using and implementing them in molecular

The code (and data) in this article has been certified as Reproducible by Code Ocean: (<https://codeocean.com/>). More information on the Reproducibility Badge Initiative is available at <https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals>.

* Corresponding author.


















E-mail addresses: e.i.assaf@tudelft.nl (E.I. Assaf), x.liu@tudelft.nl (X. Liu), s.m.j.g.erkens@tudelft.nl (S. Erkens).

<https://doi.org/10.1016/j.simpa.2024.100686>

Received 24 June 2024; Received in revised form 9 July 2024; Accepted 9 July 2024

Table 1

Detailed description of all the 17 bead types in the UA the force field, identified from A through Q. These are used by AA2UA to identify and assign different bead types in the PDB file.

Type	Color	Atom label	Atomic Formula	Mass	Description
A		C-H1-SP2-2-True-True	CH	13.019	Aromatic tertiary carbon
B		O-H0-SP2-1-False-False	O	15.999	Sulfoxide oxygen
C		C-H1-SP3-3-True-False	CH	13.019	Cycloalkane tertiary carbon
D		C-H2-SP3-2-True-False	CH ₂	14.027	Cycloalkane secondary carbon
E		N-H0-SP2-2-True-True	N	14.007	Pyridinic nitrogen
F		S-H0-SP2-2-True-True	S	32.065	Thiophenic sulfur
G		O-H0-SP2-2-True-False	O	15.999	Oxane oxygen
H		O-H1-SP2-1-False-False	OH	17.007	Phenolic oxygen
I		S-H0-SP3-3-True-False	S	32.065	Sulfoxide sulfur
J		C-H1-SP3-3-False-False	CH	13.019	Tertiary carbon
K		C-H3-SP3-1-False-False	CH ₃	15.035	Primary carbon
L		C-H2-SP3-2-False-False	CH ₂	14.027	Secondary carbon
M		N-H1-SP2-2-True-True	NH	15.015	Amine nitrogen
N		C-H0-SP3-4-True-False	C	12.011	Cuaternary carbon
O		C-H0-SP2-3-True-False	C	12.011	Thiophenic quaternary carbon
P		C-H1-SP2-2-True-False	CH	13.019	Thiophenic tertiary carbon
Q		C-H0-SP2-3-True-True	C	12.011	Aromatic quaternary carbon

models with LAMMPS requires a specific set of steps that very few, if any, software packages are adept at performing comprehensively. The use of the force field, especially when it is unconventional (CG instead of AA), by other scientists is heavily limited due to three main reasons: (1) the steps required to map all-atom models into CG bead models by grouping atoms according to predefined mapping rules [8]; (2) the steps needed to assign appropriate force field parameters, including atom types, masses, and charges, to each particle/bead in the system [9,10]; and (3) ensuring that this conversion is performed in a way compatible with popular MD engines, such as LAMMPS, so that the transition from already built all-atom models to CG models is almost unnoticeable by the MD engine. These steps could be performed manually, but doing so would be incredibly time-consuming, effectively restricting the use of the developed UA force field by other researchers almost entirely. Existing tools, such as the PyCGTool [11], Insane [12], msi2lmp [13], OpenMSCG [14], CGBuilder [15], and PackMol [16] offer partial solutions to the limitations, requiring that users integrate these tools meticulously to address all three challenges comprehensively.

This manuscript focuses on introducing an open-source, easy-to-use software package, AA2UA (*All-Atom to United Atom*), designed to facilitate the conversion of all-atom molecular models into the Coarse-Grained United Atom counterparts defined in the force field developed by our team, ready for use in LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator [17]). The tool efficiently converts Protein Data Bank (PDB [18]) files, widely used in the molecular modeling literature to describe atomistic systems, into LAMMPS-readable [structure topology files](#). These files include bead positions, bonding information, masses, and all the force field-related input parameters necessary for performing conventional LAMMPS MD runs, similar to those used in all-atom simulations.

This capability significantly enhances the usability of the developed UA force field across various molecular systems, including those already developed, enabling researchers to explore material properties on larger spatial and temporal scales than previously possible. The software ensures that the transition from all-atom to CG models is seamless for LAMMPS, thereby facilitating broader adoption and application of the UA force field developed by our team, which might otherwise remain underutilized due to the significant barriers to its application. The manuscript begins by providing a comprehensive overview of the AA2UA script, detailing its design and functionality. It proceeds with an in-depth description of the code architecture, emphasizing the flow of variables, classes, and function calls during execution to give a clear picture of its operational dynamics. The manuscript concludes by evaluating AA2UA's impact on current research projects and by identifying limitations and potential improvements for future work.

2. Software description

The program is developed in Python 3.12 and requires only one external dependency, RDKit [19], which handles chemistry-related operations. The software processes *.PDB files located in the input/directory. For each PDB file, the program performs the following steps:

- 1. Extract Box Dimensions:** The program reads the box dimensions from the file's header and iterates over all atoms in the file.
- 2. Identify Molecular Blocks:** It identifies and separates individual molecular blocks, effectively breaking down the system into its constituent molecules.
- 3. Determine Bonding Information:** Using XYZ2MOL [20], the program determines higher-order bonding information, such as aromatic bonds.
- 4. Identify Atom Types:** It iterates through the non-hydrogen atoms, identifying their types by generating labels that contain detailed chemical information. The format for these labels is:

```
$atom_type-H$N_hydrogens-$hybridization-$degree-$is_in_ring-$is_aromatic
```

where:

- **atom_type:** The chemical symbol of the atom.
- **n_hydrogens:** The number of hydrogen atoms bonded to the atom.
- **hybridization:** The hybridization state of the atom.
- **degree:** The number of directly bonded neighbors.
- **is_in_ring:** A boolean indicating if the atom is part of a ring.
- **is_aromatic:** A boolean indicating if the atom is in an aromatic ring.

This format ensures proper discretization of atoms based on the differentiation rules established by the developed UA force field.

- 5. Assign Force Field Types and Masses:** The program identifies whether the generated atom labels exist in the developed force field and assigns them the corresponding force field type and mass. The force field's atom types, masses, and corresponding labels can be found in [Table 1](#).
- 6. Remove Hydrogen Atoms:** It physically removes all hydrogen atoms from the molecules, leaving only their parent atoms.
- 7. Generate Interaction List:** The program generates a list of pairwise, bonding, and angular interactions among all possible bead types present in the system, corresponding to the number

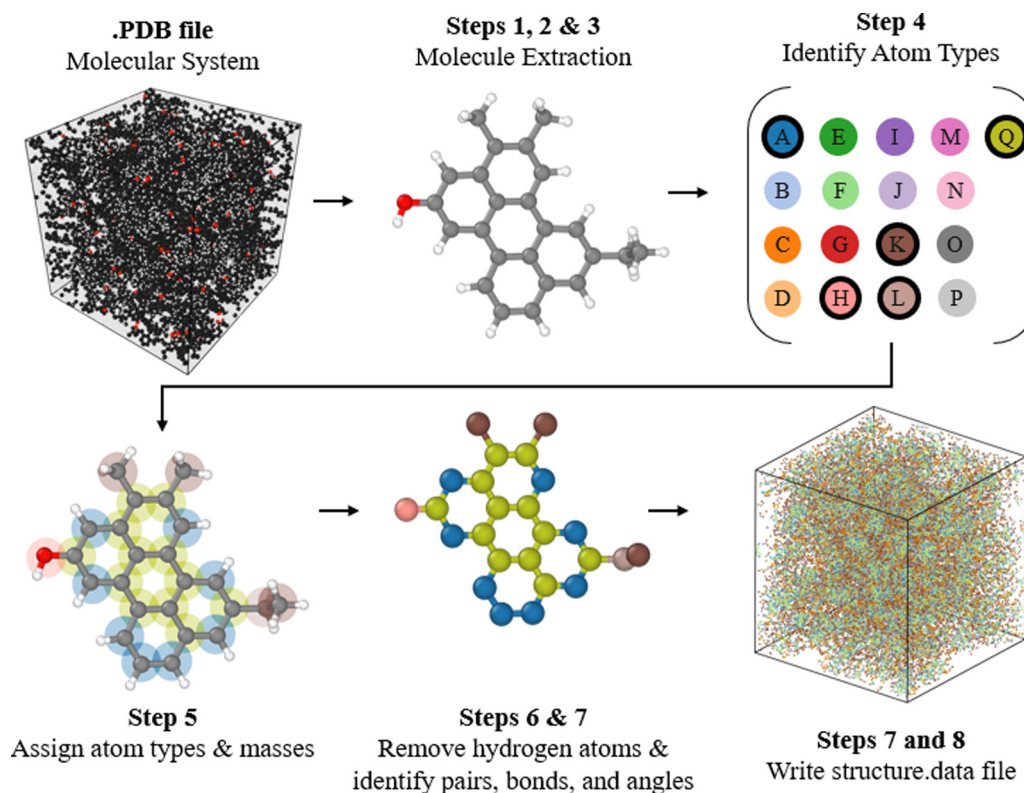


Fig. 1. Diagram summarizing the steps performed by AA2UA to map all-atom PDB models into their UA-CG analog.

of interaction terms in the developed UA force field, shown in Eq. (1).

$$E = \sum E^{nb} + E^b + E^a = \sum_{i=1}^{N_a} E_{ijk}^a + \sum_{i=1}^{N_b} E_{ij}^b + \sum_{i=1}^N \sum_{j=i+1}^N E_{ij}^{nb} \quad (1)$$

where E^{nb} is energy from non-bonded interactions, E^b from 2-body bonded interactions, and E^a from 3-body angular interactions, and N_b , and N_a are the total number of bonded and angular interactions in the system.

- 8. Generate LAMMPS Structure File:** Utilizing the box dimensions, atomic types, masses, positions, and identified bonds and angles, the program generates a **LAMMPS structure.data file**, ready to be loaded into a LAMMPS execution script by using the command “`read_data structure.data`”, often called from within a LAMMPS script file (https://docs.lammps.org/Commands_input.html).

A depiction of these steps can be found in Fig. 1.

3. Software architecture

AA2UA is divided into two files: “`aa2ua.py`”, which corresponds to a short script which initializes and executes the program, and “`core_functions.py`”, which contains all the functions and variable definitions required for the program to execute. The latter is not intended to be edited by the user, as all the tuneable parameters for the script’s execution are in `aa2ua.py`. File `core_functions.py`’s contains 19 functions which can be grouped into 3 types: simple operation functions, responsible for performing trivial operations (e.g., `convert_atom_types_to_int()`, to convert atom labels into unique integers), core functionality functions, responsible for performing chemistry-related operations (e.g., `process_angles()`, to compute all the angle types in the system), and entry-point functions, which serve as wrappers to call a group of simpler functions to perform a wider task

(e.g., `generate_lammps_data()` which write the LAMMPS `structure.data` file). `Core_functions.py` also contains variable definitions containing the force field types, masses, and colors necessary for bead identification. A comprehensive flow diagram depicting the function calls during the program’s execution can be found in Fig. 2. Table 2 contains a short description of all the functions in `core_functions.py`.

4. Impact

The introduction of AA2UA aims to simplify the application of the UA force field developed by our group, as detailed in the article “*Introducing a force-matched united atom force field to explore larger spatiotemporal domains in molecular dynamics simulations of bitumen*” [8]. Despite its availability online, this force field has seen limited implementation. AA2UA’s user-friendly design allows researchers who have already conducted all-atom simulations on bitumens, heavy oil mixtures, polymers, and other hydrocarbons to efficiently convert and map their all-atom systems into the developed CG models using LAMMPS, propelling the use of our force field while seamlessly enabling scientists to perform otherwise complicated simulations with relative ease.

AA2UA has already proven its utility in earlier studies of microscale properties of bitumen, facilitating the setup of hundreds of complex hydrocarbon systems swiftly and seamlessly [21]. It is currently employed by multiple research groups studying bituminous materials and by corporate projects, including those by TotalEnergies and the Ministry of Infrastructure and Water Management of the Netherlands, to design and characterize custom molecules in the Energy sector.

Furthermore, the core functionality of AA2UA can be readily adapted to accommodate other force fields, whether AA, UA, or CG. AA2UA performs analogous steps to those used in force field implementations in LAMMPS, such as iterating through system particles, identifying their type, mass, and charge, and generating a topology file compatible with LAMMPS.

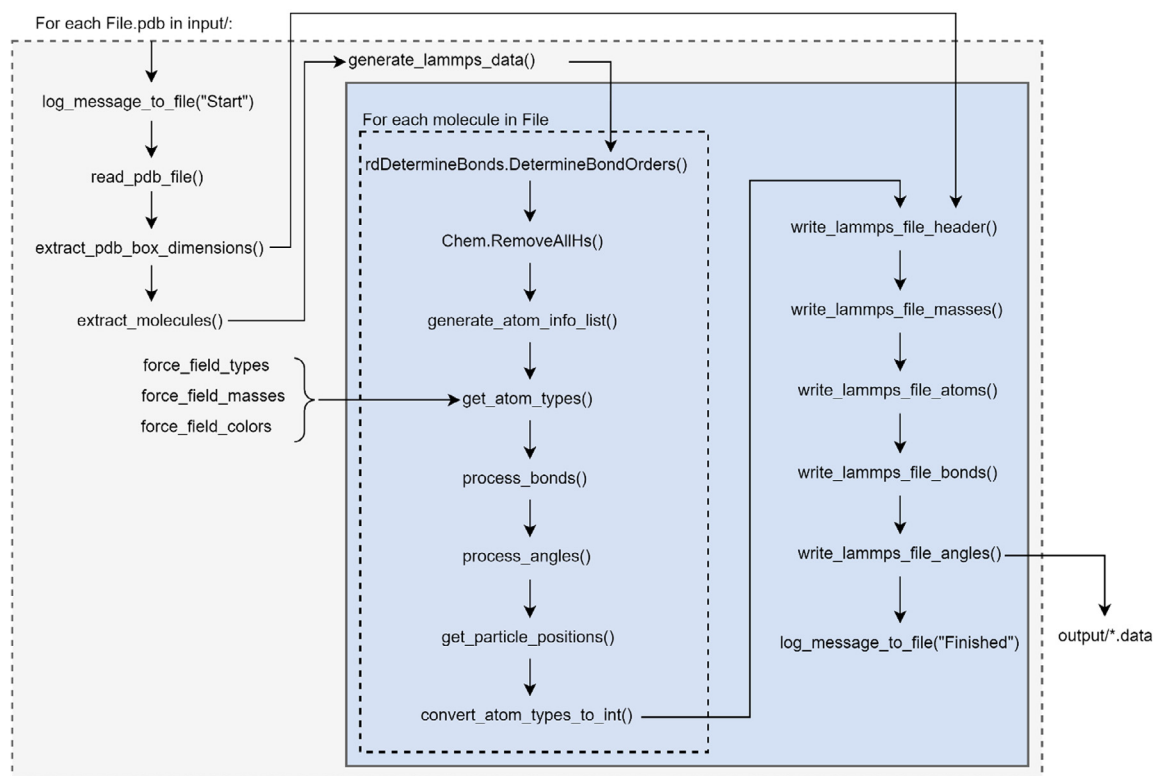


Fig. 2. A schematic of the code execution illustrating the relationships between directories, files, classes, and functions is presented. Dashed lines enclose loop structures, while the blue box encompasses operations related to LAMMPS.

Table 2
Short description of each function in AA2UA.

Function name	Steps (see above)	Description
main_	–	Main execution flow
initialize_log_file()	–	Initialize logging
log_message_to_file()	–	Log messages
read_pdb_file()	1	Read PDB file contents into list of lines
extract_pdb_box_dimensions() extract_molecules()	2	Extract simulation box dimensions Separate molecules in PDB file
generate_lammps_data() DetermineBondOrders() Chem. RemoveAllHs()	3 4–5	Generate LAMMPS data files Determine higher bonding orders (e.g., aromatic) Remove hydrogen atoms
generate_atom_info_list() get_atom_types() process_bonds()		Generate atom labels and indices as per Table 1. Map atom labels to force field types per Table 1. Identify and process bond types per Equation 1
process_angles() get_particle_positions()	6–7	Identify and process angle types per Equation 1 Retrieve 3D atom positions from PDB file
convert_atom_types_to_int() write_lammps_file_header()		Convert atom types to integers Write LAMMPS file header
write_lammps_file_masses() write_lammps_file_atoms() write_lammps_file_bonds() write_lammps_file_angles()	7–8	Write atom mass information Write bead (atom) data Write bond data Write angle data

This simplification is critical as it enables researchers, particularly those not specialized in numerical computing and coarse-grained force fields, to perform simulations with relative ease. AA2UA is expected to significantly impact the oil, gas, and construction sectors by enabling access to larger and longer spatiotemporal scales in simulations involving heavy organic mixtures. This will lead to faster and more accurate predictions of mechanical and rheological properties, thus enhancing material development and performance assessments.

5. Limitations and future work

The application of MD simulations is significantly constrained by the high entry barriers, which stem from the specialized knowledge required in numerical computing, chemistry, and particle physics, as well as the limited availability of tools to perform the necessary tasks for running these simulations. These challenges are even more pronounced when dealing with CG force fields, which are highly customized. The

use of such force fields by external researchers becomes nearly impossible without access to specialized tools. This issue frequently arises when force fields are published: while the data is available in the literature, implementing them into molecular systems is challenging due to the lack of tools developed by the corresponding research teams [22].

The introduction of AA2UA aims to simplify the use of the UA force field developed by our group, which, despite being available online, has been extremely limited in its implementation. The user-friendly design of AA2UA allows researchers who have already performed all-atom simulations on bitumens, heavy oil mixtures, polymers, and other hydrocarbons to easily convert and map their all-atom systems into CG models using LAMMPS. This facilitates the otherwise complex task of conducting UA-CG simulations efficiently.

This simplification is crucial as it enables researchers who are not fully adept in numerical computing, particularly in the application of coarse-grained force fields, to perform simulations. This advancement is expected to have a direct impact on the oil, gas, and construction sectors by providing access to larger and longer spatiotemporal scales in simulations involving heavy organic mixtures. Consequently, this will allow for faster and more accurate predictions of mechanical and rheological properties.

AA2UA forms part of a suite of tools currently being developed by our research group [23,24]. Its purpose is to enhance the influence of molecular studies on Civil Engineering practices and applications. Despite computational chemistry not traditionally being a focal point in this field, its utilization is increasingly advantageous, offering savings in resources and providing valuable observations and insights through molecular simulations.

CRediT authorship contribution statement

Eli I. Assaf: Writing – review & editing, Writing – original draft, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Xueyan Liu:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition, Conceptualization. **Sandra Erkens:** Writing – review & editing, Supervision, Resources, Project administration, Investigation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author(s) used OpenAI's ChatGPT4.0 to simplify verbose paragraph descriptions. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

Acknowledgments

This paper/article is created under the research program Knowledge-based Pavement Engineering (KPE). KPE is a cooperation between Rijkswaterstaat, TNO, and TU Delft in which scientific and applied knowledge is gained about asphalt pavements and which contributes to the aim of Rijkswaterstaat to be completely climate neutral

and to work according to the circular principle by 2030. The opinions expressed in these papers are solely from the authors.

References

- [1] K. Ohno, K. Esfarjani, Y. Kawazoe, *Computational Materials Science: From Ab Initio to Monte Carlo Methods*, Springer, 2018.
- [2] F. Ercolessi, *A Molecular Dynamics Primer*, vol. 19, Springer College in Computational Physics, ICTP, Trieste, 1997.
- [3] C. Giavarini, D. Mastrofini, M. Scarsella, L. Barré, D. Espinat, Macrostructure and rheological properties of chemically modified residues and bitumens, *Energy Fuels* 14 (2) (2000) 495–502.
- [4] D. Lesueur, The colloidal structure of bitumen: Consequences on the rheology and on the mechanisms of bitumen modification, *Adv. Colloid Interface Sci.* 145 (1–2) (2009) 42–82.
- [5] A.T. Pauli, Asphalt compatibility testing using the automated Heithaus titration test, 1996, pp. 1276–1281, Preprints of Papers-American Chemical Society Division Fuel Chemistry 41.
- [6] S.Y. Joshi, S.A. Deshmukh, A review of advancements in coarse-grained molecular dynamics simulations, *Mol. Simul.* 47 (10–11) (2021) 786–803.
- [7] M. Guenza, M. Dinpajoo, J. McCarty, I. Lyubimov, Accuracy, transferability, and efficiency of coarse-grained models of molecular liquids, *J. Phys. Chem. B* 122 (45) (2018) 10257–10278.
- [8] E.I. Assaf, X. Liu, P. Lin, S. Erkens, Introducing a force-matched united atom force field to explore larger spatiotemporal domains in molecular dynamics simulations of bitumen, *Mater. Des.* 240 (2024) 112831.
- [9] H. Sun, S.J. Mumby, J.R. Maple, A.T. Hagler, An ab initio CFF93 all-atom force field for polycarbonates, *J. Am. Chem. Soc.* 116 (7) (1994) 2978–2987.
- [10] L. Yang, C.-h. Tan, M.-J. Hsieh, J. Wang, Y. Duan, P. Cieplak, J. Caldwell, P.A. Kollman, R. Luo, New-generation amber united-atom force field, *J. Phys. Chem. B* 110 (26) (2006) 13166–13176.
- [11] J.A. Graham, J.W. Essex, S. Khalid, PyCGTOOL: Automated generation of coarse-grained molecular dynamics models from atomistic trajectories, *J. Chem. Inform. Model.* 57 (4) (2017) 650–656.
- [12] T.A. Wassenaar, H.I. Ingólfsson, R.A. Bockmann, D.P. Tieleman, S.J. Marrink, Computational lipidomics with insane: A versatile tool for generating custom membranes for molecular simulations, *J. Chem. Theory Comput.* 11 (5) (2015) 2144–2155.
- [13] J.A. Greathouse, Building LAMMPS data files with car/mdf files and the msi2lmp utility, Sandia National Laboratories: Albuquerque, NM, USA, 2010.
- [14] Y. Peng, A.J. Pak, A.E. Durumeric, P.G. Sahrman, S. Mani, J. Jin, T.D. Loose, J. Beiter, G.A. Voth, OpenMSCG: A software tool for bottom-up coarse-graining, *J. Phys. Chem. B* 127 (40) (2023) 8537–8550.
- [15] A.Y. Shih, A. Arkhipov, P.L. Freddolino, K. Schulten, Coarse grained protein-lipid model with application to lipoprotein particles, *J. Phys. Chem. B* 110 (8) (2006) 3674–3684.
- [16] 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.
- [17] A.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, LAMMPS-a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales, *Comput. Phys. Commun.* 271 (2022) 108171.
- [18] S.K. Burley, H.M. Berman, G.J. Kleywegt, J.L. Markley, H. Nakamura, S. Velankar, Protein Data Bank (PDB): The single global macromolecular structure archive, in: *Protein Crystallography: Methods and Protocols*, 2017, pp. 627–641.
- [19] G. Landrum, Rdkit documentation, 2013, p. 4, Release 1 (1–79).
- [20] J.H. Jensen, Xyz2mol, 2020, GitHub repository 985.
- [21] E.I. Assaf, X. Liu, P. Lin, S. Erkens, S. Nahar, L.I. Mensink, Studying the impact of phase behavior in the morphology of molecular dynamics models of bitumen, *Mater. Des.* 230 (2023) 111943.
- [22] S.J. Weiner, P.A. Kollman, D.A. Case, U.C. Singh, C. Ghio, G. Alagona, S. Profeta, P. Weiner, A new force field for molecular mechanical simulation of nucleic acids and proteins, *J. Am. Chem. Soc.* 106 (3) (1984) 765–784.
- [23] E.I. Assaf, X. Liu, P. Lin, S. Erkens, PDB2dat: Automating LAMMPS data file generation from PDB molecular systems using Python, Rdkit, and Pysimm, *Softw. Impacts* (2024) 100656.
- [24] E.I. Assaf, X. Liu, P. Lin, S. Erkens, SMI2PDB: A self-contained python tool to generate atomistic systems of organic molecules using their SMILES notations, *Softw. Impacts* (2024) 100655.