

An introduction to classical molecular dynamics simulation for experimental scattering users

Andrew R. McCluskey,^{a,b*} James Grant,^c Adam R. Symington,^a Tim Snow,^{b,d} James Douth,^e Benjamin J. Morgan,^{a*} Stephen C. Parker^a and Karen J. Edler^a

Received 4 February 2019

Accepted 30 March 2019

Edited by J. M. Garcia-Ruiz, CSIC-UGR, Granada, Spain

Keywords: open educational resources; classical simulation; multimodal analysis; scattering.

Supporting information: this article has supporting information at journals.iucr.org/j

^aDepartment of Chemistry, University of Bath, Claverton Down, Bath BA2 7AY, UK, ^bDiamond Light Source, Harwell Campus, Didcot OX11 0DE, UK, ^cComputing Services, University of Bath, Claverton Down, Bath BA2 7AY, UK, ^dSchool of Chemistry, University of Bristol, Bristol BS8 1TS, UK, and ^eISIS Facility, Rutherford Appleton Laboratory, STFC, Chilton, Didcot OX11 0QX, UK. *Correspondence e-mail: a.r.mccluskey@bath.ac.uk, andrew.mccluskey@diamond.ac.uk, b.j.morgan@bath.ac.uk

Classical molecular dynamics simulations are a common component of multimodal analyses of scattering measurements, such as small-angle scattering and diffraction. Users of these experimental techniques often have no formal training in the theory and practice of molecular dynamics simulation, leading to the possibility of these simulations being treated as a ‘black box’ analysis technique. This article describes an open educational resource (OER) designed to introduce classical molecular dynamics to users of scattering methods. This resource is available as a series of interactive web pages, which can be easily accessed by students, and as an open-source software repository, which can be freely copied, modified and redistributed by educators. The topics covered in this OER include classical atomistic modelling, parameterizing interatomic potentials, molecular dynamics simulations, typical sources of error and some of the approaches to using simulations in the analysis of scattering data.

1. Introduction

The use of molecular dynamics simulations to help analyse and interpret experimental data from small-angle scattering and diffraction studies has grown significantly over the past ten years (Pan *et al.*, 2012; Boldon *et al.*, 2015; Hub, 2018; Ivanović *et al.*, 2018; East *et al.*, 2016; Wall *et al.*, 2014; Wall, 2018; Satoh *et al.*, 2015), with the percentage of small-angle scattering publications that mention molecular dynamics reaching more than 20% in 2018 (Fig. 1). Users of scattering and diffraction techniques often have backgrounds in experimental science and may have received little formal training in the theory or practice of computational modelling. This can lead to molecular dynamics simulations being used as a ‘black box’, without understanding the underlying methodologies or considering possible sources of error. The use of molecular simulations without any technical understanding can lead to unintended but severe errors. To help support researchers’ use of molecular dynamics simulation in their analysis of scattering data while reducing this risk of modelling errors, a number of software tools, such as *WAXSiS* and *SASSIE* (Chen & Hub, 2014; Knight & Hub, 2015; Perkins *et al.*, 2016), that present easy-to-use graphical web-based user interfaces have been developed.

A complementary approach is to organize educational activities, such as lectures or workshops, tailored to introduce molecular simulation techniques to audiences of scattering and diffraction users. One example is the annual ISIS Neutron Training Course, which includes a module titled *An Introduction to Molecular Dynamics for Neutron Scattering*. This



module covers the fundamentals of classical molecular dynamics simulation, presents applications of these methods in neutron science and gives students practical hands-on experience with the *SASSIE* software package (Perkins *et al.*, 2016).

While lectures and workshops are an effective tool for education and training, participation can be limited owing to difficulties attending in person (due to location or cost) or physical limits on student numbers. An alternative educational strategy gaining popularity within scientific and engineering communities is the publication of technology-enhanced open educational resources (OERs). These are courses, lectures or learning resources published online that are freely available for use by anyone. In addition to their broad accessibility, these resources have permissive ‘open’ copyright licences that allow their use in the ‘5R activities’: retain, reuse, revise, remix and redistribute (<https://opencontent.org/definition>). Publishing a resource as an OER increases its reach and impact, because others may use it in their own teaching in its original form and are also free to modify, and redistribute, the material to better suit their aims. The Jupyter Notebook framework (Kluyver *et al.*, 2016) has become a popular platform for OERs that teach computational skills, because it allows authors to include instructional text, images and other media, alongside executable editable code, in an example of ‘literate programming’ (Knuth, 1984). This format encourages students to directly interact with code examples by running, editing and rerunning these within the source document (Barba *et al.*, 2017), supporting exploratory experiential learning (Papert, 1993).

Here, we present an online open-source interactive learning resource written to introduce members of the scattering and diffraction community to molecular dynamics simulations. This OER comprises five lessons that introduce classical molecular dynamics methods and show how these can be used to assist in the analysis of experimental scattering data by the calculation of a simulated scattering profile from the molecular dynamics simulation. We use the open-source Python library *pylj* (McCluskey *et al.*, 2018; McCluskey, Symington *et al.*, 2019) to provide simple, but computationally authentic,

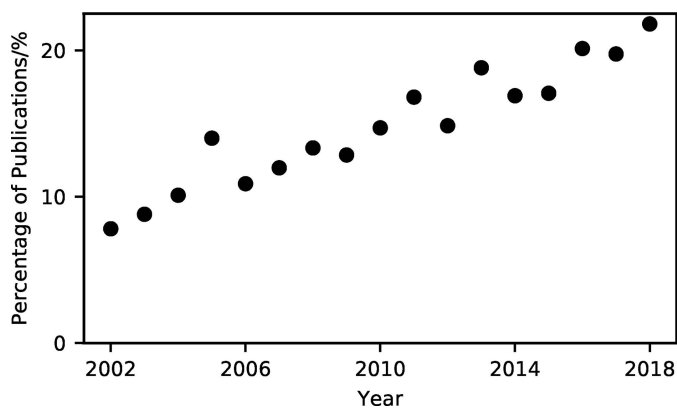


Figure 1

The percentage of publications that mention ‘small-angle scattering’ that also mention ‘molecular dynamics’, determined from the numbers of matching Google Scholar results.

examples of simulations that demonstrate, visually and programmatically, the conceptual relationships between simulation and scattering techniques. In this article, we discuss the structure of the resource and describe how a student may get the most from the resource. [In this work ‘the student’ refers to anyone working through the OER, regardless of career position.]

2. Assumed prior knowledge

The OER, entitled *The Interaction Between Simulation and Scattering*, makes use of the Python programming language to include interactive examples of mathematical and algorithmic content. To be able to fully use this resource, therefore, requires some knowledge of, or willingness to learn, Python. We have, however, attempted to design the resource in such a fashion that an in-depth knowledge of Python is not a prerequisite. It is anticipated that the users of this resource will have some experience of at least undergraduate level chemistry or physics, and a commensurate level of mathematical understanding. This background knowledge will be particularly important for the sections dealing with the functional form and chemical basis of classical interaction potentials.

3. Resource construction

The resource is available in two main formats. The first is a series of web pages, hosted at http://pythoninchemistry.org/sim_and_scatt. The second is the source-code repository used to build these web pages, hosted at https://github.com/pythoninchemistry/sim_and_scatt (McCluskey, Grant *et al.*, 2019). The source content consists of a set of Jupyter Notebooks and Markdown files, which are automatically compiled using the *jupyter-book* tool (Lau & Holdgraf, 2019) to generate the web version. This system allows the resulting web pages to include text, equations and figures, which we use to describe key concepts and to explain details of algorithms, as well as Python code blocks, which we use to provide specific examples. The web pages have *Thebelab* and *BinderHub* integrations (Ragan-Kelley, Thiery *et al.*, 2019; Ragan-Kelley, Panda *et al.*, 2019; Project Jupyter *et al.*, 2018), which allow students to launch interactive versions of these web pages that allow execution and editing of the included Python code. We believe that the ability to read the resource as an ‘interactive document’ will help students to develop a deeper understanding of the materials than they would from a traditional static document. The resource is provided under a CC-BY licence (<https://creativecommons.org/licences/by/4.0/>), while the *jupyter-book* software is shared under an MIT licence (<https://opensource.org/licenses/MIT>), both of which are open and highly permissive. This allows readers to reuse or remix the material to enhance their own educational platform, and for secondary authors to contribute to improving the source material.

4. Resource outline

The resource follows a simple outline that introduces key aspects of molecular dynamics simulations.

4.1. Home

The welcome page introduces the resource and gives the student information about how the resource may be used, including details of the *Thebelab* and *BinderHub* integrations. This page also contains details about the permitted use and sharing of the content of the resource, and about the resource licence. In addition, this page includes a list of authors and contributors.

4.2. Classical methods

After a brief introduction, this section introduces concepts related to classical simulation methods. This includes the use of interatomic potential functions, alongside some examples, such as the Lennard-Jones and Buckingham potential models (Jones, 1924; Buckingham, 1938). We then introduce the problem of parameterizing a potential model, including the use of higher-accuracy quantum-mechanical calculations to do so. The presence of off-the-shelf general potential models is discussed, with the caveat that they may still require system-specific optimization. Finally, we mention mixing rules, again discussing the possible problems that a user may encounter if applying these blindly to specific systems.

4.3. Molecular dynamics

Having introduced the concept of classical interatomic potentials, we then discuss how these are used in molecular dynamics simulations. We work through how a one-dimensional NVE (constant number, volume and energy) molecular dynamics simulation may be built, using the velocity–Verlet algorithm and the Lennard-Jones potential model (Jones, 1924; Swope *et al.*, 1982). The velocity–Verlet algorithm is introduced in terms of Newton's laws of motion and the generalized equations of motion. Finally, we discuss a range of key factors that can affect molecular dynamics simulations, including choice of simulation ensembles, setting the range cut-off for an interatomic potential and the use of periodic boundary conditions.

4.4. *pylj* and the interaction with scattering

The final aspect of the resource covers using molecular dynamics simulations to understand scattering profiles. This is initially presented as a practical example, using the open-source *pylj* package (McCluskey *et al.*, 2018; McCluskey, Symington *et al.*, 2019). We demonstrate a two-dimensional molecular dynamics simulation of argon particles interacting through a Lennard-Jones potential. The student is first shown a working *pylj* simulation and invited to interact with the simulation and the custom plotting functionality of *pylj*. The concept of a radial distribution function (RDF) is then shown, and the students are given the opportunity to run some *pylj* simulations with the RDF being output alongside the simulation window. Next we present the Debye (1915) equation and show how it may be used to calculate scattering data from a simulation. The student is invited to observe the effect of simulation temperature on the resulting scattering profile. We finish by discussing alternative, faster, algorithms for calcu-

lating scattering profiles, such as the Fibonacci sequence or golden vectors methods (Svergun, 1994; Watson & Curtis, 2013).

4.5. 'Real' simulation and scattering

Having shown the development of a scattering profile from an idealized system, we then direct the student to a popular resource for the *GROMACS* (Berendsen *et al.*, 1995) molecular dynamics software. This resource gives a quick introduction to using *GROMACS* to simulate a lysozyme molecule in buffer (Lemkul, 2019). The student may then use their own simulated trajectory or one that can be downloaded from the OER. We show how the system may be visualized, introduce the *MDAnalysis* Python package for the analysis of molecular dynamics trajectories (Michaud-Agrawal *et al.*, 2011; Gowers *et al.*, 2016) and show the scattering profile developed from the lysozyme simulation compared with experimental data (Franke *et al.*, 2015). The module finishes by pointing the student to resources to more easily resolve scattering data from molecular simulation, such as *SASSIE* and *CRYSOL* (Perkins *et al.*, 2016; Svergun *et al.*, 1995). The focus of this module is to introduce simulation methodologies to users of scattering to aid their understanding, not to derive the exact mechanics of the calculation of scattering from simulation. Resources for that purpose already exist and have well developed tutorials, so it is not necessary to recreate such software here (Perkins *et al.*, 2016; Svergun *et al.*, 1995).

5. Future outlook

The aim of this resource is to benefit students and researchers interested in combining molecular dynamics analysis with scattering and diffraction experiments. Our intention in publishing this material as an OER is to promote interest from other parties towards reuse and remixing the material for their own educational purposes. We plan to implement this material within training at the ISIS Neutron and Muon Source and Diamond Light Source. Finally, it is hoped, *via* student and community feedback, that the implementation, content and pedagogical approach of this resource can be increased over time.

6. Usage

All analysis/plotting scripts and figure files, allowing for a fully reproducible and automated analysis workflow for the work presented, are available at https://github.com/arm61/sim_and_scat_paper and <https://doi.org/10.5281/zenodo.2654370> under a CC BY-SA 4.0 license.

7. Author contributions

The open education resource was developed, and the manuscript written, by ARM with input from all authors.

Funding information

ARM is grateful to the University of Bath and Diamond Light Source for co-funding a studentship (studentship No.

STU0149). BJM acknowledges support from the Royal Society (grant No. UF130329).

References

- Barba, L. A., Wickenheiser, A. & Watkins, R. (2017). *CyberTraining: DSE – The Code Maker: Computational Thinking for Engineers with Interactive, Contextual Learning*, <https://doi.org/10.6084/m9.figshare.5662051.v1>.
- Berendsen, H. J. C., van der Spoel, D. & van Drunen, R. (1995). *Comput. Phys. Commun.* **91**, 43–56.
- Boldon, L., Laliberte, F. & Liu, L. (2015). *Nano Rev.* **6**, 25661.
- Buckingham, R. A. (1938). *Proc. R. Soc. London Ser. A*, **168**, 264–283.
- Chen, P.-C. & Hub, J. S. (2014). *Biophys. J.* **107**, 435–447.
- Debye, P. (1915). *Ann. Phys.* **351**, 809–823.
- East, A., Mechaly, A. E., Huysmans, G. H. M., Bernarde, C., Tello-Manigne, D., Nadeau, N., Pugsley, A. P., Buschiazzi, A., Alzari, P. M., Bond, P. J. & Francetic, O. (2016). *Structure*, **24**, 92–104.
- Franke, D., Jeffries, C. M. & Svergun, D. I. (2015). *Nat. Methods*, **12**, 419–422.
- Gowers, R., Linke, M., Barnoud, J., Reddy, T., Melo, M., Seyler, S., Domanski, J., Dotson, D., Buchoux, S., Kenney, I. & Beckstein, O. (2016). *Proceedings of the 15th Python in Science Conference (SciPy 2016)*, pp. 98–105. <https://doi.org/10.25080/Majora-629e541a-00e>.
- Hub, J. S. (2018). *Curr. Opin. Struct. Biol.* **49**, 18–26.
- Ivanović, M. T., Bruetzel, L. K., Lipfert, J. & Hub, J. S. (2018). *Angew. Chem. Int. Ed.* **57**, 5635–5639.
- Jones, J. E. (1924). *Proc. R. Soc. London Ser. A*, **106**, 463–477.
- Kluyver, T., Ragan-Kelley, B., Perez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., Ivanov, P., Avila, D., Abdalla, S. & Willing, C. (2016). *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pp. 87–90. Amsterdam: IOS Press.
- Knight, C. J. & Hub, J. S. (2015). *Nucleic Acids Res.* **43**, W225–W230.
- Knuth, D. E. (1984). *Comput. J.* **27**, 97–111.
- Lau, S. & Holdgraf, C. (2019). *jupyter/jupyter-book Beta-2*, <https://github.com/jupyter/jupyter-book>.
- Lemkul, J. A. (2019). *GROMACS Tutorial: Lysozyme in Water*, <http://www.md-tutorials.com/gmx/lysozyme/index.html>.
- McCluskey, A. R., Grant, J., Symington, A. R., Snow, T., Douth, J., Morgan, B. J., Parker, S. C. & Edler, K. J. (2019). *pythoninchemistry/sim_and_scat: v1.0-paper*, <http://doi.org/10.5281/zenodo.2654373>.
- McCluskey, A. R., Morgan, B. J., Edler, K. J. & Parker, S. C. (2018). *J. Open Source Educ.* **1**, 19.
- McCluskey, A. R., Symington, A. R., Morgan, B. J., Edler, K. J. & Parker, S. C. (2019). *Pylj*, <http://doi.org/10.5281/zenodo.2587898>.
- Michaud-Agrawal, N., Denning, E. J., Woolf, T. B. & Beckstein, O. (2011). *J. Comput. Chem.* **32**, 2319–2327.
- Pan, J., Heberle, F. A., Tristram-Nagle, S., Szymanski, M., Koepfner, M., Katsaras, J. & Kucerka, N. (2012). *BBA Biomembranes*, **1818**, 2135–2148.
- Papert, S. (1993). *Mindstorms*. New York: Basic Books.
- Perkins, S. J., Wright, D. W., Zhang, H., Brookes, E. H., Chen, J., Irving, T. C., Krueger, S., Barlow, D. J., Edler, K. J., Scott, D. J., Terrill, N. J., King, S. M., Butler, P. D. & Curtis, J. E. (2016). *J. Appl. Cryst.* **49**, 1861–1875.
- Project Jupyter, Bussonnier, M., Forde, J., Freeman, J., Granger, B., Head, T., Holdgraf, C., Kelley, K., Nalvarte, G., Osheroff, A., Pacer, M., Panda, Y., Perez, F., Ragan-Kelley, B., Willing, C. (2018). *Proceedings of the 17th Python in Science Conference (SciPy 2018)*, pp. 113–120. <https://doi.org/10.25080/Majora-4af1f417-011>.
- Ragan-Kelley, B., Panda, Y., Willing, C., Holdgraf, C., Erdogan, K., Head, T., Bussonnier, M., Zarea, H., van Niekerk, M., Yick, C. & Sundell, E. (2019). *jupyterhub/binderhub*, <https://github.com/jupyterhub/binderhub>.
- Ragan-Kelley, B., Thiery, N. M., Corlay, S., Gutsche, S., Holdgraf, C. & Head, N. J. (2019). *minrk/thebelab*, <https://github.com/minrk/thebelab>.
- Satoh, M., Saburi, H., Tanaka, T., Matsuura, Y., Naitow, H., Shimozono, R., Yamamoto, N., Inoue, H., Nakamura, N., Yoshizawa, Y., Aoki, T., Tanimura, R. & Kunishima, N. (2015). *FEBS Open Bio.* **5**, 557–570.
- Svergun, D. I. (1994). *Acta Cryst.* **A50**, 391–402.
- Svergun, D., Barberato, C. & Koch, M. H. J. (1995). *J. Appl. Cryst.* **28**, 768–773.
- Swope, W. C., Andersen, H. C., Berens, P. H. & Wilson, K. R. (1982). *J. Chem. Phys.* **76**, 637–649.
- Wall, M. E. (2018). *IUCrJ*, **5**, 172–181.
- Wall, M. E., Van Benschoten, A. H., Sauter, N. K., Adams, P. D., Fraser, J. S. & Terwilliger, T. C. (2014). *Proc. Natl Acad. Sci. USA*, **111**, 17887–17892.
- Watson, M. C. & Curtis, J. E. (2013). *J. Appl. Cryst.* **46**, 1171–1177.