

Review of Algorithms Used in Molecular Dynamics Simulations



Aislyn A. Laurent, BSc Student [1]*

[1] Department of Chemistry and Biochemistry, University of Windsor, Windsor, Ontario, Canada, N9B 3P4

*Corresponding Author: lewisla@uwindsor.ca



Abstract

Molecular Dynamics simulations are part of the study of chemical systems via the computational method. This review looks at algorithms used during this technique and examines how they work, and how they contribute to successful results using this critical method.

Keywords: Molecular Dynamics; MD; Algorithms; Verlet; Fast Multipole Method; Review

Introduction

Molecular Dynamics is a technique which simulates the movement of atoms in a system relative to each other and their environment [1]. Software is used to generate a coordinate file containing the location of each atom in 3D space. Classical Newtonian equations are applied in order to determine the energy, trajectory and speed of each atom [1]. In this way a molecule can be optimized into the lowest energy configuration.

These simulations rely on algorithms derived from numerical methodologies. This work discusses these algorithms, their relative efficiency and their practical application in commonly used software. The algorithmic process of NAMD, a free and open source program, will be investigated here in the most detail.

While the applications of these methods are limited to (relatively) simple calculations, numerical methods are able to achieve suitably accurate results within an acceptable timeframe.

Methods

NAMD (Nanoscale Molecular Dynamics, formerly Not Another Molecular Dynamics Program) is a popular open source program used for MD simulations. This software employs several algorithms in order to determine the lowest energy for the provided system.

Most of these algorithms are classical solutions for N-body interaction problems. By using different methods to solve for specific interactions, the efficiency of the overall process can be increased. The two most well documented algorithms used by NAMD are the Verlet method and the Fast Multipole method, which are used to solve the kinematic equations for motion and the movement of charges through the system respectively [3].

As seen in [Figure 1](#), the system is first decomposed into several rectangular regions [3]. A cluster of these regularly sized sections is assigned to each available processor. By ensuring that the size of each section is such that no section cuts off vital information, communication can be limited to sections adjacent to each other [3]. This design allows for easy scalability and dynamic allocation of resources.

Trajectories are then calculated for each particle, considering their interaction with their neighbours and the system around them. This process proceeds stepwise via the Verlet method, which solves the kinematic equations for the motion of an object [2]. The result of each step is compared via a central difference calculation. The process continues until a maximum number of steps is reached, or the energy of the system no longer decreases [1].

The Verlet method is able to run in $O(n \log(n))$ in the best case [2]. [Figure 2](#) demonstrates Stormer Verlet, which includes calculations for particle velocities [2], as written in python.

The movement of charge is calculated using the Fast Multipole Method, which again is a numerical method. A numerical approach is used to determine N-body interactions. [Figure 3](#) shows how this is accomplished by recursively solving for the integrations of particles in successively smaller sections [4]. It also allows charges nearby to each other to be considered as one source [3]. The Fast Multipole Method is able to operate as quickly as $O(N)$ in the best case [4].

Conclusions

Optimizations to each method, which allow work to be distributed across the system, allow for flexibility and scalability. It is this flexibility which allows for the wide application of Molecular Dynamics simulations. A broad picture of how a system may appear in nature can be drawn.

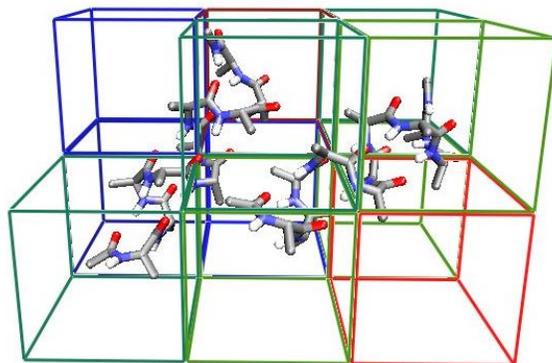


Figure 1: An image showing “patches” in NAMD. [Image retrieved from: <https://www.ks.uiuc.edu/Research/Algorithms/>]

```
def stormer_verlet(pos, acc, dt):  
    prev_pos = pos  
    time = 0  
    vel = 0  
  
    while pos > 0:  
        time += dt  
        next_pos = pos * 2 - prev_pos + acc * dt * dt  
        prev_pos, pos = pos, next_pos  
        vel += acc * dt  
  
    return time, vel
```

Figure 2: Verlet method as written in python. [Image retrieved from: https://www.algorithm-archive.org/contents/verlet_integration/verlet_integration.html]

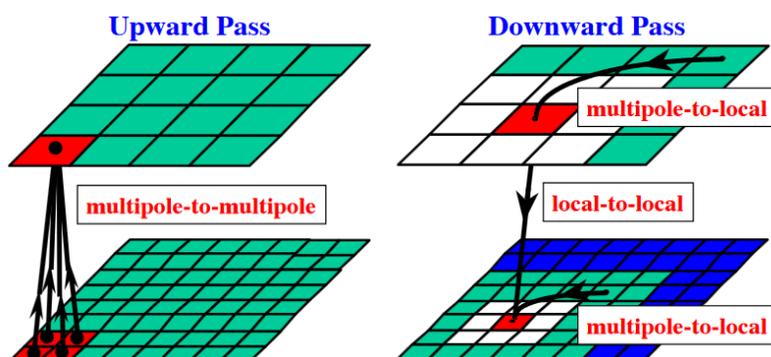


Figure 3: Interactions are derived by traversing up and down a tree, where nodes represent sections of particles. [Image retrieved from: http://www-personal.umich.edu/~hanliang/publications/FMM_Tutorial_Hanliang.pdf]

Therefore, MD simulations are often a necessary step – allowing conclusions to be drawn about which more resource heavy methods can be applied.

In future work, the efficacy of each method may be evaluated, compared, and contrasted to similar competing methods.

List of Abbreviations Used (If Any)

MD: Molecular Dynamics

NAMD: Nanoscale Molecular Dynamics (formerly Not Another Molecular Dynamics Program)

Conflicts of Interest

The author declares that they have no conflict of interests.

Ethics Approval and/or Participant Consent

This work is a literature review and therefore does not require ethics board approval.

Authors' Contributions

AAL: completed literature review, prepared data, wrote and edited manuscript, and approved final published version

Acknowledgements

Thanks to Dr. James Gauld and his research group for their support during this project. Additionally, thank you to Dr. Pooya Moradian Zadeh for his assistance with editing this work.

Funding

This study was not funded.

References

- [1] Martin-Santamaria, S. (2018). *Computational tools for chemical biology*. London, UK: Royal Society of Chemistry.
- [2] Schloss, J. (n.d.). Verlet Integration. Retrieved July 11, 2019, from https://www.algorithm-archive.org/contents/verlet_integration/verlet_integration.html
- [3] NAMD: Algorithms. (2012, May 17). Retrieved July 11, 2019, from <https://www.ks.uiuc.edu/Research/Algorithms/>
- [4] Guo, H. (2018, April). A Crash Course on Fast Multipole Method (FMM). Retrieved July 22, 2019, from http://www-personal.umich.edu/~hanliang/publications/FMM_Tutorial_Hanliang.pdf

Article Information

Managing Editor: Jeremy Y. Ng

Peer Reviewers: Ikjot Saini, Kalyani Selvarajah, Mahreen Nasir Butt

Article Dates: Received Aug 14 19; Published Oct 16 19

Citation

Please cite this article as follows:

Laurent, AA. Review of algorithms used in molecular dynamics simulations. URNCST Journal. 2019 Oct 16: 3(9).

<https://urncst.com/index.php/urncst/article/view/157>

DOI Link: <https://doi.org/10.26685/urncst.157>

Copyright

© Aislyn A. Laurent (2019). Published first in the Undergraduate Research in Natural and Clinical Science and Technology (URNCST) Journal. This is an open access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work, first published in the Undergraduate Research in Natural and Clinical Science and Technology (URNCST) Journal, is properly cited. The complete bibliographic information, a link to the original publication on <http://www.urncst.com>, as well as this copyright and license information must be included.



URNCST Journal
"Research in Earnest"

Funded by the
Government
of Canada

Canada

Do you research in earnest? Submit your next undergraduate research article to the URNCST Journal!

| Open Access | Peer-Reviewed | Rapid Turnaround Time | International |

| Broad and Multidisciplinary | Indexed | Innovative | Social Media Promoted |

Pre-submission inquiries? Send us an email at info@urncst.com | [Facebook](#), [Twitter](#) and [LinkedIn](#): @URNCST

Submit YOUR manuscript today at <https://www.urncst.com>!