The Journal of Open Source Software

Efficient Polyhedral Gravity Modeling in Modern C++ and Python

Jonas Schuhmacher [1], **Fabio Gratl** [1], **and Pablo Gómez** [2]

1 Technische Universität München, Arcisstraße 21, 80333 München, Germany **2** Advanced Concepts Team, European Space Agency, European Space Research and Technology Centre (ESTEC), Keplerlaan 1, 2201 AZ Noordwijk, The Netherlands

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Open Journals

Reviewers:

- @openjournals

Submitted: 01 January 1970
Published: unpublished

Summary

Polyhedral gravity models are ubiquitous for modeling the gravitational field of irregular bodies, such as asteroids and comets. We present an open-source C++ library for the efficient, parallelized computation of a polyhedral gravity model. We also provide a Python interface to the library using *pybind11*. The library is particularly focused on delivering high performance and scalability which we achieve through vectorization and parallelization with *xsimd* and *thrust*, respectively. The library supports many common formats, such as *.stl*, *.off*, *.ply*, *.mesh* and *tetgen*'s *.node* and *.face*.

Statement of Need

The complex gravitational fields of irregular bodies, such as asteroids and comets, are often modeled using polyhedral gravity models as they provide an analytic solution for the computation of the gravitational potential, acceleration (and second derivative) given a mesh of the body (Tsoulis, 2012; Tsoulis & Gavriilidou, 2021). The computation of the gravitational potential and acceleration is a computationally expensive task, especially for large meshes, which can however benefit from parallelization either over computed targets points for which we seek potential and acceleration or over the mesh. Thus, a high-performance implementation of a polyhedral gravity model is desirable.

While some research code for these models exists, they are not focused on usability and limited to FORTRAN **TODO LINK** and proprietary software like MATLAB **TODO LINK**. There is a lack of well-documented, maintained open-source implementations, particularly in modern programming languages and with a focus on scalability and performance.

The presented software has already seen application in several research works. It has been used to optimize trajectories around the highly irregular comet 67P/Churyumov-Gerasimenko (Marãk et al., 2023). Further, it has been used to study the effectiveness of so-called neural density fields (Izzo & Gómez, 2022), where it can serve as a ground truth and to pretrain neural networks (Schuhmacher et al., 2023). **TODO_add_more_examples**

Thus, overall this model is highly versatile and can be used in a wide range of applications. We hope it will enable further research in the field, especially related to recent machine learning techniques, which typically rely on Python implementations.

Polyhedral Model

On a mathematical level, the implemented model follows the approach by Petrović (Petrović, 1996) as refined by Tsoulis and Petrović (Tsoulis & Petrović, 2001). A comprehensive description of the mathematical foundations of the model is given in the associated student report (Schuhmacher, 2022).

Implementation-wise it makes use of the inherent parallelization opportunity of the approach as it iterates over the mesh. This parallelization is achieved via *thrust* which allows utilizing *OpenMP* and *Intel TBB*. On a finer scale, individual costly operations were investigated and, e.g., the `arctan` operations were vectorized using *xsimd*. On an application side, the user may use the implemented caching mechanism to avoid recomputation of mesh properties, such as the face normals.

Extensive tests using GoogleTest are used via GitHub Actions to ensure the (continued) correctness of the implementation.

Installation & Contribution

The library is available on GitHub [1] and can be installed with *pip* or from *conda* [2]. Build instructions using *CMake* are provided in the repository. The library is licensed under a GPL license.

The project is open to contributions via pull requests with instructions on how to contribute provided in the repository.

Usage Instructions

We provide detailed usage instructions in the technical documentation on ReadTheDocs [3]. Additionally, a minimal working example is given in the repository readme and more extensive examples as a *Jupyter* notebook [4].

Acknowledgements

The authors would like to thank Dario Izzo and Emmanuel Blazquez for their feedback on the original model implementation.

References

Izzo, D., & Gómez, P. (2022). Geodesy of irregular small bodies via neural density fields. *Communications Engineering*, *1*(1), 48.

Marǻk, R., Blazquez, E., & Gómez, P. (2023). Trajectory optimization of a spacecraft swarm orbiting around 67P/Churyumov-Gerasimenko. *Proceedings of the 12th International Conference on Guidance, Navigation & Control Systems (GNC)*.

Petrović, S. (1996). Determination of the potential of homogeneous polyhedral bodies using line integrals. *Journal of Geodesy*, *71*, 44–52.

Schuhmacher, J. (2022). *Efficient polyhedral gravity modeling in modern c++* [Master's thesis]. Technical University of Munich.

Schuhmacher, J., Gratl, F., Izzo, D., & Gómez, P. (2023). Investigation of the robustness of neural density fields. *Proceedings of the 9th International Conference on Astrodynamics Tools and Techniques, ICATT*.

[1] https://github.com/esa/polyhedral-gravity-model
[2] https://anaconda.org/conda-forge/polyhedral-gravity-model
[3] https://polyhedral-gravity-model-cpp.readthedocs.io/en/latest/
[4] https://github.com/esa/polyhedral-gravity-model/blob/main/script/polyhedral-gravity.ipynb

Tsoulis, D. (2012). Analytical computation of the full gravity tensor of a homogeneous arbitrarily shaped polyhedral source using line integrals. *Geophysics*, *77*(2), F1–F11.

Tsoulis, D., & Gavriilidou, G. (2021). A computational review of the line integral analytical formulation of the polyhedral gravity signal. *Geophysical Prospecting*, *69*(8-9), 1745–1760.

Tsoulis, D., & Petrović, S. (2001). On the singularities of the gravity field of a homogeneous polyhedral body. *Geophysics*, *66*(2), 535–539.