

¹ Brahe: A Modern Astrodynamics Dynamics Library for Research and Engineering Applications

³ Duncan Eddy  ¹ and Mykel J. Kochenderfer  ¹

⁴ 1 Stanford University

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: [Open Journals](#) 

Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).
¹⁶
¹⁷
¹⁸
¹⁹
²⁰
²¹

⁵ Summary

⁶ [brahe](#) is a modern astrodynamics dynamics library for research and engineering applications. The representation and prediction of satellite motion is the fundamental problem of astrodynamics.
⁷ The motion of celestial bodies has been studied for centuries with initial equations of motion
⁸ dating back to Kepler ([Kepler, 1619](#)) and Newton ([Newton, 1687](#)). Current research and
⁹ applications in space situational awareness, satellite task planning, and space mission operations
¹⁰ require accurate and efficient numerical tools to perform coordinate transformations, model
¹¹ perturbations, and propagate orbits. [brahe](#) incorporates the latest conventions and models
¹² for time systems and reference frame transformations from the International Astronomical
¹³ Union (IAU) ([Hohenkerk, 2017](#)) and International Earth Rotation and Reference Systems
¹⁴ Service (IERS) ([Petit & Luzum, 2010](#)). It implements force models for Earth-orbiting satellites
¹⁵ including atmospheric drag, solar radiation pressure, and third-body perturbations from the
¹⁶ Sun and Moon ([Montenbruck & Gill, 2000](#); [D. A. Vallado, 2001](#)). It also provides standard
¹⁷ orbit propagation algorithms, including the Simplified General Perturbations (SGP) Model ([D.](#)
¹⁸[Vallado et al., 2006](#)). Finally, it implements recent algorithms for fast, parallelized computation
¹⁹ of ground station and imaging-target visibility ([Eddy & Kochenderfer, 2021](#)), a foundational
²⁰ problem in satellite scheduling and mission planning.

²² With [brahe](#), predicting upcoming satellite passes over ground stations or imaging targets can
²³ be accomplished in seconds and three lines of code.

```
24
25 import brahe as bh
26 bh.initialize_eop()
27 passes = bh.location_accesses(
28     bh.PointLocation(-122.4194, 37.7749, 0.0), # San Francisco
29     bh.celestrak.get_tle_by_id_as_propagator(25544, 60.0, "active"), # ISS
30     bh.Epoch.now(),
31     bh.Epoch.now() + 24 * 3600.0, # Next 24 hours
32     bh.ElevationConstraint(min_elevation_deg=10.0)
33 )
34
35
```

³⁷ [brahe](#) allows users to quickly access Two-Line Element (TLE) data from Celestrak ([Kelso, T.](#)
³⁸ [S., 2025](#)) and propagate orbits using the SGP4 dynamics model. This can be used to perform
³⁹ space situational awareness tasks such as predicting the orbits of all Starlink satellites over the
⁴⁰ next 24 hours.

```

41
42
43 import brahe as bh
44 bh.initialize_eop()
45 starlink = bh.datasets.celestrak.get_tles_as_propagators("starlink", 60.0)
46 for sat in starlink:
47     sat.propagate_to(sat.epoch + 86400.0) # Propagate one orbit (24 hours)
48

```

50 The above routine can propagate orbits for all ~9000 Starlink satellites in approximately 5
 51 minutes on an M1 Max MacBook Pro with 10 cores and 64 GB RAM. Finally, the package
 52 provides direct, easy-to-use functions for low-level astrodynamics routines such as Keplerian to
 53 Cartesian state conversions and reference frame transformations.

```

54
55 import brahe as bh
56 import numpy as np
57
58 # Initialize Earth Orientation Parameter data
59 bh.initialize_eop()
60
61 # Define orbital elements
62 a = bh.constants.R_EARTH + 700e3 # Semi-major axis in meters (700 km altitude)
63 e = 0.001 # Eccentricity
64 i = 98.7 # Inclination in radians
65 raan = 15.0 # Right Ascension of Ascending Node in radians
66 arg_periapsis = 30.0 # Argument of Periapsis in radians
67 mean_anomaly = 45.0 # Mean Anomaly
68
69 # Create a state vector from orbital elements
70 state_kep = np.array([a, e, i, raan, arg_periapsis, mean_anomaly])
71
72 # Convert Keplerian state to ECI coordinates
73 state_eci = bh.state_osculating_to_cartesian(state_kep, bh.AngleFormat.DEGREES)
74
75 # Define a time epoch
76 epoch = bh.Epoch(2024, 6, 1, 12, 0, 0.0, time_system=bh.TimeSystem.UTC)
77
78 # Convert ECI coordinates to ECEF coordinates at the given epoch
79 state_ecef = bh.state_eci_to_ecef(epoch, state_eci)
80
81 # Convert back from ECEF to ECI coordinates
82 state_eci_2 = bh.state_ecef_to_eci(epoch, state_ecef)
83
84 # Convert back from ECI to Keplerian elements
85 state_kep_2 = bh.state_cartesian_to_osculating(state_eci_2, bh.AngleFormat.DEGREES)
86
87
88

```

Statement of Need

89 While the core algorithms for predicting and modeling satellite motion have been known for
 90 decades, there is a lack of modern, open-source software that implements these algorithms
 91 in a way that is accessible to researchers and engineers. Generally, existing astrodynamics
 92 software packages have one or more barriers to entry for individuals and organizations looking
 93 to develop astrodynamics applications, and often leads to duplicated and redundant effort as
 94 researchers and engineers are forced to re-implement foundational algorithms.

95 Flagship commercial astrodynamics software like Systems Tool Kit (STK) ([Analytic Graphics, 2023](#)) and FreeFlyer ([a.i. Solutions, Inc., 2025](#)) are individually licensed and closed-source. The
 96 licensing costs can be prohibitive for researchers, individuals, small organizations, and start-ups.

Even for larger organizations, the per-node licensing cost can make large-scale deployment prohibitive. The closed-source nature of these packages makes it difficult to understand and verify the exact algorithms and model implementations, which is critical for high-stakes applications like space mission operations (Board, 1999). Major open-source projects like Orekit (Maisonobe et al., 2010) and GMAT (Hughes et al., 2014) provide extensive functionality, but are large codebases with steep learning curves, making quick-adoption and integration into projects difficult. Furthermore, Orekit is implemented in Java, which can be a barrier to adoption in the current scientific ecosystem with users who are more familiar with Python. GMAT uses a domain-specific scripting language and has limited documentation and examples, making it difficult for new users to get started. Finally, there are academic libraries such as poliastro (Cano Rodriguez & Martínez Garrido, 2022) which are not actively maintained. Other tools like Basilisk (Kenneally et al., 2020), provide high-fidelity modeling capabilities for full spacecraft guidance, navigation, and control (GNC) simulations, but are not directly distributed through standard package managers like PyPI and must be compiled from source to be used. Finally, academic work often has limited documentation and usage examples, making it difficult for new users to get started.

brahe seeks to address these challenges by providing a modern, open-source astrodynamics library following design principles of the *Zen of Python* (Peters, 2004). The core functionality is implemented in Rust for performance and safety, with Python bindings for ease-of-use and integration with the scientific Python ecosystem. *brahe* is provided under an MIT License to encourage adoption and facilitate integration and extensibility. To further promote adoption and aid user learning, the library is extensively documented following the Diátaxis framework (Procida, 2024)—every Rust and Python function documented with types and usage examples, there is a user guide that explains the major concepts of the library, and set of longer-form examples demonstrating how to accomplish common tasks. To maintain high code quality, the library has a comprehensive test suite for both Rust and Python. Additionally, all code samples in the documentation are automatically tested to ensure they remain functional, and that the documentation accurately reflects the library's capabilities.

brahe has already been used in a number of scientific publications (Eddy et al., 2025; Kim et al., 2025). It has also been used by aerospace companies such as Northwood Space, Xona Space (Reid et al., 2020), and Kongsberg Satellite Services for mission analysis and planning. The Earth Observation satellite imaging prediction and task planning algorithms have been used by Capella Space and demonstrated on-orbit with their synthetic aperture radar (SAR) constellation (Stringham et al., 2019).

Acknowledgments

We also want to acknowledge Shaurya Luthra, Adrien Perkins, and Arthur Kvalheim Merlin for supporting the adoption of the project in their organizations and providing valuable feedback. Finally, we would like to thank the Stanford Institute for Human-Centered AI for funding in part this work.

References

- 139 a.i. Solutions, Inc. (2025). *FreeFlyer: Spacecraft Mission Analysis and Operations Software*
140 (Version 7.10). <https://www.ai-solutions.com/freeflyer>
- 141 Analytic Graphics, Inc. (AGI). (2023). *Systems Tool Kit (STK)* (Version 12.7). <https://www.agi.com/products/stk>
- 143 Board, M. C. O. M. I. (1999). *Mars Climate Orbiter Mishap Investigation Board Phase I Report* [Tech. Report]. Jet Propulsion Laboratory / National Aeronautics and Space Administration. https://llis.nasa.gov/llis_lib/pdf/1009464main1_0641-mr.pdf

- 146 Cano Rodriguez, J. L., & Martínez Garrido, J. (2022). *poliastro* (Version v0.17.0). <https://github.com/poliastro/poliastro/>
- 147
- 148 Eddy, D., Ho, M., & Kochenderfer, M. J. (2025). Optimal Ground Station Selection for Low-
149 Earth Orbiting Satellites. *IEEE Aerospace Conference*. <https://arxiv.org/abs/2410.16282>
- 150 Eddy, D., & Kochenderfer, M. J. (2021). A Maximum Independent Set Method for Schedul-
151 ing Earth-Observing Satellite Constellations. *Journal of Spacecraft and Rockets*, 58(5),
152 1416–1429.
- 153 Hohenkerk, C. (2017). IAU Standards of Fundamental Astronomy (SOFA): Time and Date.
154 In *The Science of Time 2016: Time in Astronomy & Society, Past, Present and Future*.
155 Springer.
- 156 Hughes, S. P., Qureshi, R. H., Cooley, S. D., & Parker, J. J. (2014). Verification and
157 Validation of the General Mission Analysis Tool (GMAT). *AIAA/AAS Astrodynamics
158 Specialist Conference*.
- 159 Kelso, T. S. (2025). *Celestrak Active Satellite Database*. <https://celestak.com/>
- 160 Kenneally, P. W., Piggott, S., & Schaub, H. (2020). Basilisk: A Flexible, Scalable and
161 Modular Astrodynamics Simulation Framework. *Journal of Aerospace Information Systems*.
162 <https://doi.org/10.2514/1.I010762>
- 163 Kepler, J. (1619). *Epitome Astronomiae Copernicanae*.
- 164 Kim, G. R., Eddy, D., Srinivas, V., & Kochenderfer, M. J. (2025). Scalable Ground Station
165 Selection for Large LEO Constellations. *arXiv Preprint arXiv:2510.03438*. <https://arxiv.org/abs/2510.03438>
- 166
- 167 Maisonobe, L., Pommier, V., & Parraud, P. (2010). Orekit: An Open Source Library for
168 Operational Flight Dynamics Applications. *International Conference on Astrodynamics
169 Tools and Techniques*.
- 170 Montenbruck, O., & Gill, E. (2000). *Satellite Orbits: Models, Methods and Applications*.
171 Springer.
- 172 Newton, I. (1687). *Philosophiae Naturalis Principia Mathematica*.
- 173 Peters, T. (2004). The Zen of Python. In *Pro Python*. Springer.
- 174 Petit, G., & Luzum, B. (2010). *IERS Conventions, Technical Note 36*.
- 175 Procida, D. (2024). *Diátaxis: A Systematic Approach to Technical Documentation Authoring*.
176 <https://diataxis.fr/>
- 177 Reid, T. G., Chan, B., Goel, A., Gunning, K., Manning, B., Martin, J., Neish, A., Perkins, A.,
178 & Tarantino, P. (2020). Satellite Navigation for the Age of Autonomy. *2020 IEEE/ION
179 Position, Location and Navigation Symposium (PLANS)*.
- 180 Stringham, C., Farquharson, G., Castelletti, D., Quist, E., Riggi, L., Eddy, D., & Soenen, S.
181 (2019). The Capella X-band SAR Constellation for Rapid Imaging. *IEEE International
182 Geoscience and Remote Sensing Symposium*.
- 183 Vallado, D. A. (2001). *Fundamentals of Astrodynamics and Applications*.
- 184 Vallado, D., Crawford, P., Hujasak, R., & Kelso, T. S. (2006). Revisiting Spacetrack Report
185 #3. *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*.