# clustertools: A Python Package for Analyzing Star Cluster Simulations

Jeremy J. Webb ⓘ [1]

1 David A. Dunlap Department of Astronomy and Astrophysics, University of Toronto, 50 St. George Street, Toronto, ON, M5S 3H4, Canada

Summary

`clustertools` is a Python package for analyzing star cluster simulations. The package is built around the `StarCluster` class, which stores all data read in from the snapshot of a given model star cluster. The package contains functions for loading data from commonly used N-body codes, generic snapshots, and software for generating initial conditions. All operations and functions within `clustertools` are then designed to act on a `StarCluster`. `clustertools` can be used for unit and coordinate transformations, the calculation of key structural and kinematic parameters, analysis of the cluster's orbit and tidal tails, and measuring common cluster properties like its mass function, density profile, and velocity dispersion profile (among others). While originally designed with star clusters in mind, `clustertools` can be used to study other types of N-body systems, including stellar streams and dark matter sub-halos.

Statement of need

Stars do not form alone, but in clustered environments that in some cases can remain gravitationally bound as a star cluster for billions of years. The details of how exactly star clusters form, either at high-redshifts or at present day, remain unknown. After formation, the subsequent evolution of these stellar systems has been shown to be strongly linked to that of their host galaxy. Hence star clusters are often used as tools for studying star formation, galaxy evolution, and galaxy structure. The comparison of simulated star clusters to observations offers the ability to explore what formation conditions reproduce the properties of observed star cluster populations and how the evolution and structure of a galaxy affects star clusters.

A large number of N-body codes and software packages exist to generate star cluster models and simulate their long-term evolution in different environments (e.g. AMUSE, Portegies Zwart & McMillan, 2018; NBODY6, Aarseth, 2003; NBODY6++, Wang et al., 2015; NEMO, Teuben, 1995; PETAR, Wang et al., 2020). Additional software exists for generating model star clusters directly from a known distribution function (e.g. GALPY, Bovy, 2015; LIMEPY, Gieles & Zocchi, 2015; MCLUSTER, Küpper et al., 2011). The output from these codes and packages can differ in units, coordinate system, and format. The subsequent analysis of star cluster simulations and models can then be quite inhomogeneous between studies, as various methods exist for determining things like a cluster's centre, relaxation time, or tidal radius. `clustertools` allows for data from a star cluster simulation or model to be loaded and analyzed homogeneously by first loading a `StarCluster` instance. Within a `StarCluster`, stellar masses, positions, velocities, identification numbers, and all relevant meta-data from the source is saved. `clustertools` then contains an array of operations for converting units and coordinate systems, functions for calculating key cluster parameters, and functions for measuring how certain parameters vary with clustercentric distance. In some cases, multiple different methods are available for calculating a specific parameter. The software is particularly

---

useful to students or users new to star cluster studies, as functions and profile measurements have a plotting feature that helps illustrate how certain parameters are measured when possible. This approach also ensures that the analysis of any star cluster is done in a homogenous way with open-source code, regardless of the simulation or model from which that data was produced.

Acknowledgements

References

Aarseth, S. J. (2003). *Gravitational N-Body Simulations*.

Bovy, J. (2015). galpy: A python Library for Galactic Dynamics. *Astrophysical Journal, Supplement*, *216*(2), 29. https://doi.org/10.1088/0067-0049/216/2/29

Gieles, M., & Zocchi, A. (2015). A family of lowered isothermal models. *Monthly Notices of the RAS*, *454*(1), 576–592. https://doi.org/10.1093/mnras/stv1848

Küpper, A. H. W., Maschberger, T., Kroupa, P., & Baumgardt, H. (2011). Mass segregation and fractal substructure in young massive clusters - I. The McLuster code and method calibration. *Monthly Notices of the RAS*, *417*(3), 2300–2317. https://doi.org/10.1111/j.1365-2966.2011.19412.x

Portegies Zwart, S., & McMillan, S. (2018). *Astrophysical Recipes; The art of AMUSE*. https://doi.org/10.1088/978-0-7503-1320-9

Teuben, P. (1995). The Stellar Dynamics Toolbox NEMO. In R. A. Shaw, H. E. Payne, & J. J. E. Hayes (Eds.), *Astronomical data analysis software and systems IV* (Vol. 77, p. 398).

Wang, L., Iwasawa, M., Nitadori, K., & Makino, J. (2020). PETAR: a high-performance N-body code for modelling massive collisional stellar systems. *Monthly Notices of the RAS*, *497*(1), 536–555. https://doi.org/10.1093/mnras/staa1915

Wang, L., Spurzem, R., Aarseth, S., Nitadori, K., Berczik, P., Kouwenhoven, M. B. N., & Naab, T. (2015). NBODY6++GPU: ready for the gravitational million-body problem. *Monthly Notices of the RAS*, *450*(4), 4070–4080. https://doi.org/10.1093/mnras/stv817