

binary_c-python: A python-based stellar population synthesis tool and interface to binary_c

D. D. Hendriks¹ and R. G. Izzard¹

¹ Department of Physics, University of Surrey, Guildford, GU2 7XH, Surrey, UK

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Published: unpublished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](#)).

Summary

We present our package [binary_c-python](#), which is aimed to provide a convenient and easy-to-use interface to the [binary_c](#) ([Robert G. Izzard et al., 2018](#); [Robert G. Izzard et al., 2004](#); [R. G. Izzard et al., 2006](#); [R. G. Izzard et al., 2009](#)) framework, allowing the user to rapidly evolve individual systems and populations of stars. [binary_c-python](#) is available on [Pip](#) and on [Gitlab](#).

The user can control output from [binary_c](#) by providing [binary_c-python](#) with logging statements that are dynamically compiled and loaded into [binary_c](#). [binary_c-python](#) uses multiprocessing to utilise all the cores on a particular machine, and can run populations with HPC cluster workload managers like HTCondor and Slurm, allowing the user to run simulations on very large computing clusters.

[binary_c-python](#) is easily interfaced or integrated with other Python-based codes and libraries, e.g. sampling codes like Emcee or Dynesty, or the astrophysics oriented package Astropy ([Astropy Collaboration et al., 2018](#); [Astropy Collaboration et al., 2013](#); [Foreman-Mackey et al., 2013](#); [Speagle, 2020](#)). Moreover, it is possible to provide custom system-generating functions through our function hooks, allowing third-party packages to manage the properties of the stars in the populations and evolve them through [binary_c-python](#).

Recent developments in [binary_c](#) include standardised output datasets called *ensembles*. [binary_c-python](#) easily processes these datasets and provides a suite of utility functions to handle them. Furthermore, [binary_c](#) now includes the *ensemble-manager* class, which makes use of the core functions and classes of [binary_c-python](#) to evolve a grid of stellar populations with varying input physics, allowing for large, automated parameter studies through a single interface.

We provide [documentation](#) that is automatically generated based on docstrings and a suite of Jupyter [notebooks](#). These notebooks consist of technical tutorials on how to use [binary_c-python](#), and use-case scenarios aimed at doing science. Much of [binary_c-python](#) is covered by unit tests to ensure reliability and correctness, and the test coverage is continually increased as the package is being improved.

Statement of need

In the current scientific climate Python is ubiquitous, and while lower-level codes written in, e.g., Fortran or C are still widely used, much of the newer software is written in Python, either entirely or as a wrapper around other codes and libraries. Education in programming also often includes Python courses because of its ease of use and its flexibility. Moreover, Python has a large community with many resources and tutorials. We have created [binary_c-python](#) to

*E-mail: dh00601@surrey.ac.uk (DDH)

allow students and scientists alike to explore current scientific issues while enjoying the familiar syntax, and at the same time make use of the plentiful scientific and astrophysical packages like Numpy, Scipy, Pandas, Astropy and platforms like Jupyter.

Earlier versions of `binary_c-python` were written in Perl, where much of the logic and structure were developed and debugged. This made porting to Python relatively easy.

Projects that use `binary_c-python`

`binary_c-python` has already been used in a variety of situations, ranging from pure research to educational purposes, as well as in outreach events. In the summer of 2021 we used `binary_c-python` as the basis for the interactive classes on stellar ecosystems during the [International Max-Planck Research School summer school 2021 in Heidelberg](#), where students were introduced to the topic of population synthesis and were able to use our notebooks to perform their own calculations. `binary_c-python` has been used in Mirouh et al. ([submitted](#)), where improvements to tidal interactions between stars were implemented, and initial birth parameter distributions were varied to match to observed binary systems in star clusters. A Master's thesis project, aimed at finding the birth system parameters of the V106 stellar system, comparing observations to results of `binary_c` and calculating the maximum likelihood with Bayesian inference through Markov chain Monte Carlo sampling. The project made use of `binary_c-python` and the Emcee package.

Currently `binary_c-python` is used in several ongoing projects that study the effect of birth distributions on the occurrence of carbon-enhanced metal-poor (CEMP) stars, the occurrence and properties of accretion disks in main-sequence stars and the predicted observable black hole distribution by combining star formation and metallicity distributions with the output of `binary_c`. Moreover, we use the *ensemble* output structure to generate datasets for galactic chemical evolution on cosmological timescales, where we rely heavily on the utilities of `binary_c-python`.

Acknowledgements

We acknowledge the helpful discussions and early testing efforts from M. Delorme, G. Mirouh, and D. Tracey, and the early work of J. Andrews which inspired our Python-C interface code. DDH thanks the UKRI/UoS for the funding grant H120341A. RGI thanks STFC for funding grants [ST/R000603/1](#) and [ST/L003910/2](#).

References

- Astropy Collaboration, Price-Whelan, A. M., Sipőcz, B. M., Günther, H. M., Lim, P. L., Crawford, S. M., Conseil, S., Shupe, D. L., Craig, M. W., Dencheva, N., Ginsburg, A., VanderPlas, J. T., Bradley, L. D., Pérez-Suárez, D., de Val-Borro, M., Aldcroft, T. L., Cruz, K. L., Robitaille, T. P., Tollerud, E. J., ... Astropy Contributors. (2018). The Astropy Project: Building an Open-science Project and Status of the v2.0 Core Package. *The Astronomical Journal*, 156, 123. <https://doi.org/10.3847/1538-3881/aabc4f>
- Astropy Collaboration, Robitaille, T. P., Tollerud, E. J., Greenfield, P., Droettboom, M., Bray, E., Aldcroft, T., Davis, M., Ginsburg, A., Price-Whelan, A. M., Kerzendorf, W. E., Conley, A., Crighton, N., Barbary, K., Muna, D., Ferguson, H., Grollier, F., Parikh, M. M., Nair, P. H., ... Streicher, O. (2013). Astropy: A community Python package for astronomy. *Astronomy and Astrophysics*, 558, A33. <https://doi.org/10.1051/0004-6361/201322068>
- Foreman-Mackey, D., Hogg, D. W., Lang, D., & Goodman, J. (2013). Emcee: The MCMC Hammer. *Publications of the Astronomical Society of the Pacific*, 125(925), 306–312. <https://doi.org/10.1086/670067>

- 84 Izzard, R. G., Dray, L. M., Karakas, A. I., Lugaro, M., & Tout, C. A. (2006). Population
85 nucleosynthesis in single and binary stars - I. Model. *Astronomy & Astrophysics*, 460(2),
86 565–572. <https://doi.org/10.1051/0004-6361:20066129>
- 87 Izzard, R. G., Glebbeek, E., Stancliffe, R. J., & Pols, O. R. (2009). Population synthesis of
88 binary carbon-enhanced metal-poor stars. *Astronomy and Astrophysics*, 508(3), 1359–1374.
89 <https://doi.org/10.1051/0004-6361/200912827>
- 90 Izzard, Robert G., Preece, H., Jofre, P., Halabi, G. M., Masseron, T., & Tout, C. A. (2018).
91 Binary stars in the Galactic thick disc. *Monthly Notices of the Royal Astronomical Society*,
92 473, 2984–2999. <https://doi.org/10.1093/mnras/stx2355>
- 93 Izzard, Robert G., Tout, C. A., Karakas, A. I., & Pols, O. R. (2004). A new synthetic model
94 for asymptotic giant branch stars. *Monthly Notices of the Royal Astronomical Society*,
95 350(2), 407–426. <https://doi.org/10.1111/j.1365-2966.2004.07446.x>
- 96 Mirouh, G. M., Hendriks, D. D., Dykes, S., Moe, M., & Izzard, R. G. (submitted). Detailed
97 equilibrium and dynamical tides: impact on circularization and synchronization in open
98 clusters.
- 99 Speagle, J. S. (2020). Dynesty: A dynamic nested sampling package for estimating Bayesian
100 posteriors and evidences. *Monthly Notices of the Royal Astronomical Society*, 493(3),
101 3132–3158. <https://doi.org/10.1093/mnras/staa278>

DRAFT