phasespace: n-body phase space generation in Python

Albert Puig Navarro[1] **and Jonas Eschle**[1]

1 Physik-Institut, Universität Zürich, Zürich (Switzerland)

DOI: 10.21105/joss.01570

Software

- Review
- Repository
- Archive

Submitted: 03 June 2019
Published: 29 October 2019

Summary

Simulated particle decays are common in experimental particle physics. They are used to study a wide variety of aspects of a physics analysis, such as signal response, detector effects, and the efficiency of selection requirements, in a controlled manner. While it is possible to encode complex physics dynamics into these simulations at the cost of increased complexity and larger computer requirements, in many cases it is enough to generate these simulated samples as if only kinematic physics occurred, i.e., in an isotropic way. This type of generation, called "phase space generation", is very fast and offers simple and predictable patterns, making it an attractive first step in many physics analyses.

The `phasespace` package implements phase space event generation based on the Raubold and Lynch method described in (James, 1968). This method was previously implemented in the `GENBOD` function of the FORTRAN-based `CERNLIB` library. It was posteriorly ported to C++ for the ROOT toolkit (Brun & Rademakers, 1997) as the `TGenPhaseSpace` class, which is currently the most used implementation in particle physics. The `phasespace` package provides a pure Python implementation of the Raubold and Lynch method using the *Tensorflow* platform (Abadi et al., 2015) as its computational backend. Unlike `TGenPhaseSpace`, the `phasespace` approach offers seamless integration with the scientific Python ecosystem (*numpy*, *pandas*, *scikit-learn*...) while at the same time provides excellent performance and scalability both in CPUs and GPUs thanks to *Tensorflow*.

In addition, `phasespace` allows the generation of complex multi-decay chains, including non-constant masses as is needed for the simulation of resonant particles. This functionality opens the door for its use as the basis for importance sampling in Dalitz and amplitude decay fitters, which typically need to implement their own solution based on `TGenPhaseSpace`; in this sense, `phasespace` is currently being used for the implementation of amplitude fit sampling in the `zfit` fitter (Eschle, Puig Navarro, & Silva Coutinho, 2019).

The correctness of `phasespace` is continuously validated through its test suite against `TGenPhaseSpace` and the `RapidSim` package (Cowan, Craik, & Needham, 2017), an application for the simulation of heavy-quark hadron decays; this latter application also uses `TGenPhaseSpace`, but adds features such as multi-decay chains and simulation of the kinematics found in colliders such as the LHC.

In summary, `phasespace` is designed to fill an important gap in the recent paradigm shift of particle physics analysis towards integration with the scientific Python ecosystem. To do so it also has more advanced functionality than its C++-based predecessors. With its ease of use, clear interface and direct interoperability with other packages, `phasespace` provides a solid foundation to build upon in the quest for a full Python-based particle physics analysis software stack. The source code for `phasespace` has been archived to Zenodo with the linked DOI: (Puig Navarro & Eschle, 2019).

Acknowledgements

A.P. acknowledges support from the Swiss National Science Foundation under contract 168169.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., et al. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Retrieved from https://www.tensorflow.org/

Brun, R., & Rademakers, F. (1997). ROOT: An object oriented data analysis framework. *Nucl. Instrum. Meth.*, *A389*, 81–86. doi:10.1016/S0168-9002(97)00048-X

Cowan, G. A., Craik, D. C., & Needham, M. D. (2017). RapidSim: an application for the fast simulation of heavy-quark hadron decays. *Comput. Phys. Commun.*, *214*, 239–246. doi:10.1016/j.cpc.2017.01.029

Eschle, J., Puig Navarro, A., & Silva Coutinho, R. (2019). zfit: scalable pythonic fitting. doi:10.5281/zenodo.2602043

James, F. (1968). Monte-Carlo phase space.

Puig Navarro, A., & Eschle, J. (2019). Phasespace: N-body phase space generation in python. doi:10.5281/zenodo.2591993