# PyKEP 3: A coolbox for interplanetary trajectory design

**Dario Izzo** [1,]

**1** European Space Agency's Advanced Concepts Team, The Netherlands **2** European Space Agency's Advanced Concepts Team, The Netherlands

## Summary

PyKEP 3 is a Python toolbox developed at the European Space Agency by the Advanced Concepts Team to perform quick analysis of interplanetary trajectory design problems. It is designed to be used by researchers and engineers to prototype and test new ideas in the field of astrodynamics. The library provides efficient implementations of algorithms for solving the multiple revolutions Lambert's problem, low-thrust problems, multiple asteroid rendezvous problems, and more. It also provides support for JPL SPICE, SGP4 propagation, and the Heyoka Taylor integration suite.

## Introduction

PyKEP 3 is the third iteration of the PyKEP library, which has been used by the Advanced Concepts Team at the European Space Agency for over a decade. The library was originally developed to support the participation of non astrodynamics experts in the Global Trajectory Optimization Competition and has since grown into a generic tool for the preliminary analysis of interplanetary trajectory design problems. Previous versions of the code were used internally to perform some preliminary analysis for the design of the Hera mission, the M-ARGO interplanetary cubesat concept, the preliminary analysis of the the initial concepts for the Titan and Enceladus Mission (TandEM) as well as the Laplace mission. At the core of the library are the necessary building blocks to perform a preliminary design of interplanetary trajectory design problems, with the aim to provide the general scientific community with a tool to prototype and test new ideas in the field of astrodynamics focussing on novel ideas rather than on on the implementation of known algorithms. Most techniques implemented in the library are largely based on original research performed at the ACT, as well as on the state-of-the art algorithms available from the literature. A Lambert solver as well as low-thrust optimization algorithms (both based on direct and indirect methods) are available together with a number of approximate methods to estimate transfer costs. The library interfaces to a number of open source projects providing relevant functionalities including spicepy for interfacing to JPL SPICE, SGP4 to get standard Low Earth Orbit orbits, Heyoka to perform efficient Taylor integration and pygmo to interface to state-of-the art optimization algorithms.

## Mathematics

Single dollars ($) are required for inline mathematics e.g. $f(x) = e^{\pi/x}$

Double dollars make self-standing equations:

$$\Theta(x) = \left\{ \begin{array}{l} 0 \text{ if } x < 0 \\ 1 \text{ else} \end{array} \right.$$

37  You can also use plain LaTeX for equations

$$\hat{f}(\omega) = \int_{-\infty}^{\infty} f(x)e^{i\omega x}dx \tag{1}$$

38  and refer to Equation 1 from text.

## Citations

40  Citations to entries in paper.bib should be in rMarkdown format.

41  If you want to cite a software repository URL (e.g. something on GitHub without a preferred
42  citation) then you can do it with the example BibTeX entry below for Smith et al. (2020).

43  For a quick reference, the following citation commands can be used: - `@author:2001` ->
44  "Author et al. (2001)" - `[@author:2001]` -> "(Author et al., 2001)" - `[@author1:2001;`
45  `@author2:2001]` -> "(Author1 et al., 2001; Author2 et al., 2002)"

## Figures

47  Figures can be included like this: Caption for example figure. and referenced from text using
48  section .

49  Figure sizes can be customized by adding an optional second parameter: Caption for example
50  figure.

## Acknowledgements

52  We acknowledge contributions from …. and support from …

## References

54  Smith, A. M., Thaney, K., & Hahnel, M. (2020). Fidgit: An ungodly union of GitHub and
55      figshare. In *GitHub repository*. GitHub. https://github.com/arfon/fidgit