plasmapy

An open source software ecosystem for plasma physics

Nicholas Murphy,[1] Erik Everson,[2] Dominik Stańczak,[3] Ramiz Qudsi,[4,5] Elliot Johnson,[5] Shane Brown,[5] Haman Bagherianlemraski,[6] David Schaffner,[7] Stephen Vincena,[2] & Bennett Maruca[5] on behalf of the PlasmaPy Community

[1] Center for Astrophysics | Harvard & Smithsonian, [2] UCLA, [3] IPPLM, [4] Boston University, [5] University of Delaware, [6] University of Massachusetts Amherst, [7] Bryn Mawr College

We acknowledge support from:

NSF CSSI awards 1931388, 1931393, 1931429, and 1931435

astropy-powered
astropy.org

python powered

What is PlasmaPy?

plasmapy

Mission

*To grow an open source **software ecosystem** for plasma research & education*

https://www.plasmapy.org/

Building a software ecosystem

- **PlasmaPy core package**
  - Most frequently needed functionality
  - Currently under active development (version 0.8.1 just released!)
- **Affiliated packages**
  - Will contain specialized functionality
  - To be created by broader community
- **Educational resources**
  - Introduce plasma concepts using PlasmaPy
- **Community**
  - Informal online community meetings & virtual “office” hours
  - Code of conduct

Anticipated benefits for research

- Well-documented & well-tested code
- Reliable code with validated physics
- Improved interoperability
- Less duplication of functionality
- Lower code development costs
- Open, reproducible, and efficient research
- Community-driven development

Anticipated benefits for education

- Reduce barriers to entry
- Improve transfer of knowledge
  - Documentation describing code & physics
- Provide tools for plasma education
  - Introduce plasma concepts using PlasmaPy
  - Classroom exercises
- Introduce collaborative code development practices
  - Provide students with skills beyond the classroom

What's new in PlasmaPy 0.8.1

- New capabilities
  - Thomson scattering diagnostic modeling functions (credit: P. Heuer)
  - Plasma calculator prototype (credit: R. Gangadharan)
  - Null point finder prototype (credit: H. Bagherianlemraski)
  - Lite-functions for performance-critical applications (credit: E. Everson)
- Educational notebooks
  - Plasma β in the solar atmosphere
  - Plasma parameters in Earth's magnetosphere
  - Coulomb logarithms
- Adoption of Contributor Covenant Code of Conduct v2.1
- Improvements to documentation & testing infrastructure

PlasmaPy subpackages

`plasmapy.particles`

- Object-oriented & functional interfaces to information on ions, electrons, and fundamental particles

`plasmapy.formulary`

- Commonly needed formulas for plasma parameters and transport coefficients

`plasmapy.simulation`

- To include building blocks of plasma simulations and a particle tracker

PlasmaPy subpackages

`plasmapy.analysis`

- Analysis techniques for data from simulations, experiments, and observations

`plasmapy.diagnostics`

- For representations of plasma diagnostics such as Langmuir probes & Thomson scattering, as well as synthetic diagnostics

`plasmapy.dispersion`

- For dispersion relation solvers for plasma waves & instabilities

`plasmapy.plasma`

- For base classes to represent different plasmas

Example subpackage: `plasmapy.particles`

`plasmapy.particles` provides functional and object-oriented access to basic particle data.

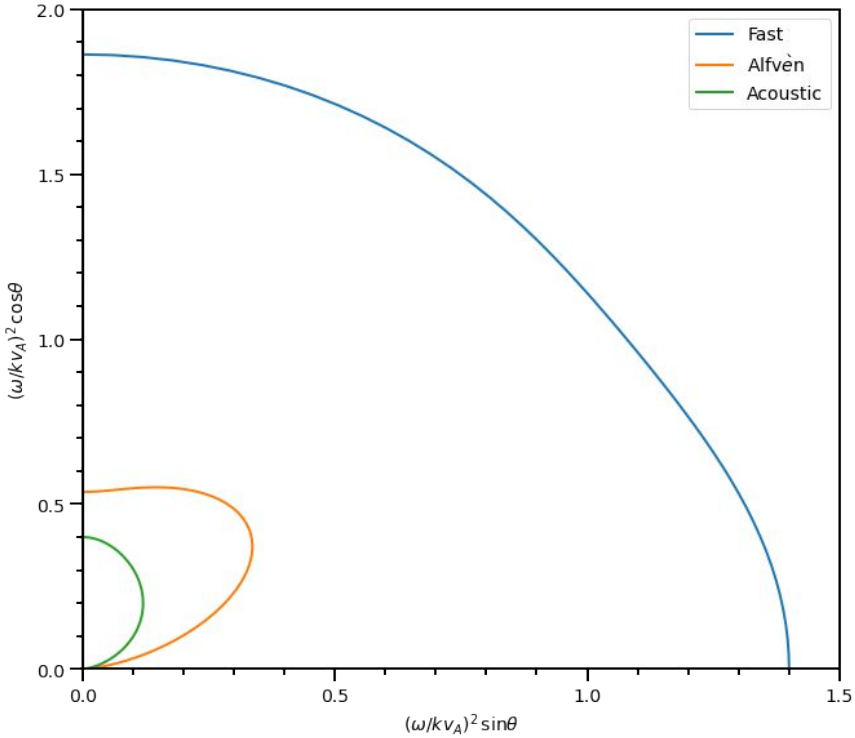
```
>>> from plasmapy.particles import Particle, ParticleList

>>> proton = Particle("p+")
>>> proton.mass
<Quantity 1.67262192e-27 kg>
>>> proton.is_category("baryon")
True

>>> helium_ions = ParticleList(["He-4 1+", "α"])
>>> helium_ions.charge
<Quantity [1.60217663e-19, 3.20435327e-19] C>
```

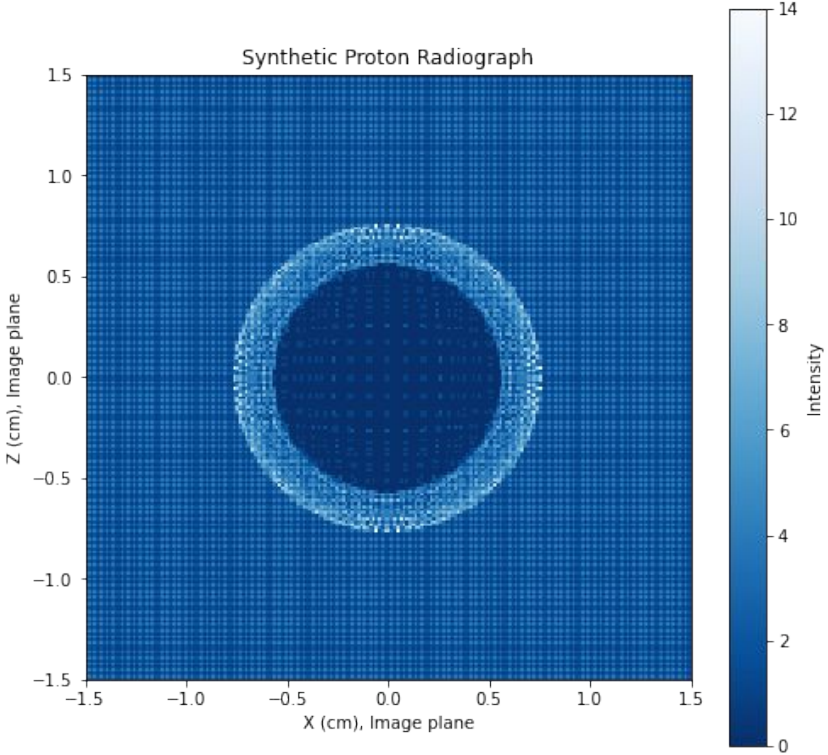
Two-fluid dispersion relation solver

- The plasmapy.dispersion subpackage contains the two-fluid dispersion relation solution from Bellan (2012) (credit: R. Qudsi)
- The Stix (1992) dispersion relation for cold plasma waves and the Hollweg (1999) dispersion relations were added in v0.8.1 (credit: S. Brown & E. Johnson)



New capabilities for high energy density plasmas

- Send protons organized in a grid through a laser-produced plasma
- Protons are deflected by electric and magnetic fields
- Synthetic images constrain the electromagnetic field in the exploding plasma
- Example Jupyter notebooks in PlasmaPy documentation

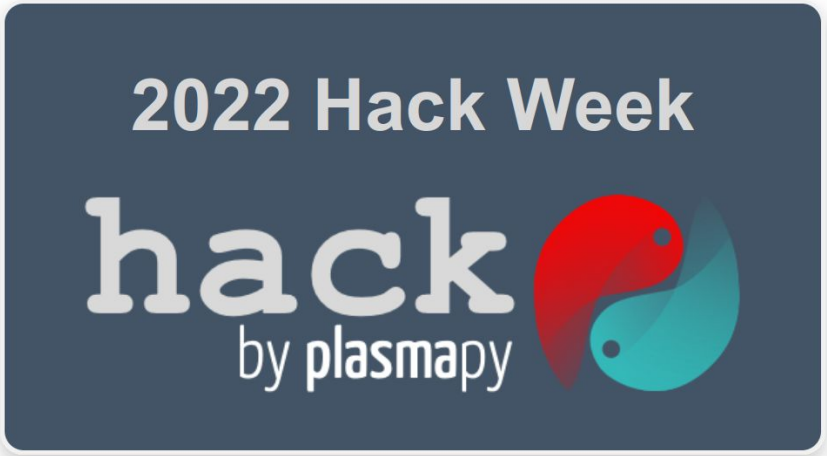


Credit: P. Heuer

Overhaul of contributor guide

- PlasmaPy's documentation has a contributor guide
  - Describes how to contribute, run/write tests, build documentation, & make releases
- The testing guide has been overhauled to include:
  - How (and why) to write & run PlasmaPy's test suite
  - Best practices for testing scientific software
- The documentation guide has been overhauled to include:
  - How to build documentation
  - Tools used to build the documentation
  - PlasmaPy's documentation guidelines

Plasma Hack Week (July 11–15, 2022)



- Mix of a summer school and a hackathon
- Tutorials on research software engineering
- Chance to learn how to contribute to open source projects

https://hack.plasmapy.org

Biggest challenges for PlasmaPy

- Building a community of users & contributors in a pandemic
- Motivating the plasma community to embrace open science
- Reducing the code review bottleneck
- Lack of a community-wide portal for plasma data sets
- Lack of metadata standardization

Many ways to be part of the community

- Come to PlasmaPy's weekly events
  - Office hours (Thursdays at 3 pm ET)
  - Community meeting (Tuesdays at 2 pm ET)
- Join our Element chat
- Request new features or share ideas on GitHub
- Organize next year's Plasma Hack Week
- Contribute!

Summary

- The PlasmaPy project's mission is to forge an open source software ecosystem for plasma research and education
- Version 0.8.1 released with many contributions from across the plasma community
- Overhaul of contributor guide to improve developer experience
- Plasma Hack Week was begun and held virtually in 2021 and 2022 as a community learning experience