SNEWPY: Overview

Jost Migenda
they/them

https://github.com/SNEWS2/snewpy

Agenda

- What's SNEWPY? Introduction
- What's new? Progress since 2021 collaboration meeting
- What's next? Upcoming changes

What is SNEWPY?

SNEWPY offers ...

- ... a simple and unified interface to hundreds of supernova simulations.

Can use these in your code!

- ... a large library of flavor transformations that relate neutrino fluxes produced in the supernova to those reaching a detector on Earth.

- ... and a Python interface to SNOwGLoBES which lets you estimate and plot event rates in many different neutrino detectors.

Usage of SNEWPY

- SNEWS-internal (see e.g. Marta's talk tomorrow)
- By other software:
  - sntools (DOI:10.21105/joss.02877)
  - ASTERIA (DOI:10.5281/zenodo.3926834)

smooth transition from quick initial estimates to advanced analyses

- In non-SNEWS papers:

Neutrino Echos following Black Hole Formation in Core-Collapse Supernovae

Samuel Gullin,[1] Evan P. O'Connor,[1] Jia-Shian Wang,[2] and Jeff Tseng[2]

[1] *The Oskar Klein Centre, Departme*
Stockholm University, AlbaNova, SE-106
[2] *Department of Physics, Oxford University, Oxfo*

arXiv:2109.13242

arXiv:2203.05141

Detectability of hadron-quark phase transition in neutrino signals of failing core-collapse supernova

Zidu Lin,[1] Shuai Zha,[2] Evan P. O'Connor,[3] and Andrew W. Steiner[1,4]
[1] *Department of Physics and Astronomy, University of Tennessee Knoxville*
[2] *Tsung-Dao Lee Institute, Shanghai Jiao Tong University, Shanghai 200240, China*
[3] *The Oskar Klein Centre, Department of Astronomy, Stockholm University, AlbaNova, SE-106 91 Stockholm, Sweden*
[4] *Physics Division, Oak Ridge National Laboratory*
(Dated: March 11, 2022)

What's New(ish) in SNEWPY?

- Two papers published in late 2021:
  - ApJ (describes underlying physics & usage) DOI:10.3847/1538-4357/ac350f
  - JOSS (review of code & infrastructure) DOI:10.21105/joss.03772
- New releases:
  - v1.1 (November 2021) *after JOSS review*
    - Infrastructure improvements (unit tests, docs, PyPI)
    - Added many new models & a downloader
    - New SNOwGLoBES interface
    - Bugfixes, code cleanup, etc.
  - v1.2 (January 2022)
    - Performance improvements for SNOwGLoBES interface
    - Initial SimpleRate calculator (see Sonia's talk!)
    - Initial support for preSN models
    - Bugfixes & other minor improvements
  - v1.2.1 (June 2022)
    - Compatibility and bugfixes

What's Next for SNEWPY?

- Lot of activity before & during the hackathon!
- Upcoming talks on some major features
  - Advanced flavor transformations (Jim's talk)
  - Simple rate calculation (Sonia's talk)
- Other new features:
  - Pre-SN neutrinos (Andrey's talk)
  - SNOwGLoBES v1.3 support (Sebastian's talk)
  - Updated mixing parameters (NuFIT 5.1, PDG2022)
  - Model registry ➔ next slides

What's Next for SNEWPY?

SNEWPY Model Registry

Lives on GitHub [SNEWPY/model_registry_jan22]

Model Usage (Front-end)

- Added model initialization from physics parameters
- Valid parameters and combinations in `model.param` & `model.param_combinations`
  - Simplifies iteration through models
- Previous initialization by file path still works, but uses a new `loader` class
- General `init_model` allows use of any model without additional imports

```
1 from snewpy.models.util import init_model
2 import astropy.units as u
3
4 param = {'progenitor_mass': 27*u.Msun,
5         'eos': 'LS220'}
6
7 init_model('Sukhbold_2015', **param)
```

Sukhbold_2015 Model: sukhbold-LS220-s27.0.fits

Parameter	Value
Progenitor mass	27 $M_\odot$
EOS	LS220

```
1 from snewpy.models.ccsn import Sukhbold_2015
2 print
3 for key, val in Sukhbold_2015.param.items():
4     print(f'{key:>16} {val}')
```

```
progenitor_mass [27.   9.6] solMass
            eos ['LS220', 'SFHo']
```

```
1 Sukhbold_2015.param_combinations
```

```
({'progenitor_mass': <Quantity 27. solMass>, 'eos': 'LS220'},
 {'progenitor_mass': <Quantity 27. solMass>, 'eos': 'SFHo'},
 {'progenitor_mass': <Quantity 9.6 solMass>, 'eos': 'LS220'},
 {'progenitor_mass': <Quantity 9.6 solMass>, 'eos': 'SFHo'})
```

```
1 from snewpy import model_path
2 from snewpy.models.loaders import Sukhbold_2015
3 import os
4 file_path = os.path.join(model_path,
5                          'Sukhbold_2015/sukhbold-LS220-s27.0.fits')
6 Sukhbold_2015(file_path)
```

Sukhbold_2015 Model: sukhbold-LS220-s27.0.fits

Slide from Spencer Griswold

What's Next for SNEWPY?

SNEWPY Model Registry

Lives on GitHub [SNEWPY/model_registry_jan22]

Need to change existing code!

Model Usage (Front-end)

- Added model initialization from physics parameters
- Valid parameters and combinations in `model.param` & `model.param_combinations`
  - Simplifies iteration through models
- Previous initialization by file path still works, but uses a new `loader` class
- General `init_model` allows use of any model without additional imports

```
from snewpy.models.util import init_model
import astropy.units as u

param = {'progenitor_mass': 27*u.Msun,
         'eos': 'LS220'}

init_model('Sukhbold_2015', **param)
```

Sukhbold_2015 Model: sukhbold-LS220-s27.0.fits

Parameter	Value
Progenitor mass	27 $M_\odot$
EOS	LS220

```
from snewpy.models.ccsn import Sukhbold_2015
print
for key, val in Sukhbold_2015.param.items():
    print(f'{key:>16} {val}')
```

```
progenitor_mass [27.   9.6] solMass
            eos ['LS220', 'SFHo']
```

```
Sukhbold_2015.param_combinations
```

```
({'progenitor_mass': <Quantity 27. solMass>, 'eos': 'LS220'},
 {'progenitor_mass': <Quantity 27. solMass>, 'eos': 'SFHo'},
 {'progenitor_mass': <Quantity 9.6 solMass>, 'eos': 'LS220'},
 {'progenitor_mass': <Quantity 9.6 solMass>, 'eos': 'SFHo'})
```

```
from snewpy import model_path
from snewpy.models.loaders import Sukhbold_2015
import os
file_path = os.path.join(model_path,
                         'Sukhbold_2015/sukhbold-LS220-s27.0.fits')
Sukhbold_2015(file_path)
```

Sukhbold_2015 Model: sukhbold-LS220-s27.0.fits

Slide from Spencer Griswold

What's Next for SNEWPY?

SNEWPY Model Registry

Storage and Downloads (Back-end)

- Official SNEWPY models are downloaded to new `snewpy/models` folder in `astropy` cache
- New `model_downloader` detects if official models are missing & automatically downloads from Github or Zenodo

Documentation & Unit Testing

- SNEWPY Docs are automatically populated with valid model parameters
- Unit test currently passing for (almost) all official models

```
1 from snewpy.models.util import init_model
2 import astropy.units as u
3
4 param = {'progenitor_mass': 27*u.Msun, 'eos': 'LS220'}
5
6 init_model('Sukhbold_2015', **param)
```

sukhbold-LS220-s27.0.fits: 100% 782k/782k [00:00<00:00, 6.33MiB/s]

Sukhbold_2015 Model: sukhbold-LS220-s27.0.fits

Parameter	Value
Progenitor mass	27 $M_\odot$
EOS	LS220

Slide from Spencer Griswold

Summary

- SNEWPY provides ...
  - ... a library of SN models
  - ... a library of flavour transformations
  - ... a Python interface to SNOwGLoBES

- Used both within SNEWS & in the wider community

- Modern code, under active development

- New ideas/contributors are welcome!

Interested?
Talk to me at dinner!