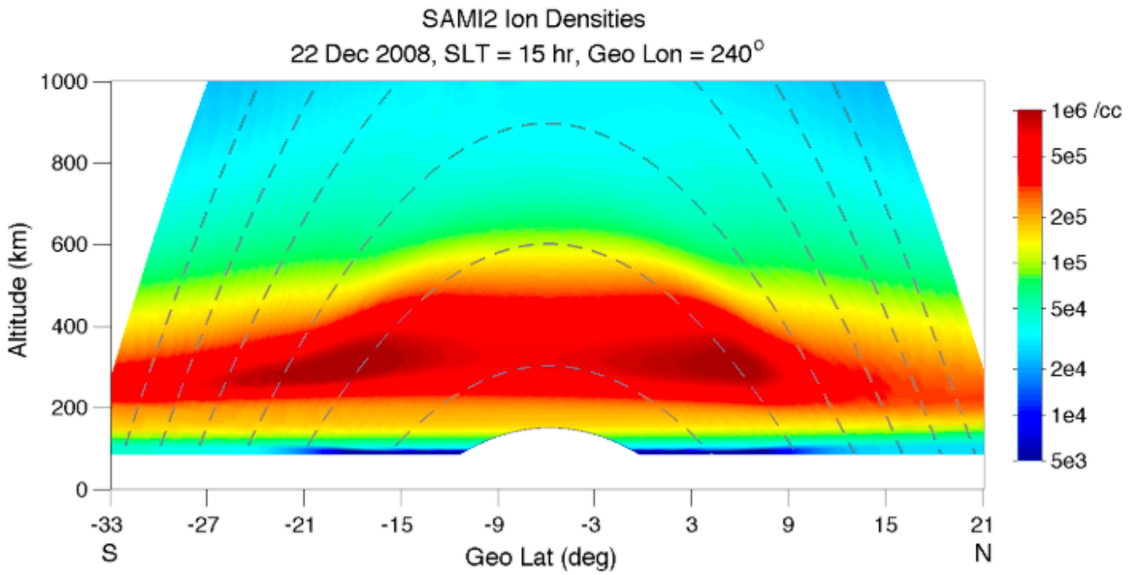
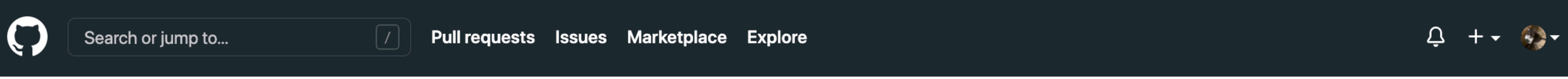
sami2py — Overview and Applications

Jeff Klenzing (NASA/GSFC)
CEDAR Workshop, June 21 2021

Background

- Sami2 is an open source 2D model of the low/mid-latitude ionosphere [Huba et al 2000]
- Previous user interface developed in Matlab (circa 2012)
- sami2py ports this code to a python interface
- Goal: Improved user access to altering the model





sami2py / sami2py

Unwatch 3 | Unstar 8 | Fork 5

Code | Issues 18 | Pull requests | Actions | Projects 5 | Wiki | Security | Insights | Settings

main | 10 branches | 7 tags | Go to file | Add file | Code

jklenzing Merge pull request #141 from sami2py/rc0.2.3 ... 3c0b866 5 days ago 884 commits

.github	TST: remove mac tests	17 days ago
docs	STY: version in docs	11 days ago
sami2py	STY: remove py2.7 imports	11 days ago
.codeclimate.yml	plugins for codeclimate	17 months ago
.coveragerc	test	2 years ago
.gitignore	added default exb file and changed gitignore to reflect the case of t...	10 months ago
.zenodo.json	DOC: update zenodo	20 days ago
CHANGELOG.md	DOC: update release date	5 days ago
CODE_OF_CONDUCT.md	Create CODE_OF_CONDUCT.md	2 years ago

About

Python wrapper to run, read, and plot the SAMI2 ionospheric model

python | ionosphere | ionosphere-modeling

Readme

BSD-3-Clause License

Releases 7

Version 0.2.3 Latest
5 days ago

+ 6 releases

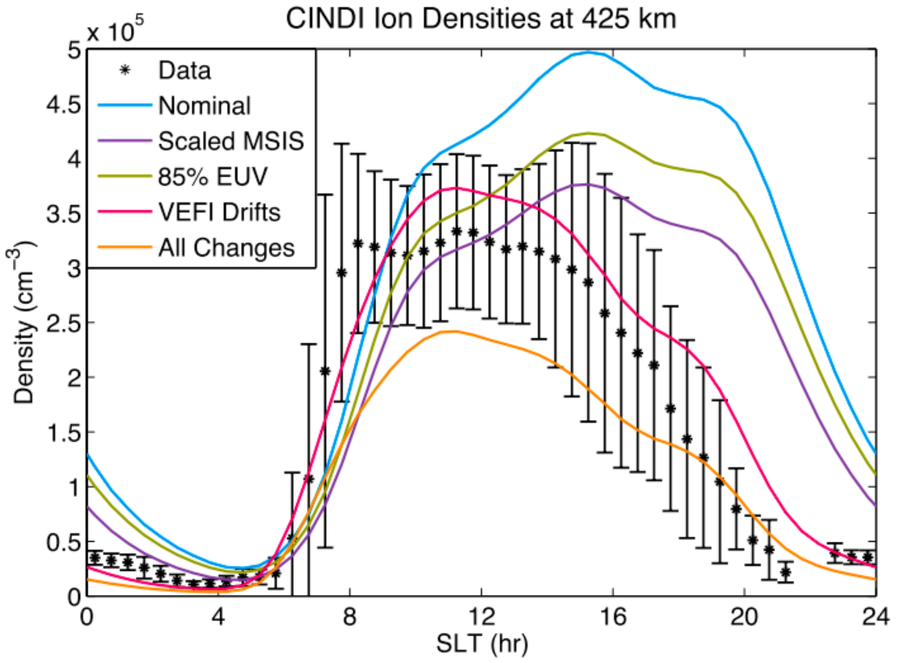
https://www.github.com/sami2py/sami2py

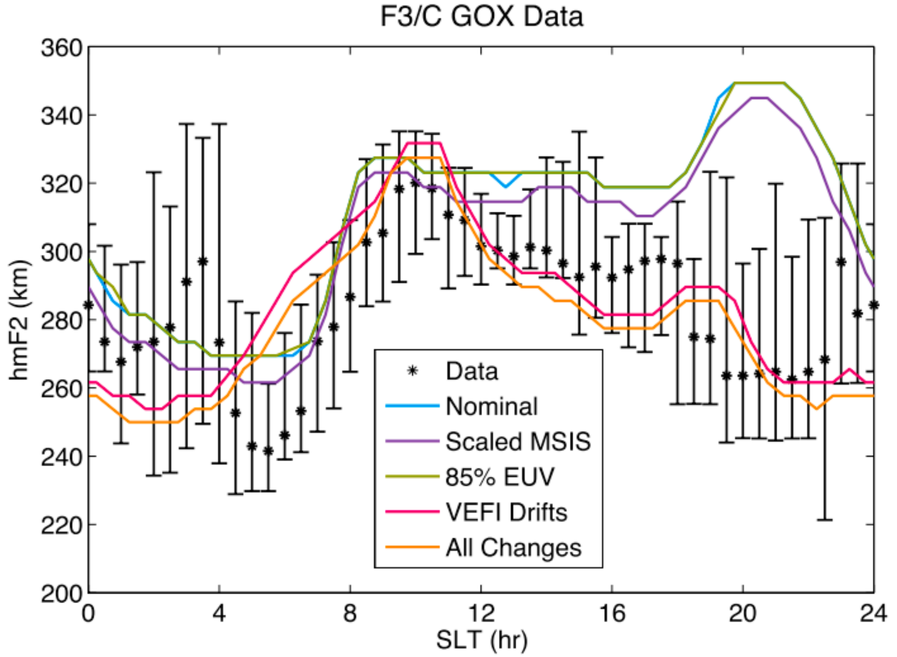
Welcome to sami2py's documentation!

https://sami2py.readthedocs.io/

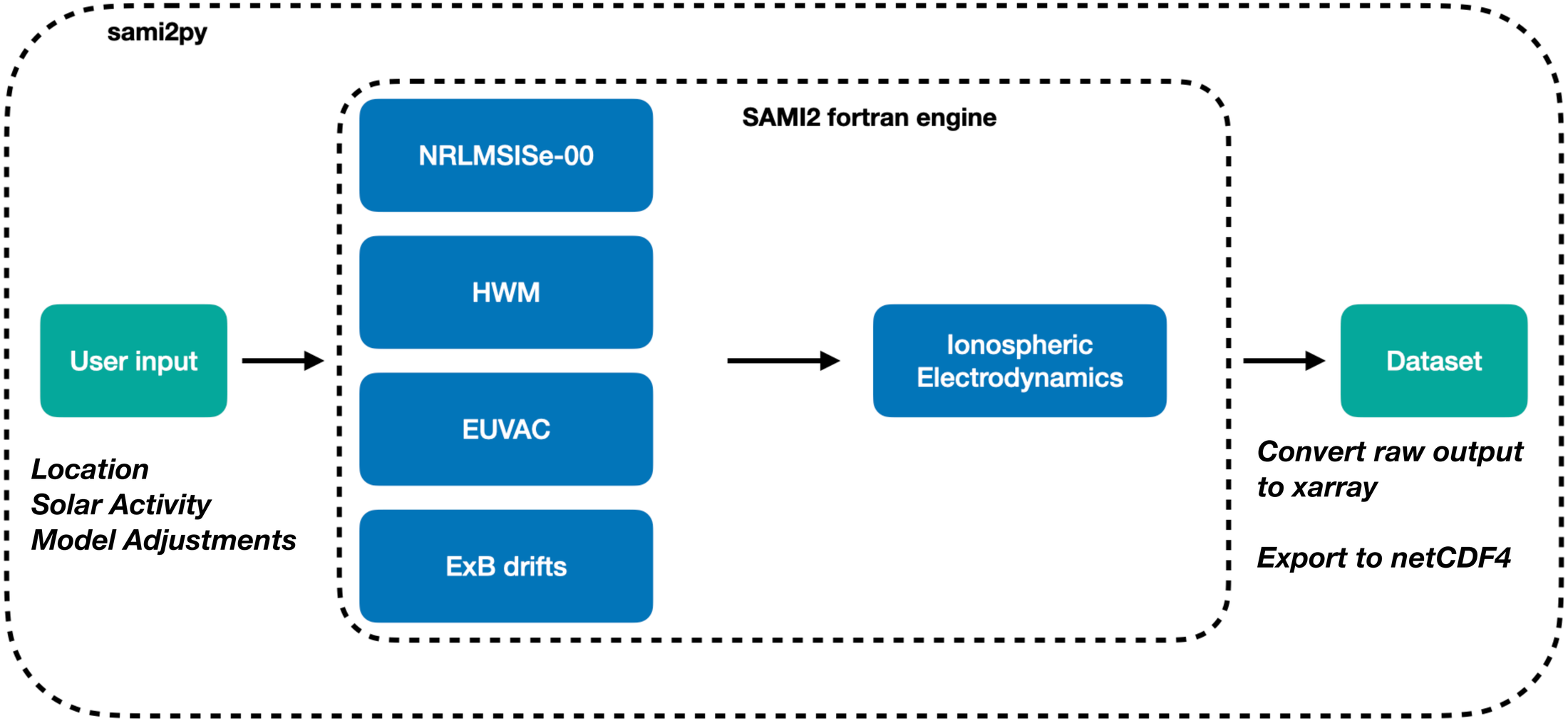
A sample application

- Investigate potential changes to the ionosphere under the extreme solar minimum between solar cycles 23 and 24 [Klenzing et al, 2013]
- Enable rapid prototype investigations of hypotheses
- For multiple runs, better archive and document the various model runs





Interface


sami2py
SAMI2 fortran engine
NRLMSISe-00
HWM
EUVAC
ExB drifts
User input
Location
Solar Activity
Model Adjustments
Ionospheric Electrodynamics
Dataset
Convert raw output to xarray
Export to netCDF4

Interface Rationale

- The current iteration uses a compiled fortran module run via `subprocess`.
- Inputs to the fortran model are set through modification of a namelist file. Sami2py provides direct access to these values via keywords.
- Raw output data files are moved to an archived data structure. The paths include a user-specified name for the run, as well as the date and longitude of the run.
  - EX: /data/sami2py/custom_run_name/lon256/1999_256

Partial sample of namelist

```
&go
  fmtout   = .true.,
  maxstep  =  100000000,
  hrmax    =  0.110000,
  dt0      =  30.000000,
  dthr     =  0.050000,
  hrpr     =  0.000000,
  grad_in  =  300.000000,
  glat_in  =  0.000000,
  glon_in  =  256.000000,
  fejer    =  .true.,
  rmin     =  100.000000,
  rmax     =  2000.000000,
  altmin   =  85.000000,
  fbar     =  120.000000,
  f10p7    =  120.000000,
  ap       =  0,
  year     =  1999,
  day      =  256,
```

SAMI2 - Overview

- SAMI2 uses a series of empirical models to drive the ionosphere
- Solves the continuity and momentum equations
- Produces ion density, composition, temperature, and parallel drift

Physical Mechanism	Model Name	Scalable Parameters
Neutral Atmosphere	NRLMSISe-00	Neutral Species, Exospheric Temperature
Photoionization Rate	EUVAC	Total Ionization
Neutral Winds	HWM-14 (default) HWM-07 HWM-93	Wind Magnitude
ExB drifts	Fejer-Scherliess (default) Fourier coefficients F(SLT)	Drift magnitude, offset from zero

Updated options in sami2py

Sample Workflow

- Set an archive directory (only needed the first time)
  - sami2py.utils.set_archive_dir(path='/data/sami2py/')
- Running a model
  - sami2py.run_model(tag='run_name', lon=0, year=2012, day=210)
- Load the data
  - ModelRun = sami2py.Model(tag='run_name', lon=0, year=2012, day=210)
- Export to netcdf
  - ModelRun.to_netcdf('your_filename.nc')

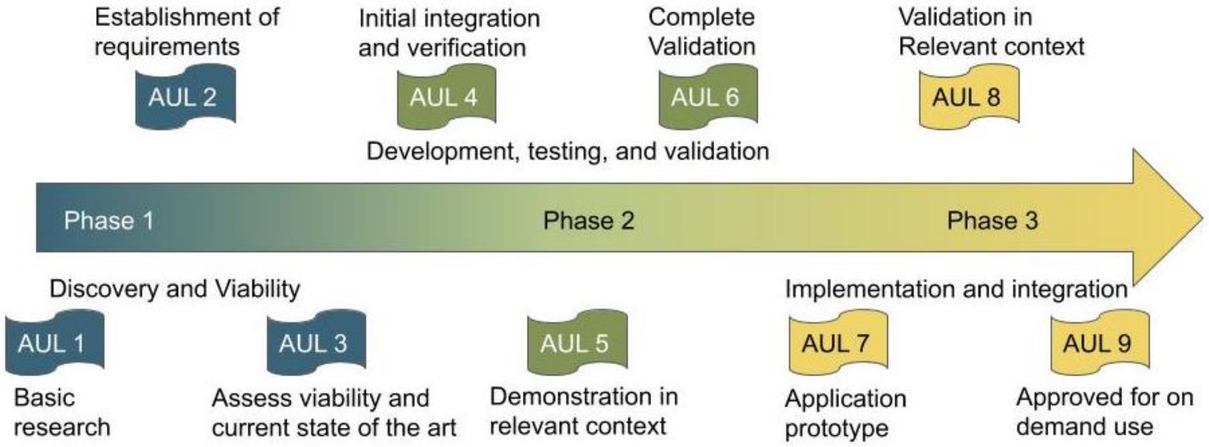
The Model class

- The Model class includes both data and metadata components
- Metadata can be accessed directly or via the __repr__ function
- Data is stored as an xarray object
- The built in `to_netcdf` function will package all metadata about the model run (including commit hash) to the netCDF4 attributes for better traceability

```
In [70]: model = sami2py.Model(tag='pysatmodels', year=2019, day=1, lon=254)

In [71]: model
Out[71]:

Model Run Name = pysatmodels
Day 001, 2019
Longitude = 254.0 deg
6 time steps from  0.0 to  1.3 UT
Ions Used: H+, O+, NO+, O2+, He+, N2+

Solar Activity
--------------
F10.7: 120.0 sfu
F10.7A: 120.0 sfu
ap: 0

Component Models Used
---------------------
Neutral Atmosphere: NRLMSISe-2000
Winds: HWM-14
Photoproduction: EUVAC
ExB Drifts: Fejer-Scherliess

No modifications to empirical models

In [72]: model.data
Out[72]:
<xarray.Dataset>
Dimensions:  (f: 98, ion: 7, ut: 6, z: 101)
Coordinates:
    glat     (z, f) float64 -10.89 -11.41 -11.87 -12.28 ... 18.83 19.24 19.64
    glon     (z, f) float64 251.8 251.7 251.6 251.5 ... 257.1 257.2 257.3 257.4
    zalt     (z, f) float64 85.0 85.01 85.0 85.0 85.01 ... 85.01 85.0 85.01 85.0
  * ut       (ut) float64 0.007778 0.2578 0.5078 0.7578 1.008 1.251
Dimensions without coordinates: f, ion, z
Data variables:
    deni     (z, f, ion, ut) float64 1e-06 1e-06 1e-06 ... 0.002514 0.002232
    vsi      (z, f, ion, ut) float64 0.0 0.0 0.0 0.0 ... -8.739 -5.68 -2.654
    ti       (z, f, ion, ut) float64 205.7 205.5 205.6 ... 186.6 184.9 183.4
    te       (z, f, ut) float64 205.4 205.2 205.2 205.4 ... 184.6 183.2 181.9
    slt      (ut) float64 17.02 17.27 17.52 17.77 18.02 18.26
```

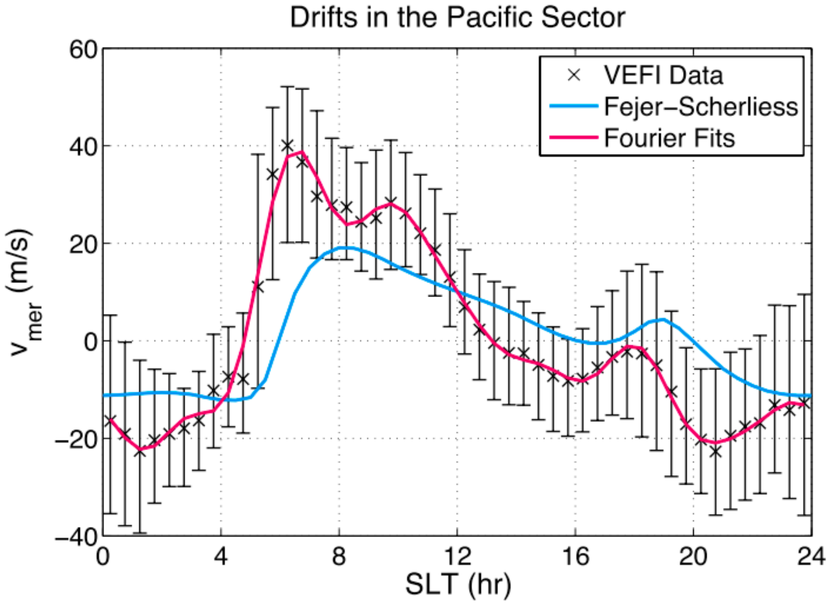
Application Overview

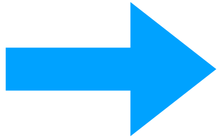
- Application Usability Levels establish maturity of tools for a given usage.
  - Rapid investigation of hypotheses (AUL 7)
  - Input into the `growin` package for calculations of instability growth rates (AUL 5)
  - Usage in teaching environments (AUL 1)

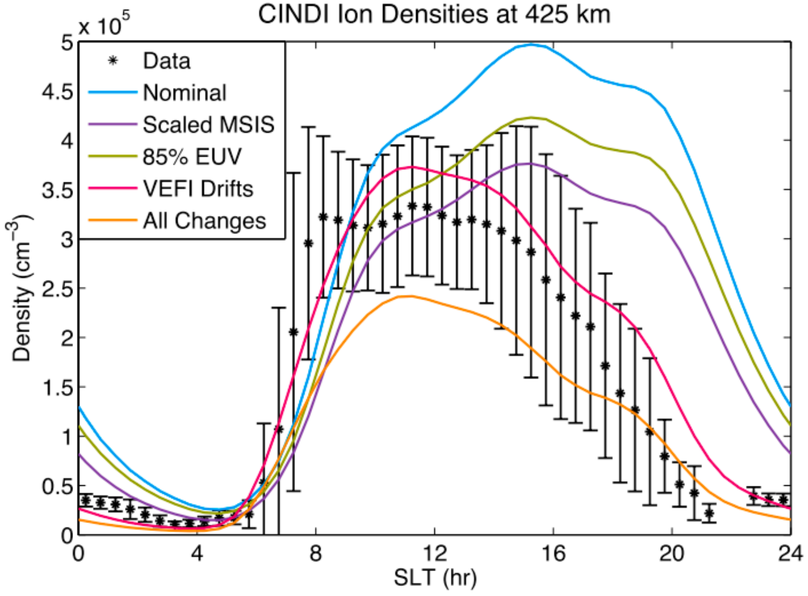


[Halford et al, 2019]

Hypothesis Testing



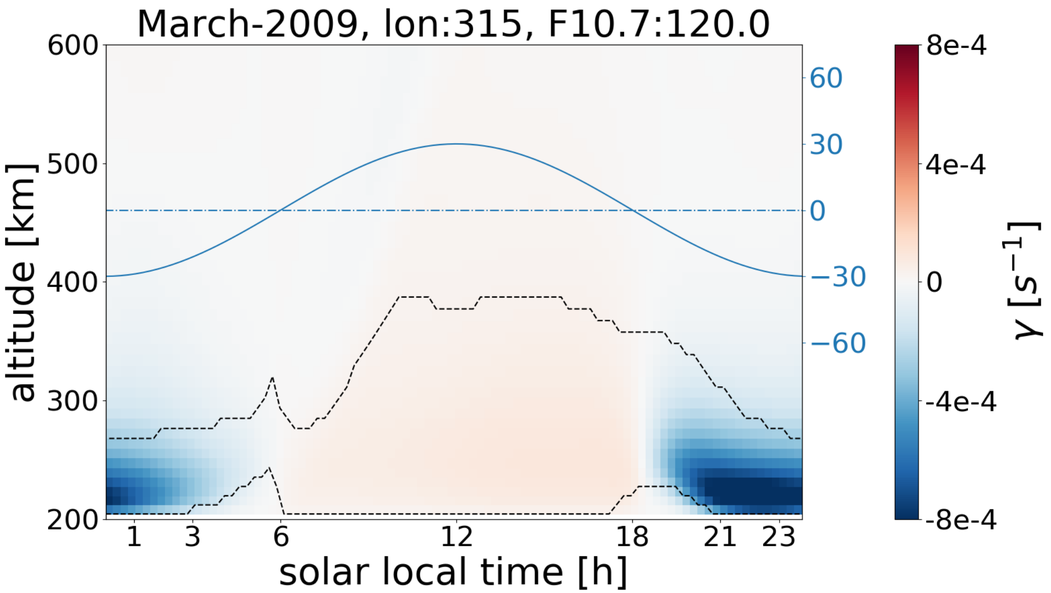


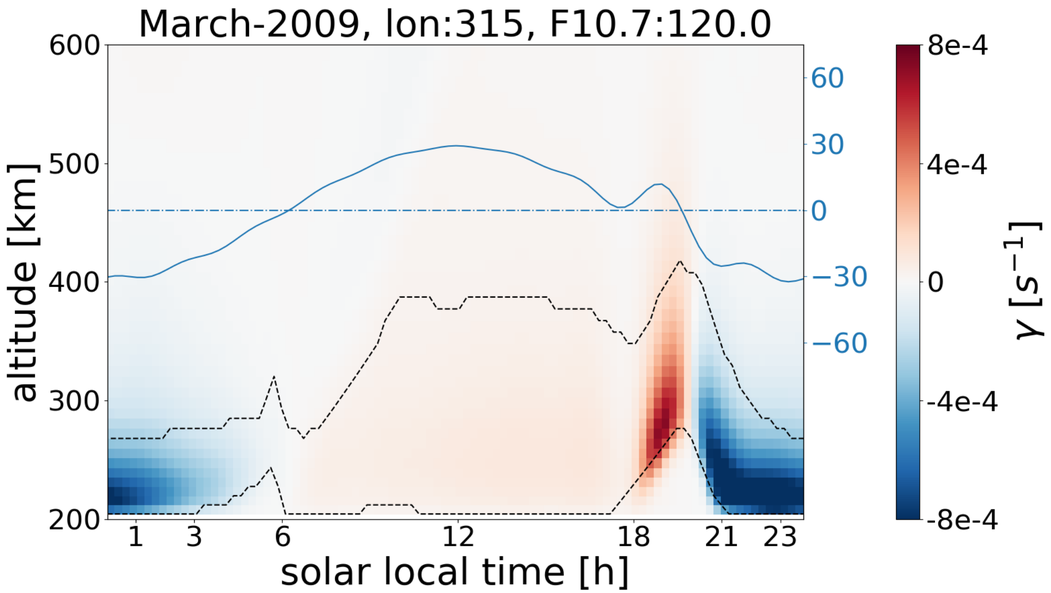


Interaction with growyn

- Calculates Rayleigh-Taylor Instability growth rates for given ionosphere and thermosphere

- Prototyping initial studies with sami2py, with the intent of deploying on SAMI3 in the future.





[courtesy Jonathon Smith]

Possible Development Paths

- Using f2py instead of a compiled fortran executable
- Reorganizing the class structure for a streamlined usage
- pypi compatibility? (Currently GitHub only)
- Building a library of plotting tools?

- Looking for community feedback to guide where efforts are spent

Final Thoughts

- Sami2py 0.2.3 was released last week.
- Looking for feedback for future development / applications.

Development team
Jeff Klenzing
Jonathon Smith

Code Contributors
Angeline Burrell
Michael Hirsch
Reika Kitano
zzyzty

Useful Discussions
Joe Huba
Alexa Halford