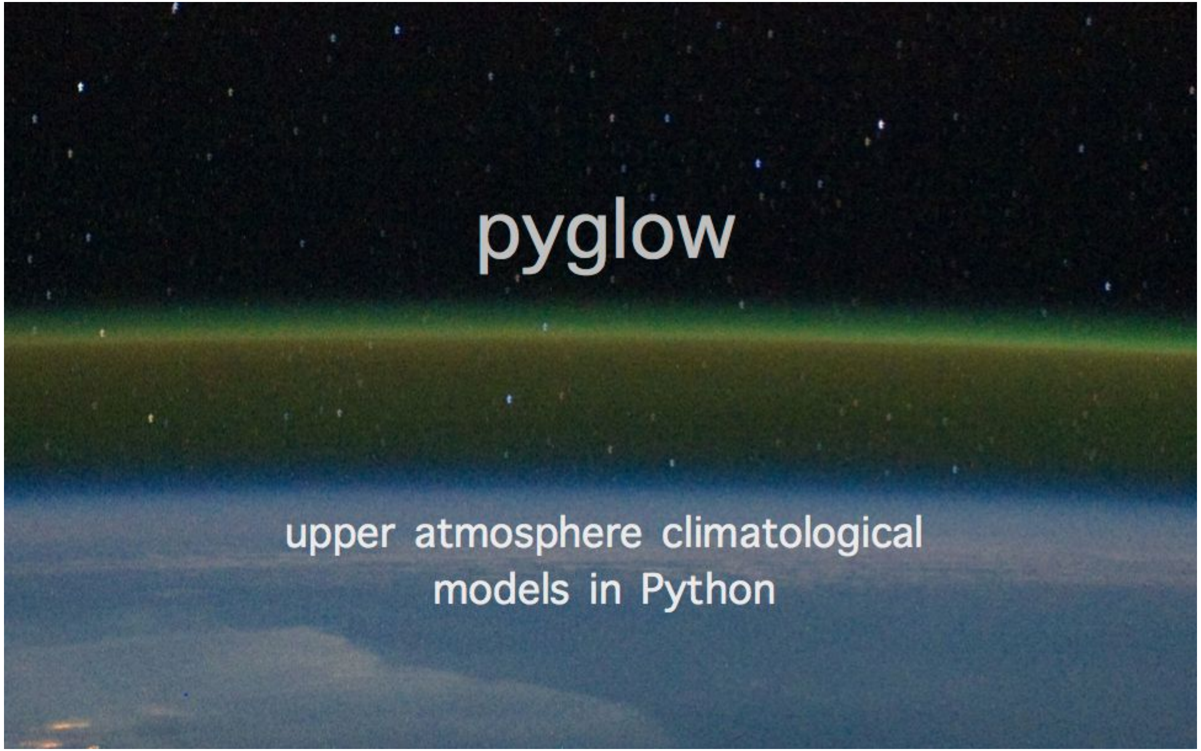


pyglow

upper atmosphere climatological models in Python

Tim Duly[1]

Mark Butala[2], Dan Fisher[2], & Brian Harding[2]

Download pyglow at:

https://github.com/timduly4/pyglow

1. Spire Global, Inc., Boulder, CO
2. University of Illinois at Urbana-Champaign, Urbana, IL

Outline

1. Introduction & background (*what problem we're trying to solve)*

2. GitHub tour & installation guide

3. Demo with IPython

4. Real world application (*TEC data from a GNSS-RO CubeSat satellite*)

5. Future work

Climatological (Empirical) models

- Often used within the upper atmosphere research community for a wide variety of applications
  - Initializing values in numerical models
  - Baseline comparisons against measured data
- Commonly derived from ground and space measurements
- Provide a “probabilistic seasonal forecast”

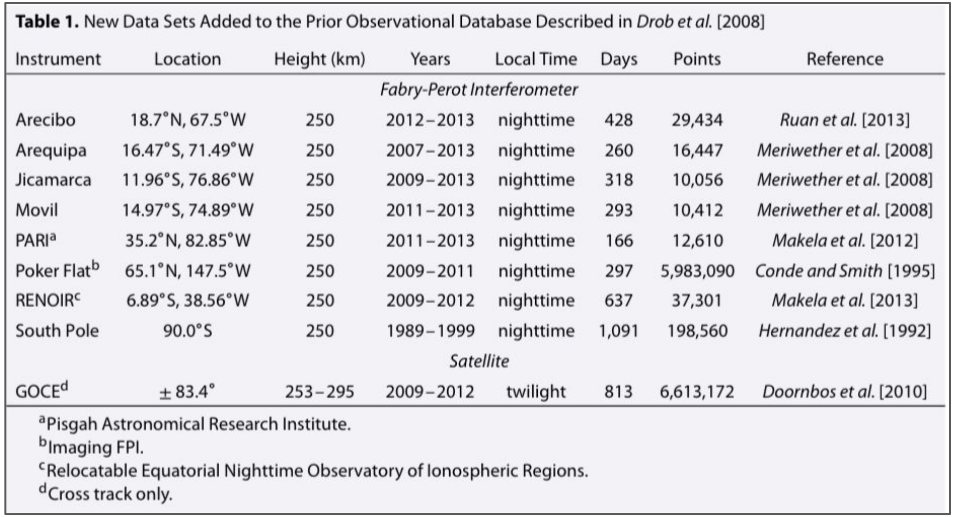


Table 1. New Data Sets Added to the Prior Observational Database Described in *Drob et al.* [2008]

Instrument	Location	Height (km)	Years	Local Time	Days	Points	Reference
			Fabry-Perot Interferometer				
Arecibo	18.7°N, 67.5°W	250	2012–2013	nighttime	428	29,434	*Ruan et al.* [2013]
Arequipa	16.47°S, 71.49°W	250	2007–2013	nighttime	260	16,447	*Meriwether et al.* [2008]
Jicamarca	11.96°S, 76.86°W	250	2009–2013	nighttime	318	10,056	*Meriwether et al.* [2008]
Movil	14.97°S, 74.89°W	250	2011–2013	nighttime	293	10,412	*Meriwether et al.* [2008]
PARI[a]	35.2°N, 82.85°W	250	2011–2013	nighttime	166	12,610	*Makela et al.* [2012]
Poker Flat[b]	65.1°N, 147.5°W	250	2009–2011	nighttime	297	5,983,090	*Conde and Smith* [1995]
RENOIR[c]	6.89°S, 38.56°W	250	2009–2012	nighttime	637	37,301	*Makela et al.* [2013]
South Pole	90.0°S	250	1989–1999	nighttime	1,091	198,560	*Hernandez et al.* [1992]
			Satellite				
GOCE[d]	± 83.4°	253–295	2009–2012	twilight	813	6,613,172	*Doornbos et al.* [2010]

[a] Pisgah Astronomical Research Institute.
[b] Imaging FPI.
[c] Relocatable Equatorial Nighttime Observatory of Ionospheric Regions.
[d] Cross track only.

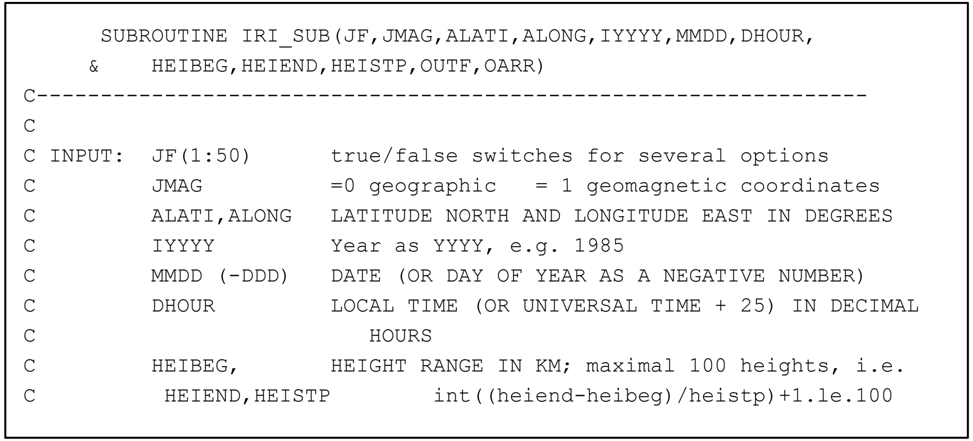
Data sets contributing to the Horizontal Wind Model (HWM) 2014 climatological model [Drob et al. 2015]

Example climatological models in pyglow

Climatological Model	*Description*	*Terms Modeled*	*Reference*
IRI	*International Reference Ionosphere*	*Plasma*	*Bilitza and Reinisch [2008]*
MSIS	*Mass Spectrometer Incoherent Scatter Radar*	*Neutral*	*Picone [2002]*
HWM	*Horizontal Wind Model*	*Neutral Wind*	*Hedin and Biondi [1996]*
IGRF	*International Geomagnetic Reference Field*	*Magnetic Field*	*Finlay et al. [2010]*

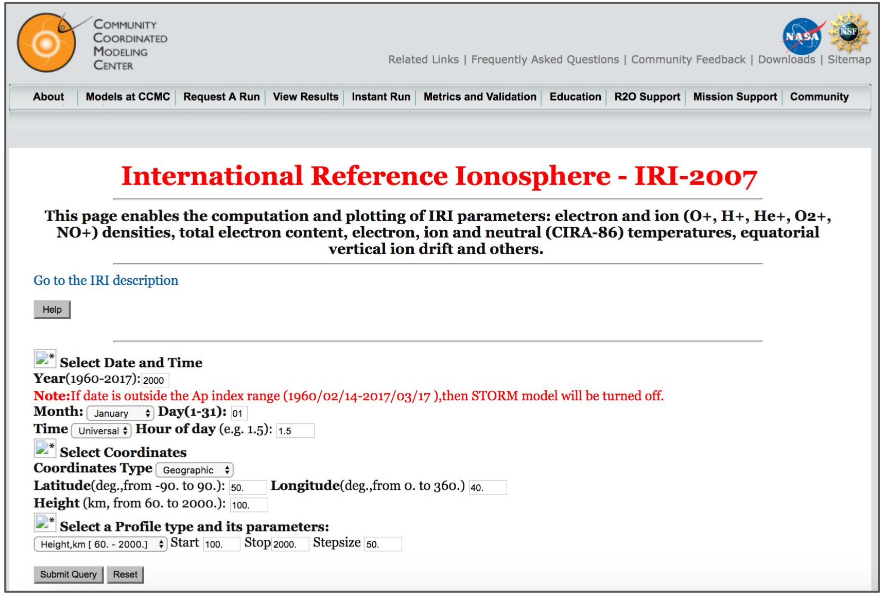
Access to climatological models

Option 1: Download, compile, and call FORTRAN source code:



```
      SUBROUTINE IRI_SUB(JF,JMAG,ALATI,ALONG,IYYYY,MMDD,DHOUR,
     &    HEIBEG,HEIEND,HEISTP,OUTF,OARR)
C------------------------------------------------------------------
C
C INPUT:  JF(1:50)      true/false switches for several options
C         JMAG          =0 geographic   = 1 geomagnetic coordinates
C         ALATI,ALONG   LATITUDE NORTH AND LONGITUDE EAST IN DEGREES
C         IYYYY         Year as YYYY, e.g. 1985
C         MMDD (-DDD)   DATE (OR DAY OF YEAR AS A NEGATIVE NUMBER)
C         DHOUR         LOCAL TIME (OR UNIVERSAL TIME + 25) IN DECIMAL
C                          HOURS
C         HEIBEG,       HEIGHT RANGE IN KM; maximal 100 heights, i.e.
C          HEIEND,HEISTP        int((heiend-heibeg)/heistp)+1.le.100
```

Option 2: Use the Community Coordinated Modeling Center (CCMC) web interface



International Reference Ionosphere - IRI-2007

This page enables the computation and plotting of IRI parameters: electron and ion (O+, H+, He+, O2+, NO+) densities, total electron content, electron, ion and neutral (CIRA-86) temperatures, equatorial vertical ion drift and others.

Go to the IRI description

Help

Select Date and Time

Year(1960-2017): 2000

Note: If date is outside the Ap index range (1960/02/14-2017/03/17),then STORM model will be turned off.

Month: January **Day(1-31):** 01

Time Universal **Hour of day** (e.g. 1.5): 1.5

Select Coordinates

Coordinates Type Geographic

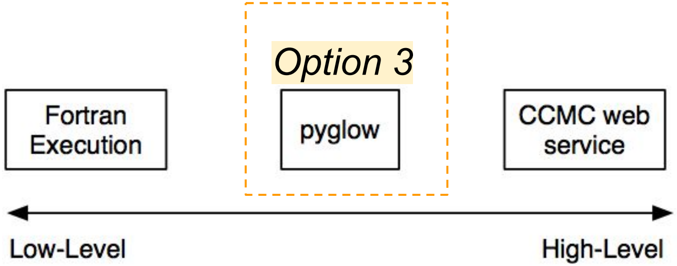
Latitude(deg.,from -90. to 90.): 50. **Longitude**(deg.,from 0. to 360.) 40.

Height (km, from 60. to 2000.): 100.

Select a Profile type and its parameters:

Height,km [60. - 2000.] Start 100. Stop 2000. Stepsize 50.

Submit Query Reset



Option 3

Fortran Execution

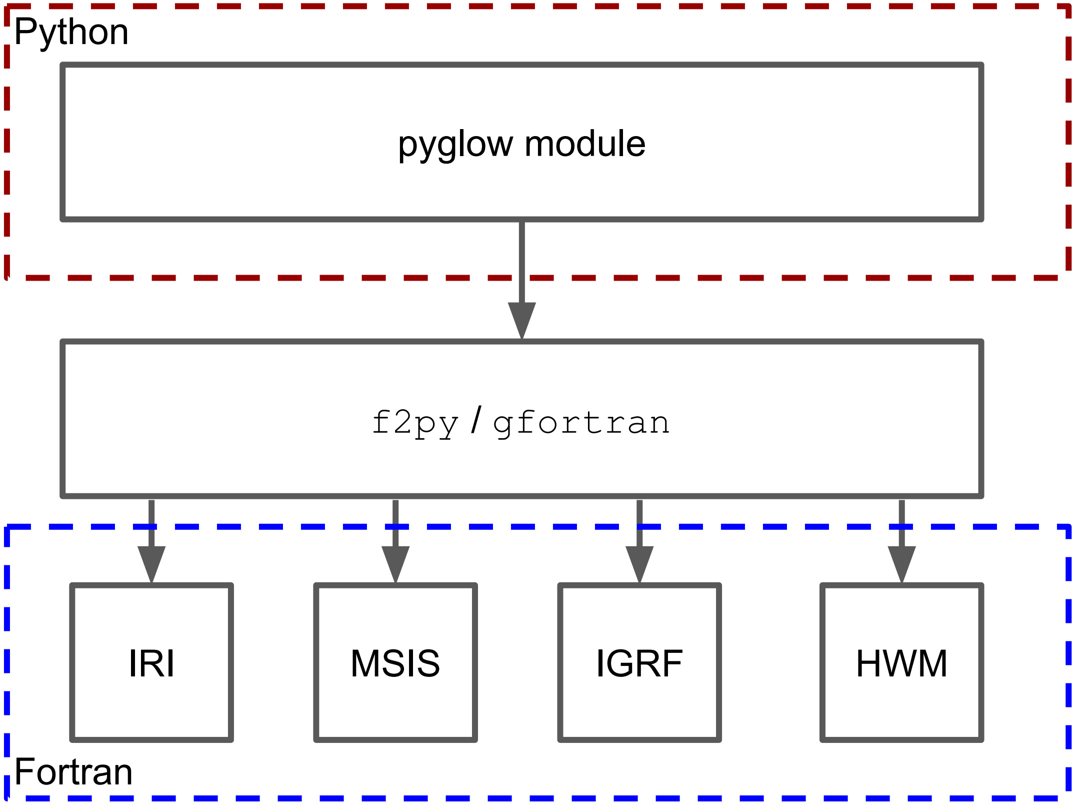
pyglow

CCMC web service

Low-Level

High-Level

pyglow



- f2py (available in numpy) is used as the “glue” between the pyglow module and the Fortran models

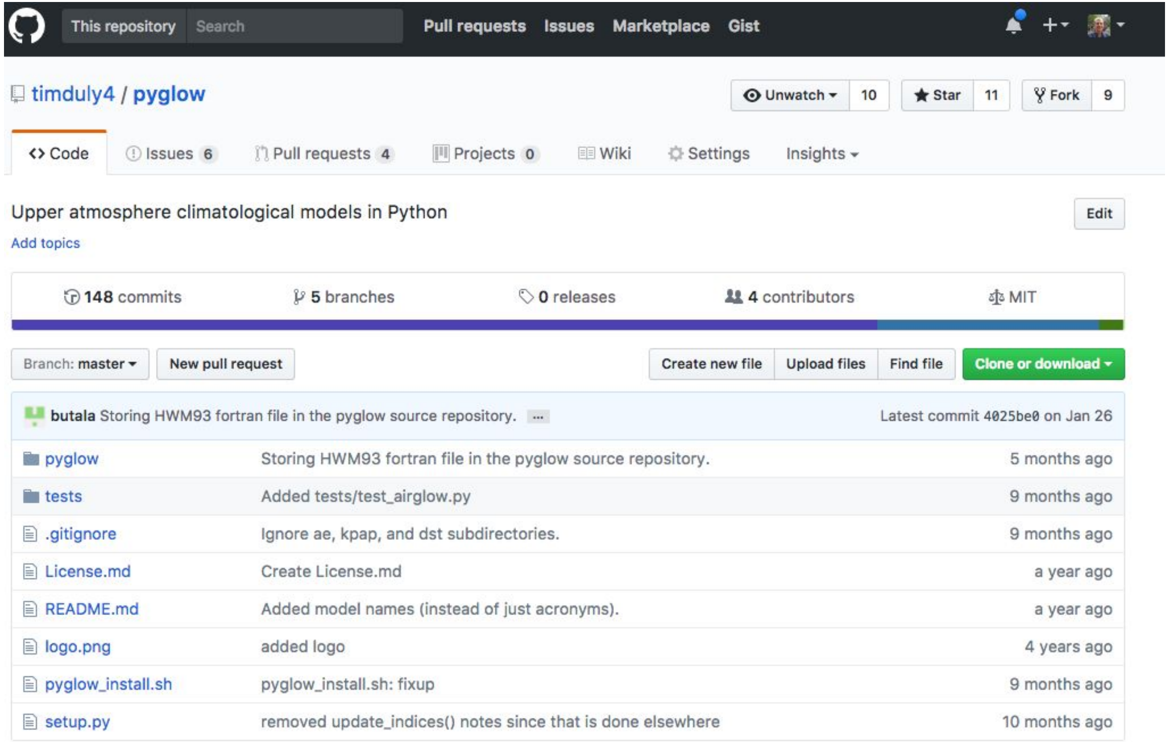
- Access to models under common framework enables derived parameters to be calculated (e.g., airglow emission) and synergy between models (e.g., integrating electron densities along magnetic field line)

Modules are compiled in Fortran, resulting in minor performance hits for calling each climatological modules

Could also wrap C++ functions

GitHub tour & installation guide

<demo>



timduly4 / pyglow

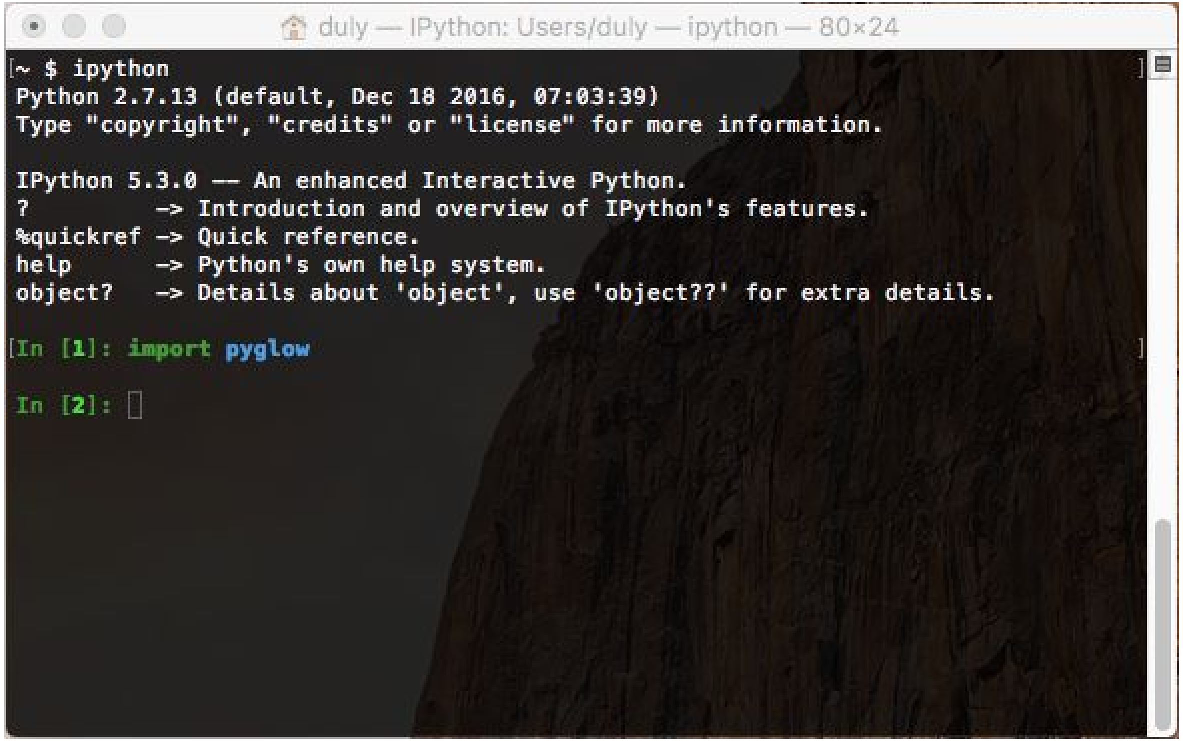
Upper atmosphere climatological models in Python

butala Storing HWM93 fortran file in the pyglow source repository. ... Latest commit 4025be0 on Jan 26

pyglow	Storing HWM93 fortran file in the pyglow source repository.	5 months ago
tests	Added tests/test_airglow.py	9 months ago
.gitignore	Ignore ae, kpap, and dst subdirectories.	9 months ago
License.md	Create License.md	a year ago
README.md	Added model names (instead of just acronyms).	a year ago
logo.png	added logo	4 years ago
pyglow_install.sh	pyglow_install.sh: fixup	9 months ago
setup.py	removed update_indices() notes since that is done elsewhere	10 months ago

Demo with IPython

<demo>



```
~ $ ipython
Python 2.7.13 (default, Dec 18 2016, 07:03:39)
Type "copyright", "credits" or "license" for more information.

IPython 5.3.0 -- An enhanced Interactive Python.
?         -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help      -> Python's own help system.
object?   -> Details about 'object', use 'object??' for extra details.

In [1]: import pyglow

In [2]: 
```

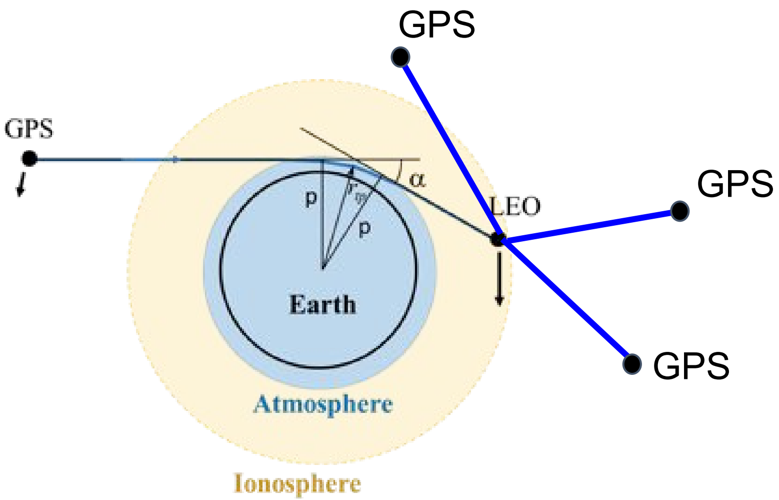
Real world application: TEC derived from GNSS-RO CubeSat

Satellite Production (up to 12 at once)

Deployment of two Spire 3U CubeSats from the ISS

spire

Real world application: TEC derived from GNSS-RO CubeSat



$$\mathrm{TEC} \equiv \int n_e(l)\, dl$$

[electrons/m²]

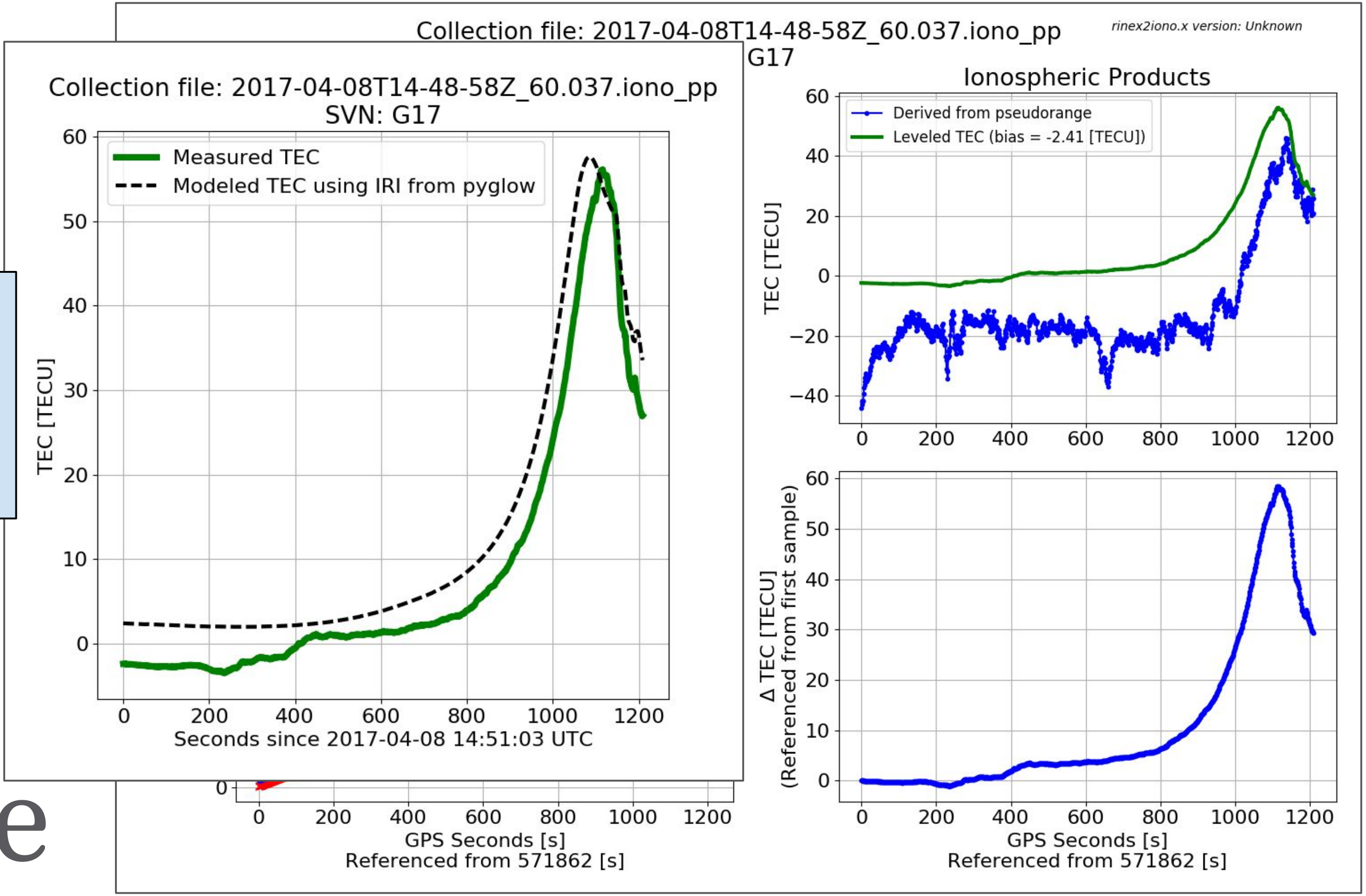
Electron density [electrons / m³]

Infinitesimal path segment [m]

Sum along a path length

Real world application: TEC derived from GNSS-RO CubeSat

Ionospheric data products collected from a dual frequency GNSS receiver



Future work

1. Develop API & documentation
2. Unit tests

Always welcome contributions and improvements

Feel free to download, modify, and submit Pull Requests (PR) via GitHub

Download pyglow at:

https://github.com/timduly4/pyglow

Contact:

Tim Duly: `timduly4@gmail.com`