

magplots and conjugate_map: Two Python Libraries for Visualizing Conjugate Ground Magnetometers

Kristina Collins and Michelle Salzano
Space Science Institute

Pull requests welcome!

Abstract

The polar regions are valuable observing platforms in geospace science, because their magnetospheric and ionospheric conditions are markedly different from conditions in the midlatitudes. Geomagnetic conjugate points in the northern and southern hemispheres – i.e., points connected by magnetic field lines – affect one another’s space weather conditions on the order of minutes. Space weather in Antarctica, therefore, influences and is influenced by space weather in the northern hemisphere, as observed in, e.g., auroral conditions and radio propagation. Geomagnetically conjugate ground magnetometers are an integral tool for the investigation of interhemispheric asymmetries. However, the geomagnetic conjugate relationship is not straightforward to visualize with many common mapping tools, due to the ubiquity of midlatitude-oriented map projections and the drift of Earth’s magnetic field over time. Here, we present two Python libraries to facilitate observations with conjugate magnetometers: **magplots**, which presents tools for time domain, spectrogram, and wave power estimation, specifically for conjugate magnetometer data; and **conjugate_map**, which facilitates calculation and mapping of multiple instrumentation networks, including ground-based instruments, radars and satellites. These tools may be used to streamline interhemispheric investigations and highlight regions of potential interest for future deployments.

Upcoming AGU Session – Seeking Submissions!

SA023: The MacGyver Session: The Place for Novel, Exciting, Self-Made, Hacked, or Improved Sensors and Software Solutions for the Year of Open Science and the Heliophysics Big Year

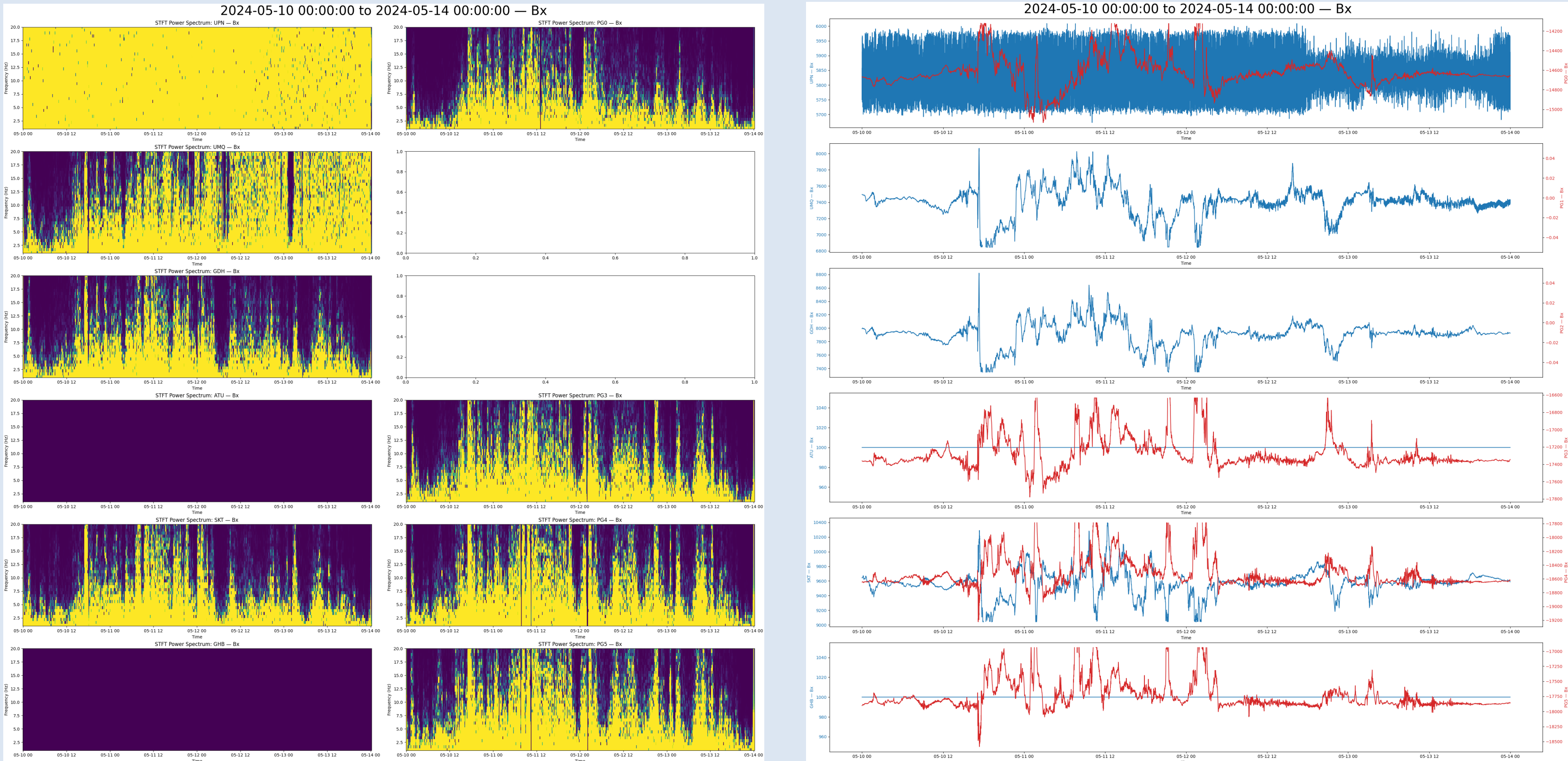
2024 is a Big Year for Heliophysics with a focus on eclipses, solar max, participatory science, and a global celebration of the Sun’s influence on Earth and the entire solar system. Makers, citizen scientists, ham radio enthusiasts, educators, and artists are welcome to bring broad, open science and STEAM outreach to this “Heliophysics Big Year” version of a MacGyver Session. Started in Hydrology more than a decade ago, this expanded concept now engages Space Physics and Aeronomy with MacGyver-style innovations, such as: new sensor systems which use technologies in novel or unintended ways; new software, algorithms, data storage or transmission solutions that send data from the field; and initiatives that facilitate the creation and sharing of novel sensors, open source code and data, and software systems. Feel free to bring prototypes and demonstrations. We look forward to sharing our MacGyvering for the broader AGU community during this Heliophysics Big Year.

AGU24 WHAT'S NEXT FOR SCIENCE

pip install magplots

DOI 10.5281/zenodo.10703227

```
from magplots.magFunctions import *
import numpy as np
import datetime
start=datetime.datetime(2024, 5, 10, 0, 0, 0)
end=datetime.datetime(2024, 5, 14, 0, 0, 0)
magall(start = start, end = end, is_saved = True)
```



pip install conjugate_map

DOI 10.5281/zenodo.10056624

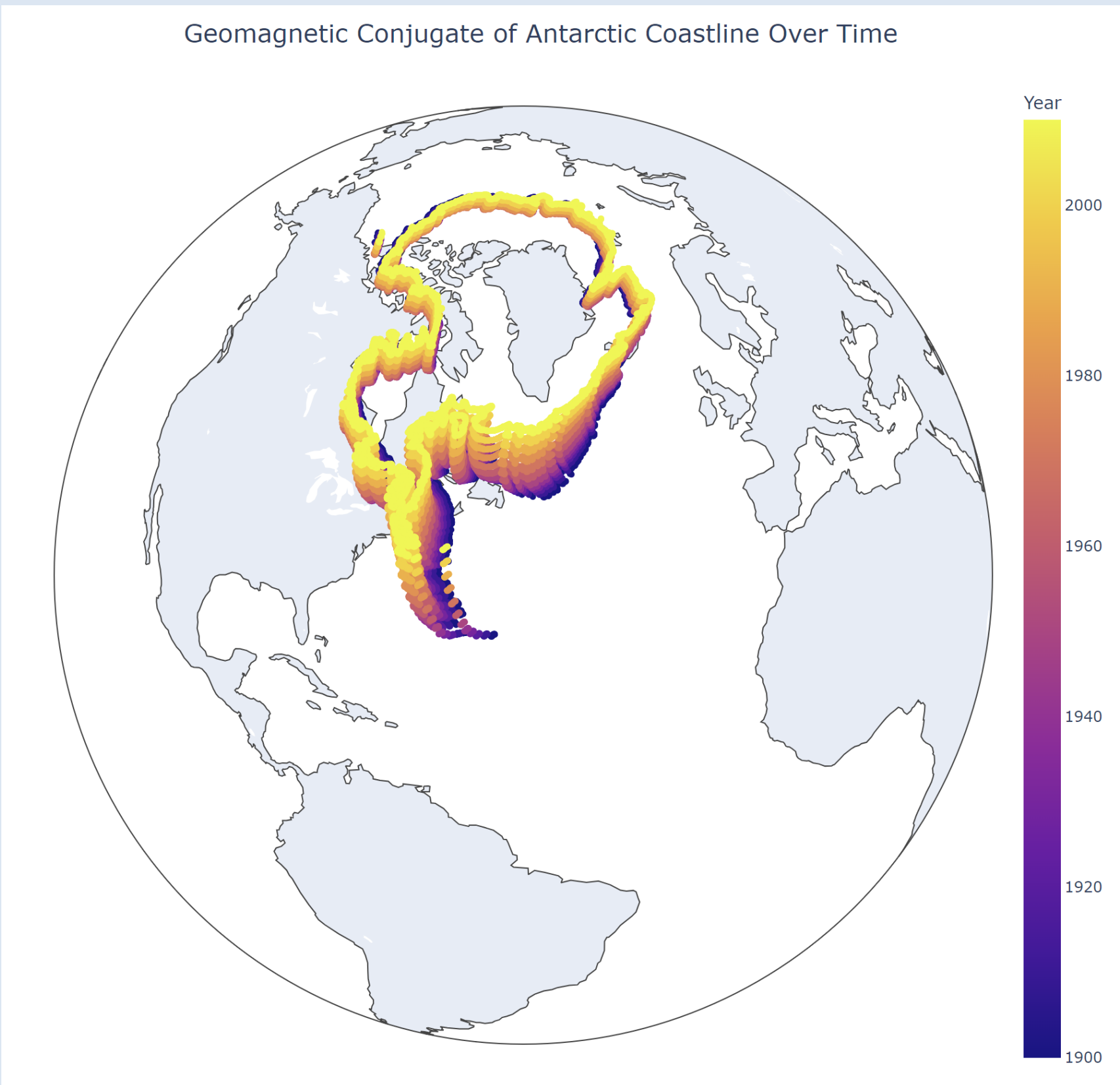
```
Compute conjugate points for one point:

In [12]: ut = dt.datetime(1980, 11, 3, 18, 0, 0)
         lat, lon = [-64, -64]

         print("Geopack: ")
         print(findconj(lat, lon, ut, method="geopack"))
         print("AACGM: ")
         print(findconj(lat, lon, ut, method="aacgm"))
         print("\nAutomatic: ")
         print(findconj(lat, lon, ut, method="auto", is_verbose = True))

Geopack:
(39.240006321966526, 291.07394597054764)
AACGM:
(36.700798441742975, -68.00768971281151)

Automatic:
Setting method according to latitude limits: aacgm
.....Calculating conjugate point for -64, -64 at 1980-11-03 18:00:00 with AACGMV2:
Magnetic lat/lon: [-47.84424934117308, 8.872298220296411]
Conjugate geographic lat/lon: [36.700798441742975, -68.00768971281151]
(36.700798441742975, -68.00768971281151)
```



```
[8]: conjcalc(df, latname = 'lat', lonname = 'lon')
```

	Name	lat	lon	Hemisphere	PLAT	PLON
0	McMurdo	-77.846323	166.668235	S	72.877386	-97.167701
1	SPA	-90.000000	0.000000	S	68.032785	-66.272482
2	PGC	44.984308	-93.182207	N	44.984308	-93.182207
3	SSI	40.019511	-105.240014	N	40.019511	-105.240014