P127

An Extension of the Python SHARPpy GUI to Display NASA AIRS Satellite Sounding Profiles and CAPE in Near-Real Time

Grace Przybyl, Callyn Bloch, Jessica Gartzke, Matthew Westphal, Robert Knuteson

University of Wisconsin-Madison, Space Science and Engineering Center

1. INTRODUCTION

SHARPpy is a collection of Python libraries created by Greg Blumberg and Kelton Halbert at the Uni. of Oklahoma that displays data in a sounding format used by the NWS storm prediction center (SPC). Currently SHARPpy supports operational radiosonde upper air soundings of temperature and water vapor as well as selected NWP model fields. The sounding display shows temperature and moisture vertical profiles on a skewt plot as well as tabulates values of CAPE (Convective Available Potential Energy) and other convective indices.

The NASA AIRS (Atmospheric InfraRed Sounder) sensor operates on a polar orbiting satellite with sun-synchronous overpasses at about 1:30 am and 1:30 pm. Retrievals of temperature and moisture from the AIRS data are produced in near real-time. The Automated Surface Observing System (ASOS) is used to measure surface temperature and dewpoint at airports across the U.S. at 2 m height in real-time. Since the AIRS surface temperature and dewpoint are often inaccurate, it was decided that these values could optionally be replaced with ASOS surface station data, creating AIRS profiles with the surface data from the nearest ASOS station.

The SHARPpy Graphical User Interface (GUI) plots points on a map using the latitudes and longitudes present in the locations file indicated by the corresponding .xml file. The goal was to recreate .xml and locations files for the AIRS satellite data so that SHARPpy would have the possibility of displaying satellite data in near real-time. Both the displaying of satellite data in near real-time and the substitution of ASOS surface temperature and dewpoint for the AIRS values will provide more accurate calculations of convective indices and more detailed temperature and moisture profiles, allowing users to gain a better understanding of the atmospheric instability and thereby improve the interpretation of a forecast.

2. EXTENSION OF SHARPpy

The existing SHARPpy (version 1.3.0. Xenia) uses a series of .xml files (Figure 1), locations files (Figure 2), and corresponding data profiles (Figure 3) to produce the map of sounding locations as seen in Figure 4. The .xml and locations files are located in a hidden folder on the user's computer while the profiles are located on FTP and HTTPS sites. In order to allow for the display of AIRS satellite and surface data, new .xml and location files needed to be created and placed into a hidden folder on the local computer running SHARPpy. The AIRS.xml file is a copy of the standard.xml that comes with the SHARPpy program. The AIRS_locations.txt file was created using a python script that ran over all of the satellite profiles, printing out the pertinent information into the locations files. Once both the AIRS.xml and AIRS_locations.txt were placed into the hidden folder, SHARPpy was able to read in and display the AIRS satellite data.

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<sourcelist>
    <datasource name="SURFACE" ensemble="false" observed="true">
    <outlet name="SURFACE" url="ftp://ftp.ssec.wisc.edu/pub/ssec/gracep/Surface/{srcid}.csv" format="spc" >
            <time range="0" delta="0" offset="0" delay="1" cycle="24" archive="24"/>
            <points csv="surface_locations.txt" />
        </outlet>
    </datasource>
</sourcelist>
```

Figure 1. Surface.xml file used to direct SHARPpy to display the AIRS with ASOS surface data.

```
icao,iata,synop,name,state,country,lat,lon,elev,priority,srcid
AIRS_1,AIRS,,AIRS,,US,19.5149, -96.4443,229,3,profile1
AIRS_10,AIRS,,AIRS,,US,23.7762, -97.7166,229,3,profile10
AIRS_100,AIRS,,AIRS,,US,24.1523, -95.8424,229,3,profile100
AIRS_101,AIRS,,AIRS,,US,24.6277, -95.9842,229,3,profile101
AIRS_102,AIRS,,AIRS,,US,25.1028, -96.1272,229,3,profile102
AIRS_103,AIRS,,AIRS,,US,25.5779, -96.2715,229,3,profile103
AIRS_104,AIRS,,AIRS,,US,26.0528, -96.4172,229,3,profile104
AIRS_105,AIRS,,AIRS,,US,26.5273, -96.5642,229,3,profile105
AIRS_1055,AIRS,,AIRS,,US,30.6156, -86.8405,229,3,profile1055
AIRS_1055_S,AIRS,,AIRS,,US,30.6156, -86.9405,229,3,profile1055_S
AIRS_1056,AIRS,,AIRS,,US,31.0979, -86.9501,229,3,profile1056
AIRS_1057,AIRS,,AIRS,,US,31.5805, -87.0570,229,3,profile1057
AIRS_1058,AIRS,,AIRS,,US,32.0633, -87.1614,229,3,profile1058
AIRS_1059,AIRS,,AIRS,,US,32.5454, -87.2723,229,3,profile1059
AIRS_106,AIRS,,AIRS,,US,27.0016, -96.7127,229,3,profile106
AIRS_1060,AIRS,,AIRS,,US,33.0279, -87.3807,229,3,profile1060
AIRS_1061,AIRS,,AIRS,,US,33.5101, -87.4896,229,3,profile1061
AIRS_1062,AIRS,,AIRS,,US,33.9925, -87.5990,229,3,profile1062
AIRS_1063,AIRS,,AIRS,,US,34.4747, -87.7090,229,3,profile1063
AIRS_1063_S,AIRS,,AIRS,,US,34.4747, -87.8090,229,3,profile1063_S
251.0-5.5-4.5-KASX,ASOS,,ASOS,,US,43.2,-90.7,251,3,ASOS
202.0-24.7-14.9-KOVS,ASOS,,ASOS,,US,44.9,-91.5,202,3,ASOS
276.0-20.0-12.7-KEAU,ASOS,,ASOS,,US,43.8,-88.5,276,3,ASOS
240.0-22.8-14.3-KFLD,ASOS,,ASOS,,US,42.6,-87.9,240,3,ASOS
219.0-26.0-13.6-KENW,ASOS,,ASOS,,US,43.9,-91.3,219,3,ASOS
199.0-22.3-15.2-KLSE,ASOS,,ASOS,,US,43.2,-90.2,199,3,ASOS
217.0-21.4-16.5-KLNR,ASOS,,ASOS,,US,43.1,-89.4,217,3,ASOS
261.0-21.8-14.9-KMSN,ASOS,,ASOS,,US,44.6,-90.2,261,3,ASOS
```

Figure 2. Surface_locations.txt, used by SHARPpy to plot the profile locations in the GUI display.

```
%TITLE%
profile   130531/1920
  29.52701
 -96.69506
%RAW%
    986.05,      228.79,       27.79,       12.59,
    958.58,      465.18,       25.40,       15.24,
    931.51,      703.49,       23.38,       14.14,
    904.85,      943.73,       21.79,       13.24,
    878.61,     1185.93,       20.50,       11.95,
    852.78,     1430.10,       19.50,        9.54,
    827.36,     1676.26,       18.67,        7.16,
    802.36,     1924.43,       17.86,        4.54,
    777.78,     2174.64,       17.00,        2.02,
    753.62,     2426.89,       16.06,       -1.36,
    729.87,     2681.22,       14.62,       -5.16,
    706.55,     2937.63,       12.98,       -8.37,
    683.66,     3196.17,       11.19,      -11.27,
    661.18,     3456.84,        9.42,      -13.71,
    639.13,     3719.67,        7.44,      -16.29,
    617.50,     3984.68,        5.37,      -18.45,
    596.30,     4251.91,        3.26,      -21.14,
    575.51,     4521.37,        1.42,      -23.61,
    555.16,     4793.09,       -0.48,      -26.23,
```
Figure 3. Example AIRS profile in csv format.

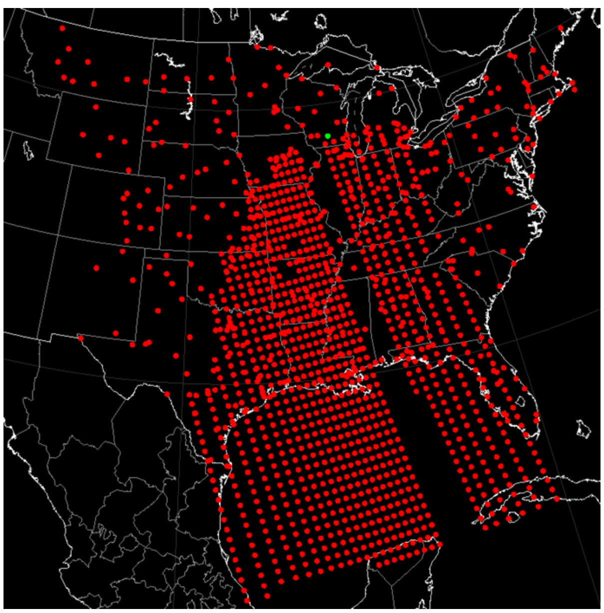
Figure 4. An overlay of AIRS overpass sounding locations and ASOS surface station locations.

Once the AIRS data was able to be viewed in the SHARPpy GUI, it was determined that the surface data in the satellite sounding profiles could be causing inaccuracies in the calculations of convective indices. This is what lead to the decision to replace AIRS surface temperature and dewpoint with that of the ASOS surface data. In order to display the ASOS station locations to find which AIRS profiles were the closest, new .xml and locations files were created by the same process as the AIRS. After determining the AIRS profiles with the closest proximity to ASOS stations, Surface.xml and Surface_locations.txt were created. The resulting drop down menu in the SHARPpy GUI (Figure 5) now had options for displaying AIRS data, ASOS station locations, and Surface data which is the AIRS files with surface temperature and dewpoint replaced by time and space coincident ASOS observations.

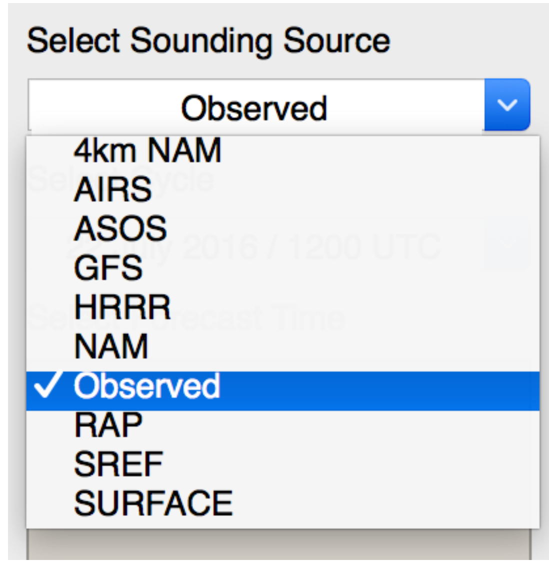

Figure 5. New SHARPpy drop down menu, including the added AIRS satellite data (AIRS) and ASOS station locations (ASOS), and the AIRS data with replaced surface station data (SURFACE).

3. RESULTS

By modifying the .xml files used in the default configuration of SHARPpy, AIRS satellite profiles have been successfully displayed in the SHARPpy GUI. This displays the satellite sounding as a skewT plot and uses the SHARP algorithms to tabulate CAPE and other convective indices. By adding an AIRS menu item to the GUI data source selection, the user can easily compare AIRS soundings to those of conventional radiosondes or NWP profiles already supported by the default SHARPpy distribution.

Replacing the temperature and dewpoint in the first line of data in AIRS profiles closest to ASOS surface stations with the temperature and dewpoint from those stations is selectable from the GUI locations map. This results in improved values in the calculated surface CAPE. An example of the degree of change the surface data can cause is illustrated in Figure 6 by the AIRS satellite profile in SE Louisiana and in Figure 7 by the same AIRS satellite profile with ASOS station KBTR surface temperature and dewpoint observations. The surface CAPE changed from 0 for the AIRS sounding alone to over 2700 J/Kg using the ASOS surface T/Tdew observations at the overpass time. This presents a change in CAPE values from no risk to moderate risk. The merging of surface observations and satellite retrievals are expected to lead to an improvement in the usefulness of the satellite soundings from AIRS, IASI, and CrIS sensors.

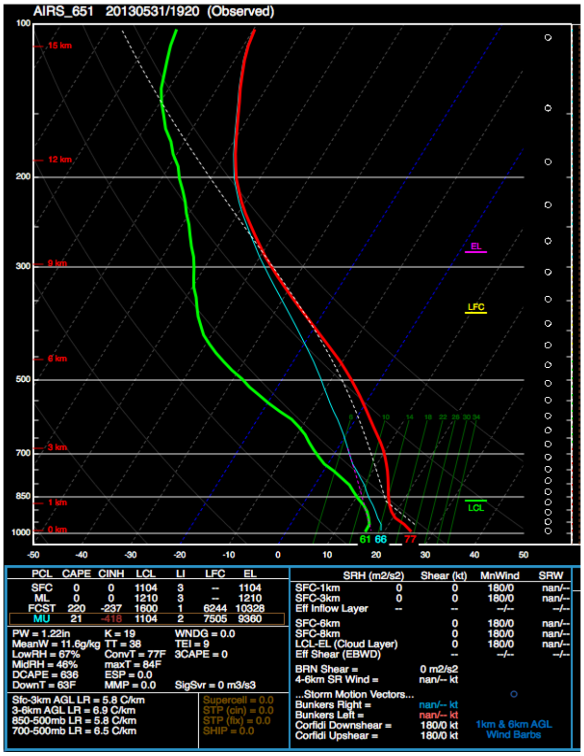

Figure 6. AIRS located in SE Louisiana, displayed in SHARPpy as a temperature and moisture profile. Surface CAPE is 0 J/Kg.

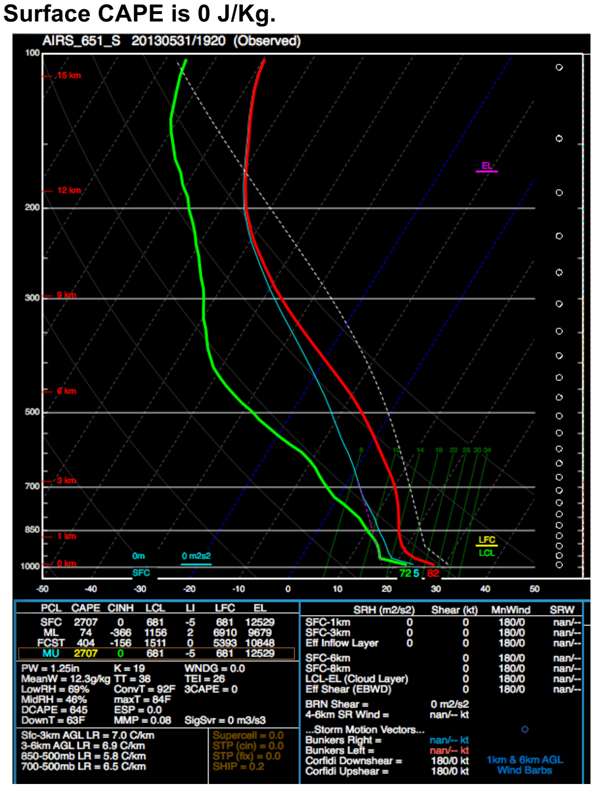

Figure 7. Same as Fig. 6 with ASOS T/Tdew in place of AIRS. Surface CAPE is 2707 J/Kg.

4. CONCLUSIONS and FUTURE WORK

This paper describes details of a modification to the default SHARPpy distribution to allow it to display soundings from polar orbiting satellites. Currently the user must download and store the .xml and locations files on their own computer in a hidden folder if they want to display any outside data sources in SHARPpy as seen on the left side of Figure 8. However, in the future, changes to the .xml file and the implementation of an airs_decoder.py will allow for remote access to these files so that the user is able to simply download and open SHARPpy and have the AIRS and ASOS data already present as a data source option (right side of Figure 8). Another new feature goal is the movement of the AIRS profiles from their current place on the SSEC FTP site to a HTTPS site where all files will be contained on a single web page and the .xml file with simply point SHARPpy to this site to retrieve the profile data. The increase in CAPE caused by merging of ASOS data indicates that while the AIRS satellite gathers accurate data higher up in the atmosphere, the estimate of the surface temperature and dewpoint is not completely accurate. Inclusion of surface observations are expected to enhance the value of SHARPpy for satellite applications with application to near real-time weather forecasting.

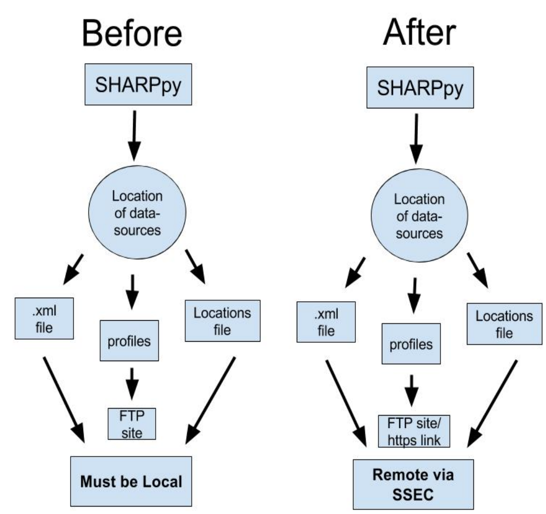

Figure 8. Proposed future mods to SHARPpy.

REFERENCES

Halbert, K. T., W. G. Blumberg, and P. T. Marsh, 2015: SHARPpy: Fueling the Python Cult, 5th Symp on Adv in Modeling and Analysis Using Python, Phoenix AZ.

Gartzke, J., R. Knuteson, G. Przybyl, S. Ackerman, and H. Revercomb, 2016: Comparison Of Satellite, Model, And Radiosonde Derived Convective Available Convective Energy (Cape) in the Southern Great Plains Region, submitted to Journal of Applied Meteorology and Climatology.