Working with ICRF Catalogues

Sayan Basu & Dr Aletha de Witt HartRAO/SARAO

2019 AVN Training









In this tutorial we will use the International Celestial Reference Frame (ICRF) catalogue to produce a plot showing the distribution of the sources on the sky, using Aitoff Hammer Projection. Furthermore, we will analyse the error in the source position. For this, TOPCAT will be used first and then you will write a **Python script** for the same purpose.

TOPCAT

<u>Tool for OP</u>erations on <u>Catalogues And Tables (TOPCAT)</u> is an interactive graphical viewer and editor to work with table data. It is a very helpful tool for astronomers for analysis and manipulation of source catalogues and other tables. Different types of table formats (ASCII, FITS, VOTable etc.) can be used in TOPCAT for analysis purposes.

The program is written in Java and available under the GNU General Public Licence.

> Tutorial

Aims of the tutorial are:

- 1. Use TOPCAT to download catalogues which are available ONLINE as well as catalogues that are not.
- 2. Make all sky plot (Aitoff Hammer Projection) of source positions in the ICRF-2 and ICRF-3 catalogues.
- 3. How to work with columns and rows in a table.

Start TOPCAT from your terminal:

1. \$: java -jar <topcatfilename>.jar. You should get the window in Figure 1.

	$\square \square $
r Table List	Current Table Properties
38 / 911 M	SAMP- Messages: O Clients: • 🌺

Figure 1: TOPCAT start up window.

- 2. Now we will look for the ICRF-2 catalogue which is available online.
 - (a) Click on the **folder** icon and then click on the **ViZiER** icon. It will take you to the window in Figure 2.

	TOPCAT
	$ \bigcirc \bigcirc$
Table	
	Format: (auto)
	Location: OK
	Filestore Browser
	System Browser
	Loading Tables

Figure 2: Online catalogue search.

- 3. Once we click on the ViZiER icon it will direct us to the window in Figure 3. We will use the **keywords** box to search the available International Celestial Reference Frame catalogue.
- 4. Once we have downloaded the catalogue successfully, the window in Figure 4 will appear.
- 5. How many sources are in the ICRF-2 catalogue?

VizieR Server Server: http://vizier.u-strasbg.fr/	• •
Row Selection	
Object Name:	Resolve
RA:	degrees 🗘 (J2000)
Dec:	degrees 🗘 (J2000)
Radius:	degrees 🗘
All Rows Aiximum Row Count: 50000	
Column Selection Output Columns: standard	<u></u>
Catalogue Selection By Category	Keyword Surveys Missions
Sub-Table Details Include O	Obsolete Tables
	Search Catalogues Cancel Search
△ Name Popularity Density	Description Wavelengths Astronomy

Figure 3: Catalogue search window.

- 6. Now to produce an all-sky plot of sources in the ICRF-2 catalogue, click on the **sky plotting window** icon.
- 7. Now, we will make a plot like Figure 5.

Looking at Figure 5, can you please tell us why did we need ICRF-3?

	τορςατ
Table List	rCurrent Table Properties -
1: I_323_icrf2	Label: I_323_icrf2
	Location: I_323_icrf2 Name: I/323/icrf2 Rows: 3.414
	Columns: 16 Sort Order: 1
	Row Subset: All 😳
	Activation Actions: 1 / 7
119 / 911 M	Messages: Clients: 💿 🎂

Figure 4: Working window.

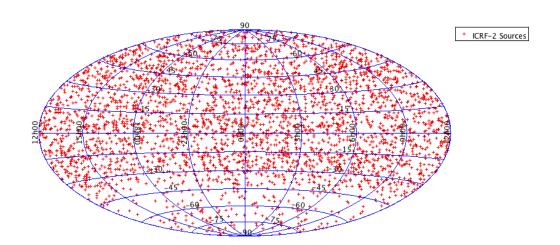


Figure 5: All sky plot using ICRF-2 catalogue.

We will now use the ICRF-3 catalogue:

The ICRF-3 catalogue is available inside the TOPCAT directory. The table data is in ASCII format.

How to load an ASCII table data using file browser?

- 1. Since the table data is in ASCII format, change the **Format** to **ASCII** from **auto**.
- 2. Click on the **Filestore Browser**. Look for the file name **icrf3.txt**. Please make sure you change the **Table Format** to **ASCII**. Finally, click on **OK**.
- 3. Now in TOPCAT **Table list**, we should have both ICRF-2 and ICRF-3 catalogues.

What is the basic difference between the two catalogues?

Lets now plot the ICRF-3 on top of the ICRF-2 catalogue to see the difference.

- 1. Select icrf3.txt now. We will use the same set-up as the ICRF-2 plot.
- 2. Data sky system galactic. Lon RAJ2000, Lat DEJ2000.

We expect a similar plot to Figure 7. Now can you spot the difference between the ICRF-2 and ICRF-3 source distribution? [Hint: Look at the small arrows inside the plot and think!!!]

Now, we move to the next part of the tutorial where we will work with the tables in the ICRF-3 catalogue.

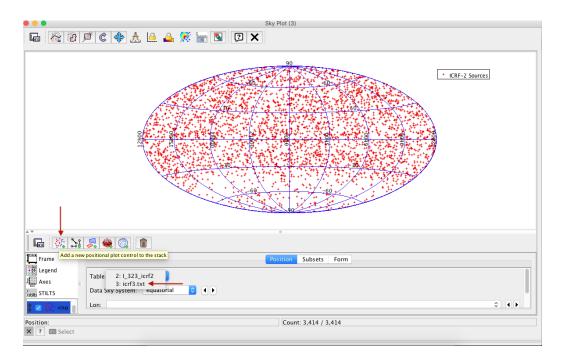


Figure 6: Handling more than one table data.

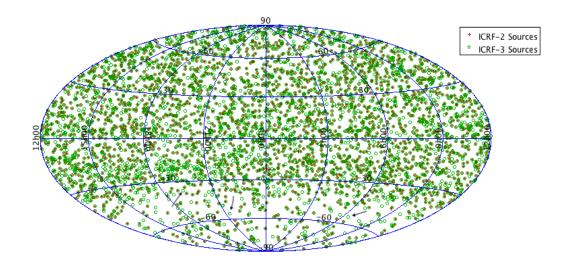


Figure 7: ICRF-2 and ICRF-3 all sky projection plot.

Working with columns and rows

In this section, the objective is to arrange the table data into subsets. Creating subsets according to our criteria makes analysis work easy in many cases.

There are 295 defining sources in the ICRF-2 catalogue which were used to maintain axes stability of the frame. Firstly, we will create a <u>subset</u> of defining sources in the ICRF-3 catalogue.

To do so:

1. Open the icrf3.txt. Look for the **status** column in the table (red block in Figure 8).

R B		for 3: icrf3.txt		_						
	TYLE	Designation	IERS	Status	RAJ2000	DEJ2000	D RA		D DEC	cor
3 00	ICF F	J224703.9-365746	2244-372	D	22 47 03.91732859	-36 57 46.3041035	3.860000	0E-6	6.480000E-5	-0
308	ICE F	J224838.6-323552	2245-328	D	22 48 38.68573790	-32 35 52.1881477	3.450000	0E-6	5.800000E-5	-6
320	ICEF	J225307.3+194234	2250+190	D	22 53 07.36916889	+19 42 34.6286731	2.420000	0E-6	4.040000E-5	-(
336	ICF F	J225717.3+074312	2254+074	D	22 57 17.30311778	+07 43 12.3025633	2.330000	9E-6	3.500000E-5	
407	ICF F	J232044.8+051349	2318+049	D	23 20 44.85659166	+05 13 49.9526016	2.050000	0E-6	3.100000E-5	-1
411	ICF.F	J232154.9+320407	2319+317	D	23 21 54.95598750	+32 04 07.6225103	2.480000	0E-6	3.210000E-5	-
414	ICRF	J232220.3+444542	2319+444	D	23 22 20.35808015	+44 45 42.3536071	3.560000	0E-6	4.460000E-5	
435	ICRF	J232747.9-144755	2325-150	D	23 27 47.96426891	-14 47 55.7511114	4.860000	0E-6	0.00012	-
454	ICRF	J233355.2-234340	2331-240	D	23 33 55.23782715	-23 43 40.6581980	4.920000	0E-6	9.600000E-5	-
462	ICRF	J233757.3-023057	2335-027	D	23 37 57.33907021	-02 30 57.6293054	2.250000	9E-6	3.420000E-5	
466	ICRF	J233921.1+601011	2336+598	D	23 39 21.12519661	+60 10 11.8495795	1.182000)E-5	8.640000E-5	-
522	ICRF	J235600.6-682003	2353-686	D	23 56 00.68142337	-68 20 03.4720105	8.290000	0E-6	5.030000E-5	-
523	ICRF	J235622.7+815252	2353+816	D	23 56 22.79391486	+81 52 52.2550985	1.919000	9E-5	4.080000E-5	-
527	ICRF	J235753.2-531113	2355-534	D	23 57 53.26605016	-53 11 13.6895021	3.890000)E-6	3.850000E-5	
529	ICRF	J235810.8-102008	2355-106	D	23 58 10.88239662	-10 20 08.6114174	2.100000	0E-6	3.140000E-5	-
535	ICRF	J235933.1+385042	2356+385	D	23 59 33.18079487	+38 50 42.3182764	2.640000	0E-6	3.160000E-5	-
1	ICRF	J000020.3-322101	2357-326	IN	00 00 20.39997606	-32 21 01.2337415	8.040000	9E-6	0.00026	-
2	ICRF	J000027.0+030715	2357+0 🔴		New	Subset		i-5	0.00034	-
3	ICRF	J000053.0+405401	2358+4					-5	0.00027	-
4	ICRF	J000105.3-155107	2358-1		New Subset Nam	e: icrf-3 defining	0	-6	0.00023	-
5	ICRF	J000107.0+605122	2358+6	1		e. Ich-5 denning	<u> </u>	-5	0.00019	
6	ICRF	J000108.6+191433	2358+1	Ś	Add Subset			-6	4.720000E-5	-
7	ICRF	J000211.9-215309	2359-2:	-	Add Subset			-5	0.00047	-
8	ICRF	J000315.9-194150	0000-1		Add and Set (urrent Subset		-6	0.0003	-
9	ICRF	J000318.6-192722	0000-1		Add and Set C	urrent subset		-5	0.00043	-
10	ICRF	J000319.3+212944	0000+2		Trees with Carb			-6	0.00017	-
11	ICRF	J000327.2-154705	0000-1		Transmit Sub	set 📎 All Clien	ts (0) ᅌ	-5	0.00042	-
12	ICRF	J000346.0+480704	0001+4					-5	0.0005	-
13	ICRF	J000404.9-114858	0001-1				Cancel	-6	0.00015	-
14	ICRF	J000416.1+461517	0001+4				cancer	-5	0.00014	1
16	ICRF	J000435.7+201942	0002+2					-6	0.00014	-

Figure 8: Creating subset with defining sources.

- 2. To select a specific rows in the table click on the box indicated by the blue arrow. Otherwise click on the box indicated by the black arrow. After selecting all the sources with "D", we will create a subset with name icrf-3 defining. Click on Add Subset.
- 3. Now create a subset with ICRF-3 non defining sources.
- 4. Now when we click on <u>the Row Subset</u>, a window like Figure 9 should appear.

5. Now if we select **icrf-3 defining**, TOPCAT will return the number of defining sources. You should get a number 303.

	TOPCAT
	🗄 👁 Σ 🐺 🛄 🛄 ⊕ 🚱 \ominus 💐 🖼 🂢 🤇 🖉 🖲
Table List	Current Table Properties
2: I_323_icrf2 3: icrf3.txt	Label: icrf3.txt
	Location: /Users/Sayan/Desktop/Desktop/icrf3.txt
	Name:
	Rows: 4,536
	Columns: 15
	Sort Order: 🕂 Status ᅌ
	Row Subset: 🗸 All
	Activation Actions: icrf-3 defining
	icrf-3 non defining
	SAMP-
209 / 911 M	Messages: Clients: 💿 🌺

Figure 9: TOPCAT subset window

- 6. Now make an all sky plot with only defining sources.
- 7. However, we can do a further interesting analysis. Here you will plot all the defining sources in the declination range $[-30^{\circ}, -90^{\circ}]$.
 - (a) Select icrf-3 defining in the row subset and DEJ2000 from Sort Order. We will look at the table.
 - (b) The table now should only show sources with "status" D.
 - (c) Now select declination in the mentioned range and create a new subset.
 - (d) TOPCAT should return a number 78 when selecting the newly created subset from Row Subset.
 - (e) How does the number of defining sources in the Southern Hemisphere compared to the Northern Hemisphere?
- 8. Since we are by now comfortable with creating subsets and all sky plots, we will now do some scatter plots using TOPCAT. In this section, we will:
 - (a) Look at the formal error in the source positions in ICRF-3 catalogue.
 - (b) Learn how to add new columns in TOPCAT.
- 9. Now, we will not select anything in Sort Order and Row Subset.

- 10. Again look at the table. This time we are interested in columns D_RA (unit sec) and D_DEC (unit-arcsec).
 - (a) We will convert RA and DEC columns into mas and take *log10* of them. Finally create two new columns with these new values in mas for RA and DEC.
 - (b) We will also take *log*10 of the column "Obs" and create a new column.
 - (c) Figure 10 is an example of this conversion and creating a new column.
 - (d) How will you convert the D_RA column which is in units of seconds to mas and create a new column.
 - (e) Do the same for column "Obs".

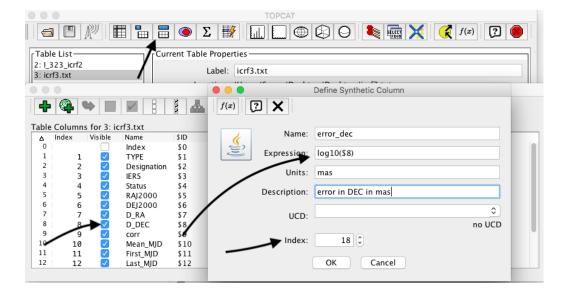


Figure 10: How to convert an existing column and create a new column.

(f) Finally we will get a plot like in Figure 11. Do the same for the formal error in RA vs. Obs.

By now you all should be familiar with (1) all sky projection plot, (2) creating subsets and (3) creating new columns. Now make an all-sky projection plot of the sources with source position errors: > 1 mas, between 0.2 mas & 1 mas and < 0.2 mas (Figure 12).

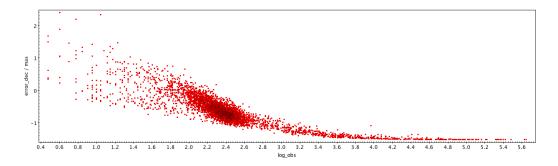


Figure 11: Distribution of the formal error in declination in ICRF-3 sources.

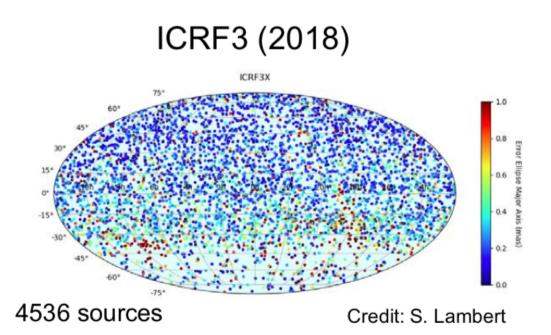


Figure 12: ICRF-3 S/X source distribution with source position error >1 mas, in between 0.2 and 1 mas and < 0.2 mas. Plot courtesy: Dr Patrick Charlot.

We completed the basic TOPCAT session on handling a source catalogue. However, there is one more thing that TOPCAT can do for us. A "session" in any desired format can be saved so that we do not lose our hard work.

Some Hint for converting the error in RA from sec to mas: $24 \text{ hr} = 360^{\circ}$ 1 hr = 15° 1 sec = $\left(\frac{1^{\circ}}{3600}\right) \times 15$ or, 15 arcsec (1 arcsecond is equal to exactly one thirtysix-hundredth arc dergees).

Note: $1 \operatorname{arcsec} = 10^3 \operatorname{mas}$

Python

In this part, the aim is to produce an all sky source distribution plot like the one available on the ICRF webpage (Figure 13). However instead of plotting defining and non defining sources, we will plot all the sources.

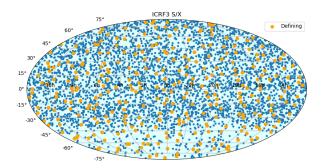


Figure 13: ICRF-3 source distribution at S/X-band. Image source: http://hpiers.obspm.fr/icrs-pc/newww/icrf/index.php

But this time, Python will be used. You are free to use Python, iPython, Jupyter Notebook for this exercise. In the terminal you can type: Python or ipython –pylab or jupyter notebook.

As a very first step, you will learn how to make a basic aitoff projection using a random RA and DEC points using a jupyter notebook.

Jupyter Notebook

In your terminal type:

- 1. \$ ls and cd pymc3_exercises.
- 2. cd icrf3_exercise, there should be a jupyter notebook file with .ipynb extension.
- 3. In your terminal, type: jupyter notebook <filename>

Apart from numpy and matplotlib there is a new library called astropy.

"The Astropy Project is a community effort to develop a core package for astronomy using the Python programming language and improve usability, interoperability, and collaboration between astronomy Python packages."

As we are working with coordinates and units (e.g. degree), astropy.coordinates and astropy.units were imported respectively.

After running the script, what is the output?

Note: In this very first script, very simple RA and DEC coordinates have been used which are in units of degree. However, from the TOPCAT exercise we have noticed that the RA coordinates are in hh:mm:ss format.

Python script for all-sky projection plot

You will try to write a script that will produce a similiar plot as in Figure 14.

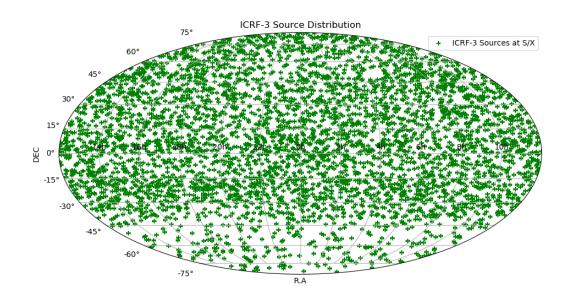


Figure 14: Source distribution in the ICRF-3 catalogue.

Guide to the script:

First, use **mkdir** to create a directory <u>icrfplot</u>. Then **copy** icrf3.txt into this folder from the TOPCAT folder. Now, create a python script with name icrfplot.py.

- 1. Import numpy and matplotlib.
- 2. Use astropy.coordinates and astropy.units.
- 3. Use **astropy.io** to work with ASCII file.
- 4. Once done, load the file using data = ascii.read("filename").
- 5. Now do **Print** to see what's inside the catalogue.
- 6. Print only **RAJ2000 and DEJ2000** columns.

- (a) **RA** column is in hh:mm:ss unit. You need to convert it to degree.
- (b) To do so: ra = coord.Angle(< *RAJ2000* >, unit=u.hour) and then ra=ra.wrap_at(180*u.degree).
- (c) If you do **print(ra.degree)** it should return RA into degree unit now.
- (d) To deal with DEC column: dec=coord.Angle(< DEJ2000 >, unit=u.degree).
- 7. Since the conversion of the RA and DEC are done, you will start working now on making the plot. For this:
 - (a) fig=plt.figure(figsize=(8,6)).
 - (b) **ax=fig.add_subplot(111, projection="mollweide")**.
 - (c) Please add some title to the plot using: plt.title("ICRF-3 S/X Source Distribution").
 - (d) Now do: ax.scatter(ra.radian, dec.radian, c="r", marker="+", label="ICRF-3 Sources at S/X").
 - (e) You need to put some labels: ax.set_xticklabels(['14h', '16h', '18h', '20h', '22h', '0h', '2h', '4h', '6h', '8h', '10h']).
 - (f) Put some grid: **ax.grid(True)**.
 - (g) plt.xlabel("R.A", fontsize=10), plt.ylabel("DEC", fontsize=10)
 - (h) plt.legend(loc="upper right")
 - (i) plt.show()
- 8. To save the plot: fig.savefig("choice of directory with plot name").