



## JUMPING JIVE

Project ID: 730884

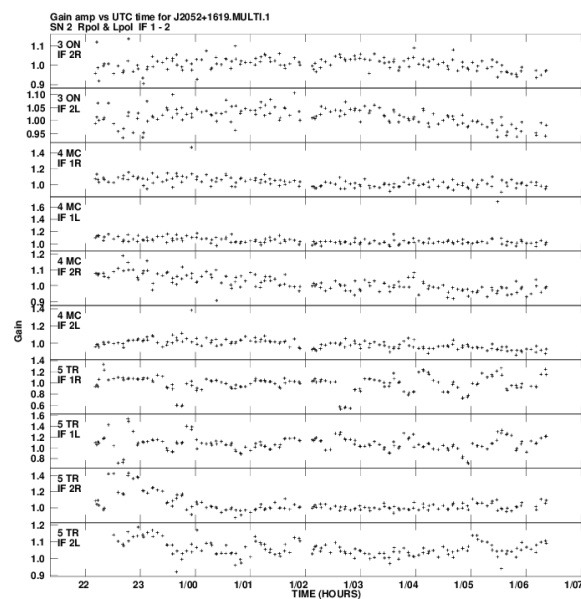
Pipeline and EVN Archive interfaces for monitoring telescope performance

<i>Deliverable:</i>	D5.2
<i>Lead beneficiary:</i>	JIV-ERIC
<i>Submission date:</i>	05.06.2020
<i>Dissemination level:</i>	Public

## 1. Background

The EVN feedback facility is the primary means through which participating EVN telescopes report on their success in individual observations. It is driven by a php program, which presents a web form for completion at the end of the observations. There are sections for identification (e.g., telescope, observation code, reporting person) and for self-assessment of the overall result (success, minor failure, failure) plus an enumeration of specific problems that have been known to occur, which may be individually selected as appropriate, and a space for free-text descriptions. Feedbacks from individual telescopes are bundled together per observation, each stored in a flat, html-formatted file that can be viewed publicly via links on the EVN web page. The indexing that the user encounters is by session, then by observation.

The EVN Archive serves as the primary means through which EVN users access their data. The Archive is reachable from both the EVN and JIV-ERIC web pages, with the top-level indexing again by observation. The Archive contains the FITS files resulting from correlation, plus diagnostic plots made during the post-correlation review and a summary letter by the JIV-ERIC support-scientist for the observations. Most relevant to this task, the Archive also houses the results of the EVN pipeline, providing among other things calibration tables for the user to apply. The pipeline also performs some iterations of self-calibration on selected sources, whose resultant gain corrections (multiplicative factors applied to the a priori gain values derived from telescope-provided data) would be of direct interest to the telescopes. Figure 1 presents an illustration of such a time-series of gain-correction factors.



*Figure 1. Gain correction factors resulting from self-calibration performed in the EVN pipeline. This observation had two baseband channels per polarization; this extracted page shows two entire stations (Mc, Tr) and the second BBC for a third (On).*



During meetings of the EVN Technical and Operations Group, held typically at 7- to 9-month intervals, JIV-ERIC support scientists give presentations about recent telescope performance and amplitude calibration results to the assembled technical staff from the telescopes. Figure 2 shows an example table of median absolute fractional gain errors per telescope per frequency band within an EVN session, as presented at a recent EVN TOG meeting.

	18cm	6cm	5cm
<b>AR</b>	0.006 (1)		
<b>BD</b>	0.14 (10)	0.11 (19)	
<b>EF</b>	0.12 (12)	0.30 (14)	0.04 (2)
<b>HH</b>	0.09 (13)	0.11 (18)	
<b>IR</b>	0.20 (2)	0.40 (21)	0.07 (2)
<b>J1</b>	0.03 (2)	0.06 (2)	
<b>J2</b>	0.30 (14)	0.40 (21)	0.80 (2)
<b>KM</b>		0.30 (6)	
<b>MC</b>	0.20 (14)	0.15 (25)	0.04 (2)
<b>NT</b>		0.15 (19)	0.50 (2)
<b>ON</b>	0.10 (15)	0.13 (18)	0.04 (2)
<b>RO</b>	0.58 (3)		
<b>SV</b>	0.10 (10)	0.11 (19)	
<b>T6</b>	0.30 (9)	0.19 (16)	
<b>TR</b>	0.40 (10)	0.20 (19)	0.04 (2)
<b>UR</b>	0.17 (8)	0.15 (9)	
<b>WB</b>	0.20 (14)	0.19 (25)	0.20 (1)
<b>YS</b>		0.09 (25)	0.05 (2)
<b>ZC</b>	0.30 (9)	0.40 (16)	

Figure 2. Median absolute gain errors for three frequency bands within a session. The number of experiments upon which each entry is based is in parentheses.

From the viewpoint of the telescopes' technical staff, the current arrangement of the EVN Archive and EVN feedback facility is not ideal. Indeed, they were originally designed with the EVN user in mind, and thus the guiding organizational structure has been based on individual observations. But these structures have proven to be less convenient for participating EVN telescopes when trying to investigate their own performance over time, and to not lend themselves well to interrogations. The TOG meetings, which do extract and present telescope-specific results, may not be frequent enough for the telescopes' operational use. This task developed additional means of databasing telescope-feedback and amplitude-calibration data, as well as more direct paths for telescope operators to make standard or custom queries.



## 2. Design of a New Approach

To overcome these limitations in the current organization of the Archive and the feedback facility, with a view towards a system more convenient for participating EVN telescopes to investigate their own performance over time, the two principal design decisions centred on how to store the relevant data and how to provide more flexible access to them.

For the first aspect, a new MySQL database was created on an existing JIV-ERIC database server, containing separate tables for the telescope feedback and amplitude-calibration data. The field structure was established with considerations of being able to maintain the feedback-submission procedures for the telescope operators while opening up other possible avenues of entry. Reporting from the database would at a minimum parallel the existing kinds of reports. Figure 3 shows a brief description of the structure of the tables.

Field	Type
id	int(11)
exp	varchar(10)
obs_date	datetime
source	varchar(20)
telescope	varchar(10)
frequency	decimal(10, 8)
correction	decimal(10, 8)

Field	Type
feedback_id	int(11)
exp	varchar(10)
site	varchar(20)
friend	varchar(40)
date	datetime
frequency	double
session	varchar(20)
session_long	varchar(40)
result	varchar(20)
problem	varchar(100)
text	varchar(500)
wbtype	varchar(20)
jbtype	varchar(20)

Figure 3. Fields within the new database tables to store amplitude-calibration and telescope-feedback data.

After investigation of various options, the Grafana system was the choice for the front-end interface to the new database. Grafana is a relatively new open-source web-based platform for data analytics, with a natural application to time-series data. Because the information in the new database can be thought of as essentially various time series per telescope, Grafana started high in the list of packages considered. It also proved that it connects well to the relational MySQL database. Grafana's strengths include the ability to craft dashboards with graphs and tables, and also permitting the telescope staff to create/save their user-defined dashboards, or even compose their own custom SQL queries, through (read) access to the underlying database.



### 3. Implementation and Operations

A key goal of implementing the feedback database was to preserve the per-observation submission mechanism for the telescopes, which is via a php-driven web form. That php program now also parses the telescope input and directly connects to the MySQL database to load the extracted information. It also scrapes the feedback web page for information about the observation not explicitly contained in the telescope's input, such as the date and the frequency band. A completely new python program parses the summary letter written for each observation by a JIV-ERIC support scientist to extract comments about individual stations. For each such comment, the support scientist is prompted to classify the consequence in the scale of [success | minor failure | failure], or whether to skip loading it to the database altogether.

There are now two independent paths for telescope feedback to enter into the database, one with the advantage of post-correlation hindsight. In those cases where the telescope operator and support scientist have disagreed on the classification of the observation, the support scientist's assessment takes priority, but both are loaded into the database. Historic telescope feedback information has been loaded back to November 2002. Figure 4 shows the telescopes' success rate for the October/November 2019 EVN session (the most recent one completely correlated/distributed) from the generic Feedback dashboard.

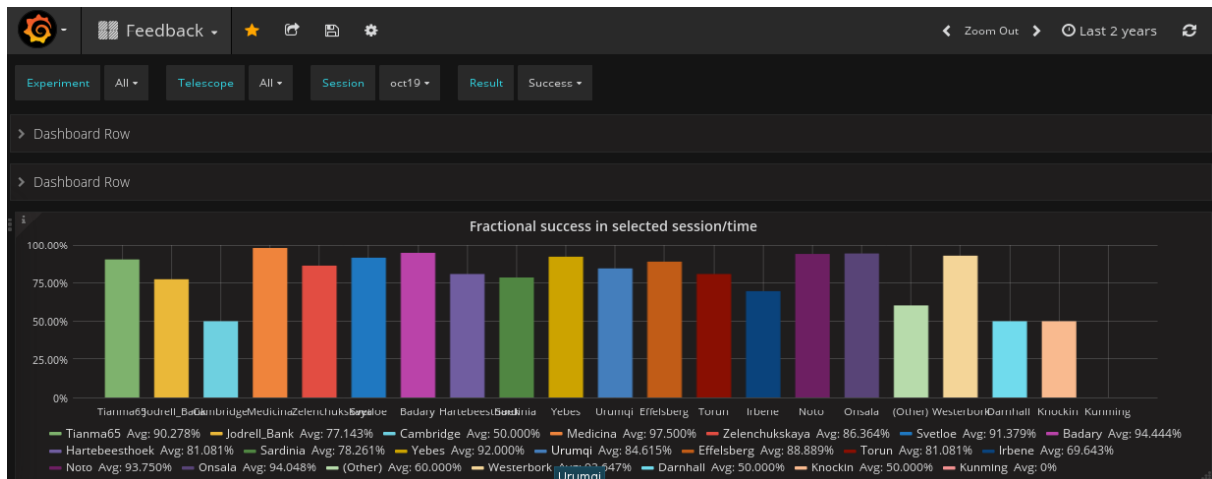


Figure 4. Snapshot of a portion of the Grafana Feedback dashboard.

The amplitude-calibration database derives from the self-calibration iterations in the EVN pipeline. The pipeline starts with a priori gain information provided by the telescopes; self-calibration essentially treats telescope gains as adjustable parameters, to be optimized in a manner constrained by gain-independent "closure amplitudes" to optimally match the observed visibilities. Results from this self-calibration process become more robust as the number of telescopes increases and the source becomes more point-like. The pipeline for



each observation outputs a separate statistical summary, for each source not designated as a target of a phase-referencing pair, of the derived amplitude-gain corrections per telescope per baseband channel, plus the weighted average over all BBCs.

A new program extracts the median values from these pipeline output files for insertion into the database, limited to only those sources that the JIV-ERIC support scientist selected as bandpass calibrators in the pipeline-control input file. This selection serves as a proxy for including only bright/compact sources. The program also scrapes the correlation-control vex file for the observing date and the determination of which specific telescope was being used for sites with multiple antennas. Historic amplitude-calibration data has been loaded back to November 2005, prior to which point the relevant information was not output by the pipeline on a per-source basis. Figure 5 shows the top portion of the generic Amplitude Calibration dashboard, here covering all telescopes over the past 2 years, and Figure 6 shows the same time-range but only for 18cm observations. The y-axis scan is the absolute fractional gain correction derived from the pipeline self-calibration, with each observation plotted separately. Focusing on a telescope-created dashboard, Figure 7 shows the past five-year results for Effelsberg over all frequencies, and Figure 8 shows the sub-sets of 3.6cm (X-band) and 1.3cm (K-band) observations separately.

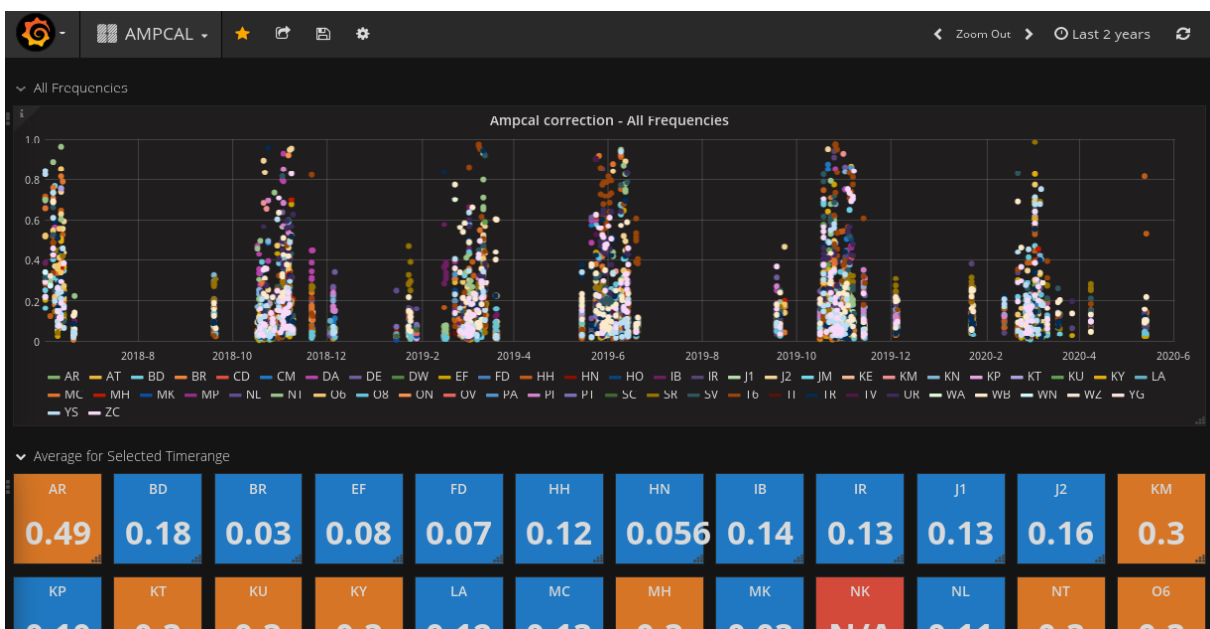


Figure 5. Snapshot of the top portion of the Grafana Amplitude Calibration dashboard, where the plotted time-range covers the past two years.



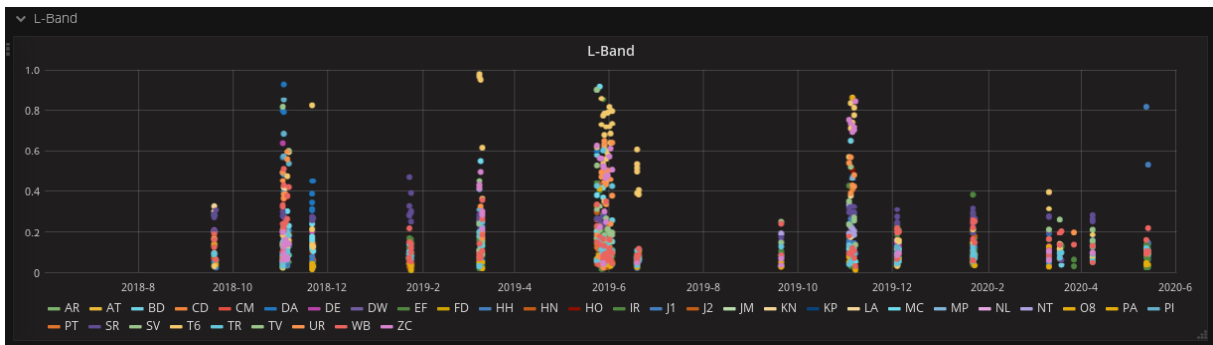


Figure 6. The same time-range of Amplitude Calibration results as in Figure 5, but here only for 18cm observations.

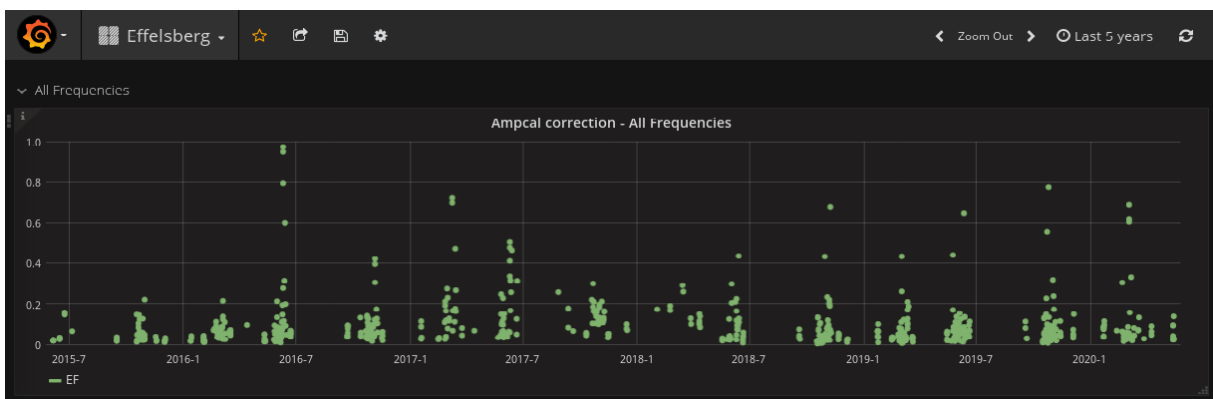


Figure 7. Top panel of Effelsberg's custom-made Grafana Amplitude Calibration dashboard, here over the past five years.



Figure 8. The same time-range of Amplitude Calibration results as in Figure 7, but here only for 3.6cm and 1.3cm Effelsberg observations.



As telescope staff provide their experiences with these new Grafana-fronted feedback and amplitude-calibration tools, the cycle of suggestions and improvements can continue to drive their evolution. Topics currently under investigation include storing/accessing amplitude-calibration also by channel, incorporating the sign of the fractional gain correction (i.e., whether the factor was greater than or less than unity), and refining the feedback "result" categorization together with updating the roster of specific problems encountered.

## 4. Related Developments

There have been some recent developments somewhat outside the narrow purview of this deliverable, but which have worked to increase the robustness of the amplitude calibration of the EVN telescopes.

The `antabfs` program is used by telescopes operators to read their field-system log files and self-maintained "rxg" files in order to extract the necessary information to assemble a table of system temperatures per baseband channel along with their gain curves and gains per IF. Such an output file is fed directly into image analysis packages by the astronomer (as does also the EVN pipeline). This was originally developed in perl. Engineers at Yebes have recently translated `antabfs` to python, and have added fuller support for the new digital back-ends being used by most EVN stations and with ability to handle continuous-calibration systems, which provides more reliable system-temperature data and removes the need for the PI to worry about scheduling sufficient gaps for firing a calibration-diode.

The newer python version cycles through the baseband channels, opening a plot window showing the derived system temperature plus a fit and bounds, with outliers appearing in red. The user can interactively edit out specific (sets of) system-temperature points by making drag-n-click boxes. When happy with this baseband channel, the user closes the plot window to proceed on to the next. Figure 9 shows the initial plot window for a well-behaved baseband channel; Figure 10 shows the initial plot window and one after a few iterations of editing for a BBC heavily affected by RFI (both BBCs from the same observation).





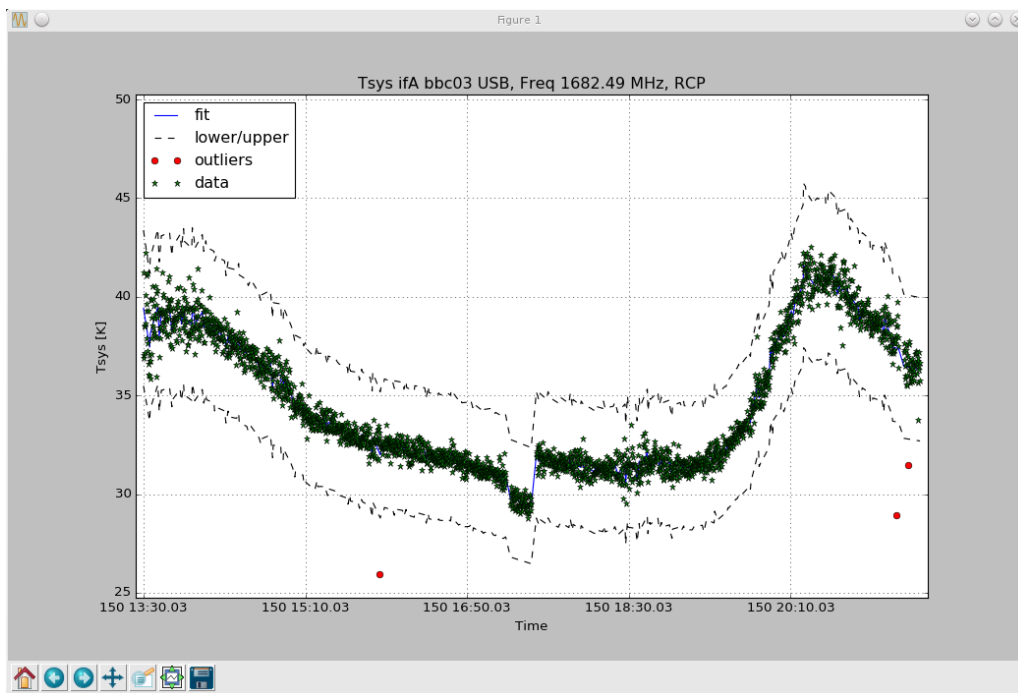


Figure 9. *antabfs.py* initial plot of system temperature for a well-behaved BBC, from an 18cm observation.

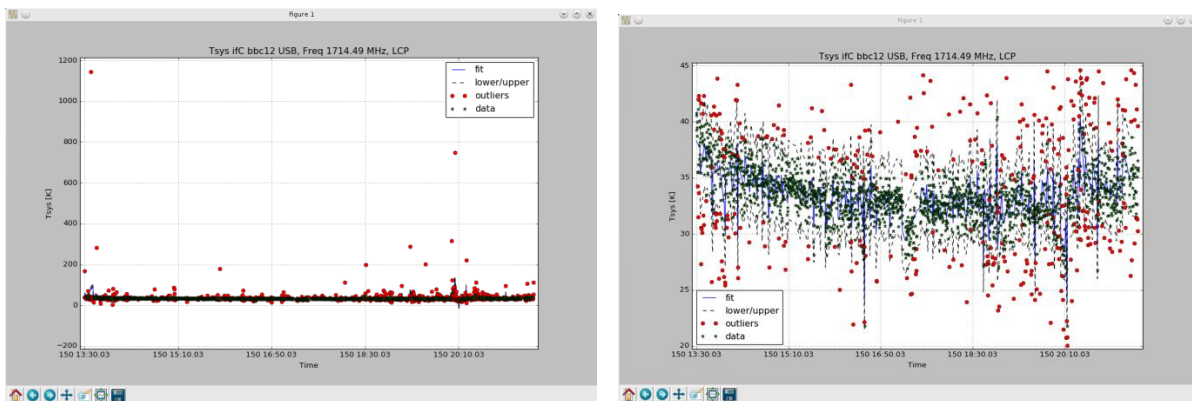


Figure 10. *antabfs.py* initial plot of system temperature for a BBC with significant RFI, plus the plot for the same BBC following some iterations of editing away outliers.

When JIV-ERIC support scientists run the EVN pipeline using the antab files provided by the telescopes, they sometimes encounter problems that need correcting. The root of such problems can often be easy to overlook when dealing with the antab file in its native text format. JIV-ERIC has recently written *antab\_editor.py*, which offers the support scientists a more graphical-based approach. Figure 11 illustrates the gain-curve plots computed within *antab\_editor.py*. Within the antab files gain curves are parameterized as polynomials with respect to a specified angular measure (e.g., elevation); experience has shown that telescopes can swap the order of their polynomial coefficients, or even have a badly delimited list -- the effect of either would be readily apparent in such plots. Figure 12 shows an example of a telescope's dashboard within *antab\_editor.py*, with a plot similar to the final



all-channel plot from antabfs.py on the left, and a tabular summary of header and system-temperature data on the right, along with some "action" buttons that facilitate editing the system-temperature table.

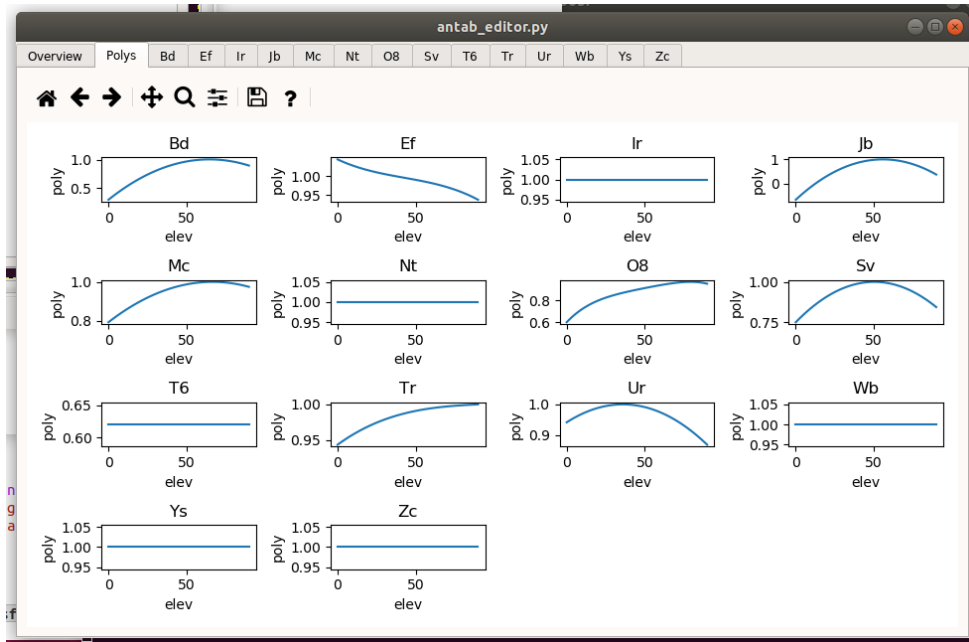


Figure 11. antab\_editor.py plot of gain curves as encoded by telescopes' antab files.

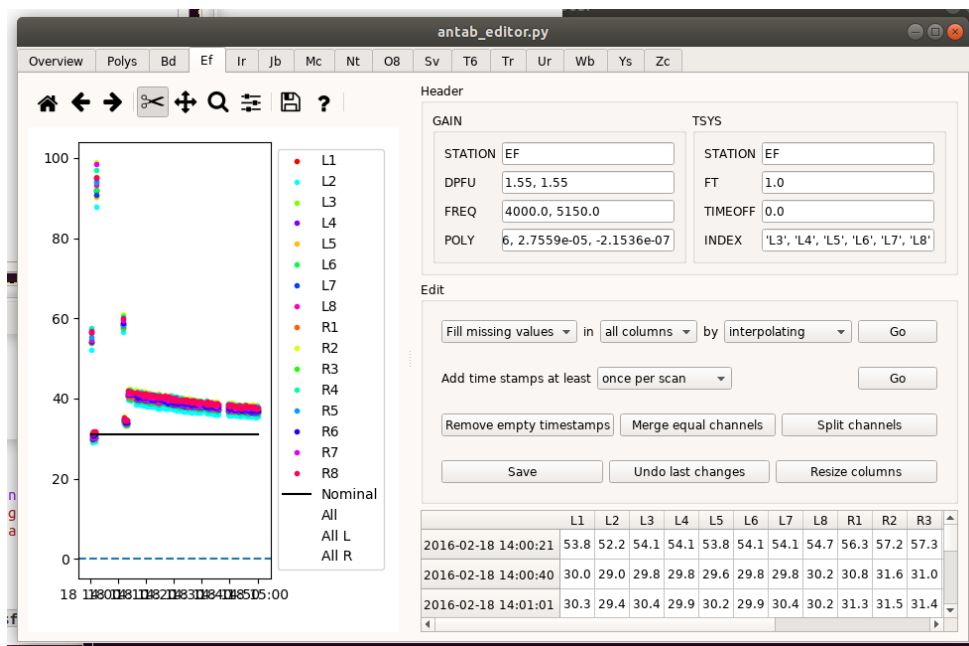


Figure 12. A telescope dashboard from antab\_editor.py.

