

# VLBI Session VO2187: Comparison of Pseudo-I Products Before and After Polarization Conversion

L. V. Benkevitch<sup>1</sup>

<sup>1</sup>MIT Haystack observatory, Westford, MA 01886, USA.

March 5, 2025

## Abstract

The VGOS data are collected with the use of linearly polarized receivers. However, circular polarization has a number of benefits. The PolConvert software allows VGOS data conversion from linear to circular polarization. In this study we do a statistical comparison of the pseudo-Stokes data before and after the conversion in order to characterize the changes PolConvert introduces.

## Contents

<b>1</b>	<b>VGOS Data</b>	<b>1</b>
<b>2</b>	<b>Parameter Temporal Variations Before and After PolConvert</b>	<b>2</b>
<b>3</b>	<b>Parameter residual Statistics Before and After PolConvert</b>	<b>4</b>
<b>4</b>	<b>Conclusion</b>	<b>5</b>
<b>5</b>	<b>Appendix: Software</b>	<b>7</b>
5.1	make_sorted_idx_2187.py: Saving VGOS data in Python dictionaries . . . . .	7
5.2	group_tails.py: Group Tail Bins with Sparse Data . . . . .	8
5.3	Plot Parameter Temporal Variations . . . . .	9
5.4	Plot Parameter Histograms . . . . .	9
<b>6</b>	<b>Appendix: Tables</b>	<b>11</b>
<b>7</b>	<b>Appendix: Figures</b>	<b>15</b>

## 1 VGOS Data

This study is based on the data acquired during the VLBI session started on 2022-07-06, 18:00 and ended on 2022-07-07, 18:00 with the antennas of the eighth stations listed in Tab. 1.

Table 1: Stations in the Schedule for the Session VO2187

Code		Name	Description
1-ltr	2-ltr		
E	Wf	WESTFORD	Westford, MA, USA
G	Gs	GGAO12M 1	Goddard Geophysical and Astronomical Observatory (GGAO)
H	K2	KOKEE12M	Kokee Park Geophysical Observatory
I	Is	ISHIOKA	Ishioka VLBI Station
M	Mg	MACGO12M	McDonald Geodetic Observatory (MGO), TX, USA
S	Oe	ONSA13NE	Onsala 13m antenna northeast, Sweden
T	Ow	ONSA13SW	Onsala 13m antenna southwest, Sweden
Y	Yj	RAEGYEB	13m at Yebes, Spain

## 2 Parameter Temporal Variations Before and After Pol-Convert

Converting linear polarization of the source data to the circular polarization with the PolConvert software introduces changes into its parameters. Here we consider the changes in multi-band delays (MBD), single-band delays (SBD), and signal-to-noise ratios (SNR) for the Stokes polarization products, of which we only select the pseudo-Stokes  $I$  parameters as functions of time,  $t$ . The the original VO2107 data set in SWIN (DiFX) format obtained from the stations with linearly polarized antennae is converted into two data sets in the fringe-fit Mark4 format: with linear and circular polarizations. The linearly polarized Mark4 fringe-fit data set is resulted from `difx2mark4` followed by `fourfit` conversions. The circularly polarized Mark4 fringe-fit data set requires additional step. First, the PolConvert software converts the original linearly polarized SWIN (DiFX) data set into the circularly polarized SWIN (DiFX) data set. Then the same toolchain of `difx2mark4` and `fourfit` convert it into the circularly polarized Mark4 fringe-fit data set. From each of the two data sets we read pseudo-Stokes  $I$  of MBD, SBD, and SNR. Below, working with each of these parameters, for  $I$  in the two different polarizations we use the names  $\text{LinI}(t)$  and  $\text{CirI}(t)$ . The data are presented in the discrete, numerical format, so instead of the continuous time,  $t$ , integer indices are used, like  $k = \overline{1..N}$ , where  $N$  is the number of data points in the sample. The following Figures show temporal variations of the  $\text{LinI}[k]$  and  $\text{CirI}[k]$  parameters during the experiment before and after PolConvert, one graph for each baseline:

MBD variations in Fig. 1;  
SBD variations in Fig. 4;  
SNR variations in Fig. 7.

One can see that most of the graphs after PolConvert,  $\text{CirI}[k]$ , are significantly biased with respect to  $\text{LinI}[k]$ . The biases calculated as  $\text{LinI}[k] - \text{CirI}[k]$  are plotted in the Figures below:

MBD bias variations in Fig. 2;  
SBD bias variations in Fig. 5;  
SNR bias variations in Fig. 8.

For each of the biases the bias mean,  $\overline{\text{Bias}}$ , is computed.

However, removing the bias by subtracting the mean values of each curve makes the differences much less significant:

$$\text{LinI}[k] = \text{LinI}[k] - \text{mean}(\text{LinI}[k]) \quad (1)$$

$$\text{CirI}[k] = \text{CirI}[k] - \text{mean}(\text{CirI}[k]) \quad (2)$$

In order to compare the unbiased  $\text{LinI}[k]$  with  $\text{CirI}[k]$ , the residual,  $\text{res}[k]$ , is computed as their difference:

$$\text{res}[k] = \text{LinI}[k] - \text{CirI}[k] \quad (3)$$

The graphs of residuals with the means subtracted are shown in the following Figures:

MBD residual variations in Fig. 3;  
SBD residual variations in Fig. 6;  
SNR residual variations in Fig. 9.

To assess the similarity of  $\text{LinI}[k]$  and  $\text{CirI}[k]$ , the following parameters are computed:

- Pearson correlation coefficient  $r_{corr}$ ;
- Root Mean Square Error, RMSE;

Since  $\text{LinI}[k]$  and  $\text{CirI}[k]$  already have their means subtracted, the correlation formula is simplified:

$$r_{corr} = \frac{\sum \text{LinI}[k] \cdot \text{CirI}[k]}{\sqrt{\sum \text{LinI}[k]^2 \cdot \sum \text{CirI}[k]^2}} \quad (4)$$

One can see for most of the baselines the coefficient of the correlation between  $\text{LinI}$  and  $\text{CirI}$  shows high values around 0.9 and above.

The root mean square error, RMSE, in our case is actually the standard deviation of the residual,  $\sigma$ :

$$\text{RMSE} = \sigma = \sqrt{\frac{1}{N} \sum \text{res}[k]^2} \quad (5)$$

Lower RMSE indicates a better fit between the curves. However, RMSE is not an “absolute” indicator of the curves’ proximity. It needs to be related to the absolute values of both  $\text{LinI}$  and  $\text{CirI}$ . They are shifted, however, to very roughly juxtapose RMSE with the absolute values of the MBD, SBD, and SNR parameters (here denoted as  $\text{PRM}$ ) we have introduced their “absolute averages”:

$$\overline{\text{PRM}} = (|\text{mean}(\text{LinI})| + |\text{mean}(\text{CirI})|) / 2. \quad (6)$$

To compare the observed values of the parameters  $\text{LinI}$  and  $\text{CirI}$  with their unbiased residual RMS error, we write it in the table format with the columns including the time-averaged signal-to-noise ratio  $\overline{\text{SNR}}$ ,  $\overline{\text{PRM}} \pm \text{RMSE}$  or  $\overline{\text{PRM}} \pm \sigma$ , the relative error (%), the time-averaged bias

between LinI and CirI,  $\overline{\text{Bias}}(ps)$ , and the correlation coefficient,  $r_{corr}$ , for each of the baselines. The relative error between the unbiased LinI and CirI can be expressed as

$$\text{relerr}(\%) = \text{RMSE}/\overline{\text{PRM}}. \quad (7)$$

These characteristics are summarized in the Tables Appendix:

MBD PolConversion parameters in Table 2;  
SBD PolConversion parameters in Table 3;  
SNR PolConversion parameters in Table 4.

### 3 Parameter residual Statistics Before and After PolConvert

In the previous Section MBD, SBD, and SNR pseudo-Stokes  $I$  parameter data were shown as functions of time. Here we conduct their statistical study. In order to estimate the errors introduced by PolConvert, histograms of the unbiased parameter residuals (after the mean subtraction, see Eqs. (1) and (2)) are plotted. Below are the histograms for all of the baselines:

MBD residual histogram for all of the baselines in Fig. 10;  
SBD residual histogram for all of the baselines in Fig. 12;  
SNR residual histogram for all of the baselines in Fig. 14.

It is interesting to see which of the individual stations contribute to the errors. We plotted histograms of the parameter residuals for the baselines involving each particular station. They are shown in the following Figures:

MBD residual histograms for the baselines including one station in Fig. 11;  
SBD residual histograms for the baselines including one station in Fig. 13;  
SNR residual histograms for the baselines including one station in Fig. 15.

In order to evaluate the significance of the error we attempted to use the Pearson's chi-squared test comparing the histograms to the normal distributions:

$$f(\mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}. \quad (8)$$

All the histograms of residuals are computed for 41 bins. However, the histograms have long tails filled with zeros or small numbers because the data have occasional outlier points far from their means. Plotting such distributions with all the bins makes the histograms too narrow and their most significant portions near the means are poorly visible. Therefore, to plot them and to perform the Pearson's  $\chi^2$  test, the left-tail and right-tail bins with sparse data (frequencies less than 8) were grouped and the test used fewer number of bins  $N_b$  (see it printed in each plot, Fig. 10 - Fig. 15). The Pearson's  $\chi^2$  test compares the observed data counts in the bins  $B_i$  with the expected (theoretical) counts  $E_i$ , where  $i = \overline{1..N_b}$ , obtained from the normal PDF (8) as

$E_i = Np_i$ . Here  $p_i$  is the probability that a random value sampled from  $f(\mu, \sigma)$  is within the  $i$ -th interval. The value of the test-statistic is calculated as

$$X^2 = \sum_{i=1}^{N_b} \frac{(B_i - E_i)^2}{E_i}, \quad (9)$$

where  $N_b$  is the number of bins. The  $X^2$  test statistic asymptotically approaches the  $\chi^2_{df}$  distribution, where  $df$  is the number of degrees of freedom:

$$df = N_b - k - 1, \quad (10)$$

where  $k = 2$  is the number of parameters estimated from the data, mean  $\mu$  and standard deviation  $\sigma$ . The additional  $-1$  is due to the fact that the  $\chi^2$  test itself has a restriction imposed by the sum of the observed frequencies being equal to the sum of the expected frequencies.

If we want to be 95% confident that the data is distributed normally,  $X^2$  cannot exceed the critical value  $\chi^2_{cr}$ , such that the probability for  $X^2$  to appear within the interval  $[0 \dots \chi^2_{cr}]$  is  $p = 0.95$ . Hence we should set the level of significance  $\alpha = 1 - p = 0.05$ . The critical  $\chi^2_{df}$  value can be calculated as the value of  $\chi^2$  Probability Point Function (PPF), the inverse of the  $\chi^2$  Cumulative Distribution Function or CDF. In Python this is done as `scipy.stats.chi2.ppf(1-alpha, df)`. Alternatively, the Inverse Survival Function (ISF) may be used as `scipy.stats.chi2.isf(alpha, df)`.

Unfortunately, in our case the Pearson's  $\chi^2$  test fails, because the residuals are grouped around their means substantially denser than the normal distributions with the same standard deviations. In most distributions, the calculated values of the histogram chi-squares are many times greater than the critical chi-square values for the p-values less than or equal to 0.05 (printed in each of the histogram plots). So, we had to use another method to assess how close are the parameter values before (LinI[i]) and after (CirI[i]) the polarization conversion. Since most of the residuals LinI[i] - CirI[i] are within  $\pm\sigma$  (standard deviation), we compute their proportion and compare it to the expected proportion under a normal distribution, which is about 68.27%. The proportion is printed near each histogram. It is at the level of about 80%-90% and more, which is better than the standard normal distribution for all the histograms considered.

## 4 Conclusion

We have conducted a study to assess how well the PolConvert software makes VGOS data conversion from linear to circular polarization. The pseudo-Stokes data of the Multi-Band Delays (MBD), Single-Band Delays (SBD), and Signal-to-Noise Ratios (SNR) as functions of time before and after the conversion were plotted as functions of time for visual inspection. We also did a statistical comparison of the pseudo-Stokes data of MBD, SBD, and SNR before and after the conversion in order to characterize the variations PolConvert introduces. Two criteria were used to assess the residuals: the Pearson's  $\chi^2$  test and the proportions of residuals within  $\pm\sigma$  compared to the normal distribution.

PolConvert introduces large (and random) biases between the pseudo-Stokes data parameters.

On the other hand side, after the bias removal, the average errors (computed as unbiased rms residual) are very small making tenths of percent for the most of baselines for MBD (see Table 2) and SNR (see Table 4). For SBD the relative error is an order of magnitude greater. The SBD conversion biases are generally larger, too (see Table 3).

The unbiased residuals generally have good distributions, denser than the normal with the same STD, as can be seen in Figs. 10 to 15.



## 5 Appendix: Software

### 5.1 `make_sorted_idx_2187.py`: Saving VGOS data in Python dictionaries

The VLBI Global Observing System (VGOS) database is organized as a directory structure. For our purpose of statistical analysis of a small number of parameters scattered across many directories and files below the root directory of the experiment, this implies significant overhead in opening multiple files and accessing the parameters in each of them. The data files in their names only provide the station or baseline names, and no time or polarization information. For example, extraction of, say, SNR data for a particular polarization product and within a specific time range would require opening *all* the files and accessing their times and polarizations using HOPS API calls.

I wrote a script, `make_sorted_idx_2187.py`, to extract the parameters for statistical analysis for the whole experiment and to put it in a Python dictionary, preserving the temporal order. I call such dictionaries “indices”. The index can be “pickled” and saved on disk. These files are small, in the hundreds of kilobytes to one-two megabytes. The scripts for data analysis and plotting read the index files, unpickle them into the Python dictionaries, and use data from the dictionaries.

A pickle file can be unpickled into a dictionary using the `pickle.load()` function. For example:

```
import pickle
with open('idx2187cI.pkl', 'rb') as finp:
    idx2187c = pickle.load(finp)
```

The script `make_sorted_idx_2187.py` is also a Python module defining the function

`make_idx(base_dir, pol='lin', max_depth=2)` with parameters:

`base_dir`: the directory containing the VGOS data. For example, it may be `Work/2187/`.

`pol`: polarization, 'lin' - linear, 'cir' - circular.

This parameter is used for the data generated by PolConvert.

It converts the polarization product names

'XX', 'XY', 'YX', 'YY', and the lists ['XX', 'YY'] into the correct names 'LL', 'LR', 'RL', 'RR', and 'I', respectively.

`max_depth`: Limits the maximum depth of recursing into the subdirectories of `base_dir`.

`make_idx()` creates and returns the index dictionary with the data from `base_dir`.

The index dictionary has three dimensions: the baseline name, the polarization, and the data proper, including 'time', 'file', 'mbdelay', 'sbdelay', and 'snr'. Consider a particular index named `idx2187I` (experiment 2187, linear polarization). Its first dimension is indexed with the baseline names derived from the set of stations, {'E', 'M', 'S', 'T', 'V', 'Y'}.

The possible first indices are the baseline names:

```
idx2187I.keys()
```

```
dict_keys(['SE', 'VY', 'MV', 'MT', 'TV', 'EY', 'SY', 'TY', 'MS', 'SV',
```

```
'TE', 'EV', 'MY', 'ME']])
```

Each of the baselines is associated with the cross-correlation products and the pseudo-Stokes I parameter. Thus the second index is one of the products. For example, for the 'ME' baseline:

```
idx21871['ME'].keys()
dict_keys(['XX', 'XY', 'YX', 'YY', 'I'])
```

For example, the times, the full data file names, SNRs, multi- and single-band delays for the 'SY' baseline and the 'XY' polarization products from this baseline are contained in the index dictionary under

```
idx21871['SY']['XY']:
```

```
idx21871['SY']['XY'].keys() prints
dict_keys(['time', 'file', 'mbdelay', 'sbdelay', 'snr']).
```

For example, in order to access the multi-band delay data list in the ascending temporal order for the baseline 'SV' and the pseudo-Stokes I, one should issue the following command:

```
mbd = idx21871['SV']['I']['mbdelay'].
```

## 5.2 group\_tails.py: Group Tail Bins with Sparse Data

group\_tails.py is a Python module defining the functions `find_tail_bounds(ni, fni)` and `group_tails(ni [, thr=5])`:

Usage:

```
from group_tails import find_tail_bounds, group_tails
l_idx, r_idx = find_tail_bounds(ni)
ni = group_tails(ni, (l_idx, r_idx))
```

`find_tail_bounds(ni)`:

- finds two indices, left and right, at which the sparse tails of `ni` to be cut after grouping. The tail is considered sparse if it contains values less than or equal to the threshold `thr`.

The input parameter of `find_tail_bounds(ni)` is an array containing a distribution of observed frequencies. The `find_tail_bounds(ni)` function outputs the indices, `(l_idx, r_idx)`, where

- `l_idx` points AT the first `ni` element greater than `thr`, and
- `r_idx` points AFTER the last `ni` element greater than `thr`.

So, `ni[l_idx:r_idx]` is the subarray whose elements are  $> thr$  at either ends.

`group_tails(ni, (l_idx, r_idx))`:

- sums up the sparse tail values in `ni` and adds them to the border values on the either side.
- outputs the array `ni_grp`, the result of the sparse tails grouping of `ni`.

Using the tail indices `(l_idx, r_idx)` one can make the tail grouping on the same size array `fni` of expected (theoretical) frequencies, calculated from a distribution to which `ni` has been fit:

```
fni = group_tails(fni, (l_idx, r_idx)).
```



### 5.3 Plot Parameter Temporal Variations

The script `plot_time_par_2187_2187.py` creates plots for MBD, SBD, and SNR time variations. The data are read from the dictionaries `idx2187lI.pkl` and `idx2187cI.pkl`.

Usage:

```
$ python plot_time_par_2187.py <par> [save],  
    where <par> is either MBD or SBD or SNR (lower case permitted).  
    save (optional): save figures in pdf format.
```

The plots generated are shown in Figures 1 to 9.

### 5.4 Plot Parameter Histograms

The script `plot_st_difhist_2187.py` creates histograms of the residual distributions for MBD, SBD, and SNR. The residuals are differences between the pseudo-Stokes I parameters before and after the PolConvert. The data are read from the dictionaries `idx2187l.pkl` and `idx2187cI.pkl`.

Usage:

```
$ python plot_st_difhist_2187.py <par> [save],  
    where <par> is either MBD or SBD or SNR (lower case permitted).  
    save (optional): save figures in pdf format.
```

The plots generated are shown in Figures 10 to 15.



## 6 Appendix: Tables

Table 2: Pseudo-Stokes I MBD PolConversion Parameters

BL	$\overline{\text{SNR}}$	$\overline{\text{MBD}} \pm \sigma (ps)$	relerr (%)	$\overline{\text{Bias}} (ps)$	$r_{corr}$
GE	102.5	$3225.3 \pm 2.0$	0.06	-686.2	0.999986
GH	91.7	$4111.4 \pm 674.5$	16.40	-869.3	0.959041
GI	53.9	$2414.4 \pm 916.4$	37.96	-951.9	0.946194
GM	104.5	$297.4 \pm 320.7$	107.85	345.1	0.984252
GS	104.8	$534.3 \pm 423.6$	27.61	-225.0	0.980334
GT	113.0	$1982.8 \pm 433.7$	21.87	-1189.7	0.979027
HE	82.2	$710.6 \pm 645.6$	90.86	198.2	0.759337
HM	121.9	$2669.1 \pm 860.4$	32.23	1129.1	0.875116
HS	97.1	$2428.3 \pm 666.3$	27.44	551.5	0.902311
HT	103.6	$1948.1 \pm 698.4$	35.85	-440.9	0.892581
IE	54.1	$163.4 \pm 660.3$	404.18	326.7	0.907950
IH	79.7	$183.0 \pm 1310.7$	716.20	295.7	0.705438
IM	64.1	$2005.6 \pm 1159.0$	57.78	1035.4	0.885539
IS	70.5	$1486.7 \pm 1474.5$	99.18	711.1	0.667996
IT	75.2	$1076.8 \pm 1271.6$	118.10	-137.5	0.721770
ME	94.9	$2404.2 \pm 386.4$	16.07	-977.8	0.872342
MS	98.1	$678.9 \pm 531.6$	78.30	-436.5	0.922517
MT	102.8	$1050.8 \pm 540.8$	51.46	-1402.0	0.913555
SE	97.4	$1587.4 \pm 391.0$	24.63	-481.4	0.916335
TE	107.9	$1072.9 \pm 398.0$	37.09	492.0	0.913627

Table 3: Pseudo-Stokes I SBD PolConversion Parameters

BL	$\overline{\text{SNR}}$	$\overline{\text{SBD}} \pm \sigma$ (ps)	relerr (%)	$\overline{\text{Bias}}$ (ps)	$r_{corr}$
GE	102.5	$2978.3 \pm 84.2$	2.83	319.1	0.956047
GH	91.7	$4690.5 \pm 365.2$	7.79	902.8	0.989360
GI	53.9	$2351.5 \pm 4417.6$	187.87	909.2	0.463318
GM	104.5	$1825.1 \pm 215.0$	11.78	1497.9	0.993014
GS	104.8	$2094.7 \pm 259.0$	12.36	940.5	0.992972
GT	113.0	$2157.8 \pm 206.5$	9.57	566.3	0.995751
HE	82.2	$1485.2 \pm 367.7$	24.76	-579.3	0.967322
HM	121.9	$1753.4 \pm 413.7$	23.59	592.7	0.982415
HS	97.1	$2334.3 \pm 273.7$	11.73	180.0	0.981453
HT	103.6	$2176.8 \pm 261.5$	12.01	-67.3	0.984784
IE	54.1	$189.2 \pm 3996.7$	2112.57	-378.4	0.689708
IH	79.7	$545.2 \pm 8128.4$	1490.91	1090.4	0.484396
IM	64.1	$356.9 \pm 600.5$	168.25	527.4	0.955738
IS	70.5	$1006.4 \pm 4542.5$	451.38	104.3	0.188261
IT	75.2	$1004.2 \pm 6604.8$	657.72	198.8	-0.020218
ME	94.9	$678.5 \pm 236.1$	34.79	-1171.3	0.970733
MS	98.1	$252.4 \pm 241.6$	95.74	-504.8	0.988171
MT	102.8	$380.9 \pm 231.2$	60.69	-761.7	0.988901
SE	97.4	$993.0 \pm 249.0$	25.07	-673.3	0.980868
TE	107.9	$777.2 \pm 248.6$	31.98	-293.0	0.983206

Table 4: Pseudo-Stokes I SNR PolConversion Parameters

BL	$\overline{\text{SNR}} \pm \sigma$	relerr (%)	$\overline{\text{Bias}}$	$r_{corr}$
GE	$102.5 \pm 1.0$	0.99	1.1	0.999915
GH	$91.7 \pm 12.2$	13.28	9.8	0.980171
GI	$53.9 \pm 6.7$	12.45	5.7	0.963876
GM	$104.5 \pm 7.7$	7.37	7.7	0.995619
GS	$104.8 \pm 9.0$	8.63	5.4	0.988439
GT	$113.0 \pm 10.4$	9.17	4.8	0.987994
HE	$82.2 \pm 8.8$	10.67	8.1	0.982994
HM	$121.9 \pm 17.8$	14.61	17.7	0.983780
HS	$97.1 \pm 13.4$	13.75	9.2	0.959090
HT	$103.6 \pm 13.0$	12.59	8.2	0.965008
IE	$54.1 \pm 5.8$	10.76	5.1	0.965831
IH	$79.7 \pm 10.6$	13.35	8.3	0.984451
IM	$64.1 \pm 12.4$	19.35	13.2	0.959665
IS	$70.5 \pm 7.0$	9.94	4.1	0.984968
IT	$75.2 \pm 7.1$	9.47	3.6	0.987480
ME	$94.9 \pm 6.0$	6.31	6.9	0.995813
MS	$98.1 \pm 8.1$	8.21	10.2	0.990928
MT	$102.8 \pm 8.2$	8.01	10.2	0.990570
SE	$97.4 \pm 7.9$	8.06	4.9	0.989331
TE	$107.9 \pm 9.3$	8.57	5.3	0.988900



## 7 Appendix: Figures

VO2187: Pseudo-Stokes  $I$  MBD (ps) vs Time (hours), Lin (blue) & Cir (green) Pol after PolConvert

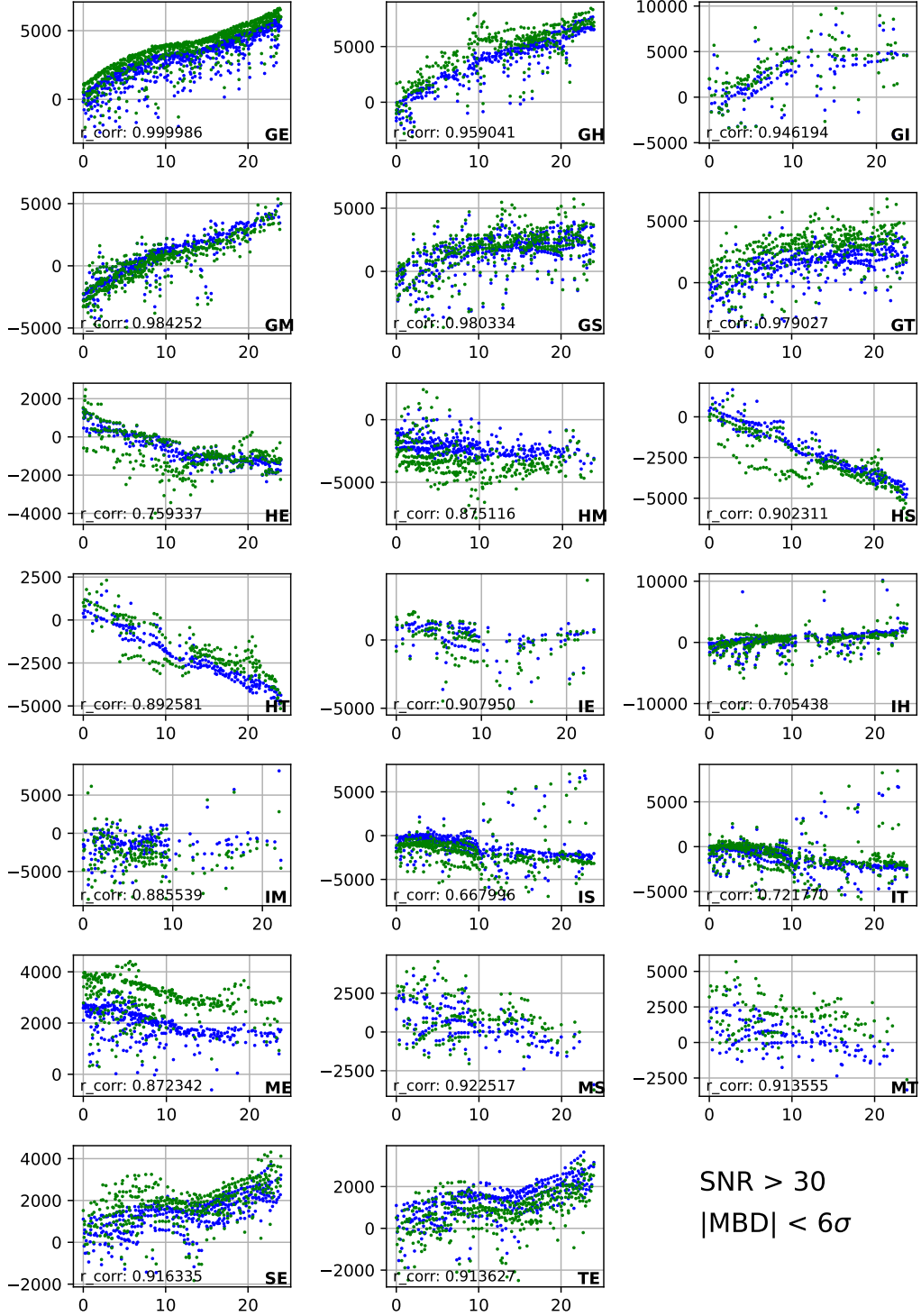


Figure 1: Evolution of Multi-Band Delays (in picoseconds) during the experiment (time in minutes) for the pseudo-Stokes  $I$  parameter for every baseline. Each panel shows two curves: Lin $I$ ,  $I$  from the original, linearly polarized data set and Cir $I$ ,  $I$  from the circularly polarized data set obtained by applying PolConvert to the original data. Graphs have large biases. However, the coefficients of correlation  $r_{\text{corr}}$  between Lin $I$  and Cir $I$  are very close to unity.



VO2187: MBD Bias (ps) vs Time (hours), between Lin & Cir Pol after PolConvert

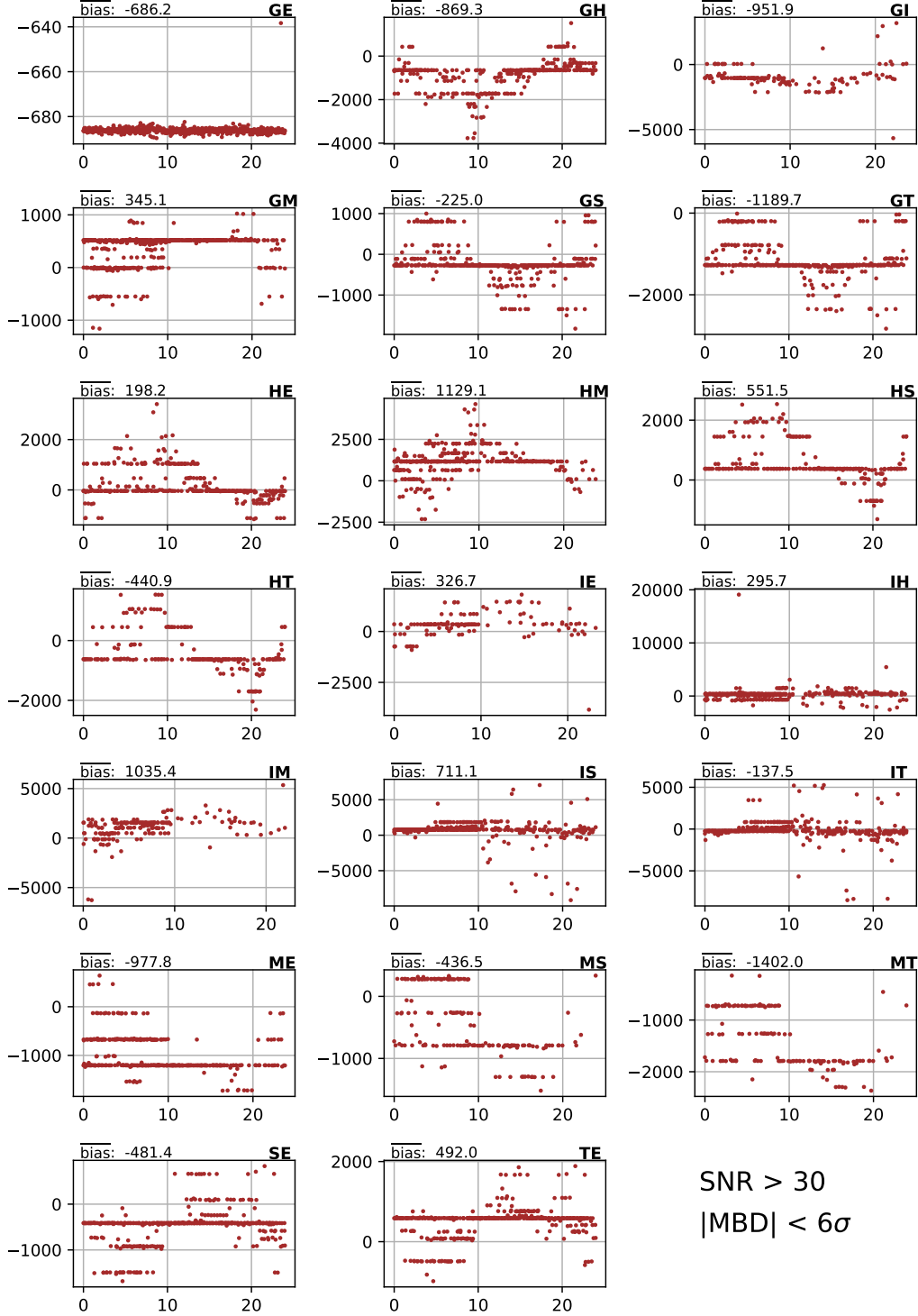


Figure 2: Evolution of the bias between LinI and CirI Multi-Band Delays (in picoseconds) during the experiment (time in minutes) for the pseudo-Stokes  $I$  parameter for every baseline. Each panel shows one curve of LinI - CirI, where LinI is  $I$  from the original, linearly polarized data set and CirI is  $I$  from the circularly polarized data set obtained by applying PolConvert to the original data. The means for each curve are given.

VO2187: MBD Residuals (ps) vs Time (hours), between Lin & Cir Pol after PolConvert

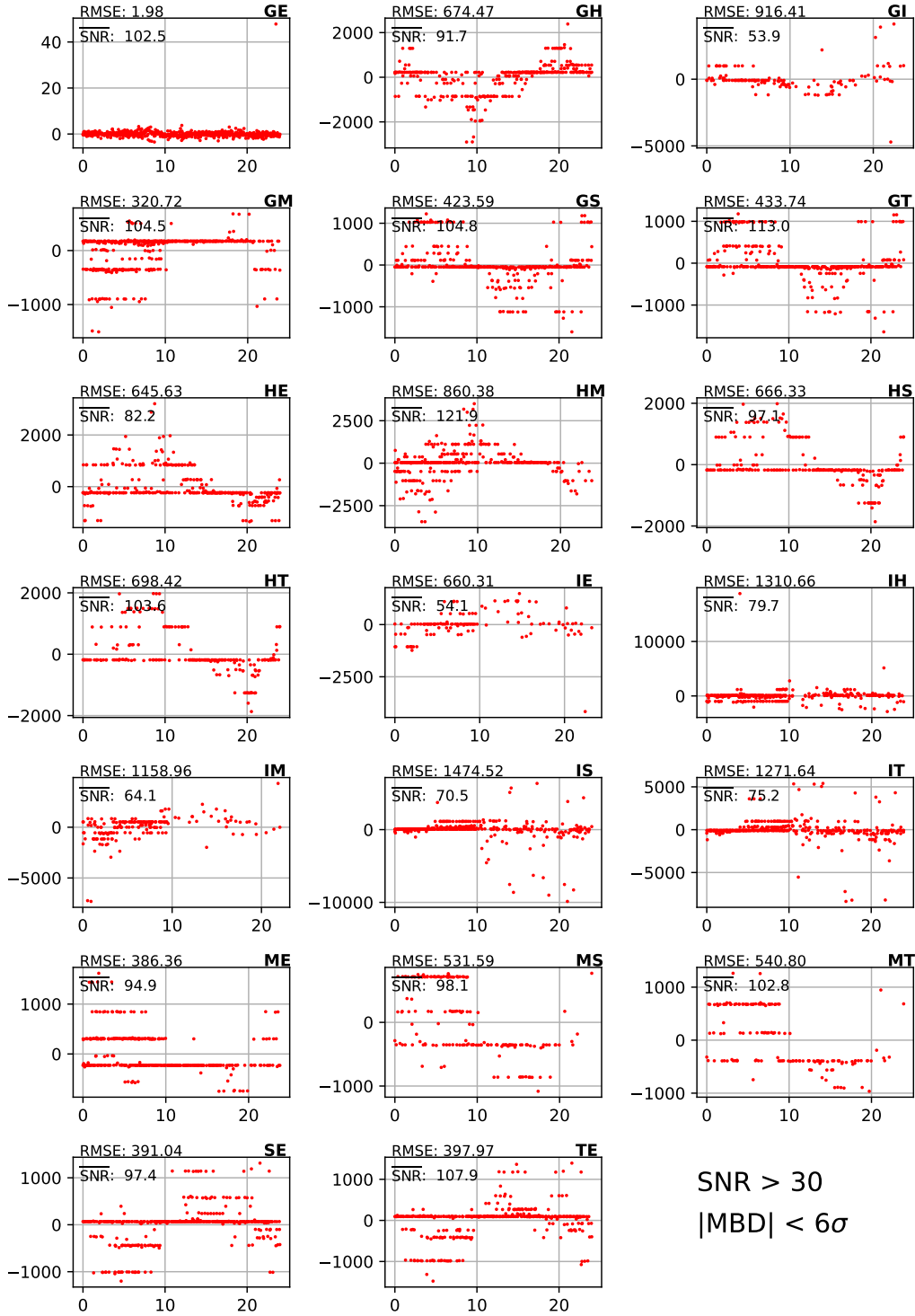


Figure 3: Evolution of the residuals between unbiased LinI and CirI Multi-Band Delays (in picoseconds) during the experiment (time in minutes) for the pseudo-Stokes  $I$  parameter for every baseline. Each panel shows a curve of LinI - CirI, both with subtracted means, where LinI is  $I$  from the original, linearly polarized data set and CirI is  $I$  from the circularly polarized data set obtained by applying PolConvert to the original data. The time-averaged SNRs are given. The Root Mean Square Error (RMSE) shows the standard deviation of the residuals.

VO2187: Pseudo-Stokes I SBD (ps) vs Time (hours), Lin (blue) & Cir (green) Pol after PolConvert

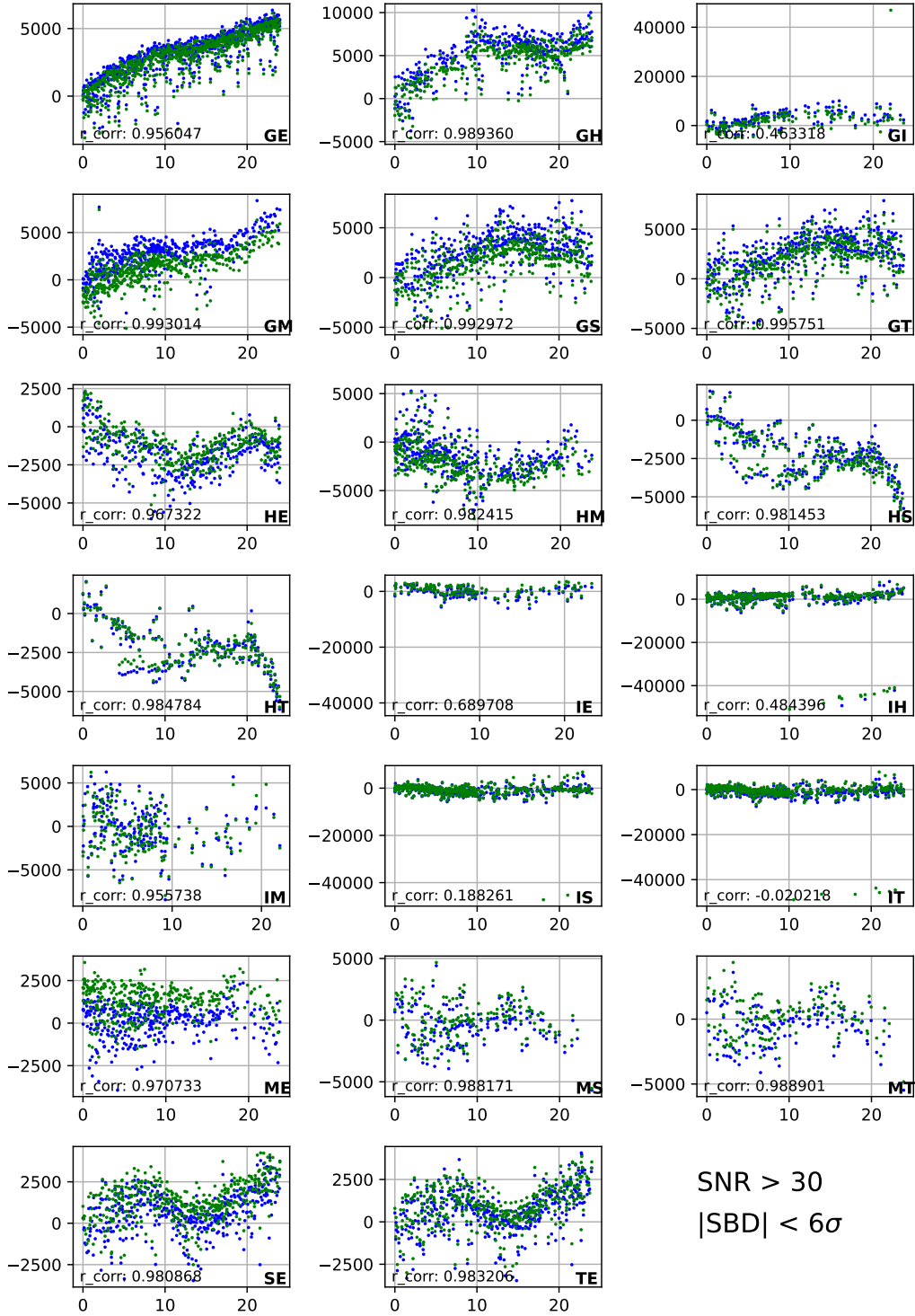


Figure 4: Evolution of Single-Band Delays (in picoseconds) during the experiment (time in minutes) for the pseudo-Stokes  $I$  parameter for every baseline. Each panel shows two curves: Lin $I$ ,  $I$  from the original, linearly polarized data set and Cir $I$ ,  $I$  from the circularly polarized data set obtained by applying PolConvert to the original data. Graphs have large biases. The coefficients of correlation  $r\_corr$  between Lin\_ $I$  and Cir $I$  for most of the baselines are close to unity. If it were not for the biases, the curves could be considered close to identical.

VO2187: SBD Bias (ps) vs Time (hours), between Lin & Cir Pol after PolConvert

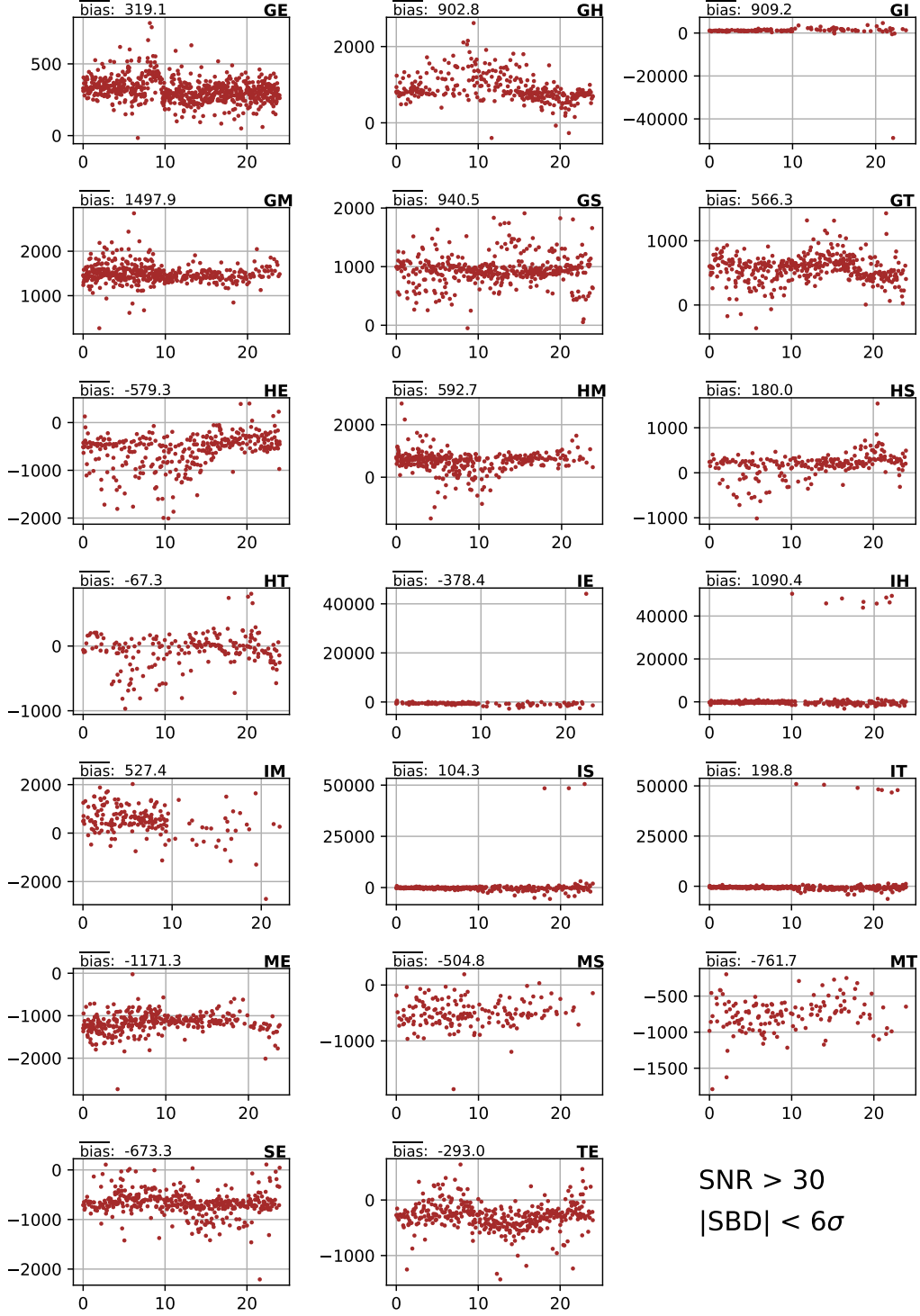


Figure 5: Evolution of the bias between LinI and CirI Single-Band Delays (in picoseconds) during the experiment (time in minutes) for the pseudo-Stokes  $I$  parameter for every baseline. Each panel shows a curve of LinI - CirI, where LinI is  $I$  from the original, linearly polarized data set and CirI is  $I$  from the circularly polarized data set obtained by applying PolConvert to the original data. The means for each curve are given.

VO2187: SBD Residuals (ps) vs Time (hours), between Lin & Cir Pol after PolConvert

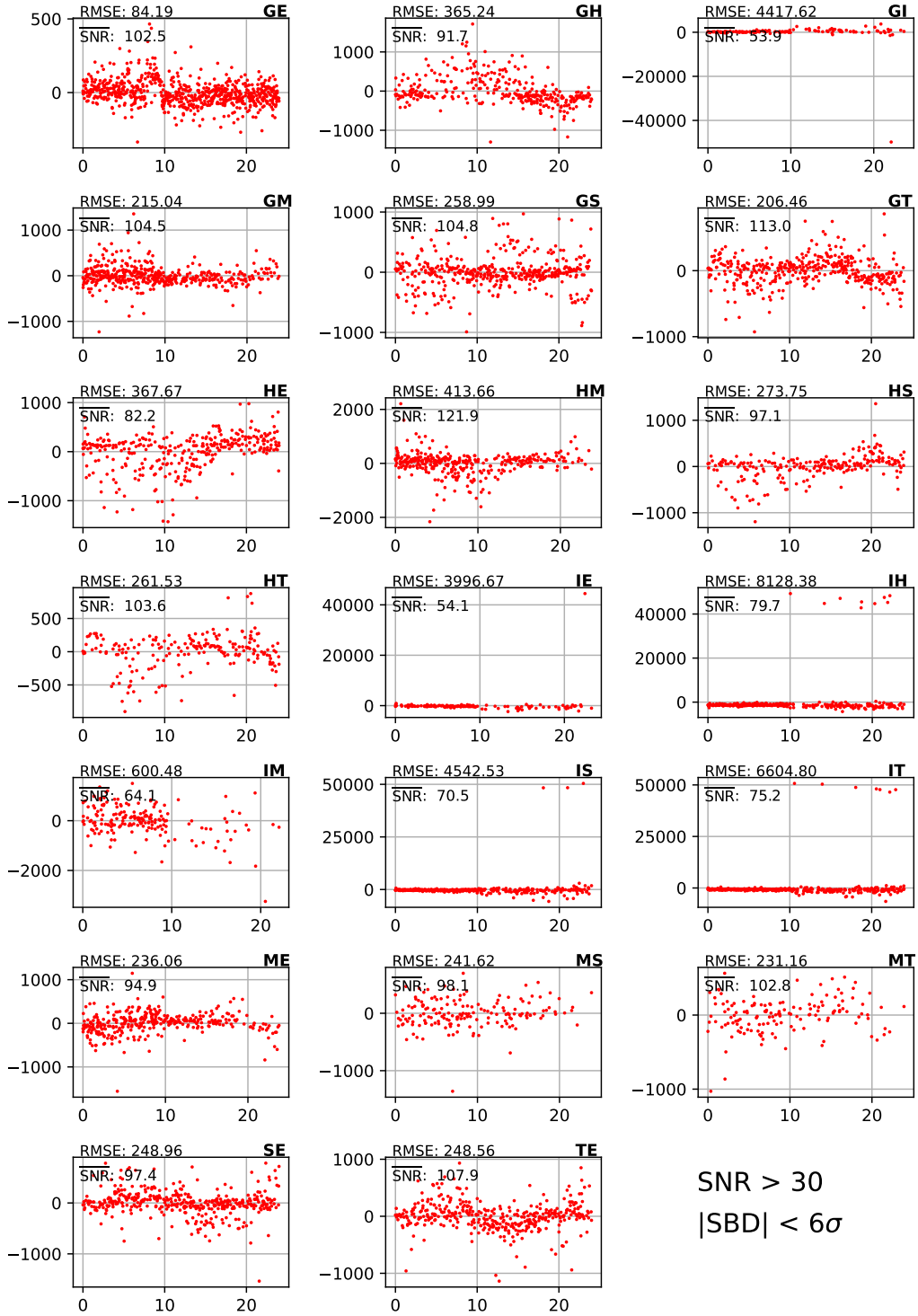


Figure 6: Evolution of the residuals between unbiased LinI and CirI Single-Band Delays (in picoseconds) during the experiment (time in minutes) for the pseudo-Stokes  $I$  parameter for every baseline. Each panel shows a curve of LinI - CirI, both with subtracted means, where LinI is  $I$  from the original, linearly polarized data set and CirI is  $I$  from the circularly polarized data set obtained by applying PolConvert to the original data. The Root Mean Square Error (RMSE) shows the standard deviation of the residuals.

VO2187: Pseudo-Stokes  $I$  SNR vs Time (hours), Lin (blue) & Cir (green) Pol after PolConvert

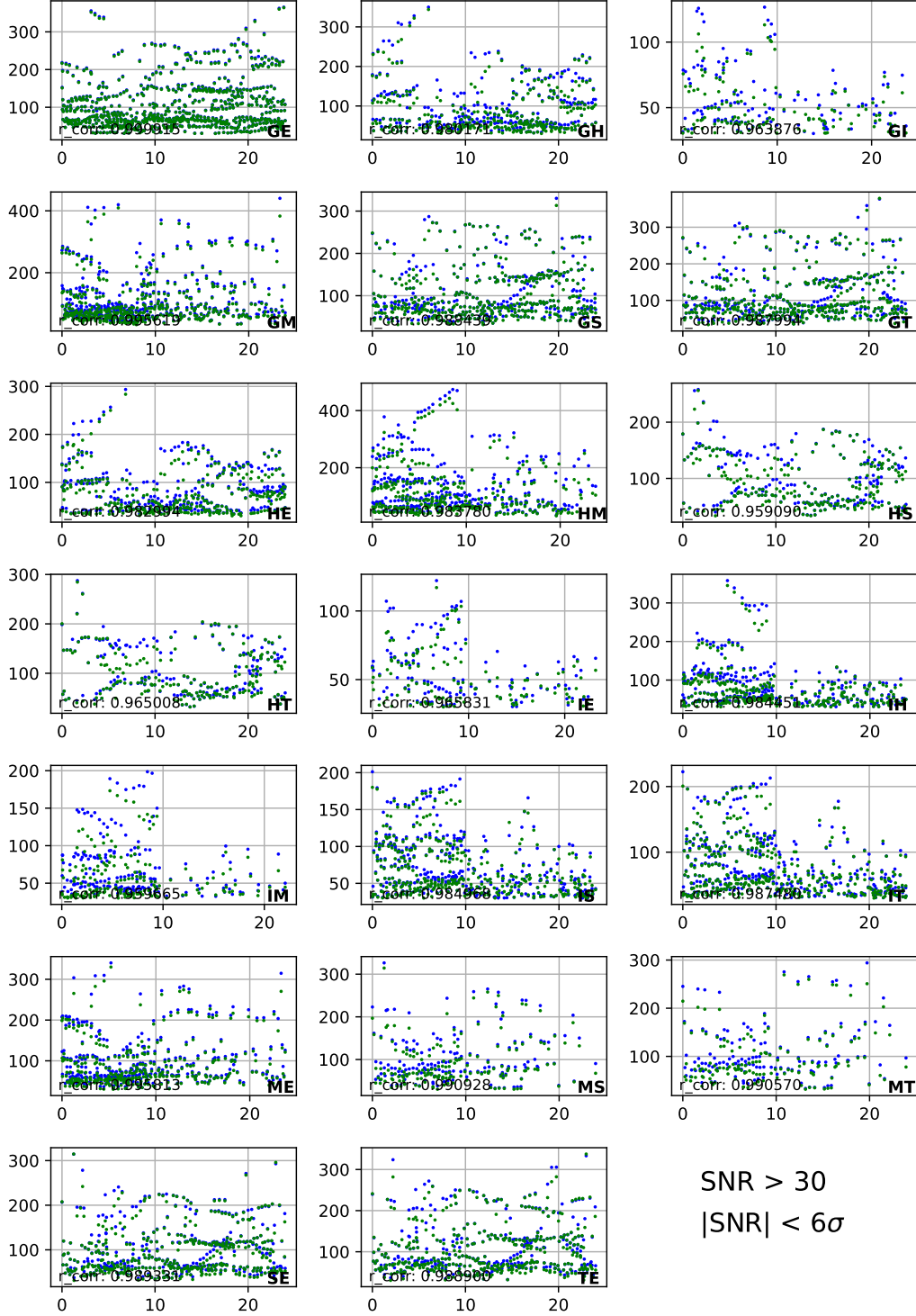


Figure 7: Evolution of Signal-to-Noise Ratios during the experiment (time in minutes) for the pseudo-Stokes  $I$  parameter for every baseline. Each panel shows two curves: LinI,  $I$  from the original, linearly polarized data set and CirI,  $I$  from the circularly polarized data set obtained by applying PolConvert to the original data. Graphs have large biases.

VO2187: SNR Bias vs Time (hours), between Lin & Cir Pol after PolConvert

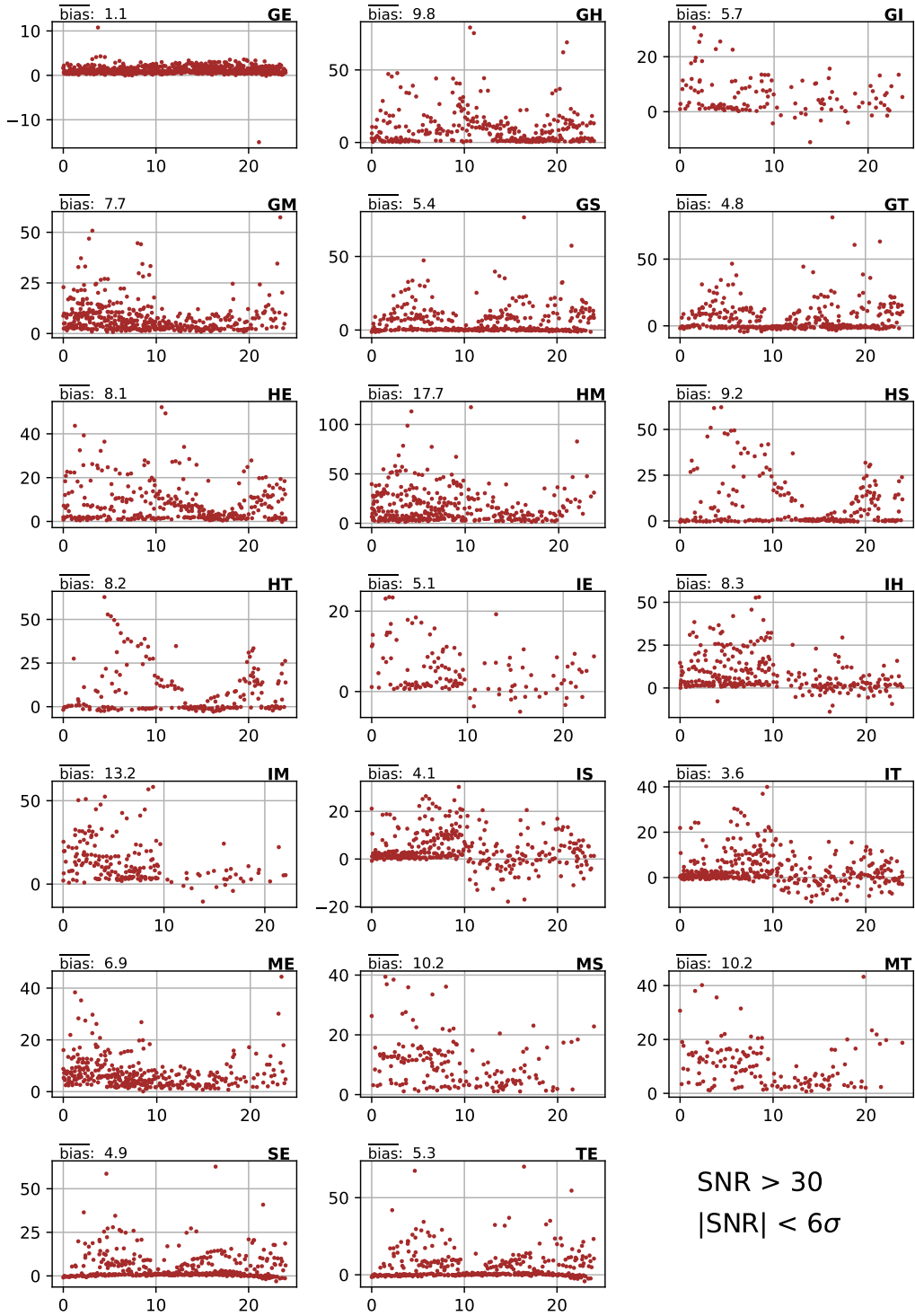


Figure 8: Evolution of the bias between LinI and CirI Signal-to-Noise Ratios during the experiment (time in minutes) for the pseudo-Stokes  $I$  parameter for every baseline. Each panel shows a curve of LinI - CirI, where LinI is  $I$  from the original, linearly polarized data set and CirI is  $I$  from the circularly polarized data set obtained by applying PolConvert to the original data. The means for each curve are given.

VO2187: SNR Residuals vs Time (hours), between Lin & Cir Pol after PolConvert

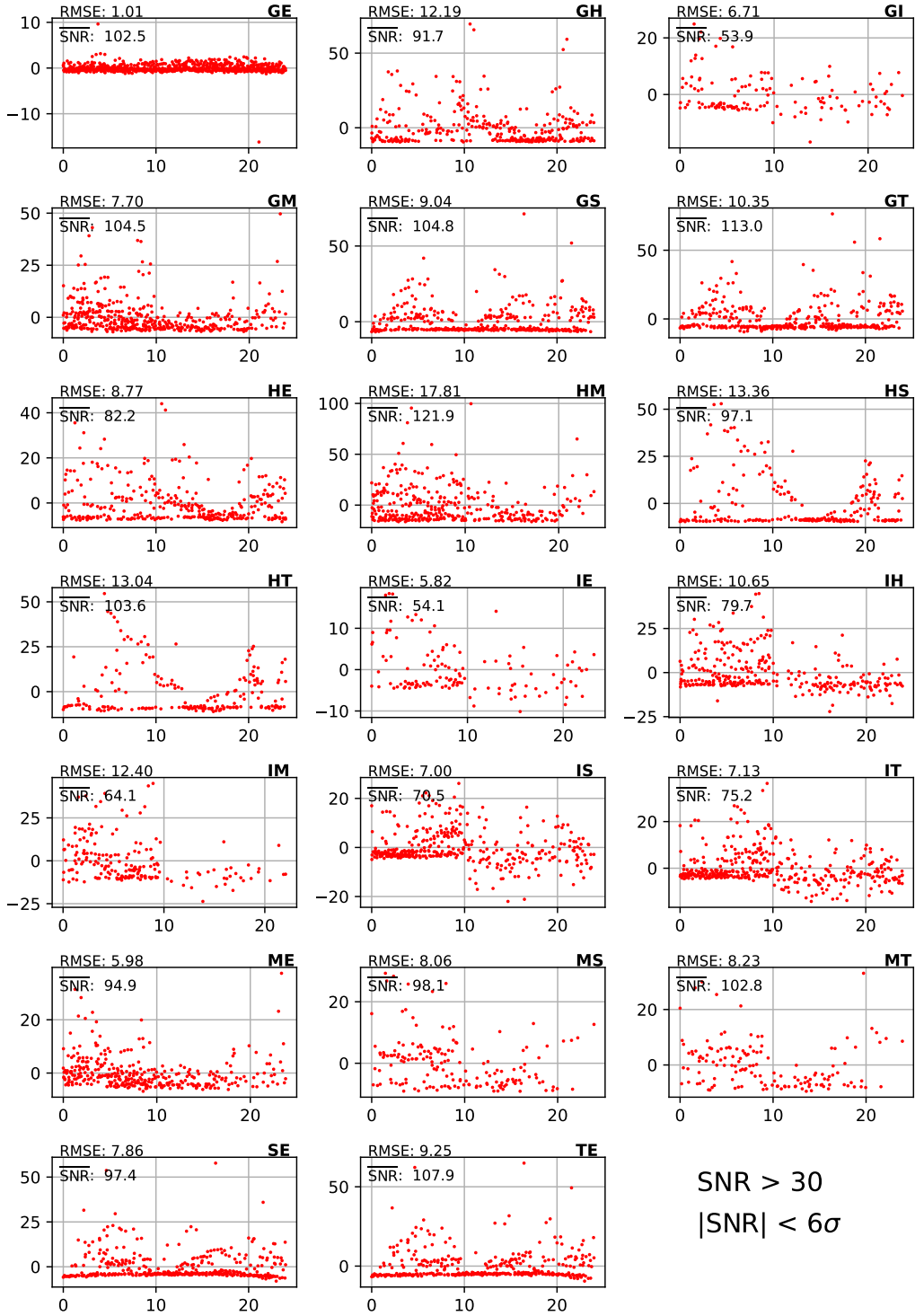


Figure 9: Evolution of the residuals between unbiased LinI and CirI Signal-to-Noise Ratios during the experiment (time in minutes) for the pseudo-Stokes  $I$  parameter for every baseline. Each panel shows a curve of LinI - CirI, both with subtracted means, where LinI is  $I$  from the original, linearly polarized data set and CirI is  $I$  from the circularly polarized data set obtained by applying PolConvert to the original data. The time-averaged SNRs are provided. The Root Mean Square Error (RMSE) shows the standard deviation of the residuals. The coefficients of correlation  $r\_corr$  between LinI and CirI are very close to unity.



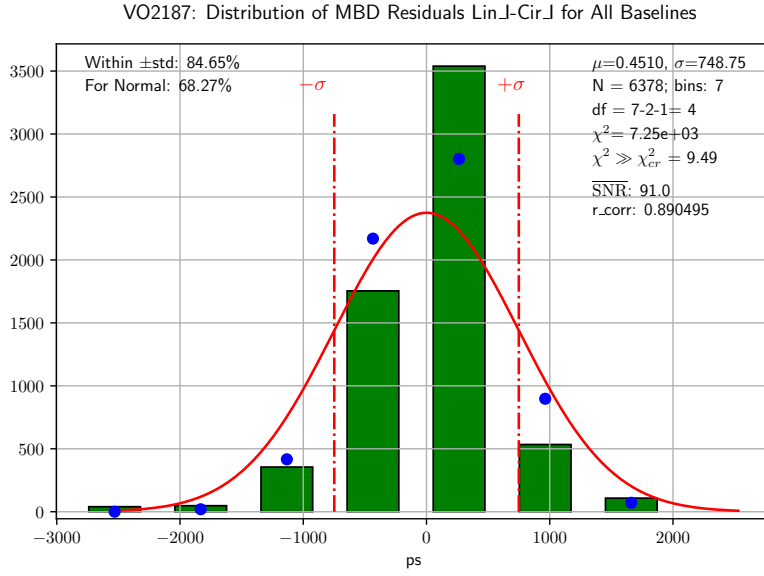


Figure 10: Histogram of the Multi-Band Delay residuals (see Eq. (3)) for all of the baselines.  $N$  is the total number of residuals spread over 21 bin. For the Pearson's  $\chi^2$  test, the left-tail and right-tail bins with sparse data (less than 5 counts in each) were grouped to make only  $N_b = 8$  bins. The number of degrees of freedom  $\text{df}$  is 5 (i.e.  $N_b - 3$ ). With this  $\text{df}$ , the critical value,  $\chi^2_{cr}$  is 11.07. The  $\chi^2$  observed, 183.48, greatly exceeds  $\chi^2_{cr}$ , so the test fails. However, the proportion of residuals within  $\pm\sigma$ , 83.8%, is much better than 68.27% of a normal distribution.

VO2187: Distributions of MBD Residuals Lin-I-Cir-I for Stations

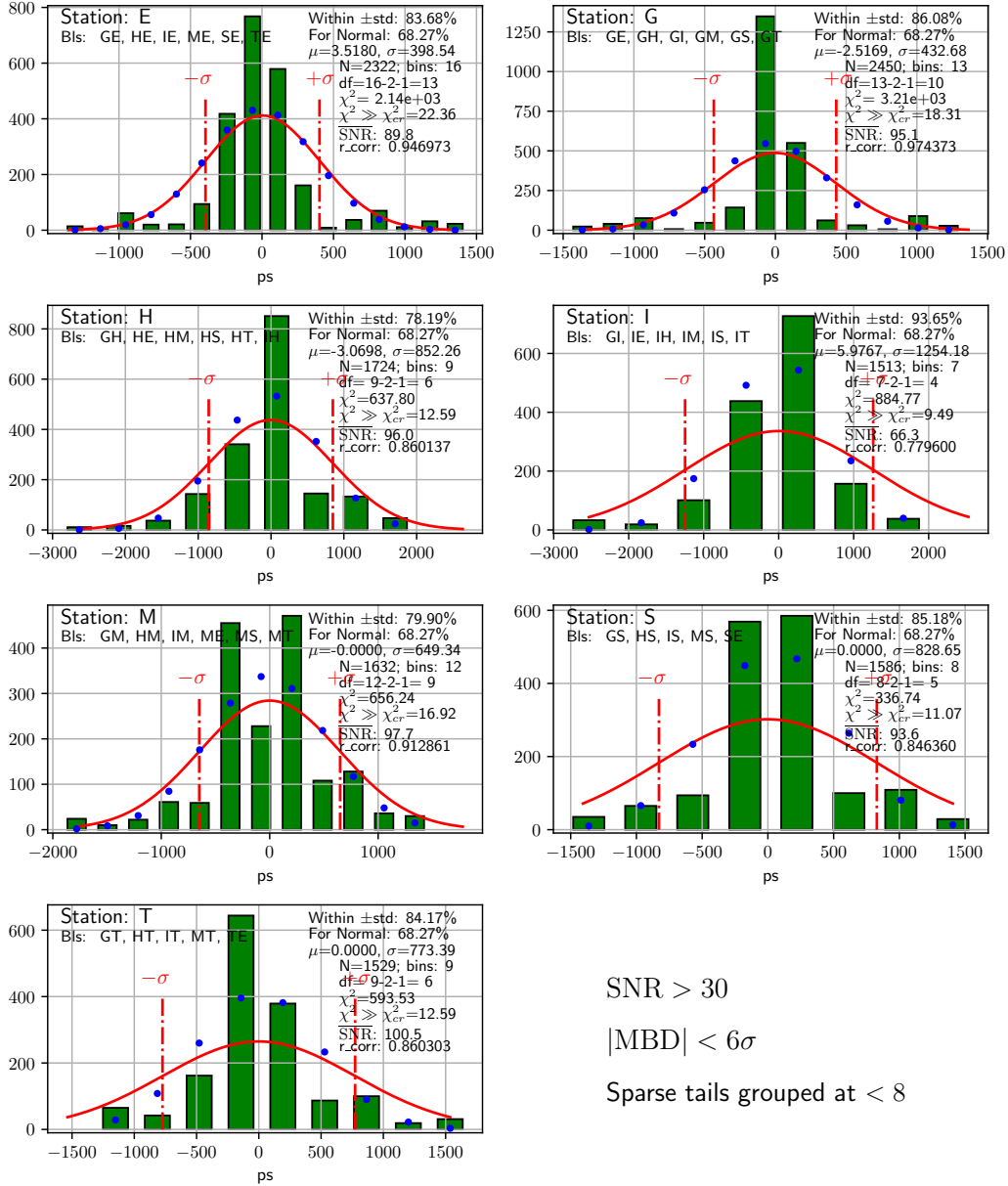


Figure 11: Histograms of the Multi-Band Delay residuals (see Eq. (3)) for the groups of baselines including an individual station. N is the total number of residuals spread over 21 bin. For the Pearson's  $\chi^2$  test, the left-tail and right-tail bins with sparse data (less than 5 counts in each) were grouped to make smaller number of bins  $N_b$ . The number of degrees of freedom df is  $N_b - 3$ . With these df, the critical values  $\chi^2_{cr}$  are calculated. For all the distributions, the  $\chi^2$  observed exceed  $\chi^2_{cr}$ , so the tests fail. However, the proportions of residuals within  $\pm\sigma$  are much better than 68.27% of a normal distribution.

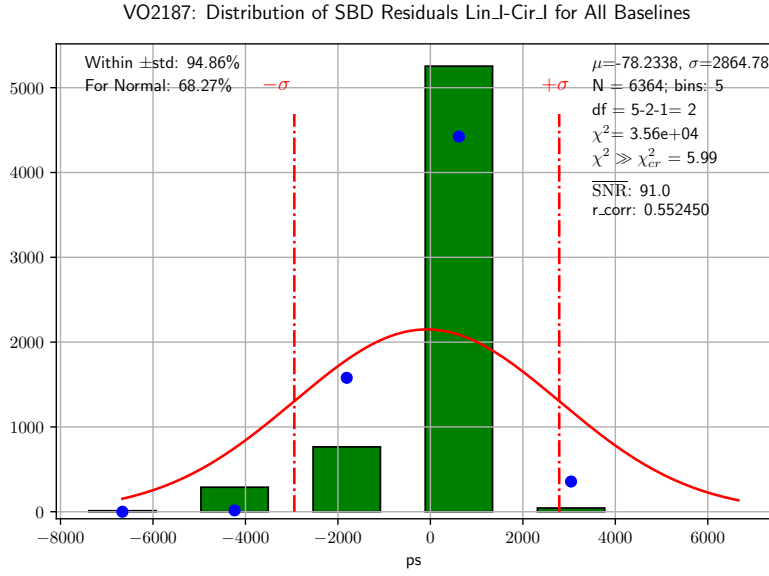


Figure 12: Histogram of the Single-Band Delay residuals (see Eq. (3)) for all of the baselines.  $N$  is the total number of residuals spread over 21 bin. For the Pearson's  $\chi^2$  test, the left-tail and right-tail bins with sparse data (less than 5 counts in each) were grouped to make only  $N_b = 9$  bins. The number of degrees of freedom  $df$  is 6 (i.e.  $N_b - 3$ ). With this  $df$ , the critical value,  $\chi^2_{cr}$  is 12.59. The  $\chi^2$  observed, 44.99, significantly exceeds  $\chi^2_{cr}$ , so the test fails. However, the proportion of residuals within  $\pm\sigma$ , 77.19%, is much better than 68.27% of a normal distribution.

VO2187: Distributions of SBD Residuals Lin.J-Cir.I for Stations

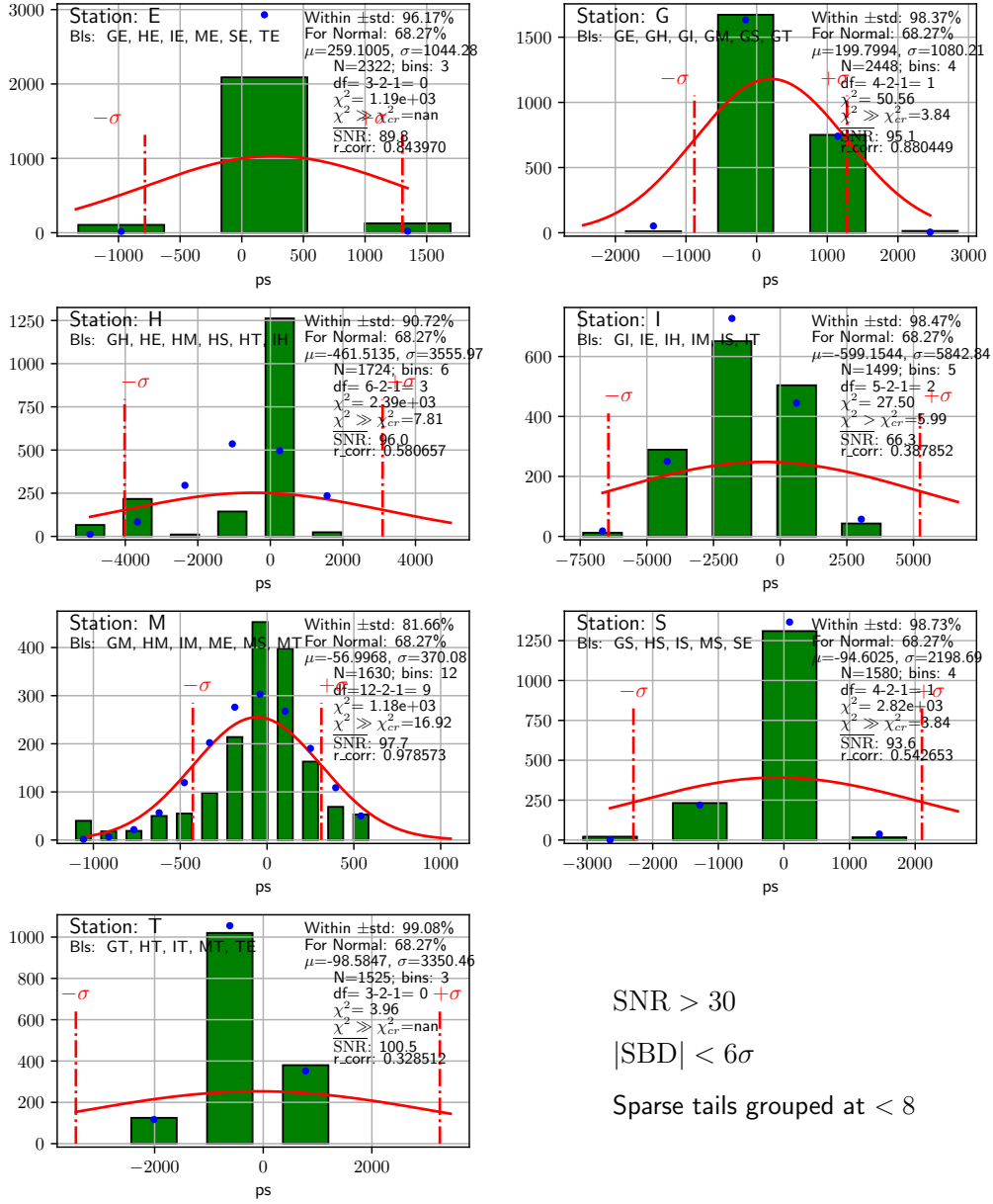


Figure 13: Histograms of the Single-Band Delay residuals (see Eq. (3)) for the groups of baselines including an individual station.  $N$  is the total number of residuals spread over 21 bin. For the Pearson's  $\chi^2$  test, the left-tail and right-tail bins with sparse data (less than 5 counts in each) were grouped to make smaller number of bins  $N_b$ . The number of degrees of freedom  $df$  is  $N_b - 3$ . With these  $df$ , the critical values  $\chi^2_{cr}$  are calculated. For most of the distributions (except those including stations E and V), the  $\chi^2$  observed exceed  $\chi^2_{cr}$ , so the tests fail. Stations E and V have normally distributed residuals with 95% confidence. However, for all of the stations, the proportions of residuals within  $\pm\sigma$  are better than 68.27% of a normal distribution.

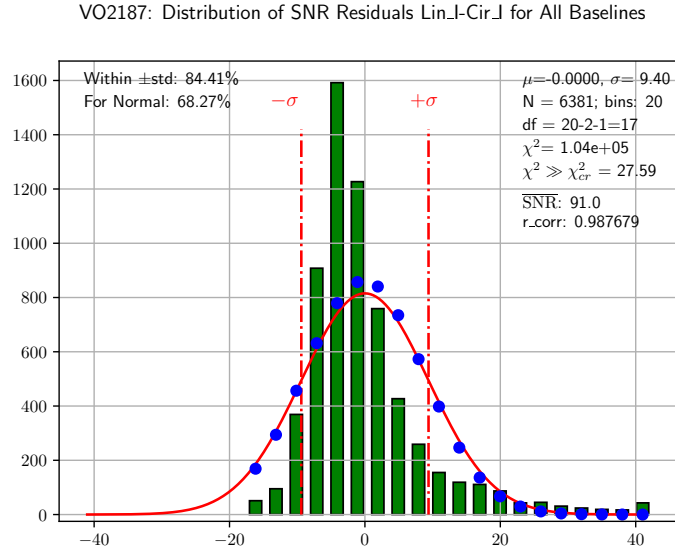


Figure 14: Histogram of the Signal-to-Noise Ratio residuals (see Eq. (3)) for all of the baselines.  $N$  is the total number of residuals spread over 21 bin. For the Pearson's  $\chi^2$  test, the left-tail and right-tail bins with sparse data (less than 5 counts in each) were grouped to make only  $N_b = 11$  bins. The number of degrees of freedom  $df$  is 8 (i.e.  $N_b - 3$ ). With this  $df$ , the critical value,  $\chi^2_{cr}$  is 15.51. The  $\chi^2$  observed, 130.72, greatly exceeds  $\chi^2_{cr}$ , so the test fails. However, the proportion of residuals within  $\pm\sigma$ , 82.94%, is much better than 68.27% of a normal distribution.

VO2187: Distributions of SNR Residuals Lin-I-Cir-I for Stations

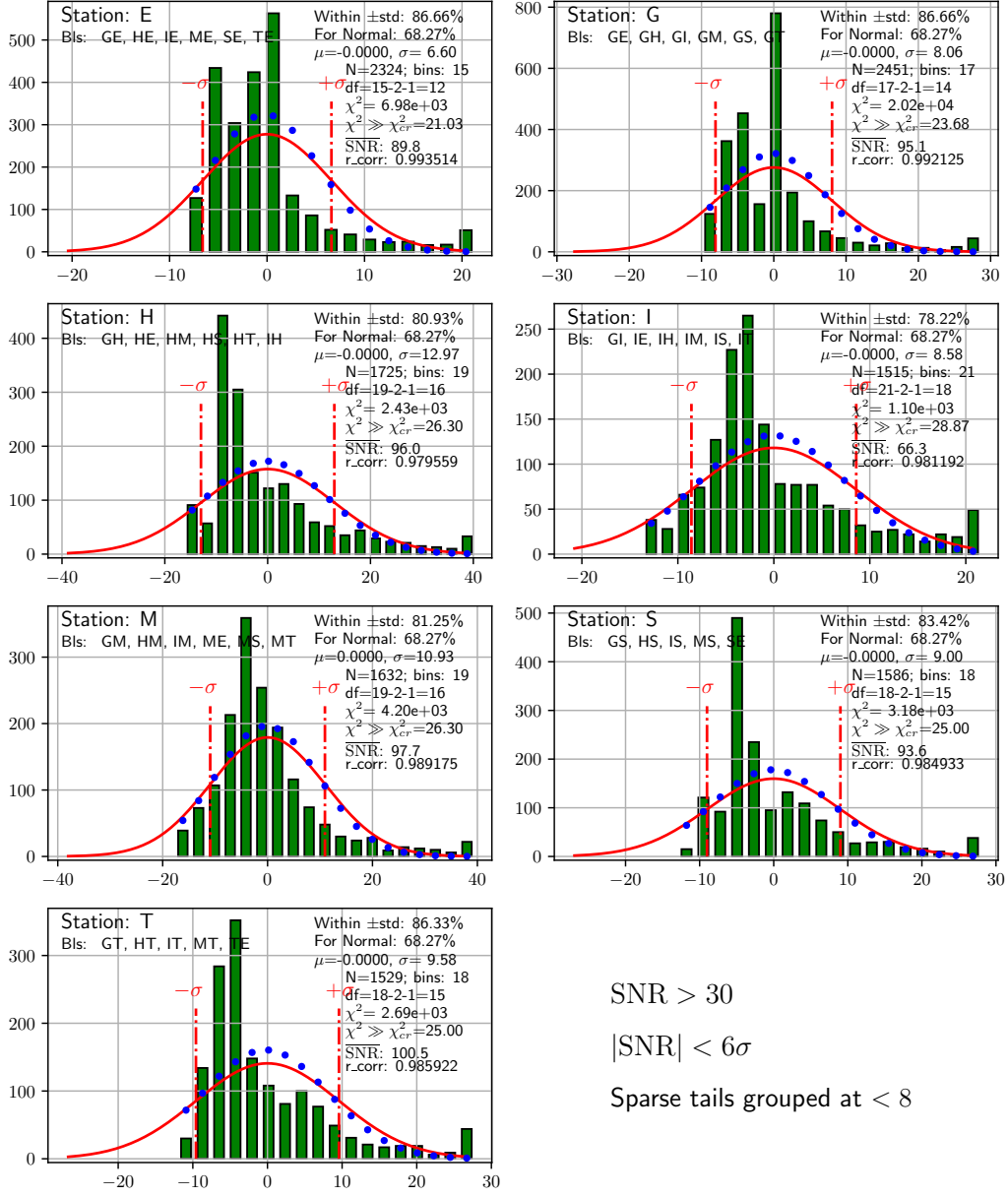


Figure 15: Histograms of the Signal-to-Noise Ratio residuals (see Eq. (3)) for the groups of baselines including an individual station. N is the total number of residuals spread over 21 bin. For the Pearson's  $\chi^2$  test, the left-tail and right-tail bins with sparse data (less than 5 counts in each) were grouped to make smaller number of bins  $N_b$ . The number of degrees of freedom df is  $N_b - 3$ . With these df, the critical values  $\chi^2_{cr}$  are calculated. For all the distributions, the  $\chi^2$  observed exceed  $\chi^2_{cr}$ , so the tests fail. However, the proportions of residuals within  $\pm\sigma$  (except station M with its 64.81%) are much better than 68.27% of a normal distribution.