The ngVLA Prototype Pointing Model: Correspondence Between TPOINT and PEEK

ngVLA Antenna Memo 18

Pedro P.B. Beaklini

September, 30th 2025

Abstract

The present memo briefly describes the correspondence between the pointing model terms used in TPOINT and PEEK. The goal is to match the software for pointing analysis that will be used for the ngVLA prototype antenna with the current procedure followed by EVLA operations team.

1. Introduction

A telescope pointing error is well described by a model that predicts antenna offsets as a function of elevation and azimuth and depends on several terms. The offset is defined as the difference between the real position and the commanded position of a given telescope, and each term has a physical meaning, such as tilt, encoder offsets, collimation, gravitational bending, and refraction(e.g. Meeks et al. 1968, Baars 1973, Wade 1990, Greve et al. 1996, Baars 2007). The elevation and azimuth models must be obtained together since, besides the direct interdependence of the tilt terms, there are overall deformations on the structures that depend on both azimuth as in elevation changes. Although different observatories can have different strategies for obtaining the model, describing azimuth and elevation errors as a linear combination of both coordinates is a common aspect of the observatories, from radio to optical (Penalver et al. 2000, Wallace 2002, Patel et al. 2004, Mangum et al, 2006). Therefore, there is extensive literature and several tools for a good analysis (see also Dumke & MacAuliffe 2010, Kong et al. 2014 Walker 2015). Still, each telescope has its particularities, and dealing with all the terms and sign conventions can be tricky.

The ngVLA pointing recommendation is well described by ngVLA Antenna Memo 17 (Mangum 2024), and the software tool presented to obtain the pointing model is TPOINT (Wallace 2002), which is the same as the one used by ALMA (Mangum 2001). The idea is to use TPOINT to get the model for the prototype antenna during the science verification period, which is expected to start in April 2025. TPOINT is particularly easy to handle and it does a good job with residual plots and identifying outliers and, eventually, new terms. The out-of-axis optics could add additional dependencies and extra terms could be included. It is likely that any unexpected dependence would be proportional to azimuth and/or elevation, but the understanding of the terms can be different.

Historically, VLA operation uses its own code to fit an appropriate pointing model for the antennas. It is called PEEK, written in Fortran by multiple hands. Version 1.0 is from August 1987, and while it has been on version 9.0 since September 2005, it went through several minor changes led by Ken Sowinski. The last one in 2024 to add na29 - the ngVLA prototype antenna - to its code. In the last B to A configuration change (October 2024), I computed the models using the regular and the new version of PEEK which includes na29, and both revealed the same results, as expected.

After almost 4 decades of use, VLA operations softwares are already in synchrony among them. The current system has worked since the upgrade to EVLA in 2010, although the implementation started even earlier in 2007. The TELCAL software provides the pointing solutions of a given observation in a text file that will feed as input for PEEK to get the best-fitted model, providing as output a solution that will be uploaded to the parminator and finally, the executor will get the model of each antenna on the parminator. Therefore, signs and tilt conventions are set to be understood by the entire system, and none of the parts can be understood alone. The model is done differentially, i.e., it fits the changes in each term instead of computing the absolute value.

The present memo aims to integrate the TPOINT input/output files with the VLA operation system used daily by the VLA support telescope team. The idea is to produce a document that easily translates the terms PEEK used to TPOINT description. And, of course, confirm that they both got the same model for the antennas.

2. PEEK INPUT / TELCAL OUTPUT

Figure 1 shows an example of a TELCAL output file of a pointing run, equivalent to the PEEK input file. The series of zeros on the header indicates that we will compute the differential model, i.e., we will start as if all the terms were set to zero. After that, we have the header of each scan, which contains information about wind, band, and target.

Table 1: Understanding the columns on the PNT file (Figure 1)

Ant	Pol	cosEL	sin EL	cosAz	sin AZ	ΔEL	beam	ΔAZ	beam	amp	OTT	old
01	R	0.9963	0.0862	0.9457	-0.325	-0.415	1.053	-0.106	1.092	0.043	F	F
01	L	0.9963	0.0862	0.9457	-0.325	0.113	1.049	-0.249	1.090	0.059	F	F

The pointing information starts after it. Each row corresponds to an antenna, with the right-hand polarization printed first. Therefore, if all 28 antennas were used, a pointing scan would provide 56 lines.The meaning of each column is shown in Table 1 for easy reading, where beam means beamwidth and amp means amplitude. The last numerical column corresponds to the amplitude, while the two final booleans correspond to that order: if the observation was over-the-top and a former flagging label no longer used by the VLA.

```
//VER  TELCAL
//RFN                   2024_119_05:32
24_119_05:32 //PAD     01   W12/
24_119_05:32 //PAD     02   N24/
24_119_05:32 //PAD     03   E28/
24_119_05:32 //PAD     04   E04/
24_119_05:32 //PAD     05   W04/
24_119_05:32 //PAD     06   E20/
24_119_05:32 //PAD     07   N20/
24_119_05:32 //PAD     08   N32/
24_119_05:32 //PAD     09   W20/
24_119_05:32 //PAD     10   N12/
24_119_05:32 //PAD     11   W08/
24_119_05:32 //PAD     12   E08/
24_119_05:32 //PAD     13   W32/
24_119_05:32 //PAD     14   N04/
24_119_05:32 //PAD     15   W16/
24_119_05:32 //PAD     16   E24/
24_119_05:32 //PAD     17   N16/
24_119_05:32 //PAD     18   N36/
24_119_05:32 //PAD     19   E12/
24_119_05:32 //PAD     20   W24/
24_119_05:32 //PAD     21   E36/
24_119_05:32 //PAD     22   N28/
24_119_05:32 //PAD     23   N08/
24_119_05:32 //PAD     24   W36/
24_119_05:32 //PAD     25   E32/
24_119_05:32 //PAD     27   E16/
24_119_05:32 //PAD     28   W28/
24_119_05:32 //ANT  01   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  02   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  03   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  04   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  05   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  06   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  07   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  08   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  09   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  10   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  11   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  12   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  13   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  14   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  15   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  16   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  17   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  18   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  19   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  20   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  21   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  22   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  23   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  24   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  25   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  27   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
24_119_05:32 //ANT  28   0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
//OFS1  24 119 05:32   1310+323  C  SUBBAND: 0  REF_ANT: ea12
//OFS2                      0.533786   12h48m39.095
//WX1  24 119 05:32         3.60      4.83   276.55  78470.00   275.25
//WX2  24 119 05:32         4.30
//SUB  1 OBSERVE FILE:
//SUB  2 DATA FILE:
//SUB  3 PROGRAM NAME: pnt_bl_check.60428.228998032406
//SUB  4 SUBN:  27
//SUB 11 GDIDS1: 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 27 28
//SUB 14 BDIDS1:
01 R   0.9963   0.0862   0.9457  -0.3250 -0.415  1.053 -0.106  1.092  0.043 F F
01 L   0.9963   0.0862   0.9457  -0.3250  0.113  1.049 -0.249  1.090  0.059 F F
02 R   0.9963   0.0862   0.9457  -0.3250  0.123  1.051 -0.039  1.142  0.056 F F
02 L   0.9963   0.0862   0.9457  -0.3250  0.604  1.047 -0.210  1.136  0.043 F F
03 R   0.9963   0.0862   0.9457  -0.3250 -0.322  1.057 -0.071  1.046  0.053 F F
```

Figure 1: Example of PNT file (TELCAL output and PEEK input)

PEEK works with the average between left and right polarization, with the measured value of the split between the two polarisations hard coded. Since the table is presented in sine and cosine (and not directly in elevation and azimuth), PEEK uses this boolean column to identify if it was over-the-top. The user can select a threshold for wind (typically 8 m/s at C-band and 5 m/s at X-band), bandwidth (ideal value is one, not used when the deviation is higher than 20%), and amplitude. PEEK also allows the user to fit the terms. Hardcoded in the PEEK code is the relation between the antenna's tilt and the pad's tilt and the sign angle convention used by the executor. A PEEK run produces three output files:

- PTR file: solutions of each antenna with the rms
- Changes file: which changes are significant to be updated
- gp file to make plots

Strictly speaking about VLA operations, we do not update all the parameters pointed out by PEEK without an analysis made by the team. Wind direction and weather conditions can play a role. We typically focus on recently moved antennas, where we expected a new pointing model. We are conservative in updating the model in antennas that did not move, but that may happen.

3. TPOINT input

Mangum described the TPOINT input file on ngVLA Antenna Memo 17 (Appendix A). To ensure we will be able to use TPOINT on the commissioning of the ngVLA prototype, I worked in a simple Python code to convert the TELCAL output to the TPOINT input format, with the commanded and real elevation and azimuth. A few important considerations on that file conversion:

- I obtained the commanded elevation and azimuth using the cosine and sine information of elevation and azimuth presented on the TELCAL output.
- I obtained the real elevation and azimuth after adding the measured errors to the commanded position.
- I've averaged right and left polarization errors to be appropriate with TPOINT input.
- The executor operates at a sky angle, and the TELCAL az errors must be divided by the cosine of elevation before being added to the azimuth real position.
- I converted the solutions to arcseconds (TELCAL provides solutions in arcmin and TPOINT assumes solutions in arcsecond).
- If necessary, I need to evaluate all exclusion criterions used by PEEK before running the conversion code: wind, amplitude, and beamwidth. For the tests shown here, I decided not to exclude any scans based on those quantities.

The final file corresponding to the one shown in Figure 1 is presented in Figure 2, and it is ready to be read by TPOINT. Since the prototype won't perform over-the-top observation, I've excluded them from the TPOINT modelling, although they have been used by PEEK.

4. Goals

After setting the input file for TPOINT, the major goal is to verify if TPOINT and peek point to the same solution. I followed two different strategies: (i) I reanalyzed pointing data from the last reconfiguration with TPOINT, and I compared it with the values found by PEEK. By reconfiguration cycle, I mean (C to B move on April 24, B to A move on October 2024, A to D move on February 25, D to C on May 25 move and again C to B move on August 25). That is, in fact, a verification of the differential model strategy. (ii) We have set a model of one particular antenna to zero (ea04) and have fitted both PEEK and TPOINT to check the solutions.

```
VLA Antenna Test Facility: Test to ngVLA prototype
: ALTAZ
: ALLSKY
34 4 24.636 !
! az, el, azEnc, elEnc, offx, offy, centx, centy, s2n
    109.10017    85.08219   108.96879    85.05748 !     1310+323   0
    334.53102    73.59902   334.67479    73.59867 !     1310+323   2
    356.12828    42.62547   356.19862    42.60947 !     1310+323   4
    352.25273    12.56528   352.30696    12.55215 !     1310+323   6
    329.10096    22.79165   329.15884    22.79522 !     1310+323   8
    281.90425    14.21128   281.95601    14.23433 !     1310+323   10
    302.33763    32.16130   302.39648    32.17870 !     1310+323   12
    293.94865    69.78639   294.05849    69.80805 !     1310+323   14
    257.45363    83.12828   257.53451    83.15652 !     1310+323   16
    224.34078    46.06192   224.38557    46.08560 !     1310+323   18
    228.17165    18.49910   228.22166    18.52286 !     1310+323   20
    204.96189    11.90994   205.01004    11.93060 !     1310+323   22
    168.25411    16.92117   168.29852    16.92287 !     1310+323   24
    181.36371    40.44796   181.39822    40.45759 !     1310+323   26
    178.83851    84.36658   178.60178    84.37848 !     1310+323   28
    208.89447    83.62279   208.78444    83.64103 !     1310+323   30
    120.46372    44.41740   120.50100    44.39088 !     1310+323   32
    116.20115    17.26168   116.24776    17.23503 !     1310+323   34
     80.62861    25.25110    80.67938    25.21784 !     1310+323   36
     56.67770    16.71696    56.73065    16.68503 !     1310+323   38
     68.19323    50.10857    68.24939    50.07530 !     1310+323   40
    126.49032    47.39187   126.53062    47.36838 !     1310+323   42
    122.15672     9.30383   122.20521     9.28456 !     1310+323   44
     83.92078    31.11165    83.97531    31.07507 !     1310+323   46
     67.02043    17.11487    67.07237    17.07964 !     1310+323   48
     68.92782    55.45437    68.99393    55.41860 !     1310+323   50
     32.01441    78.17406    32.15718    78.14939 !     1310+323   52
    355.03666    79.19600   355.20475    79.18375 !     1219+484   54
    358.52895    41.97801   358.59647    41.96225 !     1219+484   56
    357.96192    18.15200   358.01506    18.14028 !     1219+484   58
    320.12067    15.73676   320.17345    15.74101 !     1219+484   60
    283.00094    21.10510   283.05354    21.12505 !     1219+484   62
    297.12167    44.54788   297.18540    44.56263 !     1219+484   64
    334.87514    77.89879   335.03891    77.89753 !     1219+484   66
    207.25841    71.30317   207.24764    71.32103 !     1219+484   68
    234.12485    38.15660   234.17210    38.17860 !     1219+484   70
    251.07069    22.67016   251.12207    22.69148 !     1219+484   72
    207.54805    15.33840   207.59595    15.35625 !     1219+484   74
    172.18176    17.87409   172.22713    17.88083 !     1219+484   76
      5.68983    76.73546     5.84575    76.72285 !     1219+484   78
    355.33815    49.91515   355.41498    49.90324 !     1219+484   80
    351.09847    14.61416   351.15179    14.60251 !     1219+484   82
    329.12487    24.53888   329.18120    24.53680 !     1219+484   84
```

Figure 2: Converted file presented in figure 1 to be read by TPOINT for ea04.

5. Nomenclature

Below is a general notation for the pointing equation, where equation (1) corresponds to the azimuth error and equation (2) corresponds to the elevation error.

$$\Delta Az\cos(El) = C_1 + C_2\cos(El) + C_3\sin(El) + C_4\cos(Az)\sin(El) + C_5\sin(Az)\sin(El)$$
$$+C_6\cos(Az)\cos(El) + C_7\sin(Az)\cos(El) + C_8\cos(2Az)\cos(El) + C_9\sin(2Az)\cos(El)$$
$$+C_{10}\cos(3Az)\cos(El) + C_{11}\sin(3Az)\cos(El)$$

(1)

$$\Delta El = D_1 + D_2\cos(El) + D_3\sin(El) + D_4\cos(Az) + D_5\sin(Az)$$
$$+D_6\cos(2Az) + D_7\sin(2Az) + D_8\cos(3Az) + D_9\sin(3Az)$$

(2)

Although the physical meaning is given by the sine and cosine dependence of each term, there are a variety of names and labels that depend on each observatory. At VLA high order terms, i.e., 2AZ and 3AZ are not used, and the tilt terms C4/D5 and C5/D4 are connected. A more simple and useful version of the equation is presented below, which also considers the tilt terms interdependence (Baars 2007):

$$\Delta Az\cos(El) = P_1 + P_2\cos(El) + P_3\sin(El) + P_4\sin(El)\cos(Az) + P_5\sin(El)\sin(Az)$$

(3)

$$\Delta El = P_6 + P_7\cos(El) - P_4\sin(Az) + P_5\cos(Az) + P_8(\cos(El)/\sin(El))$$

(4)

There is also a diversity of sign notation; while PEEK uses the classical notation presented above (named Stumpff notation, Stumpff 1972), TPOINT uses a different one, flipping the sign of the azimuth terms (identified by Baars as Wallace notation - see also ngVLA Antenna Memo 17 and Wallace 2002). In Table 1, I present a dictionary of the various terms and how the parminator identifies them.

The refraction term (P8) is typically tiny, It can be found independently and it has no affect on the other terms of the pointing model. Therefore, we do not change it during VLA regular operation, and I won't consider it in the TPOINT and PEEK correspondence analysis (for completeness, the refraction term is identified as REFR in PEEK solutions). The terms A3, A4, and E4 in PEEK represent the encoder center errors in Azimuth (EW and NS direction - C6 and C7 in equation 1) and Elevation (horizontal plane - D3 in equation 2). Typically, for the VLA antennas, they are small and stable terms, and we can find a suitable model that only fits the P terms of equations 3 and 4. At EVLA, those items were updated by Ken mostly in 2015 - but for a few antennas, it is from the pre-EVLA era.

Finally, it is essential to have measurements well spread around the space of parameters (azimuth and elevation) to obtain a reliable pointing model.

Table 1: Sign and terms conversion between Parminator, Peek and TPOINT

Term	Parminator	Conversion Parminator to Peek	Peek Parameter	Conversion Peek to TPOINT	TPOINT Parameter
P1	AZCOL	1	A6	-1	CA
P2	AZENC	1	A7	-1	IA
P3	PERPENDICULARITY	1	A5	-1	NPAE
P4	NSTILT	1	A2/E2	-1	AN
P5	EWTILT	-1	A1/E1	-1	AW
P6	ELCOL	1	E5	1	IE
P7	ELCENTERINGCOS	1	E3	1	ECEC

6. Results: Correspondence between TPOINT and PEEK

I will present the results of both strategies (6.1 and 6.2). But first, I must explain how the VLA operation team handles the tilt terms. The tilt on VLA is a combination of two terms: the antenna tilt and the pad tilt. We can not separate between them when we model the pointing errors. Therefore, we use the master pad to separate those terms. When we measure the pointing errors in an antenna in the master pad, where we know the pad tilt, we assume that the errors are due to the antenna tilt. When an antenna is not on the master pad, we attribute the errors to the pad tilt. That won't be an obstacle to our tests, but we must consider how the executor combines antenna and pad tilt on section 6,2, where we aim to obtain the absolute pointing model. Finally, for convenience, I opt to work in arcsec on section 6.1 and in arcmin on section 6.2 (for a direct check with the parminator).

6.1 Differential Pointing model on the current base

TPOINT was used on former VLA pointing data to validate it to the ngVLA prototype. We did not get new data, we have used data from the previous VLA reconfiguration (see Section4). Since we are working with a differential pointing model during EVLA operations, we do not fit ECEC and CA because we do not expect those quantities to change. Therefore, for this first analysis, we have modeled only five terms of table 1: NSTILT (North-South Tilt), EWTILT (East-West Tilt), AZENCZERO (Azimuth zero of the encoders), PERPENDICULARITY (a non-perpendicularity of the azimuth axis) and ELCollimation (Collimation in the elevation direction).

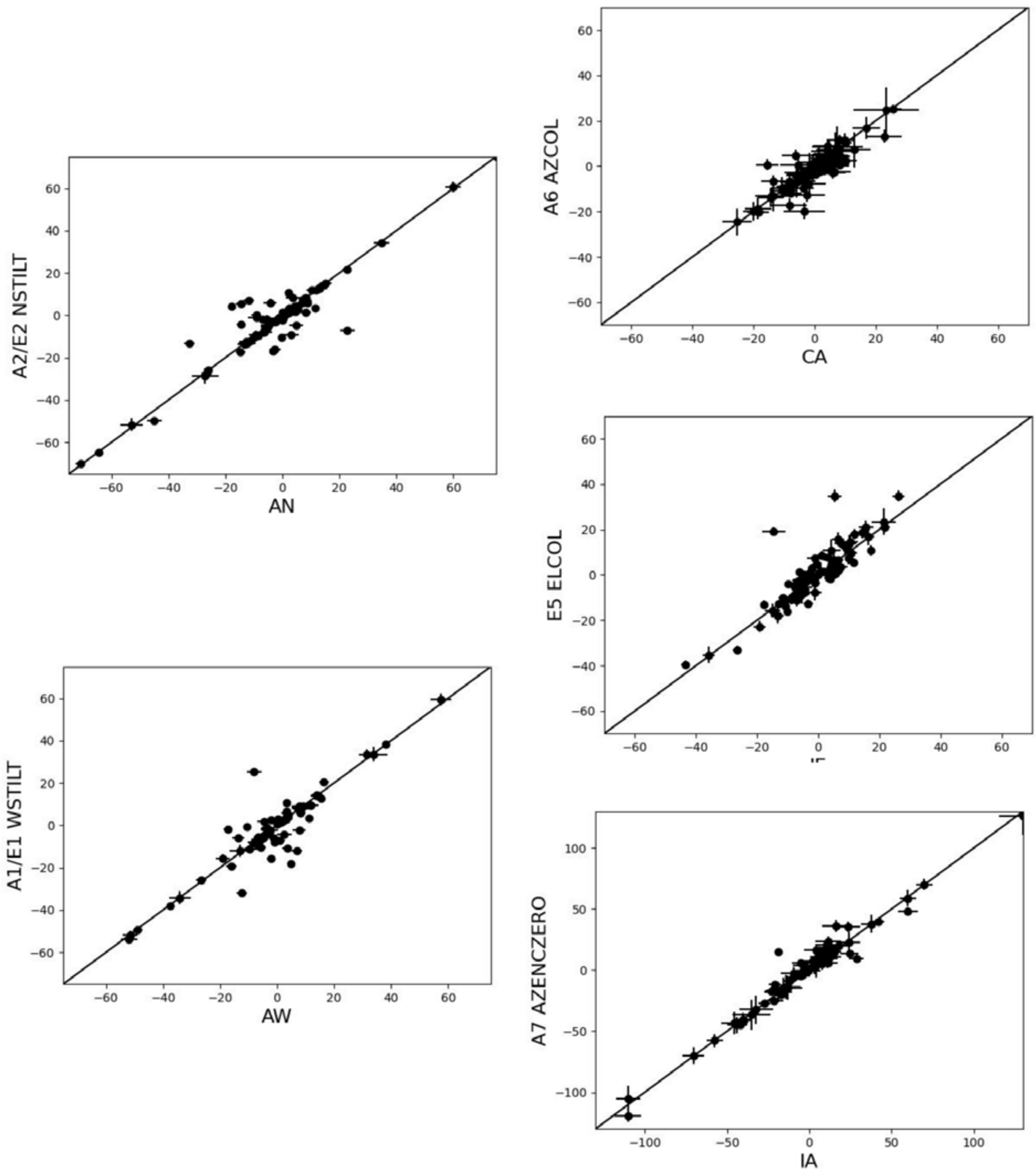

Figure 3: Differential model results of PEEK (y axis) and TPOINT (x axis). The terms are presented in arcsec.

In Figure 3, I show the correspondence between those five terms of the pointing model, computed using PEEK and TPOINT. The codes use different fitting algorithms, so there are minor differences in the results that may become larger on noise or sparse datasets. Typically, we fit 36 points around the celestial sphere, and the results using both codes trend to an agreement that we have a more sampled space of parameters. Anyway, despite an outlier, the results generally agree even in short runs with 36 scans - which also validates the conversion file script I made.

A quick comment needs to be made on the errors. Since they use different algorithms, they are slightly different. The peek uses linear regression to estimate the errors, while TPOINT penalizes the small sample that we use to get the model, which explains the higher errors on the x-axis.

Additionally, to increase the sample size, I have included long pointing runs that extend beyond the top, which affected PEEK solutions. Even then, the differences are majority in less than 10 arcsecs, which is the VLA accuracy for the global pointing model. In total, I have used 86 datasets from diverse antennas: 10 at C band and 76 at X band. Figura 4. I show a histogram of the differences between TPOINT and PEEK, considering all five parameters (total sample of 430), 392 are within 10 arcsec of difference. In Figure 4, I show the total histogram in bins of 1 arcsec.

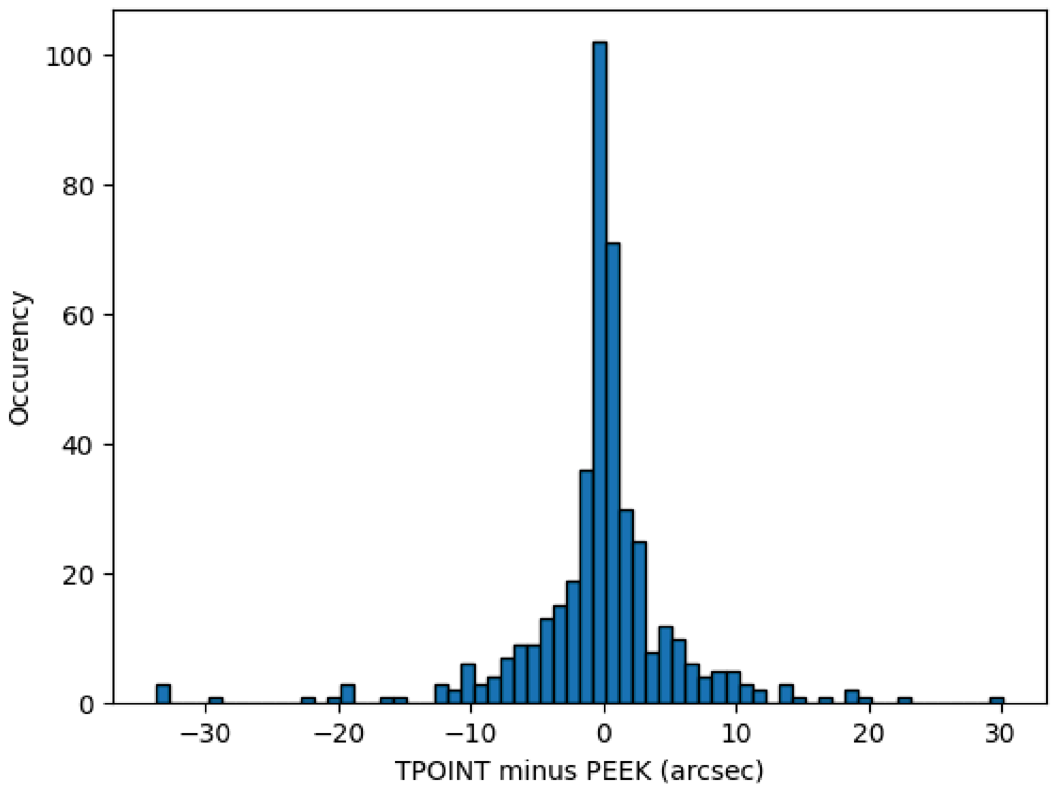

Figure 4: Histogram showing the differences between TPOINT and PEEK for the five pointing model parameters in our sample

6.2 Absolute pointing model of ea04

For the second strategy, I have used a special version of the pointing script used by the VLA operation to get pointing errors around the sky. Ken Sowinski made changes to set all terms of the model to zero before starting any scan. That choice was preferred over changing all the parminator terms to zero, so we would prevent our tests from interfering with any user observation by mistake. The disadvantage of such a strategy, we realize after running the test, is that during the interferometric pointing scan, the executor gets the collimation terms directly from the parminator, regardless of whether we have set them to zero prior to the scan. In other words, they were not set to zero as our initial intent. Below I summarize the main characteristics of the test:

- The collimation term in elevation and the perpendicularity plus collimation in azimuth were not set to zero. So, if we aim to fit an absolute pointing model, those terms need to be close to zero. That is more critical in the Azimuth error because there is a degeneracy between the perpendicularity and the AZ collimation term that requires a large dataset to be resolved.
- The tilt term set to zero encompasses the pad and the antenna tilt.
- All the other terms are set to zero.
- In total I have combined 3 observation days, totalizing 82 observations, which 20 of them were over-the-top.

I chose to use antenna ea04 because it had minor values on the model, and I could ensure we would not miss the target by setting the pointing model terms to zero. The ea04 also has the advantage of being close to the center of the array during our observations. The observation was performed on April 20th, May 20th, and June 3rd, 2024, and mostly during nighttime at C-band. We tried to get data with wind no higher than 8 m/s, but a few observations were obtained with higher wind. Nevertheless, we did not exclude any observation in the analysis. The script used is called pnt_bl_check.evla. On June 3rd, operations ran the script for two hours, soon after the usual C-band pointing script, which we could use as a reference if needed.

Table 2: Model solution using PEEK and TPOINT and the expected values. For the collimation terms where we expected zero, I show the value of Parmiantor in parenthesis. All values are in arc minutes.

TERM	NAME	EXPECTED	PEEK	TPOINT
CA	AZCOL	0.30	0.36 ±0.18	0.32±0.45
IA	AZENC	-3.28	-3.27 ±0.11	-3.23 ±0.36
NPAE	PERPENDICULARITY	-0.51	-0.35 ±0.20	-0.30±0.38
AN	NSTILT	0.63	0.56 ±0.03	0.58 ±0.04
AW	EWTILT	1.50	1.56 ±0.03	1.54 ±0.04
IE	ELCOL	0.00 (3.59)	0.09 ±0.04	0.11 ±0.08
ECEC	ELCENTERCOL	0.19	0.23 ±0.06	0.19 ±0.11
CA plus NPAE	AZCOL+PERP	0.00 (0.21)	0.01 ±0.38	0.02 ±0.83

In Table 2, I present the model fitted with Peek and TPOINT and the expected values (i.e., the values given in the Parminator, which was the model we had been using for ea04). To recover the expected pad tilt terms, I needed to read the antenna term and use equations (5) and (6) to get expected terms, since we had set everything to zero and we are fitting the total term. I also need to know the absolute pad tilt. That won't be necessary for the ngVLA prototype since the executor will not have a priori term to the test pad as we have now, and the tilt term we found can be directly fed in the parminator.

$$EWTILT_{total} = EWTILT_{pad} + \cos(Az_{pad})EWTILT_{ant} - \sin(Az_{pad})NSTILT_{ant} \quad (5)$$

$$NSTILT_{total} = NSTILT_{pad} + \sin(Az_{pad})EWTILT_{ant} + \cos(Az_{pad})NSTILT_{ant} \quad (6)$$

Where the total east-west tilt ($EWTILT_{total}$) and the north-sourth tilt ($NSTILT_{total}$) are written as a function the tilt terms of the antenna ($EWTILT_{ant}$ and $NSTILT_{ant}$) and the tilt terms of the pad ($EWTILT_{PAD}$ and $NSTILT_{PAD}$). The $\mathrm{Az_{pad}}$ corresponds to the azimuth of the pad, which is also present in the parminator. Since the total tilt was set to zero in our tests, we need to combine those terms on the parminator to get the expected value presented in Table 2.

Finally, because of the degeneracy of small datasets of the azimuth terms perpendicularity and collimation - meaning that they mascarade each other. As I mentioned above, the reader will notice that they both tend to zero together because they were not set to zero in our observations. I added an extra line in Table 2 to show it. It is curious that both PEEK and TPOINT were close to separate the absolute value of those terms, even with higher errors.

ACKNOWLEDGEMENT

I appreciate the discussion I had with Jeff Mangum and Ken Sowinski during the attempt to translate the PEEK solutions to the TPOINT values. I also appreciate the feedback Jeff provided on the memo text. I also thank Daniel Faes for working on a code to extract the value of the parminator on a given day; that made my work on the present memo easier.

References

Baars 1973 IEEE, 21, 461
Baars 2007, The Paraboloical Refletor in Radio Astronomy and Communication, Springer
Dumke & MacAuliffe 2010, SPIE, 7737, 77371J
Grave et al. 1996 AA(PS), 115, 379
Kong et al. 2014, RAA, 14, 733
Mangum 2001, ALMA Memo 366
Mangum 2024, ngVLA Antenna Memo 18
Mangum et al. 2006 PASP, 118, 1257
Meeks et al. 1968 IEEE, 16, 746
Patel et al. 2004, SPIE, 5496, 639
Penalver et al. 2000, SPIE, 4015, 632
Stumpff 1972, Kleinheubacher Berichte 15, 431
Wade 1990, VLA/VLBA Pointing Memorandum 101
Walker 2015, VLBA Test Memo 72
Wallace 2002, SPIE, 4848, 125

The National Radio Astronomy Observatory and Green Bank Observatory are facilities of the U.S. National Science Foundation operated under cooperative agreement by Associated Universities, Inc. This work was supported by awards AST-2034328 (MSIP Prototype Antenna) and AST-2334267 (ngVLA Design Activities); NRAO related activities are funded under award AST-1647378 (NRAO Operations/Development).