



The CHARA Array

Center for High Angular Resolution Astronomy of Georgia State University

Mount Wilson Observatory

Mount Wilson, California, 91023 USA

Notes on Tpoints and Pointing Solutions.

CHARA Technical Report #118

Theo ten Brummelaar

Version 1: 2022-03-10

1.0 INTRODUCTION

This technical report is intended as a brief HOWTO on creating a new pointing solution using Tpoints for the CHARA Array telescopes. The Tpoints program was written by P.T. Wallace and it used by almost all professional observatories. Our copy of Tpoints is kept in the “tpoint” repository in gitlab.chara.gsu.edu, and if it needs to be reinstalled there are instructions there on how that can be done. There is a PDF manual for Tpoints in the repository and I strongly recommend that you read it – I will not be repeating much of that information in this document other than to say that typing “help” works. Note that Tpoints is from 2012 so beware of OS upgrades stopping it from working.

2.0 COLLECTING THE DATA

Like the baseline solution pointing data is collected automatically, or at least can be if the right switch is on in `cosmic_debris`, but we’ve found that the automated data is not great for a new solution as it doesn’t take into account times when telescopes are re-initialized and data taken over many nights doesn’t produce a great solution. Once the pointing of a scope is poor it is best to re-initialize the scope on a bright star that is not close to the zenith and then collect pointing model data using the command “tpoint” or “tp” in a telescope server. This routine will move up and down in elevation and from 0 to 360 degrees in azimuth pointing at stars in the FK5 catalogue and ask you to center the stars. The correct and apparent azimuth and elevation for each of these stars is then saved in a data file that will either be in the `/ctrscrut/chara/data` directory that night or in the CHARA data archive later.

The tpoints routine in the telescope server has numerous parameters for make a model and it will ask you a series of questions. These are, in the order in which they will be asked:

1. Data list size. This is either small (about 50 stars), medium (about 100 stars), or large (about 200 stars). Usually small is sufficient and large takes way too long. If you have the time using medium is a good number of stars for a solution that will work well in all parts of the sky.
2. The V magnitude limit for the solution. The default is 6 which is normally fine.

The routine will then proceed to try and point at various stars, wait for you to place them in the correct position in the acquisition field and record the data. If you hit the escape key it will stop, otherwise it will keep going until it goes through the full 360 degrees of azimuth.

For each star the process is:

1. Find the next star in the FK5 catalogue. As mentioned above the routine moves up and down in elevation as it increases azimuth.
2. Slew the telescope to this star and wait for tracking to stabilize. If the telescope oscillations are large for some reason, most often poor tuning, you can bypass this wait by hitting the escape key.
3. Wait for the user to center the star and hit either the enter key to collect data or the 's' key to skip this star and move on.

This process is repeated until the full range of azimuth has been done and takes about an hour. Note that if the pointing solution is really bad you can make a solution based only on the finder scope, implement that solution, and then later make a new solution using the acquisition system. In the end you will have a file with a name like `2021_08_03_S1_point_001.dat` whose contents will look like this:

CHARA S1

: NODA

```
34 13 33.31 2021 8 3 10 750.0 1725.210000
13.733452,29.409420      13.597229,29.418807
21.398002,43.867541      21.309450,43.870148
38.533472,37.052331      38.408724,37.059963
39.447940,52.153866      39.391696,52.151842
42.518544,67.153296      42.594511,67.141019
60.341361,38.925220      60.212034,38.932183
70.394413,58.261671      70.355114,58.258006
81.376923,36.159408      81.230060,36.172680
103.580230,67.715458     103.610545,67.715222
109.507533,33.851806     109.346699,33.872252
130.864432,39.462945     130.714149,39.489483
131.927825,70.273084     131.989041,70.280218
139.810876,25.260205     139.622996,25.297639
145.219196,47.574820     145.095815,47.600469
159.230067,34.022332     159.062542,34.059662
175.408892,50.314556     175.299154,50.346218
179.390534,29.927541     179.210362,29.971703
181.649999,61.882040     181.623352,61.906842
196.754367,36.731498     196.594612,36.774085
226.140484,37.831508     225.988178,37.873650
232.196134,70.026803     232.310670,70.048510
219.717485,55.710383     219.656496,55.742258
248.601513,35.083707     248.446269,35.125079
256.987572,57.661331     256.955954,57.683966
```

257.511646,21.079141	257.314651,21.124666
267.823951,41.124577	267.691970,41.157778
268.062862,61.676248	268.069476,61.697079
283.421718,36.070759	283.272196,36.102128
292.604910,51.643418	292.527908,51.662388
309.591647,37.746316	309.440667,37.767792
318.193693,64.193691	318.229994,64.196674
328.504858,42.434170	328.372260,42.447768
346.916533,53.340620	346.834854,53.344309
358.039551,78.644484	358.029101,78.659487
352.082897,41.922240	351.938588,41.931207

Here the first few lines contain some global information and the remaining entries have the “true” star position followed by the “apparent” star position. If you want some data to play with cut and paste these data into a file and it should be fine.

3.0 RUNNING TPOINTS

Once you have the data in a working directory the first step is to get a copy of the current pointing model. These are kept in the /ctrscrut/chara/etc directory and have names like “dome_S1.tps”. For example, the current S1 pointing model at the time of writing this document contains:

```
CHARA S1
T 32 24.3671 43.066 -0.0489
& HASA +58.4178 10.74726
  NPAAE +459.9306 13.89048
  AN -55.6171 5.89738
  AW +37.3030 4.92186
  TF +171.4830 25.94503
END
```

The HASA term is an azimuth sine harmonic term and the other terms represent telescope flexure of one sort or another and non-perpendicularity of the elevation and azimuth axes. There are many possible terms to use but we’ve found that starting with the same terms that are currently being used is a good starting point.

It is now time to start tpoints by simply typing “tpoint”. You should see something like:

```
+-----+
|           TPOINT           |
| Telescope Pointing Analysis System |
|           Version 18.22           |
+-----+
```

Copyright 2012 P.T.Wallace - all rights reserved.

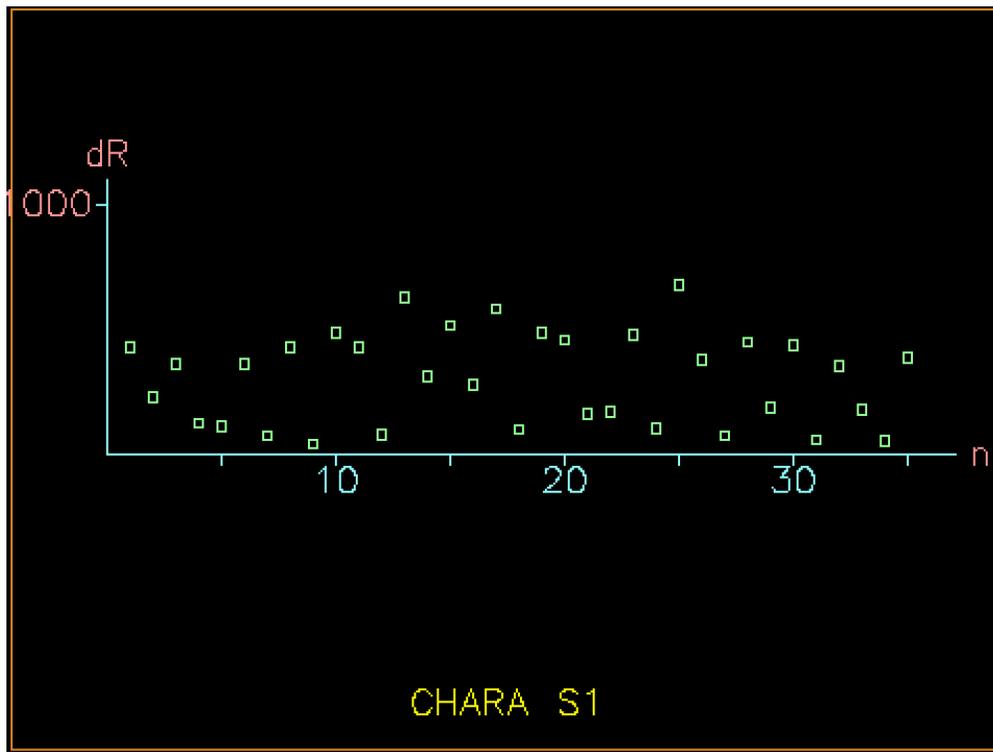
This copy of TPOINT is licensed for use on the CHARA Interferometer, Mt. Wilson. Supplied July 21st, 2012.

There are 70 standard pointing terms.
Reading procedures from file procs.dat ...
The library now contains 230 lines.
Reading star catalog entries from file stars.dat ...
The catalog contains 210 stars.

TPOINT ready for use: type HELP for assistance, END to quit.

*

Where the '*' is the tpoint prompt. You then load the data using a command like "indat 2021_08_03_S1_point_001.dat" and then the old model using a command like "inmod dome_S1.tps". This is normally a good time to see how well the old solution works on these data by plotting the pointing errors. There are many plotting commands (RTFM) but one of the more useful ones is a macro "call a9" which puts several good diagnostic plots up at once. A simpler but still very useful plot is to plot the errors in the order they were collected use "g r n" which in this case looks like this:



You can see that the fit isn't great with errors in the hundreds of arcseconds, but that's OK at this point. The next step is to add the two parameters IA and IE using the command "use IA IE" and then perform a fit using the command, and this will surprise you, called "fit". The output should look something like this:

```
* use ie ia
* fit
  coeff  change  value  sigma
```

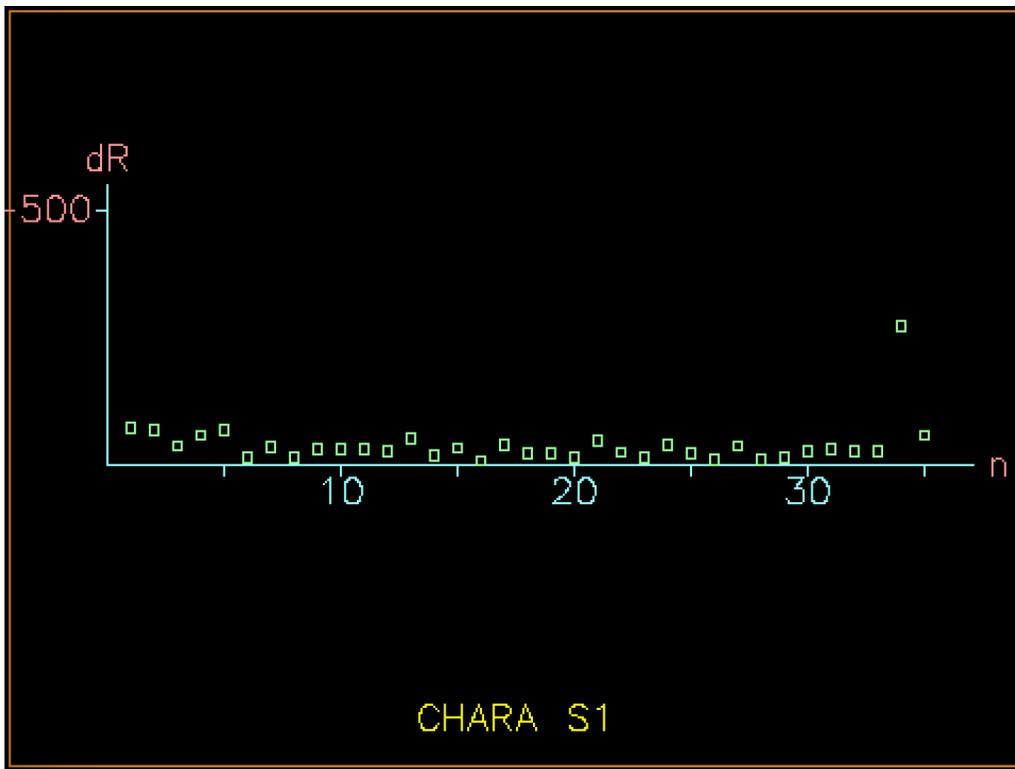
1	& HASA	-6.553	+51.86	23.810
2	NPAE	-72.493	+387.44	28.835
3	AN	+5.656	-49.96	13.300
4	AW	-8.245	+29.06	11.334
5	TF	-43.948	+127.54	53.332
6	IE	+15.634	+15.63	36.607
7	IA	-814.258	-814.26	30.506

Sky RMS = 58.50

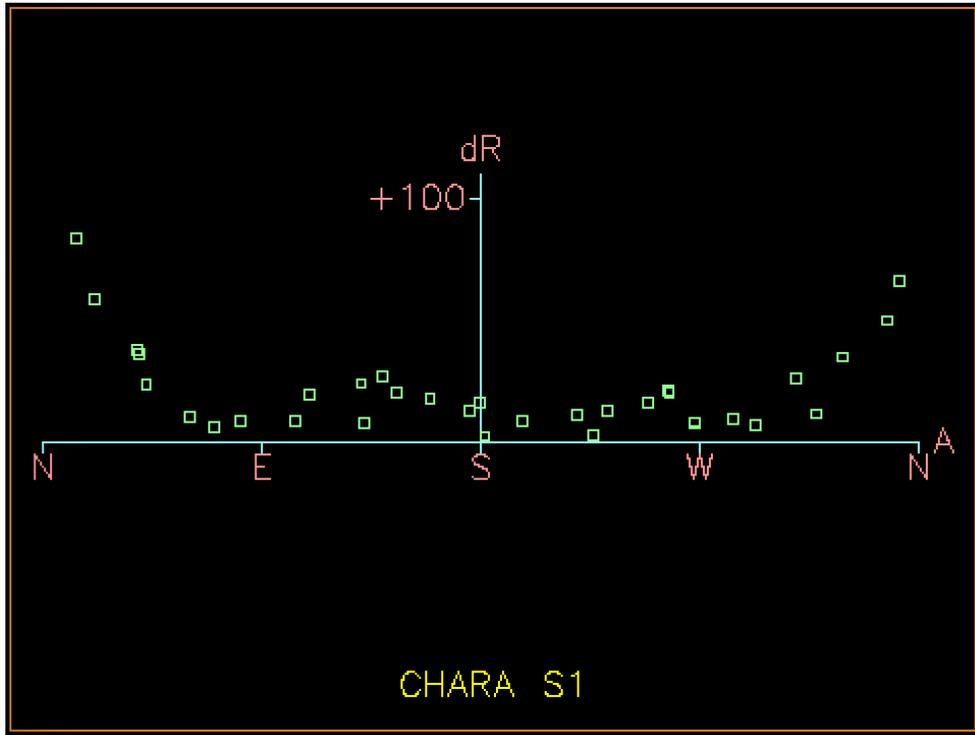
Popn SD = 65.40

Observation #34 is an outlier candidate.

So now we are down to an RMS error of about 1 minute of arc. The reason we add the IA and IE parameters is that they represent an origin offset in the azimuth and elevation. We don't use these values on the sky as they are the equivalent of re-initializing the scope on a star, but they are necessary to get a decent fit to the data in tpoints. The plot of errors now looks like this:



The fit is a lot better, but there is one point that is an outlier, and if you go back to the output of the fit Tpoints actually warns you about this saying that “Observation #34 is an outlier candidate”. You can remove this point using the command “mask 34” and rerun the fit with the result that the RMS error goes down to 28.54 arcseconds, which is fine. If you plot the pointing errors against azimuth using “g r a” you will now get something like this:



This makes it clear that there is something going on with the azimuth part of the pointing model which can often be resolved by adding the term PAA1 discussed below.

4.0 TRYING HIGHER ORDER TERMS

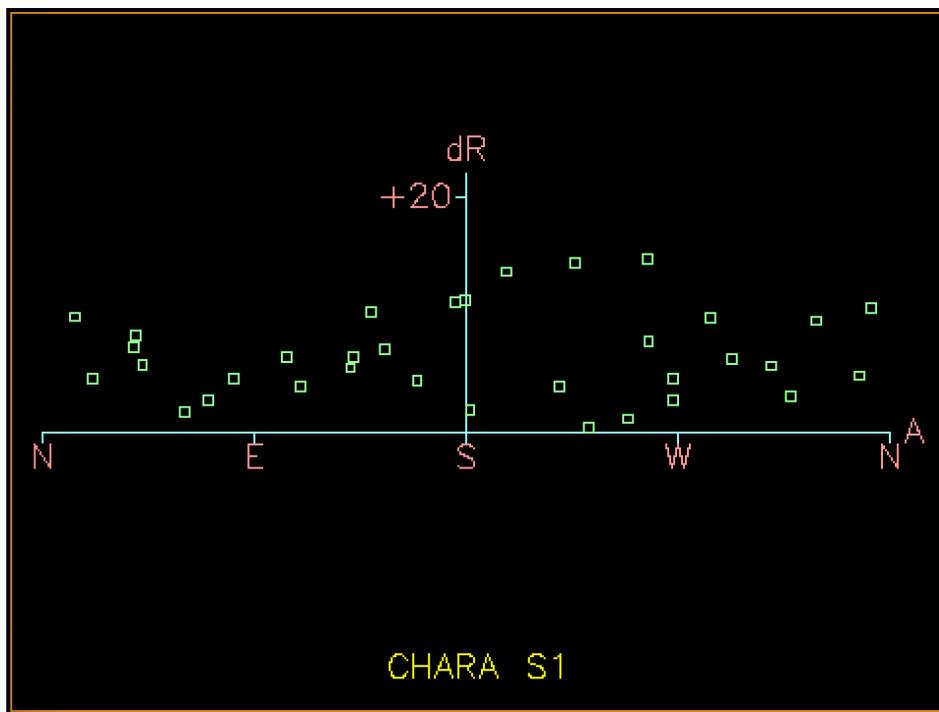
If you are keen to try and get a better solution feel free, it can be fun, but be warned that it can be a real rabbit hole and you can end up trying to have as many terms as there are data which is never a good idea. If you really want to get a much better model you should collect more than the 50 odd stars of the smaller collection routine and get at least 100.

Note also that our system doesn't understand all the terms that tpoint provides as they have not all been implemented. At this time the terms that are implemented for an Alt/Az mount by default are

- TF tube flexure
- FO fork flexure
- NP HA/dec nonperp
- CH boresight/dec nonperp
- IA azimuth index error
- IE elevation index error
- NPAE Az/El nonperp
- CA boresight/el nonperp
- AN azimuth axis tilt north
- AW azimuth axis tilt west
- TX Alternative tube flexure
- PAA1 Azimuth scale multiplier
- PEE1 Elevation scale multiplier

The harmonic terms of the type HASA#, HACA#, HESE#, HECE#, HSSA#, HSCA#, HSSE# and HSCE# where '#' is the order are supported up to order 10. We have never found that the higher orders help much except in causing confusion.

The last two terms in the list above, PAA1 and PEE1, are special and can often help, especially PAA1. They are a multiplier of the scales for the two axes. We have rarely found that PEE1 helps, but PAA1 can often help if the encoder radii are not exactly the same. Note that if you use either of these terms you can **not** leave them in the final model file. For esoteric reasons I will not go into here they can't work in the standard way and need to be implemented in the command line of the dome_server program using the -A and -E flags. This means changing the socket_manager.list so only do this if it really helps the solution. The standard recommendation is to use PAA1 and not PEE1. At this time there are -A flags in the socket manager list for all scopes except S1. Ironically, I have just found that adding PAA1 to these data reduces the RMS error of the fit to 7.62 arcseconds and the "g r a" plot now looks like this:



It would be a good idea to collect a new set of data and new solution for S1 soon and add a -A flag to it in this way, but I will not do it at this time as this solution is based on quite old data.

Finally, there is a command to automatically find a solution called "fauto". You will not find this in the manual as it was added by Patrick experimentally and he was kind enough to give us a version that contains this. We've found that the fauto routine is at best a way to give you clues as to which parameters might help the fit, but it's best to stick with a manually construct fit.

5.0 USING THE NEW MODEL

The final step is to install the new model into the on-sky system. You can create a new pointing model file using a command like "outmod dome_S1.new". This then needs to be copied to the working file /ctrscrut/chara/etc/dome_S1.tps where of course you need to change the telescope name. There are a few important things to remember in this process:

1. Make a backup of the old pointing model.
2. Remove the IA, IE, PAA1 and PEE1 terms from the file before you install it.
3. If you have used PAA1 or PEE1 (I recommend against using PEE1) you need to change the command in the socket_manager.list file.

Once all this has been done you need to restart the dome_server. There should one day be a way of reloading the pointing model while the program is running, I just haven't ever got around to it as it doesn't happen very often. Once you have installed the new model it's best to try it out on a few stars before leaving an observer in the lurch. You should also warn everyone that a new model is in place and that the telescope in question needs to be re-initialized.

6.0 THOUGHTS ON AUTOMATING THE PROCESS

If we assume that there will be minimal outliers it should be possible to automate at least part of the process, if not all of it. I checked the existing pointing models just now and they all contain the AN, AW, CA, NPAE, TF and TX parameters. Many of them also contain various amounts of harmonic parameters and all but S1 has a PAA1 parameter in the socket_manager.list file. Incidentally while checking this I found W2 had PAA1, IA and IE parameters in the model file which I have removed as they should never be in there as explained above. It may be that a single set of useful parameters can be found that will work for all scopes and I suggest investigating this.

Even if that is not true, it would be relatively easy to automate at least the fitting process if we skip the removal of outliers. The script should do the following things:

1. Obtain the current model and assemble the pointing data.
2. Load the data using the 'indat' command.
3. Load the existing model using the 'inmod' command.
4. Add the IA, IE and PAA1 parameters with 'use IE IA PAA1'.
5. Run 'fit' several times till the model converges.
6. Output the final model using 'outmod'.
7. Remove the IA, IE and PAA1 parameters from the model.
8. Place the new PAA1 parameter in the socket_manager.list file.
9. Copy the new model to /ctrscrut/chara/etc after backing up the old one.
10. Restart the dome server.

Even if this process is not automated if you add removing outliers and experimenting with higher order terms this represents a simple recipe for creating a new model.

Finally, while we gave up using the pointing data collected automatically by cosmic_debris we did that many years ago on the assumption that the scopes were being re-initialized regularly. It may very well be that it is worth investigating using these data again. After all, it works for the baseline solution.