The CHARA Array
Center for High Angular Resolution Astronomy
of Georgia State University
Mount Wilson Observatory
Mount Wilson, California, 91023 USA

Notes on Iphase and Baseline Solutions.

CHARA Technical Report #115

Theo ten Brummelaar

Version 1: 2022-03-01

1.0 INTRODUCTION

This technical report is intended as a brief HOWTO on creating a new baseline solution (BLS) using Iphase for the CHARA Array. The Iphase program was written by P.T. Wallace, the same person that wrote Tpoints, for the VLTI. It works very much like Tpoints, so if you are familiar with that you will find Iphase easy to learn. Our copy of Iphase 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 iphase 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 Iphase is from 2008 so beware of OS upgrades stopping it from working.

2.0 COLLECTING THE DATA

Baseline solution data is collected automatically during normal observations, and also at times manually on specially assigned nights for that purpose. All CHARA BLS data is in the archive with filenames like "2021_11_29_Baseline_Solution_no_pops.dat". The "no_pops" part means that these data have not been changed to take into account which pops are used. There are also files with filenames like "2021_11_29_Baseline_Solution.dat" in which the pops and beam information has been removed. We found long ago that these do not make better solutions that including the POP information, but feel free to experiment with them.

Each entry in these files consists of three lines:

1. The first line designates which scopes, pops, and beams where used, for example "D/S2E1P21B23" meaning that this is data for S2-pop2-beam2 and E1-pop1-beam3.
2. The second line, which is too long to show here, lists a lot of information about the instrument setup including which beam combiner, the position of the LDCs, the time and so on. This can be used as shown below for editing these data.
3. The third line contains a header followd by the Alt/El position of the star and the position of the two delay line carts in optical path, for example:

```
    N_S2-E1   248.072191229    48.059579225   84.750955440    22.503952947
```

The normal procedure is to choose the date range over which you wish to make a BLS, collect these files together in a fresh directory and concatenate them into a single file. Once you have done this you need to put the two lines:

```
CHARA ALL

:NOSITE
```

At the top of the file.

You are now in a position to do some editing on this file using the program “cleaniphase” which has a number of flags:

```
usage: cleaniphase [-flags] {datafile}
Flags:
-C[COMBINER]Restrict data to one combiner (OFF)
-f              FK5 dtars only (OFF)
-h              Print this message.
-v              Verbose mode (OFF)
-z              Reject zero offset entries (ON)

-p[maxPM]     Set maximum proper motion (mas/Yr).
```

This allows you to restrict the data set to a single beam combiner, to use only FK5 stars, reject data lines with no offsets, or to restrict the data to stars with a given maximum proper motion. In practice we’ve found that only the first of these is useful, though it’s worth looking into the other options in more detail at some point.

3.0 PREPARING THE IPHASE COMMAND

The Iphase program has many commands, and with a large data set it can be quite complex to create the set of commands you need. The program “iphasecmd” can help here:

```
usage: iphasecmd [-flags] {datafile} {telescopes.chara}
Flags:
-c        Force current solution (OFF)
-e        Use EC in fit (ON)
-s        Set current solution (OFF)
-S        Use SAS and SAC in fit (OFF)
-h                Print this message.
```

The iphasecmd program will read your data file and spit out the complete set of iphase commands you will need to start work on fitting these data. The command set it produces assumes that S1 is the origin of the solution, fitting the X, Y, Z, EC, SAS and SAC parameters for each of the other scopes across the entire data set, and fitting a separate internal path for each combination of POPs. The later of these is a compromise as attempting to fit all the POPs as well, or removing them using the current baseline solution as done in the alternative data files discussion above, has not been very successful. By default, the EC parameter (-e), the change in internal path with changes in elevation, is always included. The SAS and SAC (-S) parameters, which cover the change in internal path with azimuth, are off by default as we have not yet found that they help very much. It is also possible to include an old solution as a starting point (-s), or as a fixed set of points (-c) which can be useful for testing old solutions and making comparisons with new ones.

In the end, once you have decided on the options create a set of iphase commands and save them in a file. Note that there will be a lot of them! An incomplete example of the output of iphasecmd is:

```
use S1.X S1.Y S1.Z
S1.X 0.0
S1.Y 0.0
S1.Z 0.0
FIX S1.X S1.Y S1.Z
use E1.X E1.Y E1.Z
use E1.EC
use S1-E1.A/S1E1P11B51
use E2.X E2.Y E2.Z
use E2.EC
use E1-E2.A/E1E2P12B16
use W1.X W1.Y W1.Z
use W1.EC
use W2.X W2.Y W2.Z
use W2.EC
use W1-W2.A/W1W2P45B32
use S2.X S2.Y S2.Z
use S2.EC
use S2-W2.A/S2W2P25B42
use S1-W2.A/S1W2P15B52
use E2-W2.A/E2W2P25B62
use S1-W1.A/S1W1P14B53
use E2-W1.A/E2W1P24B63
use S1-S2.A/S1S2P12B54
use S2-E2.A/S2E2P22B46
use S1-E2.A/S1E2P12B56
use E1-W2.A/E1W2P15B12
use E1-W1.A/E1W1P14B13
use S2-E1.A/S2E1P21B41
use S2-W1.A/S2W1P24B43
use W1-W2.A/W1W2P35B32
use S2-W1.A/S2W1P23B43
use S1-W1.A/S1W1P13B53
use E2-W1.A/E2W1P23B63
use E1-W1.A/E1W1P13B13
use E1-W2.A/E1W2P35B12
use E1-W1.A/E1W1P31B13
use S2-E1.A/S2E1P23B41
use S1-E1.A/S1E1P13B51
use E1-E2.A/E1E2P34B16
```

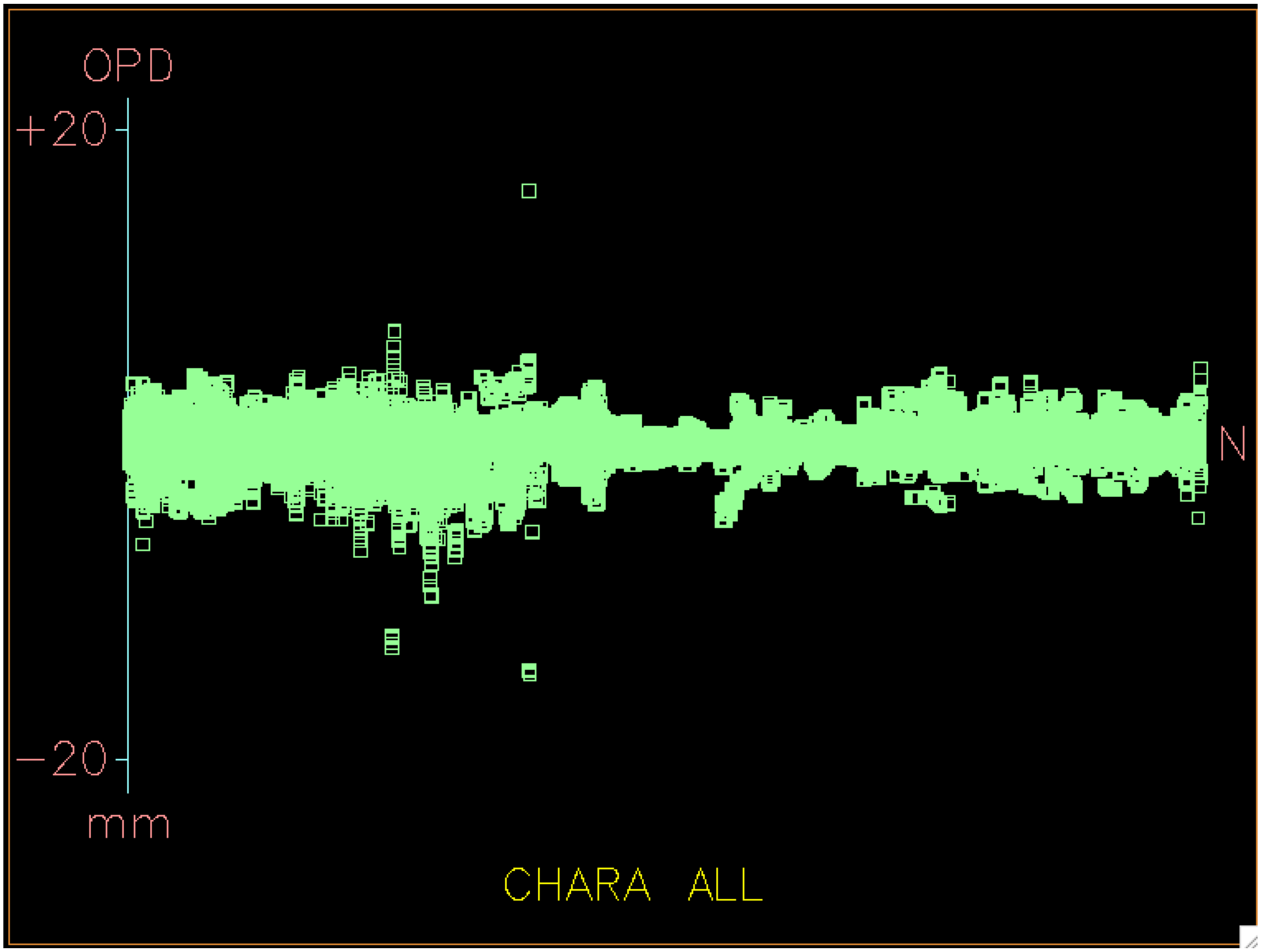
4.0 RUNNING IPHASE

As mentioned above, it's important that you know how to run iphase. If you're familiar with tpoints this won't take long but if not RTFM. The first step is to read in the data file:

indat filename.dat

You will see all the data shoot by, along with some messages concerning why some data points are rejected. There are many plotting routines in iphase and I recommend you play with them and look at the data set before fitting anything. The two most useful are "gsmap" which plots all the data in azimuth and elevation, and "g n" which plots the delay errors in the same order as the data set. This is helpful later when it comes time to edit the data.

The next step is to copy and paste the full set of iphase commands produced by iphasecmd. You are now ready to try and fit the data using the command "fit". Beware that this can take many minutes if there is a lot of data, which there usually is. Once the fit is complete you can use "gsmap" and the "g" command again to see how it went. Normally you will see errors at about the mm level with a number of obvious outliers as shown in the example below:



Most of these outliers don't have a big impact on the final fit, but you can start masking them by using, and this will surprise you, the "mask" command for example:

```
mask w g 10000
mask w l -10000
```

Will remove all those outliers either greater than 10mm or less than -10mm. In this case doing this reduces the rms error from 1375µm to 1321µm. It's worth playing around with masking different parts. For example, in some cases there are obvious places where the data just isn't very good and those can be removed with "mask 100 200" which will remove data points 100 through 200.

5.0 USING THE NEW MODEL

Once you have a model you like write it to a file using the command "outmod filename". This will save the current fit into a file called "filename.dat" and you can exit iphase at this point. You can start again from the same place by using the "inmod" command later.

The final step is to get the fit into a format suitable for our BLS files, the default name being "telescopes.chara" with the file held in the directory "/ctrscrut/chara/etc". This is done using the "readiphase" command:

```
Usage: readiphase [-flags] input_file [scopes.chara [output_file]]
Flags:
-b      Fit BEAMS by delta (OFF)
-L      Fit light using current beams (OFF)
-l      Fit LIGHT by delta (ON)
-r       Toggle using refractive index (ON)
-s      Fit LIGHT assuming beam delta same in all beams (OFF)
-p      Fit POPS by delta (ON)
-x      Use XYZ positions from fit (ON)
-h      Print this message.
```

Most of the flags are rarely used, but it is common to add an old telescopes.chara data file which is used as a starting point for the remainder of the process. The output of readiphase looks like this:

```
# WORKING OUT POP POSITIONS USING DELTAS.
# Calculated S1 P2-1 = 36.563 based on 16 points
# Calculated S1 P3-2 = 36.567 based on 14 points
# Calculated S1 P4-3 = 36.595 based on 13 points
# Calculated S1 P5-4 = 33.311 based on 10 points
# Calculated S2 P2-1 = 36.544 based on 16 points
# Calculated S2 P3-2 = 36.569 based on 15 points
# Calculated S2 P4-3 = 36.590 based on 13 points
# Calculated S2 P5-4 = 33.310 based on 13 points
# Calculated E1 P2-1 = 36.586 based on 9 points
# Calculated E1 P3-2 = 36.535 based on 6 points
# Calculated E1 P4-3 = 36.587 based on 6 points
# Calculated E1 P5-4 = 33.313 based on 5 points
# Calculated E2 P2-1 = 36.575 based on 7 points
# Calculated E2 P3-2 = 36.537 based on 6 points
# Calculated E2 P4-3 = 36.578 based on 4 points
# Calculated E2 P5-4 = 33.342 based on 4 points
# Calculated W1 P2-1 = 36.575 based on 2 points
# Calculated W1 P3-2 = 36.537 based on 2 points
# Calculated W1 P4-3 = 36.593 based on 1 points
# Calculated W1 P5-4 = 33.312 based on 1 points
# Calculated W2 P2-1 = 36.600 based on 0 points
# Calculated W2 P3-2 = 36.544 based on 0 points
# Calculated W2 P4-3 = 36.605 based on 0 points
# Calculated W2 P5-4 = 33.306 based on 0 points

# For telescope S1: value stddev delta (total delta 0.000)
XOFFSET            0.000     0.000     0.000
YOFFSET            0.000     0.000     0.000
ZOFFSET            0.000     0.000     0.000
EC            0.000     0.000     0.000
LIGHT         0.000     0.000     0.000

# For telescope S2: value stddev delta (total delta 9144.235)
XOFFSET        -5745754.818    61.115  -962.480
YOFFSET         33577551.415   112.245  -362.482
ZOFFSET           654377.851   596.125 -7281.775
EC          -10673.979   355.208  5434.613
LIGHT   4105949.724   391.479 -10289.822

# For telescope E1: value stddev delta (total delta 1624.966)
XOFFSET       125334868.564    92.348 -1120.701
YOFFSET       305927558.474   113.549  -668.836
ZOFFSET        -5915897.249   667.560   949.796
EC           2551.706   403.764   187.319
LIGHT  10761247.238   437.757 481725.785

# For telescope E2: value stddev delta (total delta 5826.176)
XOFFSET        70391896.054    80.597 -1543.738
YOFFSET       269712667.356   107.699   129.437
ZOFFSET        -2789410.971   614.204 -4463.625
EC          -4260.838   378.930  3408.885
LIGHT  22683595.062   542.855 -6676.798

# For telescope W1: value stddev delta (total delta 6570.151)
XOFFSET      -175074921.463    85.320  -480.069
YOFFSET       216320674.635   102.377  -896.764
ZOFFSET       -10792227.771   594.101  5831.487
EC           -646.443   357.591 -2850.612
LIGHT  27305867.768   555.268  5345.519
```

```
# For telescope W2: value stddev delta (total delta 2750.460)
XOFFSET       -69090034.125    62.639   114.980
YOFFSET       199332028.473    98.486   803.419
ZOFFSET          470846.648   592.956 -2192.765
EC          -5069.914   355.147  1448.485
LIGHT -10864097.886   934.876 -1829.393
```

The first section is readiphase doing its best effort to reconcile the internal path fits between each pop combination into the difference between each set of consecutive pops in each line. These can, if you want, be converted into the POP1 type entries in the new solution. Keep in mind that the so-called default POP is always 0. I don’t do this each time and have been using the POP positions measured by Judit many years ago. As we often say, everything is moving all of the time, and a POP error is a constant term offset that isn’t usually a big deal.

The second section are new entries you can use for the new model, but you will need to manually edit the file to use them. Please make sure to keep a copy of the old file before changing it.

6.0 OTHER USEFUL COMMANDS

There are several other commands in the src directory of the tpoints repository but only two of them are commonly used:

```
usage: chkbls [-flags] datafile {telescopes.chara}
Flags:
-h            Print this message.
-v            Verbose mode.
```

This command will read the data file you give, that is the input data file for a new BLS, and calculate for each data point the difference between the measurement and the BLS file you give it. An example of the output is:

```
S1/POP3/BEAM2-S2/POP3/BEAM3  126.095   63.334 -10.858343 -10.859067    724.1
S1/POP3/BEAM2-E1/POP4/BEAM1  126.095   63.334  46.500325  46.502709  -2384.0
S2/POP3/BEAM3-E1/POP4/BEAM1  126.095   63.334  57.358668  57.361777  -3108.1
S1/POP3/BEAM2-S2/POP3/BEAM3  150.790   51.489  -6.305801  -6.306599    797.8
S1/POP3/BEAM2-E1/POP4/BEAM1  150.790   51.489  92.546081  92.548395  -2314.9
S2/POP3/BEAM3-E1/POP4/BEAM1  150.790   51.489  98.851882  98.854995  -3112.7
S1/POP3/BEAM2-S2/POP3/BEAM3  149.716   30.437  -2.465639  -2.466453    814.1
S1/POP3/BEAM2-E1/POP4/BEAM1  149.716   30.437 114.279026 114.281670  -2644.1
S2/POP3/BEAM3-E1/POP4/BEAM1  149.716   30.437 116.744664 116.748123  -3458.2
S1/POP3/BEAM2-S2/POP3/BEAM3  199.318   83.434 -14.667759 -14.668492    733.5
```

Where you get the configuration, azimuth, elevation, cart positions and error in microns.

```
usage: diffbls [-flags] telescope1.data telescope2.data
Flags:
-CS1F1     Change from default to alt scope.
-v              Toggle verbose mode (OFF)
```

Which will tell you the difference between two BLS files.