The OYSTER Menu

Based on **The Hitch Hiker's Guide to OYSTER**:
A quick reference to NPOI Data Reduction and Calibration

About this guide:

This guide consists of two parts. The first lists the steps that you will most likely need while reducing data from the NPOI in Oyster. The second part is a glossary. If any word or phrase is **in bold print,** you may find more detailed information about it in the back of the guide for extra help. This is a reference guide meant for people with little Oyster experience, or to be used as a checklist. A much longer and more technical guide can be found on Christian Hummel's website.

-Haley Hurowitz

1. After you have **downloaded your data**, you may start the program. Type:

   - **idl /directory oyster is in/oyster** #start OYSTER
   - get_data,'YYYY-MM-DD.con' # load constrictor file
   - **list_scans** # Check that the configurations are correct
   - oyster #bring up the main widget

   Note that at any point, if you feel something is particularly wrong with your data you may look at the **observing log** to try and find some answers. Other useful tools for oyster process are: **Identifying Points** and **Camera.**

2. Edit the data on FDL Residuals

   - Reduce → Point Data → Astrometry → PLOT
   - Y-axis: FDLDelay (res.)
   - X axis: time
   - IndexSelection, y widget: InputBeam: 1
   - Delay Plot widget: Plot →Screen
   - **Hand flag obvious bad data**
   - Edit → Auto #Auto edit outliers
   - Plot → Screen
   - Select the remaining InputBeams on the IndexSelection widget and repeat this process to edit outliers.

   You do not need to edit **reference beams**.

3. **Save the flagtable** so that you may **reload the flagtable** later.

4. Edit data based on delay line jitter:

   - Oisdr widget: Reduce → Point Data → Imaging → PLOT
   - Y-axis: Delay Jitter
   - X-axis: time
   - OutputBeam:1
   - Baselines: All
   - Plot → Screen
   - Edit → Auto
   - Plot → Screen
   - OutputBeam: 2
   - Plot → Screen

- Plot → Auto
- Plot → Screen

If there are still obvious outliers after auto editing, hand flag it. Sometimes you may need to **select baselines**.

5. Edit based on Nat Counts

    - PointPlot widget: y-axis: NATCounts
    - Auto edit all the inputbeams

6. Edit based on NAT Jitter

    - PointPlot widget: y-axis: NATJitter
    - Auto edit all the inputbeams

7. Edit based on photon rates

    - Y-axis: PhotonRate
    - X-axis- time
    - OutputBeam : 1
    - Channels: all
    - Plot → Auto
    - OutputBeam: 2
    - Plot → Auto

If you have older data you may have 32 instead of 16 baselines. If this is the case, you may want to **select channels** to see the data better. Also make sure you check **channel 11** on OutputBeam 1 because there is often leakage.

8. Edit bad squared visibility amplitude data

    - Y-axis: VisSq
    - Channels: All
    - Baselines: All
    - OutputBeam:1
    - PointPlot widget: All in 1
    - Auto edit
    - OutptBeam: 2
    - Auto edit

Note sometimes you may need to get rid of **channel 4** output beam 2.

9. Edit the Background Data

    - Reduce →BG Data→Process BG
    - Reduce →BG Data→ Expand BG
    - oisdr widget: Reduce→BG Data→Plot
    - y-axis: BGRate
    - Channels: All
    - OutputBeam:1
    - Plot→Screen
    - Edit the data
    - OutputBeam:2
    - Plot→Screen
    - Edit the data

Note that auto editing does not work here. You must edit by hand, but only edit extreme outliers.

10. The bias correction should be done next.

    - oisdr widget: Reduce→Average
    - oisdr widget: Calibrate→ System
    - CalibrateSystem widget: Type: V2Bias
    - All in 1
    - Stars: Sel
    - IndexSelection,y widget: Channels: All
    - Baselines: All
    - OutputBeam:1
    - Points: **pick numbers**
    - CalibrateSystem widgets: select each target star and **calibrator** and **plot to screen** (Plot→Screen). Remember to do these **individually**.
    - OutputBeam:2
    - Plot stars to screen individually
    - **Bias correct triple data** if neccessary

11. Write a chameleon (scan-averaged) file for calibration.

    - Reduce→Average
    - Calibrate→Visibility→FlagIncoh #Flag the incoherent scans
    - Access→ Write→HDS #Write a .cha file
    - Type: hds_close #close the .con file
    - Get_data, “YYYY-MM-DD.cha’ #open the .cha file

Note that you are now working on the chameleon file instead of the constrictor file. This means that you cannot save your new work through the flagtable. Now you **save the chameleon file.**

12. We are now ready to calibrate. We can start with the Triple Phase.

- Calibrate→Visibility→PLOT
- Y-axis: **Triple Phase c**
- X-axis: Channel
- Slice: Ch
- Stars: Sel
- **Select your target and calibrator stars**
- All in 1
- Channels:All
- Triple:1
- Plot→Screen
- Calibrate→Visibility→Calibrate
- Visibility Calibrate widget: Use:Sel (then select your calibrator star)
- Variable: TriplePhase
- Loop
- Time #this makes time an independent variable
- Time widget: S_80
- **Unwrap the data** (if necessary)
- Visibility Calibrate widget: Calibrate
- Triple:2
- Calibrate

Most of the time you will only have two triples. On the off chance you have fewer (or more) than two, calibrate all the triples you can.

13. Now calibrate the Triple Amplitude

- VisibiltyPlot widget: y-axis: TrplAmp c
- Everything on the IndexSelection,y, Visibility Calibrate, and Time widgets should be already set up correctly from step 12.(Except you must change 'Variable:' on the Visibility Calibrate widget to 'TripleAmp')
- Calibrate
- Select the other triples and calibrate those as well.

14. Calibrate the Visibility Squared data

- VisibiltyPlot widget: y-axis: VisSq c
- Everything on the IndexSelection,y, Visibility Calibrate, and Time widgets should be already set up correctly from step 12.(Except you must change 'Variable:' on the Visibility Calibrate widget to 'VisSq')
- Calibrate
- OutputBeam:2 (for VisSq you do not need to select different triples only different OutputBeams)
- Calibrate
- Save: Access→Write→HDS

15. **Fit a model** to the data

- After you have obtained your **model file,** go to the oisdr widget: Model→Read
- Select the appropriate model file. (If you have multiple targets for this date, select one model file. After you have obtained the final position angle and separation you may come back to this point and select the next model file.)
- Click 'OK'
- Model→Calc
- VisibilityPlot widget: y-axis:VisSq c
- X-axis: Channel
- Slice:ch
- Select target star (ONLY the target star)
- Errors
- Model
- OutputBeam:1
- Plot→Screen
- Also check the second OutputBeam. (Remember you can select baselines if there are too many plots.)
- oisdr widget: Fit→INTERFEROMETRY
- Select both Rho and Theta
- oisdr widget: Fit→CONTROL
- on the **FitControl widget** select the Lambda, Tolerance, and Convergence values
- FitInterferometry widget (the one with Rho and theta): Fit
- Look at the new fit on both OutputBeams. If the fit is still bad, try re-calibrating the data.
- Now you may close OYSTER-type 'exit' into the terminal
- Get your final data from the **fitnights.psn** file.

You can check if your star is a binary by looking at the **UV-Radius.**