HTCC Mirror and Winston Cone Testing, Spring 2013

From HallBWiki

Contents

Documentation

Draft CLAS-note (https://userweb.jlab.org/~puckett/HTCC_reflectance_tests/CLASnote /HTCC_mirror_test_CLASnote.pdf) on the HTCC mirror test apparatus and initial results.

Test procedures

Documentation of procedure steps will be here:

Handling of optics/mirrors/WCs:

When handling sensitive optics such as lenses or mirrors, gloves should be worn at all times, as human oils will absorb UV light, and otherwise interfere with readings taken. Care should be taken not to scratch or drop optical parts, and when not in use, they should be returned to their proper container. Even when wearing gloves, try to avoid directly touching the functional surfaces of mirrors and lenses.

To mount the Winston cones:

There is a custom-built mount for the Winston cones. To mount, put the Winston cone through the large hole so that the body of the cone is held above the base of the mount, then attach it with screws and washers through the four holes. If the monochromator is elevated, the Winston cone mount will also need to be elevated, placed on an elevated breadboard. Line up the legs of the elevated breadboard so that its axes are parallel to the those of the optical bench. Given the design of the HTCC, the Winston cone itself should be aligned parallel to the beam of the monochromator, or about 18 degrees from the y-axis of the optical bench. Once the Winston cone is aligned, screw it down to the optical bench (or elevated breadboard).

To mount the mirrors:

There is a custom-built vacuum mount for the HTCC mirror segments. Each type of segment has its own particular shape, and there are accompanying drawings for each noting the position of pins on the vacuum mount. Insert these pins in the correct locations, then carefully place the mount so that it rests on the bottom pins. It should stay there while you turn on the vacuum, but to be absolutely sure, it is best to have a partner turn on the pump while you hold the mirror in place. Once the pump is on, the mirror can be swiveled into any desired alignment.

Alignment of detectors/beam splitter/lamps:

Before aligning, make sure that the shutter to the monochromator lamps is closed. Then remove the lid to the monochromator itself and place the plexiglass plate with the laser mounted on it on top of the open monochromator so that the laser rests in the clear area next to the collimator. Turn the laser on, and adjust the plate until the laser points directly through the collimator and into the black box. When performing multiple measurements, this is most easily done by shining the laser back on to the center of the target of the control photodiode, which should not be moved between tests (for the first alignment, read below). With the laser in place, move the experimental photodiode on its rail to intercept the beam after its reflection off the mirror or Winston cone. Place a reflective surface on top of the frost glass window (if not using the lens tube, the custom-made alignment cover is used), and adjust the angle of the experimental photodiode until the laser reflects back on itself onto the collimator. Adjust for height, realign for angle if necessary, then translate the photodiode on the rail until the beam enters the pinhole of the frosted glass window (or rests on the center of the crosshairs of the alignment cover). Then turn off the laser, replace the monochromator cover, remove the frosted glass windows (or alignment covers), and close the black box. You are ready for testing.

Personnel/equipment safety:

DO NOT shine the alignment laser directly or indirectly into someone's eye. DO NOT open the monochromator lid unless the shutter is closed or the deuterium lamp is off. DO NOT lift the black box unless the locking ring for the collimator is removed. If you are done with a piece of optical equipment, such as a mirror, lens or beamsplitter, put it back into its proper container. When you are done with testing for the day, turn off all equipment, especially the monochromator lamps.

Beamsplitter calibration and first alignment:

You may find this easiest to do with the black box off - if you decide this is the case, remember to remove the locking ring from the collimator before removing the black box.

First, if necessary, carefully replace the beamsplitter with the desired one, making sure the arrow marked on the side points away from the collimator. As described in "Alignment of...", place the laser in the monochromator. Carefully adjust the laser until you're reasonably sure that is centered through the collimator, then move the beamsplitter into place so that the laser goes through as near the center as possible, adjusting where necessary. Then, swivel the vacuum mount 180 degrees so that the gimbal faces the collimator. Mount the control mirror into the gimbal, still wearing gloves. Place the control photodiode at a reasonable guess as to where you think it will need to be, then adjust the gimbal to point at the control photodiode. Then align the control photodiode to the control mirror as was described for the experimental photodiode in "Alignment of...". Once aligned, note the angle reading of the rotation stage for the control photodiode. Use trigonometry to determine the angle necessary to rotate the control photodiode so that its normal vector makes a right angle to the laser (i.e., the normal vector of the collimator), and then rotate the control photodiode through that angle. Now adjust the beamsplitter to point the reflected part of the laser onto the control photodiode. The control photodiode should not change position or rotation until a new calibration is taken. If it does change position, you will need to take a new calibration. To take the calibration, mount the experimental photodiode so as to intercept the path of the laser before it reaches the control mirror, and align it, as detailed in "Alignment of...". Once aligned, remove the laser and the frosted glass windows (or alignment covers), and take a reading. This is your calibration.

Comprehensive test plan:

Each Winston cone and HTCC mirror should be tested fully in visible and ultraviolet light. To this end, each cone/mirror will require two sets of tests - one with the halogen monochromator lamp from 370 to 650 nm, and one with the deuterium monochromator lamp from 200 to 430 nm, incrementing by 10 nm for each reading. Both sets should have results taken from a variety of angles and positions on the surface of the mirror/cone to make sure there is high homogeneous reflectivity.

To take a test, first plug the USB cable from the monochromator into the computer. Then turn on the monochromator, and then begin the scanning program on the computer. Changing up this order will result in the program not working. Before testing, make sure that the blackbox is sealed, and that the covers have been taken off of the photodiodes. You will then need to read from the picoammeters, recording data to the spreadsheet template (found below). As the picoammeters tend to fluctuate at higher readings, three significant digits are sufficient.

Excel template for data collection

This template is currently set for reading from the deuterium lamp - if you are using visible light, you will need to start reading from 370 nm instead of 200 nm.

Excel Template for testing of Winston cones and HTCC mirrors (https://userweb.jlab.org/~puckett/HTCC_reflectance_tests/HTCCMirrorTestingTemplate2013.xlsx)

Reflectance wavelength scan checklist

The following should be recorded for any test:

- Date and time
- Temperature
- Humidity
- Position and angle of control photodiode on rail
- Position of beamsplitter on rail
- On and off times for monochromator lamp (only for first and last measurments of the day, respectively)

Winston Cones:

- Winston cone ID number
- Physical status of Winston cone
- Position of experimental photodiode rail in blackbox
- Position of experimental photodiode on rail
- Approximate position of beam spot on Winston cone (near collimator, in middle, etc)

HTCC mirrors:

- HTCC mirror ID number
- Physical status of HTCC mirror
- Position of experimental photodiode on rail
- Position of HTCC mirror vacuum mount in blackbox
- Approximate position of beam spot on HTCC mirror

Flat mirrors:

- Flat mirror ID number
- Physical status of flat mirror
- Position and angle of gimbal in blackbox
- Position of experimental photodiode on rail

Geometry notes

Coordinate system

The coordinate system of the optical table is defined as a right-handed Cartesian coordinate system in which the origin is at beam height (roughly 11.4 inches above the breadboard surface) above the "bottom right" mounting hole, as viewed from behind the monochromator (see image in the "simulation" section below). The +X axis points from right to left along the short side of the table, the +Y axis points vertically upward, and the +Z axis points along the long side of the table. The optical table has 48 columns and 72 rows of threaded holes on a 1" $\times$ 1" grid. The first row and column define (X, Z) = (0,0). The 48th column corresponds to X = 47 inches. Similarly, the 72nd row corresponds to Z = 71 inches.

Monochromator geometry

The relevant quantities for the simulation of each reflectance measurement and the determination of the AOI are the location of the monochromator focal point (exit slit) and the angle of the central axis of the collimator relative to the optical table.

Focal point location From the design geometry of the monochromator and the baseplate on which it is mounted, and the locations of the mounting holes used to secure the monochromator to the optical table, the position of the focal point of the monochromator optics, which coincides with the monochromator exit slit, was determined to be at $(X, Z) = (24.70, 14.25)$ (in inches).

Central beam angle

- The nominal beam angle θ_{beam} (relative to the Z axis) calculated from the monochromator design drawings is $\theta_{beam}^{(nominal)} = 18.29^\circ$
- The beam/collimator angle relative to the table was measured using the alignment laser, the flat control mirror and the precision angle measurement hardware to be $\theta_{beam}^{(actual)} = (17.83 \pm 0.34)^\circ$, where the error is the quadrature sum of the uncertainty in the laser-collimator alignment and the resolution of the Vernier scale used for the angle measurement.

UTR160 ("big" rotation platform)

To simulate the reflectance test, the simulation program needs to know the position and orientation of the Newport model UTR160 rotation platform which supports the Newport model 401 XY linear translation stage and the mirror vacuum fixture.

- The position of the center of the UTR160 is determined by the mounting holes used to bolt the UTR160's base plate to the optical table. The base plate has four English-spaced through holes on its sides. These holes form a rectangle that is $2 \times 8 \text{ in}^2$. The UTR160 center coincides with the center of this rectangle. ***This position must be recorded for any test in order to simulate the test and***

calculate its AOI.

- The orientation angle ϕ_0 of the UTR160 angular scale relative to the table was calibrated using the alignment laser and found to be $\phi_0 = 336.33^\circ$. A conservative estimate of the uncertainty in ϕ_0 is roughly 0.5 degrees.
- The angular scale of the UTR160 is oriented such that the angle increases for clock-wise rotations (as viewed from above the table). Therefore, the angle ϕ needed for the simulation is obtained from the measured angle by $\phi_{sim} = \phi_{meas} - \phi_0$.

Data analysis

How to calculate reflectance from mirror scan data + calibration scan data

Assuming that the respective roles of the two photodiodes are the same in both the calibration of the beam splitter and the reflectance test, the reflectivity is calculated from:

$$R = \left(\frac{I_E}{I_C}\right)_{test} \Big/ \left(\frac{I_E}{I_C}\right)_{cal}$$

where I_E and I_C are the measured currents on the experimental and control photodiodes, respectively, and the subscripts "test" and "cal" refer to the currents measured during the reflectance test and the beam splitter calibration, respectively. Under the chosen nomenclature, "control" refers to the photodiode that views the light reflected by the beam splitter and "experimental" refers to the photodiode that views the light transmitted through the beam splitter during the calibration and the light reflected by the mirror being tested during the reflectance measurement.

In general, it is good practice after every measurement to calculate and plot the measured reflectivity curve directly in the Excel spreadsheet, in order to check for errors in data entry. Any typos found can then be corrected by going back to the wavelength where the typo occurred while the test is still aligned. The ROOT macros below are useful for further analysis and generating publication-quality plots for presentation.

ROOT macros

How to analyze reflectance scan data using ROOT

MirrorTestAnalysis_new.C (https://userweb.jlab.org/~puckett/HTCC_reflectance_tests /reflectance_test_analysis_code/src/MirrorTestAnalysis_new.C) This ROOT macro is designed to generate plots and a root file with TGraph objects for later analysis from reflectance scan data copied and pasted from the spreadsheets into text files with configuration info. An example input file is given at:

Tests_062813_UV.txt (https://userweb.jlab.org/~puckett/HTCC_reflectance_tests /reflectance_test_analysis_code/src/config_file_examples/Tests_062813_UV.txt) This configuration file is a typical example.

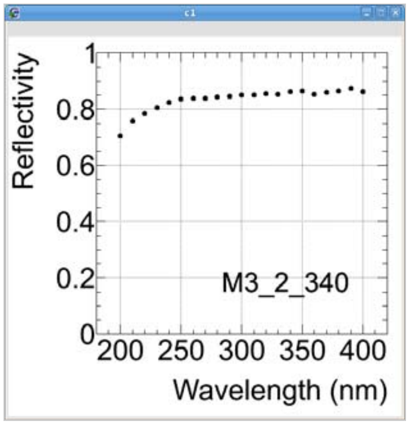
Usage: at a ROOT prompt, do:

root [0] .L MirrorTestAnalysis_new.C+

root [1] MirrorTestAnalysis("in.txt", "out.root");

The first argument is the name of the text file containing the configuration information and data copied from the spreadsheet. The second argument is the name of the output root file to be generated.

If the input file is correctly prepared, the macro generates plots of reflectance vs. wavelength for each individual scan. An example screenshot of one of the plots is shown at right. Instructions for preparing the text file are given below:



Screenshot of plot generated by MirrorTestAnalysis_new.C

- The program reads configuration information from the text file line-by-line until the keyword **"startdata"** is found. Everything after the **"startdata"** keyword is assumed to be reflectance data. Lines starting with "#" are treated as comment lines (i.e., are ignored).
- The reflectance data are typically copied and pasted directly from an Excel spreadsheet, organized in columns according to the Excel template (https://userweb.jlab.org/~puckett/HTCC_reflectance_tests/HTCCMirrorTestingTemplate2013.xlsx) shown above.
  - The data columns are always assumed to be in the order: $\lambda, I_C^{(1)}, I_E^{(1)}, \ldots, I_C^{(n)}, I_E^{(n)}$, with data columns separated by the "tab" character. **Only tab-separated numerical data are accepted.** If the data are copied and pasted directly from the spreadsheet, this format is satisfied automatically. Here λ is the wavelength in nm, and $I_C^{(i)}$ and $I_E^{(i)}$ are the measured photocurrents on the control and experimental photodiodes, respectively, for the i^{th} scan.
  - The program can handle an arbitrary number of wavelength scans.
  - The program assumes that the actual detectors used in the roles of "control" and "experimental" are the same in both the calibration and the reflectance measurement. Otherwise, the detector responsivities do not cancel and the formula given above for the reflectivity does not apply.
- The program requires all of the following configuration information *before* the "startdata" keyword to plot the data correctly. The format for the header information is **"keyword value"**, where keyword and value appear on the same line separated by at least one space character. All keywords and values are case-sensitive.
  - **keyword="nsamples"**: The number of scans corresponding to actual mirror reflectance tests.
  - **keyword="ncalib"**: The number of scans corresponding to beam splitter calibrations. **Note: The values of ncalib and nsamples must match the number of data columns; i.e., the number of columns must equal** $n_{col} = 2 \times (n_{samples} + n_{calib}) + 1$
  - **keyword="TestSampleName"**: A list of *nsamples* unique names (character strings) associated with each test scan, separated by spaces. No plots will be generated if the number of names is less than *nsamples*. Names associated with each scan should consist of useful information indicating the sample identity and/or conditions of the scan. If multiple scans of the same mirror are present in the same spreadsheet, unique names should be given to each scan according to the conditions of the scan, e.g. "Mirror1_AOI30deg_scan1".
  - **keyword="TestSample_iscan"**: A list of *nsamples* scan numbers associated with each test scan, separated by spaces. Valid values for scan numbers are in the range $0 \leq i_{scan} \leq n_{samples} + n_{calib} - 1$. This list of integers maps test sample names onto reflectance test scan numbers.
  - **keyword="TestSample_icalib"**: A list of *nsamples* calibration numbers associated with each test scan, separated by spaces. Valid values for calibration numbers are $0 \leq i_{calib} \leq n_{calib} - 1$. For example, if there is only one calibration scan in the spreadsheet, then all *iscan* values should be zero. If there are multiple calibrations present, *iscan* values should map each test scan onto the applicable calibration scan. Each spreadsheet is

assumed to have at least one calibration scan.

- **keyword="BScalib_iscan"**: A list of *ncalib* scan numbers associated with each calibration scan, separated by spaces. Valid values for scan numbers are in the range $0 \leq i_{scan} \leq n_{samples} + n_{calib} - 1$. This list maps calibration numbers onto scan numbers.

The program was designed to be flexible enough to handle almost any arrangement of test scans and calibration scans in a single spreadsheet. The ROOT plots and files containing TGraphs for subsequent analysis can be generated with minimal effort by copying and pasting the data columns from the spreadsheet without rearrangement. The only additional step required is to supply the configuration information telling the program how to interpret the data. The only restrictions on the arrangement of data columns are

- The first column is always wavelength in nm, and this column appears exactly once.
- The remaining columns must always be pairs of current measurements from the control and experimental photodiodes, in that order.
- At least one calibration scan is required for the wavelength range in question.

How to combine data from different scans

CombineMirrorTests.C (https://userweb.jlab.org/~puckett/HTCC_reflectance_tests/reflectance_test_analysis_code/src/CombineMirrorTests.C) ROOT macro used to combine data from different scans for statistical analysis.

setup_all_combined.txt (https://userweb.jlab.org/~puckett/HTCC_reflectance_tests/reflectance_test_analysis_code/src/config_file_examples/setup_all_combined.txt) Example input file to CombineMirrorTests.C.

Usage: At a ROOT prompt, do:

root [0] .L CombineMirrorTests.C+

root [1] CombineMirrorTests("in.txt","out.root");

where the first argument is the name of a text file listing all the scans to combine, and the second argument is the output ROOT file to be generated.

The purpose of the program is to organize and aggregate data from many different scans, often including multiple tests of the same mirror.

Input File:

The program parses the entire input text file, line-by-line. There are two valid types of entries in the text file. Comment lines start with the "#" character.

- Lines starting with keyword **"lambda"** define the wavelength bins in which data will be combined and averaged. The format is "lambda min max step" (fields separated by spaces). This creates wavelength bins of width "step" from "min - step/2" to "max + step/2". The number of bins is "(max - min)/step + 1". The purpose of this line is to combine and average all measurements of the reflectance of a given mirror in a given wavelength bin.
- All other lines refer to individual mirror tests. Each line consists of six fields separated by spaces. The required fields are:

1. **Mirror name:** A character string with a name identifying which mirror was tested. All scans with identical mirror names (i.e., all lines for which this field is the same) will be combined.

2. **Mirror scan number:** An integer representing each individual scan. This field has no effect on the grouping and averaging of scans, but it should be unique for each scan, as a ROOT tree will also be created, and when plotting results from the root tree, this variable is used to distinguish different scans from the same mirror.
3. **ROOT file name:** The name of the ROOT file containing the TGraph of reflectance vs wavelength for this scan. The program will read the graph of this scan from the ROOT file. The ROOT file should already exist (usually, this refers to one of the ROOT files created by the plotting macro MirrorTestAnalysis_new.C). If the file resides in a different directory from the working directory, either the relative path to the file from the working directory or the full pathname of the file must be given.
4. **TGraph object name:** The name of the TGraph object containing the data (reflectance vs. wavelength) resulting from the scan. The TGraph object is assumed to reside within the corresponding ROOT file.
5. **"Timestamp":** Format is **yyyymmddhhmm** (year, month, date, hours (24-hr format), minutes). This provides the date and time (1 minute resolution) of the scan. In the ROOT tree, the timestamp is stored in units of days since August 15, 2012 (approximate date of the first reflectance tests using this apparatus).
6. **"AOI":** Angle of incidence of the scan (if unknown, just use "0"). The program does not care whether this is given in degrees or radians, but this should be given in degrees for readability and consistency. The program used to calculate the AOI for a given scan is described below.

Output file:

The output of the program is a ROOT file containing several different graphs representing the combined reflectance data for each unique mirror name, and a ROOT tree with "events" representing each individual reflectance measurement. The graphs summarize the combined statistical information about each mirror, while the tree can be used to plot correlations of the data with AOI, time, etc.

The contents of the file can be viewed within root by loading the file as a TFile object; e.g.,

root [0] TFile F("out.root");

root [1] F.ls();

For each unique mirror with name "MirrorName" found in the input file, three graphs will be generated:

1. **"MirrorName_rms":** TGraphErrors with data points equal to the simple average of all reflectance measurements in each wavelength bin, and "error bars" equal to the rms deviation from the mean of all measurements in that bin.
2. **"MirrorName_range":** TGraphAsymmErrors with data points equal to the simple average of all reflectance measurements in each wavelength bin and asymmetric error bars with high (low) error bar corresponding to the maximum (minimum) reflectance result in that bin.
3. **"MirrorName_stderr":** TGraphErrors with data points equal to the simple average of all reflectance measurements in each wavelength bin and error bars equal to the rms deviation from the mean of all measurements in that bin divided by $\sqrt{N}$, where N is the number of data points in that bin.

In addition, a ROOT tree will be written which contains details of each individual reflectance measurement as "events". The root tree is named "T" and the variables in the tree (case-sensitive) are:

1. **lambda:** wavelength in nm
2. **R:** Reflectivity
3. **AOI:** Angle of incidence in whatever units were supplied in the config file (we use degrees for consistency). If insufficient information was recorded to calculate the AOI for a given test, "zero" is

typically used.
4. **time:** Time in days since midnight on August 15, 2012
5. **ratio:** Ratio of reflectance for this measurement to the average of all reflectance measurements on the same mirror in the same wavelength bin.
6. **Scan:** Scan ID, corresponds to the mirror scan number given in the second column of the line defining a scan in the config file. The scan number for each scan should be uniquely defined for that mirror in order to distinguish between different scans.
7. **Mirror:** Mirror ID number. ID numbers start at 1 and are assigned according to mirror names *in alphabetical order*.

Since visible and UV measurements are typically performed as separate scans, this program is especially useful for combining UV and visible data!

How to simulate the reflectance test and calculate collection efficiency and AOI using ROOT

The figure at right shows the ray-tracing simulation of the reflectance tests. The origin of coordinates is at beam height above the "bottom right" hole of the optical table. The Y axis is vertically up, the Z axis coincides with the right-most column of holes along the long edge of the table, and the X axis coincides with the first row of holes along the shorter edge of the table. The origin of the beam is at the monochromator focal point, the location of which relative to the mounted position of the monochromator baseplate was determined from the monochromator design drawings and hand measurements of the monochromator location on its baseplate.

Rays are generated with a uniform transverse spatial profile relative to the collimator axis at the beam origin, a uniform azimuthal angular distribution, and a uniform distribution in $\cos\theta$ for $0 \leq \theta \leq 1^\circ$, where θ is the polar angle relative to the collimator axis. The rays are projected to both apertures of the collimator (blue discs), and rays whose radial displacement exceed the hole radius at either aperture are rejected. Rays passing through both apertures are then projected to the mirror surface, where the angles of incidence and the reflected ray are calculated. Finally, the reflected rays are projected to the surface of the lens, traced through the lens using Snell's law and the known properties of the lens, and projected to the photodiode surface. Only the first 100 rays are drawn, but an arbitary number of rays may be generated for the analysis of collection efficiency and other optical features of any given measurement.

How to get the code:

reflectance_test_simulation.C (https://userweb.jlab.org/~puckett/HTCC_reflectance_tests /reflectance_test_analysis_code/src/reflectance_test_simulation.C) ROOT program to carry out ray-tracing Monte Carlo simulation of HTCC mirror reflectance tests.

Usage:

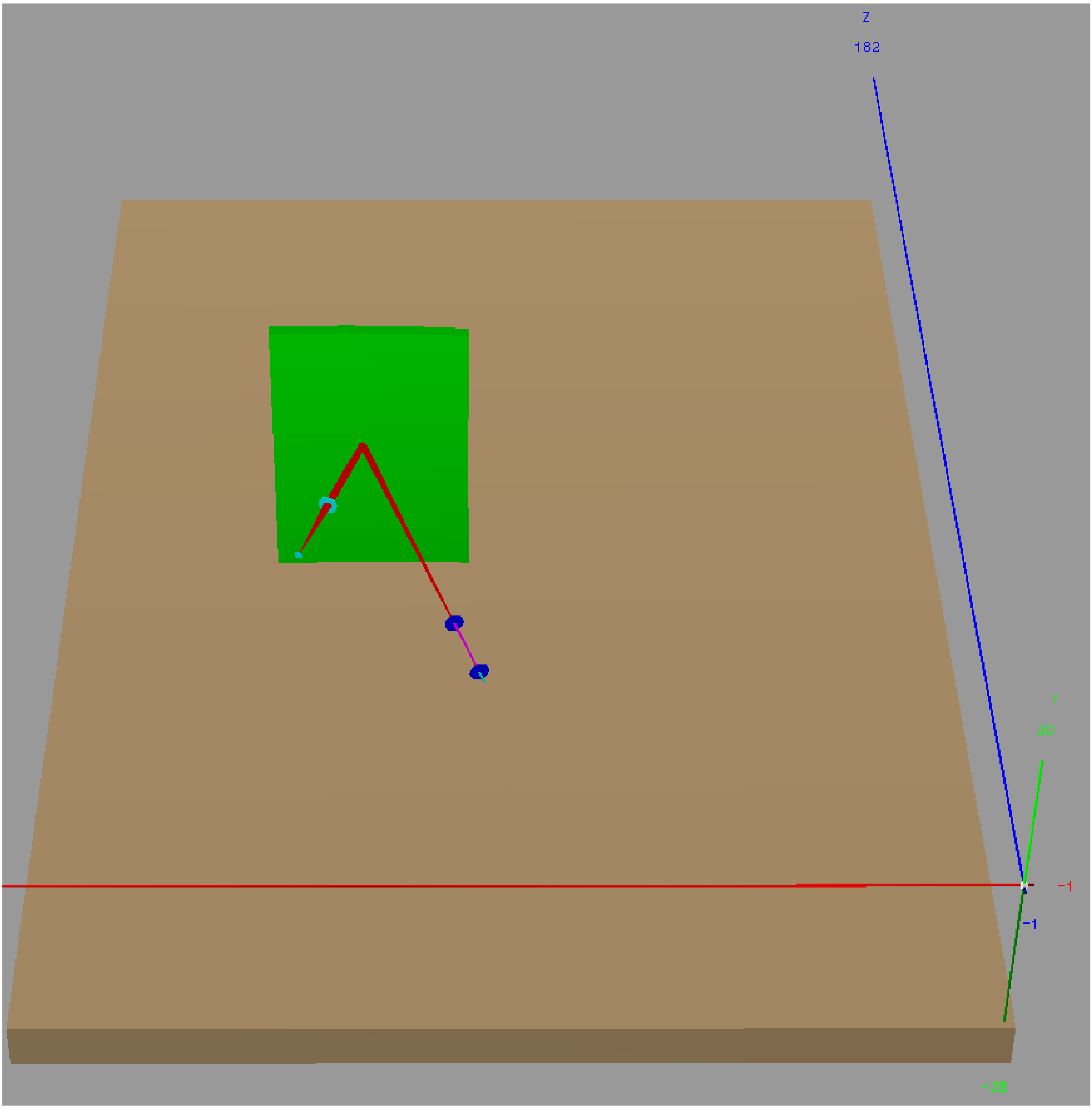
At a ROOT prompt, do e.g.:

root [0] .L reflectance_test_simulation.C+

root [1] reflectance_test_simulation(imirror=1, "in.txt", "out.root", drawflag_nolens=0);

The main function "reflectance_test_simulation" expects four arguments:

1. **"imirror":** Mirror facet type (integer). Valid values are:
    1. **imirror=0:** Mirror 1 left



:ctance test simulation example

2. **imirror=1:** Mirror 1 right
3. **imirror=2:** Mirror 2
4. **imirror=3:** Mirror 3
5. **imirror=4:** Mirror 4

2. **"setupfilename":** Name of configuration text file (character string). An example configuration file can be found here (https://userweb.jlab.org/~puckett/HTCC_reflectance_tests/reflectance_test_analysis_code/src/config_file_examples/setup_simulation_example.txt) .
3. **"outputfilename":** Name of output ROOT file (character string). The output root file contains a tree with relevant data for each generated ray:
    1. **xgen:** local x (horizontal) coordinate of initial position of ray relative to the beam origin in the plane perpendicular to the central collimator/beam axis.
    2. **ygen:** local y (vertical) coordinate of initial position of ray relative to the beam origin in the plane perpendicular to the central collimator/beam axis.

3. **thetagen:** polar angle of generated ray relative to central collimator/beam axis (radians).
4. **phigen:** azimuthal angle of generated ray relative to central collimator/beam axis (radians).
5. **xdet:** local x (horizontal) coordinate (cm) of reflected ray position at detector surface relative to nominal detector center position, *without the focusing lens*.
6. **ydet:** local y (vertical) coordinate (cm) of reflected ray position at detector surface relative to nominal detector center position, *without the focusing lens*.
7. **thetadet:** polar angle (radians) of ray incident on detector surface, relative to detector surface normal.
8. **phidet:** azimuthal angle (radians) of ray incident on detector surface, relative to surface normal.
9. **xdet_lens:** same as "xdet", but *with focusing lens*
10. **ydet_lens:** same as "ydet", but *with focusing lens*
11. **thetadet_lens:** same as "thetadet", but *with focusing lens*
12. **phidet_lens:** same as "phidet", but *with focusing lens*
13. **AOI:** angle of incidence of the reflectance test (in radians). This is the AOI of the rays incident on the mirror, relative to the mirror surface normal.
14. **hit_detector:** Equals 0(1) if the reflected ray misses (hits) the detector surface (*without focusing lens*)
15. **hit_detector_lens:** Same as "hit_detector", but *with focusing lens*.

4. **"drawflag_nolens":** If zero, the lens is drawn and the first 1000 drawn rays are traced through the lens. If non-zero, the lens is not drawn, and the reflected rays are projected directly to the detector surface at its nominal position. This flag only affects the drawing/visualization of the setup. In either case, all generated rays written to the ROOT tree are traced to the detector surface for both cases (with and without focusing lens).

How to configure the code:

The program expects to read all necessary configuration information from the text file given as the second argument to the main function "reflectance_test_simulation(...)". All angles are assumed to be given in degrees, and all dimensions are assumed to be given in cm. The configuration information consists of numerical parameters given in *exactly* the following order. Parameters can be given on separate lines or on the same line, separated by spaces:

- ***Group 1: Number of rays to generate and angle of collimator axis:***

1. **ngen (integer):** Number of rays to simulate. The program stops when **ngen** rays have passed through both apertures of the collimator. The actual number of generated rays will be much larger, since only a small fraction of generated rays pass through both holes of the collimator.
2. **thetaZ_beam (double):** The angle of the beam/collimator axis relative to the Z axis of the simulation coordinate system, which corresponds to the long side of the table. ***See "geometry notes" above for actual and nominal values of this angle, which does not change from test to test.***

- ***Group 2: collimator geometry:*** The next group of parameters describe the collimator geometry. The nominal values of these parameters correspond to the current experimental setup, and they do not change from test to test. They could be varied in the simulation to test alternate collimator geometries for future improvement of the measurements.

1. **z0_collimator (double):** This is the spacing between the beam origin (monochromator focal point) and the first aperture of the collimator. Its nominal value is 2.0 cm.
2. **Length_collimator (double):** This is the spacing between the first and second apertures of the collimator. Its nominal value is 10.0 cm.
3. **d1_collimator (double):** This is the diameter of the first collimator aperture. Its nominal value is 0.1 cm.

4. **d2_collimator (double):** This is the diameter of the second collimator aperture. Its nominal value is 0.1 cm.

- ***Group 3: Monte-Carlo generation limits***. These parameters should not change from test to test. The main guideline for choosing these parameters is that the limits should be tight enough that the simulation is efficient, but loose enough to populate the full acceptance of the collimator.

1. **beam_initial_diameter:** Initial diameter of the beam for generation of rays. Should be at least 1.6 mm to populate the full geometric acceptance of the collimator. The radial displacement of the initial ray from the beam origin in the plane perpendicular to the collimator axis is generated such that the initial beam has a uniform transverse spatial profile relative to the collimator axis within this diameter.
2. **beam_initial_divergence:** Maximum polar angle of generated rays relative to the collimator axis. Should be at least 0.6 degrees to populate the full collimator acceptance. The rays are generated with a uniform distribution in cos(θ) for $\cos(\theta_{max}) \leq \cos\theta \leq 1$.

- ***Group 4: Coordinates and rotation angle of the UTR160 platform.*** These parameters are changing from test to test and should be recorded for each test! See "geometry notes" and the screenshot from the simulation above for the definition of the coordinate system and the orientation angle.

1. **X_platform (double):** X coordinate of the center of the UTR160 platform in cm.
2. **Z_platform (double):** Z coordinate of the center of the UTR160 platform in cm.
3. **phi_platform (double):** Angle of the UTR160 measured from the +X axis toward the +Z axis. In the usual configuration, this angle is obtained from the "raw" UTR160 angle reading by subtracting 336.33 deg.

- ***Group 5: Orientation and position of the rail carrying the experimental photodiode*** These parameters describe the position and orientation of the rail and should be recorded after any and every time the rail is moved to a different position on the table. In the real measurement, the experimental photodiode is aligned with the reflected beam by translating along the rail direction and rotating until the beam goes through the center holes of both alignment discs. In the simulation, the position of the experimental photodiode is calculated as the intersection of the central reflected ray with the plane parallel to the rail direction. The position of the experimental photodiode should be recorded after every test so it can be compared to the position calculated by the simulation program.

1. **railflag (integer):** Integer specifying whether the experimental photodiode, which is mounted on a rail carriage, moves along X at fixed Z (railflag=0, the usual case where the rail is mounted on a fixed row) or whether it moves along Z at fixed X (railflag!=0, rail mounted on a fixed column).
2. **Rail coordinate (double):** If railflag is zero, this parameter is interpreted as the Z position (row) of the rail carrying the experimental photodiode. Otherwise, it is interpreted as the X position (column) of the rail.

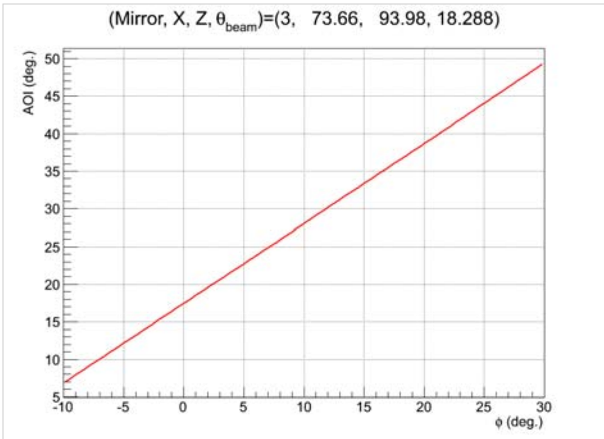
- ***Group 6: Geometry of the focusing lens*** These parameters do not change from test to test and should be left at their nominal values. They can be varied to test the focusing properties of different lenses.

1. **n_lens:** Index of refraction of the lens. The nominal value is 1.460.
2. **thicklens_center:** Thickness of the lens at its center. The nominal value is 0.38 cm.
3. **thicklens_edge:** Thickness of the lens at its edge. The nominal value is 0.2 cm.
4. **Radius_of_curvature_lens:** Radius of curvature of the front (spherical) lens surface. The nominal value is 4.6 cm.
5. **diameter_lens:** Diameter of the lens. Nominal value is 2.54 cm.
6. **clear_aperture_lens:** Diameter of the clear aperture of the lens mounting hardware.

- ***Group 7: Position of lens and detector relative to nominal detector position***. These parameters do not change frequently, but any time the lens-detector spacing is changed by threading or unthreading the lens mount, or any time the position of the lens tube relative to the clamping ring is changed, these parameters should be updated.

1. **zoffset_detector:** This is the distance along the lens tube axis from the center of the post on which the lens tube is mounted to the photodiode surface. This distance can be obtained from the measured distance between the back edge of the clamping ring and the front edge of the nylon spacer by adding 25.325 mm.
2. **lens_detector_distance:** This is the spacing between the front edge of the lens at its center and the photodiode surface. This distance can be obtained by adding 100.275 mm to the length of exposed thread of the SM1V10 in which the lens is mounted.

How to determine the orientation angle of the UTR160 rotation platform corresponding to a desired AOI



Screenshot of the plot generated by AOI_vs_phi.C

AOI_vs_phi.C (https://userweb.jlab.org/~puckett/HTCC_reflectance_tests/reflectance_test_analysis_code/src/AOI_vs_phi.C) Utility macro to determine the AOI for a range of angles of the UTR160. No configuration file required.

Dependency: reflectance_test_simulation_C.so, the shared library generated from reflectance_test_simulation.C (see above). If the library does not exist in the current directory, do ".L reflectance_test_simulation.C+" at a ROOT prompt (assuming that the source file reflectance_test_simulation.C exists in the current directory).

Usage: At a ROOT prompt, type ".L AOI_vs_phi.C+". Then, invoke the main function as follows:

root [1] AOI_vs_phi(imirror, xplatform, zplatform, phimin, phimax, thetabeam);

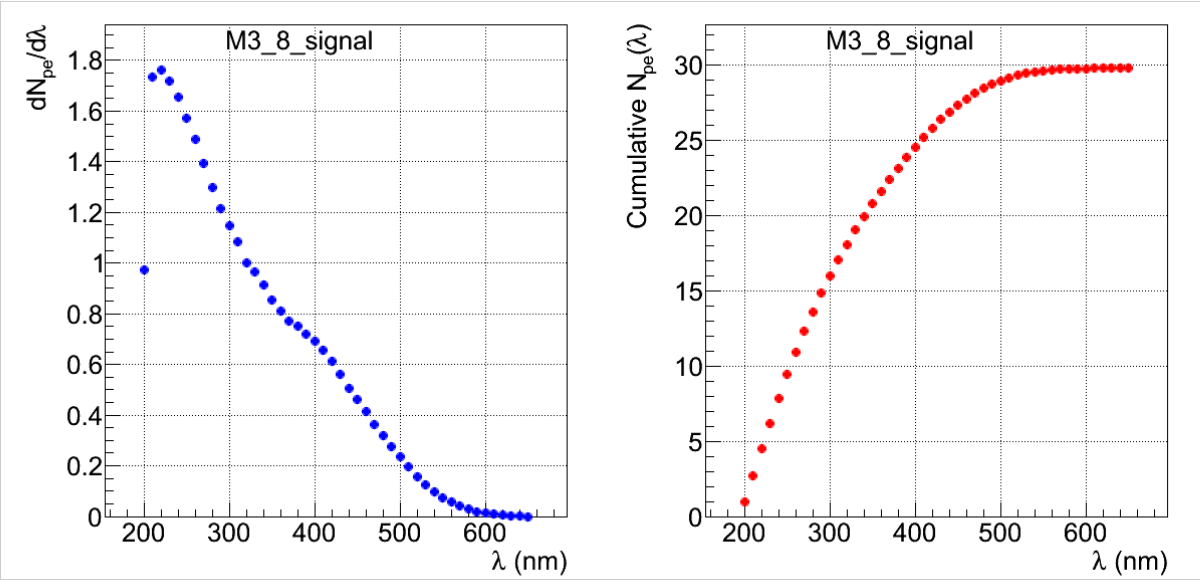
The required arguments are:

1. **imirror (int):** Mirror number. Numbering convention is the same as above. 0 (1) = Mirror 1 left (right), 2-4 = Mirror 2-4, respectively.
2. **xplatform (double):** X coordinate of the center of the UTR160 rotation platform in cm.
3. **zplatform (double):** Z coordinate of the center of the UTR160 rotation platform in cm.
4. **phimin (double):** Minimum phi angle in degrees to calculate the AOI.
5. **phimax (double):** Maximum phi angle in degrees to calculate the AOI.
6. **thetabeam (double):** Angle of the central beam/collimator axis relative to the table in degrees.

The program will then pop-up a plot of the AOI vs the "phi" angle of the UTR160 platform within the range of phi specified when the function was called, and you will be prompted to enter a desired AOI in degrees. After you supply your desired AOI, the root prompt will calculate the phi angle corresponding to that AOI and print it out.

In the current configuration of the UTR160, the UTR160 angle reading corresponding to the value of ϕ calculated by this program is $\phi_{UTR160} = \phi + 336.33^\circ$ (see geometry notes above).

How to estimate the average signal strength from reflectance data



Screenshot of plots representing differential and cumulative average signal strength from a given reflectance curve.

calc_signal_strength.C (https://userweb.jlab.org/~puckett/HTCC_reflectance_tests /reflectance_test_analysis_code/src/calc_signal_strength.C) ROOT macro to estimate the signal strength from a given set of reflectance graphs contained in a single ROOT file.

det_CO2.dat (https://userweb.jlab.org/~puckett/HTCC_reflectance_tests/reflectance_test_analysis_code /src/det_CO2.dat) Text file with data on PMT quantum efficiency and CO2 transparency. This file is required by the ROOT macro and must exist in the working directory.

Example configuration file. (https://userweb.jlab.org/~puckett/HTCC_reflectance_tests /reflectance_test_analysis_code/src/config_file_examples/setup_signal_strength.txt) The graphs must exist in the input root file or the program will not work.

Usage: At a ROOT prompt, do ".L calc_signal_strength.C+". Then call the main function "calc_signal_strength" as follows:

root [1] calc_signal_strength("in.root", "setup.txt", "out.root");

where "in.root" is the name of a ROOT file containing one or more TGraphs with reflectance data, "setup.txt" is a text file containing configuration information, and "out.root" is the name of an output root file to be generated.

The "setup.txt" file contains configuration info as "keyword value" pairs on separate lines. An example configuration file is available here (https://userweb.jlab.org/~puckett/HTCC_reflectance_tests /reflectance_test_analysis_code/src/config_file_examples/setup_signal_strength.txt) . Lines starting with "#" are treated as comments. The following types of lines are recognized:

1. keyword **lambda**: followed by values min, max and step (in nm) separated by spaces, this line defines the wavelength binning for the presentation of the differential and cumulative signal strength as a

function of wavelength. Only the "step" parameter is actually used for the calculation of the number of Cherenkov photons emitted in a small wavelength interval.

2. keyword **mirror**: followed by a number from 1-4, defines which mirror facet is being used. The mirror number is used to calculate the approximate average path length in CO2 traversed by emitted Cherenkov photons.
3. keyword **path_length**: followed by value representing the path length in meters traversed by the scattered electron in the gas before the mirror (see table below).
4. keyword **graph**: followed by character strings "ingraphname" and "outgraphname" representing, respectively, the name of the TGraph object containing the reflectance data and the prefix for the name of the output graphs to be created. For each input reflectance graph, two output graphs are created (see screenshot at right). The first graph shows the differential distribution of the expected average number of photoelectrons as a function of wavelength, where each data point represents the average number of photoelectrons in the wavelength interval represented by the "step" parameter above (typically 10 nm). The second graph shows the cumulative number of photoelectrons as a function of wavelength, and is the integral of the first graph from the minimum wavelength (typically 200 nm) to the wavelength in question.

The program will estimate the number of photoelectrons and produce graphs for each graph found in the input root file. There is no limit on the number of graphs. The lines with keyword "lambda", "mirror" and "path_length" are optional (sensible default values are defined for all the parameters) and may be used as many times as desired. Each new parameter definition supersedes the previous definition, and the most recent definition will be applied to all subsequent graphs until a new definition is found.

Table of HTCC mirror angles of incidence for rays originating from the origin of CLAS12

The path length in the gas is defined as the length of the ray from the origin to the mirror half-sector center line for a given scattering angle θ, minus the distance from the origin to the HTCC entry window at the same θ, which is given by 38 cm/cos(θ). This path length can be used to calculate the average number of Cherenkov photons emitted.

Mirror Facet	Scattering angle θ (deg.)	AOI (along half-sector center line, deg.)	Path length in gas (meters)
1	27.5	25.03	1.48
1	31.25	24.96	1.53
1	35.0	24.77	1.57
2	20.0	29.037	1.37
2	23.75	29.108	1.43
2	27.5	29.042	1.48
3	12.5	32.73	1.24
3	16.25	32.96	1.31
3	20.0	33.06	1.37
4	5.0	36.03	1.10
4	8.75	36.43	1.17
4	12.5	36.70	1.24

Results database

Raw data

Raw data (http://www.jlab.org/Hall-B/secure/e1/markov/HTCC/reflectance_test_results_September_19_2013.tar.gz) as a gzipped tar file. After extracting the archive, cd to the folder "reflectance_test_results" and see "README.txt" for further instructions.

Mirrors

Links to individual mirror pages

Mirror 1 left, number 12 (accepted)

Mirror 1 right, number 12 (rejected--low reflectance)

Mirror 2, number 12 (rejected--low reflectance)

Mirror 3, number 10 (rejected--large stain)

Mirror 4, number 12 (tentatively accepted)

Mirror 3, number 2 (spare)

Mirror 3, number 8 (spare)

Mirror 3, number 10 (spare)

Mirror 2, number 11 (accepted)

Mirror 4, number 11 (accepted)

Mirror 3, number 1(rejected)

Mirror 1 right, number 7 (rejected)

Mirror 1 left, number 2 (accepted)

February 2014

Mirror 1R, 1
Mirror 1R, 2
Mirror 1R, 3

Mirror 1L, 5
Mirror 1L, 7

Mirror 2, 2
Mirror 2, 8

Mirror 3, 2
Mirror 3, 3
Mirror 3, 4

Mirror 4, 2
Mirror 4, 10

March 2014
Mirror 1L, 1
Mirror 1L, 2
Mirror 1L, 3

Mirror 1R, 1
Mirror 1R, 9

Mirror 2, 2 (8-16-12)
Mirror 2, 7
Mirror 2, 10

Mirror 3, 6
Mirror 3, 7

Mirror 4, 3
Mirror 4, 5

Summary table

	1	1(9-20-12)	2	2(10-1-13)	2(8-16-12)	3	4	5	6	7	8	9	10	11	12	Total accepted
1L	X		X	X		X		X		X					X	2
1R	X	X	X			X				X		X			X	0
2			X		X					X	X		X		X	1
3	X		X			X	X		X	X			X			0
4			X			X		X					X	X	X	2

Winston Cones

Link to summary table: pdf file maintained by M. Burke/Youri/Andrew Internal links to individual Winston Cone pages with comprehensive test data/simulations/etc.

Storage

The Winston Cones are now stored in the TED clean room.
The list of the cones:

277
278
279
280
281
282
283
285
286
287
288
289
290
291
292
294
295
296
297
298
299
300
382
383
384
385
387
388
389
390
391
392
394
395
398
399
400
401
402
403
404
405
406
407
408
First Article ECI,
which adds to 46 good WC present in the storage.
From the originally sent cones 6 are absent:
9-284
9-293
9-386
9-393
9-396

9-397.
Four of them are sent to ECI for remanufacturing and will return for testing (386, 393, 396, 397).
Two are already replaced (293 and most likely 284).
As a results we will have 50 WC.

WC storage in the TED High Bay (Shelves with gray boxes on the left).