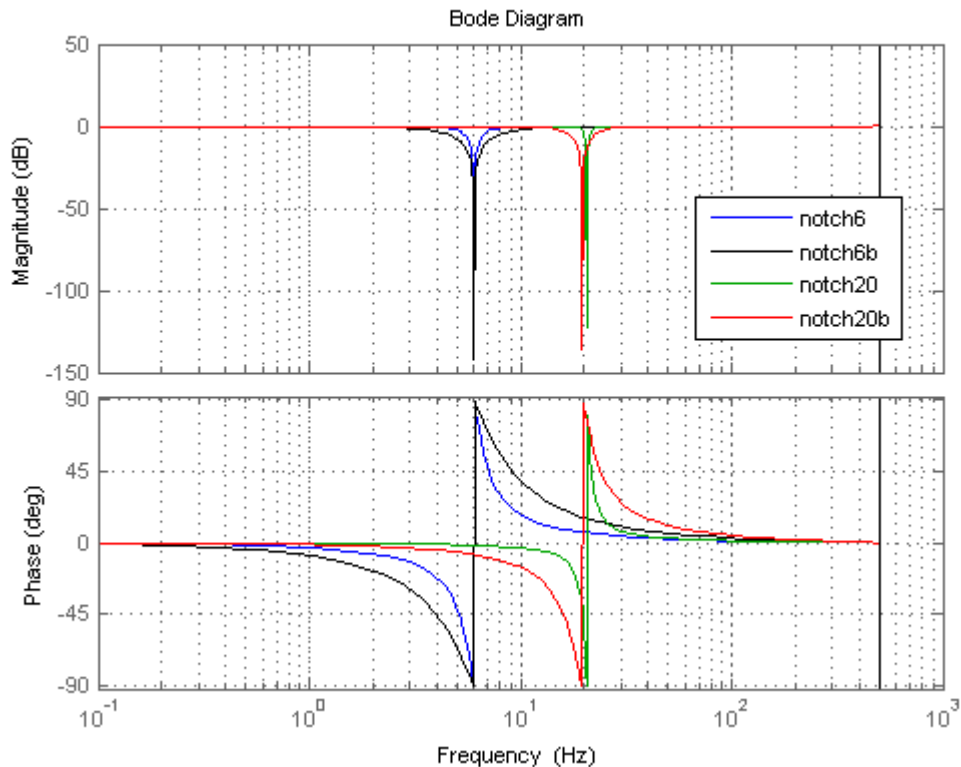


Servo Project Progress Report
Dates covered: 2/16/07 to 3/1/07
D. Clark

At last report, we had planned the following activities:

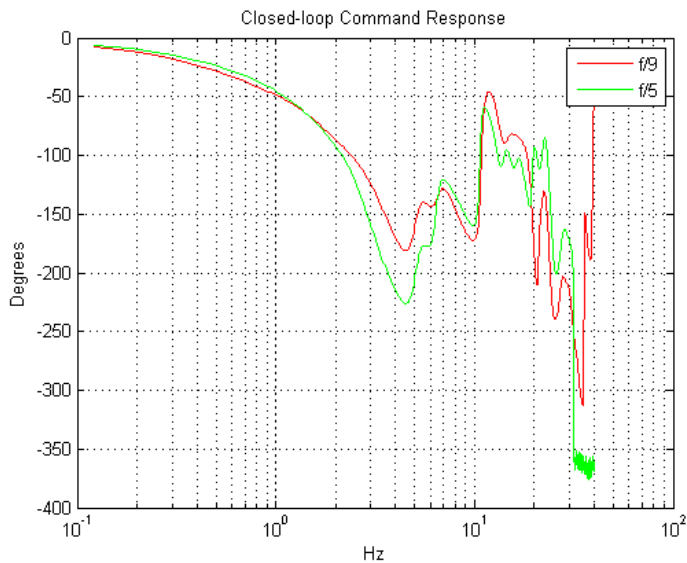
1. Adjust notch filters as necessary to optimize performance.
2. Increase servo loop gains and verify simulation results.
3. Measure $f/5$ configuration response with the same controller setup as used for $f/9$ to characterize changes (if any) to the telescope response.
4. Measure new controller gains and filters on hardware.
5. Prepare for actual deployment of improved controller.

Using the measured data from the open-loop response of the $f/9$ telescope configuration, the notch filters previously designed at 6 and 20Hz were evaluated w.r.t. their center frequencies and Q against the data from the measurements, and a new pair of filters were generated using the Matlab Filter Design Toolbox. Below is a comparison of the filter designs:



The initial filter design was fairly conservative in terms of Q because of the obvious degradation in phase margin on the new filter response plots above to try to preserve overall control loop bandwidth. Once these were designed, testing the controller with the $f/5$ telescope configuration was in order. Earlier results had indicated that the controller and filter bank were robust enough to handle secondary and instrument changes without

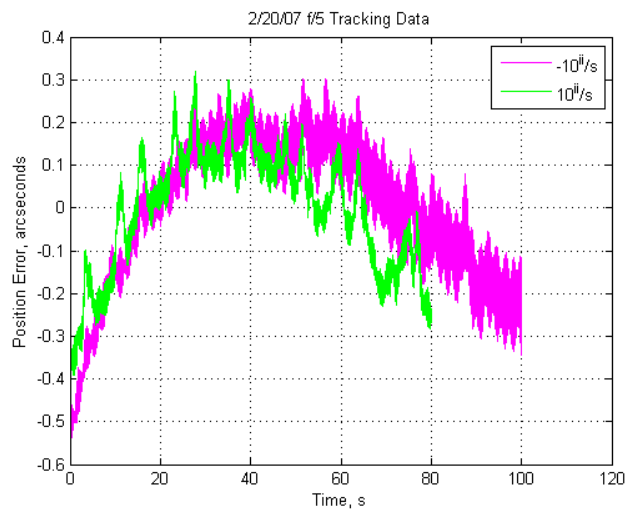
requiring changes to the controller gains or filters. On 2/20/2007, this testing was done. When the controller was built and run as it had been set up for $f/9$, the controller oscillated at the familiar 20Hz frequency. This indicated that the fairly narrow 20Hz notch filter did not completely capture the modal energy, whose center frequency had shifted due to the change in inertia from the telescope configuration change. Inserting the new 20Hz notch filter resulted in a stable controller. This encouraged the installation of the new 6Hz filter, as well, and all testing was subsequently carried out with both these new filters in place. Several slew motions were carried out, as well as the more interesting closed-loop bandwidth tests. When measuring closed-loop bandwidth, a chirp signal is inserted into the controller as a position command in place of the command pre-processor's output, and the output variable is the telescope's position response. This allows a direct measurement of the overall bandwidth of the controller to command signals. Below we have the Bode magnitude and phase measurements of the $f/9$ taken earlier in the month, and the $f/5$ measurements:



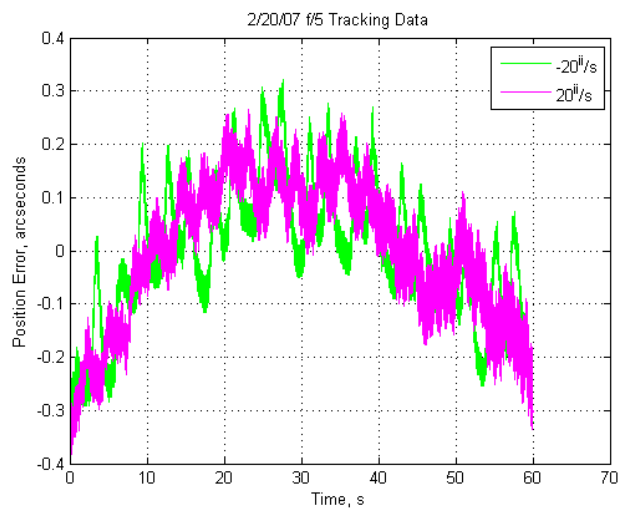
For both closed-loop tests, the controller gains were the same; all that changed was the two notch filters. The effect of the wider notch filters is clear: both the 6 and 20Hz modes are down by ~ 5 to 10db from the earlier measurements. These new filters are yet to be tested with $f/9$, but it is expected that they will work for all telescope configurations.

Additional tracking data were taken on 2/20 with the new controller filters to gain more insight into the tracking data artifacts noted in the last report. For the purposes of this data collection, the same velocities of ± 10 , 20, and 50 arcseconds/second were commanded. A surprising result of this testing is the indication that the low-frequency “hitching” is both velocity and direction-dependent.

Below is the ± 10 arcsecond/second data plot:



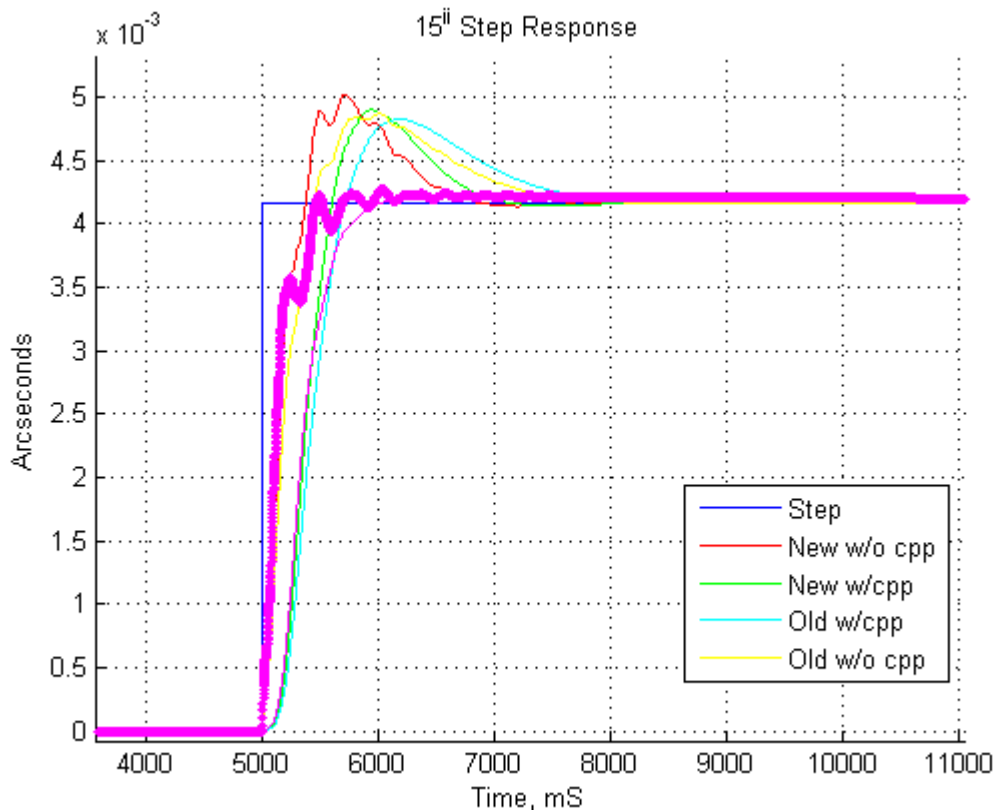
Followed by ± 20 arcseconds/second:



For the next series of tests with $f/5$, the task list grew a bit longer—

- Verify the controller stability with the filters and gains as previously tested.
- Collect more tracking data over a longer period of time to capture data for more intensive reduction to PSDs.
- Test new controller gains for increasing overall closed-loop bandwidth.
- Collect disturbance-rejection data for characterization of potential performance during operation.

During simulation, a couple of new gain parameter sets were tested, both with a without the command pre-processor in series with a 15 arcsecond step input:

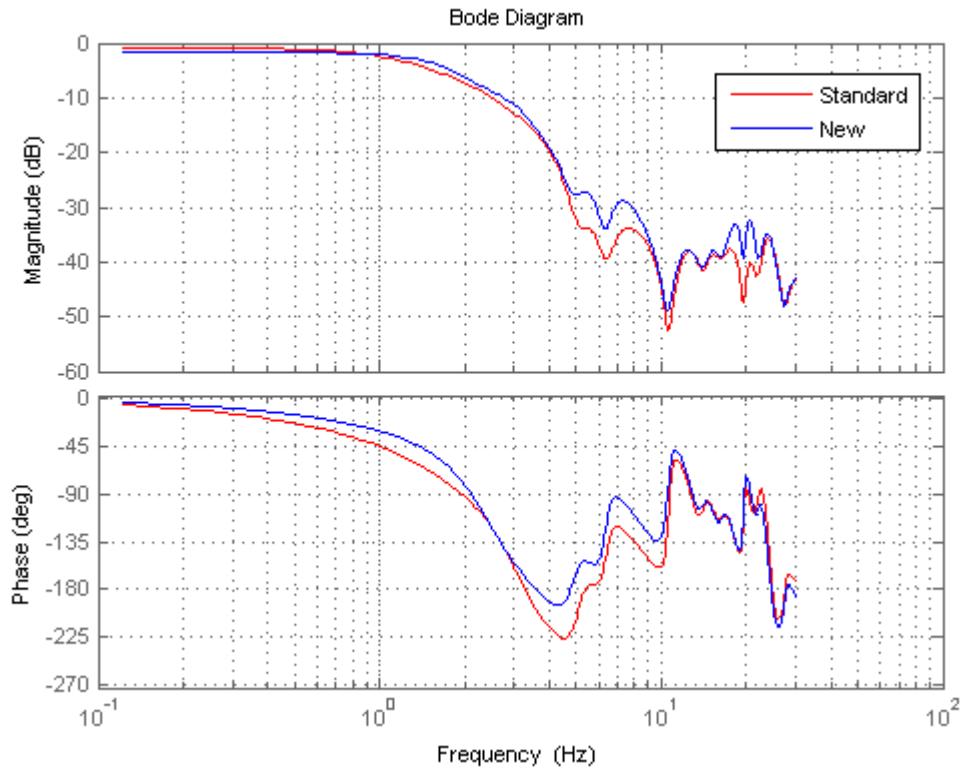


The legend on this plot is a bit out of date: the magenta lines are a different set of gains added after the fact to the plot. For the next round of testing we then had 3 possible gain sets to use, one that is the “standard” set used for all testing up to this point, a “hot” set that was manually tuned for the maximum possible closed-loop bandwidth, and a “medium” set generated with the classic Ziegler-Nichols tuning method for the velocity loop, and Tyreus-Luyben’s method for the position loop. The command pre-processor clearly eliminates “ringing” in the response by smoothing out command inputs to eliminate all discontinuities in the command signal; this is one of the primary reasons to *have* the command pre-processor.

On 2/27/07, the newest set of tests with $f/5$ were performed. The controller was built as previously, several slews were conducted, then tracking data collection, then closed-loop bandwidth and closed-loop disturbance rejection, using both the “standard” controller

gains and the “medium” set. The “hot” set turned out to be unstable, and no further testing was conducted with it.

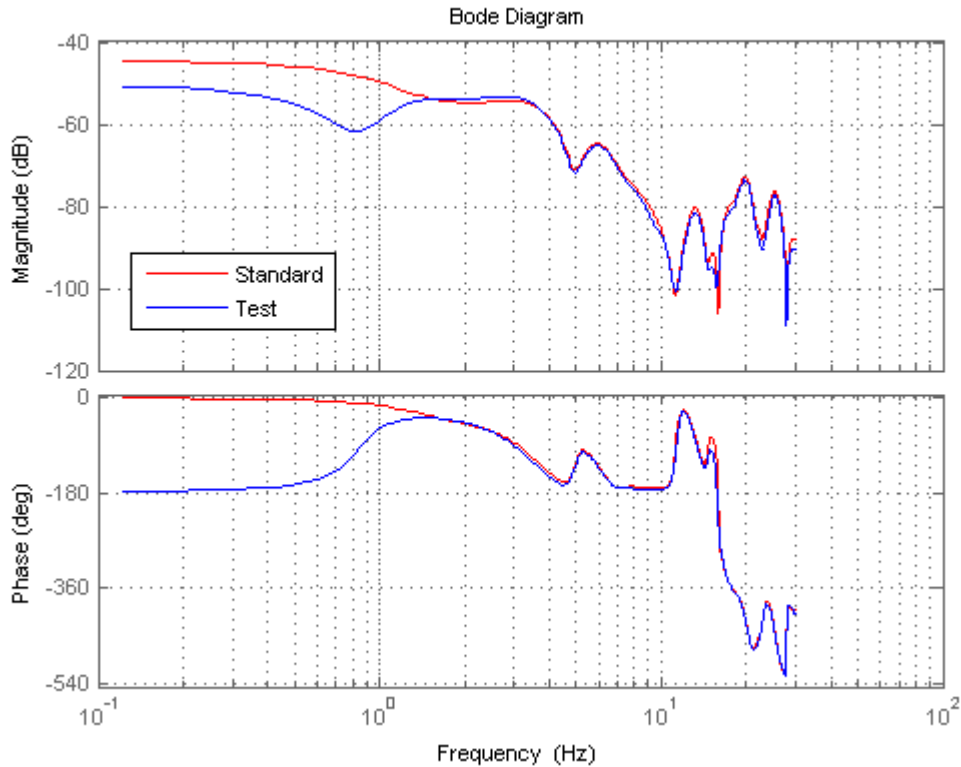
Below is a comparison of the two closed-loop bandwidth tests with the gain parameter changes – the improved notch filters were used:



As expected from simulation, increasing the servo gains increases the bandwidth at the cost of also increasing the servo response to structural modal frequencies. We therefore must be circumspect in balancing bandwidth and modal response in tuning the servo gains.

To explore the servo response to torque disturbances, the same chirp signal source blocks in the Simulink code were moved over to sum with the DAC output, while the command input was held at the latched absolute encoder value at startup. The chirp signal was rescaled to 0.1V peak-to-peak, which corresponds to a 308lbf torque on the telescope. Since the mechanical admittance of the telescope to wind gust inputs is at present unknown (an ancient MathCad model of the wind torque exists), the disturbance torque scaling in this test cannot be directly related to wind loading. A more precise test would scale the torque disturbance signal with both the wind loading admittance and wind-gust load PSD (as is done with the classic aerospace Davenport wind spectrum). The purpose of testing disturbance rejection is to characterize the servo bandwidth in terms of the disturbance signal, and measure the response to input frequencies that are equal to structural resonances.

The following Bode plot shows the response to a volts input to degrees of telescope disturbance motion. Ideally, the rejection would be $-\infty$ at 180° -- obviously the servo is not ideal, but the “test” gains show improvement:

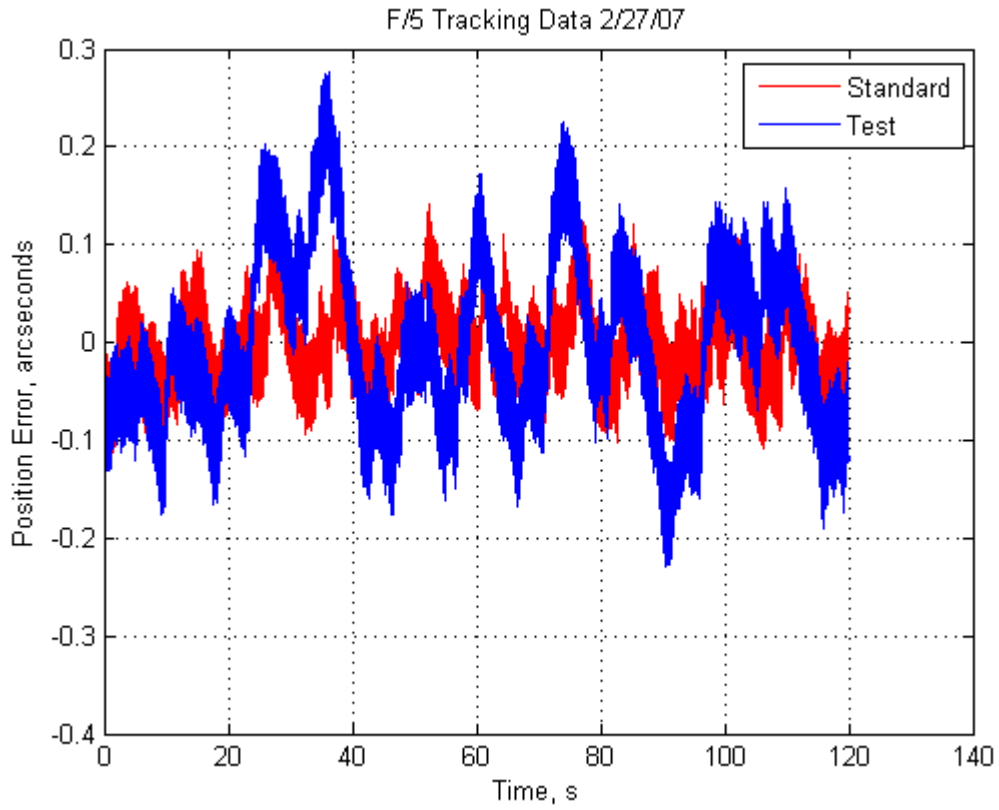


Clearly, most of the controller bandwidth is below about 1.5Hz, with almost no difference in the two servo gain parameters above that. However, the servo *is* rejecting structural modal frequencies, as desired. No good explanation comes to mind for the sign flip in the phase, and will need more detailed examination of the data.

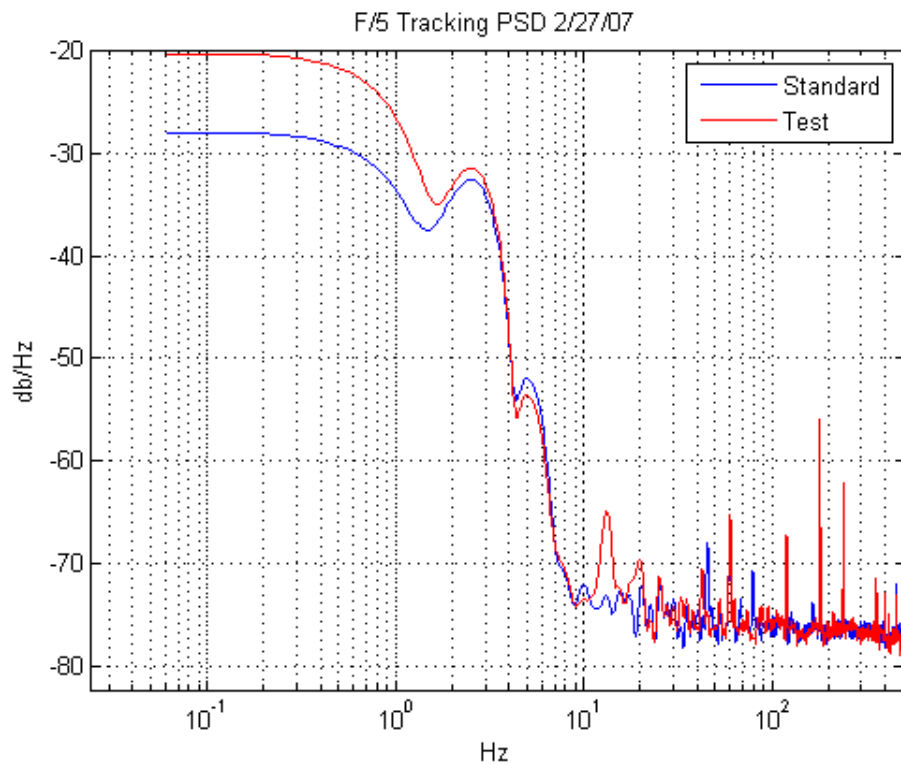
These results are the opposite of the results reported in **MMTO Internal Technical Memorandum #02-2**, “*MMT Mount Control System Operation and Performance*”, to which the interested readers are directed, in that the servo rejection of structural modes is properly working.

This latest round of testing included collection of ~300s of tracking data using both gain sets, at a more reasonable velocity of 5 arcseconds/second. Some massaging of the data was done to detrend the position error signal to more accurately calculate the error signal PSD without having a large DC term swamp out all the (relatively) low frequencies of interest.

First, we show the time series of the tracking with each gain set:



Next, the PSD calculated from these time data using Welch's averaged-periodogram method:

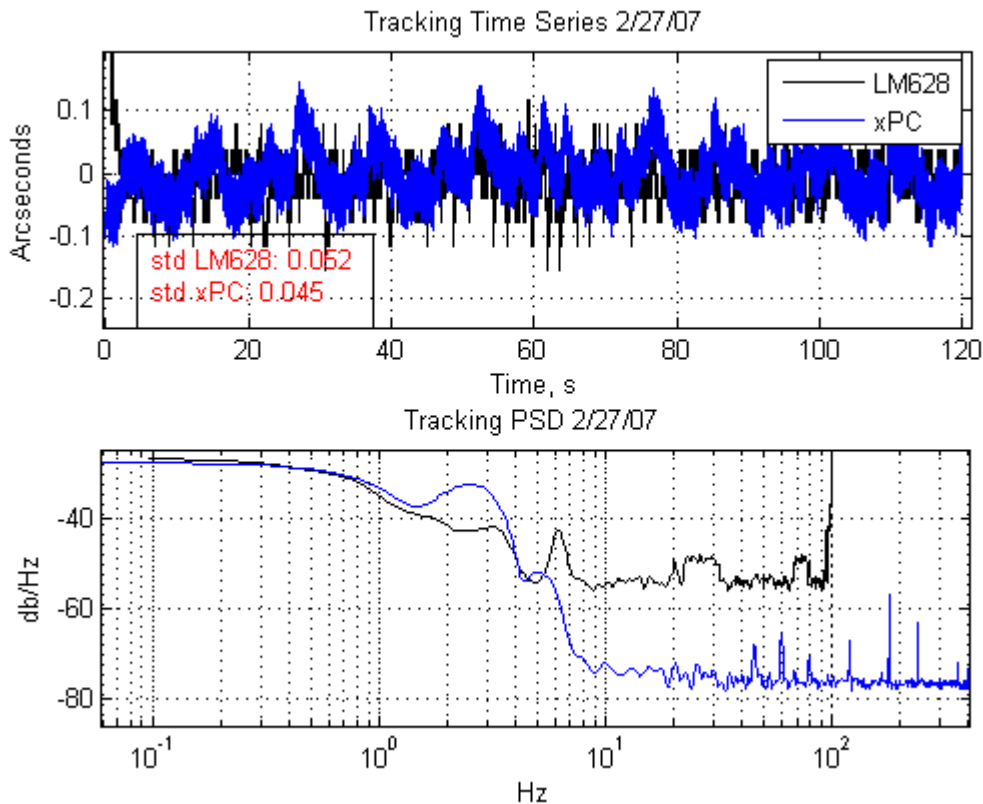


For the above time series, the RMS of the “standard” gain position error is 0.0452 arcseconds, with a range of 0.261 arcseconds, while the “test” gain has an RMS of 0.0897 arcseconds, and a range of 0.505 arcseconds.

As might be expected from the time series, the PSD towards DC is higher overall for the “test” servo gains. Both show that the heretofore troublesome 2.5, 6, 20, and 38Hz modes are well down from levels experienced with the existing LM628-based servo controller. High-frequency signal power is present, however. Much of this is 60Hz and its integer multiples, and can most likely be attributed to power supply ripple in the linear amplifiers used for testing. This will go away when the controller is using the Copley Controls PWM amplifiers, which have both more DC bus capacitor filtering, and output-smoothing filters.

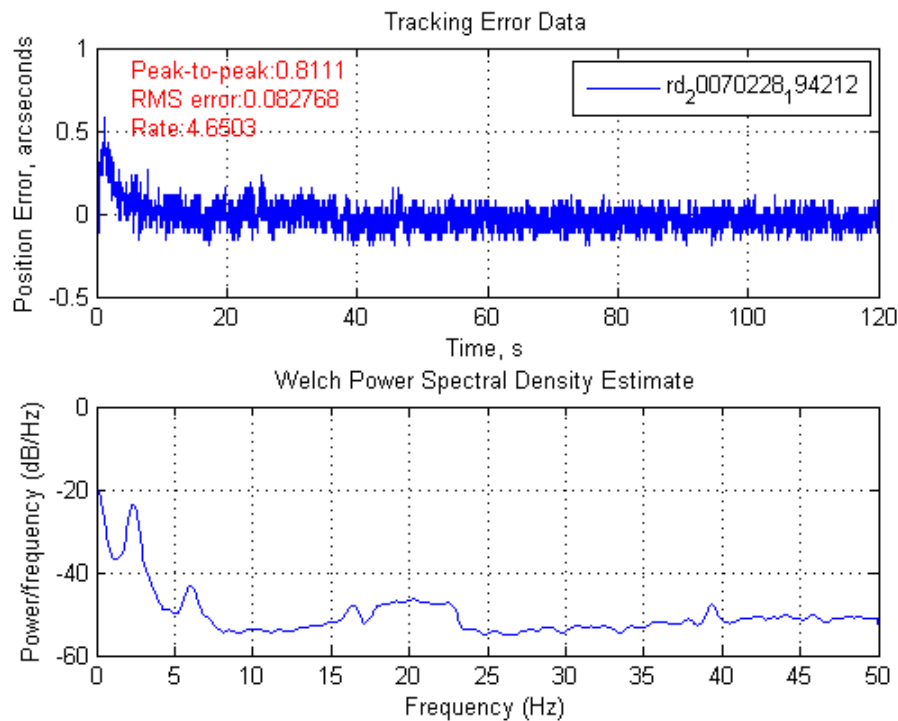
For direct comparison to the existing servo, we have data collected on 2/27 with the chamber doors closed on the LM628-based servos, as well as logging data collected nightly on every object the MMT acquires. A small reduction script was written to handle these logging files, and generate a plot of both the time series and its PSD for direct analysis of the MMT tracking performance.

Comparing the chamber-closed tracking data between the two controllers, we have the graph below.



The new controller appears to have more low-frequency content than the LM628, but the reader should be cautioned that the LM628 data are sampled at 100Hz, while the xPC data are sampled at 1kHz, so there may be other features in the LM628 data that remain unseen. The peaking at 2.5Hz is gain peaking in the closed-loop servo. The xPC Target controller is clearly superior in suppressing modal frequencies than the LM628 servo. The very-low frequency structure in the xPC Target tracking data are not detected in simulation, so nonlinearities such as friction and quantization of the signals will probably have to be added to bring the time-simulation more into line with the real system, as we may be experiencing limit-cycling or some other issue when tracking with the xPC Target system.

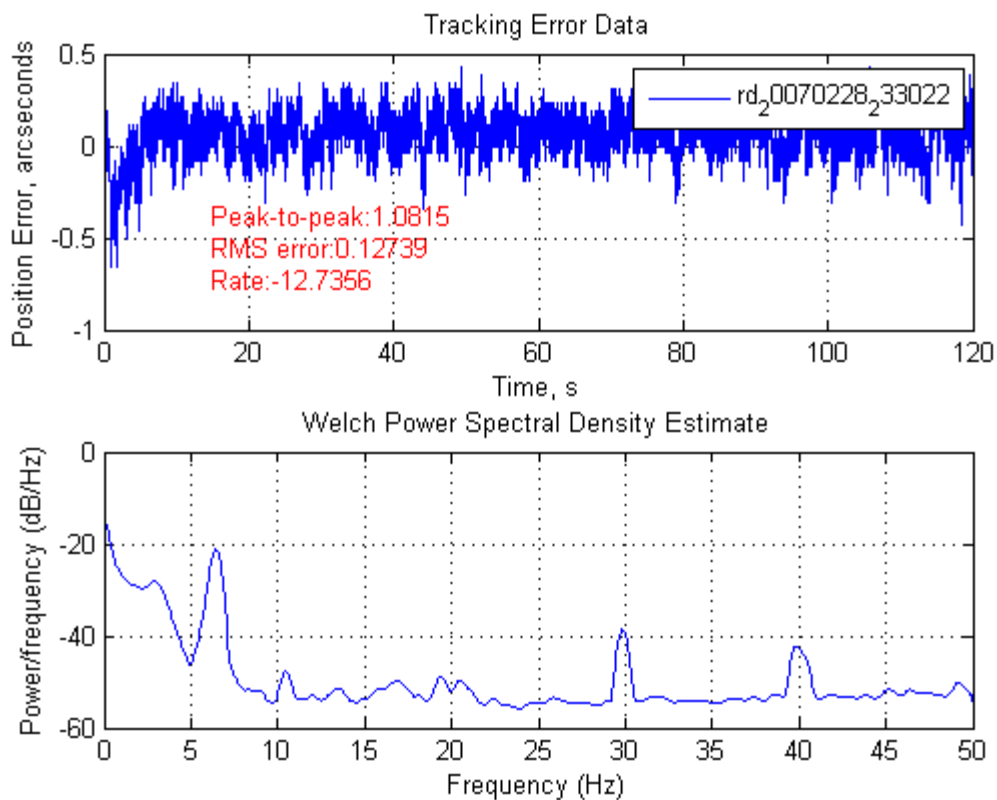
The log-file data reduction script reduces night-time tracking data in a similar fashion to the graph above. The night of January 28 had a couple of relevant tracking files:



The afternoon of 2/27/07, two separate chamber-closed tracking log files were collected with the LM628 system. The data can be compared to the tracking data collected with the new controller in tabular form, in arcseconds:

	LM628 tracking #1	LM628 tracking #2	xPC Standard Gains	xPC Test Gains
Peak-to-peak	0.2317	0.1931	0.2611	0.5053
RMS	0.0316	0.0348	0.0452	0.08971
Velocity	1.045	-1.68	5	5

In operation, it's clear that 2.5Hz servo gain peaking is present in the LM628 servo, as well as 6, 20, and 38Hz. The tracking rate was similar to that collected on 2/27, so it can be compared to the data collected that day. Increases in tracking velocity, as was seen in the data collected earlier in the month, appears to increase to overall servo error, as this next tracking log file output shows:



Notice how the 6Hz mode has increased, along with a new feature at 30Hz. Information that would be helpful in analysis of the tracking data would include the relative wind azimuth and speed, combined with some estimate of the total wind load during this tracking period.

In all, 11 objects were tracked and the mount data logged on the night of 2/28/07. The data from the logs, when reduced, give the following tracking error values for 10 of them (one was only there for 3.5 seconds). The wind varied from the teens to mid-20s.

File	Peak-to-peak	RMS error	Tracking rate
rd 20070228 1942*	0.811	0.0828	4.65
* 1944*	1.158	0.076	4.4
* 1946*	0.7725	0.064	4.29
* 2007*	0.579	0.0615	1.88
* 2010*	0.386	0.055	1.62
* 2011*	1.12	0.064	1.34
* 2049*	1.12	0.076	3.16
* 2055*	1.31	0.095	-3.78
* 2330*	1.08	0.1274	-12.73
* 2334*	1.39	0.115	-12.73

For all the above logging files, the PSD data show clear frequency spikes at 2.5, 6, 20, and 40Hz, with most of the power contained in the 2.5 and 6Hz modes. A great deal of the total tracking error budget is consumed in these two oscillatory frequencies.

Only one obstacle now remains to release the controller for testing in operation: safely starting the controller without the oscillation noted in the last report. More data was collected on this phenomenon, and some effort was expended to discover its source, without success. A version of the controller was built and tested by Trebisky on his mount-simulator computer that exhibits the same behavior, so we have a tool to track this down and correct it.

Next Steps:

The notch filter designs are complete, and no more optimization for them is contemplated. There is more work to be done in servo gain optimization, and getting a wind-loading simulation put together, in addition to whatever work is needed to implement the servo on the VxWorks controller. The follow-up work ahead is:

1. Find and fix the startup oscillation problem.
2. Work on gain optimization to increase the controller bandwidth.
3. Work on improving model fidelity to the actual system to more properly predict gain margins and tracking performance.
4. Simulate and verify gains and wind rejection tests.
5. Simulate and verify slewing results from the past two test runs.
6. Release off-line data reduction scripts for tracking logs to allow staff access to tracking data via the MMT web-browser interface software. Ensure wind loading data are available as part of the reduction annotations.
7. Test the latest controller updates on the telescope with the VxWorks implementation.
8. Test the latest controller iteration with the $f/9$ configuration.
9. Simulate and test on hardware changing the motor amplifiers over to the Copley PWM units.
10. Release for nightly use, changing over hardware as needed to bring up the new system.

