LASER METROLOGY SYSTEM FOR STELLAR INTERFEROMETRY

by

Braden E. Hines

Submitted to the Department of Electrical Engineering and Computer Science on January 15, 1988 in partial fulfillment of the requirements for the Degree of Master of Science in Electrical Engineering.

ABSTRACT

Optical interferometry promises to produce astrometric measurements of unparalleled accuracy. However, since optical interferometry by its very nature pushes against the frontiers of many different technologies at once, sophisticated and highly accurate systems are needed to monitor many aspects of the instrument's performance. This thesis is concerned with one such system, a laser metrology system designed to measure motions of the siderostats which reflect starlight into the interferometer optics.

As the siderostats sweep the sky to point at and track different stars, the surface of the siderostat mirror, unfortunately, does not pivot about a fixed point. There are a number of imperfections in the siderostats; among them are irregularities in the bearings and gears of the siderostats, noticeable backlash in the gears, and nonintersection of the two axes of rotation. In the Mark III Interferometer at Mt. Wilson Observatory, the siderostats have been designed in such a way as to accommodate the placement of a laser plate beneath the siderostat mirror. This laser plate consists of five laser interferometers, four of which are pointed at the siderostat mirror. These laser interferometers are used to measure the motion of the surface of the siderostat mirror; knowledge of this motion then allows computation of a correction to the measured fringe position for the star, reducing the residual errors after solving for the baseline. Without the laser plate, errors on the order of 5-20 microns in fringe position can occur.

A laser plate system was installed and tested at Mt. Wilson to determine the resultant reduction in the residuals. Hardware was designed and built to allow a computer to read the values measured by the laser interferometers to an accuracy of \( \lambda /64 \), where \( \lambda \) equals 633 nm. Software was implemented to record the data in real time, process the raw data, solve for the laser plate model, convert the laser readings to mirror motion, and, lastly, convert mirror motion into changes in fringe position. The system was tested and debugged at Mt. Wilson, and data is presented which illustrates the function of the system. This thesis describes the design and testing of the laser metrology system.

The Mark III interferometer is a joint effort of the Massachusetts Institute of Technology, the Harvard/Smithsonian Astronomical Observatory, the United States Naval Observatory, and the Naval Research Laboratory.

Thesis Supervisor: David H. Staelin
Title: Professor of Electrical Engineering
ACKNOWLEDGEMENTS

I’ve really enjoyed working on this project, and, while the work itself is very stim-
ulating, it’s the people that make a project enjoyable, and this project has been that.
First off, I’d like to extend an enormous vote of thanks to Dr. Mike Shao and Dr. Mark
Colavita for their continual presence in and interest in this project and in my progress.
I also want to acknowledge the guiding hand of Prof. Dave Staelin in this thesis and in
making sure it would really happen someday. Thanks to Peter Cheimets for all his work
with the “pokey-wokeys” and all the other mechanical gizmos I can’t begin to fathom.
George Kaplan was a real hero for making CALC and SOLVE the best-written programs
I’ve ever seen (at least in FORTRAN), and for providing documentation to match. And
thanks to Ed Kim at MIT and all the guys at USNO and NRL for the flame sessions.

And, on the personal side, I should mention Susan, who’s always wanted the best
for me, and I have to thank Dave and Chris and Jay for listening even when I was going
way too fast, and for helping me see the perspective I was missing. Special thanks to you,
Julianne, for you clutch hits in the waning moments of this thesis.

And I am forever indebted to my parents and my family for ever having this op-
portunity.

And I am grateful to God for giving me a deadline. Japan, here I come!
TABLE OF CONTENTS

Introduction ................................................. 6
Chapter 1. Concept: Laser Plate ......................... 15
Chapter 2. Hardware ......................................... 22
Chapter 3. A Geometric Model for the Laser Plate ...... 34
Chapter 4. Correcting for Fringe Position ............... 58
Chapter 5. Glitches and Deglitching ...................... 66
Chapter 6. Software Structure and Design ............... 83
Chapter 7. Experimental Data and Results ............... 87
Chapter 8. Conclusions ...................................... 92
Appendix. Programs and Data Files ..................... 94
Bibliography ................................................. 101
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Basic layout of an astrometric interferometer.</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>Fringe tracker / Delay line.</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>Star tracker.</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>A siderostat.</td>
<td>11</td>
</tr>
<tr>
<td>5</td>
<td>Fringe position equation.</td>
<td>14</td>
</tr>
<tr>
<td>6</td>
<td>Bowling ball on siderostat mirror.</td>
<td>17</td>
</tr>
<tr>
<td>7</td>
<td>A laser plate.</td>
<td>18</td>
</tr>
<tr>
<td>8</td>
<td>Laser plate with siderostat.</td>
<td>19</td>
</tr>
<tr>
<td>9</td>
<td>Laser plate electronics block diagram.</td>
<td>20</td>
</tr>
<tr>
<td>10</td>
<td>First-order beating.</td>
<td>23</td>
</tr>
<tr>
<td>11</td>
<td>Laser interferometers and relative phase.</td>
<td>26</td>
</tr>
<tr>
<td>12</td>
<td>Preamp.</td>
<td>29</td>
</tr>
<tr>
<td>13</td>
<td>Laser coordinate frames.</td>
<td>36</td>
</tr>
<tr>
<td>14</td>
<td>Laser plate model.</td>
<td>38</td>
</tr>
<tr>
<td>15</td>
<td>Bowling ball displacement vector.</td>
<td>39</td>
</tr>
<tr>
<td>16</td>
<td>Laser plate measuring jig.</td>
<td>43</td>
</tr>
<tr>
<td>17</td>
<td>Kalman filter equations.</td>
<td>48</td>
</tr>
<tr>
<td>18</td>
<td>Singular value decomposition.</td>
<td>55-56</td>
</tr>
<tr>
<td>19</td>
<td>Starlight Path Length.</td>
<td>60</td>
</tr>
<tr>
<td>20</td>
<td>a: Laser Data. b: A Glitch.</td>
<td>67-68</td>
</tr>
<tr>
<td>21</td>
<td>Root Sum Square error plots for each pier.</td>
<td>72-74</td>
</tr>
<tr>
<td>22</td>
<td>Root Sum Square error plots for each pier after deglitching.</td>
<td>80-82</td>
</tr>
<tr>
<td>23</td>
<td>System data flow.</td>
<td>84</td>
</tr>
<tr>
<td>1</td>
<td>Glitches removed from data set FT116F1.</td>
<td>79</td>
</tr>
<tr>
<td>2</td>
<td>Results of analysis without laser data.</td>
<td>88</td>
</tr>
<tr>
<td>3</td>
<td>Results of analysis using DEGLITCH.</td>
<td>89</td>
</tr>
<tr>
<td>4</td>
<td>Results of analysis using eyeball method deglitching.</td>
<td>90</td>
</tr>
<tr>
<td>5</td>
<td>Results of analysis using eyeball method but no FITMOD.</td>
<td>91</td>
</tr>
</tbody>
</table>
I. Introduction

Accurate determination of stellar positions has long been of considerable interest in the world of astronomy. Traditionally, the most prolific sources of astrometric measurements have been series of photographic plates made on ordinary telescopes, with accuracies on the order of 10 milliarcseconds.\(^1\) Recently, however, interferometric techniques, particularly in the visible and infrared, have shown considerable promise as means of making highly accurate astrometric measurements, potentially capable of wide-angle measurements with accuracies on the order of better than one milliarcsecond.\(^2\) In order to achieve this accuracy, however, there are a number of problems to be overcome, such as atmospheric disturbance and technological complexity. One such problem involves systematic errors in the instrument itself, errors which can be eliminated by appropriate design or calibration measures.

In order to envision places where such errors might arise, it is necessary to first have some familiarity with the basic layout of the instrument. The basic layout of the Mark III Optical Interferometer\(^3\) at Mt. Wilson, California is depicted schematically in Figure 1.

There are essentially three fundamental subsystems necessary for basic interferometry: siderostats, a star tracker, and an optical delay line/fringe tracker. The job of the siderostats is to reflect the light from a chosen star into the rest of the interferometer, making the sidereal motion of the stars appear stationary, hence the name siderostat. The job of the star tracker is to keep the two incoming wavefronts parallel so that interference fringes can be detected by the fringe tracker. The delay line is responsible for introducing additional optical path length into one arm of the interferometer so that the path lengths of the light in both arms of the interferometer (from the star to the main beam combiner)

---

\(^1\) Shao and Staelin, "First Fringe Measurements with a Phase-Tracking Stellar Interferometer."

\(^2\) Shao and Staelin, "A Long-Baseline Interferometer for Astrometry."

\(^3\) Shao et al, "The Mark III Stellar Interferometer"
Figure 1. Schematic Description of Mark III Optical Interferometer.
is the same, and the fringe tracker actually detects the interference peaks and nulls that result when the two path lengths are equal or nearly equal.

THE DELAY LINE/FRINGE TRACKER

The fringe tracker/delay line\textsuperscript{4} is depicted in Figure 2.

There is a movable cart which rides on rails inside an evacuated pipe; this cart is moved along the rails by a corrugated belt driven by a stepper motor. Finer control of delay line position is achieved with a voice coil (as in an audio speaker) and finally with a piezoelectric stack. There are actually two delay lines in the system to allow for both positive and negative delay values. When searching for or tracking a fringe, the delay line position is modulated at a high rate over a small range, causing the fringes to repeatedly “go by.” Demodulation is done in hardware and software in order to determine the fringe position, which is the delay line position at which the optical path length in each arm is exactly equal.

THE STAR TRACKER

The star tracker\textsuperscript{5} is depicted schematically in Figure 3.

Two star images enter the star tracker separated by some distance (nominally 45 arc-seconds). The job of the star tracker is to keep the two wavefronts parallel by holding the two images at two predesignated positions. In order to achieve this, it uses a photon camera as its position sensor, and it controls piezoelectrically driven beam-steering mirrors as its actuators. The photon camera reports a nine-bit x and y location for every photon that strikes the camera face. Star tracker hardware and software then computes the centroid of each star image, which then allows computation of an error signal which is the deviation

\textsuperscript{4} Holm, Richard Walt, “A High-Speed, High-Resolution Optical Delay Line for Stellar Interferometry Applications.”

\textsuperscript{5} Clark, Lloyd, “A Photon Camera Star Tracker for Stellar Interferometry.”
Side View

≈ 7 meters

Movable Retroreflector

Vacuum Tanks

Concrete Pedestal

Windows

Rails

Top View

Optical Paths

Delay Line Physical Configuration

Figure 2. Optical Delay Line.
Figure 3. Star Tracker.
of the star image from its desired location. This error signal then drives a servo which controls the piezoelectric lumps which are attached to the actuator mirror. In order to keep the piezoelectric (or PZT) mirrors from running out of dynamic range, the star tracker computer communicates the PZT positions to the siderostat control computer, causing the siderostats to move in such a way that the PZT actuators are kept near the center of their dynamic range.

THE SIDEROSTATS

The siderostat control system\(^6\) is the most closely intertwined with the laser metrology system which is the subject of this thesis. The responsibility of the siderostat system is to reflect light from the desired star into the optics of the star tracker and the rest of the interferometer. A siderostat is pictured in Figure 4.

\(^6\) Kelly, Matthew J., “A Siderostat Control System.”
The azimuth axis is tilted at an angle of 75 degrees so that the mirror itself will stick out so that a laser plate can be placed beneath it. The elevation axis is orthogonal to the azimuth axis and rotates with it. At the nominal rest position, the elevation axis is parallel to the ground. There are two stepper motors which are used to position the siderostat, one for each axis. These motors are microstepping motors which have been geared to provide 100,800,000 microsteps per revolution of each axis, giving an angular resolution of about 13 milliarcseconds. However, geartrain backlash and eccentricity cause actual performance to be somewhat worse than this. Also, there are a number of nonidealities that arise in the geometric configuration of the siderostat; for example, the elevation axis may not be exactly orthogonal to the azimuth axis or, due to inaccuracies in the placement of the siderostat or the siderostat pier, the angle of the stellar feedbeam into the main interferometer optics may not be exactly what it is designed to be. For this reason, the siderostat control computer maintains and continuously updates an eight-parameter geometric model of the siderostat based on information received from the star tracker computer indicating that stars have been found. The difference between the motor positions that were computed and those that were actually needed to point at the star forms an error signal that is used to drive a Kalman filter which computes corrections to the siderostat model.

WHY THAT'S NOT ENOUGH

The siderostat pointing system just discussed works rather well; however, when doing astrometry, the system as described does not provide some crucial information. In order to see what's missing, we first need to come to an understanding of the way the interferometer is used to make astrometric measurements. The basic equation relating fringe position to star position is

\[ x = \hat{S} \cdot B + C, \]

where \( x \) is the fringe position (defined above), \( \hat{S} \) is the unit vector pointing in the direction from the interferometer to the star, \( B \) is the baseline vector, and \( C \) is the constant term.
The baseline vector is simply the displacement from the pivot point of one siderostat to the pivot point of the other siderostat. The constant term arises from the fact that the two arms of the interferometer are not exactly equal in length even when the delay lines are both at zero position; $C$ is the amount that the delay line must be moved in order to achieve equal path length in both arms. A quick glance at Figure 5 will reveal how this equation is apparent by inspection.

In real systems, of course, there is always the possibility of a sign error if the direction of the baseline vector is not consistent with the definition of the direction of positive optical delay.

It is clear, then, that in order to obtain accurate values for the star position, it is necessary to have precise knowledge of the fringe position, baseline, and the constant term. The fringe position is ideally known to within 0.005 microns since the baseline is monitored by a laser interferometer with a resolution of $\lambda/128$. The baseline and the constant term are determined by performing a least-squares fit to the fringe position data for a night’s observing. This is where potential for inaccuracy creeps in. In order to achieve our goal of milliarcsecond astrometry, these quantities must be known to within 0.1 micron; however, the residual errors in fringe position after computing the least-squares fit can be as high as 10 microns. It is believed that most of this error is due to motion of the siderostat’s “pivot point” (i.e., a fixed point on the mirror surface about which the rest of the mirror rotates). If the pivot point moves as the siderostat sweeps the sky, then both the baseline and the constant term will change as the siderostats move. This pivot point motion is anticipated to be less than 100 microns from the rest position. While this degree of motion is virtually unnoticeable as far as siderostat pointing, its effects on astrometric measurements are devastating. Hence, a method for measuring the motion of the siderostat pivot point is desired; that is what this thesis is about.
Figure 5. Fringe position equation represented schematically.

\[ x = \hat{S} \cdot \vec{B} + C \]
CHAPTER 1: CONCEPT: LASER PLATE

Our mission is to measure the motion of the pivot point of the siderostat to a very high level of accuracy, approximately 0.1 micron. Many techniques are available for measuring distance, but for measuring distances this tiny, the only practical tool is a laser interferometer. The technical details of the use of laser interferometers for metrology are discussed later; for now, suffice it to say that a laser interferometer, when interfaced to appropriate hardware, has the ability to make extremely accurate measurements of distance. However, this technique can only measure relative motion from some reference point. A laser interferometer cannot measure absolute distances. In other words, in order to use a laser interferometer, one moves the object being measured to some known position and then "resets the hardware" (details later). This process then references all future measurements to some absolute zero position that is known beforehand.

Laser interferometers, then, can be used to measure minuscule distances. The next thing we need to do in designing a measurement system is choose something for the laser interferometers to measure. The very nature of a laser is that it is a beam of light; hence, in order to measure the distance to something, that something must have a mirrored surface that will reflect the laser light.

What we really want to measure is the motion of the pivot point about which the siderostat mirror rotates. Hence, it would be nice to just be able to point a laser beam at the pivot point and have it return. Unfortunately, it is not clear how you make what is literally a point reflect a laser beam, and, also, the pivot point is on the mirror’s front surface, which means that the laser beam (which is coming from a laser plate mounted below the siderostat mirror) is separated from the pivot point by more than an inch of opaque glass (since the mirror is 1.5 inches thick). What is needed is some magical surface that causes the lasers to act as if they are reflecting off the pivot point even though they aren’t. It turns out that such a surface isn’t all that magical; a mirrored sphere whose
center coincides with the pivot point would perform exactly as described, and, in fact, a 
mirrored hemisphere (affectionately known as a "bowling ball") has been mounted on the 
back of each siderostat mirror in such a way that the center of curvature of the hemisphere 
coincides with the pivot point of the siderostat, as in Figure 6.

Of course we are not able to place the hemisphere perfectly, but this problem can 
be circumvented and is discussed later.

Now we have a system which will cause a laser beam pointed at the pivot point of 
the siderostat mirror to be retroreflected (i.e., reflected back to its source). Clearly, if we 
want to know the motion of the point in three dimensions, one laser beam is not enough. 
In fact, three measurements are needed to completely determine the position of the center 
of the bowling ball. The design we settled upon uses four lasers, so that we have redundant 
information which can be used to perform consistency checks or to quantify any unknown 
parameters having to do with the laser plate. The laser plate is depicted in Figure 7; the 
laser plate-siderostat combination is shown in Figure 8.

Laser interferometers are nice, but they need to be interfaced to appropriate hard-
ware before they can provide any useful information. The optical output of the laser 
interferometers is a laser beam that is intensity modulated at 20 kHz. This 20 kHz pulsing 
is detected by combination photodetectors/transimpedance amplifiers that are mounted 
directly on the laser plate itself. These preamps produce a sinusoidal output that is am-
plified, filtered, and squared by postamps. The 20 kHz digital output of the postamps is 
then fed into a set of digital counters that keep track of the motion detected by the various 
laser interferometers. These counters are then interfaced to an IBM Personal Computer 
which is responsible for making sure that the data from the counters is recorded. This 
system is depicted in block diagram form in Figure 9.

The values stored in the counters (these counters will be referred to as the laser
Figure 6. Bowling ball mounted on siderostat mirror.
Figure 7. A Laser Plate.
Figure 8. Laser Plate with Siderostat and Bowling Ball.
Figure 9. Block Diagram of Laser Plate Electronics.

Fringe counters) are read into the computer every two seconds. These values are then sent to a Data General Nova that is responsible for recording the night's data on tape.

Later, the night's data goes through a sequence of processing steps. The raw data is dumped from a 9-inch reel on the Nova to the PC. Then the PC converts the raw data to fringe position, laser readings, etc. The data from the laser interferometers is first used to compute a least-squares fit to a geometric model of the laser plate (discussed later);
this can be done since the fourth laser contains redundant information. Next, the laser readings and the laser plate model are combined with information from the siderostat system concerning the orientation of the laser plate, allowing conversion of laser readings to x-y-z motion of the bowling ball. This information is then fed into a program which does a least-squares fit to determine the baseline and/or star positions. This least-squares fit also computes a parameter which is the offset of the center of each bowling ball from the surface of the siderostat mirror. The bowling ball motions are then converted to small motions of the siderostat mirror which are interpreted as changes in the baseline and constant term. These are then converted to small corrections in fringe position which are subtracted from the observed fringe position for each star, yielding what is hopefully a more accurate estimate of the “true” fringe position for the star.

That completes what is a rather sketchy outline of how the lasers are used to improve the accuracy of the system. The “proof in the pudding” that the laser metrology system is actually helping things will be a reduction in the residual fringe position errors when solving for the baseline.
CHAPTER 2 : HARDWARE

The laser metrology system, while posing a number of interesting systems modelling problems, also requires quite a bit of optical and electronic hardware just to get the data of interest into a computer where it can be used. This hardware has been described earlier in a rather sketchy manner; now we'll go into more detail.

Let's start by studying the optical half of the problem, which means a description of the laser interferometers and how we get the laser light to the right place and then how we prepare it for electronic detection. The basic idea behind laser interferometry is the use of two slightly mistuned laser beams, i.e., two beams that are separated in frequency by some very small amount. In the case of the siderostat laser plate, we have two beams that are separated in frequency by 20 kHz. When the two beams are added together, first order beating occurs (just as in mistuned musical instruments), and the intensity of the total signal is modulated at the difference frequency. To see why this happens, let's look at what happens when we add two sinusoids, one of frequency \( \omega_0 + \Delta \omega \) and the other at frequency \( \omega_0 - \Delta \omega \), with \( \omega_0 \).

\[
\frac{1}{2} \cos[(\omega_0 + \Delta \omega)t] + \frac{1}{2} \cos[(\omega_0 - \Delta \omega)t] = \frac{1}{2} \cos(\omega_0 t) \cos(\Delta \omega t) - \frac{1}{2} \sin(\omega_0 t) \sin(\Delta \omega t) \\
+ \frac{1}{2} \cos(\omega_0 t) \cos(\Delta \omega t) + \frac{1}{2} \sin(\omega_0 t) \sin(\Delta \omega t) \\
= \cos(\omega_0 t) \cos(\Delta \omega t).
\]

The intensity of the beam is proportional to the square of the amplitude:

\[
\text{Intensity } \propto \cos^2(\omega_0 t) \cos^2(\Delta \omega t).
\]

Note that \( \cos^2(\Delta \omega t) \) is periodic with a period that is half the period of the cosine. Hence, the \( \cos^2(\Delta \omega t) \) term modulates the beam intensity at an angular frequency \( 2\Delta \omega \), which is just the difference in frequency of the two beams. In other words, there is a carrier at the optical frequency \( \omega_0 \) that is amplitude modulated at the difference frequency, which is
20 kHz in this case, so a photodetector will detect laser light pulsing at a 20 kHz rate (see Figure 10).

The neat thing about this is that, while the modulation of the wave in time is 20 kHz, the modulation of the wave with path length is at the wavelength of the laser light carrier. To show this, let's assume that the two beams separate at $z = 0$. One of the beams travels to the target and back through a path length $l$ and is then reunited with its companion back at $z = 0$. If $c$ is the speed of light, then the resulting normalized wave
amplitude as a function of time and path length is

\[
E(t, l)/E_0 = \cos(\omega + \Delta \omega)t + \cos(\omega - \Delta \omega)(t - \frac{l}{c})
\]

\[
= \cos(\omega + \Delta \omega)t + \cos(\omega - \Delta \omega)t \cos(\omega - \Delta \omega)\frac{l}{c} + \sin(\omega - \Delta \omega)t \sin(\omega - \Delta \omega)\frac{l}{c}
\]

\[
= \cos \omega t \cos \Delta \omega t - \sin \omega t \sin \Delta \omega t
\]

\[
+ (\cos \omega t \cos \Delta \omega t + \sin \omega t \sin \Delta \omega t) \cos(\omega - \Delta \omega)\frac{l}{c}
\]

\[
+ (\sin \omega t \cos \Delta \omega t - \cos \omega t \sin \Delta \omega t) \sin(\omega - \Delta \omega)\frac{l}{c}
\]

\[
= \cos \omega t \{\cos \Delta \omega t[1 + \cos(\omega - \Delta \omega)\frac{l}{c}] - \sin \Delta \omega t \sin(\omega - \Delta \omega)\frac{l}{c}\}
\]

\[
- \sin \omega t \{\sin \Delta \omega t[1 - \cos(\omega - \Delta \omega)\frac{l}{c}] - \cos \Delta \omega t \sin(\omega - \Delta \omega)\frac{l}{c}\}.
\]

For simplicity's sake, let's look at the functional dependence when \( t = 0 \). Then any terms involving \( \sin \omega t \) drop out, \( \cos \omega t \) is 1, and we get

\[
E(t = 0, l)/E_0 = 1 + \cos(\omega - \Delta \omega)\frac{l}{c}
\]

\[
\approx 1 + \cos \omega \frac{l}{c}
\]

\[
= 1 + \cos 2\pi \frac{l}{\lambda}
\]

Note that this last function has nulls spaced \( \lambda \) apart, and the intensity, which is proportional to \( E^2 \), has nulls at the same points. This means that, as we alter the measured path length at some instant in time, we see intensity nulls at a rate of one every \( \lambda \) (\( \lambda = 633\text{nm} \) for the lasers we use), yet the nulls are spaced out comparatively broadly in time.

So what we want is two optical beams separated in frequency by 20 kHz. These can be produced using acousto-optic modulators (AOMs) and a pair of ordinary CB radios tuned to different channels. We feed a laser beam and channel 38 into one AOM, and a laser beam and channel 40 into another. Since channels 38 and 40 are separated by 20 kHz, we get optical beams out of the AOMs that are separated by 20 kHz. There is some other stuff that comes out of the AOMs as well, but it comes out at a slightly different angle, and we can orient our optics in such a way the “other stuff” does not propagate through the system.

24
To use this system to measure a path length, we need to have the two beams together at some starting point. We'll send one of the two beams off to the target and back to the starting point, so that the information about changes in path length will be present in the phase of this laser beam. Then, in order to get an amplitude modulated beam out of the system, we'll need to interfere the two beams. How can we use this phase information, and how do we keep the beams from interfering until we want them to interfere? The second question is the simpler one. We can keep the beams from interfering by simply making them have orthogonal polarizations. Then, when the time comes to actually interfere the two beams, we can use an appropriately oriented half-wave plate to rotate the two polarizations, causing them to interfere.

In order to use such a system to measure changes in a path length, we need to measure the relative phases of the two beams before sending one off to the target, and then we need to measure their relative phases again when the beam has returned from its trip. Next we subtract the two relative phases to get the change in relative phase as the beam traversed its path, then we look at this difference in relative phase over time. If the difference in relative phase does not change, then we know that the path length has not changed. If, however, the relative phase at the end of the path starts to advance with no corresponding advance in the relative phase at the beginning of the path, then we can deduce that the path length is decreasing, because something is causing the beam to arrive at the detector sooner than they it did previously. Actually, it could mean the path length is increasing; it depends on which of the two beams we decide to send to the target. In practice, we must empirically observe the dependence of phase on path length changes to get the signs right. See Figure 11 for a pictorial description of what's going on.

So, it can be seen that for each measurement we want to make, we need a detector to measure the relative phase of the returning beam (this is what we normally refer to as just the "signal" from that particular interferometer), and we need a detector to measure
Laser Interferometers and relative phase

Figure 11.
the relative phase of the beam before it starts its journey (this we call the "reference"). It is important that the laser plate stay constant in size so that we are sure we’re measuring the change in length of the plate-to-bowling-ball path, and not a change in the size of the laser plate due to thermal expansion. Conveniently, if we have a number (let’s suppose there are four) of interferometers on a laser plate which does not vary in its dimensions, we can use a single reference for all four signals. Hence, each plate needs only five interferer-detector pairs rather than eight. The requirement of having a “plate whose dimensions do not vary” is satisfied by constructing the plate entirely of low-expansion invar and through the use of thermal expansion cancellation techniques. What we refer to as the laser reading associated with a particular signal is the relative phase of the returning beam minus the relative phase of the pre-departure beam. In order to be able to measure motions greater than a single wavelength of light, digital hardware is used to unwrap the phase into integer and fractional wavelengths; this hardware will be described momentarily.

The phase unwrapping that will occur later on is not foolproof; let’s look at an example of motion that will cause problems. Suppose the bowling ball, for some reason, starts to move away from the laser plate such that the path length travelled by the laser beams is increasing at a rate of 20,000 wavelengths per second. What happens in this case is that the beat frequency of 20 kHz gets Doppler shifted to zero! This means that both the analog and digital hardware have nothing to look at, which is rather difficult for them to cope with. In fact, if the path length increases at a rate even faster than 20,000 wavelengths per second, we Doppler shift into negative frequency. Unfortunately, there is no way for the electronics to discern positive frequency from negative, and the net result is that we end up counting the same pulses more than once, causing the system to incorrectly perceive the change in path length. This bit about “counting pulses twice” may seem a bit confusing, but it really happens. Consider a sound wave propagating through the air. Now imagine yourself sitting at some point, counting the wave crests as they go
by. Then, instead of sitting still, you start to move away from the source, still counting crests. As your speed increases, the crests come at a slower and slower rate. At some point, your speed matches the speed of the wave, and the crests stop coming at all. Then, as you start to increase in speed past the speed of sound, you start to overtake parts of the wave that you've already seen, counting crests that you've already counted. Now, you move back to your original starting point at a nice slow speed. When you return, the number of wave crests that you will have counted will be greater than the number actually created at the source, because you will have counted some crests more than once. This is the same problem experienced by the detector under conditions of severe Doppler shift.

While that essentially concludes the conceptual description of how the optics work, there are quite a number of real-world tasks associated with the optics that are real bears. The tasks of designing, assembling, and aligning a laser plate and the laser distribution plate (which feeds the polarized light to each of the three laser plates) are Herculean efforts requiring days or weeks (in the case of assembly and alignment) or even months (for mechanical design) to complete. There is also a considerable amount of effort that needs to be expended to measure various laser plate parameters; this technique will be described in a later chapter. For the remainder of this chapter, we'll assume that some good spirit has been benevolent enough to see to it that all the optical twiddling has been done already, and that beautiful 20 kHz amplitude-modulated laser light is striking all five detectors on each laser plate.

Once we have the light striking the detectors, we need to start worrying about things like what kind of an animal a detector is going to be, and what we're going to do with the detected signal. The strategy that will be used is as follows. We want the detectors themselves to be as small as possible so that they will not take up large amounts of space on the laser plate. This means that the function of any electronics on the plate itself will be limited; we may not be able to do all the signal processing we want to do
right on the laser plate. Therefore, we'll partition the signal processing problem into two parts — detectors or preamp modules which will be mounted on the laser plate itself, and postamps which can be larger and do more complicated signal massaging.

Let's talk about the preamps, then. There are five preamps on each plate, and each preamp fits in a small metal box approximately 1 inch square by 1/2 inch deep (see Figure 12).

![Photodetector/preamplifier](image)

Figure 12. Box is about 1 inch square.

In the metal box is a tiny circuit board; the principal component on this circuit board is a metal can that contains a combination photodiode/operational amplifier. This metal can has a transparent cover so that light can strike the photodiode; the preamps are designed so that the metal can fits into a recessed hole inside the box so that the
transparent cover of the can is exposed to the outside. When aligning the laser plates, the beams are adjusted to shine directly on the photodiode. There are a few other components inside the preamp box — resistors and capacitors which provide feedback or filtering for the op-amp. The output of the circuit is low-pass filtered before being shipped out. The preamp feeds its output to a SMA-type connector mounted on the metal box which is then connected to a postamp. The preamp operates from a nearby ±15 volt power supply. The basic function of the preamp then, is to perform the bare minimum of converting pulsing light into a sinusoid in the 100 millivolt range, so that we get something that can be piped over short distances. Bare-bones filtering is also performed to reduce the amount of noise.

What we have now is a fairly noisy sinusoid at 20 kHz. We would like (for reasons explained later) to convert it to a 20 kHz square wave that goes from 0-15 volts. The circuitry necessary to do this signal processing is contained in the postamps. The strategy of the postamps is to filter out unwanted frequencies, provide gain, and then convert to digital using a zero-crossing detector with some hysteresis. First, the input signal goes through a passive RC low-pass filter, then it is buffered by a voltage follower. Then, since we know that we are only interested in frequencies around 20 kHz, two active circuits are used to perform 3rd-order highpass filtering at 5 kHz and 2nd-order lowpass filtering at 30 kHz. The signal is next amplified by a variable gain stage, and lastly it goes into a Schmitt trigger for conversion to a digital form.

If everything has worked well so far, we now have a set of five digital signals for each laser plate. The digital square waves still contain the information concerning their relative phases that was present in the original optical signal. What we really want now is some black box that will convert this continuously present relative phase information into information about changes in path length of each of the laser beam paths. This is the job of the laser fringe counters. The counters are used in the following manner: at some initial time, the counters are all cleared (i.e., the initial path length of each beam is set
to zero). After that, they will continuously keep track of the current path length for each beam (referenced to the previously set zero), making this information available as a set of digital numbers to a host computer. The host computer can then record sets of laser readings whenever it wants.

The laser fringe counters are rather complicated, but they are very interesting devices. Note that since there are 5 lasers per plate times 3 plates, we need 15 counters in all. Also, appropriate input buffering and computer interfacing is needed, but I won't discuss those aspects of the problem here, as they are not especially relevant. The technique used for doing the fringe counting, however, is pertinent, since it reveals constraints and abilities of the system as a whole. The laser counters are each made up of two parts, an integral fringe counter that counts path length changes in units of one wavelength of light, and a phase counter that keeps track of path length changes in units of $\lambda/128$ up to a limit of one wavelength ($\lambda$ is simply the wavelength of the laser light, or 633 nm).

Let us first discuss the method used by the integral fringe counters to keep track of the path length in units of whole wavelengths. In essence, the function of these devices is to do the phase unwrapping that has been discussed previously. The problem is somewhat easier than the typical phase unwrapping problem, however, in that we have a reference signal to which we can compare the phase of the unknown signal. If we assume that the path length travelled by the reference beam does not change, then the reference beam will always have zero phase (this is why we call it the reference). In truth, the phase of the reference signal may drift if the relative phase of the two input beams shifts (due to the feed fibers stretching from wind or temperature fluctuations, perhaps); however, all the unknown signals will be affected in the same way since they use the same two input beams, so the phase drift is a common-mode drift, occurring in the reference and the unknowns simultaneously. Hence, if we use just the phase differences between the reference and the unknowns to compute path length changes, we'll get the correct result. This instability
of the absolute phase is the main reason for including a reference signal in the first place; the other reason is that the phase unwrapping is significantly simpler if a reference signal is present.

The integral fringe counters can then be rather simple devices, in essence just up-down counters. For every rising edge of the unknown signal, the counters decrement, while they increment for every rising edge of the reference. In order to avoid confusion that could arise from reading the counters at different times in the cycle, the counter data can be latched on every rising edge of the unknown for consistency in the data. At some initial time, we reset all the counters to zero, and this defines the nominal rest position of the bowling ball. Now, if the path length starts to increase slightly, the unknown signal will start to lag the reference. Until it lags the reference by a full cycle, the counter will contain a zero whenever it is latched. Once it lags by more than a full cycle, however, the counter will contain a one whenever it is latched. In this way, the number in the latches at any given time is the number of whole wavelengths that the round-trip path length has changed. Individual counter chips are chained together to get the number of bits necessary to support the desired dynamic range. Note that this scheme is capable of reporting both positive and negative results; the number read from the latches is in two's complement form.

The fractional fringe counter (or the phase counter, as I referred to it above) is only slightly more complicated. It was decided that the laser fringe counters should be designed to have a resolution of 1/64 of a wavelength. To accomplish this, we first use a phase-locked loop with a divide-by-64 feedback to generate a clock that is 64 times the reference frequency (1.28 MHz in this case, since the reference is 20 kHz). This signal is then used to clock a counter that is reset at every rising edge of the reference. Like the integer counters, this fractional counter is latched on every rising edge of the unknown signal. So, if the rising edge of the unknown signal occurs immediately after the rising
edge of the reference, the counters will contain zero when they are latched. On the other hand, if the rising edge of the unknown does not occur until just before the next rising edge of the reference, the counters will have counted to almost 64, since they receive 64 clock pulses during each period of the reference waveform. In this way, the number stored in the latches is proportional to the phase lag of the unknown signal behind the reference, with 64 counts corresponding to one full cycle. Note that if the unknown starts to lag the reference by more than a full cycle, the fractional counter will report zero phase, but the integer counters will increment. Hence we see that the fractional counters contain just the present phase of the signal, while the integer counters are responsible for keeping track of the unwrapped phase.

Lastly, recall that the laser fringe counters are sensitive to changes in the round-trip path length travelled by the laser beam. Now note that if the bowling ball moves away from the laser source by a distance $z$, then the round-trip path length increases by $2z$. Therefore, since our counters can detect changes of $\lambda/64$ in round-trip path length, they can measure the motion of the bowling ball to twice that resolution, an accuracy of $\lambda/128$, or about 5 nm for laser light at 633 nm wavelength.

Practically speaking, the most difficult thing about the laser fringe counters was replicating the hardware 15 times and making it all work. From there, there is some rather sophisticated interface circuitry to allow the IBM PC to read the counter values at high speed through an 8255 parallel port chip. We originally used a Tegmar DADIO card in the PC for the interface; any 8255 board will do, however. At present, the DADIO card has been removed and the laser counters are read in through a spare 8255 present on the QuaTech interface board that is used by the shared memory (the shared memory is a three-bank three-port memory device used for communication between the various computers controlling the interferometer).
CHAPTER 3 : A GEOMETRIC MODEL FOR THE LASER PLATE

As has been discussed, a moderately reliable technique has been implemented for monitoring path length changes in the direction of the four laser beams on each laser plate. We still have a ways to go, however, to get to the point of converting this information into changes in fringe position. The first step to getting there is finding out how the bowling ball is moving. Later, since the bowling ball is attached to the siderostat mirror, we can figure out how the mirror is moving, then we'll be able to use that information, together with the star's position, to say something about how the fringe position should be corrected. So, the question of the moment is "How do we convert laser readings into bowling ball motion?"

Let's examine exactly what it is we're measuring with the laser interferometers. At some initial time, we reset the laser counters, which is the same as calling the initial position of the bowling ball the zero position. Later, as the bowling ball starts to move, we'll reference all its motion to that initial position, which is the point where all the laser readings are zero. Indeed, if, at some later point, we return the bowling ball to its original position, all the lasers should give zero readings again. What is it the lasers measure as the bowling ball moves? Note that the path length of the laser beam will not change for motions of the bowling ball in directions that are orthogonal to the laser beam. Hence, the laser readings are sensitive only to motions along the direction of the beam, and, in fact, what each laser reports to us is the motion of the bowling ball along the direction of that particular laser beam. So, with four lasers aimed at each bowling ball, the laser readings we end up with can be viewed as the "position" of the bowling ball along each of four "coordinate axes." Well, we know that it only takes three non-coplanar vectors to form a basis for three-space, so that means that, using any three of the four laser beams, we can form a basis for three-space and have all the information necessary to compute the position of the bowling ball in x-y-z coordinates. Since we also have a fourth laser, we
have redundant information which can be used to detect and correct inaccuracies in the system.

For the purposes of performing the desired transformations, I will attach a couple of coordinate frames to the point coinciding with the time-zero position of the center of the bowling ball (See figure 13), since that represents our nominal zero point.

These coordinate frames are attached to the center of the bowling ball, but their directions are defined by the orientations of other things in the system, such as the four laser beams. The first of these frames is a simple Cartesian frame, with the z-axis coinciding with the central laser beam (laser 4). The x-axis is orthogonal to the laser 4 beam and points towards laser 1. This coordinate frame will be referred to as the Plate frame $P$. Clearly, we will have three separate laser plate coordinate frames, $P_N$, $P_S$, and $P_E$, corresponding to the north, south, and east laser plates. We will use a prescript notation to represent which coordinate frame we’re talking about. For example, $P^x x$ refers to the x-axis in frame $P_N$.

A second coordinate frame that is of interest is what I will call the Mt. Wilson coordinate frame, $W$. This coordinate frame has its x and y axes parallel to the earth’s surface, while the z-axis $W z$ points straight up (again refer to Figure 13). $W x$ points directly north, while $W y$ points to the west.

A third coordinate frame that will be used later in the analysis is the polar coordinate frame, since this is the coordinate frame to which star positions are referred.

There are reasons for defining each of these coordinate frames. The laser plate frames $P$ are the most convenient for converting laser readings to bowling ball motion; most of the parameters involved in the geometric model of the laser plate (which will be described momentarily) are most easily described in terms of the laser plate coordinates. The Mt. Wilson frame $W$ is needed because a number of parameters of interest that are
Laser Plate With Siderostat Side View

Laser Plate Coordinates

Mt. Wilson Coordinates

Laser Plate Coordinate Frames

Figure 13.
derived by other interferometer subsystems (such as the angles of the stellar feedbeams into the interfering optics) are expressed with respect to this coordinate frame. It is therefore most convenient to do manipulations involving these quantities in Mt. Wilson coordinates. The polar coordinate frame is used in the final data reduction since a number of quantities of interest (such as elevation angle to the star) are derived from the polar coordinates of the star.

In a moment, I will outline the strategy by which these various coordinate frames will be used to figure out the bowling ball motion, but it is first necessary to understand how I attempt to remove the effects of the physical irregularities of the laser plate on the data by assuming a geometric model of the laser plate. This geometric model is very simple, involving a set of five angles for each laser plate (see Figure 14).

Laser 4 is the z-axis \( \text{^p}z \), so no model parameters are required to fully specify its orientation. The laser 1 beam, by definition, lies entirely in the \( \text{^p}x - \text{^p}z \) plane, so it is only necessary to model the elevation angle of the beam, \( \phi_1 \), to fully specify its trajectory. Lasers 2 and 3, however, do not lie in any special orientation, so two angles are required to fully describe their trajectories; these are the elevation angles \( \phi_2 \) and \( \phi_3 \) and the azimuth angles \( \theta_2 \) and \( \theta_3 \). Recall that these angles are all that is required to fully pin down the path of the beam through space, since we are implicitly given the fact that all of the beams pass through the origin (the center of the bowling ball); this is a convenient side-effect of the fact that the beams have been adjusted so that they retroreflect off of the spherical mirror.

Note that we can round out the set of angles by including \( \phi_4 = \theta_4 = \theta_1 = 0^\circ \). These angles are not part of the parameterized laser plate model; they are constants in the laser plate model that have arisen out of the way things have been defined. The laser plate has been designed such that the angles are nominally \( \phi_1 = \phi_2 = \phi_3 = 18^\circ, \theta_2 = 120^\circ, \) and \( \theta_3 = 240^\circ \).
Figure 14. Angles in Laser Plate Model.
Given these angles, then, it is apparent that the unit vector in the direction of laser beam \( i \), \( \hat{L}_i \), is simply

\[
\hat{L}_i = \begin{pmatrix} -\cos \theta_i \sin \phi_i \\ -\sin \theta_i \sin \phi_i \\ \cos \phi_i \end{pmatrix}.
\]

Next, let's define the vector \( \mathbf{D} \) as the bowling ball displacement vector (see Figure 15).

![Diagram showing bowling ball displacement](image)

Figure 15. Bowling Ball Displacement Vector.

This is the vector from the origin (which is where the bowling ball center was at time zero) to the present position of the bowling ball center. Let's also define \( l_i \) to be the "laser reading" (described in Chapter 2) of the \( i \)th laser. In other words, \( l_i \) is simply the motion of the bowling ball along the direction of the \( i \)th laser beam. Then it is again apparent from Figure 15 that

\[
l_i = \hat{L}_i \cdot \mathbf{D},
\]
or, writing it another way,

\[
\begin{pmatrix}
  l_1 \\
  l_2 \\
  l_3 \\
  l_4
\end{pmatrix}
= \begin{pmatrix}
  \hat{l}_1 \\
  \hat{l}_2 \\
  \hat{l}_3 \\
  \hat{l}_4
\end{pmatrix}
\begin{pmatrix}
  D_x \\
  D_y \\
  D_z
\end{pmatrix},
\]

where \( D_x \), \( D_y \), and \( D_z \) are the components of \( \mathbf{D} \). This is the fundamental expression relating bowling ball motion to laser readings. From this expression, we can see that if we are given \( l_1 \) through \( l_4 \) and \( \hat{l}_1 \) through \( \hat{l}_4 \), then a least-squares fit can be performed to compute values for \( D_x \), \( D_y \), and \( D_z \). If the previous expression is rewritten as

\[
1 = \mathbf{L} \begin{pmatrix}
  D_x \\
  D_y \\
  D_z
\end{pmatrix},
\]

then the least-squares solution of \( \mathbf{D} \) is

\[
\begin{pmatrix}
  D_x \\
  D_y \\
  D_z
\end{pmatrix}
= (\mathbf{L}^T \mathbf{L})^{-1} \mathbf{L}^T \mathbf{1}.
\]

Of course, in practice we use techniques that are more numerically sound than inverting the matrix to find the least-squares solution; actually we use LU decomposition and back-substitution. We first write the normal equations for the system,

\[
\mathbf{L}^T \mathbf{L} \begin{pmatrix}
  D_x \\
  D_y \\
  D_z
\end{pmatrix}
= \mathbf{L}^T \mathbf{1}.
\]

Next, if we let \( A = \mathbf{L}^T \mathbf{L} \) and \( b = \mathbf{L}^T \mathbf{1} \), then we can rewrite the equation as

\[
A \mathbf{D} = b.
\]

We can then break down \( A \) into the product of a lower triangular matrix \( A_L \) and an upper triangular matrix \( A_U \): \( A = A_L A_U \). This will then give

\[
A \mathbf{D} = (A_L A_U) \mathbf{D} = A_L (A_U \mathbf{D}) = b.
\]

Note that this equation can be solved by first finding some vector \( \mathbf{y} \) such that

\[
A_L \mathbf{y} = b
\]

40
and then solving

\[ A_y D = y. \]

Whence do things this way? The advantage is that the solution of a triangular set of equations is quite trivial. The first equation can be solved by forward substitution, while the second can be solved by back substitution. These are both quick, numerically accurate operations. The LU decomposition necessary to accomplish this is easily performed using Crout's algorithm with partial pivoting, and is well documented in the literature, so we won't dwell on it here; suffice it to say that it can be done accurately and efficiently.

Well, things look good so far, as long as the laser beam angles adhere closely to their design values, but what if they don't? This is why a parameterized laser plate model has been defined. While \( \phi_4, \theta_4, \) and \( \theta_1 \) are fixed by definition, the other angles \( \phi_1, \phi_2, \phi_3, \theta_2, \) and \( \theta_3 \) vary depending on how the laser plate has been constructed and adjusted. For this reason, assuming that the laser plate is accurately described by the nominal values of these angles leads to erroneous results. Therefore, we would like to be able to more accurately determine the values of these angles. There are two ways of doing this: 1) Measuring them somehow, or 2) Using the redundant laser information to perform some sort of optimal fit of these parameterized angles to the laser data.

Let's first talk about measuring the angles. One could, of course use some sort of glorified protractor to measure the angles between the various beams, but a little thought reminds one that retroreflection of a laser beam through a pinhole will probably yield the greatest angular accuracy. The problem is then to determine off of what to retroreflect the laser beam. Again, it doesn't take long to realize that the siderostat mirror directly above the laser, if flipped upside down, would provide an excellent surface for retroreflection. There is a bonus in using the siderostat mirror; we can determine the orientation of the laser plate relative to the siderostat mirror as well as the relative orientations of the laser beams to one another. This is a big bonus. As described earlier, while it is easy to measure
and compute bowling ball motion in the laser plate coordinate frame, we need a way of converting this motion to motion in the Mt. Wilson coordinate frame, which means we need to come up with a transformation between the two frames. This was seen as a major hurdle during much of the evolution of the laser plate system; surveying operations of frightening complexity were envisioned to perform this task. The bonus comes when we remember that the siderostat system operates in Mt. Wilson coordinates, so, by carefully noting the angles of the siderostat mirror as each of the four lasers is retroreflected, we can determine the direction of each laser beam in Mt. Wilson coordinates. We are already measuring the direction of each beam in laser plate coordinates (it is just \( \hat{L} \)), so we have a way to get this all-important transformation.

There is a small hitch in that the laser beams used by the laser plate are focused to a point on the surface of the spherical bowling ball mirror. When the siderostat mirror is flipped over so that its front surface can retroreflect the lasers, one notices that the mirror surface lies a few inches behind the focal point of the laser beam, and that the retroreflected beam is therefore not even close to being a point. For this reason, it is necessary to mount a measuring jig on the siderostat mirror (see Figure 16).

The measuring jig is simply a small adjustable mirror mounted on the end of an extension post. The purpose of the jig is to get the mirror to be at the focus of the laser beam. Of course, now we have another problem in that we have to align the mirror on the jig to the siderostat mirror so that the siderostat mirror angles that we record are the same as the jig mirror angles. To align the jig to the siderostat mirror, we first adjust the siderostat mirror to retroreflect a laser beam coming from inside the main interferometer building. Then the jig is mounted on the siderostat, and the jig mirror is adjusted until it, likewise, retroreflects that same laser beam. At this point the two mirrors are aligned to each other, and we are now ready to go through a measurement sequence involving retroreflection of each of the four laser plate lasers.
Figure 16. Measuring Jig.
Specifically, how are these measurements used to calculate the quantities of interest? First, I should explain that the siderostat control system has been expanded to assist in this measurement process, and it will report (with the help of a human to retroreflect the beams) the unit vector in the direction normal to the surface of the siderostat mirror in Mt. Wilson coordinates corresponding to the point at which each laser was retroreflected. Let's denote this as \( \hat{w} \hat{m} \), the mirror normal in the Mt. Wilson coordinate frame. The unit vector in the direction of one of the lasers is then simply \( \hat{w} \hat{L}_i = - \hat{w} \hat{m}_i \), where \( \hat{w} \hat{m}_i \) is the mirror normal vector when the \( i \)th laser is retroreflected. Given this information, we can then compute the unit vectors \( \hat{w} \hat{x}_P, \hat{w} \hat{y}_P, \) and \( \hat{w} \hat{z}_P \), which are simply the unit vectors \( \hat{p} \hat{x}, \hat{p} \hat{y}, \) and \( \hat{p} \hat{z} \) expressed in Mt. Wilson coordinates. These quantities can be computed quite readily:

\[
\begin{align*}
\hat{w} \hat{x}_P &= \hat{w} \hat{L}_4 \\
\hat{w} \hat{y}_P &= \frac{\hat{w} \hat{L}_1 \times \hat{w} \hat{L}_4}{|\hat{w} \hat{L}_1 \times \hat{w} \hat{L}_4|} \\
\hat{w} \hat{z}_P &= \hat{w} \hat{y} \times \hat{w} \hat{z}.
\end{align*}
\]

The transformation \( \hat{w} T_P \) from laser plate coordinates to Mt. Wilson coordinates is then simply

\[
\hat{w} T_P = \begin{pmatrix}
\hat{w} \hat{x}_P & \hat{w} \hat{y}_P & \hat{w} \hat{z}_P
\end{pmatrix}.
\]

And, of course, the inverse transformation is then also available: \( \hat{p} T_W = \hat{w} T_P^{-1} \). Using \( \hat{p} T_W \), we can then compute the \( \hat{p} \hat{L}_i \)'s as \( \hat{p} \hat{L}_i = \hat{p} T_W \hat{w} \hat{L}_i \). Then we can calculate the values for the laser plate angles based on the measurements:

\[
\begin{align*}
\theta_i &= \arctan(\hat{p} L_{i_x}, \hat{p} L_{i_y}) \\
\phi_i &= \arctan(\sqrt{\hat{p} L_{i_x}^2 + \hat{p} L_{i_y}^2}, \hat{p} L_{i_z}) .
\end{align*}
\]

Here, \( \arctan \) is the two-argument, four-quadrant version of the arctangent.

That's it then; we now have everything we need to convert from laser readings to bowling ball motion in the laser plate frame to bowling ball motion in the Mt. Wilson
frame. One question that springs to mind, though, is "Are our measured values for the laser plate angles good enough?" We might also ask whether our measured values for the laser plate orientation (i.e., the $^wT_p$ and $^pT_w$ transformations) are good enough, but we are stuck with these, and there is no way to improve them. The laser plate angles, however, can be improved upon by using the redundant information supplied by the fourth laser to adjust the angles so as to minimize the error of fit in a least-squares sense, so that merits looking in to.

I'll first illustrate that we have the makings of a least-squares problem here. As we've seen previously, if you know the laser beam unit vectors, you can compute the position of the center of the bowling ball using any three of the laser readings. We've also seen that you can compute what the laser readings should be if you know the unit vectors and the position of the bowling ball. What we could do, then, is use three of the lasers to compute the position of the bowling ball, and then compute what the fourth laser reading should be based on the position of the bowling ball and the unit vector for the fourth laser. Mathematically, then, we first compute the bowling ball center based on three lasers:

$$
D_3 = \begin{pmatrix}
\hat{L}_1 \\
\hat{L}_2 \\
\hat{L}_3
\end{pmatrix}
\begin{pmatrix}
l_1 \\
l_2 \\
l_3
\end{pmatrix}.
$$

Next, we can compute the ideal value of the fourth laser reading as

$$
l_4^{ideal} = \hat{L}_4 \cdot D_3 = \hat{L}_4 \cdot \begin{pmatrix}
\hat{L}_1 \\
\hat{L}_2 \\
\hat{L}_3
\end{pmatrix}^{-1}
\begin{pmatrix}
l_1 \\
l_2 \\
l_3
\end{pmatrix}.
$$

The functional form of this expression is then

$$
l_4^{ideal} = f(\hat{L}_1, \hat{L}_2, \hat{L}_3, l_1, l_2, l_3).
$$

Note, however, that we previously determined that the $\hat{L}_i$'s are simply functions of the $\phi_i$'s and $\theta_i$'s. Also recall that, due to the definition of the geometry of the problem, some
of these angles are fixed, so the \( f \) does not really depend on those angles. We see then that the functional dependence of \( l_{4,\text{ideal}} \) is

\[
l_{4,\text{ideal}} = f(\hat{L}_1, \hat{L}_2, \hat{L}_3, l_1, l_2, l_3)
\]

\[
= f(g(\phi_1), h(\phi_2, \theta_2), k(\phi_3, \theta_3), l_1, l_2, l_3)
\]

\[
= f(\phi_1, \phi_2, \phi_3, \theta_2, \theta_3, l_1, l_2, l_3).
\]

Let's define \( \mathbf{x} \) as the state vector containing our current best estimate of the system parameters:

\[
\mathbf{x} = \begin{pmatrix}
\phi_1 \\
\phi_2 \\
\phi_3 \\
\theta_2 \\
\theta_3
\end{pmatrix}.
\]

Let's also give a name to the vector of three laser readings that we use:

\[
\vec{l} = \begin{pmatrix}
l_1 \\
l_2 \\
l_3
\end{pmatrix}.
\]

Then the functional dependence of \( l_{4,\text{ideal}} \) is

\[
l_{4,\text{ideal}} = f(\mathbf{x}, \vec{l}).
\]

Now we can construct the least-squares problem. We can calculate an ideal value for \( l_4 \), and we measured the actual value for \( l_4 \). We'd like to choose our parameter vector \( \mathbf{x} \) such that we minimize our error in estimating the value of \( l_4 \). One way of expressing this mathematically, given a set of \( n \) measurements, is

\[
\text{find } \mathbf{x} \text{ which minimizes } \sum_{i=1}^{n} (l_{4,i}^{\text{ideal}} - l_{4,i}^{\text{actual}})^2.
\]

Now we have cast the problem as a least-squares problem. Unfortunately, the geometry of the problem is such that it is not a linear least-squares problem, since the expression for \( l_{4,\text{ideal}} \) involves sines and cosines of the parameterized angles. But, alas, worse things happen in life, and we have ways of solving non-linear least-squares problems. One
of the more popular and dependable non-linear least-squares algorithms is that put forth by Marquardt, related to an earlier suggestion of Levenberg. The Levenberg-Marquardt method is an iterative technique which varies continuously from the extremes of the inverse-Hessian method when far from the minimum to the steepest descent method as we approach the minimum. In-depth discussion of the Levenberg-Marquardt method is readily available in the literature,\(^7\) so we'll not discuss it in detail here.

Let's take a look at the way the least-squares problem has been set up. Each measurement that we have recorded involves four laser readings, but, since only one of these is redundant, we have set up the problem so that the error criterion is just \((\text{ideal}_i - \text{actual}_i)^2\). In other words, we have succeeded in producing a single "error signal" corresponding to the single redundancy present in each measurement, rather than having some more complicated error criterion depending on all four laser readings. With the error criterion in this form, it is possible to use a Kalman filter\(^8\) to estimate the parameter vector rather than a least squares approach. One advantage of this approach is that the estimate of the parameter vector is continuously improved as we process each new set of laser readings; we will see later that this is important when it comes to dealing with glitches in the data.

Figure 17 shows the equations necessary to implement a Kalman filter to estimate the values of the parameters. We first define the model for the system. As in the least squares problem, we define \(x\) to be the state vector, but since, for the case of the Kalman filter, we may have a different value of \(x\) at every measurement time, we denote the value of the state vector at the \(k\)th measurement by \(x_k\). We also introduce the notation \(\hat{x}_k\) to denote our estimate of the state vector at time \(k\), as opposed to the true value of the state vector at that time. The carat notation in this case indicates that this is an estimate (it does not imply that \(x_k\) is a unit vector). Our model for the evolution of the parameter values is simply that they remain constant except for the addition of a very small

---


\(^8\) Gelb, Arthur, \textit{Applied Optimal Estimation}.
Kalman Filter Parameter Estimation

Discrete Iterated Extended Filter

System Model:
\[
x_k = x_{k-1} + w_{k-1} + w_k = N(0, Q_k)
\]
\[
l_{4_k} = f(x_k, l_{1_k}, l_{2_k}, l_{3_k}) + v_k = h_k(x_k) + v_k = N(0, R_k)
\]

Initially:
\[
E[x_0] = \hat{x}_0
\]
\[
E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T] = P_0
\]
\[
E[w_k v_j^T] = 0
\]

State Vector:
\[
x_k = \begin{pmatrix} \phi_{1_k} \\ \phi_{2_k} \\ \phi_{3_k} \\ \theta_{2_k} \\ \theta_{3_k} \end{pmatrix}
\]

State Estimate Extrapolation:
\[
\hat{x}_k (-) = \hat{x}_{k-1} (+)
\]
\[
P_k (-) = P_{k-1} (+) + Q_{k-1}
\]

Define:
\[
\hat{x}_{k,i} (+) \triangleq \hat{x}_k (-)
\]
\[
H_k (\hat{x}_{k,i} (+)) \triangleq \frac{\partial h_k}{\partial x_k} \Bigg|_{x_k = \hat{x}_{k,i} (+)}
\]

Iterate:

Kalman Gain:
\[
K_{k,i} = P_k (-) H_k^T (\hat{x}_{k,i} (+))[H_k (\hat{x}_{k,i} (+)) P_k (-) H_k^T (\hat{x}_{k,i} (+)) + R_k]^{-1}
\]

Error Covariance Update:
\[
P_{k,i+1} (+) = [I - K_{k,i} H_k (\hat{x}_{k,i} (+))] P_k (-)
\]

State Estimate Update:
\[
\hat{x}_{k,i+1} (+) = \hat{x}_k (-) + K_{k,i} [l_{4_k} - h_k (\hat{x}_{k,i} (+)) - H_k (\hat{x}_{k,i} (+)) (\hat{x}_k (-) - \hat{x}_{k,i} (+))]
\]

Figure 17. Kalman Filter Equations.

*plant noise* \( w_k \), which is a zero-mean Gaussian noise whose covariance is described by the covariance matrix \( Q_k \). The model for the measurement, which is the reading of the fourth laser \( l_{4_k} \), is that it is a function of the parameters and the other three laser readings, with the addition of some measurement noise \( v_k \), which is also modelled as zero-mean Gaussian.
noise with a variance described by the parameter $R_k$.\(^9\)

We set the initial estimate of the state vector $\hat{x}_0$ equal to our best guess as to what the true values of the parameters are. In our case, these are the values that were determined using the measuring jig described earlier (the $E[\cdot]$ notation in Figure 17 means expected value). If there had been no measurement, we could set $\hat{x}_0$ equal to the ideal value of the parameter vector. We also set up the initial error covariance matrix $P_0$, which we use to describe how accurate we think our estimate of the parameters is and the degree of independence of the parameters. In the case at hand, we have no a priori information about the cross-covariance terms, so we simply set them all to zero. Then we set the diagonal elements equal to the variance that resulted from using the jig to measure the angles. Lastly, we assume that the plant noise and the measurement noise are not correlated from sample to sample.

Now that the Kalman filter is ready to go, we need to talk about what we do when we get a measurement. If the function $f$ in the measurement model were linear with respect to the parameters, then we would simply calculate the Kalman Gain matrix $K$ and then use it to update (improve) the error covariance matrix and our parameter vector estimate. However, the function $f$ (or equivalently, $h_k$), is not linear, so what we have to do is come up with a function $H_k$ which is a linear approximation to $h_k$ in the vicinity of the current value of the parameter vector $x_k$. This linearization is expressed in the definition of $H_k$ in Figure 17. However, note that our linearized function $H_k$ may not be an accurate description of $h_k$ if we have not linearized about the correct point. In other words, if our estimate of the parameter vector is not quite right, the function $H_k$ will be slightly in error. In order to avoid this problem, we iterate. We first use the extrapolated value of the state estimate, which, for our case, is simply identical to the most recent estimate of

\(^9\) In general, $R_k$ is a covariance matrix describing a noise vector $v_k$, but in our case, since we have only a single measurement $l_{4_k}$, our measurement vector is a scalar, so the quantities $v_k$ and $R_k$ both become scalars in the present case.
the state that we have available, as shown in the definition of \( \bar{x}_{k,0}(+) \). Then, as many times as necessary, we go through the steps of computing the Kalman gain matrix, the error covariance matrix, and a improved estimate of the state vector, which then allows us to come up with a better \( H_k \) to approximate \( h_k \). The state estimate after the \( i \)th iteration is denoted by \( \bar{x}_{k,i}(+) \). The iteration continues “as many times as necessary;” usually three or four repetitions are all that are necessary to converge to within a fraction of a percent. At that point, we can stop, and our state estimate based on all data up to the \( k \)th measurement is \( \bar{x}_{k,n}(+) \), where \( n \) is the number of iterations that were used. Similarly, we retain the final iterated value \( P_{k,i+1}(+) \) and use that as the error covariance matrix \( P_k \).

The Kalman filter has the pleasing property that the state estimate improves as the points come in one by one; it is not necessary to wait until all the data points we want to include in the analysis are available to begin improving our estimate of the state. This will prove helpful in the glitch removal process discussed in Chapter 5.

PROBLEM — NON-CONVERGING SOLUTIONS

Unfortunately, both the least-squares and the Kalman filter approaches ran into snags when actually implemented. Files of simulated laser data were created\(^\text{10}\), and from these, the algorithms attempted to compute the values of the five angles describing the laser plate geometry. What we observed was that the algorithms never converged to any solution, and in particular, they did not converge to the values that were chosen during the creation of the simulated data. At first, if the estimates of the parameters were rather poor, the computer would move rapidly in the right direction, but then it would begin apparently aimless meanderings through the state space.

\(^{10}\) Values were chosen for each of the five angles in the laser plate model. Next, a file was created which contained random motions (\( x-y-z \)) of an imaginary bowling ball. Finally, based on these motions and the values of the five parameters, the ideal values of the laser readings were calculated, including the effect of the curvature of the hemisphere.
After a fair amount of grief, and after overcoming a large hurdle of disbelief, I hypothesized that the problem was overparameterized; perhaps it was not possible to solve for all five parameters based on just the laser data. Indeed, this turns out to be the case. We can solve for any three of the parameters simultaneously, but any group of four or five contains a non-observable combination, i.e., the operator $f$ has a nullspace that is not dimensionless, so that there are certain combinations of the parameters (not necessarily linear combinations, though, since $f$ is a nonlinear function) that produce no change in the ideal value of the measurement $l_{4a}$.

How was this revealed? Through the use of the singular value decomposition. Let's explore how this construction can facilitate understanding of the behavior of a system. Recall that any $M \times N$ matrix $A$ can be written as the product of an $M \times N$ column-orthogonal matrix $U$, an $N \times N$ diagonal matrix $W$ with positive or zero elements, and the transpose of an $N \times N$ orthogonal matrix $V$:

$$
\begin{pmatrix}
A
\end{pmatrix}
= 
\begin{pmatrix}
U
\end{pmatrix}
\cdot 
\begin{pmatrix}
w_1 & w_2 & \cdots & w_N
\end{pmatrix}
\cdot 
\begin{pmatrix}
V^T
\end{pmatrix}
$$

This decomposition is nearly unique; rows and columns can be permuted or, if two elements of $W$ are identical, you can form linear combinations of the corresponding columns of $U$ and $V$. There are reliable, numerically sound methods for computing the singular value decomposition available in the literature. We are particularly interested in the results of the SVD performed on a square matrix such as $A = L^T L$ that arose earlier when we derived the normal equations for the problem of determining the bowling ball

\begin{itemize}
\item Note that a geometrical mechanism for this singularity has not been studied in detail. It is not easy to readily visualize why certain parameters are not observable, and it hasn't yet been deemed worthwhile for anybody to spend a lot of time trying to visualize what's going on in three dimensions. However, the singular value decomposition contains the information needed to understand the details, if necessary.
\item Numerical Recipes, pp. 52–64
\end{itemize}
center. In order to say something about whether or not we can solve the system uniquely, we need to know whether this matrix $A$ is invertible.

If the matrix is not singular, then the system can be solved. But what if the matrix is close to singular? Suppose, for example, that the matrix were close enough to singular that the inversion could not actually be accomplished on a computer due to round-off, or that we are operating in a region close to this threshold. How can we quantify how close the matrix is to being singular? The elements $w_i$ of the matrix $W$ contain this information, and are known as the singular values of the matrix $A$: since $U$ and $V$ are both square orthonormal matrices in the case when $A$ is square, we can easily construct the inverse of $A$ as

$$A^{-1} = V \begin{pmatrix} \frac{1}{w_1} & & \\ & \frac{1}{w_2} & \\ & & \ddots \\ & & & \frac{1}{w_N} \end{pmatrix} U^T.$$  

We define the condition number of the matrix to be the ratio of the largest of the $w_i$'s to the smallest. In the case of a non-invertible (singular) matrix, at least one of the $w_i$'s will be zero, and the condition number will be infinite. Note that the condition number for the identity matrix is one, the minimum possible condition number, indicating that it is easily invertible. Most matrices of interest lie somewhere in between. We begin to run into problems computationally when the reciprocal of the condition number approaches the limits of precision of the machines floating point arithmetic (i.e., $10^{-6}$ for single precision or $10^{-12}$ for double precision arithmetic).

What does the SVD tell us in the case when the matrix $A$ is singular? For singular matrices, the concepts of nullspace and range are important. Recall that the nullspace of $A$ is defined as the set of all $x$ such that $Ax = 0$, while the range of $A$ is the set of all $b$ that can be "reached" by the mapping $Ax = b$, in the sense that there is some value of $x$ that maps to $b$. SVD, it turns out, explicitly constructs orthonormal bases for the nullspace and range of a matrix. As mentioned before, one or more of the singular values $w_i$ will
be zero for a singular matrix. The columns of $U$ corresponding to the nonzero elements of $W$ are a basis for the range of the matrix, while the columns of $V$ (i.e., the rows of $V^T$) corresponding to the singular values which are zero form an orthonormal basis for the nullspace of the matrix.

Now let's look at how we use this information to draw conclusions about the way the system works by first examining the case just mentioned in which we are trying to determine the sphere center (in $x$, $y$, and $z$) from the four laser readings. In order to uniquely determine the position of the center of the sphere from the laser readings, we require that the range of the matrix $A = L^T L$ have dimension three, meaning that the nullspace must have dimension zero, so we require that all of the singular values be nonzero. In fact, it is pretty clear that this is the case given the geometry of the problem, and that we will be able to uniquely specify the center of the sphere from the laser readings. Suppose, however, that all the laser beams were collinear. In this case, the range of the matrix $A$ would have dimension one, while the nullspace would have dimension two. This is the case where the nullspace constructed by the SVD gives us useful information. Recall that the nullspace is defined as the set of all vectors $x$ such that $Ax = 0$, in other words, the set of all vectors $x$ that are "invisible" as far as the output vector $b$ is concerned. If we start with some vector $x_0$ that gives us an output $b_0$, then, by linearity, we can add any vector in the nullspace of $A$ to $x_0$ and still have the same output $b_0$. In other words, the vectors that make up the basis for the nullspace represent linear combinations of the elements of $x$ to which the system is insensitive, at least as concerns observation of the outputs; i.e., there are parts of the system that are unobservable, and these vectors explicitly show where the unobservability lies.

Okay, now, this is all pretty silly when applied to the problem of determining the center of the sphere, since we'd be fools to design the laser plate such that all the lasers were collinear. However, in the case of using the redundant laser data to figure out the values of
the angles that are parameters in the laser plate model, we have postulated that there are singularities present, and we wish to determine exactly what it is that our observations are insensitive to; this is where a peek at the singular values and the nullspace constructed in the indicated columns of \( V \) will really help us out.

When preparing to perform the decomposition, we quickly remember that this is a nonlinear problem, and we don’t have a matrix \( L \) to use to construct \( A = L^T L \). However, recall from figure 17 that we can construct the matrix \( H_k(\hat{x}_k, (+)) \), which is a linear approximation to the nonlinear function \( h_k(x_k) \). Note, however, that \( H_k(\hat{x}_k, (+)) \) relates only one measurement to five unknowns. We need at least five measurements to be able to hope to solve for the five unknown angles, so we assemble the matrix

\[
H = \begin{pmatrix}
H_0(\hat{x}_0, (+)) \\
H_1(\hat{x}_1, (+)) \\
\vdots \\
H_N(\hat{x}_N, (+))
\end{pmatrix}.
\]

\( H \) is simply the result of combining the \( N \) individual equations involving the \( H_k \)'s into one large system of equations. Now, if we make the assumption that all the \( \hat{x}_i \) are equal, we have the form of a least squares problem:

\[
\begin{pmatrix}
l_{\hat{x}_0} \\
l_{\hat{x}_1} \\
\vdots \\
l_{\hat{x}_N}
\end{pmatrix} = \begin{pmatrix}
H_0(\hat{x}_0, (+)) \\
H_1(\hat{x}_1, (+)) \\
\vdots \\
H_N(\hat{x}_N, (+))
\end{pmatrix} \times \hat{x}.
\]

When we consider whether we can solve for all five components of \( \hat{x} \), we have to ask about the invertibility of the matrix \( A = H^T H \). We can perform a singular value decomposition of the matrix \( A \) and use the results to make statements about the singularity of the matrix and observability of the system in the region near \( \hat{x} \). Assuming that \( \hat{x} \) is a decent approximation to the true value of \( x_k \), the information we get from the singular value decomposition will be a decent approximation to the truth.

The singular value decomposition together with these other manipulations was used on simulated data sets like those mentioned earlier in an attempt to discover exactly what
we are and are not able to determine about the laser plate parameters from the laser data. These results are shown in Figures 18a–b.

******* Singular value decomposition for 5-parameter system *******

\[
U U U U U
\]

\[
-0.7388229 \quad 0.3166976 \quad -0.5370756 \quad 0.2557164 \quad 0.0015187
\]

\[
-0.0000067 \quad -0.8198789 \quad -0.5430652 \quad -0.1259844 \quad 0.1304096
\]

\[
0.6085448 \quad 0.4264229 \quad -0.6452795 \quad -0.1244043 \quad -0.1263908
\]

\[
0.2528508 \quad 0.0178202 \quad -0.0155777 \quad 0.6716226 \quad 0.6960096
\]

\[
-0.1409980 \quad 0.2129485 \quad 0.0013263 \quad -0.6724460 \quad 0.6946849
\]

\[
V V V V V
\]

\[
-0.7388229 \quad 0.3166976 \quad -0.5370756 \quad 0.2557164 \quad 0.0015187
\]

\[
-0.0000067 \quad -0.8198789 \quad -0.5430652 \quad -0.1259844 \quad 0.1304096
\]

\[
0.6085448 \quad 0.4264229 \quad -0.6452795 \quad -0.1244043 \quad -0.1263908
\]

\[
0.2528508 \quad 0.0178202 \quad -0.0155777 \quad 0.6716226 \quad 0.6960096
\]

\[
-0.1409980 \quad 0.2129485 \quad 0.0013263 \quad -0.6724460 \quad 0.6946849
\]

\[
W W W W W
\]

\[
0.1971787E+11
\]

\[
0.1468843E+11
\]

\[
0.2794498E+10
\]

\[
824.3846
\]

\[
372.3205
\]

Figure 18a. Singular Value Decomposition for 5-parameter system.
***** Singular value decomposition for 3-parameter system *****

\[ U U \]
\[ U U = \]
\[ \begin{array}{cccc}
0.5543829 & -0.6191765 & -0.5561296 \\
0.5530018 & -0.2253241 & 0.8021334 \\
0.6219715 & 0.7522297 & -0.2174899
\end{array} \]

\[ V V \]
\[ V V = \]
\[ V \]
\[ \begin{array}{cccc}
0.5543829 & -0.6191765 & -0.5561296 \\
0.5530018 & -0.2253241 & 0.8021334 \\
0.6219715 & 0.7522297 & -0.2174899
\end{array} \]

\[ W W \]
\[ W W = \]
\[ \begin{array}{cccc}
0.3071642E+10 \\
0.1990621E+11 \\
0.1538969E+11
\end{array} \]

Figure 18b. Singular Value Decomposition for 3-parameter system.

Let's look first at the results of Figure 18b, the 3-parameter case. The condition number for the matrix is the quotient of the extremes of the singular values, or \(0.199E+11/0.307E+10 = 6.48\). In other words, the matrix can be inverted with good computational precision, so we can solve for the three parameters (\(\phi_1\), \(\phi_2\), and \(\phi_3\), in this case). On the other hand, a glance at Figure 18a reveals that the condition number for that matrix is \(0.197E+11/372 = 5.30E+7\). The reciprocal of this number is less than \(2 \times 10^{-8}\), which is effectively zero for single precision arithmetic; the simulated data that was used
to generate these results has less precision than that, indicating that the last two singular values (824.3846 and 372.3205) are both effectively zero, indicating that the matrix is not invertible. The columns of $\mathbf{V}$ corresponding to these values are a basis for the nullspace of the matrix, meaning that they represent linear combinations of the parameters which have no effect on our measurements. Let's look at the 5th column, normalizing the entries:

$$
\begin{pmatrix}
0.0015187 \\
0.1304096 \\
-0.1263908 \\
0.6960096 \\
0.6946849
\end{pmatrix} \rightarrow \text{normalize} \rightarrow
\begin{pmatrix}
0 \\
1 \\
-1 \\
3 \\
3
\end{pmatrix} \leftrightarrow
\begin{pmatrix}
\Delta \phi_1 \\
\Delta \phi_2 \\
\Delta \phi_3 \\
\Delta \theta_2 \\
\Delta \theta_3
\end{pmatrix}.
$$

This last result is interpretable physically. It says that if $\theta_2$ and $\theta_3$ both increase by three units (a unit being whatever you like, as long as it is small enough that our linear approximation still holds), while $\phi_2$ increases and $\phi_3$ decreases by one unit, the exact same laser readings would result. In other words, the same laser readings could be produced by another laser plate in a different orientation; in this case, that orientation is one in which the plate has been twisted slightly counterclockwise (seen from above) while the laser 2 corner has been pushed down and the laser 3 corner has been pulled up. Thus the singular value decomposition has revealed the unobservability in the model that was very difficult to discover intuitively.
CHAPTER 4: CORRECTING FOR FRINGE POSITION

In the previous chapter, we discovered how to compute the position of the bowling ball at any time given a set of laser readings and the appropriate geometric model. Now it comes time to relate bowling ball motion to fringe motion. The bowling ball is firmly attached to the back of the siderostat mirror; the center of curvature of the bowling ball is supposed to coincide with the surface of the siderostat mirror, and it is supposed to be approximately at the idealized "pivot point" of the siderostat, so that, ideally, the center of the bowling ball will never move (the techniques for mounting the bowling ball in this manner are not crucial to the present discussion, and therefore will not be addressed).

If the center of curvature of the bowling ball did coincide exactly with the pivot point of the siderostat, then coming up with corrections to the fringe position based on pivot point motion would not be that hard. The correction to the fringe position is simply

\[ \Delta x = \hat{S} \cdot \Delta B + \Delta C, \]

this being simply the incremental form of the original fringe position equation. It is clear from the linearity of the above equation and from the geometric definitions of \( B \) and \( C \) that their incremental counterparts can be expressed as

\[ \Delta B = \Delta B_N - \Delta B_S \]
\[ \Delta C = \Delta C_N - \Delta C_S \]

If the we use \( D \) to denote the motion of the pivot point, then, since the baseline is simply defined as the vector connecting the two pivot points, we quickly see that

\[ \Delta B_i = D_i, \]

where \( i \) represents one of \( N, S, \) or \( E \). The change in the constant term that results from motion of the pivot point is simply a reflection of the motion of the pivot point in the direction of the feedbeam. The feedbeam is the direction that the starlight takes as it
travels from the siderostat into the main interferometer optics; we'll refer to the unit vector in the direction of the feedbeam as $\hat{F}$. Mathematically speaking, then, we see that the change in the constant term is simply

$$\Delta C_i = P_i \cdot \hat{F}_i.$$ 

Ah, that life were so simple, though. Unfortunately, the center of curvature of the bowling ball does not coincide with any well-defined pivot point of the siderostat, but what we really need to measure in order to make fringe position corrections is the motion of such a pivot point. It seems at first like we're stuck, but a powerful result of subtle simplicity saves us. Mike Shao has shown\textsuperscript{13} that not just the pivot point itself, but any point on the surface of the siderostat mirror can be selected as a pseudo-pivot point, and this pseudo-pivot point will also satisfy the basic fringe position equation. In other words, we can choose any point on the mirror surface that we like to be the tip of the baseline vector, even though that point may be moving around with time.

The prove of this lemma is subtly simple. We first observe that, if we are tracking a fringe, then the basic fringe position equation, $x = \hat{S} \cdot B + C$, must be satisfied. Furthermore, if the path lengths are equal at one point in the reference plane (which is the main beamsplitter that combines the beams), then they are equal at every point in that plane; otherwise, we would not be able to detect and track the fringe. See Figure 19 for a pictorial depiction of this property.

Now, since the path lengths match at every point in the plane, the fringe position equation must be satisfied at every point in the plane. If the fringe position equation is satisfied at every point in the plane, then we can choose any ray we like as our "reference" ray and define the pseudo-pivot point as the point where that ray strikes the siderostat

\textsuperscript{13} Shao et al, "Preliminary Proposal to the Office of Naval Research for Research on Phase-Coherent Astrometric Interferometry."
Figure 19. Path Length is Equal at All Points in Reference Plane (dashed beamsplitter).
mirror. Or conversely, we can choose any point on the surface of the siderostat mirror that we like as the pseudo-pivot, and that will define our reference ray.

How is it that we can just pick any two arbitrary points on the siderostats to define the baseline? Isn’t the baseline a fixed physical quantity? The answer is that we often think of the baseline as a physical entity because it is a good conceptual tool. But down at the 1 micron level, the baseline is not a constant, as the siderostats wobble on their bearings, and that’s why we’ve sought to supply baseline corrections with the laser metrology system in the first place. Expanding on this idea, we can choose any point on the mirror surface we like as the tip of the baseline as long as we keep track of (and supply corrections for) the changes in fringe position caused as the tip of the baseline moves.

So we have shown that we can pick any point on the surface of the siderostat mirror and call that point the “pivot point,” as long as we stick with that point. We still don’t have any way of monitoring the position of a point on the mirror’s surface, but we can at least make an intelligent choice about some point we’d like to monitor. Let’s set up the geometry of the problem; refer to Figure 15 for aid in visualizing this. Let’s start out by denoting the vector describing the position of the center of curvature of the bowling ball as \( \mathbf{R} \). The origin, denoted by \( \mathbf{O} \) in the figure, is simply defined as \( \mathbf{R} \) at time zero. Then, let’s pick a point on the surface on the mirror that is as close as possible to the center of curvature of the bowling ball, and we’ll call that the pivot point, whose position is described by the vector \( \mathbf{D} \). The distance from the bowling ball center to the surface of the mirror will be denoted by \( d \). Note that \( d \) can be a negative number in the event that the center of the bowling ball lies above the surface of the mirror. Then, since \( d \) is the shortest distance to the mirror surface, that means that \( \mathbf{D} \) and \( \mathbf{R} \) are displaced from one another by the distance \( d \) in the direction of the mirror normal. In other words, if we denote the unit vector in the direction normal to the mirror’s surface by \( \hat{\mathbf{m}} \), then we can
relate these various quantities:

\[ \mathbf{D} = \mathbf{R} + d\mathbf{\hat{m}} \]

In the figure, the initial position of the bowling ball at time zero is shown in bold outline; at this time, the bowling ball center is at \( \mathbf{R}_0 = \mathbf{O} \), and the pseudo-pivot point is at \( \mathbf{D}_0 = d\mathbf{\hat{m}} \). At some later time (let's call it time 1), suppose the assembly has moved to the position indicated by the faint outline. The vector \( \mathbf{R}_1 \) now describes the position of the bowling ball center, while \( \mathbf{D}_1 = \mathbf{R}_1 + d\mathbf{\hat{m}} \) is the location of the pseudo-pivot point.

One might logically wonder whether all this rigmarole has made life simpler at all, since it seems that all we've done is introduce a number of unknown quantities, but let's examine the above expression for pivot point location \( \mathbf{D} \) in detail. First, we have \( \mathbf{R} \), the position of the center of curvature of the bowling ball; this is measured to a very high degree of accuracy (5 nm) by the laser interferometers. We also have \( \mathbf{\hat{m}} \), the mirror normal. At first it might seem that we don't have any information about the mirror normal. However, we note that the only times we need to know \( \mathbf{D} \) are during the times that we are tracking fringes, since that is the only time we'll have to make corrections to the fringe position. And we also note that, whenever we're tracking fringes, the siderostats of necessity must be reflecting the star's light directly along the feedbeam. In other words, the siderostat mirror normal is constrained by its function. We'll denoted the direction to the star's apparent (refracted) position by \( \mathbf{\hat{S}} \), and recalling that the feedbeam direction is \( \mathbf{\hat{F}} \), we can find

\[ \mathbf{\hat{m}}_c = \frac{\mathbf{\hat{S}} + \mathbf{\hat{F}}}{|\mathbf{\hat{S}} + \mathbf{\hat{F}}|} \]

The last thing we need to know in order to find \( \mathbf{D}_c \) is the distance \( d_c \) for each siderostat. Unfortunately, this distance is not known, but we will be able to solve for it later using fringe data, so that gives us everything we need to know to find the pivot point motion.

Once we have the pivot point motion, there are ways that I've already described for converting that into changes in fringe position. Let's thresh through the equations for
the case of the North-South baseline.

\[ \Delta x = \hat{S} \cdot \Delta B + \Delta C \]

\[ = \hat{S} \cdot (D_N - D_s) + \hat{F}_N \cdot D_N - \hat{F}_S \cdot D_s \]

\[ = \hat{S} \cdot (R_N + d_N \hat{m}_N - R_s - d_s \hat{m}_s) + \hat{F}_N \cdot (R_N + d_N \hat{m}_N) - \hat{F}_S \cdot (R_s + d_s \hat{m}_s) \]

\[ = (\hat{S} + \hat{F}_N) \cdot R_N - (\hat{S} + \hat{F}_S) \cdot R_s + d_N (\hat{S} + \hat{F}_N) \cdot \hat{m}_N - d_s (\hat{S} + \hat{F}_S) \cdot \hat{m}_s \]

\[ = (\hat{S} + \hat{F}_N) \cdot R_N - (\hat{S} + \hat{F}_S) \cdot R_s + (d_N - d_s) (\hat{S} + \hat{F}_N) \cdot \hat{m}_N - d_s [(\hat{S} + \hat{F}_N) \cdot \hat{m}_N - (\hat{S} + \hat{F}_S) \cdot \hat{m}_s] \]

Before examining the final form, let's look at the next-to-last expression above and talk about how we would use it. In order to do this, I need to digress for a moment and talk about data analysis for the interferometer in general. Remember that knowledge of both the baseline \( B \) and the constant term \( C \) is necessary to perform astrometry. The baseline and constant term can be measured only to a rather coarse level of accuracy, though, so what we have to do is solve for the baseline and constant term using fringe data from stars whose positions are known. Since there are really four unknown parameters (three for the baseline and one for the constant term), we would need at least four fringe measurements to be able to compute these quantities. With any more than four measurements, we can use the redundant information to find the best fit to the parameters in the least-squares sense. Now, if we generalize the basic fringe position equation to include corrections due to the laser metrology system, we get

\[ x_{\text{measured}} = \hat{S} \cdot B + C + \Delta x. \]

What has happened now? Well, in addition to the unknowns in \( B \) and \( C \), we have added (through the addition of \( \Delta x \)) the unknowns \( d_N \) and \( d_s \) (for the case of the North-South baseline); everything else needed to compute \( \Delta x \) (and therefore \( x \)) is known. Note that, happily, the fringe position \( x \) is linearly related to the bowling ball offsets \( d \). Hence, we
can simply add $d_N$ and $d_S$ to the pile of unknowns that we would like to solve for using fringe data. Now, instead of requiring at least four measurements to determine all the unknowns, we will need at least six, since we've added two unknowns.

Now, let's have a look at the final form of the expression that I have above, and let's consider the observability of the parameters $d_N$ and $d_S$. I've rearranged the expression above in terms of parameters $d_N - d_S$ and $d_S$. Clearly we can recover the original quantities $d_N$ and $d_S$ from these modified parameters. Now we're ready to look at the observability of the system. The term proportional to $d_N - d_S$ will take on a wide variety of values as we move through different parts of the sky, and its effect will be seen in the measured fringe position. On the other hand, the term proportional to $d_S$ will always be very close to zero, since, for a perfectly put-together interferometer, both the feedbeams and hence the siderostat mirror normals will always be identical. Thus the two terms multiplying $d_S$ are of nearly equal magnitude, but since they are opposite in sign, they cancel each other. So we see that our observed fringe position is highly insensitive to changes in $d_S$ as long as $d_N - d_S$ is held constant. In other words, $d_S$ is essentially unobservable. Or, in terms of our two original parameters $d_N$ and $d_S$, we cannot know both $d_N$ and $d_S$, but only their difference $d_N - d_S$. But that's okay, because that's all we need to know in order to make accurate corrections to the fringe position.

In practice, when we solve for the baseline and constant term, we also solve for $d_N - d_S$, and, if we feel it is important, we can also solve for $d_S$. In analyses performed to date, the magnitude of the least-squares value for $d_S$ has always been much less than the uncertainty (in terms of covariance) of $d_S$.

Note that the sign of the constant term is arbitrarily defined, and it could be that an increase in the path length of the north arm of the interferometer could decrease the fringe position. In this case (which is, in fact, the case in the Mark III as presently configured), the fringe position expressions boil down to
\[ \Delta z = \hat{S} \cdot \Delta B - \Delta C \]
\[ = \hat{S} \cdot (D_N - D_s) - \hat{F}_N \cdot D_N + \hat{F}_S \cdot D_s \]
\[ = \hat{S} \cdot (R_N + d_N \hat{m}_N - R_s - d_s \hat{m}_s) - \hat{F}_N \cdot (R_N + d_N \hat{m}_N) + \hat{F}_S \cdot (R_s + d_s \hat{m}_s) \]
\[ = (\hat{S} - \hat{F}_N) \cdot R_N - (\hat{S} - \hat{F}_S) \cdot R_s \]
\[ + d_N (\hat{S} - \hat{F}_N) \cdot \hat{m}_N - d_s (\hat{S} - \hat{F}_S) \cdot \hat{m}_s \]
\[ = (\hat{S} - \hat{F}_N) \cdot R_N - (\hat{S} - \hat{F}_S) \cdot R_s + (d_N - d_s)(\hat{S} - \hat{F}_N) \cdot \hat{m}_N \]
\[ - d_s [(\hat{S} - \hat{F}_N) \cdot \hat{m}_N - (\hat{S} - \hat{F}_S) \cdot \hat{m}_s] \]

And, plugging in for \( \hat{m} \), we get, for the present configuration,
\[ \Delta z = (\hat{S} - \hat{F}_N) \cdot R_N - (\hat{S} - \hat{F}_S) \cdot R_s + (d_N - d_s)(\hat{S} - \hat{F}_N) \cdot \frac{\hat{S}_r + \hat{F}_N}{|\hat{S}_r + \hat{F}_N|} \]
\[ - d_s [(\hat{S} - \hat{F}_N) \cdot \frac{\hat{S}_r + \hat{F}_N}{|\hat{S}_r + \hat{F}_N|} - (\hat{S} - \hat{F}_S) \cdot \frac{\hat{S}_r + \hat{F}_S}{|\hat{S}_r + \hat{F}_S|}] \]

or, if the sign of the constant term flips,
\[ \Delta z = (\hat{S} + \hat{F}_N) \cdot R_N - (\hat{S} + \hat{F}_S) \cdot R_s + (d_N - d_s)(\hat{S} + \hat{F}_N) \cdot \frac{\hat{S}_r + \hat{F}_N}{|\hat{S}_r + \hat{F}_N|} \]
\[ - d_s [(\hat{S} + \hat{F}_N) \cdot \frac{\hat{S}_r + \hat{F}_N}{|\hat{S}_r + \hat{F}_N|} - (\hat{S} + \hat{F}_S) \cdot \frac{\hat{S}_r + \hat{F}_S}{|\hat{S}_r + \hat{F}_S|}] . \]

Now the mathematics of what we're going to do is all laid out, but how do we go about using this in practice? This is the topic of the Chapter 6.
CHAPTER 5 : GLITCHES AND DEGLITCHING

Well, you've heard rumors and you've heard tales — but now, at long last, here comes the truth about glitches. The data recorded from the laser fringe counters, although it is discrete in time, is normally fairly smooth and continuous when viewed graphically, for example. However, as alluded to earlier in this thesis, there are occasions when the recorded data exhibits abrupt discontinuities which clearly don't represent physical reality, as in figure 20. Figure 20a is a plot of the four laser readings from the north laser plate for an entire data set, about 2.4 hours worth of data.

The “discontinuities” that are immediately obvious in this figure are not discontinuities at all, but rather they are points in time when the siderostats were slewing very rapidly to a new star. The flat spots on the plot represent time that the siderostats spent tracking a star. Figure 20b is a blowup of the region around time 22475 (the cryptic units used for time have a reason, but suffice it to say for now that each unit of time represents two seconds).

Laser 4 is the bottom trace in this figure, and in the figure, laser 4 has two glitches, one at time 22471 and another at time 22481. Note, however, that the hitch in the data at time 22463 is not a glitch.

There are a number of possible explanations for why this happens, all of which involve the interruption of the signal out at the laser plate. This is very bad. The accuracies we need in our measurements are on the order of a fraction of a wavelength of light; as can be seen from the figure, the glitches are always at least one full wavelength in magnitude, and often they can be significantly larger. Our only recourse is to attempt to correct the glitches by figuring out what the data should have been and insert the correct value.

Unfortunately, again, figuring out what the data should have been is not a very easy task. It would be nice if the data varied very slowly with respect to the sampling rate, so
Figure 20a. North pier laser data. Total time 2.4 hours.
North Lasers

Data Set FT116F1

Figure 20b. Glitches in north pier laser data. Total time 100 seconds.
that we’d be able to correct the glitch by just duplicating the previous value of the data, or by performing some simple extrapolation. However, this is not the case. The data is sampled only every two seconds, and, particularly when the siderostats are slewing, the laser readings can change by an amount greater in magnitude than many of the glitches that are encountered, making any sort of extrapolation virtually impossible.\textsuperscript{14} Since it is effectively impossible to correct a glitch based on just the data from that laser, we are led to explore possibilities for using the redundant information present (recall that we have four laser beams, but only three measurements are required to constrain the bowling ball’s position) to give us insight into how to correct the glitch, or to try to somehow quantify the nature of the glitch so that we have some a priori expectation of what the magnitude of the glitch might be.

Let’s talk about the second of these things first, trying to say something about the character of glitches that will be of benefit to us in removing them. This naturally leads us to examine the physical mechanism that causes the glitches.

Glitches could occur because of problems in the digital laser fringe counters described in Chapter 2, but if that were the case, we could simply redesign it to perform correctly, and, in fact, it has been determined that the digital hardware does work properly. Therefore, the problem must occur in processing prior to that step. The communication medium between the piers and the digital electronics does not introduce appreciable dis-

\textsuperscript{14} If we were in possession of a good model for the motion of the bowling ball, we would be in a position to perform accurate extrapolations of the data. While this is not the case in the situation under consideration, it is possible that we could develop a model that is informative after having solved the problem once. For after having successfully analyzed a night’s worth of data, we could extract information concerning the distance of the bowling ball center from the pivot point of the siderostat. And, while there is a well-defined pivot point only down to the level of a few microns, the motion of this “pivot point” due to irregularities in the bearings, etc., is much less than the tens of microns of motion due to the offset of the bowling ball center from the pivot point. Knowing this distance, we could account for and predict the systematic variations in the laser readings due to this offset; we would then perhaps be able to make extrapolations that would be accurate enough to effectively remove the glitch.
tor}tio}n of the signal, so we are led back to the analog postamps and preamps. Here, we begin to see potential for problems. For purposes of noise immunity, these amplifiers were designed with rather fast rollo}ffs at 5 kHz and 30 kHz. If the bowling ball is not moving, the signals processed by these amplifiers are at 20 kHz. If, however, the bowling ball were to move toward the amplifier at a rate of 5000 fringes per second, for example, then the laser light would be intensity modulated at 25 kHz due to the Doppler effect. Carrying this further, we can see that, if the bowling ball were to move to abruptly, the laser signals could be Doppler shifted into the part of the spectrum rejected by the amplifiers, and hence the signal would cease to appear at the output of the postamps. While we did not originally anticipate that the bowling ball would move at anything approaching the rate where Doppler shifting becomes a problem, we later become concerned with the acoustic effects of vibrations and resonances in the siderostat motors as they slew.

Glitches could also be caused simply by a speck of dust on the hemispherical mirror that temporarily interrupts the beam, causing the signal to temporarily drop out. At present, it is believed that the main cause of the glitches is either Doppler shifting or beam dropouts due to dirt; tests must be performed in the future to determine which of these is the culprit so that it can be eliminated. My belief is that many, if not all, of the glitches are due to Doppler shifting when the siderostat vibrates as it slews, especially during acceleration and deceleration. The reason for saying this is that all the glitches that I've looked closely at have occurred while the siderostats were slewing.

So now what we know about glitches is that they occur in units of 64 and that they tend to occur when the siderostats are slewing. This does not bode well for making the glitches easy to detect (let alone correct) just from looking at the laser data. The laser readings are recorded every two seconds; during siderostat slewing, these readings routinely change by hundreds of counts, making it nearly impossible to perceive (either by eye or by computer program) 64 counts missing here or there. Indeed, the fact that the
glitches are more likely to occur during acceleration of the siderostat makes things even tougher, since we can’t even do accurate extrapolation. The glitches shown in Figure 20 are by far the exception and not the rule; they were chosen for ease of viewing.

Since we weren’t able to make too much headway based on the character of the glitches, we can call on our big gun, which is the redundant fourth laser reading. The idea is simple; the result of determining the center of the sphere from any three of the lasers should ideally be the same as the result obtained from using any other combination of three lasers. Actually, the way we use this information is to generate an error of fit that represents how well the four lasers agree on where the bowling ball center is; if, at some point in time, they suddenly disagree violently, this error of fit will shoot way up, indicating that a glitch has probably occurred. The least-squares method for computing the center of the sphere was delineated in Chapter 3; what we want to recall from that discussion is that we can formulate the problem as attempt to find the value for the center of the sphere that will minimize the quantity

$$\text{Error} = \sqrt{\sum_{i=1}^{4} (l_{i}^{\text{ideal}} - l_{i}^{\text{actual}})^2}.$$  

In this case, what we’re interested in not the computed sphere center, but rather in Error (this is a root-sum-square error figure), which indicates how well the four lasers agree. A perfect agreement would give Error $= 0$, while a definite glitch would give errors in the hundreds. This scheme was implemented, and the results are depicted in Figures 21a–c. The horizontal axis is time in units of 12 seconds, while the vertical axis is in units of $\lambda/64$.

One of the first things one notices about this data is that the plots are distinctly different for each pier. Note that the scale for the north pier is ten times the other two; the north laser data contains some definite large glitches, and it likely contains a number of small ones, too. The south pier takes the other extreme, being very well behaved. The data between the points labelled 15 and 35 on the plot is as beautiful as it gets — the error
Figure 21a. Root Sum Square error plot for north pier. Time vs. Error. Total time 2.4 hours.
Figure 21b. Root Sum Square error plot for south pier. Time vs. Error. Total time 2.4 hours.
Figure 21c. Root Sum Square error plot for east pier. Time vs. Error. Total time 2.4 hours.
doesn’t get too large, and it frequently flirts with zero. Experience has shown that the latter is the most distinctive feature of glitch-free data, that the error occasionally goes to zero. The south pier data later in the night doesn’t look quite so wonderful, but it’s still pretty good; it bears some resemblance to the east pier data. The east pier error starts out near zero, then just slowly wanders away in a series of steps. Some of the steps here are almost too small to be glitches; if they are glitches, they are no more than 64 units.

Let’s talk about where this “error” comes from, from the perspective of wanting to interpret the east error plot. Remember that the error is the error of fit that occurs when you try to “make” the four lasers agree on one x-y-z position for the bowling ball center. What if somebody went in and moved one of the laser beams without telling us? That would change the direction of motion that the laser is sensitive to. But, remember that, in order to do the least-squares fit in the first place, we had to have laser beam orientation information in the form of a laser plate model. Now, if our knowledge of the orientation of one of the beams were incorrect, our error of fit would be larger, because that laser would be giving us readings that were slightly different from what we’d expect it should give us based on the other three. Note that, at the origin, when all four laser readings are zero, the error of fit is zero, even with an incorrect laser plate model, but, as we move farther and farther from the origin, the absolute magnitude of the laser readings starts to get larger, so the effect of an inaccurate model becomes more pronounced.

We might hypothesize, therefore, that the effect of an inaccurate laser plate model would be to cause the error of fit to increase whenever the laser readings are large in absolute value, with the error returning to zero as the lasers return to zero. In fact, a quick glance reveals that the ups and downs of the error in Figure 21c are somewhat correlated with the points of large excursion by the siderostat visible in Figure 20a.

One can begin to see that the problem of determining what is or is not a glitch can be rather difficult. In particular, glitches that are only 64 units in size generate errors
that are often not much more than the noise, making them very difficult to see in either the error plot or the raw laser data. The followup problem of deciding what correction to make to a glitch that has been found is not much easier. The simplest strategy would be to simply look at the laser data and try to extrapolate what should have happened, but for reasons discussed above, that just doesn’t work. A more intelligent approach involves hypothesizing some correction and seeing what effect that has on the error. The difficulty here lies in trying to guess which laser glitched. Since the siderostat takes on the order of minutes to slew from one side of the sky to the other, the first few laser readings after a glitch are taken with the siderostat in an orientation very similar to the orientation when the glitch occurred; this is not so great. One thing we’d hope for in trying to correct the glitch is that, if we tried correcting the wrong laser, it would make the error increase. However, since the laser readings often don’t change very fast, the next few points after a glitch are very similar to the point at which the glitch occurred, so the errors for all those points will be roughly similar to the error for the glitch point, both before and after glitch correction, regardless of which laser we choose to correct. However, this seemingly benign behavior can kill you when the siderostat later switches to a different part of the sky. This task of determining which laser glitched is really the most uncertain part of the problem for glitches that are small in size. For larger glitches, after being tipped off by the error plot, it is not hard to look at the raw laser data and readily tell which laser glitched. For the other cases, however, there is no obvious recourse to watching the error unfold over time. The problem, of course, is that, if you have to wait too long, another glitch could occur before you really decide whether or not the first correction was right.

The program DEGLITCH (see Appendix) implements this basic strategy for glitch detection and correction, with a look-ahead of about eight points when it sees a glitch. We still need to talk about how to determine how big of a correction to make (e.g., 64 or 128 or 192, etc.). What we want to do is to use the three good lasers to compute the position

76
of the center of the sphere, and then use that information to compute the anticipated ideal value of the laser that glitched. We can then compare the ideal value to the measured value; the difference is the amount that the laser glitched. Now we are fortunate in that the glitches are characterized by the property that they always occur in units of 64, so that we hope to be able to recover from the glitch exactly. For example, if we computed that the ideal value of the laser is −3214 and the measured value is −3350, the difference is 136. We then assume that this is the result of a glitch of size 128 with a little bit of noise, so we correct the glitch by adding back 128 to −3350 to get −3222, which would be our new corrected laser reading. Again, life is not quite so simple for a number of reasons; for example, if the laser plate model is not especially good, then it is possible that the computed “ideal” value for the glitchy laser will be inaccurate.

Due to the fact that, for practical reasons, DEGLITCH only has a look-ahead of about eight points, it is often not capable of discerning which laser glitched, or, often, if there even was a glitch. Also, the reader may have noticed that the description I gave above of good, glitch-free data was rather subjective (i.e., is this graph pleasing to the human eye?). For this reason, DEGLITCH has not proven extremely successful at producing laser files which have acceptable error metrics, often “correcting” glitches where there are none if the sensitivity is too high, or missing alot of the little real glitches if the sensitivity is too low. Clearly, we were in need of more high-powered signal processing. Therefore, based on the premise that the world’s most sophisticated signal processor lies between a pair of human ears, we decided to implement what I’ll call the “eyeball method.”

The eyeball method follows essentially the same sequence of steps as the automated technique, but takes a much more global approach. The steps involved are as follows:

1) Produce and look at the root-sum-square error plot. Starting at the beginning of the data set, identify the first potential glitch. This will be some place where the error either jumps suddenly or after which the error never returns to zero.
2) Look at the raw laser data (both numeric and graphical forms of this data are useful) in the vicinity of the suspected glitch, examining it for discontinuities or irregularities. Often, plots of the raw data show that all four lasers follow similar curves, and a glitch can be detected as a little hitch in one curve where there is none in the other lasers. Hypothesize a time at which the glitch occurred and which laser glitched, and make a guess as to the amount of the glitch.

3) Remove your hypothetical glitch, then create and examine error plots for your new "possibly corrected" laser data. If the glitch is gone, return to step 1 and repeat until all glitches have been removed.

4) If the glitch is not gone, perhaps it has changed in character slightly. You can try a slightly different glitch size and repeat step 3 until it goes away. Or, less likely to help, you can try hypothesizing that a different laser glitched. Or perhaps there was no glitch at all, and this point in time should be left unmolested.

The programs which I used as an aid to performing the eyeball technique are covered in the next chapter. However, the results of applying this method are pertinent to the present discussion and are shown in Figures 22a–c. The glitches that were removed are enumerated in Table 1.
Table 1. Glitches removed from data set FT116F1.

<table>
<thead>
<tr>
<th>Time</th>
<th>Pier</th>
<th>Laser</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>19969</td>
<td>East</td>
<td>1</td>
<td>+128</td>
</tr>
<tr>
<td>20060</td>
<td>North</td>
<td>4</td>
<td>-256</td>
</tr>
<tr>
<td>20126</td>
<td>North</td>
<td>1</td>
<td>+64</td>
</tr>
<tr>
<td>20140</td>
<td>East</td>
<td>3</td>
<td>-64</td>
</tr>
<tr>
<td>20275</td>
<td>North</td>
<td>3</td>
<td>+64</td>
</tr>
<tr>
<td>20739</td>
<td>East</td>
<td>1</td>
<td>+64</td>
</tr>
<tr>
<td>20767</td>
<td>North</td>
<td>4</td>
<td>-640</td>
</tr>
<tr>
<td>20840</td>
<td>North</td>
<td>2</td>
<td>+512</td>
</tr>
<tr>
<td>21783</td>
<td>North</td>
<td>1</td>
<td>-192</td>
</tr>
<tr>
<td>21981</td>
<td>North</td>
<td>2</td>
<td>+448</td>
</tr>
<tr>
<td>22471</td>
<td>North</td>
<td>4</td>
<td>-192</td>
</tr>
<tr>
<td>22481</td>
<td>North</td>
<td>4</td>
<td>-1472</td>
</tr>
<tr>
<td>22972</td>
<td>North</td>
<td>2</td>
<td>+448</td>
</tr>
<tr>
<td>23514</td>
<td>South</td>
<td>3</td>
<td>+320</td>
</tr>
<tr>
<td>23582</td>
<td>North</td>
<td>2</td>
<td>+192</td>
</tr>
<tr>
<td>23646</td>
<td>East</td>
<td>3</td>
<td>+128</td>
</tr>
</tbody>
</table>
Figure 22a. Root Sum Square error plot for north pier after deglitching. Time vs. Error.
Figure 22b. Root Sum Square error plot for north pier after deglitching. Time vs. Error.
Figure 22c. Root Sum Square error plot for north pier after deglitching. Time vs. Error.
CHAPTER 6: SOFTWARE STRUCTURE AND DESIGN

A data flowgraph for the astrometry system is depicted in Figure 23.

Producing astrometric measurement begins in the upper left corner, with the recording of a night’s worth of data. The raw data is stored by a Data General Nova on a 9-inch magnetic tape; later this data is transferred to an IBM PC for processing. Also, at the end of the night, when the siderostat control computer powers down, the feedbeam vectors for each of the three siderostats are written out to a file with the extension .FDB. For example, for data set FT116F1, the feedbeam file is called FT116F1.FDB. Hereafter, I’ll use the notation *.FDB to represent the feedbeam file. These feedbeam vectors are the \( \hat{F} \) vectors used in the computations of Chapter 4; the feedbeam file is used in the calculation of corrections to the fringe position in CALC.

At the end of the night’s observing, the laser plate should be measured, as shown in the upper right corner of the diagram. This measurement is under the control of the siderostat control computer, and since knowledge of the siderostat pointing model is necessary to get an accurate laser plate model, this measurement is best done at the end of the night when the model is accurate. In practice, the laser plate model does not drift much from day to day, so this process needs to be performed only every week or so at most.

Now we need to crunch on the raw data to get out the pieces we want. The program NNXT on the IBM PC takes the *.RAW raw data file as input and produces several output files; the ones of interest to us are the *.AVG file, which contains fringe position information, and the *.LRW file, which contains the raw laser data recorded for each laser at two-second intervals. The *.AVG file will not be used for a little while; first we have to massage the laser data.

The first thing that we have to do to the laser data to make it useful at all is remove
Observe all night

Mag Tape

"Cook" raw data (NNXT.FOR)

Raw Laser Data

Deglitch Laser Data

Average

Compute Laser Plate Model

LPLATE.DAT model file

*.LGF file

*.LAS file

*.LMD file

*.LPT file

Human Reads This

*.AVG Fringe Data file

Compute theoretical ideal values for fringe position and compare to measured; generate residual information (program CALC).

Convert to xyz motion of bowling ball

*.PVT file

Use residual information to solve for baseline and for constants d. Compute final residuals. (Program SOLVE). Can also calculate star positions.

Figure 23. System Data Flow.
any glitches. The program DEGLITCH, as described in the previous chapter, has been designed to do this, but it does not work reliably. Instead, the eyeball method has been implemented using two new programs, FINDGL, which generates the root-sum-square error plots for viewing by the QPLOT plotting routine, and RMGLITCH, which allows the user to remove a single glitch from a laser data file. Note that the measured laser plate model is needed by FINDGL to generate the error plots. Lotus 1-2-3 has also proven useful in deciding what is and isn't a glitch. The end result of all this deglitching is a glitch-free laser data file *.LGF which is identical in format to the original *.LRW laser data file.

Once we have unadulterated laser data, we run the program LGF2LAS to average the data so that we have just one set of laser readings for any given observation of any given star. It is valid to do this averaging since the siderostat barely moves at all during the time when fringe tracking is going on. The result of this averaging is the *.LAS file.

These refined laser readings are now ready to be used. First, we use the information present in the redundant fourth laser to improve our estimate of the laser plate model using FITMOD. FITMOD reads in the measured laser plate model LPLATE.DAT together with the *.LAS laser file and computes a better laser plate model using the least-squares approach described in Chapter 3. FITMOD only improves the estimates of the three elevation angles \( \phi_i \), since we can only solve for three of the five laser plate angles. The values of the five laser plate angles in human-readable form are sent to a file called *.LMD (for Laser plate MoDel file). Also produced is a set of unit vectors derived from the laser plate parameters; these unit vectors describe the trajectory of each laser beam. This unit vector file is called *.LPT, and it is in exactly the same format as the LPLATE.DAT file. If, for whatever reason, the operator feels that the computed laser plate model may be inaccurate (due to undetected glitches, for example), this model-fitting step may be skipped by simply copying LPLATE.DAT to *.LPT.

Now, with an accurate laser plate model, it is quick work to use CONVXYZ to
convert the *.LAS laser readings to bowling ball motion using the least-squares technique described in Chapter 3. This motion is recorded in an output file *.PVT, which contains one x-y-z bowling ball position for each scan (a scan corresponds to one observation of one star).

Now we are ready to compute and apply the fringe position corrections developed in Chapter 4. George Kaplan of USNO has written a pair of programs for processing the bulk data. First we run CALC, which computes theoretical ideal values for the fringe position, based upon star positions, an estimated baseline model, and the fringe position corrections computed by the laser metrology system. As shown in the diagram, the corrections due to the metrology system are actually calculated as CALC is running, through a set of subroutines that I have written. Note that the feedbeam file is required by CALC if you wish to use the metrology data, since the developments of Chapter 4 require this information. Once CALC has completed, we run SOLVE. SOLVE looks at the unaccounted-for errors in fringe position after CALC and uses them to solve for changes to various parameters (changes with respect to the estimated values fed into CALC), such as the baseline vector and, as promised earlier, the bowling-ball-to-mirror-surface offset distance $d_N - d_S$.

In addition to solving for these parameters, SOLVE provides sophisticated error-reporting information. The parameter of greatest interest to us as far as performance of the laser metrology system is the RMS error of fit to the fringe data. The idea is that the metrology system should reduce these residual errors if they are due to wobbling of the siderostats. This measure of performance applied to a real data set is the topic of the next chapter.
CHAPTER 7: EXPERIMENTAL DATA AND RESULTS

The goal of high-accuracy astrometry requires that all the different parts of the interferometer work well simultaneously; unfortunately such stellar performance is not an everyday event, and there are very few data sets that are even candidates for this metrology analysis. I have extensively studied the data set FT116F1 taken on the night of 16 October 1987, representing about 2.4 hours of observing. Even this data set has problems, though; there are spans of 20 minutes or more during the night with no fringe data because people had to go tweak this or that. But, for the most part, it seems to have the things we want — decent performance by the laser metrology system, proper calibration of the lasers at the outset, two-baseline operation, and a recently-measured laser plate model (measured October 4).

First, for comparison, I'll present the results achieved running the software without the laser data. The important numbers are the RMS errors listed — 5.46 for SN and 4.59 for ES.

Next we have a case where the deglitching was done in an automated way by the program DEGLITCH, with its sensitivity turned down so that it was not likely to detect false glitches. We see in this case that while the SE RMS error improved slightly to 4.35, the SN error basically doubled to 10.35. Why the disparity? Well, if we reflect back to Table 1, we recall that the eyeball method found alot more glitches in the north laser plate than in the others. The more glitches there are, the more likely it is for the automated deglitcher to fail to catch one of them, introducing error. So we postulate that there is error in the north lasers which would throw off the calculations for the SN baseline, while the SE baseline has fewer errors, and we would expect its errors to improve. Note also that the uncertainties in the computed values for \(d_N - d_S\) are greater than for the results in Tables 4 and 5.
SOLUTION FROM 13086 OBSERVATIONS SPANNING 0.1 DAYS
DATES 871016.064845 TO 871016.091205

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>BASE-</th>
<th>TIMESPAN</th>
<th>STAR(S)</th>
<th>NUM</th>
<th>VALUE</th>
<th>ERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ID</td>
<td>LINE(S)</td>
<td>ACTIVE</td>
<td></td>
<td>OBS'NS</td>
<td></td>
</tr>
<tr>
<td>BX</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6744</td>
<td>-6751165.4866</td>
<td>1.7868</td>
</tr>
<tr>
<td>BY</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6744</td>
<td>-30706.8833</td>
<td>0.9574</td>
</tr>
<tr>
<td>BZ</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6744</td>
<td>+9926904.6891</td>
<td>1.3536</td>
</tr>
<tr>
<td>C</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6744</td>
<td>-3124.6859</td>
<td>2.2724</td>
</tr>
<tr>
<td>BX</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6342</td>
<td>+1.7581</td>
<td>1.5912</td>
</tr>
<tr>
<td>BY</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6342</td>
<td>-84.3887</td>
<td>0.8848</td>
</tr>
<tr>
<td>BZ</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6342</td>
<td>-51.4900</td>
<td>1.1654</td>
</tr>
<tr>
<td>C</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6342</td>
<td>+172.3819</td>
<td>1.9894</td>
</tr>
</tbody>
</table>

SN POST-SOLUTION RESIDUALS
--------------------------
NUMBER OF OBS'NS = 6744
RMS OF RESIDUALS = 5.46
PEAK TO PEAK = 45.12

SE POST-SOLUTION RESIDUALS
--------------------------
NUMBER OF OBS'NS = 6342
RMS OF RESIDUALS = 4.59
PEAK TO PEAK = 32.51

Table 2. Results of analysis without laser data.

Now we present a case where the glitches were removed by the eyeball method. In this case, the error is 5.52 for the SN baseline and 4.33 for the SE baseline, neither one being a significant change from the case with no lasers. Note that FITMOD was used in this case to compute the laser plate model.
SOLUTION FROM 13086 OBSERVATIONS SPANNING 0.1 DAYS

DATES 871016.064845 TO 871016.091205

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>BASE-LINE(S)</th>
<th>TIMESPAN</th>
<th>STAR(S)</th>
<th>NUM</th>
<th>VALUE</th>
<th>ERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td>BX</td>
<td>*-----</td>
<td>*--------</td>
<td>ALL</td>
<td>6744</td>
<td>-0.8272</td>
<td>8.8511</td>
</tr>
<tr>
<td>BY</td>
<td>*-----</td>
<td>*--------</td>
<td>ALL</td>
<td>6744</td>
<td>-1.1165</td>
<td>11.7063</td>
</tr>
<tr>
<td>BZ</td>
<td>*-----</td>
<td>*--------</td>
<td>ALL</td>
<td>6744</td>
<td>-0.5273</td>
<td>7.2457</td>
</tr>
<tr>
<td>C</td>
<td>*-----</td>
<td>*--------</td>
<td>ALL</td>
<td>6744</td>
<td>-3.9967</td>
<td>14.7083</td>
</tr>
<tr>
<td>BX</td>
<td>*-----</td>
<td>*--------</td>
<td>ALL</td>
<td>6342</td>
<td>-0.0258</td>
<td>8.6115</td>
</tr>
<tr>
<td>BY</td>
<td>*-----</td>
<td>*--------</td>
<td>ALL</td>
<td>6342</td>
<td>-0.0642</td>
<td>11.7019</td>
</tr>
<tr>
<td>BZ</td>
<td>*-----</td>
<td>*--------</td>
<td>ALL</td>
<td>6342</td>
<td>-0.0206</td>
<td>6.9440</td>
</tr>
<tr>
<td>C</td>
<td>*-----</td>
<td>*--------</td>
<td>ALL</td>
<td>6342</td>
<td>+0.2409</td>
<td>14.5227</td>
</tr>
<tr>
<td>DN-DS</td>
<td>*-----</td>
<td>*--------</td>
<td>ALL</td>
<td>6744</td>
<td>+245.1468</td>
<td>17.6668</td>
</tr>
<tr>
<td>DN-DS</td>
<td>*-----</td>
<td>*--------</td>
<td>ALL</td>
<td>6342</td>
<td>+165.7162</td>
<td>17.4123</td>
</tr>
</tbody>
</table>

SN POST-SOLUTION RESIDUALS

-----------------------------
NUMBER OF OBS'NS = 6744
RMS OF RESIDUALS = 10.53
PEAK TO PEAK = 60.83
SE POST-SOLUTION RESIDUALS

-----------------------------
NUMBER OF OBS'NS = 6342
RMS OF RESIDUALS = 4.35
PEAK TO PEAK = 31.93

Table 3. Results of analysis using DEGLITCH.

Lastly, we present the results of a case just like that of Table 4, except for the fact that FITMOD was not used on the theory that the laser data might still contain a few glitches, which would ruin the fit. So the laser plate model used in this case is simply a
SOLUTION FROM 13086 OBSERVATIONS SPANNING 0.1 DAYS

DATES 871016.064845 TO 871016.091205

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>BASE-</th>
<th>TIMESPAN</th>
<th>STAR(S)</th>
<th>NUM</th>
<th>VALUE</th>
<th>ERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td>BX</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6744</td>
<td>+129.3912</td>
<td>5.4060</td>
</tr>
<tr>
<td>BY</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6744</td>
<td>-179.1619</td>
<td>7.1498</td>
</tr>
<tr>
<td>BZ</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6744</td>
<td>+51.3218</td>
<td>4.4254</td>
</tr>
<tr>
<td>C</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6744</td>
<td>-223.8275</td>
<td>8.9833</td>
</tr>
<tr>
<td>BX</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6342</td>
<td>+4.7283</td>
<td>5.2596</td>
</tr>
<tr>
<td>BY</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6342</td>
<td>-9.0901</td>
<td>7.1471</td>
</tr>
<tr>
<td>BZ</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6342</td>
<td>+0.9939</td>
<td>4.2412</td>
</tr>
<tr>
<td>C</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6342</td>
<td>-10.3662</td>
<td>8.8700</td>
</tr>
<tr>
<td>DN-DS</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6744</td>
<td>-8.5301</td>
<td>10.7903</td>
</tr>
<tr>
<td>DN-DS</td>
<td>------</td>
<td>*********</td>
<td>ALL</td>
<td>6342</td>
<td>+155.6797</td>
<td>10.6348</td>
</tr>
</tbody>
</table>

SN POST-SOLUTION RESIDUALS

-----------------------------
NUMBER OF OBS'NS = 6744
RMS OF RESIDUALS = 5.52
PEAK TO PEAK = 44.77

SE POST-SOLUTION RESIDUALS

-----------------------------
NUMBER OF OBS'NS = 6342
RMS OF RESIDUALS = 4.33
PEAK TO PEAK = 31.04

Table 4. Results of analysis using eyeball method deglitching.

Copy of LP1ATE.DAT, the measured laser plate model. We see that these results barely differ at all from the results using FITMOD.
SOLUTION FROM 13086 OBSERVATIONS SPANNING 0.1 DAYS
DATES 871016.064845 TO 871016.091205

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>BASE-</th>
<th>TIMESPAN</th>
<th>STAR(S)</th>
<th>NUM</th>
<th>VALUE</th>
<th>ERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>LINE(S)</td>
<td>ACTIVE</td>
<td>OBS'NS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BX</td>
<td><em>------</em></td>
<td><em>--------</em></td>
<td>ALL</td>
<td>6744</td>
<td>+116.9707</td>
<td>5.4112</td>
</tr>
<tr>
<td>BY</td>
<td><em>------</em></td>
<td><em>--------</em></td>
<td>ALL</td>
<td>6744</td>
<td>-162.6138</td>
<td>7.1568</td>
</tr>
<tr>
<td>BZ</td>
<td><em>------</em></td>
<td><em>--------</em></td>
<td>ALL</td>
<td>6744</td>
<td>+42.9193</td>
<td>4.4297</td>
</tr>
<tr>
<td>C</td>
<td><em>------</em></td>
<td><em>--------</em></td>
<td>ALL</td>
<td>6744</td>
<td>-203.6788</td>
<td>8.9921</td>
</tr>
<tr>
<td>BX</td>
<td><em>-------</em></td>
<td><em>--------</em></td>
<td>ALL</td>
<td>6342</td>
<td>-2.0457</td>
<td>5.2647</td>
</tr>
<tr>
<td>BY</td>
<td><em>-------</em></td>
<td><em>--------</em></td>
<td>ALL</td>
<td>6342</td>
<td>+0.4020</td>
<td>7.1541</td>
</tr>
<tr>
<td>BZ</td>
<td><em>-------</em></td>
<td><em>--------</em></td>
<td>ALL</td>
<td>6342</td>
<td>-3.2374</td>
<td>4.2453</td>
</tr>
<tr>
<td>C</td>
<td><em>-------</em></td>
<td><em>--------</em></td>
<td>ALL</td>
<td>6342</td>
<td>+1.0276</td>
<td>8.8786</td>
</tr>
<tr>
<td>DN-DS</td>
<td><em>-------</em></td>
<td><em>--------</em></td>
<td>ALL</td>
<td>6744</td>
<td>+16.4332</td>
<td>10.8008</td>
</tr>
<tr>
<td>DN-DS</td>
<td><em>-------</em></td>
<td><em>--------</em></td>
<td>ALL</td>
<td>6342</td>
<td>+169.3720</td>
<td>10.6452</td>
</tr>
</tbody>
</table>

SN POST-SOLUTION RESIDUALS

NUMBER OF OBS’NS = 6744
RMS OF RESIDUALS = 5.53
PEAK TO PEAK = 45.01

SE POST-SOLUTION RESIDUALS

NUMBER OF OBS’NS = 6342
RMS OF RESIDUALS = 4.31
PEAK TO PEAK = 31.08

Table 5. Results of analysis using eyeball method but no FITMOD.
CHAPTER 8 : CONCLUSIONS AND SUGGESTIONS

The results listed in Chapter 7 are somewhat less than had been hoped for from this laser metrology system; the results are rather inconclusive. There are a couple of conclusions that can be drawn, however. One conclusion is that glitches, even relatively small ones, can be really bad for the data reduction. This assertion is based on the results in Table 3. The North-South residual error got twice as bad as it had been previously due simply to the glitches that got through DEGLITCH, and those glitches have to be pretty small or they would have been eliminated by DEGLITCH. So, my conclusion is that just a few small glitches were enough to ruin our chances of a good result.

The obvious question is then whether the same can be said of the data deglitched by the eyeball method. Are there still a few little glitches remaining afterwards? We know that there are some suspicious-looking blips in the data, but this eyeball just couldn’t find a glitch to go with any of them. If there are glitches still remaining in the data, then that would essentially explain the remaining error. If there are no glitches in the data, then we have to start considering the possibility that there is another source of error that is more dominant than siderostat wobble.

Another interesting thing that pops out of the data is the goodness of fit to the value of $d_N - d_S$; even in the best case, the uncertainty is on the order of 10 microns or more. Is this a result of glitchy data again, or is this an uncertainty inherent in the problem due to highly correlated parameters? Either way, it indicates that using the laser data to attempt to provide fringe corrections is pretty tough. We’re interested in providing information about motions on the order of 1 micron, but that gets to be awfully hard to do when you don’t know the position of the siderostat mirror with respect to the bowling ball to within 10 microns. If this is a fundamental property of the problem, then we’re stuck, and it’s going to require a lot of observations over a number of nights to begin to pin down the exact values of $d_N - d_S$. If, on the other hand, the problem is caused by
glitches, then, again, all we have to do is eliminate them.

It seems like a pretty big coincidence that the residual errors after applying the laser data for both the SN and the SE are so similar to the residuals obtained with no laser data. Is that an indication that there are other sources of error that are dominating?

If the glitches are caused by Doppler shifting (as many believe they are), then there are a couple of possible solutions. One is to simply slew the siderostats at a much slower speed, so that the vibrations are minimized; this has actually been done in the past with reasonable success. By doing this, you also manage to make sure that none of the laser values change by more than a single fringe in 2 seconds, so that detecting and correcting glitches is much easier.

Conversely, we could keep the same slewing speed and merely increase the rate at which we record data so that the laser values would never change by more than a single fringe in one unit of time.

Another way to beat the Doppler dropouts is to go to a much higher frequency than 20 kHz. For example, schemes have been proposed which use 1 MHz as the difference frequency, although the electronics game becomes alot more difficult to play. But then the bowling ball would have to move at the phenomenal rate of $10^8$ fringes per second (about 0.6 m/s) in order to cause a dropout.

Another suggestion, and really my strongest suggestion of all, is to have a five-laser plate instead of four. I don’t feel confident that we’ll ever be able to get rid of glitches completely, and, with five lasers, error figures like those in Chapter 6 can be used to determine not only if there’s a glitch, but also which laser glitched, making the task of correcting the glitch much easier.
APPENDIX: Programs and Data Files

Programs

DEGLITCH.FOR This program is designed to detect and remove glitches from the laser data file FTxx.LRW. Its output is FTxx.LGF (for Laser Glitch-Free). This program requires a nominal laser plate model to be in the file LPLATE.DAT. DEGLITCH operates as follows. Using the laser readings, it, for each record in FTxx.LRW, determines the position of the bowling ball center based on the model in LPLATE.DAT. The determined center is unimportant for now and is ignored. What IS important is the mean-square error in the least-squares determination of that center. If the mean-square error suddenly increases dramatically from previous values, it indicates that a glitch has likely occurred in one of the lasers. To test whether a glitch has really occurred, we read in the next several points. In the event that it is a glitch, we then must correct the it. We can't immediately tell which laser glitched, so we don't know which laser to correct. So what we do is to try correcting each of the four lasers in turn and seeing what that does to the next several points. The correction that seems to improve things the most is kept, or if it seems that we're just better off making no correction at all, then no correction is made. The corrections are made as follows: Knowing which laser glitched, we can use the other three lasers to compute the position of the bowling ball center and compute a predicted value for the fourth laser. This predicted value is compared to the actual value, and the difference becomes an offset that is henceforth applied to any readings from the fourth laser. GLITCH.LOG contains information concerning detected and corrected glitches. MSE1.LOG is simply a record of the north pier error at each time, while MSE2.LOG is for the east and south.

Inputs: LPLATE.DAT, FTxx.LRW

94
Outputs: FTxx.LGF, GLITCH.LOG, MSE1.LOG, MSE2.LOG

LGF2LAS.FOR This program converts the raw laser data file FTxx.LRW to a processed laser reading file FTxx.LAS. The raw file contains laser readings in 2-second intervals, by pier, for the whole night. The processed file contains one record per scan, this record being the average of the readings for that scan. The idea is just to reduce the amount of information that has to be munched on by other programs.

Input: FTxx.LGF

Output: FTxx.LAS

CONVXYZ.FOR This program converts the laser readings in FTxx.LAS (which are in units of $\lambda/128$ into x-y-z motion of the bowling ball in the Mt. Wilson coordinate frame, in units of microns. Its output is the file FTxx.PVT, which contains one record for each scan. This program requires the laser plate file FTxx.LPT. The heart of this program is the subroutine LsCenter, which performs a least-squares fit to find the sphere center, relying heavily on Numerical Recipes routines. The log file reports the error of fit of each point.

Inputs: FTxx.LPT, FTxx.LAS

Outputs: FTxx.PVT, CONVXYZ.LOG

FITMOD.FOR This program computes the model parameters for the laser plate models using the redundant laser. It only computes the elevation angles $\phi_i$ since it can only solve for three of the parameters. The $\theta_i$'s pass through unchanged. Note that the input to the program is not in a form that lists the values of the parameters explicitly; rather, the initial values of the angles are present implicitly in the unit vectors contained in the file LPLATE.DAT. FITMOD uses the Levenberg-Marquardt method documented in Numerical Recipes to solve the nonlinear least-squares problem. The
input data to fit is in the file FTxx.LAS. The outputs are FTxx.LPT, which is just an updated version of LPLATE.DAT, and FTxx.LMD, which is a list of the five angles produced merely for human consumption.

Inputs: LPLATE.DAT, FTxx.LAS

Outputs: FTxx.LPT, FTxx.LMD

FINDGL.FOR This program allows the removal of a single glitch from a laser data file. The location of the glitch and the size of the correction are input manually. This program is used as part of the "eyeball method" for removing glitches. You observe the error plots produced, then you locate the glitch in the .LRW file as best you can based on the plot. Then the program RMGLITCH can be used to remove that glitch, creating a new file. Continue this process until all glitches have been removed. This program generates a log file called MSE.LOG that contains the error-of-fit information computed. Then QPLOT can be used to view the results. They are real numbers, and the record length is 12. North data is available at byte 0, South at byte 4, East at byte 8. FINDGL does not produce a laser file on output. This program is adapted from DEGLITCH, see comments there for an in-depth description of its function. This is one of those programs that has evolved until it has a number of non-functional options and unadvertised features. I would advise, if people get serious about the eyeball method, that someone write some new eyeball code.

Inputs: LPLATE.DAT, input laser file

Outputs: output laser file, MSE.LOG

IMGLITCH.FOR The other half of the dynamic duo (FINDGL was the first half - you weren't paying attention, were you?) This program allows the removal of a single glitch from a laser data file. The location of the glitch and the size of the correction are input manually. This program is used as part of the "eyeball method" for removing glitches. You
observe the error plots produced by FINDGL, then you locate the glitch in the .LRW file as best you can based on the plot. Then this program can be used to remove that glitch, creating a new file. Then you can use FINDGL on the new file to see if you like the results. Continue this process until all glitches have been removed. This program allows the removal of a single glitch, creating a revised laser file as its output. This program is adapted from DEGLITCH, see comments there for a in-depth description of its function.

Inputs: input laser file

Outputs: Output laser file

ISCENTER.FOR A subroutine used by many of the programs in the laser metrology system. This subroutine computes the least-squares center of the sphere and the error of fit associated with it based on four laser readings and the unit vectors in the direction of each laser. The error of fit returned is actually a root-sum-square error.

GLITSUBS.FOR A set of subroutines called by many of the laser metrology programs. Anything therein that is actually useful is well commented.

CONVSUBS.FOR This is like GLITSUBS, but is used by CONVXYZ.

MATRIX.FOR Matrix utilities.

PTW.FOR Important subroutines for converting back and forth between coordinate frames and for converting between laser plate angles and unit vectors. Used in FITMOD; has some functional overlap with CONVSUBS and GLITSUBS. Reasonably well commented.

FIT.FOR The Kalman Filter version of FITMOD, this program is reasonably commented but is still rather cryptic in places. If someone decides to be intense about having yet another Kalman Filter in the interferometer, it would probably be best to rewrite
this one. The filter is the one described in Chapter 3 of this thesis.

CALCIO8.FOR A set of subroutines for George’s CALC program. The ones pertinent to this thesis are RDPVT and FEEDBM, which read data from the FTxx.PVT and FTxx.FDB files respectively.

CALCSUB8.FOR More subroutines for CALC. The one of interest here is PIVOTP, which is the implementation of Chapter 4 of this thesis. This routine does any necessary I/O through CALCIO8.

LRW2LAS.FOR Like LGF2LAS, but goes directly from the LRW file to a LAS file.

BLGF2LAS.FOR Like LGF2LAS, but produces one output record for every input record (i.e., no averaging takes place).

BLRW2LAS.FOR Like BLGF2LAS, but goes from LRW to LAS.

Data Files

LPLATE.DAT The measured laser plate data file. You create this file by assembling the pieces LPLATEN.DAT followed by LPLATES.DAT followed by LPLATEE.DAT (in other words, just append the three files together). The file then consists of 12 records, each record containing the unit vector \((w_x, w_y, w_z)\), or, in other words, the x-y-z components of the unit vector in Mt. Wilson coordinates.

Record Format: 3F18.15

FTxx.LPT Exact same format as LPLATE.DAT.

FTxx.LRW The raw laser data file. Laser units are \(\lambda/128\). Format:

1-5 Scan number (I5)

6 EX
7–8 Hours (I2)

9:

10–11 Minutes (I2)

12:

13–14 Seconds (I2)

15 1X

16–24 North laser 1 (I9)

25 1X

26–34 North laser 2 (I9)

::

125 1X

126–134 East laser 4 (I9)

FTxx.LGF Exact same format as FTxx.LRW.

FTxx.PVT See description above under CONVXYZ. One record per scan. Format:

1–5 Scan number (I5)

6 1X

7–8 Hours (I2)

9:

10–11 Minutes (I2)

12:
13–14 Seconds (I2)

15–25 x-coordinate of North bowling ball (F11.2)

26–36 y-coordinate of North bowling ball (F11.2)

::

103–113 z-coordinate of East bowling ball (F11.2)

FTxx.FDB Azimuth and elevation angles of the feedbeams from the siderostat models (in radians). Six records of F20.17, organized as Az_N, El_N, Az_S, El_S, Az_E, El_E.

FTxx.LMD Human-readable laser plate parameter file. Five records of F18.12 for each of the three laser plates, in the order φ_1, φ_2, φ_3, θ_2, θ_3.
BIBLIOGRAPHY


