A SIDEROSTAT CONTROL SYSTEM

by

Matthew J. Kelly

B.S.E.E., Massachusetts Institute of Technology, 1984

Submitted to the
Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements
for the Degree of
Master of Science
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 1986

(c) Massachusetts Institute of Technology 1986

Signature of Author

Department of Electrical Engineering and Science

February 1986

Certified by

Professor David H. Staelin
Thesis Supervisor

Accepted by

Professor Arthur C. Smith
Chairman, Department Committee

ARCHIVES

MASSACHUSETTS INSTITUTE
OF TECHNOLOGY

APR 11 1986

LIBRARIES
ARCHIVES
A Siderostat Control System

by

Matthew J. Kelly

Submitted to the Department of Electrical Engineering and Computer Science on January 17, 1986, in partial fulfillment of the requirements for the Degree of Master of Science in Electrical Engineering and Computer Science.

ABSTRACT

A necessary subsystem for the Stellar Astrometric Interferometer is a set of siderostats composed of steerable mirrors which track the star's motion across the sky.

In previous implementations of the siderostat subsystem (such as the Mark II), the open-loop pointing accuracy achieved was on the order of 4 arcminutes. For automatic operation, the pointing requirements dictate rms astrometric error on the order of 5 to 10 arcseconds. To achieve this level of accuracy, an eight parameter geometric model was specified to describe the siderostat. Motor control hardware was designed and constructed to handle two siderostats. A real-time operating system was developed along with software tracking and servo algorithms to control the siderostats.

A single siderostat was tested at Mt. Wilson Observatory to determine the resulting pointing accuracy. Using a least-squares model reduction method, pointing accuracy was found to approach the desired specifications of sub-10 arcsecond pointing with some restrictions. Attempts to improve the least-squares method proved somewhat futile, while the development of a Kalman filter model calculation scheme should achieve significant improvement in astrometric pointing error and model stability. This thesis describes the design and testing of the siderostat subsystem.

The Stellar Astrometric Interferometer represents a joint endeavor of the Massachusetts Institute of Technology, the Harvard/Smithsonian Astrophysical Observatory, the United States Naval Observatory, and the Naval Research Laboratory.

Thesis Supervisor: David H. Staelin
Professor of Electrical Engineering
Acknowledgements

I would like to thank Professor David Staelin of MIT for inviting me to work on this project and for his suggestions on analyzing the results. Also, Dr. Michael Shao of SAO deserves many thanks for allowing me to use a COMPAQ computer instead of a Nova for the actual siderostat system, as well as for supervising my progress on a day-to-day basis. I would also like to express my appreciation to Dr. Mark Colavita of SAO for his many suggestions in improving the system as a whole, for his aid in conducting the tests at Mt. Wilson, and for his suggestion of using a Kalman filter in the data analysis.

Many thanks also to Mr. Jack Barrett at MIT for his aid in overcoming many obstacles, such as obtaining and machining parts, and to Mr. John Pohlman at USNO for producing, assembling, disassembling, and reassembling ad infinitum the siderostat for the trips to Mt. Wilson and back.

The completion of this thesis would not have been possible without the constant encouragement of my family (namely my mom) and my friends Miki Takeda, Mark Ortiz, Jim Crocker, John Trumbull, and Chris Davis, along with the office visits of Ruth Griffith, Dennis Ward, and Sandy Lugo.
SIDEROSTAT
Table of Contents

I. Introduction 7
II. Description of Siderostat Subsystem 11
III. Description of the Geometric Model 17
IV. Control Architecture 36
V. Summary of Test 52
VI. Conclusions and Extensions 85

Appendices
A. Motor Control Equations 89
B. Least-squares Parameter Vector Optimization 93
C. Kalman Filter Parameter Vector Optimization 99

Bibliography 102
List of Figures and Tables

Fig. No.   Description
1    Block diagram of interferometer
2    Simple diagram of siderostats
3    Block diagram of siderostat subsystem
4    Sample coordinate transformation
5    Polar axis to Mt. Wilson transformation
6    Azimuth axis to Mt. Wilson transformation
7    Elevation axis to azimuth axis transformation
8    Mirror axis to elevation axis transformation
9    Feed beam and azimuth axis orientation
10   The four parameters fixed to the siderostat
11   An ideal siderostat and the coordinate frames
12   Pointing at a star - geometric considerations
13   Real-time operating system
14   Servo and system timing
15   Test setup
16   Pointing test
17 to 27    Least-squares results
17 to 27    Kalman filter results
28    Least-squares results
28    Kalman filter results

A1 to A2 Motor control equations
B1 to B3 Least-squares derivation
C1 to C3 Kalman filter derivation

Table No.

1    Summary of coordinate frames and transformations
2    Summary of siderostat model parameters
3    Least-squares siderostat model parameter values
4    Least-squares astrometric errors
5    Kalman filter siderostat model parameter values
6    Kalman filter astrometric errors
A Siderostat Control System

I. Introduction

The accurate determination of stellar positions has long occupied a major portion of the effort expended in astronomical research. This science of astrometry has required the development of instruments of ever higher spatial resolution and precision. A number of methods have been attempted with varying degrees of accuracy. Precision mechanical astrolabes, for instance, are capable of determining stellar positions with an accuracy of no better than a few arcseconds.[1] Photographic averaging techniques used on large sets of plates taken over several years are good to approximately ten milliarcseconds.[2] In recent years, development has begun on instruments utilizing interferometric techniques that should eventually achieve one-tenth milliarcsecond accuracy over a few hours.[3]

One such high-accuracy instrument is the Stellar Astrometric Interferometer currently being implemented at Mt. Wilson Observatory as a joint endeavor of the Massachusetts Institute of Technology, the Harvard/Smithsonian Astrophysical Observatory, the United States Naval Observatory, and the Naval Research Laboratory. This instrument is basically a modified Michelson
interferometer capable of measuring both fringe phase and fringe visibility. [3] The interferometer consists of three main sections: a laser-monitored delay line, a photon camera star-tracker, and a siderostat mirror pointing system. A simplified diagram of the system appears in Figure 1. The purpose of this thesis is to discuss the design, implementation, and performance of the siderostat control system. For details regarding the other two systems and the operation of the interferometer as a whole, the reader is referred to the literature outlined in the bibliography.

As can be inferred from Figure 1, the purpose of the siderostat subsystem is to track and hold the stellar images and redirect the stellar beams into the interferometer. A siderostat is basically a precision mirror mounted on a structure that can rotate around two orthogonal axes. It derives its name from the fact that it causes the sidereal motion of the stars to appear stationary relative to the interferometer. In the actual implementation, the siderostat subsystem will consist of three siderostats; one placed to the north, one to the south, and a third to the east of the interferometer. Only two will be operational at a given time, and those two will determine the baseline of the instrument as a whole. In order for the interferometer to function properly, the siderostats must be able to point the
reflected stellar beam into the aperture of the star-tracker optics. This window is approximately 50 arcseconds in diameter, so it should be obvious that the pointing requirements are somewhat stringent. The problem, then, is to define, build, and test a control structure for the siderostats that enables the siderostats to track a star and reflect the stellar image into the aperture of successive systems of the interferometer. As will be shown, the solution to this problem requires a substantial geometric modelling and real-time control effort. First, though, it would be beneficial to consider the detailed mechanical structure of a siderostat and to make a comparison of a previous siderostat subsystem with the new one.
II. Description of Siderostat Subsystem

In previous designs of the interferometer (such as the Mark I and Mark II), the siderostats each consisted of a precision mirror mounted in an alt/az (altitude/azimuth) configuration as shown in Figure 2A. The azimuth axis was assumed to be perpendicular to the earth's surface and pointed in the direction of zenith. The altitude, or elevation, axis was, in turn, assumed to be perpendicular to the azimuth axis. Thus the altitude axis rotated by virtue of the azimuth axis in a plane tangent to the earth's surface. Each of the axes was driven by a stepper motor connected to a high-reduction gearbox. Each gearbox was connected by a worm drive to a large gear attached to the main shaft of the axis. All mechanical and optical portions of the siderostat were precision machined and ground. Nonetheless, the previous siderostat control system was able to achieve a pointing accuracy of no better than about 4 arcminutes. This is due mainly to the fact that no modelling was done on the siderostat. In other words, the axes were assumed to be perpendicular and oriented perfectly relative to the earth, and the mirror was assumed to be placed perfectly in place. Only two offsets were incorporated to account for errors in the initial starting position of the siderostats. The assumptions of perfect perpendicularity
FIG. 2
and parallelism are not valid at the 5 to 10 arcsecond level of accuracy necessary for the proper automatic operation of the photon camera star-tracker and interferometer as a whole. As will be seen later, a more thorough modelling effort must be undertaken to achieve the desired specifications.

The new siderostats are similar mechanically to the previously described ones. They use the same type of stepper motors and similar mirrors as before. The major physical difference, as can be seen in Figure 2B, is the fact that the azimuth axis is no longer perpendicular to the earth’s surface, but now lies 15 degrees above the plane tangent to the earth's surface, or 75 degrees from zenith. This unusual tilt of the azimuth axis is necessary to allow room for future addition of a laser metrology system that will measure the orientation of the mirror. This laser system will be placed directly below the mirror, and thus the azimuth axis is tilted to allow the mirror to hang over the laser system. In addition, better bearings and more precisely machined gear systems have been utilized to keep nonlinear effects such as backlash and gear eccentricities to a minimum. In fact, optical backlash was measured in the laboratory to be on the order of 2 to 5 arcseconds. This test was accomplished by observing the shift in position of a spot
of auto-collimated laser light reflected off of the siderostat mirror onto a wall at a distance of several meters to achieve a significant lever arm. The siderostat motors were stepped in one direction a given distance and then back the same distance. The distance (in arcseconds) from the spot's starting position to the finishing position represents the backlash.

The major difference in the system as a whole, however, is the inclusion of a complete geometric model in the control structure. This model makes none of the assumptions of perpendicularity and parallelism of the axes and actually attempts to determine the underlying physical error angles found in the siderostat and its placement relative to the earth and other optical equipment. The model solving algorithm can also be made adaptive so that as model parameters change due to temperature variations or other drifts, new updated models can be determined.

A block diagram of the complete siderostat subsystem is shown in Figure 3. The subsystem consists of the three siderostats (only two are shown to emphasize the fact that only two are operational at a given time), motor control electronics (a pulse generator and power supply), and a dedicated computer that interfaces with a paddle and real-time clock/interrupt generator. The computer, a COMPAQ Deskpro (IBM-PC fast clone), contains the entire
FIG. 3
control and model-solving algorithm software, and houses the motor pulse generator and other interface circuitry. Not shown in the diagram are two data links to other computers in the system. All of the computers that control the various subsystems will eventually be interconnected through a dedicated local-area network so that information necessary for the operation of the interferometer may be shared. Since all of the control hardware and software revolves around the existence of a modelling scheme for the siderostat, the next portion of the subsystem that should be considered is the geometric model itself.
III. Description of the Geometric Model

As noted in the previous section, the primary innovation in the new siderostat subsystem is the inclusion of a geometric modelling scheme in the control algorithm. Such a geometric model is necessary to achieve the desired open-loop pointing accuracy of 5 to 10 arcseconds required by the star-tracker. This model is based purely on geometric considerations: the only errors assumed to be present in the siderostat system are those arising from angular offsets from nominal in various axes fixed to the siderostat and the earth. Not taken into account in the model are nonlinear effects such as backlash and gear eccentricities. The motivations for choosing a strictly geometric model are threefold:

1. The parameters modelled reflect reasonable physical errors that could be expected to be produced as a result of machining processes and surveying techniques used in producing and setting up the subsystem. They are also fairly easy to visualize.

2. The model yields simple closed-form solutions for motor positions that can be generated in real time.

3. Nonlinear error effects such as backlash and gear eccentricities should be the major remaining errors after those accounted for in the model. These errors should be, as mentioned in the previous chapter, only on the order of 2 to 5 arcseconds since great care was taken in the production of the siderostat.

The above benefits should become apparent as the model
and control software are explained. First, though, it would be beneficial to consider some basic principles of coordinate transformation and vector manipulation.

The fundamental principle underlying the modelling scheme is the idea that it is possible to express a displacement vector (a vector having no origin, but possessing magnitude and direction) defined in one coordinate frame in another coordinate frame by using a transformation containing only two unique angles.[4] Another useful and more obvious way of stating the result is to say that any coordinate frame can be generated from a given coordinate frame by rotating through only two angles. As an example, consider coordinate frames 0 and 1 as shown in Figure 4A. Suppose it is desired to find the transformation that relates a displacement vector defined in coordinate frame 1 to coordinate frame 0. Furthermore, assume that coordinate frame 1 was generated from frame 0 by performing the following steps:

1. First rotate around $^0z$ an angle theta.
2. Then rotate around $^1x$ an angle alpha.

Given these rotations, then, the transformation matrix, defined as $^0T_1$, that transforms vectors in coordinate frame 1 to frame 0 is as shown in Figure 4B. Henceforth, a transformation matrix that transforms from frame b to frame a will be defined as $^aT_b$, and a vector $Q$ defined in a given frame a will be denoted as $^aQ$. Thus, if
\( \mathbf{T}_1 \) =

\[
\begin{bmatrix}
\cos \theta & -\sin \theta \cos \alpha & \sin \theta \sin \alpha \\
\sin \theta & \cos \theta \cos \alpha & -\cos \theta \sin \alpha \\
0 & \sin \alpha & \cos \alpha
\end{bmatrix}
\]
one is given a vector $b^k$ and transformation $a^k = a^i_b b^i$. Thus vectors defined in any one coordinate frame can be expressed in another frame if one is given the transformation matrix relating the two frames. In addition, since these transformations are unitary transformations, the reverse transformation is simply the transpose of the forward one. Thus $b^k_T = (a^i_b)^t$.

Note in the figures that Sangle and Cangle correspond to $\sin(\text{angle})$ and $\cos(\text{angle})$, respectively. This convention is followed throughout this thesis (including the appendices) and simplifies the representation of matrices.

Following the previous development, then, a geometric model of the siderostat subsystem can be developed. The actual model implemented consists of eight unique parameters that express various physical angles present in the siderostat and feed beam (the feed beam represents the direction of the reflected stellar beam - in the final implementation the feed beam denotes the direction into the interferometer and star-tracker, but for the test as described in this paper the feed beam denotes the direction into a telescope aimed at the siderostat). To define the parameters five distinct coordinate frames must be defined. First, a coordinate frame (the polar frame), denoted $p$, is fixed to the earth
(see Figure 5). Star positions can be referred easily to this frame using their hour angle and declination since \( \mathbf{p}_x \) points to the North Celestial Pole, \( \mathbf{p}_x \) points east, and \( \mathbf{p}_y \) is normal to the equator. Next, a local coordinate frame, denoted \( \mathbf{w} \), is fixed to Mt. Wilson. Zenith is the same as \( \mathbf{w}_z \), \( \mathbf{w}_y \) points west, and \( \mathbf{w}_x \) points north. The transformation from \( \mathbf{p} \) to \( \mathbf{w} \) is \( \mathbf{w}_p \) and is a function only of the latitude of Mt. Wilson (see Figure 5). Then a coordinate frame, named \( \mathbf{a} \), is fixed to the azimuth axis of the siderostat (see Figure 6). The axis \( \mathbf{a}_z \) is parallel to the azimuth axis. The transformation from \( \mathbf{a} \) to \( \mathbf{w} \) is \( \mathbf{w}_a \). Then, relative to the azimuth axis, a frame called \( \mathbf{e} \) is defined. The axis \( \mathbf{e}_z \) is parallel to the elevation axis (see Figure 7) and \( \mathbf{a}_e \) denotes the transformation from \( \mathbf{e} \) to \( \mathbf{a} \). Finally, a frame named \( \mathbf{m} \) is defined relative to the elevation axis (see Figure 8). In this frame, the normal to the mirror's surface, henceforth called \( \mathbf{m}_p \), is equal to \( \mathbf{m}_y \). The transformation from \( \mathbf{m} \) to \( \mathbf{e} \) is \( \mathbf{e}_m \), as shown. Given these transformations, then, the mirror normal expressed in frame \( \mathbf{m} \) can be referred to the local Mt. Wilson frame \( \mathbf{w} \) through the following equation: \( \mathbf{w}_p = (\mathbf{w}_a)(\mathbf{a}_e)(\mathbf{e}_m)\mathbf{m}_p \). A summary of the coordinate frames and their corresponding transformations appears in Table 1.

Now that the coordinate frames have been defined, a discussion of the siderostat model parameters can begin.
\[ w_{T_p} = \begin{bmatrix} -s\lambda & 0 & c\lambda \\ 0 & -1 & 0 \\ c\lambda & 0 & s\lambda \end{bmatrix} \]

FIG. 5
\[ w_{T_A} = \begin{bmatrix} \cos \alpha_A & -\sin \theta_A \cos \alpha_A & \sin \theta_A \sin \alpha_A \\ \sin \theta_A & \cos \alpha_A & -\cos \theta_A \sin \alpha_A \\ 0 & \sin \alpha_A & \cos \alpha_A \end{bmatrix} \]
\[ \theta_i = K_i M_i + D_i \]

\[ A_{T_E} = \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_e & s\theta_i s\alpha_e \\ s\theta_i & c\theta_i c\alpha_e & -c\theta_i s\alpha_e \\ 0 & s\alpha_e & c\alpha_e \end{bmatrix} \]

FIG. 7
\[ \Theta_2 = K_2 M_2 + D_2 \]

\[ E_{X} \quad E_{Y} \quad E_{Z} \]

\[ \alpha_M \]

\[ M_Y \quad M_X \]

\[ E_T^M = \begin{pmatrix}
    c\Theta_2 & -s\Theta_2 c\alpha_M & s\Theta_2 s\alpha_M \\
    s\Theta_2 & c\Theta_2 c\alpha_M & -c\Theta_2 s\alpha_M \\
    0 & s\alpha_M & c\alpha_M
\end{pmatrix} \]

**FIG. 8**
## Table 1
Summary of Coordinate Frames and the Corresponding Transformations

### Coordinate Frames

p - the polar-axis frame. The x axis points normal to the equator, the y axis points east, and the z axis points to the North Celestial Pole.

w - the Mt. Wilson frame. The x axis points north, the y axis points west, and the z axis points to zenith.

a - the siderostat azimuth axis frame. The x and y axes are normal to the azimuth axis, and the z axis is coincident to the azimuth axis.

e - the siderostat elevation axis frame. The x and y axes are normal to the elevation axis, and the z axis is coincident to the elevation axis.

m - the siderostat mirror frame. The x and z axes are perpendicular to the mirror normal, and the y axis is coincident with the mirror normal.

### Coordinate Transformations

\[ w_p \] - transforms from polar frame to Mt. Wilson frame.

\[ w_a \] - transforms from azimuth axis frame to Mt. Wilson frame.

\[ a_e \] - transforms from elevation axis frame to azimuth axis frame.

\[ e_m \] - transforms from mirror frame to elevation frame.

The reverse transformations are merely the forward ones transposed. In addition, the transformations may be combined by multiplying them in the proper order. For example, to transform from mirror frame to azimuth frame, the transformation \[ z_m \] is simply \[ z_e \] .
The first two parameters, $^w_{\text{Faz}}$ and $^w_{\text{Fel}}$, define the azimuth and elevation of the feed beam referred to the local Mt. Wilson coordinate frame $w$, as shown in Figure 9. The feed beam is actually defined as a unit vector, $^wF$, and $^w_{\text{Faz}}$ and $^w_{\text{Fel}}$ then represent the azimuth and elevation of that vector. These two parameters are necessary to account for inaccuracies in the placement of the optics that observe the siderostat, such as a telescope or the photon camera. Two other parameters, denoted $\theta_a$ and $\alpha_a$, reflect errors in the placement of the siderostat relative to the Mt. Wilson coordinate frame $w$ (again consult Figure 9). In the context of the foregoing discussion on coordinate frames, $\theta_a$ and $\alpha_a$ denote in some sense the azimuth and zenith angle of the siderostat azimuth axis (axis $^a_z$) in frame $w$. The next two parameters, $d_1$ (or $\text{daz}$) and $\alpha_e$, reflect errors in the machining of the siderostat and account for the fact that the elevation axis is not perfectly perpendicular to the azimuth axis. They, too, represent in some sense the azimuth and zenith angle of the elevation axis (axis $^e_z$) in frame $a$. The final two parameters, $d_2$ (or $\text{del}$) and $\alpha_m$, reflect inaccuracies in the placement of the mirror in the mounting cell. These parameters, like the previous four, correspond loosely to the azimuth and zenith angle of the mirror surface normal $^mP$ (axis $^m_y$) in frame $e$. A
pictorial representation of the latter four parameters relative to the siderostat appears in Figure 10. A summary of the eight parameters defining the model and their values for an ideal siderostat occurs in Table 2. Also, a diagram showing the coordinate frames relative to the ideal siderostat is found in Figure 11.

As noted earlier, a major advantage of the model as developed above is that it yields closed-form solutions for the motor positions. This should be clearly seen after first considering some of the astronomical aspects of tracking a star.

Consider the situation depicted in Figure 12. As can be seen there, it is desired to point the mirror of the siderostat in such a way that the stellar beam reflects off of the mirror directly in the direction of the feed beam into the telescope. Denote the unit vector to the star referred to the polar frame \( p \) as \( P_S \). \( P_S \) is purely a function of the hour angle and declination of the star. \( W_S \), the unit vector to the star referred to Mt. Wilson frame \( w \), is then expressible as \( W_S = (W_T)_P P_S \). Furthermore, the earth's atmosphere causes a refraction of the stellar beam that is a function of the star's zenith angle. Thus \( W_S \) is refracted to produce the vector \( W_S^R \) that points in the direction of the refracted stellar image. Using the simple principle of optics that a beam of light reflects off of a mirror at the opposite angle
M₁ = 0
D₁ = -90°

M₁ = 0
αₑ = 90°

M₂ = 0
D₂ = 0

M₂ = 0
αₘ = 0

FIG. 10
Table 2
Summary of Siderostat Model Parameters and their Nominal Values

WFaz - feed beam azimuth angle relative to Mt. Wilson coordinate frame w. Ideal value = 90 degrees.

WFel - feed beam elevation angle relative to Mt. Wilson coordinate frame w. Ideal value = 0 degrees.

thetaA - siderostat azimuth axis azimuth angle relative to Mt. Wilson coordinate frame w. Ideal value = 0 degrees.

alphaA - siderostat azimuth axis zenith angle relative to Mt. Wilson coordinate frame w. Ideal value = -75 degrees.

alphaE - siderostat elevation axis zenith angle relative to siderostat azimuth axis coordinate frame a. Ideal value = 90 degrees.

d1 - siderostat elevation axis azimuth angle relative to siderostat azimuth axis coordinate frame a. Ideal value = 90 degrees.

alphaM - siderostat mirror axis zenith angle relative to siderostat elevation axis coordinate frame e. Ideal value = 0 degrees.

d2 - siderostat mirror axis azimuth angle relative to siderostat elevation coordinate frame e. Ideal value = -15 degrees.

For an ideal siderostat having the parameter values given above and motor positions = 0, the mirror will point directly west. This represents the starting position of the siderostat.
\[ \theta_A = \theta_1 = \theta_2 = \alpha_M = 0^\circ \]

\[ \alpha_A = -75^\circ \]

\[ \alpha_E = 90^\circ \]

FIG. II
\[ w_P = \frac{w_F + w_S^R}{|w_F + w_S^R|} \]

**Also:**

\[ w_S^R = (w_F \cdot w_P)^w_P - (w_F \cdot w_R)^w_R \]

**Where:**

\[ w_R = \frac{w_P \times w_Q}{|w_P \times w_Q|} \quad w_Q = \frac{w_F \times w_P}{|w_F \times w_P|} \]

**FIG. 12**
at which it strikes it (referred to mirror normal), it should be obvious that to point the reflected star beam into the feed beam \( \mathbf{w}_F \) requires that the mirror normal \( \mathbf{m}_p \), referred to frame \( \mathbf{w} \), must satisfy the following constraint:

\[
\mathbf{w}_p = \frac{\mathbf{w}_F + \mathbf{w}_{s_F}}{|\mathbf{w}_F + \mathbf{w}_{s_F}|}
\]

\( \mathbf{w}_p \) can be found from \( \mathbf{m}_p \) using the transformations described earlier. Two of the transformations, \( \mathbf{e}_{T_m} \) and \( \mathbf{e}_{T_e} \), are actually functions of the stepper motor positions \( m_1 \) and \( m_2 \) (see Figures 7 and 8). These stepper motors drive the two axes of the siderostat, the azimuth and elevation axes. Thus, given a constant parameter model, \( \mathbf{w}_p \) is wholly defined by \( m_1 \) and \( m_2 \). It is necessary, though, to solve for \( m_1 \) and \( m_2 \) given \( \mathbf{w}_p \) and a parameter model. This derivation can be found in Appendix A and demonstrates the fact that the solution is entirely closed-form and no approximations or linearizations need be accomplished.

From the foregoing it should be clear that the framework for defining the siderostat control system has been established. The siderostat parameter model has been described and shown to yield closed-form solutions for motor positions. The mirror normal can be expressed in any of the coordinate frames defined through the use of
the transformations. It will be demonstrated later that the actual parameter model that describes a siderostat can be determined using a simple test. With this framework in hand a discussion of the actual control hardware and software can proceed.
IV. Control Architecture

The siderostat subsystem represents a tightly coupled arrangement of both hardware and software centered around the siderostat and its geometric model. At the core of the system, of course, are the siderostats and the stepper motors that drive the axes. Required to control these motors are a motor pulse generator and power supply. These, in turn, are coupled to a computer to allow real-time software control of the siderostat and analysis of the data. The reader is referred to Figure 3 for a block diagram of the system. Both the hardware and software aspects of the siderostat subsystem will now be discussed.

A. Hardware

It was decided that, in order to allow maximum flexibility in the control system, a general-purpose computer would be used as the main control unit. The computer settled upon is an IBM-PC fast clone made by COMPAQ - their model Deskpro. The Deskpro consists of 640 kilobytes of main memory, a 10 megabyte hard disk drive, a dual-speed Intel 8086 microprocessor capable of 4.77 or 8 megahertz operation, and an 8 megahertz Intel 8087 floating-point coprocessor. With the floating-point coprocessor the computer has 80 bit floating-point
internal precision and fully supports single- and double-
precision numbers. Such performance was found to provide
more than adequate computational accuracy and power to
handle the requirements of real-time star tracking and
siderostat control. In addition, the Deskpro provides 8
accessory slots on the motherboard to allow for expansion
cards. This facility is necessary to allow for the
specialized hardware that controls the stepper motors.

One of the most important pieces of hardware in the
system is the motor pulse generator. The motor pulse
generator is a custom designed board that plugs into the
PC and occupies one card slot. Its purpose is to generate
the TTL level pulses needed by the motor power supply to
drive the stepper motors. Each up or down pulse rotates
the motor a fraction of a degree either clockwise or
counterclockwise. This board consists of four dual-mode
rate-programmable pulse channels. Each pulse channel can
produce either an up pulse or a down pulse at any of
32,767 different rates from 0 to 121.178 kHz. In
addition, each pulse channel contains a 24 bit clearable
pulse count register that can be read by the PC in three
8 bit sections. These registers contain the difference in
the number of up and down pulses generated by each
channel since the last clear. Thus up to four stepper
motors, or two siderostats, may be controlled by the
motor pulse generator board. Each motor is completely
rate-programmable and its position is readable by virtue of the pulse count registers.

The motor pulse generator, in turn, sends the up/down pulses to the motor power supplies. Each motor power supply, of which there are two in the system, consists of two 28 volt motor supplies, a 5 volt digital supply, and two waveform synthesizers that generate the sinusoidal waveforms from the up/down pulses necessary for the motor windings. One of the motor power supplies will be slightly modified to allow for mechanically switching between two sets of motors (two siderostats).

The motors themselves are small 4-phase, permanent magnet stepper motors. The siderostat azimuth axis is driven by a 2.9 amps/phase motor, and the elevation axis is driven by a 1.8 amps/phase motor. The higher current, higher torque motor is used to drive the azimuth axis because of the larger mass attached to that axis. Both motors are specified to produce 200 steps per revolution. The waveform synthesizers on the motor power supplies, however, can divide each step into 64 microsteps. Thus each motor can effectively rotate 12,800 microsteps in one revolution. Each motor is further geared down by a gear system on the siderostats by a factor of 2,016:1. This means that one microstep of a motor causes a 0.05 arcsecond rotation of the corresponding axis on the
siderostat. Clearly, then, the resolution of the motor drive components should not be a limiting factor in the control of the siderostats.

It should be clear from the foregoing discussion of the computer and hardware pertaining to the motors that the siderostats are completely controllable by the computer. Since there is this flexibility of computer control, a great variety of servo and tracking algorithms can be developed and modified, if desired, without redesigning hardware. Various portions of the algorithms can be altered to optimize certain characteristics of the system as well.

In addition to the motor and computer hardware described above, there is a small amount of support circuitry necessary for the operation of the system. A precision clock that is synchronized to universal time is available for the computer to be read through a serial port so that the system may be coordinated with sidereal time. Also, interrupt circuitry was constructed that provides a time base for the real-time software and allows for synchronization of all of the computers in the entire interferometer system. A paddle interface is also provided for the computer to allow a user to move the axes of the siderostat under manual control. This paddle serves an important function in testing and operating the siderostat. Eventually, all of the computers in the
system will be tied together by a dedicated local-area network.

All of the above hardware is controlled by the software described below.

B. Software

The COMPAQ Deskpro computer, since it is an IBM-PC clone, offers a very powerful and flexible programming environment. A great variety of programming languages exists for the PC family of computers, so the challenge was to choose a language, or set of languages, in which to write the control software. The final decision was to write the real-time kernel software in assembly language for speed and access to machine dependent features, the individual tasks in Pascal for speed and high-level clarity, and some astronomy support software in Fortran since a whole host of previously programmed routines exists in that language.

The operating system available for the computer does not support multitask operation in real time. Since the siderostat subsystem requires a real-time operating system, a major effort was expended to develop a flexible real-time operating system which would interface with the present operating system and not sacrifice any of the power of the high-level languages chosen. The operating system ultimately developed was written completely in
8086 assembly language and can handle any number of tasks in a simple prioritized hierarchy and allows any task to use the floating-point unit without interfering with other tasks. In addition, while the tasks are running, the disk operating system of the PC is available to allow any user-level program to run in the background. This feature makes available to the user the flexibility of editing and compiling programs while the system is running; such flexibility was not available in the previous implementation of the siderostat control system on the Mark II interferometer. Tasks may communicate with each other and with user-level programs, as well.

The real-time operating system consists of three main routines: a queuer, a scheduler, and a suspender. The queuer prepares tasks according to their priorities for execution. It also determines whether tasks have failed to execute properly or at their scheduled times. The scheduler examines the tasks that have been affected by the queuer and performs the context switch necessary to allow the highest priority task queued to execute. When a task has fully completed execution, it calls the suspender which terminates the task. The suspender then calls the scheduler to get the next task queued for execution. If no tasks are ready, the scheduler returns to the disk operating system to handle the background
programs. Each of the three parts of the real-time operating system will now be described in detail.

Each task has associated with itself a task control block. Each task control block, or tcb, has room for storing the complete register sets of both the 8086 processor and 8087 floating-point unit. Each task also has a status that determines the present state of the task. If a task is dormant, it is shut off and will never be queued for execution. If a task is done, it is available for queueing. When a task is ready, it has been queued or is currently executing. A task status of panic indicates that a task has been queued once, then queued again without terminating execution from the first queueing. A panic situation is serious since the tasks are all periodic in nature, and some tasks must be executed on time for proper operation of the system. If all tasks are running properly, the tasks will not panic. Finally, each task has a timer that determines the period of operation of the task. Given these task characteristics, the queuer, scheduler, and suspender operate as follows:

**Queuer**

The queuer is executed once every 10 milliseconds in response to the external 100 Hz interrupt time base described earlier. Since a task or background program
must have been operating when the interrupt occurred, the queuer determines whether the interrupted program was using the 8087 floating-point unit. If it was, the state of the 8087 is saved in the interrupted task's tcb (the background program is also assigned a tcb). Next, the complete state of the 8086 main processor is saved in the interrupted task's tcb. At this point, the environment of the interrupted task has been completely saved. The queuer then examines the status of each of the tasks starting with the highest priority task. If the task is dormant, the queuer checks the next priority task. If it is not dormant, the task's timer is decremented. If the timer has run out, the queuer checks if the status of the task is still ready (i.e. has not finished executing from the last time it was scheduled). If the status is still ready, a panic situation has occurred and the offending task is dequeued by setting its status to dormant. If a panic has not occurred and the status is done, the timer is reset and the status is set to ready so the scheduler will execute it. The next priority task is then examined in the same manner. When all of the tasks have been examined, the queuer then resets the interrupt hardware and enters the scheduler.

Scheduler -

The scheduler is called by both the queuer and
suspender. The scheduler, like the queuer, examines the status of each task beginning with the highest priority one. If the status is ready, the scheduler restores the 8087 floating-point unit environment if the scheduled task uses the 8087, then restores the 8086 main processor environment. The task then resumes execution from where it left off after it was last interrupted. Again, if no real-time tasks are ready, the scheduler returns to the disk operating system to execute the background program.

**Suspender**

The suspender is called when a task fully completes execution. The suspender restores the initial state of the task so that it is ready for its next periodic queueing. It then calls the scheduler to get the next task ready for execution.

Thus the real-time operating system has been described. A block diagram of the operating system is found in Figure 13. It performs very satisfactorily in practice and presently handles six distinct tasks that fully control two siderostats in real time. Each of these tasks can now be examined in the light of the real-time operating system.

**Description of Tasks**

The six tasks presently used in the siderostat
FIG. 13
control subsystem are all written in the high-level language Pascal. This greatly simplifies the construction of the real-time tracking and servo algorithms, since those routines involve fairly intensive matrix and numeric computations. Also, clarity is preserved since the algorithms are easy to interpret. Some effort, however, was expended to interface the Pascal tasks to the assembly language real-time operating system.

The six tasks in the system are prioritized and named as follows: timer, paddle, servo, track, display, and idle. Each performs a distinct function in the control structure that will now be described.

The highest priority task is the timer. The timer task is scheduled to run once a second. It updates the sidereal and Greenwich times so that they are available for the tracking task to calculate stellar positions. The timer task also calculates the tracking rate from the motor position values passed to it from the tracking task.

The next task in priority is the paddle task. This task reads the state of the paddle and performs the function specified by the buttons pressed on the paddle. The paddle can presently perform three unique functions: run motors at desired speed, change motor positions by desired offset, and adjust present tracking positions by
the desired offset. Data recording is also performed by this task. The paddle task is scheduled to run once every 40 milliseconds.

The third task is the servo task and is one of the most important from the standpoint of the motor control system. The servo task runs once every 100 milliseconds and represents a simple first-order feedback loop with some large-signal nonlinear clipping included to control the dynamics of large changes in motor position. A block diagram of the feedback loop is found in Figure 14a. The servo reads the four pulse count registers on the motor pulse generator board, converts the 24 bit binary numbers to floating point values, and then calculates the error in position from the present position to the desired target position. From the error value, a motor speed is calculated that will cause the motor to reach the desired position. The speed value is clamped to prevent excessively rapid changes in acceleration and velocity so that the motors do not stall and so that starts and stops are smoothly controlled. The real speed values are converted to integer and the integer values are sent to the motor pulse generator to control the speed of the motors. If sidereal tracking is in operation, the servo also performs the one-tenth second motor position interpolation using the tracking rate provided by the timer and the one-second updates calculated by the
MOTOR SERVO

TARGET POSN. + \[ \rightarrow \] \[ K = 2.704 \] \[ \rightarrow \] CLIP \[ \rightarrow \] SPEED

PRESENT POSN. \[ \rightarrow \] \[ \int \]

(A)

TRACK TIMING

\[ \leftarrow 1 \text{ SEC} \rightarrow \]

\[ \uparrow \]

\[ \uparrow \]

\[ \uparrow \]

\[ \uparrow \]

\[ \uparrow \]

\[ \uparrow \]

TIME UPDATES

\[ \text{THIS TRACK CALCULATES MOTOR POSN'S. FOR THIS TIME.} \]

\[ \text{SERVO INTERPOLATES.} \]

(B)

FIG. 14
tracker.

The fourth task, the tracking task, represents the heart of the entire control system software. The tracking task calculates, in real time, a set of motor positions for two siderostats, each represented by their own geometric model, that will cause the siderostats to follow a star across the sky and point the stellar beam into the interferometer. This task runs once per second. It takes a star from the observation list that the user specifies, calculates the star's hour angle and declination using the sidereal time generated by the timer task, transforms the star's coordinates into the various coordinate systems, includes atmospheric refraction effects, and generates the motor positions using the matrix manipulations and formulae derived in Appendix A. The motor positions are calculated for the star for a time two seconds later than the most recent timer tick, and the timer takes those values and calculates the motor velocity (tracking rate) necessary to cause the siderostats to follow the star. The servo uses the tracking rate and one-second position updates to interpolate the motor positions for every one-tenth second. A timing diagram of the track-timer-servo operation appears in Figure 14B. Since the sidereal rate is quite slow, this interpolation scheme yields very
small tracking errors (< 1 arcsecond).

The two lowest priority tasks, display and idle, represent tasks useful to the user. The display task is executed once per second and updates all of the system information on the screen. The screen displays such information as present motor positions, motor speeds, and target positions; paddle status; Greenwich and sidereal times; and star number, observation time remaining, and tracking status. The idle task is run once every 30 seconds and simply calculates the percentage of CPU time remaining after all tasks have been executed. This value is also displayed by the display task.

In addition to the real-time tasks, a set of Pascal and Fortran programs are available to aid in system operation and data reduction. The main control program, STAR, gives the user access to the siderostat control system. It allows the user to operate the motors from the keyboard or paddle, create lists of stars for observation, and examine system dependent and task dependent parameters. Another program, REDUCE, contains code that performs model calculations using a least-squares method and allows storage, examination, and retrieval of models and data. The program FK4UPD performs updating of the FK4 star catalog from epoch 1950 to the present. There are also a number of smaller programs too numerous to mention that perform a variety of support
functions.

As can be seen, a complete hardware/software control system exists to handle the siderostats. What remains to be described are the tests that were performed at Mt. Wilson and in Cambridge to verify that the system does, in fact, perform as desired.
V. Summary of Test

To verify the operation of the siderostat control system, a single siderostat was tested in a setup depicted in Figure 15. The telescope was a Celestron C-5 (5-inch reflecting telescope) with a small sighting scope attached. The telescope pointed directly east at the siderostat, which was aimed west. The test setup was first used on the roof of a building at M.I.T., then dismantled and reassembled at Mt. Wilson. In both cases, the placement of the telescope and siderostat were determined to better than a degree or so. The results of the test that will be presented later were those obtained at the site on Mt. Wilson, since the M.I.T. roof test was basically just a systems integration and operation test, and the atmospheric conditions in the city of Cambridge rate rather poorly for astrometry.

The test that was actually performed is an adaptive model solving procedure that is based upon observed astrometric errors. The procedure works as follows:

1. An initial geometric model is assumed for the siderostat. Typically, for the first observations of a night, the siderostat is assumed to be ideal.

2. A set of stars from the updated FK4 catalog is chosen for observation.

3. The stars are observed in sequential order through the telescope. If the model matches the siderostat exactly, and there are no other
SKY

CLOUD (NOT NEEDED FOR TEST)

STAR (NEEDED FOR TEST)

C-5 TELESCOPE

SIDEROSTAT

TEST SETUP
errors, the star will appear in the crosshairs of the telescope. Since this will rarely, if ever, be the case, the star is moved using the paddle into the crosshairs. The number of motor steps required to move the siderostat is recorded. Motor position errors are used since there is no easy way to accurately and quantitatively determine the actual astrometric error by measuring the star's distance to the crosshairs.

4. A new geometric siderostat model is determined from the motor position errors recorded in step 3.

5. The new geometric model is then used for subsequent observations of stars to determine the usefulness of the new model. This process is repeated as many times as desired to observe changes in model parameters over time and space.

The idea behind this process is to make the siderostat model completely adaptive and to improve the pointing of the siderostat so that astrometric errors are minimized. If done properly, the models solved for in step 4 should accurately reflect the actual parameters of the siderostat and, most desirably from an operational standpoint, allow for pointing to within the desired 10 arcseconds or so necessary for the photon camera star-tracker. The first method of model determination, a least-squares approach, is the one that was actually used at the observation sites and proved moderately successful with some shortcomings. The least-squares method calculates the model that minimizes the mean-square astrometric pointing error referred to the sky. The second method, a Kalman filter approach, promises to be
much more effective and was utilized in simulation tests after noticing that the least-squares method could be improved upon. The Kalman filter minimizes the mean-square error in estimating the parameter vector. It should be emphasized, then, that the two methods minimize the errors in different quantities; the least-squares approach minimizes the astrometric error while the Kalman filter minimizes the parameter vector estimation error. Conceivably, both quantities could be minimized by either method, but with the methods chosen the state equations are quite simple to develop and it is doubtful that using, for example, a Kalman filter to minimize astrometric error would improve performance to any great degree, not to mention the fact that the state equations of such a filter are less obviously obtained. The two methods used, least-squares and Kalman filter, will now be discussed.

Method I - Least-squares Model Determination

The least-squares model determination method attempts to find a geometric siderostat model that minimizes the astrometric error referred to the sky. This method is based upon a linearization of the equations that relate motor position, model parameters, and stellar astrometric error. The complete theoretical basis for the least-squares method can be found in Appendix B.
The following results are based upon observations that were made on the night of November 4, 1985, at Mt. Wilson Observatory.

**Actual Performance of Least-squares Method**

As noted earlier, the least-squares method performs satisfactorily with some reservations. It was determined early on in the testing of the system that the least-squares method exhibits a high degree of cross-correlation amongst pairs of model parameters. The result of this crosstalk is that siderostat models calculated from a group of stars in one area of the sky do not work well for groups of stars in other parts of the sky. This result will be shown experimentally later. For most effective performance of the least-squares method, then, a large group of stars distributed throughout the sky must be observed to minimize this crosstalk phenomenon. Then the resulting model better represents the actual parameters of the siderostat, and the model may be used reasonably effectively for smaller sets of stars that occur within the region specified by the initial large group.

Figures 16 to 25 depict the results of observing first a large set of stars, then smaller subgroups of stars. The figures plot the star positions on one scale according to hour angle and declination. On a smaller
scale using the same axis rulings are plotted the astrometric errors. The smaller error scale magnitude is noted on the bottom of the graph. The squares denote the actual positions of the star in HA and DEC, while the length and direction of the rays out of the squares denote the magnitude (on the smaller scale) and direction (in 'local' az and el) of the astrometric pointing errors. The astrometric az and el errors are defined according to the 'local' coordinate frame defined in Figure B1. For a given star, the magnitude of the total error is simply the square root of the sum of the az error squared and the el error squared. These errors can be summed in this manner because the errors are typically small and the two directions (az an el) are orthogonal. For a group of stars, then, the rms astrometric error is simply these magnitudes summed in a root-mean-square sense. In addition, it should be noted that with this 'local' error coordinate frame there are no singularities in error determination. Even at zenith the magnitude of the error will be uniquely defined (although the direction will not be).

To better understand how accurate the test procedure was, a simple exercise was performed on a single star to determine the human user's accuracy in pointing at the star. Pointing accuracy is dependent somewhat on the eye's ability to find the centroid of the stellar image
in the telescope in the presence of atmospheric distortion and telescope lens aberrations, as well on as the user's ability to move the stellar image with the paddle subject to motor nonlinearities. The actual exercise performed was to observe a single star, move away from the star using the paddle, return to the star and record motor positions, move away again, return and record, and so forth for several iterations and in several directions. The resulting astrometric error is shown in Figure 16. After the mean shift is removed, the resulting rms astrometric error is approximately 2 arcseconds. This indicates that the test procedure should prove acceptable for determining astrometric errors at the sub-10 arcsecond level. Now the actual results of the test procedure will be discussed.

Figure 17 shows the actual astrometric errors that resulted from using an initial siderostat model m702 to observe data set d705. The rms astrometric error that resulted for the set was 35.23 arcseconds. Using d702, then, a new model, m703, was calculated. Figure 18 shows the theoretical astrometric errors that should result if d705 were observed using m703. The rms error should theoretically be 11.71 arcseconds. Figure 19 shows the actual astrometric errors that resulted from using m703 to observe the same set of stars as in d705, but
Star posn. plotted in HA,DEC
Errors referred to Local H2,EL

Lat. HA = 34.2 degs.

Error units (arcsec/div) : x (az) = 2  y (el) = 2

'HALLEY'S COMET'
POINTING TEST

ERROR \sim 2 \text{ sec}

FIG. 16
Star posn. plotted in HA, DEC = DEC (EL) + 90 degs.
Errors referred to Local A2, EL

Lat. HW = 39.2 degs.

Error units (arcsec/div): x (az) = 10  y (el) = 10

M702/D705

ERROR = 35.23 sec

FIG. 17
Astrometric Error - Local AZ and EL

Star posn. plotted in HA, DEC = DEC (el) - 90 degs.
Errors referred to Local AZ, EL

Lat. MH = 33.2 degs.

HA (az)
-5 hrs. 5 hrs.

Error units (arcsec/div) : x (az) = 5  y (el) = 5

M703/D705
ERROR = 11.71 sec

FIG. 18
Star posn. plotted in HA, DEC
Errors referred to Local AZ, EL

Lat. N11 = 34.2 degs.

HR (az)
-5 hrs.

Error units (arcsec/div): x (az) = 10 y (el) = 10

M703/D706
ERROR = 22.51 sec

FIG. 19
displaced in hour angle by approximately one-half hour. This new set was d706, and the rms error was 22.51 arcseconds, approximately twice that theoretically predicted. Some of this error is undoubtedly due to the crosstalk described earlier; the group of stars had drifted across the sky and the model was losing validity. Other portions of the discrepancy could have been due to thermal drifts of the table and optics since the roof of the observation shed had been opened shortly before the observation session. The table was in thermal equilibrium, then exposed to the cool night air.

After observing d706, a new model, m704, was calculated. Then, a small subgroup of stars was selected all having declinations above the latitude of Mt. Wilson. This set, d707, was observed using m704 (which was based on a much larger group of stars). The resulting astrometric errors are shown in Figure 20. The actual rms error was found to be 8.45 arcseconds — within the acceptable limits for automatic operation of the startracker. This demonstrates that a model determined from a large group of stars can represent a small subgroup of stars quite well. Next, the same model, m704, was used to observe the same small set of stars again. This set, d708, is depicted in Figure 21 and shows an actual rms error of 14.01 arcseconds. This observation was made an hour after d707 and demonstrates the drifts associated
M704/D707

ERROR = 8.45 \text{ sec}

FIG. 20
Astronomic Error: Local AZ and EL

Star posn. plotted in HA, DEC
Errors referred to Local AZ, EL

Lat. HW = 34.2 degs.

Error units (arcsec/div): x (az) = 5  y (el) = 5

M704/D708
ERROR = 14.01 sec

FIG. 21
with the system. Next, a set of stars with declinations below the latitude of Mt. Wilson was observed using the same model, m704. This set, d709, is shown in Figure 22 and demonstrates an actual rms error of 8.00 arcseconds. The large set of stars upon which m704 was based apparently represents this small group of stars quite well. The set of stars above Mt. Wilson latitude was observed yet again using m704 (d710 - see Figure 23). The resulting error grew slightly to 16.31 arcseconds. The stars below Mt. Wilson were observed yet again using m704. This data set, d711, shown in Figure 24 represents an rms astrometric error of 12.50 arcseconds - again slightly larger than the previous observation of the same set. Finally, the stars above Mt. Wilson latitude were observed using the same model (m704). This data set, d712, is shown in Figure 25 and the actual rms error was 17.12 arcseconds.

From these results it should be clear that the least-squares method can work if used properly. It does require a significant amount of human interaction since large groups of stars must periodically be observed to keep the rms error down. Since models calculated from star groups in one part of the sky may not work in other parts of the sky, new models must be determined when changing from one area of the sky to another.
Star posn. plotted in HA, DEC
Errors referred to Local AZ, EL

Lat. HW = 34.2 degs.

Error units (arcsec/div): x (az) = 5 y (el) = 5

M704/D709
ERROR = 8.00 sec

FIG. 22
Astrometric Error - Local RZ and EL

Star posn. plotted in RA, DEC
Errors referred to Local RZ, EL

Lat. WH = 39.2 degs.

Error units (arcsec/div) : x (az) = 5  y (el) = 5

M704/D710

ERROR = 16.31 sec

FIG. 23
Star posn. plotted in HA, DEC
Errors referred to Local AZ, EL

Lat. NH = 34.2 degs.

Error units (arcsec/div): x (az) = 5  y (el) = 5

M704/D711
ERROR = 12.50 sec

FIG. 24
M704/D712

ERROR = 17.12 sec

FIG. 25
Consequently, one model will represent a given group of stars ever more poorly as the stars move across the sky. A striking demonstration of how poorly a model calculated from stars in one part of the sky represents stars in another part is shown in Figures 26 and 27. Figure 26 shows a small group of stars before transit (d716). A model was calculated from that set and used to observe a much larger set (d718) as shown in Figure 27. The resulting actual rms errors are huge - 268.05 arcseconds. Thus it should be clear that while a model determined from large sets of stars may work well in different parts of the sky for smaller sets of stars, the reverse may not be the case. A model calculated from a small set of stars poorly represents a large set of stars. The question naturally arises as to whether any improvement can be made to the least-squares method to reduce these error effects.

**Attempted Improvements to Least-squares Method**

As noted earlier, a significant amount of crosstalk exists amongst the parameters of the siderostat model when the model is determined using the least-squares method. This causes some models to represent star positions in one area of the sky much more poorly than in other areas of the sky. This is some indication that the models generated by the least-squares method do not
star posn. plotted in HA, DEC
Errors referred to Local AZ, EL

Lat. NH = 39.2 degs.

Error units (arcsec/div); x (az) = 5  y (el) = 5

M704/D716

ERROR = 12.03 sec

FIG. 26
Star posn. plotted in HA, DEC
Errors referred to Local AZ, EL

Lat. NH = 39.2 degs.

Error units (arcsec/div): x (az) = 100 y (el) = 100

M705/D718
ERROR = 268.05 sec

FIG. 27
actually represent the physical angles present in the siderostat. There are some techniques, however, that will attempt to improve the manner in which the models represent the data. One of these will now be discussed.

In order to form a basis of comparison for the evaluation of model determination methods, a group of data sets from the ones previously discussed (d705 to d712) was chosen. The model determination methods were then used on the data sets to determine the simulated theoretical errors that would result from the calculated models.

One disadvantage of the least-squares method is that a whole set of stars (where the number of stars in the set is greater than four) must be used to calculate a model. In order to minimize the effects of crosstalk in the model determination process, the number of stars must be significantly greater than four (on the order of ten or more), and they must be distributed reasonably evenly throughout the sky. If a new star (not part of the original set) is observed and it is desirable to calculate a new model using the new observation, the entire set including the new star must be used to calculate a model. This represents a considerable computational burden as the number of stars gets large. One possible way around this is to break the entire observation set into smaller subsets, calculate models
from the subsets, and combine the resulting models through an averaging process. The averaging represents a negligible computational burden and as new stars are observed the size of the sets used for model determination remains manageable. In addition, the effects of crosstalk should be reduced since models from all over the sky are averaged.

The results of a simulation using a simple averaging technique are shown in Tables 3 and 4. First, models (mm705 to mm710) were calculated using the least-squares method from each of the data sets (d705 to d710). Then, a simple average model (mm7avg) was calculated using mm705 to mm710 plus 9 other models from the same night not shown. The actual parameter values of these models are shown in Table 3. The theoretical rms astrometric errors that should result are shown in Table 4, along with the average rms error for each model over all the sets of data. As can be seen, one of the individual models (mm706) represents all of the data sets somewhat better than mm7avg. Two others, mm705 and mm710, are comparable to mm7avg in terms of astrometric error, but mm7avg does do much better than mm708 and mm709. Even using this averaging method, however, the astrometric error does not fall within the desired 5 to 10 arcseconds. Also, Table 3 demonstrates the fact that the model parameter values
## Table 3

Summary of Least-squares Simulation - Model Parameter Values
All values are in degrees

<table>
<thead>
<tr>
<th></th>
<th>wFaz</th>
<th>wFel</th>
<th>thetaA</th>
<th>alphaA</th>
<th>alphaE</th>
<th>alphaM</th>
<th>daz</th>
<th>del</th>
</tr>
</thead>
<tbody>
<tr>
<td>mm705</td>
<td>90.1656</td>
<td>0.0532</td>
<td>0.2002</td>
<td>-75.1509</td>
<td>89.9786</td>
<td>0.0175</td>
<td>89.4070</td>
<td>-14.8563</td>
</tr>
<tr>
<td>mm706</td>
<td>90.1686</td>
<td>0.0487</td>
<td>0.2032</td>
<td>-75.1505</td>
<td>90.0100</td>
<td>-0.0176</td>
<td>89.4217</td>
<td>-14.8562</td>
</tr>
<tr>
<td>mm707</td>
<td>90.0817</td>
<td>0.0394</td>
<td>0.1428</td>
<td>-75.1454</td>
<td>89.6022</td>
<td>0.4174</td>
<td>89.2967</td>
<td>-14.8652</td>
</tr>
<tr>
<td>mm708</td>
<td>90.3146</td>
<td>0.0787</td>
<td>0.2941</td>
<td>-75.1429</td>
<td>90.5443</td>
<td>-0.5859</td>
<td>89.5497</td>
<td>-14.8472</td>
</tr>
<tr>
<td>mm709</td>
<td>90.0601</td>
<td>-0.0353</td>
<td>0.1295</td>
<td>-75.2375</td>
<td>91.3131</td>
<td>-1.7367</td>
<td>90.6191</td>
<td>-14.7991</td>
</tr>
<tr>
<td>mm710</td>
<td>90.2153</td>
<td>0.0478</td>
<td>0.2330</td>
<td>-75.1526</td>
<td>90.2425</td>
<td>-0.2746</td>
<td>89.5061</td>
<td>-14.8531</td>
</tr>
<tr>
<td>Average</td>
<td>90.1677</td>
<td>0.0388</td>
<td>0.2005</td>
<td>-75.1633</td>
<td>90.2818</td>
<td>-0.3633</td>
<td>89.9633</td>
<td>-14.8462</td>
</tr>
<tr>
<td>Std. Dev. (x10E-2)</td>
<td>9.253</td>
<td>3.865</td>
<td>6.032</td>
<td>3.655</td>
<td>59.36</td>
<td>75.07</td>
<td>49.08</td>
<td>2.379</td>
</tr>
</tbody>
</table>

(Note: mm7avg is a simple average of all least-squares models generated from all the observation data sets taken for the night, not just the averages of mm705 to mm710.)

The above table represents the actual values of each of the parameters of the siderostat model. The models are produced using the ideal model as input and corresponding data set (i.e. mm705 is calculated from d705, mm706 from d706, and so forth) and are the result of 5 iterations of the least-squares procedure. It is irrelevant that the ideal model is used as input since the output model after 5 iterations is independent of the input model. Note the large variances of some of the parameters. Consult Table 2 and Chapter 3 in general for parameter definitions.)
Table 4

<table>
<thead>
<tr>
<th></th>
<th>mm705</th>
<th>mm706</th>
<th>mm707</th>
<th>mm708</th>
<th>mm709</th>
<th>mm710</th>
<th>mm7avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>d705</td>
<td>11.71</td>
<td>23.02</td>
<td>56.20</td>
<td>96.66</td>
<td>206.39</td>
<td>45.14</td>
<td>43.35</td>
</tr>
<tr>
<td>d706</td>
<td>22.51</td>
<td>10.37</td>
<td>58.79</td>
<td>91.58</td>
<td>366.18</td>
<td>36.83</td>
<td>45.45</td>
</tr>
<tr>
<td>d707</td>
<td>21.29</td>
<td>8.45</td>
<td>1.48</td>
<td>21.30</td>
<td>48.76</td>
<td>18.57</td>
<td>19.63</td>
</tr>
<tr>
<td>d708</td>
<td>31.60</td>
<td>14.01</td>
<td>10.82</td>
<td>2.52</td>
<td>100.67</td>
<td>4.90</td>
<td>13.21</td>
</tr>
<tr>
<td>d709</td>
<td>23.33</td>
<td>8.00</td>
<td>103.85</td>
<td>166.61</td>
<td>0.94</td>
<td>41.52</td>
<td>7.27</td>
</tr>
<tr>
<td>d71c</td>
<td>34.43</td>
<td>16.31</td>
<td>13.95</td>
<td>4.36</td>
<td>131.59</td>
<td>0.24</td>
<td>13.55</td>
</tr>
<tr>
<td>d711</td>
<td>28.66</td>
<td>12.50</td>
<td>98.29</td>
<td>153.40</td>
<td>43.40</td>
<td>37.40</td>
<td>10.07</td>
</tr>
<tr>
<td>d712</td>
<td>34.63</td>
<td>17.12</td>
<td>15.90</td>
<td>10.68</td>
<td>156.85</td>
<td>7.51</td>
<td>16.60</td>
</tr>
<tr>
<td>Avg.</td>
<td>26.02</td>
<td>13.72</td>
<td>44.91</td>
<td>68.39</td>
<td>131.85</td>
<td>24.01</td>
<td>21.14</td>
</tr>
</tbody>
</table>

The above table represents the root-mean-square astrometric errors for the named data sets that resulted from using the models produced by the least-squares method. The data sets d705 to d712 are sequentially related in time - d705 being the earliest and d712 being the latest. The least-squares output model mm705 is based solely on data set d705, mm706 solely on d706, and so forth up to mm710. d711 and d712 are included to observe the evolution of errors over a longer period of time.

Note that the average rms error for the data sets does not decrease as time progresses (from mm705 to mm710). Some of the models better represent the data sets than others, but none approach that necessary for proper operation of the photon camera star-tracker. The 'average' model, mm7avg, which in some sense represents all of the data, still does not represent all of the data as well as, for example, mm706. This suggests that the least-squares method, even if it attempts to include more and more data, is not the best approach.
have large standard deviations - on the order of 2 to 45 arcminutes. Since it is highly unlikely that the physical angles in the siderostat have such large drifts, the parameter values shown in the table are probably not related to their physical values, but are the result of computational effects due, again, mostly to crosstalk.

It does not appear that the least-squares method is the optimal means of determining models from observation data. Only in certain instances are the design objectives of 5 to 10 arcseconds achieved. Even attempts at improving the method by averaging over groups of models prove somewhat futile. A more promising approach, the Kalman filter, will now be considered.

Method II - Kalman Filter Model Determination

From examinations of the data obtained using the least-squares model determination process it was thought that improvements could be made in the model determination method by using a fundamentally different approach. This second approach, a Kalman filter method, is considerably different in several respects. First of all, the Kalman filter attempts to minimize the errors in estimating the parameter vector itself; the least-squares method, on the other hand, determines a model that minimizes the astrometric error over the observation set. Secondly, the Kalman filter operates sequentially. In
other words, each new observation is used as it is obtained. There is no need to accumulate a set of observations as in the least-squares method, and the estimates of the parameter vector made by the Kalman filter are inherently based upon all previous observations.[8] Appendix C outlines the Kalman filter developed for determining the model parameters.

The Kalman filter requires certain characteristics of the parameters being evaluated to be determined a priori. These characteristics, namely the observation noise and the plant noise, must be given values that reflect the physical processes involved. The observation noise magnitude is approximately known from the pointing test that was done as shown in Figure 16. The plant noise was based in a somewhat ad hoc method upon what would seem to be reasonable upper bounds in the drifts of the various siderostat model parameters. These choices for noise values may or may not be entirely correct, but the results discussed below demonstrate that the Kalman filter achieves significantly better results than the least-squares method.

Tables 5 and 6 depict the theoretical results of using a Kalman filter on the same sets of data as before. Table 5 shows the model parameter values for each of the models (mk705 to mk706) that were produced by using the Kalman filter. As noted in the table, mk705 is based upon
Table 5

Summary of Kalman Filter Simulation - Model Parameter Values
All values are in degrees

<table>
<thead>
<tr>
<th></th>
<th>wFaz</th>
<th>wFel</th>
<th>thetaA</th>
<th>alphaA</th>
<th>alphaE</th>
<th>alphaM</th>
<th>daz</th>
<th>del</th>
</tr>
</thead>
<tbody>
<tr>
<td>mk705</td>
<td>90.1676</td>
<td>0.0549</td>
<td>0.2014</td>
<td>-75.1503</td>
<td>90.0106</td>
<td>-0.0213</td>
<td>89.4285</td>
<td>-14.8577</td>
</tr>
<tr>
<td>mk706</td>
<td>90.1669</td>
<td>0.0525</td>
<td>0.2019</td>
<td>-75.1484</td>
<td>89.9929</td>
<td>0.0006</td>
<td>89.4150</td>
<td>-14.8563</td>
</tr>
<tr>
<td>mk707</td>
<td>90.1664</td>
<td>0.0527</td>
<td>0.2016</td>
<td>-75.1484</td>
<td>89.9887</td>
<td>0.0050</td>
<td>89.4133</td>
<td>-14.8565</td>
</tr>
<tr>
<td>mk708</td>
<td>90.1656</td>
<td>0.0536</td>
<td>0.2013</td>
<td>-75.1467</td>
<td>89.9819</td>
<td>0.0120</td>
<td>89.4104</td>
<td>-14.8568</td>
</tr>
<tr>
<td>mk709</td>
<td>90.1668</td>
<td>0.0544</td>
<td>0.2017</td>
<td>-75.1461</td>
<td>89.9887</td>
<td>0.0045</td>
<td>89.4130</td>
<td>-14.8571</td>
</tr>
<tr>
<td>mk710</td>
<td>90.1666</td>
<td>0.0548</td>
<td>0.2019</td>
<td>-75.1449</td>
<td>89.9883</td>
<td>0.0043</td>
<td>89.4134</td>
<td>-14.8577</td>
</tr>
<tr>
<td>Average</td>
<td>90.1667</td>
<td>0.0538</td>
<td>0.2016</td>
<td>-75.1475</td>
<td>89.9992</td>
<td>0.0009</td>
<td>89.4156</td>
<td>-14.8570</td>
</tr>
<tr>
<td>Std. Dev.</td>
<td>1.155</td>
<td>1.050</td>
<td>0.2503</td>
<td>2.074</td>
<td>9.914</td>
<td>11.47</td>
<td>6.631</td>
<td>0.5654</td>
</tr>
</tbody>
</table>

(x10E-3)

The above table represents the actual values of each of the parameters of the siderostat model. The models represent the inclusion of larger amounts of data as the model number increases. Thus mk705 is calculated from data set d705 with the ideal model as the input, mk706 is calculated from d706 using mk705 as input, mk707 is calculated from d707 using mk706 as input, and so forth up to mk710. Note that the parameter variances are significantly smaller than those found in the least-squares model solutions, suggesting that the Kalman filter solutions are better behaved and more representative of the physical parameters. Consult Table 2 and Chapter 3 in general for parameter definitions.
Table 6
Kalman Filter Data Simulation Results -
RMS Astrometric Errors in Arcseconds
Night of November 4, 1985

<table>
<thead>
<tr>
<th></th>
<th>mk705</th>
<th>mk706</th>
<th>mk707</th>
<th>mk708</th>
<th>mk709</th>
<th>mk710</th>
</tr>
</thead>
<tbody>
<tr>
<td>d705</td>
<td>16.53</td>
<td>22.54</td>
<td>21.53</td>
<td>27.46</td>
<td>26.47</td>
<td>29.76</td>
</tr>
<tr>
<td>d706</td>
<td>32.44</td>
<td>10.73</td>
<td>10.82</td>
<td>12.32</td>
<td>11.88</td>
<td>13.95</td>
</tr>
<tr>
<td>d707</td>
<td>32.01</td>
<td>7.85</td>
<td>7.78</td>
<td>9.09</td>
<td>8.69</td>
<td>11.28</td>
</tr>
<tr>
<td>d708</td>
<td>41.16</td>
<td>13.60</td>
<td>13.84</td>
<td>7.81</td>
<td>8.20</td>
<td>5.72</td>
</tr>
<tr>
<td>d709</td>
<td>34.88</td>
<td>6.98</td>
<td>7.78</td>
<td>6.68</td>
<td>4.70</td>
<td>5.48</td>
</tr>
<tr>
<td>d710</td>
<td>43.29</td>
<td>15.80</td>
<td>16.00</td>
<td>8.90</td>
<td>9.57</td>
<td>5.16</td>
</tr>
<tr>
<td>d711</td>
<td>39.69</td>
<td>12.21</td>
<td>13.27</td>
<td>10.42</td>
<td>10.33</td>
<td>9.38</td>
</tr>
<tr>
<td>d712</td>
<td>42.93</td>
<td>16.65</td>
<td>16.72</td>
<td>10.20</td>
<td>10.85</td>
<td>6.90</td>
</tr>
<tr>
<td>Avg.</td>
<td>35.37</td>
<td>13.30</td>
<td>13.47</td>
<td>11.61</td>
<td>11.34</td>
<td>10.95</td>
</tr>
</tbody>
</table>

The above table represents the root-mean-square astrometric errors for the named data sets that resulted from using the models produced by the Kalman filter. The data sets d705 to d712 are sequentially related in time - d705 being the earliest and d712 being the latest. The Kalman filter output model mk705 is based upon d705 with the ideal model as the input, mk706 is based upon d706 with mk705 as input, mk707 is based upon d707 with mk706 as input, and so forth up to mk710. d711 and d712 are included to observe the evolution of errors over a longer period of time.

Note that the average rms error for the data sets decreases as the Kalman filter progresses from mk705 to mk710. The average error is significantly smaller than the corresponding errors generated by the least-squares method. The average error also approaches that necessary for proper operation of the photon camera star-tracker.
d705 using the ideal model as input. mk706 is based upon d706 with mk705 as input. This process was continued up to mk710. Table 6 shows the resulting astrometric errors along with their average values over the data sets. Note that the average rms astrometric error decreases as more and more data is included in the parameter estimate. The rms error even approaches the 10 arcsecond level needed for proper operation of the photon camera star-tracker. Note also that the standard deviations of the model parameters are much smaller (on the order of 1 to 43 arcseconds). These deviations in parameter values are comparable to those that might be expected in a physical system such as the siderostat over the time span of the 2 hours or so that the models represent. There is reason to believe, then, that the siderostat model parameters determined by the Kalman filter are far more representative of the physical angles than those determined by the least-squares method.

Another example of the improvement of the Kalman filter over the least-squares method is shown in Figure 28. This figure is the result of using a small set of stars (d716 - Figure 26) in one part of the sky to calculate a model that is then used to observe a larger set of stars (d718). The model (mt716) was generated using the Kalman filter from d716 with the ideal
Astrometric Error - Local AZ and EL

Star posn. plotted in HA, DEC

Errors referred to Local AZ, EL

Lat. NH = 34.2 degs.

Error units (arcsec/div): x (az) = 30  y (el) = 30

MT716/D718
ERROR = 93.71 sec

FIG. 28
parameter model as input. The resulting rms astrometric error is 93.71 arcseconds — an improvement of almost a factor of three relative to the 268.05 arcseconds achieved by the least-squares method for the same data sets (see Figures 26 and 27). This demonstrates that a model generated from a given set of data by the Kalman filter represents the entire sky more reasonably than a model generated from the same set of data using the least-squares method.

It appears that the Kalman filter will achieve the design objectives quite admirably without any of the restrictions found in the least-squares method. While more work can certainly be done in optimizing the filter, the advantages even at the present stage should be clear. The filter achieves the desired pointing accuracy, produces very stable parameter models, and is highly conducive to operation in a real-time environment since observation data may be used as it becomes available in a sequential, rather than batch, mode. In the future the Kalman filter will be implemented in the control system as the principal method of determining models.
VI. Conclusions and Extensions

This thesis has discussed the design, implementation, and performance of a siderostat control subsystem that will eventually be integrated with the Stellar Astrometric Interferometer being constructed at Mt. Wilson Observatory. This interferometer will eventually be capable of performing astrometric measurements at the milliarcsecond level of precision. For automatic operation of such an instrument, though, the siderostat subsystem must achieve open-loop pointing at the 5 to 10 arcsecond level of accuracy.

To meet these somewhat stringent performance objectives, a complete control architecture had to be specified. A complete geometric model had to be defined to specify the siderostats. The parameters of this model correspond to physical angles in the siderostat and yield closed-form solutions to motor values for the purposes of real-time control. Given that the siderostats are produced with great care and precision, the model should eliminate all errors except for small nonlinearities such as motor backlash and gear eccentricities. These nonlinear errors should be on the order of a few arcseconds, so the performance objectives are obtainable.

A complete set of hardware and software support was constructed. A motor pulse generator that is rate-
programmable and position-readable was designed and built to interface to a high performance microcomputer. A flexible real-time operating system and control software were written that provide user-friendly access to the siderostat subsystem.

To demonstrate that performance objectives were met, a single siderostat was tested at Mt. Wilson. A least-squares model determination method was utilized to adaptively control the siderostat. The least-squares method attempts to minimize the rms astrometric pointing error. Open-loop pointing was achieved at the sub-10 arcsecond level for small groups of stars when the model used was determined from a large, evenly distributed set of stars. Model effectiveness deteriorated, however, when models determined for one area of the sky were used in other parts of the sky or at later times. This deterioration was primarily caused by a significant amount of cross-correlation amongst parameters determined by the least-squares method. In other words, some parameters of the siderostat were poorly determined in one part of the sky, but well determined in other parts. Attempts, such as model averaging, were made to improve the least-squares method, with little success.

As a result of some of the performance restrictions on the least-squares method, a Kalman filter model
determination method was evaluated in simulated tests with some success. The objective of sub-10 arcsecond pointing with the filter is achievable and the models determined exhibit a high degree of stability. The Kalman filter, possibly due to the fact that it minimizes the parameter vector estimation error rather than the astrometric error, appears to yield significant improvements in performance over the least-squares method.

Before the siderostat subsystem is finally integrated with the complete interferometer, a significant amount of work can be done to improve siderostat performance. The actual magnitudes of the drifts present in the siderostat model parameters can be determined and used to optimize the Kalman filter. One test that might yield these values is to track a single star across the sky and record motor positions every few seconds. The residual errors would indicate the magnitudes of the drifts present. Also, the Kalman filter might be driven with known noise sources to determine the magnitudes of error due strictly to computational factors and modelling issues (such as observability and controllability). These tests would give a better indication of exactly how the filter will perform. With an optimized Kalman filter, the system should perform well within the 5 to 10 arcsecond level of open-loop
pointing desired.
Appendix A

Motor Control Equations

As noted in chapter 3, the geometric model specified to model the siderostats allows for a closed-form solution of motor positions given the mirror normal in Mt. Wilson coordinates and the geometric model parameters. This relationship will be derived in this appendix.

As can be seen from Figures 7 and 8 in the text, $e_{T_m}$ and $a_{T_e}$ (the transformations from elevation frame to mirror frame and azimuth frame to elevation frame) are functions of the motor positions, whereas $w_{T_a}$ (and $a_{T_w}$ = $(w_{T_a})^t$) is not. $w_P$, the desired mirror normal in Mt. Wilson coordinates, is determined from the hour angle and declination of the star and the latitude of Mt. Wilson.

From $w_P$, then, the vector $a_P$ can be determined as shown in equation 1.1.

$$a_P = a_{T_w} w_P \quad (1.1)$$

$a_P$ represents the desired mirror normal expressed in the coordinate frame fixed to the siderostat azimuth axis. $a_P$ can also be determined from the two motor dependent transformations, $a_{T_e}$ and $e_{T_m}$, and $m_P$, the fixed mirror normal expressed in the mirror coordinate frame. This $a_P$, which will be denoted $a'_P$ for clarity, is found from equation a2.
\[ a_P' = a_T e_T^m_p \quad (a2) \]

In order for the siderostat to point properly, \( a_P \) must be equal to \( a_P' \). In other words, the motor positions (which constrain \( a_T e_T \) and \( e_T^m_p \)) must be such that siderostat points the mirror normal in the proper direction. \( a_P' \) is dependent on motor positions as shown in Figure A1. \( a_P \) is found from the time and star coordinates. Thus, from \( a_{P_x}, a_{P_y}, \) and \( a_{P_z}, \theta_1 \) and \( \theta_2 \) (the siderostat axis angles) can be determined according to Figure A2. From \( \theta_1 \) and \( \theta_2 \), then, \( m_1 \) and \( m_2 \) (the motor positions) can be determined as shown. Thus the solution is exact, closed-form, and highly tractable for use in a real-time control system based upon a digital computer.

Note, again, that in the figures Sangle and Cangle correspond to \( \sin(\text{angle}) \) and \( \cos(\text{angle}) \), respectively.
MOTOR CONTROL

\[ A_P' = A_T E T_M^M P \]

\[ A_P' = \begin{bmatrix}
-c\theta_1 s\theta_2 c\alpha_M & -s\theta_1 c\alpha_E c\theta_2 c\alpha_M + s\theta_1 s\alpha_E s\alpha_M \\
-s\theta_1 s\theta_2 c\alpha_M + c\theta_1 c\alpha_E c\theta_2 c\alpha_M - c\theta_1 s\alpha_E s\alpha_M \\
&s\alpha_E c\theta_2 c\alpha_M + s\alpha_M c\alpha_E
\end{bmatrix} \]

\[ A_P' = A_P \]

\[ A_P' \text{ is of the form:} \]

\[ A_P' = \begin{bmatrix}
A s\theta_1 + B c\theta_1 \\
B s\theta_1 - A c\theta_1 \\
D c\theta_2 + E
\end{bmatrix} \]

WHERE:

\[ A = s\alpha_E s\alpha_M - c\alpha_E c\theta_2 c\alpha_M \]

\[ B = -s\theta_2 c\alpha_M \]

\[ D = s\alpha_E c\alpha_M \]

\[ E = s\alpha_M c\alpha_E \]

FIG. A1
THUS:

\[ \cos \theta_2 = \frac{A P_z - E}{D} \]
\[ \sin \theta_2 = \sqrt{1 - (\cos \theta_2)^2} \quad \text{since} \quad -90^\circ \leq \theta_2 \leq 90^\circ \]

\[ \Rightarrow \theta_2 = \text{ATAN2}(\sin \theta_2, \cos \theta_2) \]

ALSO:

\[ \sin \theta_1 = \frac{A^A P_x + B^A P_y}{A^2 + B^2} \]
\[ \cos \theta_1 = \frac{B^A P_x - A^A P_y}{A^2 + B^2} \]

\[ \Rightarrow \theta_1 = \text{ATAN2}(\sin \theta_1, \cos \theta_1) \]

MOTOR POSITIONS:

\[ M_1 = \frac{1}{K_1} (\theta_1 - D_1) \]
\[ M_2 = \frac{1}{K_2} (\theta_2 - D_2) \]

FIG. A2
Appendix B

Least-squares Siderostat Parameter Vector Optimization

The least-squares method of siderostat model determination as mentioned in Chapter 5 attempts to minimize the pointing error referred to the sky (the astrometric error). The method relies upon a set of observations of a group of stars and uses the motor position errors generated during the observations to produce an estimate of the siderostat model that minimizes the astrometric error over the observation set.

The astrometric error is defined for two orthogonal directions relative to the star—a 'local' azimuth and elevation. Figure B1 shows how these two directions are defined. A coordinate frame, denoted $L$, is fixed to $w_{Sr}$, the refracted star vector expressed in the Mt. Wilson coordinate frame. $L_x$ is parallel to $w_{Sr}$, $L_z$ points in the direction of increasing elevation, and $L_y$ points in the direction of increasing azimuth. A vector that points in a direction other than $w_{Sr}$ thus has a 'local' azimuth error and a 'local' elevation error relative to $w_{Sr}$. This coordinate frame $L$ is orthogonal all over the sky and does not require making elevation corrections for azimuth errors as would be the case if errors were defined by the azimuth and elevation in Mt. Wilson coordinates. Also, the frame $L$ is fixed to the star and represents a useful
means of defining the pointing errors relative to the star. To determine the error, then, the transformation $L_{Tw}$ is calculated from $WS_r$ (using $WS_r$az and $WS_r$el, where $WS_r$az and $WS_r$el correspond to the Mt. Wilson azimuth and elevation of the star, respectively). The siderostat motor positions and model determine a pointing direction, $WS'_r$, that can be referred to frame L as $LS'_r$ by multiplying $WS'_r$ by the transformation $LT_w$. If the motor positions and model are precisely correct, the pointing vector would be coincident with $WS_r$ (i.e. the siderostat is pointing perfectly) and the az and el errors would be zero. Of course, in general this will not be the case, so the pointing vector will result in az and el errors referred to frame L. It is these errors that will be used in the least-squares model optimization method.

The theoretical basis for the least-squares method is shown in Figures B2 and B3. Again, this method attempts to determine a model that minimizes the error between where the star is and where the siderostat points, or the astrometric error. This method is different from the Kalman filter method described in Appendix C which attempts to minimize the model estimation error. Note that a vector subscripted with $<$(i.e. $WS'_r$) is the 2-vector that has as its first component the azimuth and its second component the
elevation of the corresponding vector in the specified frame. In addition, as seen in Figure B3 the least-squares method operates in a type of 'batch' mode. In other words, a set of $n$ observations (where $n \geq 4$) must be used or the system is underconstrained. Four observations will uniquely constrain the eight parameters. If the least-squares method is iterated (which it typically is since the model is linearized) and the observation set is large, the computation time becomes considerable.

Note, again, that in the figures $\text{Sangle}$ and $\text{Cangle}$ correspond to $\sin(\text{angle})$ and $\cos(\text{angle})$, respectively.
\[ \begin{pmatrix}
\text{CAZCEL} & \text{SAZCEL} & \text{SEL} \\
-\text{SAZ} & \text{CAZ} & 0 \\
-\text{CAZCEL} & -\text{SAZCEL} & \text{CEL}
\end{pmatrix} \]

**FIG. BI**
LEAST-SQUARES

$L_{S_\theta}^{R'}$ is a function of motor positions and parameter vector:

$$L_{S_\theta}^{R'} = w(\bar{\theta}, \bar{q})$$

$$\begin{bmatrix} M_1 \\ M_2 \end{bmatrix}^{AZ}_{EL}$$

$$\bar{q} = \begin{bmatrix} w \\ F_{AZ} \\ F_{EL} \\ \Theta_A \\ \alpha_A \\ \alpha_E \\ \alpha_M \\ D_1 \\ D_2 \end{bmatrix}$$

LINEARIZING:

$$L_{S_\theta}^{R'}(=\bar{\theta})$$

$$\begin{bmatrix} \Delta \bar{\theta} \\ \Delta \bar{q} \end{bmatrix}$$

$$\begin{bmatrix} w \\ W_{\Theta} \\ w \\ W_{\bar{q}} \end{bmatrix}$$

$$\bar{\theta}_o = \text{TARGET POSN.}$$

$$\Delta \bar{\theta} = \bar{\theta}_{\text{MEASURED}} - \bar{\theta}_o$$

$$\bar{q}_o = \text{INPUT MODEL}$$

$$\Delta \bar{q} = \bar{q}_{\text{NEW}} - \bar{q}_o$$

$$\Rightarrow \Delta L_{S_\theta}^{R} - W_{\Theta} \Delta \bar{\theta} = W_{\bar{q}} \Delta \bar{q}$$

$$\begin{bmatrix} \Delta L_{S_\theta}^{R} \\ \Delta L_{S_\theta}^{R} \\ \Delta L_{S_\theta}^{R} \end{bmatrix}$$

FIG. B2
FORCE $\Delta^L S^r_\prec \rightarrow \bar{O}$ BY MOVING STAR INTO CROSSHAIRS.
GIVEN $N$ OBSERVATIONS:

\[
\begin{bmatrix}
(W_\theta \Delta \bar{\theta})_1 \\
(W_\theta \Delta \bar{\theta})_2 \\
\vdots \\
(W_\theta \Delta \bar{\theta})_N
\end{bmatrix}
- \begin{bmatrix}
W_{q_1} \\
W_{q_2} \\
\vdots \\
W_{q_N}
\end{bmatrix}
= \Delta \bar{q}
\]

\[
\Rightarrow \Delta \bar{q} = -(B^T B)^{-1} B^T A
\]

\[
\bar{q}_{\text{NEW}} = \bar{q}_o + \Delta \bar{q}
\]

FIG. B3
Appendix C

Kalman Filter Siderostat Parameter Vector Optimization

The Kalman filter described in Figures C1 and C2 represents an iterated-extended filter. The filter attempts to minimize the mean-square error in estimating the parameter vector, rather than the astrometric error as done in the least-squares method of Appendix B. As shown in the figures, the system model is assumed to be a Wiener process that defines the evolution of the model parameter vector, q, over time. The noise process, z, reflects the uncertainties and drifts present in the siderostat parameter model. The observation, or measurement, model reflects the uncertainty in using the eye and a telescope's crosshairs to measure the astrometric pointing error.

In the actual implementaion of the filter, the three update equations (Kalman gain, error covariance, and state estimate) are iterated from i = 0 to i = 2 for each k. These iterations are necessary to eliminate errors due to linearizing the function h that relates parameter vector and motor positions for a given star. It should be noted that the Kalman filter operates sequentially on each observation as it becomes available; it does not need to wait until a set is accumulated, as is the case in the least-squares method.
SYSTEM MODEL:
\[ \tilde{q}_k = \tilde{q}_{k-1} + \tilde{Z}_{k-1} \quad \tilde{Z}_k \sim N(\bar{\theta}, Q_k) \]

MEASUREMENT MODEL:
\[ \bar{\Theta}_k = \tilde{h}_k(\tilde{q}_k) + \tilde{v}_k \quad \tilde{v}_k \sim N(\bar{\theta}, R_k) \]

INITIAL CONDITIONS:
\[ E[\tilde{q}(0)] = \tilde{q}_{\text{ideal}} \]
\[ E[(\tilde{q}(0) - \tilde{q}_{\text{ideal}})(\tilde{q}(0) - \tilde{q}_{\text{ideal}})^T] = P_0 \]

OTHER ASSUMPTIONS:
\[ E[\tilde{Z}_k \tilde{v}_j^T] = 0 \quad \text{for all } j, k \]

STATE ESTIMATE EXTRAPOLATION:
\[ \hat{\tilde{q}}_k(-) = \hat{\tilde{q}}_{k-1}(+) \]

ERROR COVARIANCE EXTRAPOLATION:
\[ P_k(-) = P_{k-1}(+) + Q_{k-1} \]

KALMAN FILTER
FIG. Ci
KALMAN GAIN MATRIX:

\[ K_{k,i} = P_k(+)H_k^T(\hat{\alpha}_{u,k,i}(+))H_k(\hat{\alpha}_{u,k,i}(+))P_k(+)H_k^T(\hat{\alpha}_{u,k,i}(+) + R_k)^{-1} \]

ERROR COVARIANCE UPDATE:

\[ P_{k,i+1}(+) = [I - K_{k,i}H_k(\hat{\alpha}_{u,k,i}(+))]P_k(-) \]

STATE ESTIMATE UPDATE:

\[ \hat{\alpha}_{u,k,i+1}(+) = \hat{\alpha}_{u,k}(-) + K_{k,i} [\bar{\Theta}_k - H_k(\hat{\alpha}_{u,k,i}(+)) - H_k(\hat{\alpha}_{u,k,i}(+))(\hat{\alpha}_{u,k}(-) - \hat{\alpha}_{u,k,i}(+))] \]

\[ \hat{\alpha}_{u,k,0}(+) = \hat{\alpha}_{u,k}(-) \]

WHERE:

\[ H_k(\hat{\alpha}_{u,k,i}(+)) \equiv \frac{\partial h_k}{\partial \alpha_{u,k}} \bigg|_{\alpha_{u,k} = \hat{\alpha}_{u,k,i}(+)} \]

\[ \bar{\Theta} = \begin{bmatrix} M_1 \\ M_2 \end{bmatrix} \]

\[ \hat{\alpha}_{u} = \begin{bmatrix} w_{Faz} \\ w_{Fel} \\ \Theta_A \\ \alpha_A \\ \alpha_E \\ \alpha_M \\ D_1 \\ D_2 \end{bmatrix} \]

FIG. C2
BIBLIOGRAPHY


In addition, the following theses are useful for background information regarding the interferometer as a whole:

