An inverse-kinematic approach to dual-stage servo control for an optical pointing system

Eric D. Miller

Facebook, Inc., Menlo Park, CA, USA

ABSTRACT

In free-space optical communication systems, narrow beam diameters necessitate precision pointing over a relatively wide field of regard. Control systems with such large dynamic range requirements sometimes employ multi-stage architectures, where a "coarse" mechanism provides low-bandwidth control over a large range of travel, and a "fine" mechanism provides high-bandwidth disturbance rejection over a limited range of travel. In such systems, the two stages must be coordinated in their dynamic response, so as to avoid undesirable coupling and even interference with each other. This topic has been studied in various literature – especially in the field of hard disk drives, but also for optical pointing systems. However, most of this literature considers systems where the output of the two stages combines by simple linear sum. Dual-stage control becomes even more challenging when the two stages combine via nonlinear kinematic relationships, as they would in an optical pointing system employing multiple gimbaled steering mirrors. This paper presents an approach for handling these nonlinear kinematic effects, and demonstrates the viability of this approach via simulation.

Keywords: pointing, PAT, line-of-sight control, kinematics, dual-stage control, gimbaled mirrors, FSM, heliostat

1. INTRODUCTION

In free-space optical communication (FSOC) systems, narrow beam diameters necessitate precision pointing (down to single-digit microradians) over a relatively wide field of regard (up to hemispherical coverage).[1] Control systems with such large dynamic range requirements sometimes employ multi-stage architectures, where a "coarse" mechanism provides low-bandwidth control over a wide range of travel, and a "fine" mechanism provides higher-bandwidth precision positioning and disturbance rejection over a much narrower range of travel. The main challenge of this approach is that the two stages must be coordinated in their dynamic response, so as to avoid undesirable coupling and even interfering with each other.

This topic of dual-stage control has been studied in various literature, perhaps most heavily in the field of hard disk drives (HDD). In a typical HDD system, there is a read/write head that needs to be precisely positioned over a spinning disk. The read/write head is articulated by a voice-coil-actuated rotary stage, coupled with an additional piezo-electric micro-actuator stage. This arrangement has been used to achieve higher bandwidth, faster track seeking, and improved disturbance rejection, all over the full range of travel, and for less input power, compared to single-stage systems.[2–4] These results have been achieved with various different control strategies,[5–10] but common among virtually all of these studies was the fact that the system consisted of two parallel 1-degree-of-freedom plants, whose outputs combine via simple linear sum. Figure 1 illustrates the dual-stage hard-disk drive, along with a basic master-slave control architecture.

Dual-stage control has been applied to optical pointing systems as well, in the form of redundant gimbal configurations, or a gimbal working with a fast-steering mirror (FSM) in an "image-motion compensation" system.[12,13] As with the HDD systems above, the plants in these two optical pointing systems also combine linearly.

Author contact: ericmi@fb.com

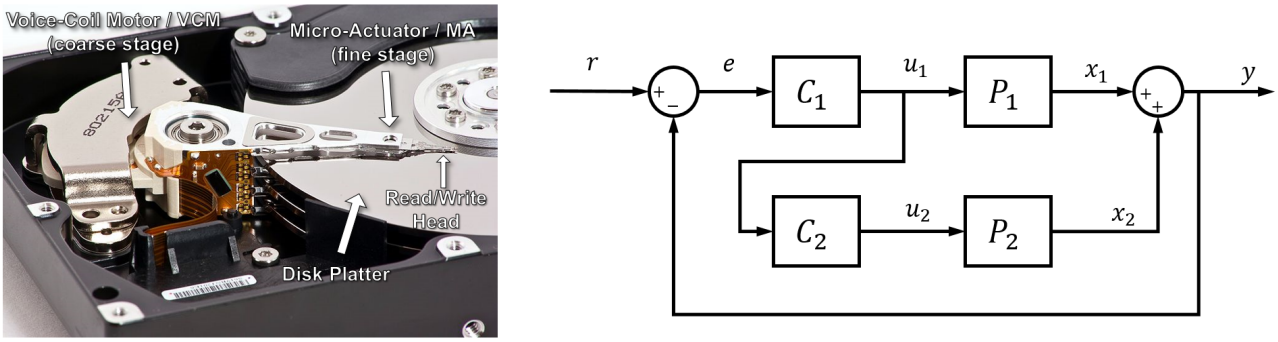

Figure 1. Dual-Stage Hard-Disk Drive System. The image on the left (adapted from Ref. 11) shows the voice-coil motor coarse stage, along with the micro-actuator fine stage. The diagram on the right depicts a basic master-slave control architecture, where P_1 represents the fine actuator, and P_2 represents the coarse actuator. Not all HDDs are able to measure the fine actuator position x_1 directly, so in some cases the output of the fine stage controller is fed to the coarse stage instead.

However, there are many optical pointing system configurations,[14] and in some cases the combination of the two outputs in a dual-stage system is not a simple linear relationship, but a more complicated nonlinear kinematic operation. Some past papers describe the pointing and stabilization kinematics associated with various gimbal and steering mirror arrangements,[15–18] however these studies focus primarily on single-stage systems, and they do not describe how to augment such a system with a dual-stage mechanism.

What has yet to be published is a paper that extends the results mentioned above to describe a dual-stage control approach for an optical pointing system that involves nonlinear kinematic combination of the two plant outputs, and does so in such a way as to ensure the two stages don't end up "fighting" each other. This is precisely the intended scope of this paper.

2. SYSTEM OVERVIEW

This paper will focus on a reference optical pointing system consisting of a heliostat for coarse pointing, an FSM for fine pointing, and an optical track sensor for closed-loop optical feedback. Figure 2 illustrates the system layout and kinematic modeling nomenclature.

The heliostat will be modeled as two rotary inertias, each actuated by a brushless DC torque motor, with low-friction bearings to facilitate smooth rotation. The range of motion of the heliostat is large about the outer ("azimuth") axis b_1 ($\pm 180°$), and narrower in the inner ("elevation") axis b_2 ($\pm 15°$, mirror motion). The inputs to this mechanism are the torques provided by the motors, and the outputs of this mechanism are the rotation angles α and β about the two axes b_1 and b_2, respectively. The dynamic plant model for this system consists of a double integrator with a fixed gain for the inertia. Dynamic coupling between the two axes is neglected, as this can be minimized in a mechanically balanced design. The control bandwidth for each of the individual drive axes is assumed to be around 20 Hz, limited by the mechanism structural modes assumed to exist in the 50 - 100 Hz range.[19]

The FSM will be modeled as a two decoupled spring-mass-damper systems, each actuated by a voice-coil actuator, with a central pivot flexure to facilitate frictionless tip/tilt rotation. The inputs to this mechanism are the torques provided by the voice coil actuators, and the outputs are the tip/tilt angles τ_1 and τ_2 about the rotation axes a_1 and a_2, respectively. The FSM range of travel is limited to less than $\pm 1.5°$ (± 26 mrad) in each axis. Due to the restoring force of the pivot flexure, the system dynamics look like a lightly-damped second-order system. The control bandwidth for each of the FSM axes is assumed to be around 500 Hz.

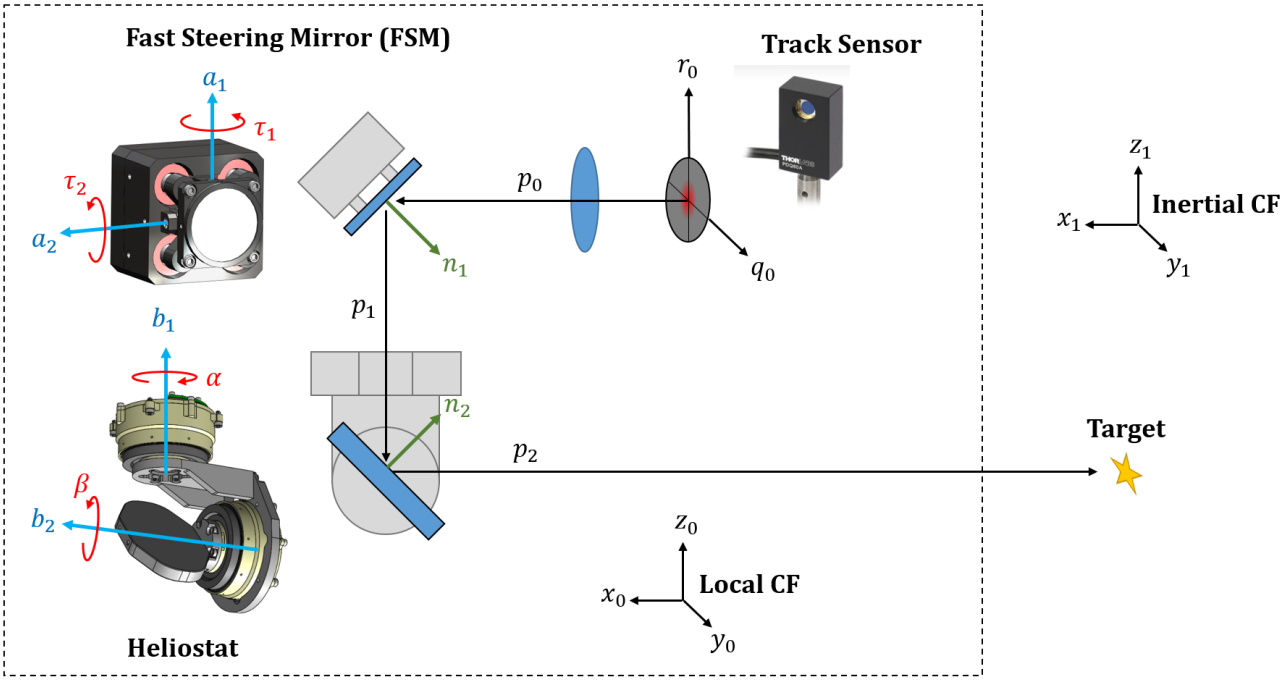

Figure 2. Dual-Stage Heliostat + FSM Pointing System

The track sensor is assumed to measure the relative line-of-sight (LOS) angle of the target in the track sensor coordinate frame. Practically, this track sensor could be a quad cell photodiode, a lateral effect cell, or a camera focal plane. For this system we assume a field of view of $\pm 0.115°$ (± 2 mrad), where the controller must keep the LOS error to less than this amount, otherwise the system will lose optical feedback and have to re-acquire.

The heliostat, FSM, and track sensor are all rigidly mounted to a common optical bench, which is itself rigidly attached to a host platform (e.g., a satellite or an aircraft) that can rotate in inertial space. The target is located at some large distance away, and it is assumed that the task of achieving initial spatial acquisition (i.e., pointing accurately enough in the first place for the track sensor to be able to "see" the target and provide pointing feedback[1]) is already solved.

Once the systems have completed spatial acquisition, the task of the dual-stage controller is to reject both low-frequency, large-amplitude disturbances (e.g., host platform rotations) in the DC to 10^{-1} Hz range, as well as higher-frequency, small-amplitude disturbances (e.g., vibration-induced opto-mechanical jitter) in the 10^1 to 10^3 Hz range. The smaller the residual pointing error, the stronger the communication signal received at each terminal, and the higher the data rate that can be transmitted through the link.

3. FORWARD KINEMATICS MODEL

The forward kinematics of the system can be modeled in three parts: calculating the relative pointing vector of the optical payload with respect to the local coordinate frame (CF), calculating the LOS vector to the target in the local CF, and calculating the track sensor measurement of the relative LOS to the target. The forward kinematics model will be used to motivate the control architecture development in Sec. 4, to develop inverse-kinematic models in Sec. 5, and to simulate the system pointing response in Sec. 6.

3.1 LOS Pointing in Local CF

The optical pointing system output-space line of sight (LOS) can be modeled as a 3-dimensional unit vector, starting with the nominal LOS of the track sensor p_0, which is reflected off each of the two mirrors to become p_2,

as in Eq. (1). Each reflection can be modeled as pre-multiplication by a reflection matrix, where the reflection matrix is defined as a function of the mirror normal, as in Eq. (2).[15,17,18,20]

The normal vectors for each of the two mirrors (i.e., n_1 and n_2) can also be modeled as 3-dimensional unit vectors, which are rotated versions of the nominal vectors n_{10} and n_{20}. The FSM mirror normal vector is rotated about axis a_2 by angle τ_2, and then about axis a_1 by angle τ_1, per Eq. (3). Similarly, the heliostat mirror normal is rotated about axis b_2 by angle β, and then about axis b_1 by angle α, per Eq. (4). The formula for generating a rotation matrix for a given axis-angle argument is provided in Eq. (5). Note that all of the vectors in this model are unit-length column vectors, defined with respect to the local CF which is fixed to the optical bench / platform structure.

$$p_2 = M_2(n_2)M_1(n_1)p_0 \tag{1}$$

$$M(n) = I - 2nn^T \tag{2}$$

$$n_1 = R(a_1, \tau_1)R(a_2, \tau_2)n_{10} \tag{3}$$

$$n_2 = R(b_1, \alpha)R(b_2, \beta)n_{20} \tag{4}$$

$$R(u,\theta) = \begin{bmatrix} \cos\theta + u_x^2(1-\cos\theta) & u_x u_y(1-\cos\theta) - u_z\sin\theta & u_x u_z(1-\cos\theta) + u_y\sin\theta \\ u_y u_x(1-\cos\theta) + u_z\sin\theta & \cos\theta + u_y^2(1-\cos\theta) & u_y u_z(1-\cos\theta) - u_x\sin\theta \\ u_z u_x(1-\cos\theta) - u_y\sin\theta & u_z u_y(1-\cos\theta) + u_x\sin\theta & \cos\theta + u_z^2(1-\cos\theta) \end{bmatrix} \tag{5}$$

3.2 Target Location in Local CF

When operating in optical closed-loop, the pointing system does not actually need to know the target location in output space. However, for the purpose of modeling the system response to large-angle disturbance, it helps to model the platform and target positions, as well as the platform orientation, in an inertially fixed coordinate frame. To this end,the platform and target positions in the inertial CF are denoted as s_P and s_T, respectively, and the rotation from inertial CF to local CF is represented as a direction cosine matrix R_{LI}. Thus, the unit vector describing the "ideal" LOS from platform to target is the normalized difference of s_T minus s_P, rotated into the local CF by R_{LI}, as in Eq. (6).

$$p_T = R_{LI}\frac{s_T - s_P}{||s_T - s_P||} \tag{6}$$

3.3 LOS Error in Track Sensor CF

The track sensor essentially measures the projection of the target LOS vector p_T onto the track sensor X/Y axes. The track sensor output-space X/Y axes can be calculated using Eq. (1), replacing p_0 with q_0 or r_0 to calculate q_2 or r_2, respectively. Since these are all unit vectors, the projection reduces to simply the dot product, as in Eq. (7).

$$y_0 = \begin{bmatrix} p_T \cdot q_2 \\ p_T \cdot r_2 \end{bmatrix} \tag{7}$$

One interesting characteristic of this model is that any individual set of joint angles $(\tau_1, \tau_2, \alpha, \beta)$ to produce a given LOS vector p_2 is not unique. Therefore, it is not possible to calculate the α and β angles required to

achieve a given p_2 without knowing (or specifying) values of τ_1 and τ_2. Another point is that the combination of joint angles to produce a given p_2 is nonlinear and coupled, which is the main difference between the dual-stage systems mentioned before which combine via simple linear sum. Figure 3 illustrates the three parts of the forward kinematic model in the context of the overall control block diagram.

4. CONTROL ARCHITECTURE

The proposed control architecture is similar to the "Master-Slave" model presented in HDD literature,[3,6,9] in that the FSM is focused entirely on minimizing the optical track sensor error (with no "knowledge" that the heliostat exists), and the heliostat is entirely focused on returning the FSM back to its null position.* However we must make some slight modifications to the results mentioned above in order to handle the fact that (a) the optical pointing system has a 4-DoF input, 2-DoF output, and 3-dimensional kinematics in-between, and (b) the input variables interact to affect the output in a nonlinear, coupled fashion. We address this with the following approach.

First, we assume that for any measured track sensor X/Y outputs, we have an accurate model of the FSM pointing kinematics, which can be used to calculate the FSM mirror angles required to steer the track sensor outputs to zero. This operation would account for factors such as focal length scaling of the track sensor, "clocking" rotations between FSM X/Y axes and track sensor X/Y axes, and angle-of-incidence-related gains that are difference between the two mirror axes. This operation will be denoted as $K_1^{-1}(y)$, where the track sensor output y includes additive disturbance, i.e., $y = y_0 + d$.

Next, we assume that for any set of FSM and heliostat positions x_1, x_2 (where x_1 is a shorthand notation for the FSM joint angles τ_1 and τ_2, and x_2 is a shorthand notation for the heliostat joint angles α and β) within the allowable operating range, we have a kinematic model $K_2^{-1}(x_1, x_2)$ of the system that we can use to calculate the desired heliostat state x_2^* which achieves virtually the same output LOS pointing vector p_2 with a FSM state $x_1 = (0, 0)$. In other words, we assume the model K_2^{-1} satisfies Eqs. (8) and (9), where $K(x_1, x_2) = p_2$ represents the actual physical pointing kinematics of the system (i.e., the physical "truth" that we sought to model in Sec. 3.)

$$x_2^* = K_2^{-1}(x_1, x_2) \tag{8}$$

$$K(0, x_2^*) \approx K(x_1, x_2) \tag{9}$$

Neither inverse-kinematic model needs to be "perfect." The FSM inverse-kinematic model K_1^{-1} only needs to be accurate enough that the low-frequency / DC response of the FSM is sufficient to reject any residual model error and bring the track sensor output to an acceptable steady-state error. To this end, we make sure to add an integral term to the FSM track loop controller. Similarly, the heliostat inverse-kinematic model K_2^{-1} only needs to be accurate enough that the low-frequency / DC response of the heliostat is sufficient to reject any residual model error and bring the FSM to its null position in steady state. To this end, we add a PI compensator to the FSM-to-heliostat offload path.

To summarize, this inverse-kinematic, dual-stage (IKDS) control architecture works essentially the same way as the Master-Slave architecture described in HDD literature,[6] except that there are two inverse-kinematic transformations: one in the FSM track loop feedback path, and another in the FSM-to-heliostat offload path. A block-diagram illustration of the IKDS approach is provided in Fig. 3.

*Here we assume that relative position feedback of the fine actuator is readily available, which is not always the case in HDD systems.

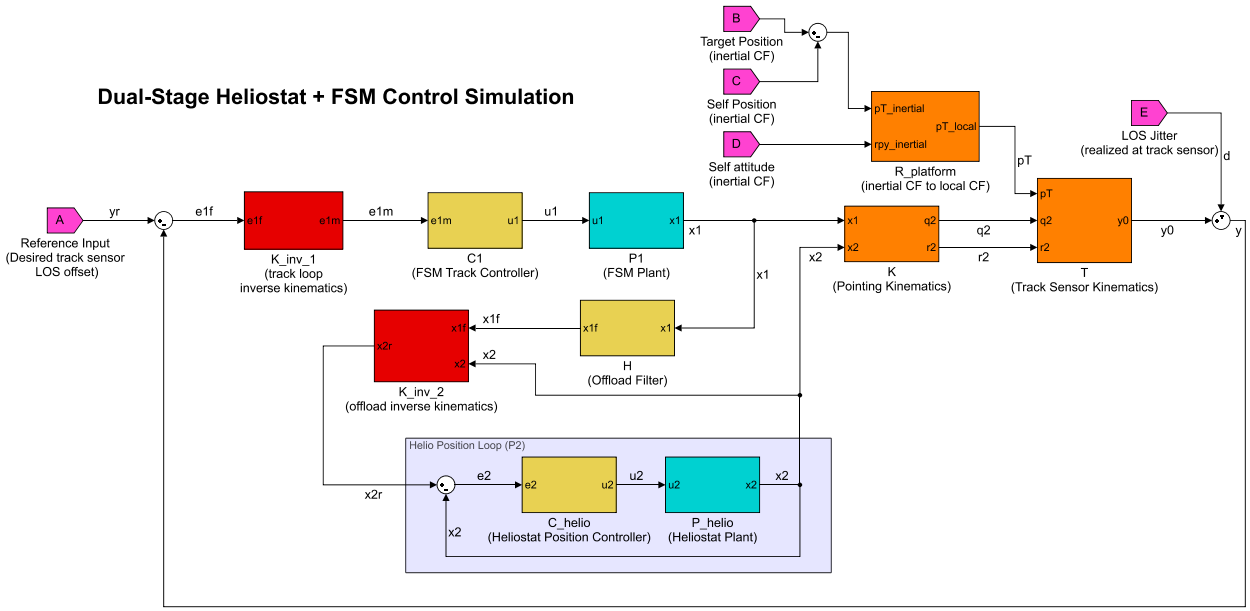

Figure 3. IKDS Control Architecture Block Diagram. In this diagram, the cyan blocks labeled P1 and P_helio represent the FSM and heliostat plant models. The yellow C1, and C_helio blocks represent the FSM and heliostat controllers. The yellow block labeled H represents the FSM-to-heliostat offload filter. The orange blocks labeled K, R_platform, and T represent the forward kinematic operations, and the red blocks labeled K_inv_1 and K_inv_2 represent inverse kinematic operations. The pink blocks represent inputs to the model (e.g., target motion, host platform rotation or LOS jitter).

5. INVERSE KINEMATICS

As mentioned in Sec. 4, there are two inverse-kinematic transforms that must be modeled: one for the FSM, and another for the heliostat. Although this paper presents one modeling approach for each of these operations, alternative approaches could certainly be conceived which may work just as well. Indeed, if one is willing to make assumptions (or enforce requirements) about how perfectly the hardware is built, then simpler kinematic models may be used. However we chose to use the approach below as it allows for more accurate modeling of a system with inevitable build imperfections.[†]

5.1 FSM/Track Sensor Inverse Kinematics

The FSM track controller operates on the two axes of rotation independently, and it must account for the angle-of-incidence gains which are generally different in the two axes, relative orientation of the FSM axes versus the track sensor axes (which may involve polarity flips and/or "clocking" rotations), and any residual scale factor associated with the track sensor output (e.g., related to effective system focal length). The ensemble of these effects may be modeled by representing the track sensor X/Y measurements as a vector, and pre-multiplying that vector with a matrix that handles scaling and rotations, as in Eq. (10). This transformation matrix can formulated analytically, or experimentally by moving the FSM in each axis individually and observing the LOS motion at the track sensor. The outputs of $K_1^{-1}(y)$ are then provided to the FSM track controller as error signals.

$$K_1^{-1}(y) = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \tag{10}$$

[†]For example, it is commonly assumed that the rotation axes of a two-axis gimbal are perfectly orthogonal, and perfectly aligned with a local CF. The inverse kinematic model presented herein does not rely on this assumption.

5.2 Heliostat Inverse Kinematics

The pointing model referred to in Sec. 4 as K_2^{-1} will be split into two parts. First, the measured FSM and heliostat positions x_1 and x_2 will be used to calculate an estimate of the current output-space LOS vector p^*. This model will be formulated exactly as described in Sec. 3.1. The second step is to calculate the heliostat azimuth and elevation angles $\hat{\alpha}$ and $\hat{\beta}$ required to achieve this LOS vector p^* with the FSM at $x_1 = (0, 0)$. This process is performed as follows.

First, calculate the nominal output LOS vector p_{20}, using Eq. (11),

$$p_{20} = M(n_{20})M(n_{10})p_0. \tag{11}$$

Next, project both p_{20} and p^* onto the plane defined by the heliostat inner rotation axis b_1 (where Eq. (12) provides the formula for projecting a vector v onto a plane defined by another vector w.)

$$v_{\perp w} = v - v_{\parallel w} = v - \frac{v \cdot w}{\|w\|^2} w \tag{12}$$

Now calculate the angle between the projections of p_{20} and p^* in the plane of the b_1 rotation axis. This angle is $\hat{\alpha}$ (where Eq. (13) provides the formula for calculating the counter-clockwise angle from vector u to vector v, in the plane of a third vector w, which is used for a sign convention).

$$\theta = \Big| \operatorname{atan2}\left(||u \times v||, u \cdot v\right) \Big| \operatorname{sgn}\left(w \cdot (u \times v)\right); \tag{13}$$

Next, rotate p^* about b_1 by $-\hat{\alpha}$, as in Eq. (14). This rotated version of p^* should now be coplanar with p_{20}.

$$p^*_{rotated} = R(b_1, -\hat{\alpha})p^*; \tag{14}$$

Finally, re-use Eq. (13) to calculate the angle between the rotated version of p^* from the previous step, and p_{20}. This angle is equal to $2\hat{\beta}$. (Note: this factor of 2 comes from angle doubling due to in-plane reflection at heliostat mirror). The resulting heliostat joint angles $\hat{\alpha}$ and $\hat{\beta}$ are then provided to the heliostat controller as position setpoints.

6. SIMULATION STUDY

A simulation was developed using Simulink[‡] to prove out this control architecture concept. A screenshot of the Simulink block diagram is provided in Fig 3.

6.1 Plant and Controller Modeling

The FSM plant was modeled as a lightly damped second-order spring-mass-damper system (as in Refs. 21–25), with a transfer function shown in Eq. (15), with $\omega_n = 251$ and $\zeta = 0.02$. The FSM controller was a PI plus lead compensator, with a transfer function shown in Eq. (16), with $K_{tot} = 59.8$, $K_p = 1$, $K_i = 164.6$, $\omega_z = 0.000825$, and $\omega_p = .000123$. The resulting 0 dB crossover of the FSM open-loop gain was 500 Hz, with a phase margin of 45 degrees (Fig. 4).

$$P_1(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{15}$$

$$C_{PI+LL}(s) = K_{tot}\left(\frac{K_p s + Ki}{s}\right)\left(\frac{\omega_z s + 1}{\omega_p s + 1}\right) \tag{16}$$

[‡]MATLAB and Simulink are technical computing programs developed by The MathWorks, Inc., Natick, MA, USA

The heliostat plant was modeled as a double integrator (as in Ref. 13), with a transfer function shown in Eq. (17), with $J = 1$. The heliostat controller was modeled as a PI plus lead compensator, with a transfer function as shown in Eq. (16), with $K_{tot} = 964.5$, $K_p = 1$, $K_i = 6.59$, $\omega_z = 0.1301$, and $\omega_p = .0005$. The resulting 0 dB crossover of the heliostat open-loop gain was 20 Hz, with a phase margin of 80°.

$$P_{helio}(s) = \frac{J}{s^2} \tag{17}$$

In designing the offload filter, we considered the relative responses of the FSM and heliostat plants to a given error signal. In a system where these two plants combine linearly, the function HP_2 (where H is the offload filter transfer function, and P_2 is the closed-loop heliostat transfer function) represents the ratio of heliostat response to FSM response. Where the magnitude of HP_2 is equal to 0 dB, the responses of the heliostat and FSM will be equal. Therefore, at the frequency where this 0 dB crossover occurs, if the phase of HP_2 reaches −180°, the two plants could end up perfectly out-of-phase with each other, essentially canceling out each other's control authority.

In order to avoid this, we made sure to select H so that HP_2 had more than 90° of phase margin where the magnitude crosses over 0 dB.[§] We ended up using a PI filter, as in Eq. (18), with $K_p = 1$ and $K_i = 62.8$, which resulted in HP_2 crossing over 0 dB at 16.5 Hz with 105° of phase margin. The integrator here also boosted the DC gain of the offload path, helping ensure that the heliostat was able to return the FSM back to zero at steady state.

$$H(s) = \frac{K_p s + Ki}{s} \tag{18}$$

Bode plots for the FSM open-loop system (P_1C_1), the heliostat closed loop system (P_2), and the combination of heliostat closed loop response and offload filter together (HP_2) are shown in Fig. 4.

6.2 Simulation Scenarios

The initial condition for the simulation involved both mechanisms at their nominal / null positions, pointing perfectly at the target. This was intended to simulate the disturbance response of an optical pointing system that had already achieved successful spatial acquisition. From here, two disturbance scenarios were simulated: one of a step response (representative of initial acquisition once the target enters the field of view of the track sensor), and one of a composite of a low-frequency, large-amplitude sine wave plus a higher-frequency, smaller-amplitude sine wave (representative of ongoing disturbances from the platform rigid body motion and vibration-induced jitter, while closing a link in fine track mode). In both cases, the same magnitude of disturbance was applied to both output axes simultaneously.

The step disturbance of height 0.115° (2 mrad) was injected into the track sensor measurement before being fed back to the FSM track controller. The system response to this step disturbance is plotted in Fig. 5. In this scenario, the FSM immediately jumps up to respond to the step input, bringing the LOS error measured at the track sensor to near zero in a matter of a couple milliseconds. The heliostat reacts slightly slower, eventually responding to bring the FSM back to its home position. By a time of about 100 millseconds seconds after the step input, the dual-stage system has essentially reached steady-state, with the heliostat rejecting the step disturbance and the FSM returned back to zero. Similar responses were observed when applying the step disturbance in the other axis, or when starting at different heliostat positions.

The composite sine disturbance was injected in two places; the low-frequency sine wave (frequency = 0.5 Hz, amplitude = 20 mrad) was applied as a large-angle platform rotation (via the kinematics described in Sec. 3.2), and the high-frequency sine (frequency = 100 Hz, amplitude = 0.001 degrees) was injected to the track sensor output (as with the step disturbance above). The time-domain response of this system to this composite

[§]We chose a large phase margin objective here to account for the fact that the relative response of the FSM and heliostat will not be exactly HP_2, since there are actually inverse kinematics in both FSM and heliostat paths.

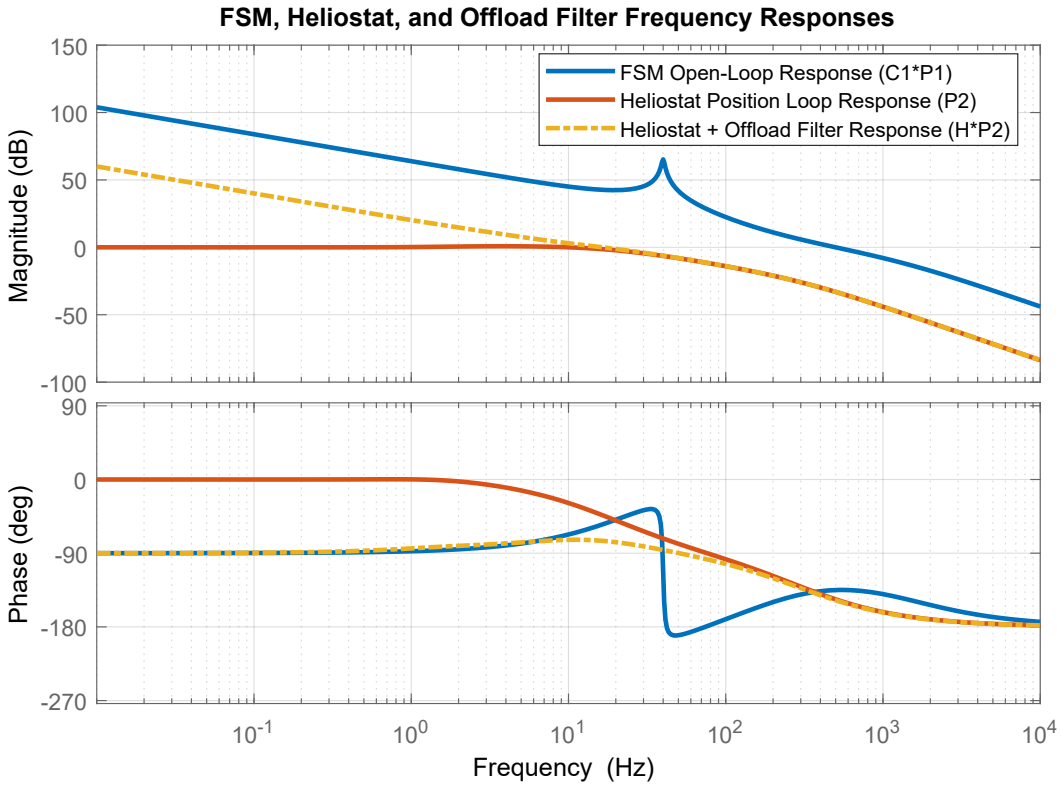

Figure 4. FSM, Heliostat, and Offload Filter Open-Loop Bode Plots

disturbance input is plotted in Figure 6. In this scenario, it is clear that the heliostat is responding primarily to the low-frequency platform motion, whereas the FSM is responding primarily to the high-frequency jitter disturbance. The net result is a very small residual pointing error over a wide range of travel.

7. CONCLUSION

In summary, results from past literature on hard disk drives and optical pointing systems were extended to describe an inverse-kinematic approach for controlling dual-stage optical pointing systems with nonlinear combination of plant outputs. A reference system consisting of a heliostat and FSM was simulated in a proof-of-concept study in Simulink. This approach is expected to generally apply to other configurations as well, such as systems using coelostats or other non-gimbaled coarse pointing approaches.

ACKNOWLEDGMENTS

The author would like to thank Chiyun Xia, Kevin Birnbaum, Brian Parris, Andrew Grier, and Hamid Hemmati for their thoughtful review and advice.

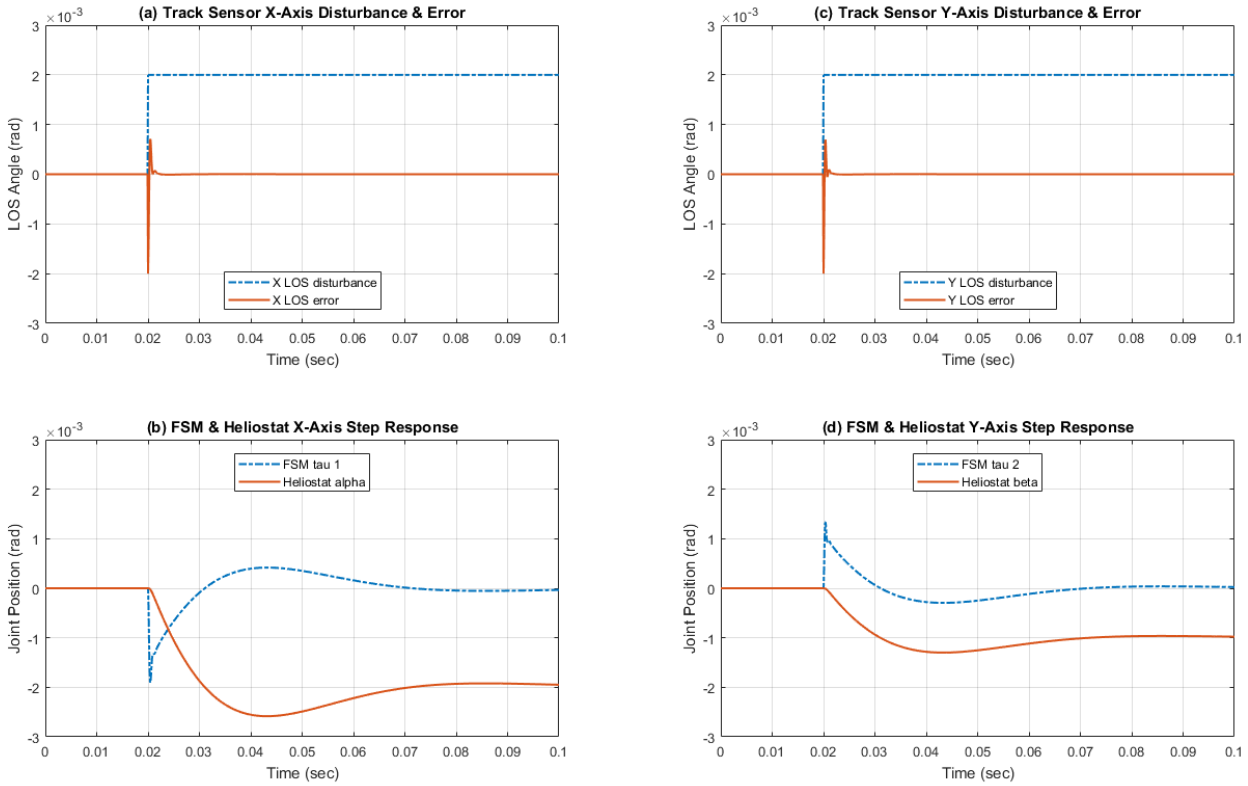

Figure 5. Heliostat + FSM System Simulated Step Response

REFERENCES

[1] Hemmati, H., ed., [*Near-Earth Laser Communications*], CRC Press (2009).

[2] Mori, K., Munemoto, T., Otsuki, H., Yamaguchi, Y., and Akagi, K., "A dual-stage magnetic disk drive actuator using a piezoelectric device for a high track density," *IEEE Transactions on Magnetics* **27** (1991).

[3] Al Mamun, A., Mareels, I., Lee, T. H., and Tay, A., "Dual Stage Actuator Control in Hard Disk Drive - A Review," *Proc. of IECON* **3** (2003).

[4] Rotunno, M., De Callafon, R. A., and Talke, F. E., "Comparison and Design of Servo Controllers for Dual-Stage Actuators in Hard Disk Drives," *IEEE Transactions on Magnetics* **39** (2003).

[5] Guo, W., Lee, T. H., and Huangt, T., "Linear Quadratic Optimal Dual-stage Servo Control," *Proc. of IEEE Industrial Electronics Society* (1998).

[6] Guo, L., Martin, D., and Brunnett, D., "Dual-stage actuator servo control for high density disk drives," *Proc. of IEEE/ASME International Conference on Advanced Intelligent Mechatronics* (1999).

[7] Graham, M., Oosterbosch, R. J. M., and De Callafon, R. A., "Fixed order PQ-control design method for dual-stage instrumented suspension," *Proc. of IFAC* **16** (2005).

[8] Li, Y. and Horowitz, R., "Analysis and Self-tuning Control of Dual Stage Servos with MEMS Micro-actuators," *Proc. of ACC* (2000).

[9] Schroeck, S. J. and Messner, W. C., "On controller design for linear time variant dual input single output systems," *Proc. of ACC* (1999).

[10] Boettcher, U., Callafon, R. a. D., and Talke, F. E., "Modeling and Control of a Dual Stage Actuator Hard Disk Drive," *Journal of Advanced Mechanical Design, Systems, and Manufacturing* **4** (2010).

[11] Jansky, C., "Samsung HD753LJ hard disk drive (750 GB storage capacity, date of manufacture: March 2008). Actuator with read/write-head, platters.," (2008).

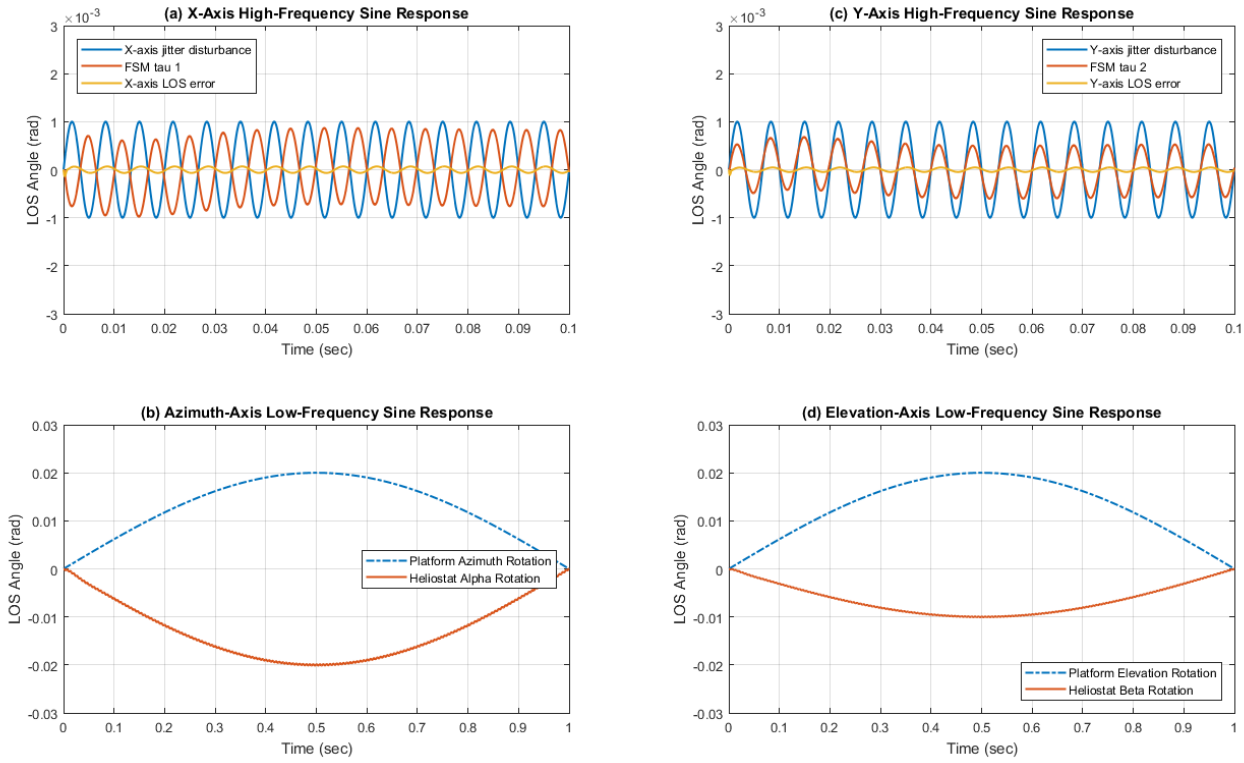

Figure 6. Heliostat + FSM System Composite Sine Response

[12] Jenkins, S. T. and Hilkert, J. M., "Line Of Sight Stabilization Using Image Motion Compensation," *Proc. of SPIE* **1111** (1989).

[13] Hilkert, J. M., "Inertially Stabilized Platform Technology: Concepts and Principles," *IEEE Control Systems Magazine* (February), 26–46 (2008).

[14] Casey, W. and Phinney, D., "Representative pointed optics and associated gimbal characteristics," in [*Proc. of SPIE*], **887** (1988).

[15] Royalty, J., "Development of kinematics for gimballed mirror systems," *Proc. of SPIE* **1304** (1990).

[16] DeBruin, J., "Derivation of line-of-sight stabilization equations for gimbaled-mirror optical systems," *Control Systems Technology Center* **1543** (1992).

[17] Hilkert, J. M., "A comparison of inertial line-of-sight stabilization techniques using mirrors," *Proc. of SPIE* **5430** (2004).

[18] Satyarthi, S., "Optical line-of-sight steering using gimbaled mirrors," *Proc. of SPIE* **9076** (2014).

[19] Hilkert, J. M. and Amil, D. L., "Structural effects and techniques in precision pointing and tracking systems: a tutorial overview," *Proc. of SPIE* **7696** (2010).

[20] Debruin, J. C. and Johnson, D. B., "Line-of-sight reference mirror frames: a unified approach to plane-mirror optical kinematics," *Proc. of SPIE* **1697** (1992).

[21] Cochran, R. W. and Vasser, R. H., "Fast Steering Mirrors in Optical Control Systems," *Proc. of SPIE* **1303** (1990).

[22] Skormin, V., Busch, T., and Givens, M., "Model reference control of a fast steering mirror of a pointing, acquisition and tracking system for laser communications," *Proc. of IEEE NAECON* **2** (1995).

[23] Xie, M. and Fu, C., "Design and experiment of a LQ controller used in high-bandwidth fast steering mirror system," *Proc. of SPIE* **4025** (2000).

[24] Portillo, A. A., Ortiz, G. G., and Racho, C., "Fine pointing control for optical communications," *Proc. of IEEE Aerospace Conference* **3** (2001).

[25] Zhou, Q., Ben-Tzvi, P., Fan, D., and Goldenberg, A. A., "Design of fast steering mirror systems for precision laser beams steering," *Proc. of IEEE International Workshop on Robotic and Sensors Environments* (2008).