AN045

Making a 3-Actuator Fast Steering Mirror Control Matrix

|	
Author:	**Dr. Justin Mansell and Jacob Thordahl**
Revision:	**7/22/15**

1   Introduction

One common fast steering mirror (FSM) design involves placing three actuators in a triangular pattern and pushing each actuator normal to the mirror surface against some reaction mass. Figure 1 shows an example fast steering mirror in this architecture. The three white columns represent the actuators. The blue top piece is the mirror. The gray bottom piece is the reaction mass.
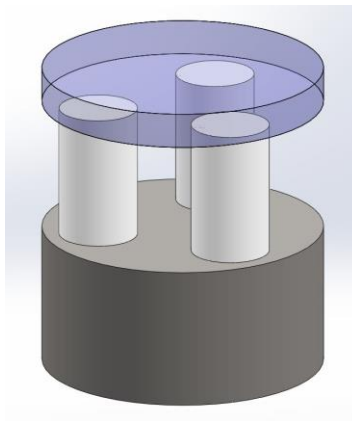
Figure 1: Example 3-Actuator FSM Architecture

Figure 2 shows a top view of the mirror surface, the location of three actuators, and labels of the actuators.
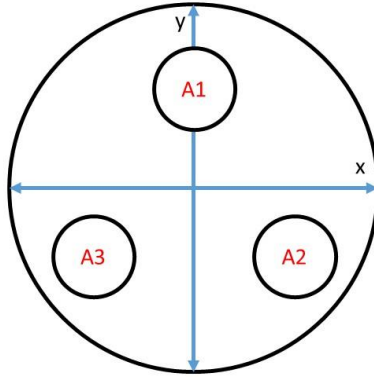
Figure 2: Actuator Location Definition on the Mirror Surface (Top View)

The relationship between the mirror tilt and the actuator displacement can be represented by the linear algebra equation,

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} 0 & -\cos(30°) & +\cos(30°) \\ 1 & -\cos(60°) & -\cos(60°) \end{bmatrix} \cdot \begin{bmatrix} A1 \\ A2 \\ A3 \end{bmatrix}$$

where X and Y are the tilts in each of these directions and A1, A2, and A3 are the actuator displacements. This equation does not provide any additional constraint on the overall piston of the mirror. This matrix can be inverted to relate the desired tilt in the two axes to the actuator commands. If there is no noise in the system (actuator gain variation, sensor noise, etc.), the matrix will invert to impose no additional net piston to the actuators. Sometimes system non-idealities can cause the actuator commands to favor two actuators above the other one. For example, actuation of A2 and A3 together will induce Y tilt and differentially will induce X tilt. This piston term can cause a walk of the beam on the FSM and, in some cases, can cause a reduction in the tilt range in one axis.

1.1   Solution: Added Constraint

To eliminate the piston term from the system, we can add a constraint to the matrix that the sum of the actuator commands must equal zero. The new equation representing this is

$$\begin{bmatrix} X \\ Y \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & -\cos(30°) & +\cos(30°) \\ 1 & -\cos(60°) & -\cos(60°) \\ 1/3 & 1/3 & 1/3 \end{bmatrix} \cdot \begin{bmatrix} A1 \\ A2 \\ A3 \end{bmatrix}.$$

The final row forces the sum of the actuator commands to equal zero. Inversion of this matrix now produces a 3x3 matrix that is fully constrained given by

$$\begin{bmatrix} 0 & 2/3 & 1 \\ -\tan(30°) & -1/3 & 1 \\ +\tan(30°) & -1/3 & 1 \end{bmatrix}.$$

Since the input vector always has 0 as the final element, the final column of the inverted matrix can be eliminated to reduce the computational load, producing the matrix,

$$\begin{bmatrix} 0 & 2/3 \\ -\tan(30°) & -1/3 \\ +\tan(30°) & -1/3 \end{bmatrix}.$$

The Matlab code in the appendix of this application note demonstrates this approach.

2   Conclusions

In this application note we have shown that we can add a row to the matrix relating tilt and actuator command prior to inversion to force the solution to have no net piston term for a 3-actuator fast steering mirror.

3   Appendix: Example Analysis Code

```matlab
% Example conditioning code for a 3-actuator FSM
clear all; clc; close all;
SF = pi/180;
M = [0 -cos(30*SF) +cos(30*SF);
     1 -cos(60*SF) -cos(60*SF);
     1/3 1/3 1/3];

%% show matrix, inverse, and condition
M
condition = cond(M)
Minv = inv(M)

%% plot result
nf;
subplot(1,2,1); imagesc(M); title('poke');
subplot(1,2,2); imagesc(Minv); title('ctrl');

%% what is the sum per dimension
sum(M')

%% what is an example command
offset = [1; 0.5; 0]

dc = Minv * offset
piston = sum(dc)

%% can I eliminate the last column
Minv2 = Minv(:,1:2)
dc2 = Minv2 * offset(1:2,1)
piston = sum(dc2)
```