Real-Time Control of a Segmented Telescope Test-Bed[1]

by

A. Abdullah and P. Ioannou

Dept. of Electrical Engineering-Systems

University of Southern California, Los Angeles, CA 90089-2563

aliabdul@usc.edu

ioannou@usc.edu

http://www-rcf.usc.edu/~ioannou

[1] This work was supported by *NASA* under Grant *NAGW-4103*

Abstract

The control of large segmented telescopes is a challenging one due to the complexity and high order of the system. The high order dynamics lead to high order controllers that require more memory and faster computations for implementation. While this may not pose a serious problem for a small number of segments, as the number of segments increases the computational requirements are becoming enormous.

In this article, we use the test-bed developed at California State University, Los Angeles, that simulates in real-time a large segmented telescope to test the performance and computational requirements of several control designs. Three decentralized control designs were selected for implementation. These include a decentralized state feedback proportional plus integral (*DSFPI*) controller, a decentralized output feedback proportional plus integral (*DOPFI*) controller, and a decentralized direct adaptive output feedback (*DDAOF*) controller.

The *DSFPI* controller requires more memory space and computational power than the *DOFPI* and *DDAOF* controllers. The *DOFPI* requires less memory space and computational operations but it fails to meet the performance requirements. The *DDAOF* requires an acceptable amount of memory space and computational operations and has better performance than the other two controllers.

1. Introduction

The main objective of designing a next generation space telescope (*NGST*) is to gather information from the far regions of space. The information will be used for understanding the: Universe structure, birth and formation of stars, and origins and evolution of planetary systems and galaxies. The key element to achieve these goals is to build a space telescope with a large reflector mirror. For instance, increasing the diameter aperture of the reflector mirror to 8 m from the 2.4 m in the Hubble Space Telescope (*HST*) the space telescope will be capable of collecting information from regions ten times farther away than the regions covered by the *HST*. Many difficulties are associated with the size of such reflectors, for example: It is difficult and expensive to manufacture a single large reflector, and a complex technology is required to launch such a space telescope. A possible way to overcome these problems is to build a space telescope with a number of small mirrors. These mirrors can be deployed inside the space launch vehicle and then adjusted together in the orbit to form a desired shape of a single larger reflector. On earth a very large reflector mirror can be developed by using smaller mirrors that are easier to manufacture and less costly. Maintaining the desired shape so that the segmented reflector behaves as a single reflector in the presence of disturbances is a challenging control problem due to the complexity of the system, the large number of sensors, and actuators and the computational requirements for implementation of the controllers.

In this article we consider the design, simulation, and implementation of a number of simple control techniques for controlling large segmented telescopes. A test-bed (Fig. 1) located at the Structures Pointing and Control Engineering (*SPACE*) Laboratory of California State University, Los Angeles, is used for testing and implementation. The test-bed was constructed by a team of faculty and students from: California State University, Los Angeles, California State University, Long Beach, University of Southern California, Los Angeles, and University of California, Berkeley The National Aeronautics and Space Administration (*NASA*) funded the development of the test-bed and control experiments.

Figure 1. *Segmented telescope test-bed at SPACE Lab*

In our approach we reduce the complexity of the controller by taking a decentralized approach [1] where each plate corresponding to a single mirror is considered as one subsystem. Then controllers with low computational requirements are developed for all subsystems. These include: decentralized state feedback proportional plus integral (*DSFPI*) control, decentralized output feedback proportional plus integral (*DOFPI*) control, and decentralized direct adaptive output feedback (*DDAOF*) control. We demonstrated experimentally that the *DDAOF* controller meets the performance requirements with an acceptable amount of memory space and computational operations. The *DDAOF* is based on output feedback and on a strict positive real condition for designing the adaptation law.

This article is organized as follows. First we describe the segmented telescope test-bed and experimental set-up. The mathematical model of the test-bed and the performance requirements are then presented. Next we design and analyze three different decentralized controllers. Finally the experimental results are presented with our conclusions.

2. Segmented Telescope Test-Bed

The segmented telescope test-bed is used to simulate a more complex next generation segmented telescope (*NGST*) system. The structure of the test-bed consists of three main parts: a primary mirror, a secondary mirror, and a support truss. The test-bed is positioned on a solid table to support the structure's weight and it is isolated from the ground with dampers in order to damp any external vibrations propagated from the ground. The primary mirror consists of 6-active panels surrounding a fixed center panel and each panel has a hexagonal shape with diameter of 101 cm. All active panels are attached to 18-actuators (three per panel) to support the weight of the panels and also to generate forces to move the panels. Furthermore, 18-edge sensors are used to measure the relative displacements between the panels. Fig. 2 shows the structure of the segmented telescope test-bed and the location of actuators and edge sensors on the bottom of the primary mirror.

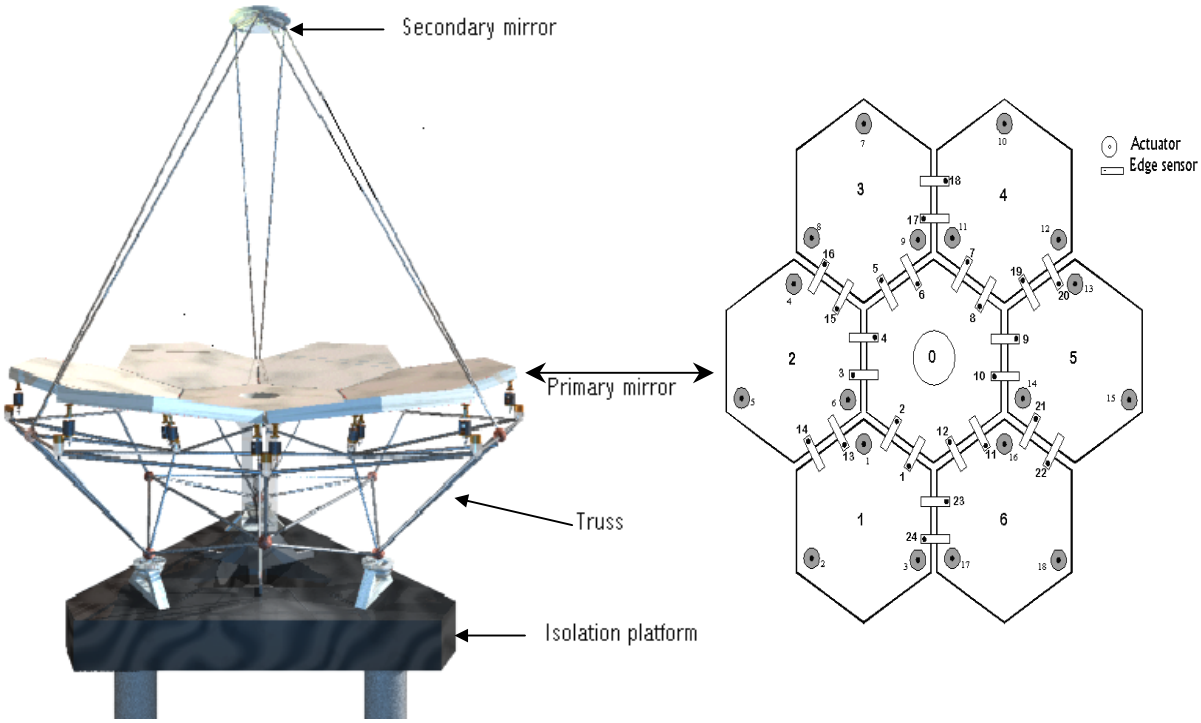

Figure 2. *The test-bed structure with location of actuators and edge sensors on the primary mirror*

The secondary mirror, which is a 6-sided pyramidal mirror, is used to reflect light from the primary mirror to the focal plane in the center panel. The secondary mirror is actively controlled by 3-actuators, which are attached to the secondary truss at 3-nodes. It is also equipped with 3-position sensors to provide the location of the secondary mirror with respect to its housing. The truss structure is designed to support the primary and the secondary mirrors. It is attached to the primary mirror at 18-nodes and to the secondary mirror at 3-nodes. The truss is made of stainless steel to provide the highest strength with lowest mass.

The test-bed components and data flow through the system are shown in Fig. 3.

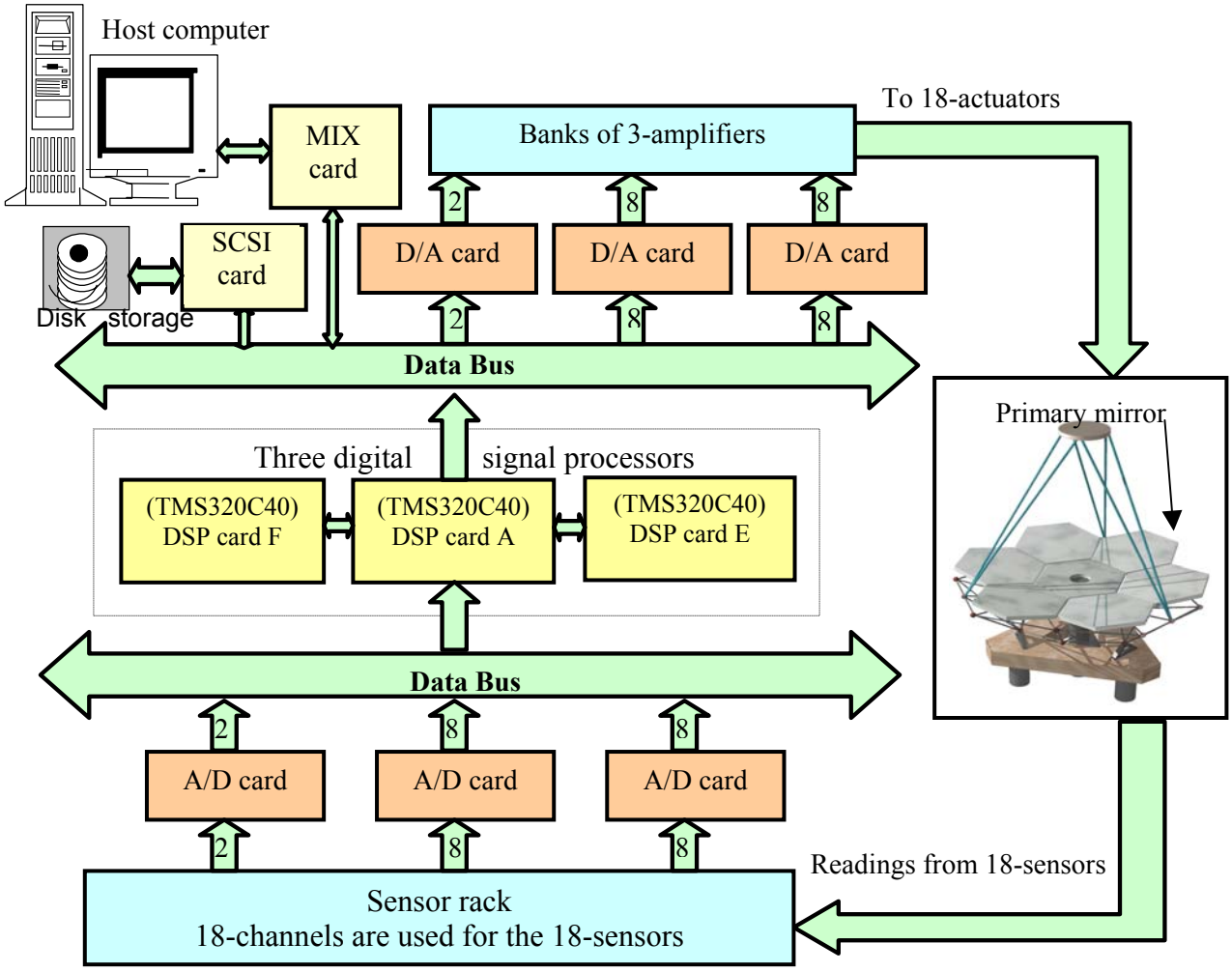

Figure 3. *Test-bed components and data flow through the system*

The readings from the 18-edge sensors that measure the relative displacements of the panels from their nominal positions corresponding to the desired primary mirror shape are sampled using 3 *A/D* cards. An 18-channel sensor display is used to display in real-time the sensor readings for monitoring purposes. The digital signal processor, *DSP* card-A, receives the sampled measurements of the 18-edge sensors and sends the 12-edge sensor readings 6 of which correspond to panels no. 1, 2 and the other 6 to panels no. 3, 4 to the *DSPs* card-E and card-F respectively. The control algorithms implemented using the 3-*DSPs* are used to generate the control commands that drive the 18-actuators. The 18-control commands are passed through the 3 *D/A* Cards to the 3-banks of amplifiers and then to the corresponding actuators. For data collection, an external memory disk (*SCSI*) is used to save the on line 18-edge sensor readings and 18-control commands. A host computer supported by *MATLAB* software is used to plot the results. The characteristics of the sensors, actuators and other important components of the experimental setup are described below

Edge sensors: The 18-inductive edge sensors are used to indicate any deviation in the primary mirror shape from its desired shape. 12-edge sensors out of 18 are used to measure the relative displacements between the fixed and active panels, and the other 6-edge sensors are used to measure the relative displacements between the active panels. Each edge sensor has a low noise level, high resolution of 0.1 μm, wide range of measurements up to 6 *mm*, bandwidth of 50 *kHz*, typical offset of 0.89 *mm*, and output range of +/- 15 volts.

Actuators: These are high performance linear electromagnetic force actuators used to actively control each active panel and support its weight. For each active panel, three actuators are used to change its position and the overall shape of the primary mirror. By applying the proper forces to the active panels, the shape of the primary mirror can be maintained to the desired one in case of any deviation. These actuators have a low noise level, a bandwidth of 100 *Hz*, and can generate forces up to 53.5 Newton.

Amplifiers: The linear amplifiers are used to amplify the signal received from the *D/A* converter before sent to the actuators. These amplifiers have high bandwidth and can be operated in velocity or current mode. In the test-bed, the amplifiers are operated in current mode. The input range of the amplifier is +/- 15 volts and its gain can be calibrated to three different gains.

A/D and D/A converters: For the purpose of control application, 3-cards of dual *A/D* and *D/A* converters are used to provide 18-channels of *A/D* and *D/A* conversion. The *A/D* converter has the appropriate analog input range consistent with the range of the output voltage of the sensor; it is used to convert the output voltage of the sensor into a proportional binary output. The *D/A* converter is used to convert the controller command into an analog signal for the actuator. These converters have 16-bit resolution with a sampling frequency range from 320 *Hz* to 250 *kHz*.

Digital signal processors: The digital signal processor (*DSP*) is the main computational unit. It uses the sampled signals from the *A/D* converters and performs arithmetic processing on these signals. The resulting signals are converted back to a continuous signal using the *D/A* converters. Three processors *TMS320C40's*, which are labeled as *DSP-A*, *DSP-E*, and *DSP-F*, are used to implement the control algorithms. Each processor has 2 megabits of *RAM*. Because all processors are linked together by common ports, all three processors can use the 6-megabits of *RAM* simultaneously. Furthermore, a data rate of 20 megabites/sec can be transferred between the processors. The three processors are connected to a host computer by a *MIX*-card and the connection is supported by a *Pentek's SwiftNet* software. For data collection, an external memory disk (*SCSI*) is used to save the data from the test-bed. In the test-bed, the control algorithms are written in *C++* language and then converted to assembly language using a *C*-compiler. The assembly codes are required in order to program the *DSPs*.

3. Model of the Primary Mirror System

A mathematical model of the primary mirror system was obtained using frequency domain techniques [2]. The relationship between the edge sensor outputs and actuator inputs is given by

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) & G_{13}(s) & G_{14}(s) & G_{15}(s) & G_{16}(s) \\ G_{21}(s) & G_{22}(s) & G_{23}(s) & G_{24}(s) & G_{25}(s) & G_{26}(s) \\ G_{31}(s) & G_{32}(s) & G_{33}(s) & G_{34}(s) & G_{35}(s) & G_{36}(s) \\ G_{41}(s) & G_{42}(s) & G_{43}(s) & G_{44}(s) & G_{45}(s) & G_{46}(s) \\ G_{51}(s) & G_{52}(s) & G_{53}(s) & G_{54}(s) & G_{55}(s) & G_{56}(s) \\ G_{61}(s) & G_{62}(s) & G_{63}(s) & G_{64}(s) & G_{65}(s) & G_{66}(s) \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} \tag{1}$$

where $G_{ij}(s) \in \mathbb{C}^{3\times 3}$ is the transfer function matrix from the *j-th* input vector to the *i-th* output vector. The input and output vectors consist of the following actuator inputs and edge sensor outputs where the subscript number i in the input u_i and output y_i corresponds to the active panel number

$$u_1 = \begin{bmatrix} A\,c_1 \\ A\,c_2 \\ A\,c_3 \end{bmatrix}, u_2 = \begin{bmatrix} A\,c_4 \\ A\,c_5 \\ A\,c_6 \end{bmatrix}, u_3 = \begin{bmatrix} A\,c_7 \\ A\,c_8 \\ A\,c_9 \end{bmatrix}, u_4 = \begin{bmatrix} A\,c_{10} \\ A\,c_{11} \\ A\,c_{12} \end{bmatrix}, u_5 = \begin{bmatrix} A\,c_{13} \\ A\,c_{14} \\ A\,c_{15} \end{bmatrix}, u_6 = \begin{bmatrix} A\,c_{16} \\ A\,c_{17} \\ A\,c_{18} \end{bmatrix},$$

$$y_1 = \begin{bmatrix} Es_1 \\ Es_2 \\ Es_{14} \end{bmatrix}, y_2 = \begin{bmatrix} Es_3 \\ Es_4 \\ Es_{16} \end{bmatrix}, y_3 = \begin{bmatrix} Es_5 \\ Es_6 \\ Es_{18} \end{bmatrix}, y_4 = \begin{bmatrix} Es_7 \\ Es_8 \\ Es_{20} \end{bmatrix}, y_5 = \begin{bmatrix} Es_9 \\ Es_{10} \\ Es_{22} \end{bmatrix}, y_6 = \begin{bmatrix} Es_{11} \\ Es_{12} \\ Es_{24} \end{bmatrix},$$

Ac and *Es* stand for the actuator input and edge sensor output, respectively. The state space representation of the primary mirror system obtained from (1) is given by

$$\begin{aligned} \dot{x} &= A\,x + \sum_{i=1}^{6} B_i u_i \\ y_i &= C_i x \qquad\qquad i = 1,2,....,6 \end{aligned} \tag{2}$$

where $x \in \Re^{320}$ is the state vector, and A, B_i and C_i are known matrices of appropriate dimensions.

Performance requirements: The quality of the information that will be collected by a segmented space telescope depends mainly on the alignment of its secondary and primary mirrors; misalignment between the mirrors can cause information loss or quality degradation. The shape of these mirrors can easily deviate from its desired shape in the presence of disturbances. Therefore, it is important for the primary mirror to behave as a desired single surface in the presence of disturbances. In the test-bed, the deviation of the primary mirror shape from the desired shape is characterized by the root mean square of the edge sensor outputs of each of the 6-active panels. The shape error Se_i values are used to indicate how far the shape of the primary mirror is from its desired shape and are defined as:

$$Se_i = \sqrt{\frac{\tilde{y}_i^T \tilde{y}_i}{3}} \qquad\qquad i = 1,2,.....,6$$

where $\tilde{y}_i$ is the average of the *i-th* local output vector y_i measurements over a window of 1000 samples. The control design objectives are as follows

- The Se_i value for each active panel is less or equal to 1 μm at steady state, i.e., $Se_i \leq 1$ for $i = 1,2,.....,6$
- The effect of disturbances on the Se_i value is reduced by the ratio of 100:1 at steady state.

4. Decentralized Control Designs

The model order of the primary mirror system is very large, which means that the computational complexity associated with the implementation of a centralized controller is very high. For example, using an H_∞ or linear quadratic regulator (*LQR)* control design technique the order of the resulting controller is equal or larger than the order of the primary mirror model. For real-time implementation of these high

order controllers, a single processor with high computational power is required. These processors are either expensive or not available for a particular implementation. One way to overcome this problem is to divide the primary mirror system into a number of subsystems and then design a less complex controller (local controller) for each subsystem. These local controllers can be simultaneously implemented using a number of less expensive processors working in parallel. In this work, three types of decentralized controllers are designed and implemented to the primary mirror system. Each local controller is responsible for the control of a single active panel, adding up to a total of 6-local controllers. Our objective is to design and implement different simple decentralized control schemes, compare their performance and computational effort, and come up with candidate controllers that meet the performance requirements with the least computational effort.

4.1 Decentralized State Feedback Proportional Plus Integral (*DSFPI*) Control

For the design of the *DSFPI* control we express the primary mirror model (1) as

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{bmatrix} = \left(\begin{bmatrix} G_{11}(s) & 0 & 0 & 0 & 0 & 0 \\ 0 & G_{22}(s) & 0 & 0 & 0 & 0 \\ 0 & 0 & G_{33}(s) & 0 & 0 & 0 \\ 0 & 0 & 0 & G_{44}(s) & 0 & 0 \\ 0 & 0 & 0 & 0 & G_{55}(s) & 0 \\ 0 & 0 & 0 & 0 & 0 & G_{66}(s) \end{bmatrix} + \begin{bmatrix} 0 & G_{12}(s) & G_{13}(s) & G_{14}(s) & G_{15}(s) & G_{16}(s) \\ G_{21}(s) & 0 & G_{23}(s) & G_{24}(s) & G_{25}(s) & G_{26}(s) \\ G_{31}(s) & G_{32}(s) & 0 & G_{34}(s) & G_{35}(s) & G_{36}(s) \\ G_{41}(s) & G_{42}(s) & G_{43}(s) & 0 & G_{45}(s) & G_{46}(s) \\ G_{51}(s) & G_{52}(s) & G_{53}(s) & G_{54}(s) & 0 & G_{56}(s) \\ G_{61}(s) & G_{62}(s) & G_{63}(s) & G_{64}(s) & G_{65}(s) & 0 \end{bmatrix} \right) \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix}$$

where the diagonal elements in the first matrix represent the dynamics of the 6-active panels when they are isolated from the other panels. The second matrix represents the interactions of the panels with each other. For the purpose of decentralized control design, theses interactions are ignored. The state space representations of the 6-decoupled subsystems are given by

$$\begin{aligned} \dot{x}_i &= \widehat{A}_i x_i + \widehat{B}_i u_i \\ y_i &= \widehat{C}_i x_i \qquad\qquad i = 1,2,.....,6 \end{aligned} \tag{3}$$

where $x_i \in \Re^{40}$ is the *i-th* local state vector, $y_i \in \Re^3$ is the *i-th* local output vector, and $u_i \in \Re^3$ is the *i-th* local input vector. The $\widehat{A}_i, \widehat{B}_i$ and $\widehat{C}_i$ are known matrices of appropriate dimensions satisfying $G_{ii}(s) = \widehat{C}_i(sI - \widehat{A}_i)^{-1}\widehat{B}_i$. The decoupled models (3) are used to design the 6-local state feedback proportional plus integral controllers as

$$\begin{aligned} &local\ state\ estimator : \dot{\hat{x}}_i(t) = \widehat{A}_i\hat{x}_i(t) + \widehat{B}_i u_i(t) + F_i(y_i - \widehat{C}_i\hat{x}_i) \\ &local\ controller : u_i(t) = -k_i\hat{x}_i(t) - l_i\int_0^t y_i(\tau)\,d\tau \qquad i = 1,2,.....,6 \end{aligned}$$

where $\hat{x}_i \in \Re^{40}$ is the estimate of the *i-th* local state vector x_i. The gain matrices $F_i \in \Re^{40\times 3}, k_i \in \Re^{3\times 40}$, and $l_i \in \Re^{3\times 3}$ are obtained following the standard *LQR* plus integral design procedure [3]. The controller gains are varied using different weights in the *LQR* cost until a desired closed-loop response is obtained.

4.2 Decentralized Output Feedback Proportional Plus Integral (*DOFPI*) Control

The state space realization of the designed *DSFPI* control has a large order for each local controller. For this reason, we seek another decentralized control design that has a small order. The obvious way is to consider a *DOFPI* control instead of using a state feedback. In this case, the order of each local controller is reduced to the number of local outputs. In this work, the *DOFPI* control is designed following different approach than the one used to design the *DSFPI* control. In fact, the local controllers are designed using the overall model (2) of the primary mirror system. The 6-local output feedback proportional plus integral controllers are generated by

$$u_i(t) = -\widehat{k}_i y_i(t) - \widehat{l}_i\int_0^t y_i(\tau)\,d\tau \qquad i = 1,2,.....,6$$

where the gain matrices $\hat{k}_i \in \Re^{3\times3}$ and $\hat{l}_i \in \Re^{3\times3}$ are obtained using a Lyapunov inequality equation and the overall system model. The details of computing the controller gains are presented in Appendix A.

4.3 Decentralized Direct Adaptive Output Feedback (*DDAOF*) Control

The structure of the proposed *DDAOF* control is shown in Fig. 4.

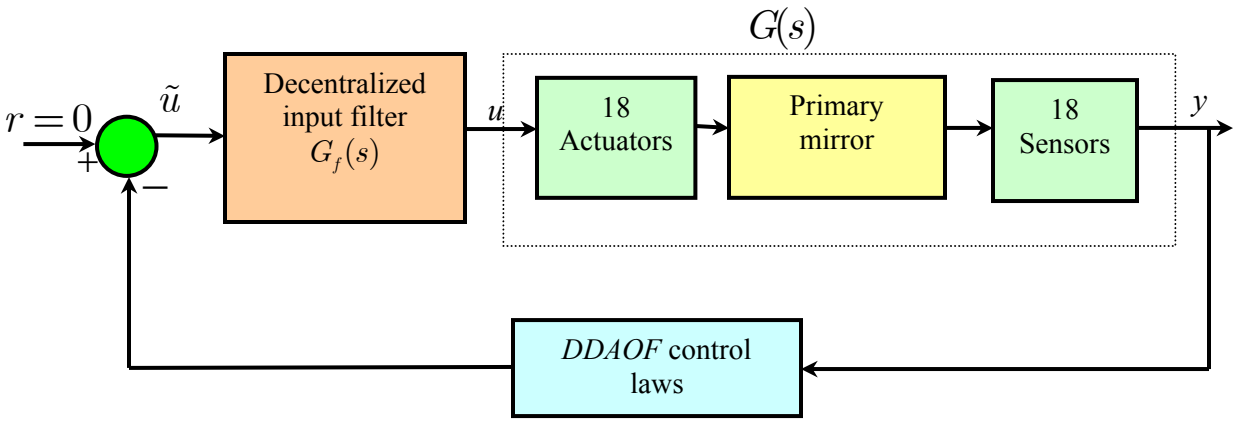

Figure 4. *The primary mirror system with DDAOF control*

In this figure, the decentralized input filter has the following structure $G_f(s) = diag\{G_{f_1}(s), G_{f_2}(s), ..., G_{f_6}(s)\}$ where $G_{f_i}(s) \in \mathbb{C}^{3\times3}$ is the *i-th* local filter. The *DDAOF* control laws are generated as follows

$$u_i = G_{f_i}(s)\,\tilde{u}_i$$

$$\tilde{u}_i(t) = -\frac{k_i(t) + l_i(t)}{1 + \delta\,(k_i(t) + l_i(t))}\,y_i(t)$$

$$k_i(t) = \alpha_i y_i^T(t) y_i(t)$$

$$\dot{l}_i(t) = \beta_i y_i^T(t) y_i(t)$$

where i =1, 2,…, 6, $k_i \in \Re$, $l_i \in \Re$, and $\delta, \alpha_i, \beta_i$ are design positive constants. The filter $G_f(s)$ and scalar δ are designed such that $G(s)G_f(s) + \delta I$ is strictly positive real (*SPR*). The proposed *DDAOF* control design is motivated from [4], [5]. The passivity is required to ensure the stability of the closed-loop system

[6]. In Appendix B we present the design details and stability analysis of the proposed *DDAOF* control scheme.

5. Experimental Results

The three controllers presented in the previous section are discretized first using a sampling period of 1 ms and then implemented to the primary mirror system. The flowchart of the decentralized control algorithm is shown in Fig. 5 where the 3-*DSPs* are used in parallel to implement the proposed decentralized controllers.

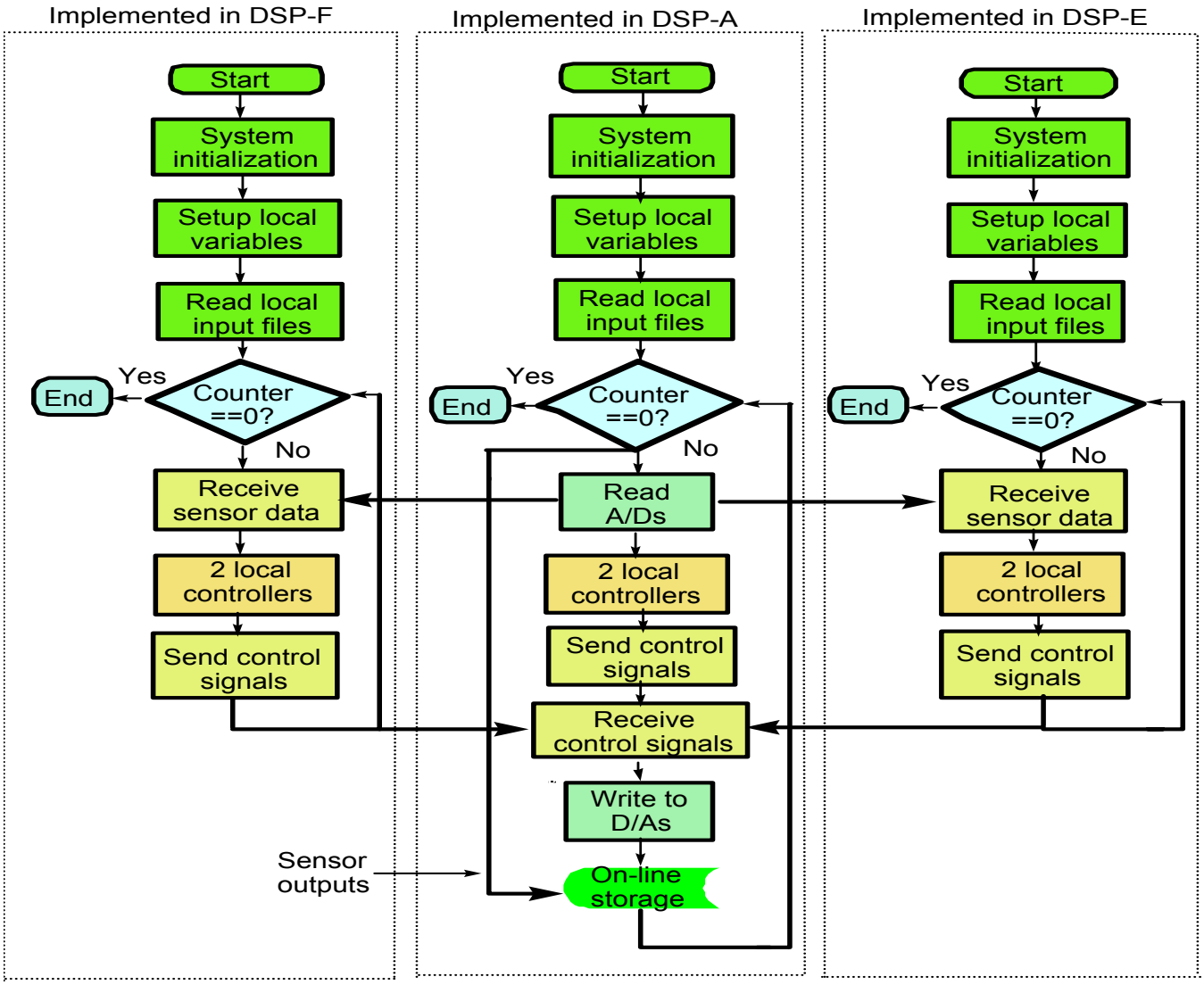

Figure 5. *The flowchart of decentralized control algorithm*

The real-time results for the three controllers are shown in Figures: 6, 7, and 8 for the 18-edge sensor outputs, 18-control commands, and 6-shape errors. In order to examine the effectiveness of the controllers, initial shape errors are created on the position of the primary mirror panels by adding constant loads on the panels. For the *DSFPI* control, the number of states for each subsystem is 40. The number of computational operations required to implement the *DSFPI* control is large and cannot be handled by the available *DSPs*. For this reason, we reduced the order of the 6-decoupled system models and designed new controllers with 17-states for each local controller. The 18-edge sensor outputs, 18-control commands, and 6-shape errors for the closed-loop system with reduced order *DSFPI* control are shown in Fig. 6.

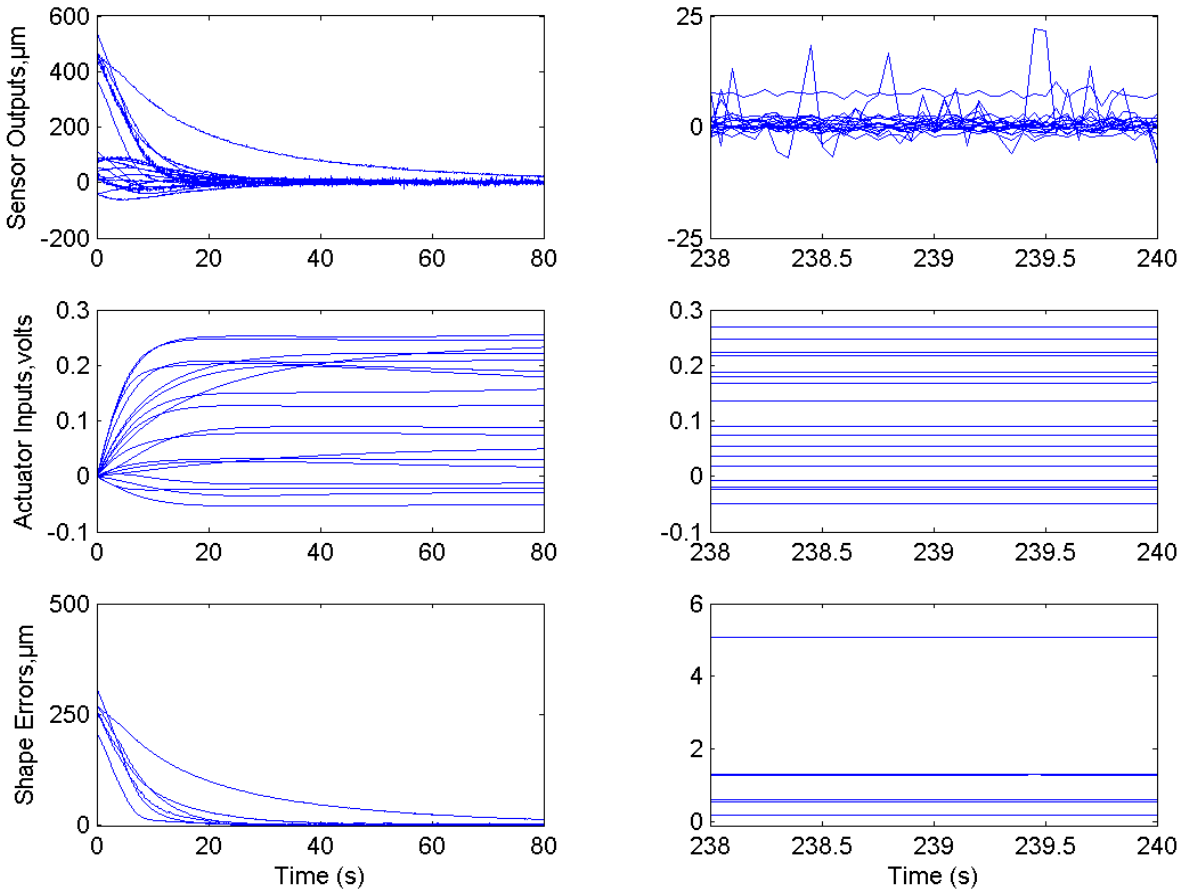

Figure 6. *Closed-loop real-time results with DSFPI control (right column shows the steady state results)*

These plots indicate that the closed-loop system is stable, the effect of the disturbances is reduced, and the control effort for all 18-actuators is within the limits (+/-1 volt). For the *DOFPI* controller, the number of computational operations is dramatically reduced when compared with the *DSFPI* controller, since the number of controller states is now 3 for each local controller. The closed-loop system performance however is worst than that of the *DSFPI* controller as shown in Fig. 7.

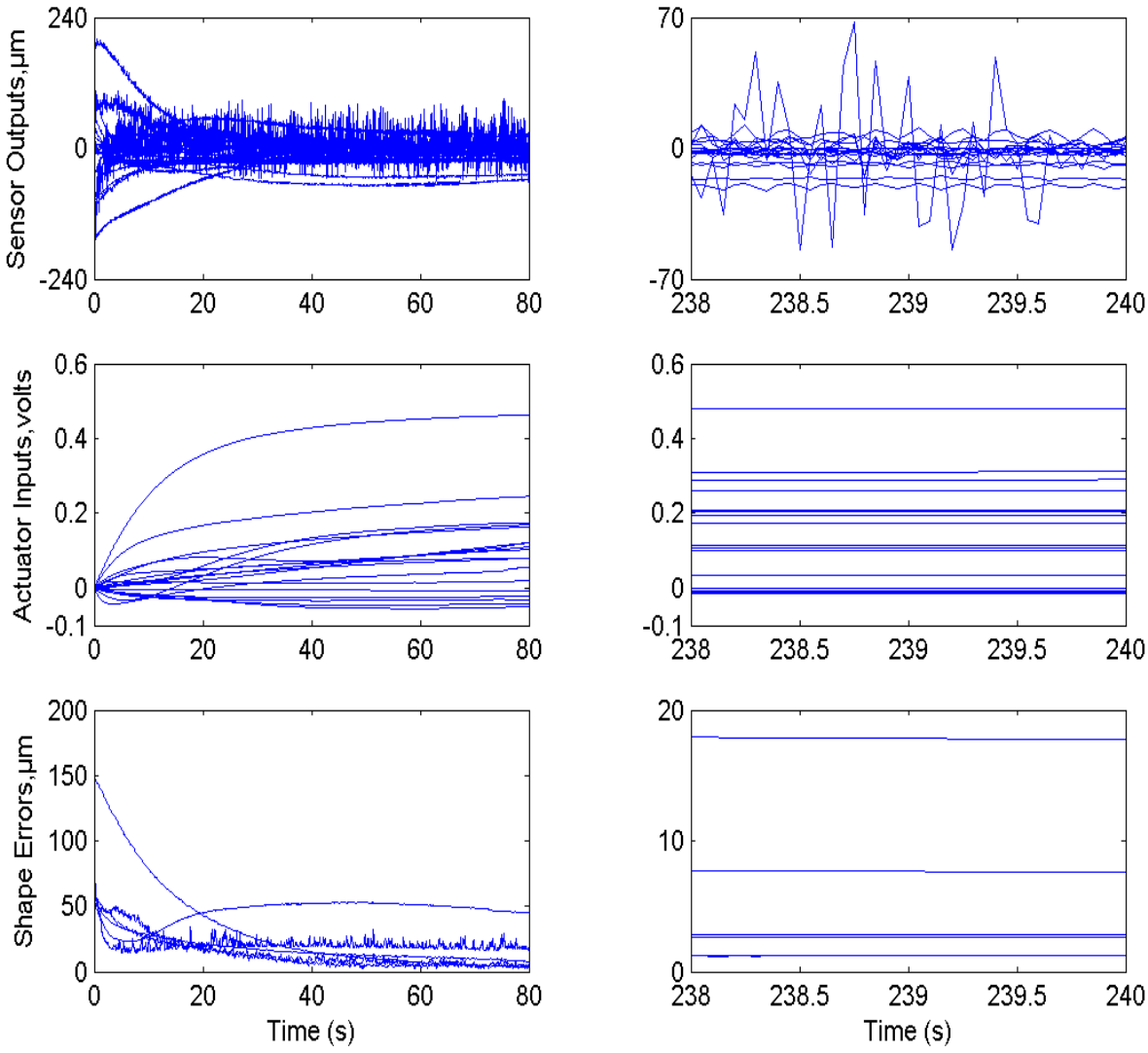

Figure 7. *Closed-loop real-time results with DOFPI control (right column shows the steady state results)*

The implementation results for the proposed *DDAOF* controller are shown in Fig. 8. The *DDAOF* controller reduces the effect of the disturbance on the primary mirror panels by 100:1 faster than the other two designs.

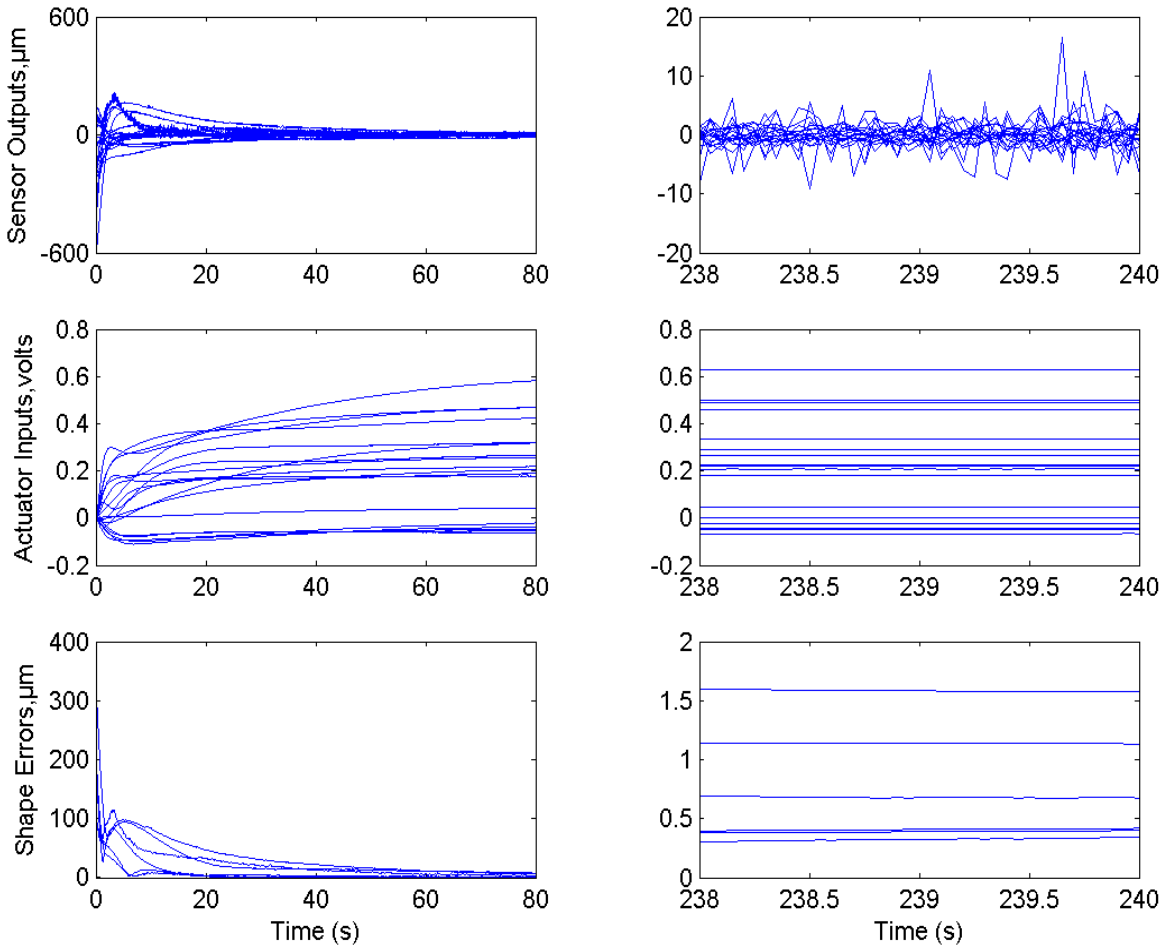

Figure 8.*Closed-loop real-time results with DDAOF control (right column shows the steady state results)*

Table 1 shows the comparison between the three types of controllers.

Panel no.	Initial shape error values in μm.			Shape error values in μm at 240 s.		
	DSFPI	*DOFPI*	*DDAOF*	*DSFPI*	*DOFPI*	*DDAOF*
1	277.339	61.640	107.979	1.43	2.6	0.66
2	318.624	71.132	251.199	0.72	15.7	0.72
3	259.205	151.658	363.237	1.01	1.44	1.52
4	214.905	52.087	233.563	0.27	1.94	1.17
5	272.448	55.378	159.361	4.57	2.69	0.5
6	278.812	101.999	124.334	2.08	7.20	0.3

TABLE 1: *The shape error values for the three types of controllers*

The time it takes to compute any 18-control command samples by the 3-*DSPs* (the sampling time is 1 ms) is

- 0.78 ms for the *DSFPI* control
- 0.2 ms for the *DOFPI* control
- 0.29 ms for the *DDAOF* control.

These results indicate that the *DOFPI* and *DDAOF* controllers take much less computational power of the 3-*DSPs* than the *DSFPI* controller. However, the *DOFPI* controller fails to meet the performance requirements. Overall the *DDAOF* controller performs better and required low computational effort.

6. Conclusions

In this article, we use the test-bed developed at California State University, Los Angeles, that simulates in real-time a large segmented telescope to test the performance and computational requirements of several control designs. Three decentralized control designs were selected for implementation. These include a decentralized state feedback proportional plus integral (*DSFPI*) controller, a decentralized output feedback proportional plus integral (*DOPFI*) controller, and a decentralized direct adaptive output feedback (*DDAOF*) controller.

The *DSFPI* controller requires more memory space and computational power than the *DOFPI* and *DDAOF* controllers. The *DOFPI* requires less memory space and computational operations but it fails to meet the performance requirements. The *DDAOF* requires an acceptable amount of memory space and computational operations and has better performance than the other two controllers.

Acknowledgements

We would like to thank *Dr*. Boussalis, *Dr*. Mirmirani, *Mr*. Rad from California State University, Los Angeles (*CSULA*), and *Dr*. Kun Li former *Ph.D*. student of the University of Southern California, Los Angeles, for numerous discussions and support. We also acknowledge the support of Demetrios, Salvador, Alex, and the rest of the students in the *SPACE* lab at *CSULA*.

Appendix A: Decentralized Output Feedback Proportional Plus Integral (*DOFPI*) Control

Consider the state space representation of a system given by

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{4}$$

where $x \in \Re^n$ is the system state vector, $u = \begin{bmatrix} u_1^T & u_2^T & \dots & u_m^T \end{bmatrix}^T \in \Re^q$ is the input vector, and $y = \begin{bmatrix} y_1^T & y_2^T & \dots & y_m^T \end{bmatrix}^T \in \Re^q$ is the output vector. The matrices A, B, and C are of appropriate dimensions. We differentiate the state equation (4) with respect to time *t* and express it as

$$\begin{aligned} \dot{v} &= A_v v + B_v \dot{u} \\ y &= C_v v \end{aligned}$$

where

$$v = [\dot{x}^T y^T]^T,\ A_v = \begin{bmatrix} A & 0 \\ C & 0 \end{bmatrix}, B_v = \begin{bmatrix} B \\ 0 \end{bmatrix}, C_v = \begin{bmatrix} 0 & I \end{bmatrix}.$$

We propose the following structure for the local controllers

$$u_i(t) = -\hat{k}_i y_i(t) - \hat{l}_i \int_0^t y_i(\tau)\, d\tau \qquad i = 1,2,\dots\dots,m \tag{5}$$

where the gain matrices $\hat{k}_i$, and $\hat{l}_i$ are to be selected. The local controllers (5) can be written in a compact form as $\dot{u} = -F v$ where *F* has the following structure

$$F = \begin{bmatrix} \widehat{k}_1 C_1 & \widehat{l}_1 & 0 & \cdots & 0 \\ \widehat{k}_2 C_2 & 0 & \widehat{l}_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ \widehat{k}_m C_m & 0 & 0 & \cdots & \widehat{l}_m \end{bmatrix}$$

where $C = [C_1^T, C_2^T, ..., C_m^T]^T$. The following Lyapunov inequality equation is used to compute the constant gain matrix *F* that stabilizes the closed-loop system.

$$P(A_v - B_v F) + (A_v - B_v F)^T P < 0 \tag{6}$$

The problem is to find a positive definite matrix *P* and the controller gain *F* satisfying (6). This problem is not easy to solve because the Lyapunov inequality equation (6) is not linear in terms of *P* and *F*. However, if we fix $P > 0$ then the Lyapunov inequality is converted to a linear matrix inequality (*LMI*), which can be easily solved for *F*.

The design procedure for the *DOFPI* is summarized in the following steps

I. Add and subtract each of PA_e to the first part and $A_e^T P$ to the second part of the inequality (6) to get

$P(A_v + A_e - A_e - B_v F) + (A_v + A_e - A_e - B_v F)^T P < 0$.

The matrix A_e is selected such that all eigenvalues of $A_v + A_e$ have a negative real part. This modification is required for the next step because A_v has a number of eigenvalues at zero. For example one can select $A_e = \begin{bmatrix} 0 & 0 \\ 0 & \varepsilon I \end{bmatrix}$, where ε is a negative scalar

II. Find the positive definite matrix *P* satisfying: $P(A_v + A_e) + (A_v + A_e)^T P = -Q$, where Q is any positive definite matrix. For example select $Q = I$

III. Solve $P(-A_e - B_v F) + (-A_e - B_v F)^T P < Q$ for F using the *LMI-toolbox* [7].

Appendix B: Decentralized Direct Adaptive Output Feedback (*DDAOF*) Control

Consider the proposed *DDAOF* controller shown in Fig. 9.

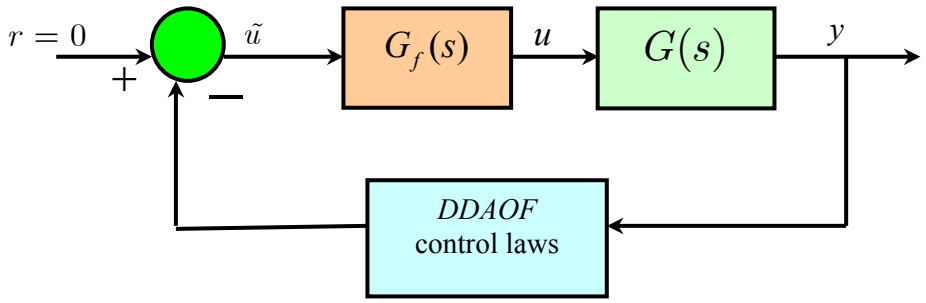

Figure 9. *The proposed DDAOF controller*

The state space realization of the filter $G_f(s)$ is given by

$$\dot{\tilde{x}} = \tilde{A}\,\tilde{x} + \tilde{B}\,\tilde{u}$$
$$u = \tilde{C}\,\tilde{x} + \tilde{D}\,\tilde{u}$$

where

$$\tilde{x} = \begin{bmatrix} \tilde{x}_1^T & \dots & \tilde{x}_i^T & \dots & \tilde{x}_m^T \end{bmatrix}^T \in R^{\tilde{n}}$$
$$\tilde{u} = \begin{bmatrix} \tilde{u}_1^T & \dots & \tilde{u}_i^T & \dots & \tilde{u}_m^T \end{bmatrix}^T \in R^{q}$$
$$\tilde{A} = diag\begin{Bmatrix} \tilde{A}_1 & \dots & \tilde{A}_i & \dots & \tilde{A}_m \end{Bmatrix}$$
$$\tilde{B} = diag\begin{Bmatrix} \tilde{B}_1 & \dots & \tilde{B}_i & \dots & \tilde{B}_m \end{Bmatrix}$$
$$\tilde{C} = diag\begin{Bmatrix} \tilde{C}_1 & \dots & \tilde{C}_i & \dots & \tilde{C}_m \end{Bmatrix}$$
$$\tilde{D} = diag\begin{Bmatrix} \tilde{D}_1 & \dots & \tilde{D}_i & \dots & \tilde{D}_m \end{Bmatrix}$$

$\tilde{A}_j, \tilde{B}_j, \tilde{C}_j, \tilde{D}_j$ are constant matrices of appropriate dimensions and $G_f(s) = \tilde{C}(sI - \tilde{A})^{-1}\tilde{B} + \tilde{D}$.

Let $G_m(s) = G(s)G_f(s) + \delta\, I$ where δ is a scalar. The state space representation of $G_m(s)$ is given by

$$\begin{aligned} \dot{\hat{x}} &= \hat{A}\,\hat{x} + \hat{B}\,\tilde{u} \\ \tilde{y} &= \hat{C}\,\hat{x} + \delta\ \tilde{u} = y + \delta\ \tilde{u} \end{aligned} \tag{7}$$

where

$$\hat{x} = \begin{bmatrix} x^T & \tilde{x}^T \end{bmatrix}^T \in R^{n+\tilde{n}},\ \hat{A} = \begin{bmatrix} A & B\tilde{C} \\ 0 & \tilde{A} \end{bmatrix},\ \hat{B} = \begin{bmatrix} B\tilde{D} \\ \tilde{B} \end{bmatrix}, \hat{C} = \begin{bmatrix} C & 0 \end{bmatrix},$$

and *A, B, C* are from the state space realization of $G(s) = C(sI - A)^{-1}B$

***Theorem* 1:** If there exist a scalar δ and filter $G_f(s)$ such that $G_m(s) = G(s)G_f(s) + \delta\, I$ is *SPR*, then the following *DDAOF* control laws

$$\begin{aligned} & u_i = G_{f_i}(s)\,\tilde{u}_i && i = 1,2,\ldots,m \\ & \tilde{u}_i(t) = -\frac{k_i(t) + l_i(t)}{1 + \delta\ (k_i(t) + l_i(t))}\, y_i(t) \\ & k_i(t) = \alpha_i y_i^T(t) y_i(t) \\ & \dot{l}_i(t) = \beta_i y_i^T(t) y_i(t) \end{aligned} \tag{8}$$

where α_i and β_i are design positive constants, can stabilize the system of Fig. 9 and force the output vector y to zero exponentially fast.

The following Lemma is used to prove Theorem 1.

***Lemma* 1-Kalman-Yakubovich-Popov lemma [8]:** The transfer function matrix $G_m(s) = \hat{C}(sI - \hat{A})^{-1}\hat{B} + \delta\ I$ is *SPR* if and only if there exist matrices $P = P^T > 0$, *L*, and *W*, and a constant ε > 0 such that

$$
\begin{aligned}
&P\hat{A} + \hat{A}^T P = -L^T L - \varepsilon P \\
&P\hat{B} = \hat{C}^T - L^T W \\
&W^T W = 2\delta\ I
\end{aligned} \tag{9}
$$

Proof of Theorem* 1*: Use

$$V(\hat{x}) = \frac{1}{2}\hat{x}^T P\,\hat{x} \tag{10}$$

as a Lyapunov function candidate. The derivative of (10) along the trajectories of the system (7) is given by

$$\dot{V}(\hat{x}) = \frac{1}{2}\hat{x}^T P\,\dot{\hat{x}} + \frac{1}{2}\dot{\hat{x}}^T P\,\hat{x} \tag{11}$$

Substituting (7) into (11), we obtain

$$\dot{V}(\hat{x}) = \frac{1}{2}\hat{x}^T(P\hat{A} + \hat{A}^T P)\,\hat{x} + \hat{x}^T P\hat{B}\,\tilde{u}$$

Using (9) from Lemma 1, yields

$$
\begin{aligned}
\dot{V}(\hat{x}) &= -\frac{1}{2}\hat{x}^T L^T L\,\hat{x} - \frac{1}{2}\varepsilon\,\hat{x}^T P\,\hat{x} + \hat{x}^T(\hat{C}^T - L^T W)\tilde{u} \\
&= -\frac{1}{2}\hat{x}^T L^T L\,\hat{x} - \frac{1}{2}\varepsilon\,\hat{x}^T P\,\hat{x} + \hat{x}^T(\hat{C}^T - L^T W)\tilde{u} + \delta\,\tilde{u}^T\tilde{u} - \delta\,\tilde{u}^T\tilde{u} \\
&= -\frac{1}{2}\hat{x}^T L^T L\,\hat{x} - \frac{1}{2}\varepsilon\,\hat{x}^T P\,\hat{x} + (\hat{C}\,\hat{x} + \delta\ \tilde{u})^T\tilde{u} - \delta\,\tilde{u}^T\ \tilde{u} - \hat{x}^T L^T W\,\tilde{u}
\end{aligned}
$$

Using (9) from Lemma 1, we obtain

$$\dot{V}(\hat{x}) = -\frac{1}{2}\varepsilon\,\hat{x}^T P\,\hat{x} - \frac{1}{2}(L\,\hat{x} + W\,\tilde{u})^T(L\,\hat{x} + W\,\tilde{u}) + \tilde{y}^T\tilde{u} \tag{12}$$

Substituting $\tilde{y}^T\ \tilde{u} = \tilde{y}_1^T\tilde{u}_1 + \ldots + \tilde{y}_j^T\tilde{u}_j + \ldots + \tilde{y}_m^T\tilde{u}_m = \sum_{i=1}^{m}\tilde{y}_i^T\tilde{u}_i$ into (12), we have

$$\dot{V}(\hat{x}) = -\frac{1}{2}\varepsilon\,\hat{x}^T P\,\hat{x} - \frac{1}{2}(L\,\hat{x} + W\,\tilde{u})^T(L\,\hat{x} + W\,\tilde{u}) + \sum_{i=1}^{m}\tilde{y}_i^T\tilde{u}_i \tag{13}$$

Using the local controllers of Theorem 1, we have

$$\begin{aligned}
&(1+\delta\ (k_i(t)+l_i(t)))\,\tilde{u}_i(t) = -(k_i(t)+l_i(t))\,y_i(t)\\
&or\\
&\tilde{u}_i(t) = -(k_i(t)+l_i(t))(y_i(t)+\delta\,\tilde{u}_i(t))\\
&\qquad = -(k_i(t)+l_i(t))\,\tilde{y}_i(t)
\end{aligned} \tag{14}$$

Substituting (14) into (13), we have

$$\dot{V}(\hat{x}) = -\frac{1}{2}\varepsilon\,\hat{x}^T P\,\hat{x} - \frac{1}{2}(L\,\hat{x} + W\,\tilde{u})^T(L\,\hat{x} + W\,\tilde{u}) - \sum_{i=1}^{m}(k_i + l_i)\tilde{y}_i^T\tilde{y}_i$$

Since $(k_i + l_i)\tilde{y}_i^T\tilde{y}_i \geq 0$, we have

$$\dot{V}(\hat{x}) \leq -\frac{1}{2}\varepsilon\,\hat{x}^T P\,\hat{x}$$

which implies that $\|\hat{x}(t)\|$ is bounded and converges to zero exponentially fast. Since $\hat{x} = \begin{bmatrix} x^T & \tilde{x}^T \end{bmatrix}^T$ we also have $\|x(t)\|$, $\|\tilde{x}(t)\|$ going to zero exponentially fast. Since $y = Cx$ we have that $\|y(t)\|$ going to zero exponentially fast. The above analysis also implies that x, $\tilde{x}$ and y are square integrable i.e. they are L_2-signals. From (8) we have $l_i(t) = \beta_i\int_0^t y_i^T(\tau)y_i(\tau)\,d\tau$, since $y_i \in L_2$ it follows that $l_i(t)$ is bounded and $\lim_{t\to\infty} l_i(t) = \beta_i\int_0^\infty y_i^T(\tau)y_i(\tau)\,d\tau \triangleq \overline{l_i} < \infty$. Hence all signals are bounded in addition to $\|y_i(t)\|$ converging to zero exponentially fast ▪

***Remark* 1:** The following lemma shows that there always exists a scalar δ such that $G_m(s)$ is *SPR*.

***Lemma* 2:** For any $n \times n$ proper transfer function matrix $\tilde{G}_m(s) = \hat{C}(sI - \hat{A})^{-1}\hat{B}$ with all elements analytic in the closed right-half complex plane, there exists a scalar δ such that $G_m(s) = \tilde{G}_m(s) + \delta I$ is *SPR*.

The following Definition is used for the proof of Lemma 2.

***Definition* 1[9]:** Let $G_m(s)$ be an $n \times n$ transfer function matrix and also let $G_h(jw) = G_m(jw) + G_m^T(-jw)$. Then $G_m(s)$ is *SPR* if

1. $G_m(s)$ is analytic in the closed right-half complex plane
2. $G_h(jw) > 0 \quad \forall\, w \in (-\infty, \infty)$
3. $G_h(\infty) \geq 0$
4. $\lim_{w \to \infty} w^2 G_h(jw) > 0$ if $G_h(\infty)$ is singular.

***Proof of Lemma* 2:** Since we assumed that all elements of $G_m(s)$ are analytic in the closed right-half complex plane, therefore condition 1 of definition 1 is satisfied.

Now $G_m(s)$ is *SPR* if $G_h(jw)$ is positive definite for any real *w* or the eigenvalues of $G_h(jw)$ are positive for any real *w* (including $w = \infty$).

The characteristic equation of $G_h(jw)$ at each *w* is

$$\begin{aligned}\Delta(\lambda_w, w) &\triangleq \left|\lambda_w I - G_h(jw)\right| \\ &= \left|\lambda_w I - (\tilde{G}_m(jw) + \tilde{G}_m^T(-jw) + 2\delta I)\right| \\ &= \left|(\lambda_w - 2\delta)I - (\tilde{G}_m(jw) + \tilde{G}_m^T(-jw))\right| \\ &= 0\end{aligned}$$

where λ_w is the eigenvalue of $G_h(jw)$ at each *w*. Let $\lambda_{\min}$ be the minimum eigenvalue of $\tilde{G}_m(jw) + \tilde{G}_m^T(-jw) \quad \forall\, w \in (-\infty, \infty)$, then for

$$\delta > -\frac{1}{2}\lambda_{\min}$$

we have $\lambda_w > 0$. Therefore condition 2 of definition 1 is satisfied. For condition 3 of definition 1 we have $G_h(\infty) = 2\delta I \geq 0$ for $\delta \geq 0$. Therefore condition 3 is satisfied. Choosing $\delta > \max\{0, -\frac{1}{2}\lambda_{\min}\}$ we satisfied conditions 2 and 3 as well as condition 4 ▪

***Remark* 2:** The following Lemma can be used to find the constant matrices of the filter and δ analytically.

***Lemma* 3 [10]:** The transfer function matrix $G_m(s) = \hat{C}(sI - \hat{A})^{-1}\hat{B} + \delta I$ is *SPR* if and only if there exists a positive definite matrix *H* such that

$$\begin{bmatrix} \hat{A}H + H\hat{A}^T & \hat{B} - H\hat{C}^T \\ \hat{B}^T - \hat{C}H & -2\delta I \end{bmatrix} < 0 \qquad (15)$$

The matrix inequality (15) is not linear in terms of *H* and the filter matrices and therefore it is not easy to solve. However, if we select some filter matrices we can convert it to an *LMI* problem. For example, choose $\tilde{A} = \alpha I \ and \ \tilde{C} = I$ where α is some negative value. In this case the matrix inequality (15) can be easily solved for the other design matrices and δ using the *LMI toolbox* [7].

References

[1] D.D. Siljak, *Decentralized Control of Complex Systems*, New York, NY: Academic Press, 1991.

[2] K. Li, "Modeling and Overlapping Decentralized Control of a Large Segmented Telescope Test-Bed," Ph.D. Thesis, Dept. of Electrical Engineering-System, University of Southern California, 2001.

[3] B. Anderson, and J. Moore, *Optimal Control*, New Jersey, NJ: Prentice-Hall, 1990.

[4] I. Bar-Kana, "Parallel feed-forward and simplified adaptive control," *Int. J. of Adap. Contr. and Sign. Proc.,* vol. 1, pp. 95-109, 1987.

[5] C. Huang, P.A. Ioannou, J. Maroulas, and M. Safonov, "Design of strictly positive real systems using constant output feedback," *IEEE Trans. on Automat. Contr.*, vol. 44, no. 3, pp. 569-573, 1999.

[6] H. K. Khalil, *Nonlinear Systems*, Upper Saddle River, NJ: Prentice-Hall, 1996.

[7] P. Gahinet, A. Nemirovski, A. Laub, M. Chilali, *LMI Control Toolbox*, MathWorks, Inc.: Mass., 1995.

[8] P. Ioannou and J. Sun, *Robust Adaptive Control*, Upper Saddle River, NJ: Prentice-Hall, 1996.

[9] G. Tao and P. A. Ioannou, "Strictly positive real matrices and the Lefschetz-Kalman-Yakubovich lemma," *IEEE Trans. on Automat. Contr.*, vol. 33, no. 12, pp. 1183-1185, 1988.

[10] W. Sun, P. Khargonekar, and D. Shim, "Solution to the positive real control problem for linear time-invariant systems," *IEEE Trans. on Automat. Contr.*, vol. 39, no. 10, pp. 2034-2046, 1994.