JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 17, No. 3, May–June 1994

Integrated Modeling and Control for the Large Spacecraft Control Laboratory Experiment Facility

Guoming Zhu and Robert E. Skelton
Purdue University, West Lafayette, Indiana 47907

A software package which integrates model reduction and controller design is applied to design controllers for the Jet Propulsion Laboratory Large Spacecraft Control Laboratory experiment facility. Modal cost analysis is used for the model reduction, and various output covariance constraints are guaranteed by the controller design. The main motivation is to find the controller with the best performance with respect to output covariances. It is shown that by iterating on the reduced-order design model, the controller designed does have better performance than that obtained with the first model reduction, demonstrating an effective strategy for integrating modeling and control design.

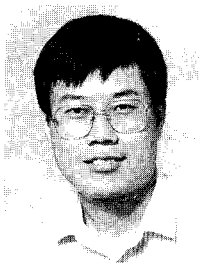
Introduction

THE objective of this research is to develop controller design software integrated modeling and control (IMC) for a realistic flexible space structure control problem. The main interests are two-fold: the design of high-performance, fixed-order dynamic controllers for this complex structure, and to test the efficacy of the IMC software for the search of the controller with the best performance, among all model-based controllers.

Almost all available controller design techniques are based on a given model of the physical plant. In general, perfect models are impossible to construct. Modeling error exists in every mathematical model used for control design. There are three ways to deal with modeling error in a controller design procedure. First, one may use robust control theory. The controller designed with robust control theory is tolerant to a specified set of modeling errors. But a poor model may lead to a poor controller even if the controller is robust with respect to the given model. Second, one may treat the modeling and controller design as a combined problem, and try to refine the design model to find one that is appropriate for controller design in the sense of best closed-loop operation. The third method is adaptive control which intends to adjust the controller in real time to compensate for modeling errors.

From our experience a nominal controller design procedure based on an appropriate model may yield better performance than a robust controller that is based on a poor model (say, given by finite element modeling or identification). Hence, we use the second method to obtain a design model that is more compatible to the particular controller design than the other two methods.

In this research the integrated design procedure is applied to design controllers for the Large Spacecraft Control Laboratory (LSCL) experiment facility. Assuming that a true enough high-order mathematical model can be obtained by some modeling method (either analytical or by identification), our procedure reduces the true enough model (we shall call this the evaluation model) to an order appropriate for full-order controller design based on the reduced-order model. Repeating the model reduction and controller design by using closed-loop information such that the process is convergent, the integrated procedure produces a design model appropriate to the corresponding controller.



Guoming Zhu received the B.S. and M.S. degrees in mechanical and electrical engineering from the Beijing University of Aeronautics and Astronautics, People's Republic of China, in 1982 and 1984, respectively, and the Ph.D. degree in Aerospace Engineering from Purdue University in 1992. Now he is a post doctoral research associate in the Space Systems Control Lab at Purdue University. His professional interests are multiobjective optimal control, robust control, and control theories for discrete periodic and multirate systems.

Robert E. Skelton received the B.S.E.E. degree from Clemson University in 1963, the M.S.E.E. degree from the University of Alabama at Huntsville in 1969, and the Ph.D. degree in mechanics and structures from the University of California, Los Angeles. After 12 years in the space industry (Lockheed Missiles and Space Co., and Sperry Rand Corporation), he has taught dynamics and control at Purdue University since 1975, where he is currently Director of the Space Systems Control Laboratory. He served on the National Research Council's Aeronautics and Space Engineering Board, and has received awards from the Japan Society for the Promotion of Science in 1986, the Alexander von Humboldt Foundation in 1991, and held the Russell Serverance Springer chair at University of California, Berkeley in 1991.

Received July 13, 1992; revision received Aug. 9, 1993; accepted for publication Sept. 26, 1993. Copyright © 1993 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved.

The model reduction technique used in this experiment is the modal cost analysis (MCA), which calculates each modal contribution ν_i to a weighted quadratic cost function.[7-9]

$$\nu \triangleq \lim_{k\to\infty} Ey^T(k)Qy(k) = \sum_{i=1}^{N} \nu_i \qquad (1)$$

where E is an expectation operator, and N is the number of modes in the model. The smallest contribution (smallest ν_i) indicates the modes to be deleted in the reduced model. Closed and analytical expression[8] of ν_i are available.

Two controller design methods [block output covariance constraint (OCC) and extended output L_∞ constraint (EOL_∞)] were applied to this experiment. The block OCC algorithm[1-4] designs controllers minimizing the control effort subject to output covariance constraints (for zero mean white noise input). The block OCC algorithm can be also used to satisfy the output l_∞ constraints when the input is an l_2 disturbance. The EOL_∞ algorithm[5] is an extension of the deterministic interpretation of the block OCC. The EOL_∞ designs controllers to satisfy given output l_∞ constraints when the input l_2 disturbances have an outer product matrix upper bound. The main difference between those two design algorithms is that the block OCC algorithm only iterates on the feedback gain, but the EOL_∞ algorithm iterates on both estimator and control feedback gains. We only present the block OCC results in this paper. The definition and solution of the block OCC and EOL_∞ can be found in Refs. 3–5.

There are two iteration loops in the IMC software, one inner loop and one outer, used to realize the integration of model reduction and controller design. The inner loop, called the performance tuning loop, intends to obtain the controller for the best performance (with respect to the evaluation model) with the given reduced-order model (called the design model) by gradually increasing the required performance (smaller covariance constraints). The outer loop iterates on the design model to make the design model appropriate to the corresponding controller with the best performance.

Integration of Model Reduction and Controller Design

It is well known that finding a good model for control design is a difficult problem because of uncertain parameters, nonlinearity, and neglected dynamics of the physical system. It is impossible to separate the modeling and controller design problems. For example, considering a linear system with a nonlinear actuator, one may apply linear control theory to design a controller. In this case the nonlinear actuator should be linearized at some nominal point, but the nominal point is related to the control signal level of the controller which will be designed after linearization of the actuator model. Consequently, the modeling and controller design problems become an iterative process.[6]

In this section we mainly consider the effect of the neglected dynamics of the physical system. We are trying to obtain the best performance for a high order physical system with a fixed-order controller. There are at least three ways to find a fixed-order controller for a given linear system. The first way is to design a fixed-order controller directly. The second is to design a full-order controller first and then reduce the controller to the required order. The last one is to reduce the model first and then perform the full-order control design based on the reduced-order model. The advantage of the first method is that the performance of the closed-loop system with the designed controller is guaranteed. Unfortunately no closed form exists for the design of such controllers. Because full-order controller design methods are available for most control theories, LQG, H_∞, H_2/H_∞, and so on, we will use a variation of the third method, we call the integration of model reduction and controller design, to design reduced-order controllers.

The integrated design procedure, utilizing MCA for model reduction and the block OCC or EOL_∞ for controller design, is shown in Fig. 1. The design procedure searches for the controller with the best performance by tuning the design model until the design model corresponds to the controller with the best performance. This procedure is developed under the following basic assumption that the only modeling errors existing in the design model are from the model reduction, i.e., the evaluation model is assumed to be true enough. This assumption allows us to evaluate the designed controller based on the evaluation model, prior to hardware testing in the lab. Of course, we also compare these analytical results with the experimental results.

The evaluation model in Fig. 1 can be obtained either from system identification or from mathematical modeling, e.g., the finite-element model combining with the sensor and actuator dynamics. Generally, the size of the evaluation model is too large for controller design. Hence, the model reduction is necessary.

The cost function defined in Eq. (1) used in the model reduction is the summation of the weighted output variance with respect to the white noise input. Note that the modal cost is very much dependent upon the input and output weighting matrices $\bar{W}$ = block diag $[W, R]$ and Q, where the input weighting matrix $\bar{W}$ is used to compute the output covariance. Hence, the choice of those two matrices will directly affect the model reduction. How to choose Q and $\bar{W}$ is a major subject of this paper. For the first iteration of this experiment, matrix W is the input white noise covariance matrix W_p, and Q and R are diagonal matrices whose elements are the inversed square of the hard limitation on inputs and outputs, respectively.

The main philosophy of our performance tuning loop in Fig. 2 is to obtain a sequence of controllers from low to high control effort. The controller sequence is obtained by reducing the required performance specification during controller design.

The main purpose of the block OCC performance tuning loop is to obtain the best performance with the given (reduced-order) design model (obtained from MCA model reduction of the evaluation model), which is expressed in the following form:

$$\begin{cases} x_p(k+1) &= A_p x_p(k) + B_p u(k) + D_p w_p(k) \\ y_p(k) &= C_p x_p(k) \\ z(k) &= M_p x_p(k) + v(k) \end{cases} \qquad (2)$$

The block OCC performance tuning loop starts with the evaluation and design models. Suppose that the output y_p can be divided into m output groups $\hat{y}_i$. Let $Y_i(0)$ $(i = 1, 2, \ldots, m)$ denote the open-loop output covariance of the evaluation model

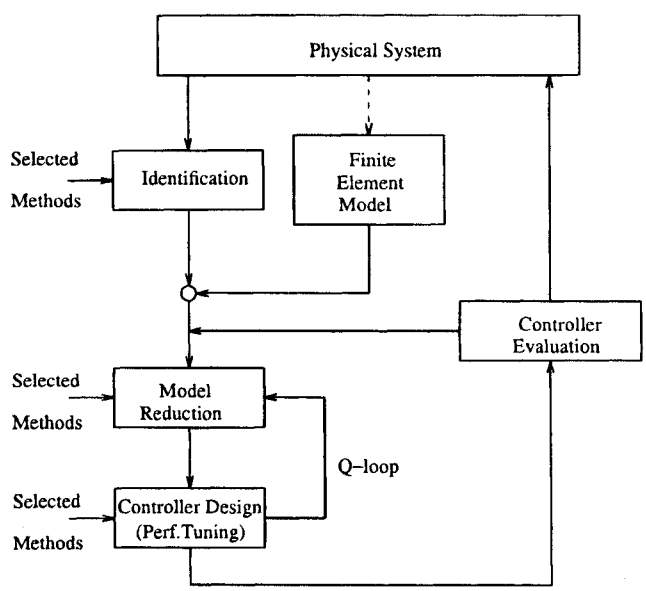


Fig. 1 Flowchart of the IMC software.

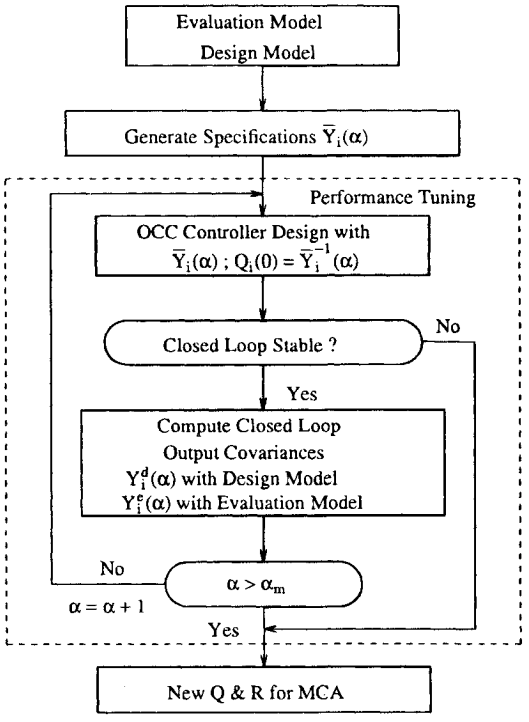


Fig. 2 Block OCC control design with performance tuning.

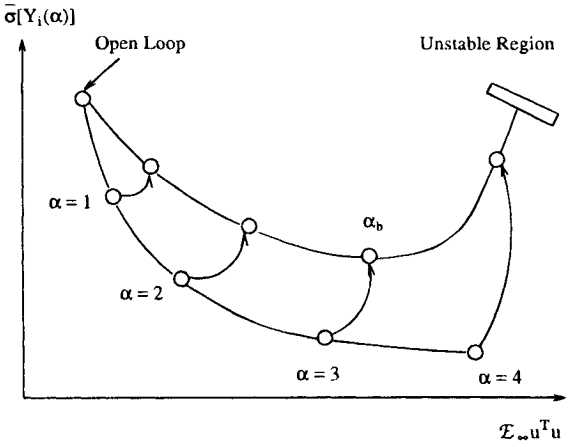


Fig. 3 Tradeoff of output covariance and input variances: lower curve, output covariances with respect to the design model, upper curve, output covariances with respect to the evaluation model.

for output groups $\hat{y}_i$, assuming that the open-loop system is asymptotically stable. Define L_i ($i = 1, 2, \ldots, m$) to be a lower bound of the output covariance of the closed-loop system with any full-order controller. Hence, any specification which is less than or equal to L_i is unachievable with respect to the design model. Then the specification matrix $\bar{Y}_i(\alpha)$ ($i = 1, 2, \ldots, m$) can be generated by the following equation

$$\bar{Y}_i(\alpha) = [Y_i(0) - L_i](1 - \beta)^\alpha + L_i; \qquad \alpha = 1, 2, \ldots, \alpha_m \tag{3}$$

where $0 < \beta < 1$ is a design parameter and α is the integer counter (iteration number for the performance tuning loop). Note that the specifications are gradually reduced as α increases, which depends on the step size β (the step size of the specifications increase as β does). The main reason to use Eq. (3) to produce specification $\bar{Y}_i(\alpha)$ is to make the change of specification small (from one iteration to the next) when it is close to its lower bound L_i.

With each set of design specifications $\bar{Y}_i(\alpha)$, the block OCC algorithm will produce a controller with index α, called the αth controller, using the design model (2). The closed-loop system with the design model and the αth controller is asymptotically stable because the block OCC controller is an LQG controller with a special choice of the output weighting matrix. But the closed-loop system with the evaluation model may not be stable. If the closed-loop system with respect to the evaluation model is unstable, the performance tuning loop will be terminated, according to the block OCC performance tuning loop diagram in Fig. 2, otherwise the output covariance matrices $Y_i^e(\alpha)$ and $Y_i^d(\alpha)$, with respect to the evaluation and design models, will be computed for future use.

Because the open-loop system is asymptotically stable, the closed-loop system will be asymptotically stable if the controller gain is small enough. As the control gains increase, i.e., α increases, the closed-loop system with respect to the evaluation model may become unstable. Hence, a plot similar to Fig. 3 can be generated for analysis. We use α_b to denote the point with the best performance with respect to the evaluation model. The information on the α_bth controller will be used for the new model reduction because we want the design model to be appropriate to the controller with the best performance. The new output and input weighting matrices Q and R will be computed in the following way

$$Q_i = (1 - \alpha_q)Q_i(\alpha_b) + \alpha_q\left|\bar{\sigma}[Y_i^e(\alpha^b)]\right. - \left.\bar{\sigma}[Y_i^d(\alpha_b)]\right|I_{m_i}, \quad Q = \text{block diag}\,[\ldots Q_i \ldots] \tag{4a}$$

$$R = (1 - \alpha_r)R(0) + \alpha_r \text{diag}\,([\ldots, \left|U_j^e(\alpha_b) - U_j^d(\alpha_b)\right|, \ldots]) \tag{4b}$$

where $0 \le \alpha_i \le 1$ ($i = q, r$) are design parameters. The selections of α_q and α_r depend on the percentage of the new modeling information one wants to use in this process. $R(0)$ is the controller channel input weighting matrix used in the first MCA model reduction. $U_j^e(\alpha_b)$ and $U_j^d(\alpha_b)$ ($j = 1, 2, \ldots, n_u$) are the closed-loop input variances of the α_bth controller with respect to the evaluation and design model, respectively. Similarly, $Y_i^e(\alpha_b)$ and $Y_i^d(\alpha_b)$ are the output covariances. The main reason to add these items to correct the input and output weighting matrices is to reduce the differences between the evaluation and design covariances ($Y_i^e - Y_i^d$) for the α_bth controller.

$Q_i(\alpha_b)$ is the convergent output weighting matrix for the ith block during the design of the α_bth controller. The importance of $Q_i(\alpha_b)$ can be clearly observed in the output variance constraint (scalar OCC) problem[1,2] (a special case of the block OCC problem when each block has dimension equal to one). It is noted that during the scalar OCC design iteration procedure the output weighting matrix Q is adjusted so that if a particular output specification $\bar{Y}_i$ is not achieved, the corresponding Q_i will be increased according to the discrepancy between the current output variance Y_i and the specification $\bar{Y}_i$. Consequently, those outputs with hard-to-achieve specifications (indicated by $Y_i = \bar{Y}_i$) will end up with large Q_i, and those with easy-to-achieve specifications ($Y_i < \bar{Y}_i$) will have the small Q_i. In fact, for those outputs that end up with variances smaller than the corresponding $\bar{Y}_i$ the final convergent Q_i will be zero. This implies that these output constraints are not important and can be disregarded during design. However, at the beginning, this information is unknown. As a result, the convergent Q appropriately reflects the importance of each output with respect to the given specification. This property is very helpful for the model reduction using MCA, because MCA calculates the contribution of each mode to a weighted output cost $E_x y^T Q y$ and deletes the least important modes accordingly. Hence, if the weighting matrix can appropriately reflect the importance of each output, then the reduced model using MCA will keep the information which is important to the required performance.

The controller evaluation part mainly evaluates the designed controllers in the performance tuning loop to see if the performance is satisfactory or not. The evaluation will provide the information to adjust these design parameters, e.g., α_q, α_r, and so on.

As a result, it is clear now that in the integration of model reduction and controller design there are two iterative loops, the Q loop and the performance tuning loop. The Q loop is used to combine the model reduction and the controller design process such that at convergence the design model corresponds to the controller with the best performance. The performance tuning loop intends to search for the controller of the best performance with respect to the evaluation model, and a given design model.

Integrated Modeling and Controller Software

An IMC software package has been developed to integrate the model reduction and controller design process presented in the last section. The IMC software makes it possible to obtain the rapid redesign capability in a workstation environment using MATLAB.

The idea of the integrated procedure of model reduction and controller design was first applied to design controllers for NASA's minimast at Langley Research Center.[10,15] The realization of this integrated idea needs a certain amount of computation, and some expertise is needed to manage the whole integrated design process. Some parameters must be chosen, and if changed, the whole process must be repeated. To reduce the repeated work during the integrated controller design process, we are motivated to put all the independent software modules, e.g., MCA model reduction, scalar OCC, block OCC, and EOL_∞ controller design softwares together to form a software package IMC. If some information of the physical system, (like pulse responses), or a mathematical model is available, the software will go through the whole integrated process automatically such that a person who has no knowledge of MATLAB can design controllers using this software. This software is programmed in MATLAB which is available in most workstations.

The main idea of this software is shown in Fig. 1. For a physical system, the mathematical model of the given system can be obtained by identification or by mathematical modeling. Then the software starts either with the signals which are necessary for identification or with the given mathematical model. Based on the given model or identified model, the integrated process will produce controllers for evaluation. If the requirements of the evaluation are satisfied, the controllers can be implemented for testing.

For this experiment, we used the finite element model plus sensor and actuator dynamics as our evaluation model. The IMC controller design process is shown in Fig. 2. The IMC software has seven modules for forming continuous/discrete evaluation models using the given finite element model, constructing a design model by MCA model reduction, constructing a discrete evaluation model by identification (not available), scalar OCC control design with performance tuning, block OCC control design with performance tuning, EOL_∞ control design with performance tuning, and an evaluation tool.

To design a controller from the finite element model, one can use modules 1 and 2 to form the discrete state space evaluation and design models. Choosing a control design module, (for example, the block OCC control design with performance tuning), one can iterate on the modules 2 and 4 to carry on the Q loop. After the Q loop has converged, one can evaluate designed controllers using module 7. Now let us introduce each module in detail.

Using frequencies and mode shape vectors obtained from the finite element analysis, the first module combines the finite element model with sensor and actuator dynamics to form a continuous time state space model. By choosing a proper sampling rate, the discrete evaluation model can be obtained by discretizing the continuous time model. In a case where the order of the finite element model is relatively high, an additional (optional) MCA model reduction can be applied to obtain a lower order evaluation model.

The MCA model reduction module includes two kinds of MCA model reduction routines, continuous and discrete versions. The discrete reduced-order model can be obtained from the discretized high-order model by both continuous and discrete MCA model reductions, because both MCA results provide the contribution of each mode to the total cost, which can be used to decide which mode should remain in the design model. A modal cost analysis table will also be generated.

Using the pulse responses or white noise responses, the identification module (not yet available) will produce an identified evaluation model by the Q Markov cover method.[11-13]

The design modules for the scalar OCC, block OCC, and EOL_∞ controllers are similar. Here we only discuss the block OCC control design module. The block diagram of the block OCC control design is shown in Fig. 2. The main philosophy of the performance tuning is to obtain a sequence of the controllers from low- to high-control effort. As a result, the controller of the best performance can be obtained among those controllers.

The evaluation tool contains seven blocks for plotting pole locations, simulating pulse responses for discrete systems, plotting output covariances with respect to α, simulating the responses of arbitrary input functions, simulating impulse responses for continuous systems, transferring MATLAB data file to ASCII code data files, and plotting FORTRAN simulation responses.

System Description and State Space Model

The Jet Provision Laboratory (JPL) Large Spacecraft Control Laboratory (LSCL) experiment facility[14] is shown in Fig. 4. The main component of the apparatus consists of a central hub to which 12 ribs are attached. The diameter of the dish-like structure is slightly less than about 19 ft, the large size being necessary to achieve the low frequencies desired. The ribs are coupled together by two rings of wires which are maintained under nearly constant tension. Functionally, the wires provide coupling of motion in the circumferential direction which would otherwise occur only through the hub. The ribs, being quite flexible and unable to support their own weight without excessive droop, are each supported at two locations along their free length by levitators. A levitator assembly consists of a pulley, a counterweight, and a wire attached to the counterweight which passes over the pulley and attaches to the rib. The hub is mounted to the backup structure through a gimbal arrangement so that it is free to rotate about two perpendicular axes in the horizontal plane. A flexible boom is attached to the hub and

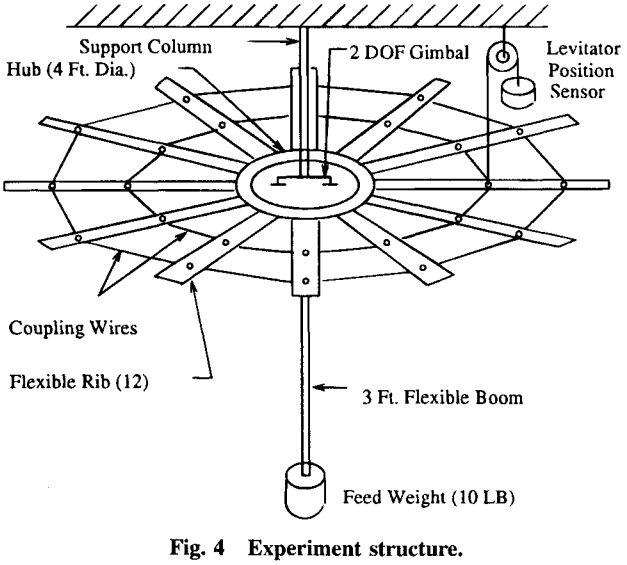


Fig. 4 Experiment structure.

hangs below it, and a weight, simulating the feed horn of an antenna, is attached at the bottom end of the boom. A 3-ft-long boom is used for this experiment.

Actuation of the structure is as follows. Each rib can be individually manipulated by a rib-root actuator mounted on that rib near the hub. A rib-root actuator reacts against a mount which is rigidly attached to the hub. In addition, two actuators are provided which torque the hub about its two gimbal axes. The hub torquers do not provide torque directly but rather are linear force actuators which produce torque by pushing or pulling at the outer circumference of the hub. The placement of these actuators guarantees good controllability of all of the flexible modes of motion. The locations of the actuators are shown in Fig. 5. Two hub actuators are used for control in x and y directions. They are denoted by HA1 and HA10, respectively. The transfer function from command torque u_i (i = 1, 2) to net torque u_i^a (i = 1, 2) is shown as follows.

$$\frac{u_i^a(s)}{u_i(s)} = \frac{3947.8}{s^2 + 44.43s + 3947.8} \tag{5}$$

Only four rib root actuators are used in this experiment. They are rib root actuators on ribs 1, 4, 7, and 10, denoted by RA1, RA4, RA7, and RA10. The transfer function from the command force u_i (i = 3, 4, 5, 6) to the net force u_i^a (i = 3, 4, 5, 6) is

$$\frac{u_i^a(s)}{u_i(s)} = \frac{24674}{s^2 + 111.1s + 24674} \tag{6}$$

The sensor locations are also shown in Fig. 5. First, each of the 24 levitators is equipped with a sensor which measures the relative angle of the levitator pulley. The levitator sensors thus provide, in an indirect manner, the measurement of the vertical position of the corresponding ribs at the points where the levitators are attached. Four position sensors measure rib displacement at the rib-root actuator locations. Sensing for the hub consists of two rotation sensors which are mounted directly at the gimbal bearing. There are a total of 24 levitator sensors used for measurements. They are denoted by LS1–LS24. The transfer function from the physical output to the measurement is assumed to be one because the optical sensor has a pretty wide bandwidth. Two hub optical angle sensors, HS1 and HS10, are used to measure the hub angle in the x and y directions. Similarly, the transfer function is assumed to be one. Only four rib root sensors, RS1, RS4, RS7, and RS10, are available for measurements. The dynamics are omitted (the transfer function is assumed to be one). Since the hub and rib-root sensors are very noisy, a first-order filter is applied for each of those six sensors. The transfer function of the filter is

$$H(s) = \frac{502.65}{s + 502.65} \tag{7}$$

A summary of outputs and inputs are contained in Table 1.

The JPL created two finite-element models with 30 and 84 modes, respectively. The 30-mode finite-element model is used in this experiment. The first 18 modes are given in Table 2. Let the structure be described in its modal coordinate by the following equations

$$\begin{cases} \ddot{\eta}_i + 2\xi_i\omega_i\dot{\eta}_i + \omega_i^2\eta_i = b_i^T u^a \quad i = 1, 2, \ldots, 30 \\ y = \sum_{i=1}^{30} p_i\eta_i \end{cases} \tag{8}$$

where u^a is the actuator output signal and y is the displacement vector colocated with the sensor inputs. The JPL provided 30 frequencies (ω_i, i = 1, 2, . . . , 30) and 30 mode shapes (p_i, i = 1, 2, . . . , 30) obtained from a finite-element analysis.

The actuator output signal u^a is now filtered by hub actuator and rib-root actuator dynamics modeled by the following equations:

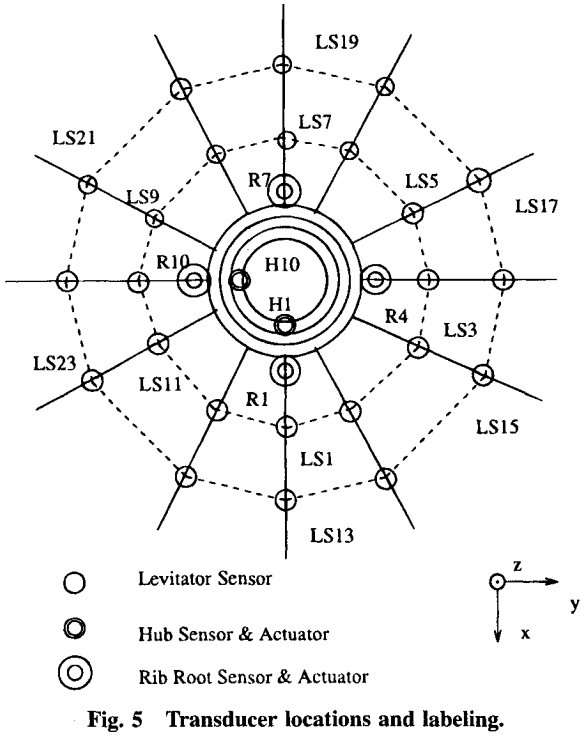


Fig. 5 Transducer locations and labeling.

Table 1 Inputs, outputs, and their limits

Inputs		Outputs	
Hub Actuator		Hub Sensor	
Notation	Limit	Notation	Limit
HA10 (u_1)	2 (N-m)	HS1 (y_{25})	69.8 (mrad)
HA1 (u_2)	2 (N-m)	HS10 (y_{26})	69.8 (mrad)
Rib-Root Actuators		Rib-Root Sensors	
Notation	Limit	Notation	Limit
RA1 (u_3)	2 (N)	RS1 (y_{27})	10 (mm)
RA4 (u_4)	2 (N)	RS4 (y_{28})	10 (mm)
RA7 (u_5)	2 (N)	RS7 (y_{29})	10 (mm)
RA10 (u_6)	2 (N)	RS10 (y_{30})	10 (mm)
		Levitator Sensors	
		Notation	Limit
		LS1-LS24 ($y_1 - y_{24}$)	114.3 (mm)

Table 2 Frequencies and damping coefficients

	Frequency, Hz		Damping Coeff.	
Mode No.	(Original)	(Modified)	(Original)	(Modified)
1	0.0902	0.0975	0.0100	0.1225
2	0.0902	0.0917	0.0100	0.2500
3	0.2089	0.2089	0.0263	0.0263
4	0.2527	0.2527	0.0100	0.0100
5	0.2527	0.2527	0.0100	0.0100
6	0.2894	0.2894	0.0100	0.0100
7	0.2894	0.2894	0.0100	0.0100
8	0.3218	0.3218	0.0100	0.0100
9	0.3218	0.3218	0.0100	0.0100
10	0.3435	0.3435	0.0100	0.0100
11	0.3435	0.3435	0.0100	0.0100
12	0.3509	0.3509	0.0100	0.0100
13	0.6150	0.6250	0.0200	0.0200
14	0.6150	0.6200	0.0300	0.0300

$$\dot{x}_a = A_a x_a + B_a u\,; \qquad u^a = C_a x_a + w_p \tag{9}$$

where u is composed of the command signals to the hub and rib-root actuators, and w_p is the actuator noise with intensity $\overline{W}_p$. The measurement output z now can be presented by

$$\dot{x}_s = A_s x_s + B_s y; \qquad z = C_s x_s + D_s y + v \tag{10}$$

where v is the sensor noise with intensity $\overline{V}$. Combining models (8, 9, 10), we can obtain a continuous-time, full-order model. Because the frequencies of all modes in our model are less than 5 Hz, we discretize the continuous model at 25 Hz, which is the computer sample rate. The discrete evaluation model is as follows.

$$\begin{cases} x_e(k+1) &= A_e x_e(k) + B_e u(k) + D_e w_p(k) \\ y_p(k) &= C_e x_e(k) \\ z(k) &= M_e x_e(k) + v(k) \end{cases} \tag{11}$$

where w_p and v are white noise with covariance matrix $W_p = \overline{W}_p/0.04$ and $V = \overline{V}/0.04$, respectively.

Output Covariance Constraint Controller Designs

The design strategy used here is the integration of model reduction and controller design introduced in the last section. Using the open-loop experimental results at the JPL, we adjusted some frequencies, damping coefficients, and input/output magnitudes such that the responses of the finite element model combining with the sensor and actuator dynamics were closer to the experimental pulse responses. The adjusted frequencies and damping coefficients of the first 18 modes are shown in Table 2. The magnitude coefficients vary in different designs.

We start the OCC controller design with the scalar output blocks. Note that in this case the constraints on the output covariance matrices reduce to those on output variances. Hence, all the constraints are scalars, and the OCC algorithm reduces to the scalar OCC results in the Purdue report.[16] After some preliminary iterations on model order, the order of the scalar OCC controller was fixed at sixteen, although the order of the model (and hence the subsequent controller) could be selected on each iterations, instead of being fixed as in our design.

The MCA results for Q loops 1 and 3 can be found in Table 3. The first eight dominant modes in Q loops 1 and 3 are the same. Hence, in this case the Q loop will not converge but will oscillate between the two models which are obtained in Q loops 1 and 2. Because the best performance with respect to the evaluation model is obtained in Q loop 2, we use the reduced-order model of Q loop 2 which keeps modes 2, 1, 14, 13, 27, 28, 4, and 6 as the final design model. The iteration on the Q loop is terminated at Q loop 3.

To show the benefit of iterating the design model, we compute the differences between the output variances of Q loop 0 and those of Q loop 2 for each output. Note that for each Q loop the OCC performance tuning loop produces a number of controllers from low- to high-control effort. The controller obtained on the αth iteration of performance tuning loop is called the αth controller. Let $Y_2(i, \alpha)$ and $Y_0(i, \alpha)$ denote the ith output variance obtained by evaluating the αth controller of Q loops 2 and 0 with the evaluation model, respectively. Plot $[Y_2(i, 9) - Y_0(i, 9)]/Y_0(i, 9)$ can be found in Fig. 6. Because the plot for controller 9 (α_b) is negative for almost all outputs (by as much as 8%), the Q loop improves the model reduction and controller design process, i.e., a better controller with respect to the evaluation model can be obtained by integration of model reduction and controller design.

From the experience of the scalar OCC controller design we feel that it is not necessary to use all outputs for controller design because of the symmetrical property of the structure. Hence, we choose to reduce the output number for the model reduction and control design process but still use all 30 measurements for the control design purpose. Outputs used for the block OCC design are

$$y_p = [y_1\ y_4\ y_{13}\ y_{16}\ y_{25}\ y_{26}\ y_{27}\ y_{28}]^T \tag{12}$$

We group outputs in the following way

$$\hat{y}_1 \triangleq \begin{bmatrix} y_1 \\ y_4 \end{bmatrix}; \quad \hat{y}_2 \triangleq \begin{bmatrix} y_{13} \\ y_{16} \end{bmatrix}; \quad \hat{y}_3 \triangleq \begin{bmatrix} y_{25} \\ y_{26} \end{bmatrix}; \quad \hat{y}_4 \triangleq \begin{bmatrix} y_{27} \\ y_{28} \end{bmatrix} \tag{13}$$

Hence, in this case constraints of the block OCC problem are 2×2 matrices for all output groups. Physical interpretation of this design is clear. Consider the output group $\hat{y}_3$ containing hub angles in the x and y directions. Suppose that the maximal singular value of the constraint matrix is σ_3. Then the design will guarantee that the hub angle at any direction of $x - y$ plane will be less than or equal to the square root of σ_3 times the input l_2 norm.

The evaluation model used in this design is obtained by combining the finite element model, the sensor, and the actuator dynamics in (8, 9, 10). The evaluation model is discretized at a sampling frequency of 25 Hz. The state space realization of this model is in the form (11), where A_e, B_e, D_e, C_e, and M_e are the system matrices, respectively, of dimension 78×78, 78×6, 78×6, 8×78, and 30×78, and u, y_p, and z are input, output, and measurement vectors, respectively, as described in Table 1. Vector w_p is the system noise from the hub and rib-root actuators with the following covariance,

Table 3 Modal cost analysis of the scalar output covariance constraint design for Q loops 1 and 3

Index	Q loop 1 $\alpha_b = 9$		Q loop 3 $\alpha_b = 9$	
	Modal Cost	Mode No.	Modal Cost	Model No.
1	6.6723e + 1	2	6.7071e + 1	2
2	2.5040e + 1	1	2.3619e + 1	1
3	4.2676e + 0	14	4.8554e + 0	14
4	2.2173e + 0	13	2.6104e + 0	13
5	4.7986e − 1	4	4.7182e − 1	4
6	2.0138e − 1	8	1.9765e − 1	8
7	1.9129e − 1	3	1.8981e − 1	7
8	1.7050e − 1	7	1.7689e − 1	3
9	1.4221e − 1	6	9.9888e − 2	6
10	8.7302e − 2	11	7.9285e − 2	16
11	7.9527e − 2	12	7.8692e − 2	11
12	7.8317e − 2	10	7.5395e − 2	10
13	5.2735e − 2	16	7.3416e − 2	20
14	4.8844e − 2	20	7.3290e − 2	12

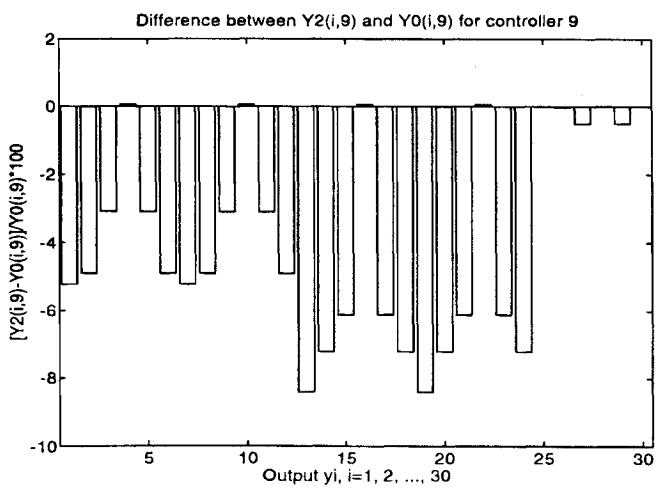


Fig. 6 Output variance differences of Q loops 0 and 2.

$$W_p = \text{block diag } [0.04I_2, 0.36I_4 + 0.36 \times 1_{4\times 4}] \quad (14)$$

where the suffix of matrix I indicates the dimension of the identity matrix, and $1_{4\times 4}$ denotes a 4 by 4 matrix with every entry equal to one. Vector v is the measurement noise of the levitator, hub, and rib-root sensors with the following covariance,

$$V = \text{block diag } [1.5625I_{22}, 3.0500I_2, 0.2500I_4] \quad (15)$$

All of the variances are taken from signal-to-noise ratios.

We choose to design a 20th-order controller for the block OCC design as opposed to a 16th-order controller for the scalar OCC design. The design parameters used in this design for the Q loop are

$$\beta = 0.2; \quad \alpha_q = 0.5; \quad \alpha_r = 0.5; \quad \alpha_m = 15 \quad (16)$$

The specification matrices $\bar{Y}_i(\alpha)$ $(i = 1, 2, \ldots, \alpha_m)$ are generated by Eq. (3).

The MCA model reduction results of the block OCC design are quite similar to those of the scalar OCC case. The Q loop does not converge but oscillates between the two design models. We plot the closed-loop output maximal singular value curves with respect to the summation of input variances, where the maximal singular values are computed with respect to the design and evaluation models. The plot for Q loop 2 is shown in Fig. 7. The solid curve with "o" is the performance of the controllers obtained from the block OCC algorithm evaluated with the design model. The dashed line with "*" evaluates these controllers with the evaluation model. It is observed that the best performance of output group 1, which is difficult to achieve with the given design, is provided by the 12th controller designed in Q loop 2. Those controllers designed in Q loop 2 were tested in the lab. In the performance tuning loop, 15 controllers are designed. All controllers stabilize the evaluation model. The 12th controller provides the best performance for output group 2.

Block Output Covariance Constraint Controller Experiments

The controllers 1, 3, 5, 7, 9, 11, and 13 obtained in the performance tuning loop of Q loop 2 were tested on the JPL LSCL Experiment Facility. It is expected that the responses of controller 1 will be close to the open-loop ones due to low control effort. The sequence of controllers from low gain to high gain allows one to perform lab tests easily with little risk of damaging the system. Starting with a low-control effort controller, we can test controllers one by one with increased control effort, and stop the test when some controller destabilizes the system, or when the oscillations become unacceptable. Because the control effort is increased gradually, the test facility will not be damaged. This is a nice feature of the integrated controller design strategy.

Because the system is highly damped, a pulse input with the width equal to a sample period (0.04 s) does not excite the system much. Hence, it is difficult to compute all the output covariances by experimental data. We performed the pulse experiments for each controller obtained in Q loop 2 with pulse input on HA1 and HA10, respectively, where the magnitude of the pulse is 2 N-m, and the width is 4 s (100 sample periods). We computed the input and output l_2 norms in the following way:

$$\|u(\cdot)\|_2^2 \triangleq \Delta^2 \sum_{k=101}^{p} u^T(k)u(k) \quad (17)$$

$$\|\hat{y}_i(\cdot)\|_2^2 \triangleq \Delta^2 \sum_{k=101}^{p} \hat{y}_i^T(k)\hat{y}_i(k) \quad ; \quad i = 1, 2, \ldots, 4 \quad (18)$$

Using Eqs. (17) and (18), Fig. 8 presents plots of input/output l_2 norms for output groups 1, 2, 3, and 4, where the dotted line with "+" is associated with experimental data, the dashed line with "*" is obtained from simulated data with the evaluation model, and the solid line with "o" is also from simulated data but with the design model. Note that we did not test every controller designed in the performance tuning loop of Q loop 2. Hence, the "+" signs in Fig. 8 are the l_2 norms of the open-

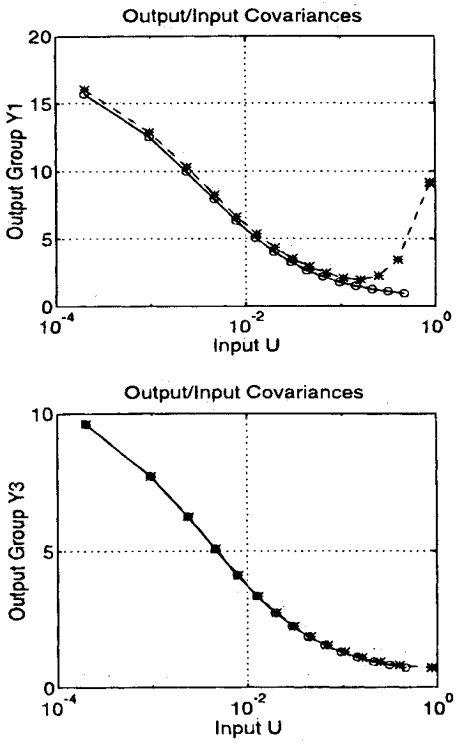


Fig. 7 Input/output covariance curves for the block OCC design.

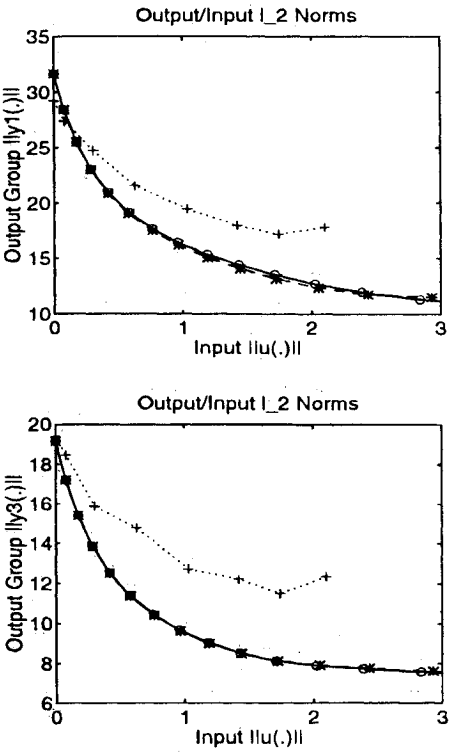


Fig. 8 Input/output l_2 norm curves for the block OCC design.

loop responses and closed-loop responses related to controllers 1, 3, 5, 7, 9, 11, and 13 from left to right. The best l_2 performances are obtained by the 11th controller of Q loop 2. We attribute the differences between the analytical l_2 responses and the test responses to the modeling error, nonlinearity, and friction.

Because the control experiment facility has no special channels to apply disturbances, the test has been done in such a way that the system is open loop at $t = 0$, when exciting signals are applied to the structure through control actuators. When the open-loop command signals vanish, the control loop will be closed to conduct the closed-loop experiment. Hence, the exciting signals applied through the actuator channels provide the initial condition for the structure.

The pulse responses of HA1 with controller 11 of Q loop 2 are shown in Fig. 9, where all input pulses are of the magnitude of 2 N-m and a period of 4 s. Hence, the closed-loop control started at the 4th second, and open- and closed-loop responses are the same for the first 4 s. It is obvious that the first two modes with a frequency of 0.0902 Hz are excited from those responses and it is clear that the controller improves the performance of the system.

Conclusions

This paper has developed and demonstrated the efficacy of an integrated approach to modeling and control design. The thesis of this paper is that the modeling and the control problem are not independent, and that a more robust design with better performance capabilities can be obtained by improving the compatibility of the model and controller, rather than the traditional approach of relying completely on robust control to compensate for bad models. To get a model that is more appropriate for control design we must utilize the control design objectives in the modeling process. This is accomplished by using the Kuhn-Tucker parameters (called matrix Q) of the controller optimization as the weighting matrix on the output in the model reduction problem. This is an iterative process which is shown to converge for the examples shown. At convergence, we say that the model is now compatible with the (model-based) controller design.

The effectiveness of this reduced-order controller design methodology has been demonstrated on the JPL Large Spacecraft Control Laboratory (LSCL) experiment facility. For two different design objectives [the scalar Output Covariance Constraint (scalar OCC) and block Output Covariance Constraint (block OCC) designs], the iteration in the Q matrix improves the design (by 8% on the variances of some outputs, compared with design without model/controller iteration), demonstrating that the integration of model reduction and controller design improves the closed-loop performance capabilities. While it is true that robust control techniques could stabilize our system without model/controller iterations, such designs must sacrifice performance for this robustness. Our iterative method allows improved models on which to base both nominal and robust designs. The demonstration of improved robustness by our techniques is an interesting topic for future work.

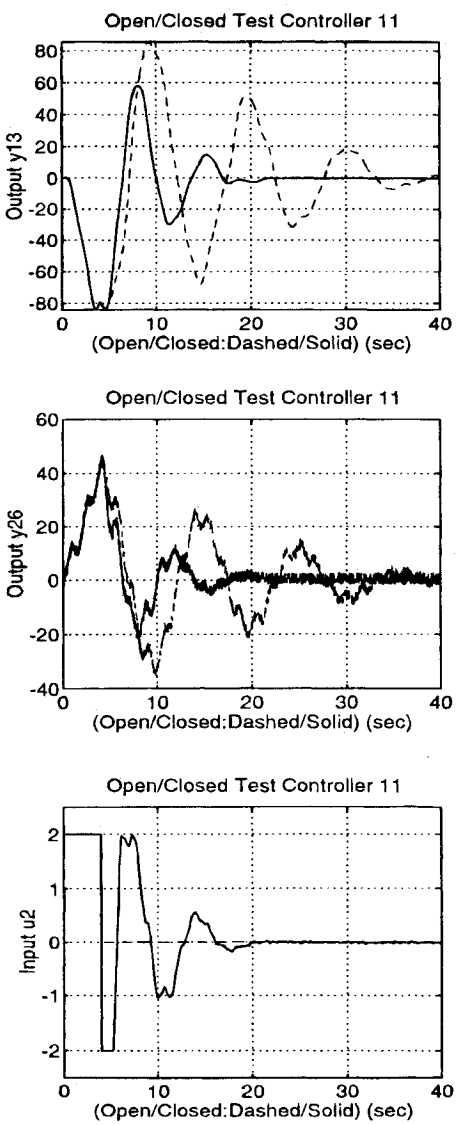


Fig. 9 *Y* **direction pulse responses of the block OCC design.**

Acknowledgments

This work was supported under JPL contract number 958934, monitored by Don Wang of the JPL. We would particularly like to thank Asif Ahmed of the JPL, whose assistance and expertise in the lab were indispensable.

References

[1] Zhu, G., and Skelton, R. E., "Mixed $\mathcal{L}_2$ and $\mathcal{L}_\infty$ Problems by Weight Selection in Quadratic Optimal Control," *International Journal of Control*, Vol. 53, No. 5, 1991, pp. 1161–1176.

[2] Hsieh, C., Skelton, R. E., and Daram, F. M., "Minimum Energy Controllers Satisfying Inequality Output Variance Constraints," *Optimal Control Applications & Methods*, Vol. 10, No. 4, 1989, pp. 347–366.

[3] Zhu, G., and Skelton, R. E., "Controller Design to Achieve Covariance Constraints," *Proceedings of the IFAC Symposium on Design Methods of Control System* (Zurich, Switzerland), Vol. 1, 1991, pp. 252–257.

[4] Zhu, G., "$\mathcal{L}_2$ and $\mathcal{L}_\infty$ Multiobjective Control for Linear Systems," Ph.D., School of Aeronautics and Astronautics. Dissertation, Purdue Univ. West Lafayette, IN, May 1992.

[5] Zhu, G., and Skelton, R. E., "A Two-Riccati Feasible Algorithm for Guaranteeing Output " L_∞ Constraints," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 114, No. 3, 1992, pp. 329–338.

[6] Skelton, R. E., *Dynamics System Control*, Wiley, New York, 1988, Ch. 10.

[7] Skelton, R. E., Singh R., and Ramakrishnan, J., "Component Model Reduction by Component Cost Analysis," *Proceedings of the AIAA Guidance and Control Conference* (Minneapolis, MN), Vol. 1, AIAA, Washington, DC 1988, pp. 264–274 (AIAA Paper 88-4086).

[8] Kim, J. H., and Skelton, R. E., "Model Reduction by Weighted Component Cost Analysis," *Proceedings of the AIAA Dynamics Specialists Conference* (Long Beach, CA), Vol. 1, AIAA Washington, DC, 1990, pp. 153–160 (AIAA Paper 90-1207).

[9] Skelton, R. E., Hughes, P. C., and Hablani, H. B., "Order Reduction for Models of Space Structure Using Modal Cost Analysis," *Journal of Guidance, Control, and Dynamics*, Vol. 5, No. 4, 1982, pp. 351–357.

[10] Hsieh, C., Kim, J. H., Zhu, G., Liu, K., and Skelton, R. E., "An Iterative Algorithm Combining Model Reduction and Control Design," *Proceedings of the American Control Conference* (San Diego, CA), 1990, pp. 2120–2125.

[11] Skelton, R. E., and Anderson, B. D. O., "Q-Markov Covariance

Equivalent Realization," *International Journal of Control*, Vol. 44, No. 5, 1986, pp. 1477–1490.

[12] Wagie, D. A., and Skelton, R. E., "A Projection Approach to Covariance Equivalent Realizations of Discrete Systems," *IEEE Transactions on Automatic Control*, Vol. 31, No. 12, 1986, pp. 1114–1120.

[13] Skelton, R. E., and Collins, E. G., "Set of Q-Markov Covariance Equivalent Model of Discrete Systems," *International Journal of Control*, Vol. 46, No. 1, 1987, pp. 1–12.

[14] Vivian, H. C., Blaire, P. E., Eldred, D. B., Fleischer, G. E., Ih, C.-H. C., Nerheim, N. M., Scheid R. E., and Wen, J. T., "Flexible Structure Control Laboratory Development Technology Demonstration," Jet Propulsion Laboratory, California Inst. of Technology, October 1987.

[15] Hsieh, C., Kim, J. H., and Skelton, R. E., "NASA GI Project—Purdue Annual Report," NASA GI Project Annual Rep. Meeting, Hampton, VA, Jan. 1990.

[16] Zhu, G., and Skelton, R. E., "Integration of Model Reduction and Controller Design for Large Flexible Space Structures—An Experiment on the JPL LSCL Facility," Final Rep., Purdue Univ., West Lafayette, IN, March 1992.