Artificial Intelligence-Based Model-Adaptive Approach to Flexible Structure Control

S. Hanagud,* B. J. Glass,† and A. J. Calise*
Georgia Institute of Technology, Atlanta, Georgia

A method of explicit self-tuning adaptive control for discontinuously time-varying flexible structures has been developed, by using artificial intelligence techniques of heuristic search and object-oriented programming to find new models when the controlled structure undergoes sudden model changes. The frequency-domain response of the structure is used to generate a heuristic model-evaluation index based on response error for possible models. The optimal model, returned by a best-first search algorithm, is then reformulated in a state form, and a new controller is obtained by using an optimal output feedback approach. Convergence of this adaptive procedure has been analyzed, and this controller model replacement (CMR) approach has been demonstrated for an example problem of a beam undergoing boundary condition changes. A performance comparison has been made with the classical explicit self-tuning regulator method of adaptive control, demonstrating the enhanced adaptiveness made possible in CMR by identifying changes in the form of the model as well as parameter shifts.

I. Introduction

A NEED for adaptive controls exists when it becomes necessary to control a system for which the form of the model and critical parameter values are uncertain, or when the system operates under a wide range of conditions. Historically, systems subjected to adaptive control have been assumed to be time-invariant or subject to incremental changes in parameter values over time. However, certain flexible structures, such as large space structures or helicopter structures, may undergo unpredictable discontinuous changes in both the model form and the model parameters due to failures, construction, added payloads, or configuration changes. Large changes in the appropriate model form tend to invalidate the a priori model on which the current adaptive-control procedures, such as self-tuning regulators or model-reference control, are based. Such changes might include new boundary conditions, system order, or large discontinuous swings in parameter values.

Controller model replacement (CMR) is related to explicit self-tuning methods, for which identification and control are separate steps, e.g., Astrom and Peterka.[1] Most current self-tuners are of the implicit type, which combines these steps. In comparison with model reference control, CMR does not depend on a fixed a priori model form μ, as shown in Fig. 1. Large changes in model form will not invalidate CMR unless they are of a type not reflected in the space of possible models.

In order to adaptively control time-varying flexible structures (TVFS), the time-varying form of the mathematical model must be identified as well as its parameters. In contrast, classical structural dynamic system-identification methods[2-5] are based on a time-invariant model form, obtained from initial assumptions, with an a priori set of initial parameters. These methods consider either parameter or frequency-domain response in identifying model parameters. Frequency-domain methods are usually based on minimizing variance, output error, or equation error.

An identification approach to the control of TVFS is to consider a set of possible a priori models and to choose the model that best matches the response. Related to this approach are parameter-set identification methods, such as the works by Stein and Saridis[6] or Deshpande et al.[7] Wittenmark[8] extended these efforts to add a second level of identification, creating a two-level self-tuning regulator. This two-level adaptive controller incorporated a procedure for probabilistic parameter-set switching on top of a conventional implicit self-tuner. More recently, Schumann et al.[9] matched model order and controller deadtime, given a priori parameters, within a range of alternative values by using stored empirical correlation functions. Linear polynomial equations have been used to represent all of these control models, and model changes have referred only to switching between fixed parameter sets. In addition, Schumann et al. have considered only time-invariant systems. Within their space of alternatives, these procedures conducted global, or blind, searches requiring evaluation of every possible alternative. None of the above works has considered control of flexible bodies.

In this paper, observed system characteristics obtained by modal analysis will be used to identify the best model by using artificial intelligence (AI) methods of best first, heuristic search, the concept of levels of abstraction, and frames or object-oriented programming techniques for knowledge representation. An object-oriented approach has been used to represent the model space as a search tree, with models organized into families at varying levels of abstraction.[10] These methods, in addition to finding a new model form, have resulted in considerable search economy over conventional programming methods. Once a model form has been identified, parameter identification has been performed to improve the initial a priori parameter set used for calculating model responses.

Although an adaptive procedure incorporating both model and parameter identification is not dependent, in general, on the use of a particular control-design method, an example has been given in this paper by using a current linear optimal output feedback control-design method of Moerder and Calise[11] when a discontinuous change to the plant results in the

Presented as Paper 87-2457 at the AIAA Guidance, Navigation, and Control Conference, Monterey, CA, Aug. 17–19, 1987; received Aug. 10, 1987; revision received July 25, 1988.

*Professor, School of Aerospace Engineering. Member AIAA.

†Graduate Research Assistant, School of Aerospace Engineering; currently, Research Scientist, Advanced Missions Technology Branch, NASA Ames Research Center, Moffett Field, CA. Member AIAA.

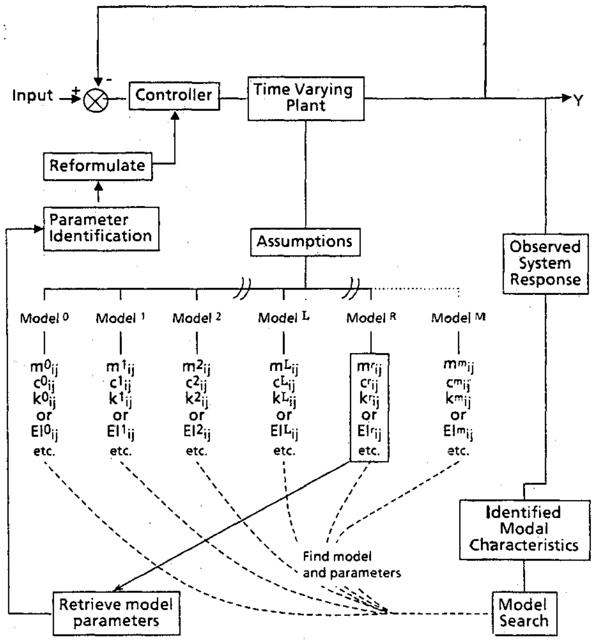

Fig. 1 A controller model replacement (CMR) self-tuning regulator.

identification of a new model form. Results show that this controller model-replacement method improves damping, time needed to reduce vibrations below a certain level, and stability for a beam subjected to discontinuous model form changes, when compared with an explicit self-tuning approach to the same problem that uses only parameter identification.

To introduce the enhanced model-adaptive approach, model- and parameter-identification procedures are discussed in Sec. II along with the relative merits of the AI methods used. In Sec. III, the formulation of the optimal control problem from the identified structural dynamic model is discussed, along with monitoring and decision-making procedures. A convergence analysis is given for the adaptive algorithm is Sec. IV, followed by an example in Sec. V of a possible heuristic model-evaluation function. A numerical example, considering a space of Euler-Bernoulli beam models and using optimal output feedback, is discussed in Sec. VI. Finally, conclusions and recommendations are given in Sec. VII.

II. AI-Based Model-Identification Procedure

Model and Parameter Forms

For TVFS, a model form μ is defined in terms of differential operators, boundary and initial conditions. In general, for structural dynamics problems and in conventional parameter-identification problems, the form of the model is assumed to be known. Given an a priori set of model parameters θ_i, the model form may be used to solve for the system response (in structural dynamics problems) or used to improve the parameter set according to some error criteria (in parameter identification). For TVFS, discrete degree-of-freedom (DOF) models have parameter sets θ_i that include the mass, damping, and stiffness matrices, M, C, and K, respectively. In this context, identification means selecting a model form μ and parameter set values θ_i so as to minimize an error function $J(\mu,\theta)$. Formulation of an error function requires that response measurements are available from the plant. For the control of TVFS, a convenient form of system response is to use the open-loop eigenvalues and eigenvectors, i.e., the natural frequencies ω and mode shapes Φ of the flexible structure.

The form of the model search space is a discontinuous, denumerably infinite set of all possible TVFS models. Since it is impractical to use J to evaluate the members of an infinite set, the model search space is constrained to a finite, a priori subset of the most likely models, R. R is chosen such that individual model responses $H(\mu,\theta_i)$ for an a priori set of parameters θ_i are distinct. Given a finite model space R, however, in practice the possibility exists of the plant changing to a model form not found in R. This possibility, plus the possible adverse effect of noisy measurements, requires that the best model, in terms of the error criteria J, be found even if no exact match is found within R.

Obtaining System Response

For a body with p discrete sensors and m actuators, the Fourier transforms of their signals can be computed to get the frequency-domain output response and input force, $w_i(\omega)$ and $F_j(\omega)$, respectively, for input/output locations x_i and x_j. A $p \times m$ transfer matrix H can be defined in terms of the ratio of the cross correlation of the response to the autocorrelation of the input, or

$$H_{ij} = \frac{w_i(\omega)F_j^*(\omega)}{F_j(\omega)F_j^*(\omega)} \tag{1}$$

where the asterisk designates a complex conjugate.

A selected column of H, then, represents the p responses due to a given actuator input. Various techniques, such as circle-fitting, exist for estimating eigenfunctions from these, and eigenvalues and system damping values can be measured directly from resonance peaks. Because of reciprocity, either rows or columns of H may be used in practice.

Determining the Optimal Model

An optimization problem requires the definition of a measure of optimality or objective function. For model identification, the optimality criterion is the model error function $J(\mu,\theta_i)$. In conventional parameter-identification problems, the model from μ is assumed, so the error function is a function of parameters only. As parameters are continuous variables, in general it is possible to set $\partial J/\partial\theta_i = 0$, and thereby to identify the parameters by solving for θ_i.

However, the models in the model space R are distinct and discontinuous, so $\partial J/\partial\mu$ cannot be defined. Since conventional identification methods cannot be used in a model space, it is necessary to pursue another approach. In this paper, the AI method of heuristic, best-first search[12,13] is used to select the model in R with the lowest associated values of J, without necessarily checking all models in R.

In comparison, in other approaches[8] the error function J is calculated successively for every model in R, selecting a model that sufficiently minimizes J. If no suitable model match is found, the entire linear search space is traversed. The advantages of such a global search are that the search methods require no knowledge of relationships and similarities between model forms, such as a shared boundary condition. For nontrivial model spaces, however, the exhaustiveness of a global search makes it an impractical method. For heuristic search methods, the objective function J is used to direct search within an organized model space, called a search tree. Abstract data forms of this type have been used in artificial intelligence research for several decades.

Organizing the Model Space

To construct a search tree, the distinguishing characteristics v_j of models are considered. If both simple and detailed models are grouped taxonomically by their similarities (common attributes such as boundary conditions) and indexed by their differences (e.g., presence or absence of added point masses), a hierarchical model network results. Networks of this sort are commonly known as discrimination nets, as defined by Feigenbaum.[14] They are easily represented as tree-like directed graphs, such as is shown in Fig. 2.

In order to organize simple and detailed models into one search tree, the level of abstraction of a particular family of

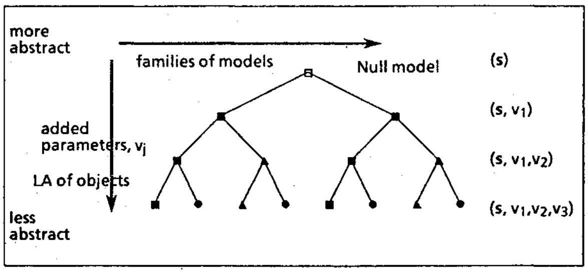

Fig. 2 Directed graph of R with $N = 8$ (a regular search tree).

models is defined as inversely proportional to the number of common attributes v_j specified for its member models:

$$LA \propto [\mathrm{tr}\ (vI)]^{-1} \tag{2}$$

In other words, detailed models fall on a lower level of abstraction than simple, prototypical models. The level of abstraction of a particular model μ_j is defined as the total number of distinguishing characteristics less the number defined for that model, or

$$LA = C(\bar{v}) - C(v^j) \tag{3}$$

where $\bar{v}$ is the total set of distinguishing characteristics used in search-tree construction, v^j is the set of characteristics defined for model μ_j, and $C(\,)$ designates a cardinal number.

In this paper, the search tree departs from usual AI practice in that all models, once introduced, are carried down through the search tree. This is necessary for identification, since an uncomplicated model at a higher level of abstraction may be a better match to the plant than a more detailed permutation of the same model at a lower level. This propagation of higher-level models down to lower levels is accomplished by including each model in the set of its children (if any). In Fig. 2, the geometric shapes at nodes represent models in a search tree of this type. Defining the level of search in the tree k and the branching factor n, it can easily be shown that for a regular tree such as shown in Fig. 2, $n^k = N$, where N is the total number of models in R and the search tree exhibits model propagation. The order in which the v_j are considered defines the levels of the tree. Although this order is not specified by n^k, it can be shown[15] that if the order of v_j is such that the difference between model responses decreases as k increases, then the optimal model can be selected by best-first search without backtracking. If this criterion for a "well-ordered" tree is not met, lower-level models might be closer in response error J to other higher-level models than to their own ancestors in the tree.

Advantages of Organizing the Model Space

By organizing the model space R into a search tree, two important benefits are realized. First, a search tree permits best-first or other heuristic search algorithms to be used. Defining C as the total number of models searched, the use of an exhaustive search method will cause all N models to be checked, or,

$$C = n^k = N \tag{4}$$

In contrast, if best-first search is used in a regular well-ordered search tree, it can be shown[15] that

$$C = n + \sum_{i=2}^{k} (n-1), \qquad C \propto O(kn) \tag{5}$$

For this example, organizing the model space results in reducing the number of model evaluations from $O(n^k)$ to $O(kn)$. A similar result can be obtained for irregular search trees, i.e., trees with a variable branching factor (n_{km}).

Another advantage of organizing the model space is the use of inheritance of variables, allowing low-level models to refer to values stored in a parent model, rather than redundantly replicating identical variable values. By using inheritance for both common variables (analogous to a FORTRAN common block) and also for the v_j values common only to specific model families, memory needs are reduced. For instance, without inheritance, a value is stored for each variable in each model:

$$\zeta \propto Nk + Np \tag{6}$$

where k is the number of distinguishing characteristics v, and p is the number of parameters common to all models in R. In contrast, given full inheritance in a search tree, memory needs can be shown to be reduced:

$$\zeta'' \propto N + p \tag{7}$$

for regular search trees. Equation (7) follows from storing the p common values only once, and from storing each new value of v_j only in the highest model where it is defined. No more than one previously undefined distinguishing characteristic v_j is assumed to be specified for any of the N models.

Object-Oriented Approach Selected

Although levels of abstraction and inheritance are not synonymous, the advantages of both for model identification are such that both should be features of the programming approach chosen. In a conventional programming approach such as FORTRAN or Pascal, the search-tree relationships would have to be implemented in the search program's control structure. A separate indexing scheme would be needed to implement data inheritance, thus offsetting the memory savings due to inheritance with increased programming complexity. In contrast to conventional programming approaches, in a frame system[16] the data and procedures relevant to a given model are stored as a localized abstract entity, called a frame. Pointers within levels of frames allow hierarchies to be easily implemented within the knowledge base, rather than in the search program. An object-oriented system is a type of frame system in which program execution proceeds by means of messages between objects (frames). For the case of TVFS models implemented as objects, a search program would send a querying message to an object embodying a given model, requesting the simulated dynamic response for that model. Inheritance in an object-oriented system functions by automatic querying of an object's parents in the tree if the desired variable is not stored locally. The layered structure of an object-oriented system, together with its inherent inheritance characteristics, is found to be superior to a conventional procedural programming method for implementing levels of abstraction and inheritance of values.

Variation of Best-First Search

Given an organized model space R and an evaluation criterion J for model responses, heuristic search methods may be used. One such method, called best-first search, sorts each level of R by J and chooses the best model as either a match or a direction for further search. Best-first search begins by forming a queue of the first-level models in the tree. After sorting, if no model satisfies an error bound for J, the children of the lowest-valued model are substituted for it in the queue. The process then repeats as needed until a match is found or until the bottom level of the search tree is reached. A particular best-first algorithm called A^* has been shown[17] to return with the optimum model in the search space, provided that the heuristic evaluation function provides an optimistic estimate of the distance to the goal state (e.g., satisfying the bound on the response error J).

As implemented in this work for model identification, best-first search deviates from the conventional AI method because of the organization of the model search tree. Since each model includes itself in the set of its children, no model is ever deleted from the queue once introduced. The optimality of this version of best-first search can be shown[15,18] to depend on the assumption of a well-ordered search tree, for which the differences in J between models diminish with increasing search depth. Specifically, it can be shown that $\max|\Delta h(k+l, \bar{\theta})| < \min|\Delta h(k, \bar{\theta})|$ is a sufficient condition for selection of the model, where $\Delta h(k)$ is defined as the difference in response between two models on the kth level of the search tree, given the true plant parameters.

Parameter Identification

Since the model-identification process requires an initial a priori set of parameters θ_0 in order to generate model responses, a parameter-identification step is necessary to correct any initial parameter errors. If these parameter errors $\Delta\theta$ are significant, then iterations of model and parameter identification will be necessary to identify both the true model and parameter set. The particular parameter-identification algorithm used in this paper, called MCKID, has been shown by Hanagud et al.[19] to find accurate, symmetric parameter matrices for flexible structures, given either proportional or nonproportional damping.

III. Controller Implementation

Monitoring and Decision Making

In the absence of outside disturbances, no vibration control is expected to be needed for a TVFS at rest. Model-form changes are expected to be heralded by disturbances (e.g., jarring, sliding, and rigid-body motions). If these force inputs cannot be accurately measured, a probing actuator signal might be needed to obtain accurate response measurements for identification. It is assumed that the previous or a default control would be used until identification and control design produced a new control.

The presence of measurement noise in any real TVFS should not trigger unnecessary cycles of identification and controller redesign. Therefore, before a new control is designed, some set tolerance of a performance criteria should be violated in order to signify that a model change is necessary. The error-evaluation criteria chosen for monitoring is the model-evaluation criteria h discussed at length later in Sec. V. This model-evaluation criterion is a function of squared-frequency error and the degree of linear dependence of model mode shapes with the actual functions.

The failure of the response of the current model to meet monitoring criteria will trigger the model-identification process. In general, if multiple criteria are used, or if special cases (very noisy input, for instance) require consideration, a set of decision-making rules should be created and implemented as a higher-level control system, e.g., Astrom and Peterka[1] or Saridis.[20]

Control Formulation and Design

For controller formulation, it is assumed that a reduced-order, discrete degree-of-freedom structural dynamic model is provided by the identification program. For TVFS, these models are described by a finite set of ordinary differential equations of the form

$$Mu_{tt}(x,t) + Vu_t(x,t) + Ku(x,t) = f(x,t) \tag{8}$$

where M, V, K are mass, damping, and stiffness matrices, respectively, of order N. Displacements and velocities are assumed to be measured by point sensors:

$$y = C^*u + D^*u_t \tag{9}$$

with C^*, D^* reflecting the number and location of P displacement and velocity sensors, respectively. The total actuator force is assumed to be provided by M point actuators:

$$F(x,t) = \sum_{i=1}^{M} \delta(x - x_i) f_i(t) \tag{10}$$

where δ is a Dirac delta function and f_i is the actuator force at $x = x_i$.

The eigenvalues and eigenvectors of Eq. (8) are found as by-product of identification, or else by modal analysis. Using the eigenvectors as a linear transformation matrix

$$u = \Phi q(t) \tag{11}$$

for which

$$\Phi^T M \Phi = I, \qquad \Phi^T K \Phi = \Lambda, \qquad \Phi^T V \Phi = 2\zeta\Lambda^{1/2} \tag{12}$$

where I is the identity matrix, Λ is the diagonal eigenvalue matrix, and ζ are damping coefficients. From Eqs. (8), (9), and (12).

$$Iq_{tt} + 2\zeta\Lambda^{1/2}q_t + \Lambda q = \Phi T_f \tag{13a}$$

$$y = C^*\Phi q + D^*\Phi q_t \tag{13b}$$

In state form, the Eqs. (13) become

$$z_t = Az + Bf \tag{14a}$$

$$y = Cz \tag{14b}$$

where

$$z = [q \,|\, q_t]^T \tag{15a}$$

$$A = \begin{bmatrix} 0 & I \\ -\Lambda & -2\zeta\Lambda^{1/2} \end{bmatrix}, \qquad B = \begin{bmatrix} 0 \\ \Phi^T \end{bmatrix} \tag{15b}$$

$$C = [C^*\Phi \,|\, D^*\Phi] \tag{15c}$$

In this manner, a structural dynamics model of discrete degrees of freedom produced by the model-identification procedure, as given in Eq. (8), is used to produce a new controller, subject to the calculation of optimal feedback gains. Since the initial structural dynamics model produced by the identification procedure may be of high order, model reduction may be necessary prior to the use of Eq. (8). It is to be noted that Λ and Φ are calculated from the identified model, whose order is independent of the number of modes used in identification. Hence, Λ and Φ are obtained from a large-order model even though only one or two observed modes might actually be used for identification and feedback.

Given that large, flexible structures are in fact distributed-parameter systems, the use of a relatively low-order discrete model may lead to spillover problems due to the residual modes.[21,22] It is not the purpose of this paper to discuss the methods proposed to alleviate this problem, so the high-order residual modes will be ignored for the sake of simplicity. It is expected that CMR could be used with another method of flexible structure control, such as IMSC,[23] if desired. Instead, a method of optimal output feedback will be used, and a deterministic system assumed, to demonstrate this approach to adaptivity.

Given p sensors and m actuators, the control law will be of the form $f = -Gy$, where G is the $m \times p$ gain matrix. The elements of G are chosen to minimize the expectation of an integral quadratic performance index of control effort and state:

$$J = E_{z_0}\left\{\int_0^\infty (z^T Q z + f^T R f)\,\mathrm{d}t\right\} \tag{16}$$

where the weighting matrices Q and R are respectively nonnegative and positive definite, and Q is chosen specifically as

$$Q = \frac{1}{2}\begin{bmatrix} \Lambda & 0 \\ 0 & I \end{bmatrix} \tag{17}$$

so that the first term of Eq. (20) is the plant energy $E = z^T Q z$, as discussed by Balas.[21] Other values of Q may also be used. The expectation is taken over an assumed distribution on the initial state z_0:

$$E\{z_0\} = 0, \qquad E\{z_0 z_0^T\} = z_0 \tag{18}$$

A convergent numerical method for calculating the optimal output feedback gain matrix is given by Moerder and Calise.[11] These gains are then calculated on demand (subject to computational limitations) for newly identified boundary conditions and parameter sets. Given that R contains a finite number of model forms, some computational time savings may be realized by precomputing these gains for common model form-parameter set combinations and storing the gains in the appropriate model object.

IV. Convergence Analysis

Since the identification loop of CMR uses plant frequencies and mode shapes as input, ideally the identification process is not coupled to the current control design. In practice, a given control design may tend to excite actuator dynamics more than others, which might add noise to the observed frequencies and mode shapes. Assuming that these control design-particular effects are small, then identification and control design may be treated as sequential processes in the adaptive loop. Demonstrating convergence of this loop requires the convergence of its component processes.

The linear optimal output feedback control-design procedure used in this paper has previously been shown by Moerder and Calise[11] to converge to a local minimum for input models whose system dynamics can be stabilized by output feedback (see Sec. VI for an example). Hence, the convergence of the adaptive process depends on the convergence of the identification process. Since an initial set of parameters is assumed to generate model responses, an analysis of identification convergence must demonstrate both that model convergence occurs and that the correct model is identified if parameter error is bounded.

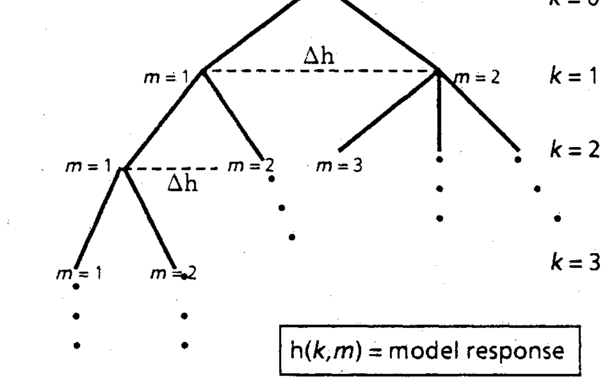

Fig. 3 A model search tree.

Definitions

Consider a search tree that represents a model space R such as shown in Fig. 3. For some set of parameters θ_i, a model in R is $\mu(k,m)$, and the analytical response of this model is $H(k,m,\theta_i)$. In this notation, k refers to the search level and m is an index on that level. A measured response of the plant H^m is assumed to be available, so a heuristic model evaluation function can be defined as

$$J(k,m,\theta_i,H^m) = \epsilon^i_{km} \tag{19}$$

The true model is defined as follows:

$$J(\bar{k},\bar{m},\bar{\theta},H^m) = \overline{\epsilon_{\bar{k}\bar{m}}} = 0 \tag{20}$$

where $(^-)$ pertains to the true model and $(\,)^m$ is a measured quantity. The true model with its parameter set $\bar{\theta}$ is assumed to be in the model space R. The quantities $\bar{k}$ and $\bar{m}$ are assumed to be the locations of the true model. A true identified model is identified with less restriction as

$$J(\bar{k},\bar{m},\bar{\theta}_i,H^m) = \epsilon^i_{\bar{k}\bar{m}} < \epsilon \tag{21a}$$

$$J(k,m,\bar{\theta}_i,H^m) = \epsilon^i_{km} > \epsilon, \qquad \text{for } k \neq \bar{k}, m \neq \bar{m} \tag{21b}$$

A bound on the true identified parameter set is defined as $|\theta - \theta_i| < \delta$. In order to start the iteration, a parameter set is guessed. It is assumed that the guess is a good guess. Then, $\theta_i = \bar{\theta} + \Delta\theta_i$. It is assumed that $|\Delta\theta_i| < \delta_b$.

Consider the separation between models on a given level k of a search tree, such as shown in Fig. 4. The difference in the heuristic evaluation function J between two models $\mu(k,m)$ and $\mu(k,m+1)$ is defined as

$$\Delta h(k,m,k,m+1) = |J(k,m,\theta_i) - J(k,m+1,\theta_i)| \tag{22}$$

Recall that the model search tree in this paper is defined such that each model includes itself in the set of its children at the next lower search level. In a tree of this type, the assumption of a well-ordered search tree, such that differences in the response Δh diminish with increasing k, can be shown[15,18] to be a sufficient condition, given accurate parameters, for selection of the true model by best-first search. In a well-ordered search tree, the largest value of J at a given level $k+1$ is no greater than the smallest J at a previous level k, i.e.,

$$\Delta h(k+1) \leq \Delta h(k), \qquad \forall k,m,\theta_i \tag{23}$$

For all models in a well-ordered search tree, it follows that the smallest model separation Δh^* is found on the bottom level, k_B:

$$\Delta h(k) \geq \Delta h^*(k_B), \qquad \forall k,m \tag{24}$$

where $(\,)^*$ denotes the lowest value at a given search level. Since each model includes itself in the set of its children, if the

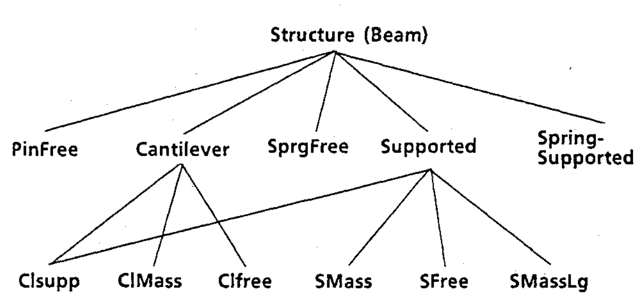

Fig. 4 A model space containing simple beam models.

model evaluation function J is chosen to satisfy Eqs. (23) and (24) for a well-ordered search tree, it is possible to show that the best value of J decreases monotonically as k increases, i.e., $J^*(k+l) \leq J^*(k)$.

Lemma 1. In a given well-ordered search tree, Δh bounds the allowable evaluation error ϵ.

Since a model's response $H(k,m,\theta_i)$ must be the closest to H^m to be selected,

$$\frac{\Delta \bar{h}}{2} \geq \epsilon \tag{25}$$

where $\Delta \bar{h}$ is the least Δh adjacent to the identified model. It is possible that the identified model might have the smallest Δh, so an error bound,

$$\frac{\Delta h^*}{2} \geq \epsilon \tag{26}$$

insures that any model satisfying ϵ is closest to the actual system, in terms of J.

Lemma 2. Upper and lower bounds for R in $\mathbf{R}$ exist, where $\mathbf{R}$ is the space of all possible models and R is the selected model space.

The heuristic evaluation function J orders R by generating a value of each member model, hence it orders $\mathbf{R}$ also. Since $(\mathbf{R},J)$ is an ordered set, and R is a non-null, finite subset of the denumerably infinite set $\mathbf{R}$, upper and lower bounds for R in $\mathbf{R}$ are definable,[24] i.e.,

$$\{\forall \mu' \epsilon R, \exists \mu_1,\mu_2 \epsilon \mathbf{R} \mid J(\mu_1) \leq J(\mu') \leq J(\mu_2)\} \tag{27}$$

Lemma 3. R is a closed set.

By the definition of R as a fixed, finite subset, $\mathbf{R} \supset R$, R must be a closed set.

Lemma 4. For given measured inputs, an optimal model μ^* exists in R.

Since the search heuristic $J(\mu,\theta)$ orders R for given parameters θ_i, by the principle of the least number,[24] there exists $\{J(\mu^*,\theta) \mid \mu^* \epsilon R\}$ such that

$$J(\mu^*,\theta) \leq J(\mu,\theta), \qquad \forall \mu(k,m) \epsilon R \tag{28}$$

Theorem 1. The heuristic search model-identification algorithm is convergent when R is represented as a well-ordered search tree.

From Lemmas 2 and 3, R is closed and bounded. Since the heuristic evaluation function h is monotonic in a well-ordered R, the sequence of values chosen by best-first search at successive levels

$$J^*(k) \geq J^*(k+1) \geq J^*(k+2) \geq \cdots \geq J^*(N_S) \tag{29}$$

is convergent, where N_S is the number of levels of abstraction in the search tree. From Lemma 4, the value upon convergence will be $J^*(\mu^*(k,m),\theta_i)$.

The model-identification process, using best-first search in well-ordered search spaces, has been shown to converge to some model given a parameter set θ_i. However, this convergence by itself says only that the best model is selected for a given θ_i, not necessarily the true identified model. For sufficiently large parameter set error $\Delta\theta$, some incorrect model $\mu(k,m)$ might appear better than the true model $\mu(\bar{k},\bar{m})$, in terms of $J(\mu,\theta_i)$.

Lemma 5. For an incorrect model to be selected,

$$\epsilon^i_{km} < \bar{\epsilon}_{km}, \qquad \text{for some } \theta_i \tag{30}$$

Suppose an incorrect model $\mu(k,m)$ were selected due to a parameter set error $\Delta\theta_i$. Then, to be selected,

$$J(k,m,\theta_i) = \epsilon^i_{km} < \epsilon \tag{31}$$

but, to be the incorrect model,

$$J(k,m,\bar{\theta}) = \bar{\epsilon}_{km} > \epsilon \tag{32}$$

Hence, for an incorrect model to be selected,

$$\epsilon^i_{km} < \bar{\epsilon}_{km} \tag{33}$$

Assumption. The search heuristic J for θ_i can be expressed as

$$J(k,m,\theta_i) = J(k,m,\bar{\theta}) + \frac{\partial J}{\partial \theta}\Delta\theta_i + \cdots \tag{34}$$

if $\Delta\theta_i$ is assumed to be sufficiently small, i.e., $|\Delta\theta_i| < \delta_b$; and, if it is possible to neglect to higher-order differences ($\Delta\theta^2$, $\Delta\theta^3$, etc.)

$$\left(\frac{\partial J}{\partial \theta}\right)_i = \lim_{\Delta\theta_i \to 0} \frac{J(k,m,\bar{\theta}+\Delta\theta_i) - J(k,m,\bar{\theta})}{\Delta\theta_i} \tag{35}$$

Since $J(\theta)$ is continuous, $\partial J/\partial\theta$ exists and is bounded:

$$\frac{\partial J}{\partial \theta}\Delta\theta_i = a_i, \qquad |a_i| < a \tag{36}$$

Theorem 2. A sufficient condition exists for model convergence to the true model, when the parameter set error is suitably bounded by δ_b.

From the previous assumption regarding J, for small δ it is possible to use Eqs. (34) and (36):

$$\epsilon^i_{km} - \bar{\epsilon}_{km} = a_i \tag{37}$$

From Eq. (32) in Lemma 5,

$$\epsilon^i_{km} \geq \epsilon + a_i \tag{38}$$

Recall also from Lemma 5 that for an incorrect model to be selected due to $\Delta\theta_i$, it is necessary that

$$\epsilon^i_{km} < \epsilon \tag{39}$$

Therefore, substituting Eq. (38) into Eq. (31),

$$\epsilon > \epsilon^i_{km} \geq \epsilon + a_i, \rightarrow a_i < 0 \tag{40}$$

If the bound on the parameter error, $\delta > |\Delta\theta_i|$, is chosen such that

$$\frac{\partial J}{\partial \theta}\Delta\theta_i = a_i > 0 \tag{41}$$

for all models, then Eq. (40) is contradicted, and Eq. (41) is sufficient to insure that an incorrect model cannot be selected.

Lemma 6. For J,θ satisfying Theorem 2, in general $J(k_r,m_s,\theta_j,H^m) > \epsilon$, where ϵ is some bound.

Recall from Eq. (20) that the true model with true parameters is defined as

$$\overline{J(k,m,\theta,H^m)} = \epsilon_{\overline{km}} = 0 \tag{42}$$

The true identified model was defined as

$$\overline{J(k,m,\theta_i,H^m)} = \epsilon^i_{\overline{km}} < \epsilon \tag{43a}$$

$$J(k,m,\bar{\theta}_i,H^m) = \epsilon^i_{km} > \epsilon, \qquad \text{for } \bar{k} \neq \bar{k}, m \neq \bar{m} \tag{43b}$$

for $\epsilon > 0$ and $k \neq \bar{k}, m \neq \bar{m}$. Bounds were defined on the parameter error: $|\bar{\theta} - \theta_i| < \delta, \theta_i = \bar{\theta} + \Delta\theta_i$. If $J(k_r, m_s, \theta_j, H^m) < \epsilon$, it is also possible to assume an expansion on θ, as shown previously in Eq. (34):

$$J(k,m,\theta_i) = J(k,m,\bar{\theta}) + \frac{\partial J}{\partial \theta}\Delta\theta_i + \cdots < \epsilon, \quad \text{H.O.T. negligible} \tag{44}$$

From the model space definition,

$$J(k_r, m_s, \bar{\theta}_i, H^m) = \epsilon + e_{rs} > \epsilon \tag{45}$$

where $e_{rs} > 0$. Then, substituting in Eq. (44),

$$\epsilon + e_{rs} + \left(\frac{\partial J}{\partial \theta}\right)_{\theta_i} (\theta_j - \bar{\theta}_i) < \epsilon \tag{46}$$

Consider two cases:

Case (a):

$$\text{If} \left(\frac{\partial J}{\partial \theta}\right)_{\theta_i} = 0, \qquad \text{then } e_{rs} < 0 \text{ is a contradiction} \tag{47a}$$

Case (b):

$$\text{If} \left(\frac{\partial J}{\partial \theta}\right)_{\theta_i} \neq 0, \quad \text{then consider } \left(\frac{\partial J}{\partial \theta}\right)_{\theta_i} (\theta_j - \bar{\theta}_i) = g_{rs} > 0 \tag{47b}$$

similar to the relation in Eq. (41) in Theorem 2. Then $e_{rs} + g_{rs} < 0$, which is also a contradiction. Hence, in general $J(k_r, m_s, \theta_j, H^m) > \epsilon$.

Lemma 7. For J, θ satisfying Theorem 2, the true model error $J = J(\bar{k}, \bar{m}, \theta_k, H^m) < \epsilon$.

Expanding again,

$$J(\bar{k},\bar{m},\theta_k,H^m) = J(k,m,\bar{\theta},H^m) + \left(\frac{\partial J}{\partial \theta}\right)_{\bar{\theta}} (\bar{\theta} - \theta_k) + \cdots (\text{H.O.T})$$

$$= \left(\frac{\partial J}{\partial \theta}\right)_{\bar{\theta}} (\bar{\theta} - \theta_k)$$

$$= \left(\frac{\partial J}{\partial \theta}\right)_{\bar{\theta}} (\bar{\theta} - \theta_k) \tag{48}$$

by the definition of the true model given in Eqs. (20) and (42). From inspection of Eq. (48), it may be observed that if

$$\left(\frac{\partial J}{\partial \theta}\right)_{\bar{\theta}} = 0, \qquad \text{then } J(\bar{k},\bar{m},\theta_k,H^m) = 0 < \epsilon \tag{49}$$

Or, if

$$\left(\frac{\partial J}{\partial \theta}\right)_{\bar{\theta}} \neq 0, \qquad \text{then } 0 \leq \left(\frac{\partial J}{\partial \theta}\right)_{\bar{\theta}} (\bar{\theta} - \theta_k) \leq \epsilon \tag{50}$$

Since $|\bar{\theta} - \theta_k|$ is bounded by δ, then Eq. (50) effectively imparts a bound on the alowable δ in terms of the convergence bound ϵ.

Theorem 3. For J, θ satisfying Theorem 2, only the true model error is less than the convergence bound ϵ.

From Lemma 6, in general for $k \neq \bar{k}$, $m \neq \bar{m}$, then $J(k,m,\theta_j,H^m) > \epsilon$. However, given $\bar{k}$, $\bar{m}$, then from Lemma 7, $J(\bar{k}, \bar{m}, \theta_k, H^m) < \epsilon$.

Corollary 1. If Lemma 7 is satisfied by a model when that model is the true model, then Eq. (41) will be satisfied in general for that model provided that δ_b is sufficiently small.

Let $\mu'(k', m')$ be an arbitrary model, with θ' as its true parameter set. When μ' is the true model,

$$\left[\frac{\partial J}{\partial \theta} |\theta_i - \theta'|\right]_{k'm'} > 0 \tag{51a}$$

or

$$\left[\frac{\partial J}{\partial \theta} \delta_b\right]_{k'm'} > 0 \tag{51b}$$

Suppose μ' is not the true model. For the true model ($\bar{k} \neq k'$, $\bar{m} \neq m'$),

$$|\theta_i - \bar{\theta}| < \delta_b \tag{52}$$

also, so for small δ_b, then θ' and $\bar{\theta}$ are close; it can therefore be said that

$$\left[\frac{\partial J}{\partial \theta} \delta_b\right]_{k'm'} > 0 \tag{53}$$

holds, despite $\theta' \neq \bar{\theta}$, if Eqs. (51) are true.

Corollary 2. δ which satisfies Theorem 2 is finite.

If δ were not finite, then $|\Delta\theta_i| \Rightarrow \infty$, and for the true model

$$\left[\frac{\partial J}{\partial \theta} \Delta\theta_i\right] \rightarrow \infty, > \epsilon \tag{54}$$

which violates Lemma 7. Therefore, δ which satisfies Theorem 2 is finite. Specific values of δ which satisfy Theorem 2 will depend on the chosen form of $J(k,m,\bar{\theta} + \Delta\theta_i)$.

Assumption. Parameter identification is assumed to produce a new parameter set estimate θ_{i+1}, when furnished the true model and previous parameter set $\bar{\mu}, \theta_i$, such that

$$\delta > |\Delta\theta_i| > |\Delta\theta_{i+1}| > \cdots > |\Delta\theta_N| \tag{55}$$

MCKID[19] is one such parameter-identification method.

Corollary 3. Cycles of model and parameter identification will converge to the true model and parameter set, for initial parameter error and J chosen to satisfy Theorem 2.

This follows directly from Eq. (55) and Theorem 2. If the initial parameter set error $\Delta\theta_0$ satisfies Lemma 7, the true model will be returned. Given the true model and initial parameter set, from Eq. (55), parameter identification will produce a θ_i such that $\epsilon > |\Delta\theta_0| > |\Delta\theta_1|$. When fed back to model identification, $\Delta\theta_1$ then also satisfies Theorem 2. Further iterations of model and parameter identification, from Eq. (55), must produce progressively smaller parameter set errors as well as the identity of the true model.

V. Class of Heuristic Model Evaluation Functions

One example of a model evaluation function can be in the following form:

$$J = \sum_{j=1}^{N_M} \left[a_j(\omega_j - \omega_j^m)^2 + b_j(\phi_j - \phi_j^m)^T(\phi_j - \phi_j^m) \right] \tag{56}$$

where N_M is the number of modes used, a and b are weights, and ω and Φ are the natural frequency and mode shape, respectively. According to the assumption

$$J(\overline{k,m,\theta},H^m) = 0 \tag{57}$$

then it follows from Eq. (56) that, ideally,

$$\omega_j^m = \omega_j(\overline{k,m,\theta}) \tag{58a}$$

$$\phi_j^m = \phi_j(\overline{k,m,\theta}) \tag{58b}$$

Then,

$$\omega_j(\overline{k,m},\theta_i) = \omega_j^m + \left(\frac{\partial_j^\omega}{\partial\theta}\right)_{\theta_i}\Delta\theta \tag{59a}$$

$$\phi_j(\overline{k,m},\theta_i) = \phi_j^m + \frac{\partial_j^\phi}{\partial\theta}\Delta\theta \tag{59b}$$

where $\Delta\theta_i = |\theta_i - \bar{\theta}|$. Terms of the order of $\Delta\theta^2$ are not present in Eq. (59) because of the assumption of the available good guess of θ_i and $|\Delta\theta| < \delta_b$. Then,

$$\left[\frac{\partial J}{\partial\theta}\Delta\theta\right]_{\overline{km}} = \sum_{j=1}^{N_M}\left[2a_j(\omega_j-\omega_j^m)\frac{\partial\omega}{\partial\theta}\Delta\theta + 2_j^b\left(\frac{\partial_j^\phi}{\partial\theta}\Delta\theta\right)^T(\phi_j-\phi_j^m)\right] \tag{60}$$

From Eqs. (59) and (60),

$$\left[\frac{\partial J}{\partial\theta}\Delta\theta\right]_{\overline{km}} = \sum_{j=1}^{N_M}\left[2a_j(\omega_j-\omega_j^m)^2 + 2_j^b(\phi_j-\phi_j^m)^T(\phi_j-\phi_j^m)\right] > 0 \tag{61}$$

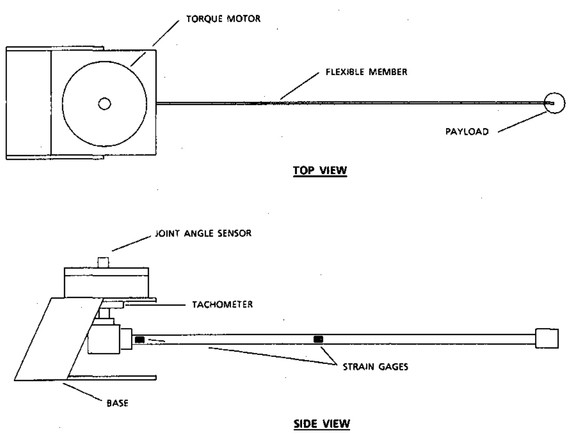

Fig. 5 Flexible single-link manipulator used for verifying "actual" TVFS models.

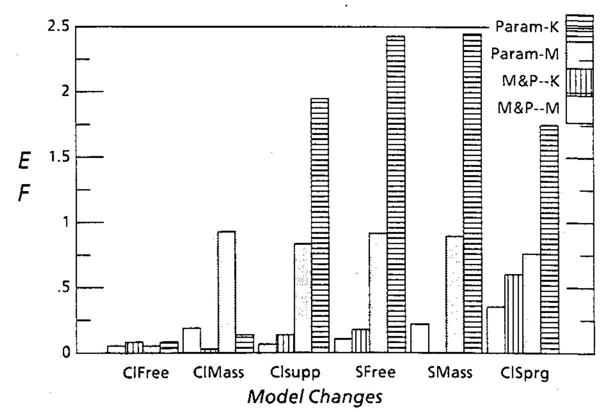

Fig. 6 Example of model identification and parameter identification vs parameter identification only.

Since Eq. (61) satisfies Eq. (41) for any true model in R, then by Corollary 1, Theorem 2 is satisfied by this example of J.

VI. Numerical Example

The TVFS example used in this paper was based on variations of simple beam configurations, such as might occur to a robot manipulator link with a locking motor at its joint. Possible configurations, with or without added mass (payload), are given in the search tree shown in Fig. 4. A series of data packets consisting of the beam's first natural frequency and mode shape was furnished to the program, which in turn triggered model and then parameter identification. The input frequency and mode-shape data was corrupted with 10% noise. After one cycle of identification, the mass, stiffness, and damping matrices associated with the identified model were then reformulated in state form.

The initial model specified was that of a cantilever. For comparison, this initial model was also the a priori model form assumed for a conventional explicit self-tuning regulator (EST) that used parameter identification alone in its adaptive loop. The initial model was also used to design a fixed optimal controller. Both the CMR, EST, and fixed-control procedures incorporated the same control-design method of Moerder and Calise. For these examples, the CMR adaptive loop used only one pass through identification.

Four successive structural-identity changes were considered in the SISO example, e.g., from an initial cantilever model, to a cantilever with a tip point mass, a clamped-free beam, and then a simply supported beam. In terms of the physical system, these would roughly correspond to an added payload, forcible contact with a surface, and freely resting (unlocked joint) against a lower surface.

A five-degree-of-freedom structural dynamic model was produced by the identification program, with associated eigenvalues and eigenvectors, for the cantilever and load-cantilever models. The clamped-supported and simply supported models were identified as fourth-order models, due to the added constraint at $X = L$.

Both MIMO and SISO examples were considered. The "actual" model was obtained for each structure from a large-scale GTSTRUDL[25] finite-element model. Some of these large-scale models were experimentally verified by tests on a robotic structure shown in Fig. 5, which is discussed also in a paper by

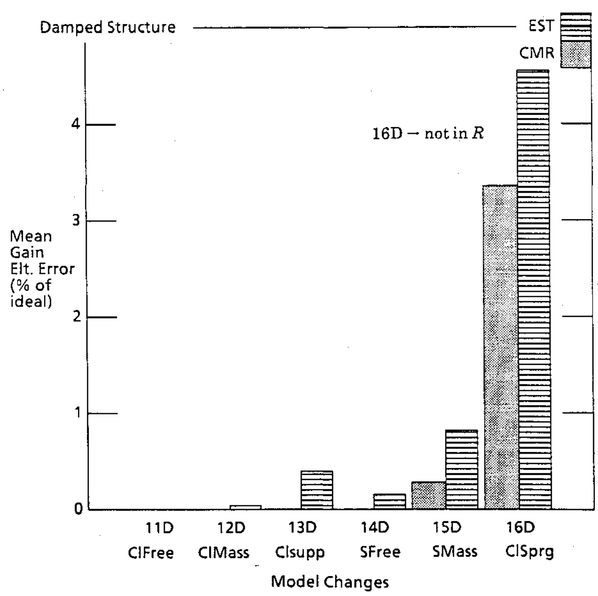

Fig. 7 Comparison of mean gain errors for damped-model changes, for optimal gains found by both CMP and EST.

Table 1 Comparison of gains for the MIMO damped loaded-cantilever example (12D), with moment feedback

G	Adaptive method		
	CMR	EST	Fixed
g_{11}	−1.3934E-3	−2.1064E-2	−8.1915E-4
g_{12}	3.9312E-5	1.2093E-2	−4.8156E-5
g_{21}	−1.5569E-4	1.0210E-2	2.6592E-4
g_{22}	−2.3276E-4	−6.0798E-3	−1.5307E-4

Table 3 Comparison of closed-loop damping ratios for the MIMO damped loaded-cantilever example (12D, $\zeta = 0.005$), with moment feedback

Order	ζ_{CL}			
	Uncontrolled structure	CMR	EST	Fixed
1	0.00500	0.00828	(unstable)	0.00134
2	—	1.0		1.0
3	0.00500	0.11484		0.08207
4	—	1.0		1.0
5	0.00500	0.13312		0.13338

Table 2 Comparison of closed-loop eigenvalues for the MIMO damped loaded-cantilever example (12D), with moment feedback

Order	Λ_{CL}		
	CMR	EST	Fixed
1	−0.1313E-1 ± 0.1585E1	−0.1462 + 0.0	−0.2119E-2 ± 0.1584E1
2	−0.2728E2 + 0.0	0.2537E2 + 0.0	−0.4468 + 0.0
3	−0.9703E1 ± 0.839332	0.5662E2 + 0.0	−0.6583E1 ± 0.7994E2
4	−0.2031E3 + 0.0	0.8136E2 + 0.0	−0.3554E3 + 0.0
5	−0.6334E2 ± 0.4716E3	−0.5004E1 ± 0.1004E3	−0.6224E2 ± 0.4624E3

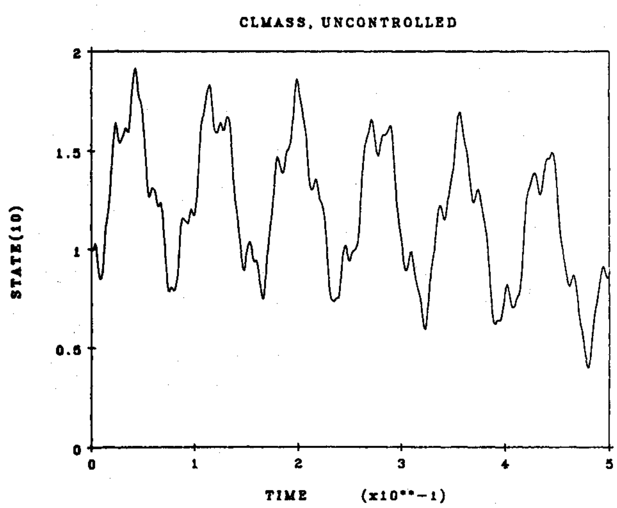

Fig. 8 Time plot of tip velocity of 12D, uncontrolled.

Hastings and Book.[26] A comparison of mean model-parameter errors for both CMR and EST identification processes, given in Fig. 6, provides a cross reference for model errors.

In the MIMO example, the changes in structural identity described for the SISO case resulted in comparable differences in the gain matrices obtained by each method. These matrices are compared in Fig. 7 for CMR and EST gain errors. The mean gain error values given are calculated with respect to the optimal gains obtained for the actual model:

$$\bar{v} = \sum_{i=1}^{N_M} \sum_{j=1}^{N_P} \left[\frac{(\bar{g}_{ij} - g_{ij})^2}{\bar{g}_{ij}} \right] \tag{62}$$

where $\bar{g}_{ij}$ is an element of the actual model's gain matrix, and g_{ij} is an element of one of the other gain matrices.

Example Using Moment-Rate Feedback

Although the previous examples using linear displacement and velocity feedback demonstrate an advantage for CMR

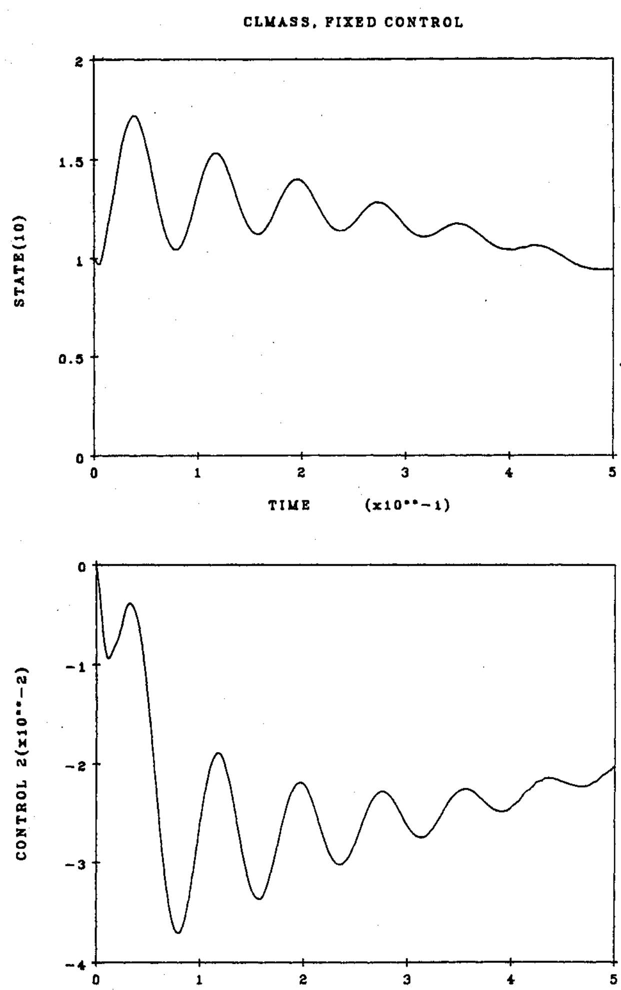

Fig. 9 Time and control input plots of tip velocity of 12D, with fixed control.

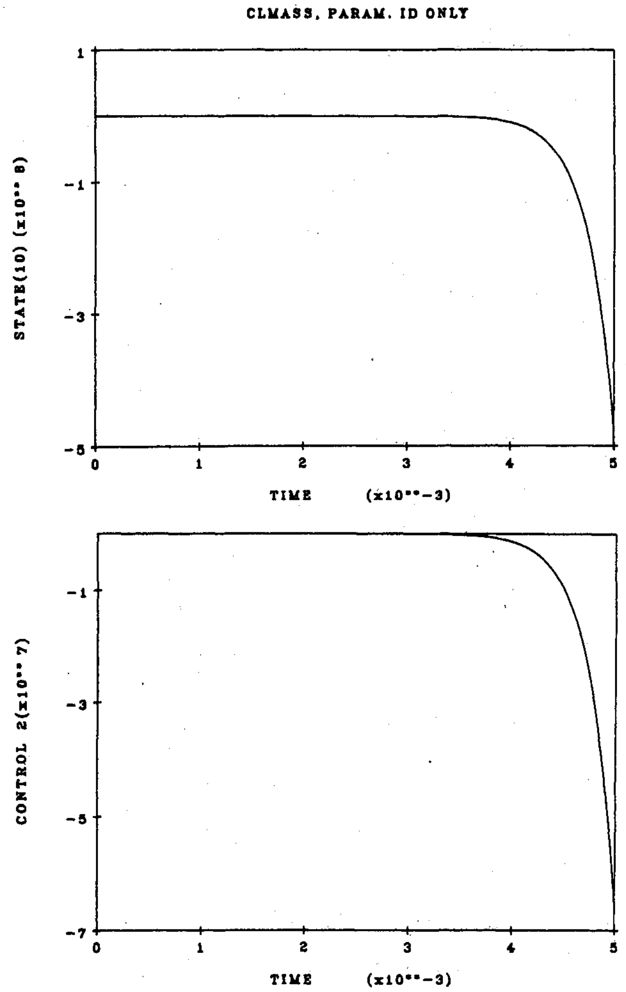

Fig. 10 Time and control input plots of tip velocity of 12D, with parameter identification only.

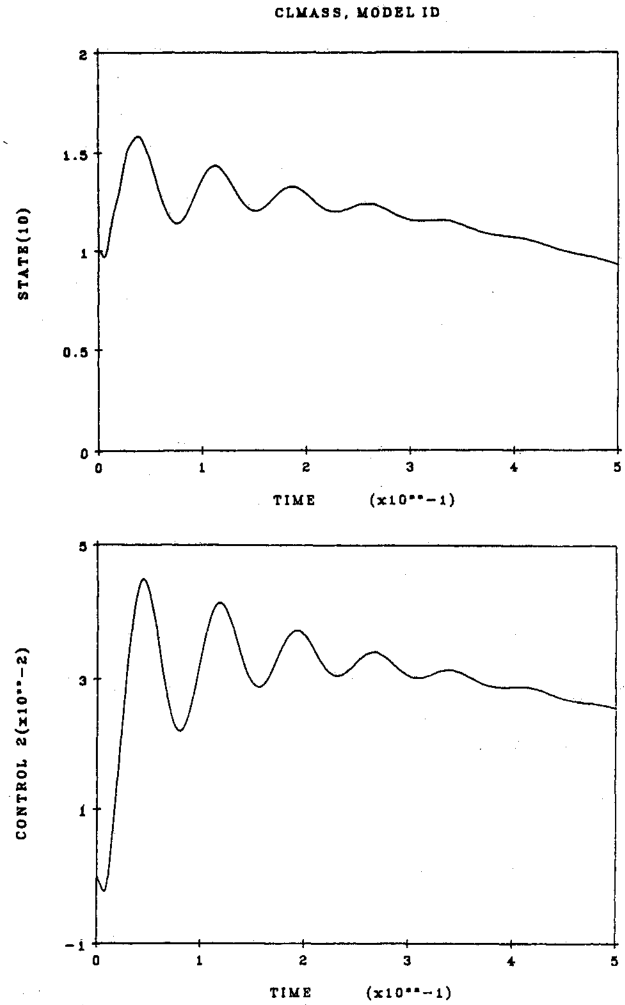

Fig. 11 Time and control input plots of tip velocity of 12D, with model identification.

over EST, the distinction is somewhat obscured by the poor damping characteristics of this type of feedback. Another possible feedback approach uses instead the rate of change of the moment of the beam, which can be related to the curvature of the beam via the moment-curvature relation. Feedback approaches of this type, e.g., Hanagud, Obal, and Calise,[27] measure the output in terms of the difference over a given sensor length of the time rate of change in the curvature, or

$$y_j = P\int_{x_{i-1}}^{x_i} \omega^i_{xxt}\,\mathrm{d}x = P[\theta^j_t(x_i) - \theta^i_t(x_{i-1})] \tag{63}$$

where P is a constant coefficient, y_i is the jth output state, and $|x_i - x_{i-1}|$ is the length of the sensor. Defining the state variables as

$$Z^T = \{\theta_1, \ldots, \theta_n; \theta_{t_1}, \ldots, \theta_{t_n}\} \tag{64}$$

it is possible to reduce Eq. (8) to Eq. (14), but with parameters defined as

$$A = \begin{bmatrix} 0 & I \\ -M^{r^{-1}}K^r & -M^{r^{-1}}V^r \end{bmatrix}, \qquad B = \begin{bmatrix} 0 \\ -M^{r^{-1}}K_D \end{bmatrix}$$

$$C = [0\;\; K_s] \tag{65}$$

where $(\ )^r$ denotes a reduced mass, stiffness, or damping matrix for which the translational degrees of freedom have been removed by a suitable condensation method, and K_D and K_s are sensor and actuator parameter matrices.

A control example was made using this feedback approach with the MIMO damped loaded-cantilever example. Optimal control gains were obtained just as described in the previous chapter, except that the diagonal elements of the weighting matrix Q were selected to be inversely proportional to the square of the eigenvalues. As shown in Tables 1–3, the gains, closed-loop eigenvalues, and damping ratios of CMR, EST, and a fixed optimal control also indicate an advantage for CMR, but with the differences increased. Note that the use of parameter identification only (EST) resulted in an unstable closed-loop system for this selected example, due to large gains that were due in turn to a widespread reduction in model stiffness values by the parameter-identification algorithm. The large advantage of CMR in damping and stability is also shown in time plots of tip velocity and control input (Figs. 8–11). Note that the control input plots refer to values at the second (outer) sensor-actuator location, and that the EST time plots differ in scale from the others due to an instability.

VII. Conclusions

An adaptive-control technique for discontinuously time-varying structures has been developed, using both model identification and parameter identification to replace controllers when large-scale discontinuous model changes occur. This method (CMR) has been demonstrated for a test problem of controlling a beam for which boundary conditions change suddenly in time. A linear optimal output feedback approach was used to design the controller, once a new model was identified. For SISO and MIMO test problems, CMR followed the actual model more closely than a comparable explicit self-tuning regulator (EST) that used only parameter identification, resulting in better closed-loop poles, hence better stability and performance. Since large, flexible structures of interest may experience sudden model-structure changes, these results suggest that the use of selected AI techniques may hold some promise for certain identification and control problems. More research will be needed to firmly establish the direct applicability of CMR to current problems with time-varying boundary conditions, such as identification and control of large-scale space structures with smooth sensors and actuators. The use of heuristic search and object-oriented programming in model identification is an example of the use of AI methods for control problems.

Acknowledgments

We wish to thank Dr. Wayne Book for access to his robotic experimental facilities that were used for model verification. Also due thanks is Dr. V. R. P. Jonnalagadda for his advice and programming assistance, and Xerox Corporation for computational support. This research was supported in part by U.S. Army Research Contract DAAG 29-82-K0094.

References

[1]Astrom K. J., and Peterka, V., "Control of Multivariable Systems with Unknown But Constant Parameters," *3rd IFAC Symposium on Identification and System Parameter Estimation,* Pergamon, Oxford, England, 1973, pp. 535–544.

[2]Hanagud, S., Meyyappa, M., and Craig, J. I., "Identification of Structural Dynamic Systems," *Recent Trends in Aeroelasticity Structures and Structural Dynamics,* Univ. of Florida Press, Gainesville, FL, 1987, pp. 120–141.

[3]Baruch, M., "Optimal Correction of Mass and Stiffness Matrices Using Measured Modes," *AIAA Journal,* Vol. 20, No. 11, 1982, pp. 1623–1626.

[4]Kubrusly, C. S., "Distributed-Parameter System Identification: A Survey," *International Journal of Control,* Vol. 26, No. 4, 1977, pp. 509–535.

[5]Sinha, N. K., and Kuszta, B., *Modeling and Identification of Dynamic Systems,* Van Nostrand Reinhold, New York, 1983, Chap. 2–5, pp. 6–104.

[6]Stein, G., and Saridis, G. N., "A Parameter-Adaptive Control Technique," *Automatica,* Vol. 5, No. 6, 1969, pp. 731–739.

[7]Deshpade, J. G., Upadhyay, T. N., and Lainiotis, D. G., "Adaptive Control of Linear Stochastic Systems,"*Automatica,* Vol. 9, No. 1, 1973, pp. 107–115.

[8]Wittenmark, B., "A Two-Level Estimator for Time-Varying Parameters," *Automatica,* Vol. 15, No. 1, 1979, pp. 85–89.

[9]Schumann, R., Lachmann, K. H., and Isermann, R., "Towards Applicability of Parameter-Adaptive Control Algorithms," *Proceedings of the 8th IFAC Triennial World Congress,* Pergammon, Oxford, England, 1981, pp. 903–910.

[10]Sacerdoti, E., "Planning in a Hierarchy of Abstraction Spaces," *Artificial Intelligence,* Vol. 5, No. 2, 1974, pp. 115–135.

[11]Moerder, D. D., and Calise, A. J., "Convergence of a Numerical Algorithm for Calculating Optimal Output Feedback Gains," *IEEE Transactions on Automatic Control,* Vol. AC-30, No. 9, Sept. 1985, pp. 900–903.

[12]Nilsson, N. J., *Principles of Artificial Intelligence,* Tioga Publishing, Palo Alto, CA, 1980, Sec. 2.4, pp. 72–88.

[13]Rich, E., *Artificial Intelligence,* McGraw-Hill, New York, 1983.

[14]Feigenbaum, E., "The Simulation of Verbal Learning Behavior," *Computers and Thought,* edited by E. Feigenbaum and J. Feldman, McGraw-Hill, New York, 1963, pp. 297–309.

[15]Glass, B. J., *Identification and Control of Time-Varying Flexible Structures: An Artificial Intelligence Approach,* Ph.D. Thesis, School of Aerospace Engineering, Georgia Inst. of Technology, Atlanta, GA, Sept. 1987.

[16]Minsky, M., "A Framework for Representing Knowledge," *The Psychology of Computer Vision,* edited by P. Winston, McGraw-Hill, New York, 1975, pp. 211–277.

[17]Hart, P. E., Nilsson, N. J., and Raphael, B., "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," *IEEE Transactions on Systems, Science and Cybernetics,* Vol. SSC-4, No. 2, 1968, pp. 100–107.

[18]Hanagud, S., and Glass, B. J., "Structural Dynamics Model Identification Using Heuristic Search," AIAA Paper 87-0747, April 1987.

[19]Hanagud, S., Meyappa, M., Cheng, Y. P., and Craig, J. I., "Identification of Structural Dynamic Systems with Nonproportional Damping," *AIAA Journal,* Vol. 24, No. 11, pp. 1880–1882, 1984.

[20]Saridis, G. N., "Intelligent Robotic Control," *IEEE Transactions on Automatic Control,* Vol. AC-28, No. 5, May 1983, pp. 547–557.

[21]Balas, M. J., "Feedback Control of Flexible Systems," *IEEE Transactions on Automatic Control,* Vol. AC-23, No. 4, Aug. 1978, pp. 673–679.

[22]Meirovitch, L., Baruh, H., and Öz, H., "A Comparison of Control Techniques for Large Flexible Systems," *Journal of Guidance, Control, and Dynamics,* Vol. 6, No. 4, July–Aug. 1983, pp. 302–310.

[23]Meirovitch, L., and Baruh, H., "Control of Self-Adjoint Distributed-Parameter Systems," *Journal of Guidance, Control, and Dynamics,* Vol. 5, Jan.–Feb. 1982, pp. 60–66.

[24]Rotman, B., and Kneebone, G. T., *The Theory of Sets and Transfinite Numbers,* Oldbourne, London, 1966, pp. 19–21.

[25]*GTSTRUDL Users Manual,* 3 Vols., GTICES Systems Lab., School of Civil Engineering, Georgia Inst. of Technology, Atlanta, GA, 1985.

[26]Hastings, G. G., and Book, W. J., "A Linear Dynamic Model for Flexible Robotic Manipulators," *IEEE Control Systems Magazine,* Vol. 7, Feb. 1987, pp. 61–64.

[27]Hanagud, S., Obal, M. W., and Calise, A. J., "Optimal Vibration Control by the Use of Piezoceramic Sensors and Actuators," AIAA Paper 87-0959, April 1987.