An Algorithm for Obtaining Proper Models of Distributed and Discrete Systems

B. H. Wilson
Department of MIME,
Northeastern University,
334 Snell Engineering Center,
Boston, MA 02115

J. L. Stein
Department of MEAM,
University of Michigan,
2250 GG Brown Lab,
Ann Arbor, MI 48109

The development of automated modeling software requires strategies for synthesizing mathematical models of systems with distributed and discrete characteristics. A model order deduction algorithm (MODA) is developed to deduce a Proper System Model by selecting the proper complexity of submodels of components in a system subject to a frequency based metric. A Proper Model in this context means that (1) the system model has the minimum spectral radius out of all possible system models of equivalent or greater complexity, and (2) any increase in the model complexity will result in spectral radius beyond a specific frequency range of interest. Proper Models are also defined to have physically meaningful parameters. Proper Models are intended to be useful for design, where mapping the relationship between design parameters and dominant system dynamics is critical. While MODA is illustrated using the application of machine-tool drive systems, it is readily applicable to other modeling applications.

1 Introduction

The development of automated modeling software (de Kleer and Williams, 1991; Rinderle and Subramaniam, 1991; Stein and Tseng, 1991) will require algorithms to develop mathematical models of distributed and discrete or network engineering systems. Systems of this type are ubiquitous; for example, drive trains of rotating machinery contain (distributed) compliant and inertial shafts that connect (discrete) pulleys, gears, and torsional loads. The input to the automated-modeling software is envisioned to be a high-level system description, i.e., a description based solely on component geometric information and material constants. The modeling algorithms should transform this description into a set of ordinary-differential equations (state equations) whose parameters and coefficients are defined directly by the component geometry and material constants found in the system description and whose complexity is commensurate with the goals of the modeling exercise. The algorithms should be efficient, i.e. the algorithm should not require an inordinate amount of time to create a model, regardless of the number of components in the system description. The models created by the algorithms can be used in conjunction with analysis and simulation software such as Matlab to study system behavior. In this manner, the availability of automated-modeling software will facilitate engineers to analyze and simulate the behavior of a variety of different configurations at an early stage in the design process, since the burden of creating a simulation model will be reduced.

State equations have been used as a basis for performing system analysis and controller design. However, only more recently has there been an awareness of the benefits of modeling early in the design phase (Rosenberg, 1991), with particular advantages resulting when low-order models are used (Hogan, 1991). This potential can be more fully realized if software tools for generating models are available to assist the designer. The idea of automating the creation of models has only recently been proposed in the engineering community. Stein and Tseng (1991) argue that the modeling process is more structured than commonly believed, and, hence, more amenable to automation. Rinderle and Subramaniam (1991) have developed an algorithm that simplifies a (potentially) complex model into a minimal set of state equations. Commercial software programs such as Adams can be used to obtain a set of algebraic and ordinary-differential equations from a system description. However, when using this program, the onus remains on the user to decide which parts of the system to model as rigid and which parts to model as compliant. In effect, the user chooses the required complexity of the model. The authors believe that the development of truly automated modeling software should not require the user to specify how the system is to be modeled; rather, the software itself should make this decision. Such software will require algorithms that are able to transform a system description based purely on component geometry, interconnections, and material constants into a model whose complexity is suited to the goal(s) of the modeling exercise. Furthermore, creating models of a complexity level suited to the goals of the modeling exercise requires that a quantitative measure must be defined to dictate the required complexity of the model. It will be shown that the *frequency range of interest* (FROI) provides one such measure.

This paper describes a model-order deduction algorithm (MODA) that coordinates the transformation of a geometric description of a one-dimensional, network-type system into a Proper Model for a given FROI. Systems of this type are characterized by a collection of components connected serially, and are ubiquitous in manufacturing, e.g. spindle and feed drives; defense, e.g. stabilized pointing and tracking platforms; and automobiles, e.g. engine and transmission drive trains. A key feature of this algorithm is the ability to relate the modeling of any individual component to (1) the goals of the modeling exercise and (2) the effect of the component's behavior on the overall system behavior. The algorithm also synthesizes models of minimum spectral radius and whose parameter coefficients are derived from design data, e.g., component geometry and material properties.

1.1 Parameter-Lumping Techniques. To provide further motivation for the utility of the algorithm discussed in this paper, an example of a modeling exercise is given. Two parameter lumping techniques are considered: one that novice modelers might employ and one that expert modelers might employ. For illustration, consider a system consisting of three torsional flywheels and two torsional shafts.

Contributed by the Dynamic Systems and Control Division for publication in the JOURNAL OF DYNAMIC SYSTEMS, MEASUREMENT, AND CONTROL. Manuscript received by the DSCD December 3, 1992. Associate Technical Editor: H. Asada.

Novice modelers may approach the modeling of this system in several ways: one might model one shaft as (infinitely) rigid and the other as a single compliance; another modeler may represent each shaft with a single compliance; yet another modeler may try to combine the compliance of each shaft into a single compliance; etc. Even if each modeler correctly relates material dimensions and physical constants to the shaft compliance (i.e., spring rate), there is still no guarantee that the models created by these individuals are well suited to the task at hand.

Expert modelers may approach the modeling of the system quite differently. They might first assess the goal(s) of the modeling exercise and translate these goals into the required model bandwidth, ω_{req}. They could then test the affect on the model of including the compliance of each shaft in the model. If including the compliance of one shaft resulted in a natural frequency greater than ω_{req}, they would tend to dismiss the compliance of this shaft as irrelevant to the task at hand, and thus not include this shaft's compliance in the model. Conversely, if including the compliance of a shaft in the model resulted in a natural frequency less than ω_{req}, the expert would certainly include this compliance in the model. Thus, through patiently trying different parameter lumping combinations, the expert determines which structural effects to include in a model and which to omit.

The modeling scenarios of the flywheel-shaft system illustrate the various techniques that engineers employ to synthesize a system model. The scenarios indicate the number of decisions involved for modeling even such a simple system. The decisions required to synthesize a model for a general N-component system, e.g., deciding how many compliances (if any) to use to model each element, are potentially extensive and time consuming. A formal methodology, or algorithm, to make these decisions is presently lacking.

1.2 Content of Paper. The requirements of MODA, the submodel-synthesis algorithms that MODA calls, and the search strategy that MODA uses to find the proper combination of component submodels follow (Section 2). An example of MODA operation is presented (Section 3). The article then addresses some of the potential problems with MODA's search strategy, extensions to MODA that enable it to synthesize models meeting different criteria, and pole migration as model rank increases (Section 4).

2 Description of MODA

2.1 Algorithm Requirements. The purpose of the algorithm is to coordinate the synthesis of a minimum complexity, lumped, physical model—a Proper Model—that is intended to represent the response of a configuration of distributed and discrete components. The input to the algorithm is a purely geometric and physical description of the configuration. During the model synthesis process, models synthesized by MODA are to meet three criteria:

1. The model parameters and coefficients relate directly to the component dimensions and material constants.
2. The sum of the individual inertias in the model will equal the total inertia of the actual system.
3. The sum of component ranks is distributed in such a manner as to minimize model spectral radius (ρ_{model}) for a given sum of ranks; any increase in model rank will result in a ρ_{model} greater than specific FROI.

The first requirement ensures that the connection between the actual system and the model is clear, so that an engineer using the model can relate changes in the dimensions and material constants of the actual system to changes in the model parameters and coefficients. The second requirement places restrictions on the model assembly process; it's intended to ensure that the assembly process includes all component inertias in the system

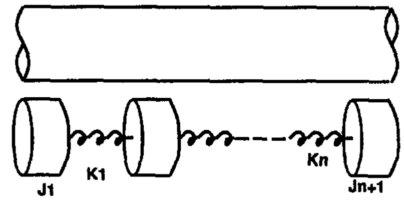

Fig. 1 Torsional shaft and physical model

model. Component inertias are retained in the system model because they affect motor-sizing and power specifications. The last requirement is motivated by a need to synthesize models that are both low order and predict the lowest resonant frequencies of a system. Before elaborating on this further, some background on submodel synthesis algorithms and the FROI is necessary.

2.2 Submodel-Synthesis Algorithms. A program implementing MODA will need a set of routines to synthesize component submodels from geometric descriptions of the components. These routines, submodel-synthesis algorithms (SSAs), are specific to each component. A variable called *rank* is associated with each component and is used in conjunction with the component's SSA to specify the complexity of the submodel of a given component. This variable has been defined because it is necessary to order the component submodels. A practical ordering scheme is one in which larger ranks correspond to more complex submodels. The simplest model of a given component is its rank-0 model. The rank-0 model is in effect a rigid-body model of a component, which in the case of mechanical components implies that no compliant elements are present in the model. Components and their associated SSAs can be separated into two categories: unbounded rank and bounded rank. These categories are discussed next.

2.2.1 Unbounded-Rank Components. The SSAs of unbounded rank components permit models of rank 0-∞ to be synthesized from them. The torsional shaft is one such component, this shaft is illustrated along with its corresponding physical model in Fig. 1.

In Fig. 1, the magnitude of the individual inertias and compliances are obtained by the following equations (Rao, 1990):

$$J_{shaft} = \frac{\rho \times \pi \times L \times D^4}{32} \tag{1}$$

$$J_i = \frac{J_{shaft}}{N + 1} \tag{2}$$

$$K_{shaft} = \frac{G \times \pi \times D^4}{32 \times L} \tag{3}$$

$$K_i = N \times K_{shaft} \tag{4}$$

$$N \geq 0 \tag{5}$$

Where

K_{shaft} = the torsional spring rate of the shaft
N = the rank associated with the shaft
L = the length of the shaft
D = the diameter of the shaft
G = the shear modulus of the shaft
J_{shaft} = the torsional inertia of the shaft
J_i = inertia coefficients in the model
K_i = spring rate coefficients in the model

Note that for the shaft, the rank N equals the number of torsional springs in the physical model. Equations (1)–(5) are equally applicable for a ballscrew with one change. In the case of a ballscrew L_{shaft} is the length of the ballscrew up to the point

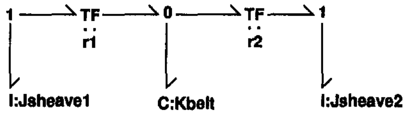

Fig. 2 Bond graph of rank-1 belt-drive

where the ballnut is located; the remaining portion of the ballscrew is treated as a torsional inertia.

2.2.2 Bounded-Rank Components. The fundamental difference between the SSAs for unbounded rank components and bounded rank components is that the most complex models of the latter are assumed to contain only a finite number of independent energy storage elements, i.e., states. Drive-train components in this category include a DC motor, a gear-pair, a belt-drive, and a ballnut. Two components whose ranks are assumed to be bounded are considered here: the belt-drive and the DC motor.

Belt-Drive. A belt-drive consists of two sheaves and a massless, compliant belt. Note, a massless belt is an application-dependent modeling assumption. In the case when the compliance of the belt is included in the model (rank-1) there are three independent energy storage elements. The bond graph for a rank-1 belt-drive is shown in Fig. 2.

In the rank-0 case the compliance of the belt is not included in the model, i.e. the belt is assumed to be infinitely stiff. The two sheaves are now kinematically coupled, and the rank-0 belt-drive has only one independent energy storage element.

DC Motor. In common practice a DC motor is modeled with either one or two independent energy storage elements (Franklin and Powell, 1991). When the motor is modeled with two independent energy storage elements (rank-1), the flux linkage is included in the model. The bond graph for the rank-1 motor is shown in Fig. 3.

In the rank-0 case, the inductance of the winding is assumed negligible and the rotor angular momentum is the only independent energy storage element. Note that in modeling the motor in this manner, effects such as heating of the winding, saturation, cogging, etc. are not included in the model. These effects are implicitly assumed to be negligible in the drive train application.

2.3 Frequency Range of Interest. Karnopp et al. (1990) use the FROI to determine the required complexity of a model synthesized from a configuration description. They discuss the FROI primarily in the context of the frequency content of an input to the system. They recommend that the model of a distributed component, such as a beam, should only be accurate to two to five times the maximum excitation frequency in the input. Accurate means that the model contains sufficient modal information to predict how the actual system would respond to frequencies in a range between 0 to $5 \times \omega_{\text{max-excitation}}$. Thus, $5 \times \omega_{\text{max-excitation}}$ becomes the required model bandwidth. The spectral radius, ρ_{model}, is the Euclidean norm of the largest eigenvalue of a state matrix, and is effectively the highest complex-scalar input frequency to which a model can reliably respond. Using the criterion from Karnopp et al., if $\rho_{\text{model}} > 5 \times \omega_{\text{max-excitation}}$, the model can accurately respond to frequencies up to $\omega_{\text{max-excitation}}$.

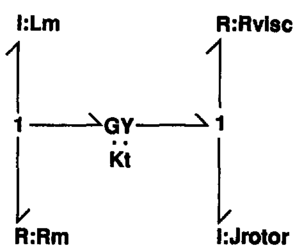

Fig. 3 Bond graph of DC motor with flux linkage modeled

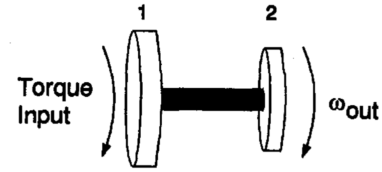

Fig. 4 Flywheel-shaft system to be controlled

The use of a model for controller design also places frequency-related requirements on the model. Consider the flywheel-shaft-flywheel system shown in Fig. 4. A torque is applied to the Flywheel 1 and the goal is to control the velocity of Flywheel 2. A proportional controller will be used to regulate the velocity. The desired open-loop frequency response of the flywheel-shaft system is shown in Fig. 5, which illustrates two frequency-response plots.

The first plot in Fig. 5 shows a resonance whose peak rises above the 0-dB line; this condition would result in unstable, closed-loop operation. If the flywheel-shaft system has a resonance of sufficient magnitude and proximity to the open-loop crossover frequency, the model should include this resonance. The second plot in Fig. 5 shows a resonance at a frequency much greater than the crossover frequency. A resonance of this magnitude at this frequency would not result in unstable operation, nor would it adversely affect system performance. Therefore, there is little reason to include this resonance in the model. The magnitude of a given resonance (the amount of peaking in the frequency-response plot) is a function of the damping in the system. An accurate, theoretical estimate of the amount of peaking caused by a resonance remains a research topic. However, a rule of thumb suggests that upper-bound of the FROI, i.e. the required ρ_{model}, should be 5 to 10 times the desired crossover frequency.

Returning now to the third MODA requirement, MODA is to distribute component ranks such that ρ_{model} is minimized and any increase in the sum of the ranks results in $\rho_{\text{model}} > \omega_{\text{req}}$. An algorithm with this capability can be used to synthesize the minimum-rank model accurate to a frequency ω_{req}. A brief example illustrates how such a model can be synthesized. Consider the three-mass, two-shaft configuration in Fig. 6, and three possible models of the configuration, shown in the same figure.

For a FROI of 1.25 (radians/second), Model A and Model B of Fig. 6 both predict poles within this range, and they both have the same rank. However, Model A has a smaller ρ_{model} than Model B, and is preferred, because it provides a more

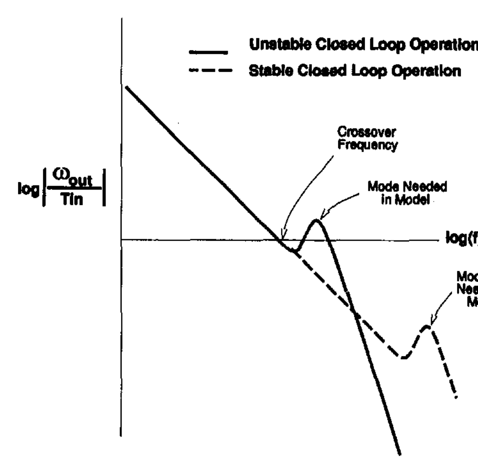

Fig. 5 Desired open-loop frequency response

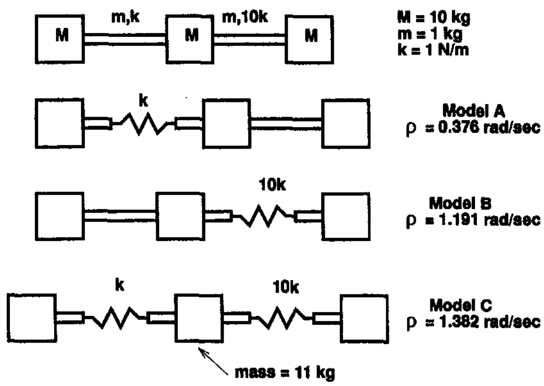

Fig. 6 Three-mass, two-shaft configuration and models

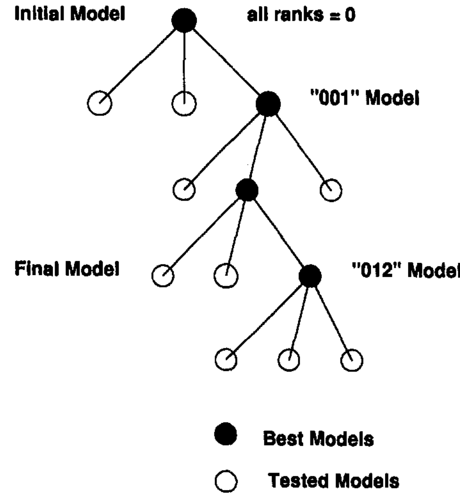

Fig. 8 Reduced search space

realistic prediction of system performance than Model B. Continuing with this example, Model C has a ρ_{model} greater than the FROI and an rank greater than Model A. Model C provides unnecessary information, and does not justify the increase in model rank. Thus, of the three models in Fig. 6, Model A is the preferred model, and the model that MODA is to synthesize.

2.4 Search Strategy. MODA's task is to find the combination of component ranks that, in conjunction with the component SSAs, synthesizes a Proper Model. An exhaustive (blind) search strategy for this combination of component ranks is likely to be inefficient, because it results in a potentially very large search-space; whereas a more informed search strategy pares this space considerably, and will thus be more efficient.

Consider the problem of finding a Proper Model for an N-component configuration. The exhaustive search strategy, depicted in Fig. 7, starts at the root, a rank-0 configuration, and creates a set of models that correspond to individually increasing the rank of each component in the configuration (level-1), where level-1 corresponds to a configuration whose component ranks sum to 1. Each level-1 model spawns another set of models, and the total number of models tested, *at each level,* equals

$$N_{\text{models-per-level}} = \frac{(N_{\text{components}} + M_{\text{level}} - 1)!}{M_{\text{level}}!(N_{\text{components}} - 1)!} \tag{6}$$

The search concludes when the bandwidth of all models at a given level M is greater than the FROI, and the solution is the model (at level $M - 1$) with the minimum bandwidth. Clearly, for an arbitrary N-component configuration and M-level search, the number of models tested can become prohibitively large.

A heuristic to guide the search can greatly reduce the search-space depicted in Fig. 7. The heuristic is derived from the ρ_{model} of the model associated with each node. The idea is to select the model with the minimum ρ_{model} (at each level), and to use *this model* as a *starting point* for finding the *next model.* The model with the minimum ρ_{model} is found by testing the effect on ρ_{model} of increasing the rank of each component. The component that causes the smallest increase in ρ_{model} when the rank of this component is increased is referred to as the weak-dynamic link component. The search-space for a three-component configuration that results from using the heuristic is shown in Fig. 8.

This search technique fits in the category known as steepest-ascent hill-climbing or gradient search. Using this technique the number of total trial solutions (models) needed to evaluate an N-component configuration (assume for simplicity that all components are of type unbounded-rank.) to a level of M equals:

$$N_{\text{total-models}} = 1 + N_{\text{components}} \times (M_{\text{level}} + 1) \tag{7}$$

A comparison of (6) and (7) clearly indicates that the exhaustive search results in a much larger search space than the steepest-ascent hill-climbing search strategy.

MODA is the algorithm that coordinates the gradient search for a Proper Model. This algorithm, depicted in Fig. 9, synthesizes a Proper Model at every level in the search, and should be used to increase the rank of the model until ρ_{model} exceeds the FROI. At this point, MODA would decrease the rank of the last component to have its rank increased. In this manner the resulting model has minimal ρ_{model} and minimal rank for a given FROI. Once the model has this rank, any additional increase of the rank of any component will result in ρ_{model} beyond the FROI.

There are several well-known problems associated with this type of search, and the ability of MODA to avoid these problems in synthesizing a Proper Model will be addressed in the Discussion.

3 Example of Algorithm Operation

An example of the use of MODA to synthesize multiple models of same configuration will illustrate how the model rank changes with the FROI, and how the sequence in which the

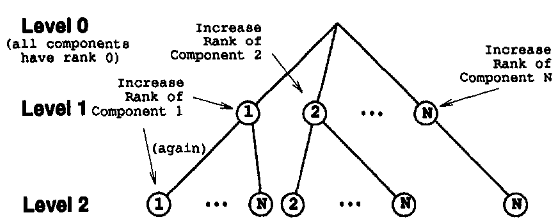

Fig. 7 Exhaustive search space for proper model

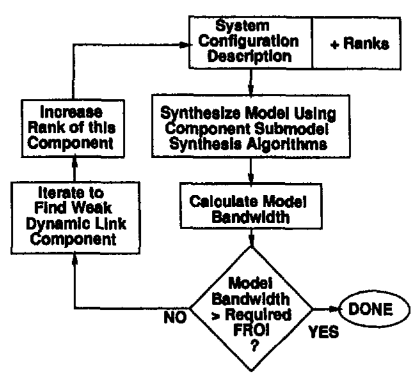

Fig. 9 Model-order-deduction algorithm

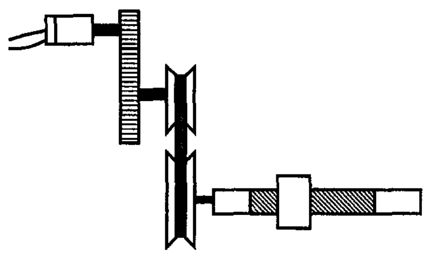

Fig. 10 Drive-train to be modeled

component ranks change depends on the values of the component parameters. The configuration, shown in Fig. 10, consists of a DC motor, a belt-drive, a torsional shaft, and a flywheel. The dimensions and material constants of these components are given in the Appendix.

MODA was applied to the drive train in Fig. 10. Different models were synthesized as the FROI increased, and the order (which is related to rank) of these models was tabulated for different FROI. Figure 11 plots the model order versus the FROI for the drive train using two sets of component parameters.

As Fig. 11 indicates, more component dynamics need to be included in the model as the FROI increases. The plots shown in Fig. 11 also illustrate that the sequence in which the component ranks change (while the FROI is increasing) depends on both the component types and their parameter values. For Parameter Set A, the relatively compliant belt-drive A requires the rank of this component to be increased first, which results in a model order of three at a frequency of 56 rad/s. Later, model order increases to six at 496 rad/s (a frequency beyond the pole caused by the inductance of the motor windings), when the compliance of the shaft is needed in the model. For Parameter Set B, in contrast, the compliance of the shaft must be included in the model first, at 112 rad/s. The relatively stiff belt compliance requires the belt rank to increase at the 209 rad/s, which is much higher than 56 rad/s, the frequency at which the belt rank was increased for Parameter Set A.

4 Discussion

4.1 Satisfaction of Model Requirements. Three criteria for models synthesized by MODA have been stated. Models synthesized by MODA will definitely meet the first two criteria, and almost certainly meet the third, provided that MODA is supported by the appropriate SSAs and submodel assembly routines. The first criterion, that the transformation of individual components into models should be physically meaningful, is satisfied provided that the SSAs provide physically meaningful relationships between component descriptions and the associated models (such as those given by Eqs. (1)–(5)). The second criterion, that the sum of the individual inertias in the system model equals the total inertia of the actual system, is also met. MODA itself does not assemble the component models into a system model; a routine that MODA will use performs this assembly. Such as assembly task is conceptually straightforward, (see, for example, [Doebelin, 80]) and should be easily implemented in an algorithm.

The third criterion, "The sum of component ranks is distributed in such a manner as to minimize (ρ_{model}) for a given sum of ranks; any increase in model rank will result in a ρ_{model} greater than specific FROI", may not be possible to prove analytically. In lieu of a proof, we note that MODA synthesizes more complex models from less complex models, and that a definite order of these models exists. MODA identifies the component—at each stage of model-rank augmentation—that results in the smallest increase in ρ_{model}, when the rank of this component is increased; this is the weak dynamic link component. As the model rank increases, different components will have this distinction. The physics of the problem suggest that the weak dynamic link component is the obvious choice to use in increasing the model rank. The first model following the rigid-body model (the rigid-body model is synthesized from a set of rank-0 components) should certainly come from the component causing the first pole (pair) in a configuration. Furthermore, as more "dynamics" are added to a model, i.e., as more complex behavior is included, we should not expect that a previously included compliance can be arbitrarily removed. Hence, we assume that a given Proper Model is the basis for a more complex model. Furthermore, as model rank is increased, the model will be able to predict modes at higher and higher frequencies. Eventually the rank will reach a point such that any further increase will result in a mode whose frequency exceeds ω_{req} being included in the model.

From a search-space perspective, the only commitment in going down one path of the search-space is an increase in the rank(s) of a given set of components. Selecting one path in the search-space does not rule out the need to increase the rank of another component at a later point. Indeed, when the compliance of a different component becomes relevant in describing the system behavior, it will be included in the model.

Further evidence of the meeting the third requirement is given by Wilson (1992), where he describes an attempt to break the algorithm. The attempt involved the creation of a number of test cases (configurations) and the use of MODA to synthesize different models of each configuration. The goal was to see if some combination of components and parameters would cause MODA to synthesize a model with non-minimum ρ_{model} for a given rank. No such model was found, adding further credibility to MODA's ability to meet the third requirement.

4.2 Potential Problems With Hill-Climbing. MODA uses a search strategy known as hill climbing or steepest-ascent. There are several well known potential problems associated with this strategy, all of which are avoided in the current context. The *plateau problem* occurs when there is a "flat area of the search space in which a whole set of neighboring states have the same value" (Rich and Knight, 1991). This problem could occur with MODA if two or more models resulting from an increase in rank of different components have the same minimum ρ_{model}. While it's unlikely that this situation will occur, it's easily circumvented. As each component is equally important to the system model, MODA could simply increase the rank of each component.

The *local maximum* problem occurs when the steepest-ascent search encounters a locally optimal solution, e.g., the search traverses a suboptimal path, leading to a suboptimal solution. MODA avoids this problem by always staying on the optimal path.

The *ridge* problem occurs when the orientation of the desired shift in the search direction is incompatible with the available choices in the search space. This problem will not occur with MODA, because increasing the rank of any unbounded-rank component is always an option and doing this always results in an increase in the model bandwidth. If a configuration consisted solely of bounded-rank components, a situation could arise in

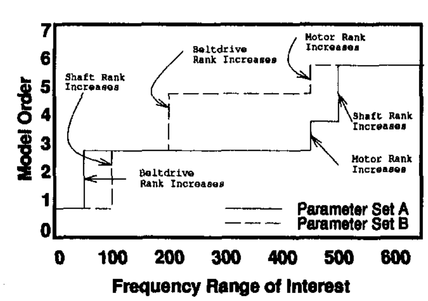

Fig. 11 Model order versus FROI for two sets of parameter values

which the model bandwidth was less than the FROI, but all components were at maximum rank. In this case, the assumption would be that the model does predict all the poles within the FROI, and no further increase in model rank is required.

Rich and Knight (1991) note that hill-climbing is "particularly unsuited to problems where the value of the heuristic function drops off suddenly as you move away from a solution." The problem of synthesizing a model of a distributed and discrete system does not have this tendency. That is, as more component dynamics (e.g., compliances) are included in the model, ρ_{model} invariably increases. In this case the ρ_{model} is the heuristic function, and this function does not drop off once a solution is reached. Indeed, a solution is only found by checking that any further increase in model rank results in a ρ_{model} beyond the FROI.

4.3 Synthesis of Alternative Models. MODA is intended to synthesize Proper Models, i.e., models that minimize the ρ_{model} for a given complexity level. MODA can be made to synthesize models meeting other criteria, by replacing the FROI condition in the algorithm (see Figure 9) with the more general condition: "Model Meet Objectives?" A description of some alternative models that MODA can synthesize follows.

Rigid-Body Model. The first step in selecting a motor for drive trains frequently requires estimating the inertia of the entire system reflected back to the motor. This can be done simply by setting the rank of each component to 0. In this manner no compliant effects are included in the model, thus the resulting model can be referred to as the rigid-body model.

First-Torsional-Resonance Model. The first torsional resonance of a system limits the closed-loop performance that can be achieved. At an early stage in the design an engineer may wish to obtain a model which predicts only this resonance and subsequently use this model to determine which component is primarily responsible for the first mode. MODA can synthesize such a model by modifying the algorithm so that it identifies the first weak-dynamic-link component, and increases the rank of only this component. The example in Section 3 illustrates this concept. In this example, the belt-drive is the weak-dynamic-link component for Parameter Set A, as shown in Fig. 11. The shaft is the weak-dynamic-link component for Parameter Set B in the same figure.

$1 - N^{th}$ Order Models. Engineers need models of varying levels of complexity to address the needs encountered at various states during the analysis. MODA is easily adapted to provide models of order 1 to N. The modifications required to synthesize a reflected inertia (first-order) model and a first-torsional-resonance (third-order) have already been addressed. At a latter design stage an engineer may want a model which has the first two or three structural modes to see where to place notch filters in a controller. MODA can be modified to create such a model by replacing the FROI condition block with the condition "Model Order $> N$?".

4.4 Pole Migration. Increases in component ranks lead to changes in system behavior as predicted by the model. An increase in the rank of an individual component adds (in general) an additional compliant element to a model. This compliance in turn adds more degrees of freedom and additional energy storage to the model, and model order increases. The additional degrees of freedom enable the model to provide a more accurate representation of the mode shapes and a more accurate estimate of the resonant frequencies. As a consequence, an increase in a component's rank causes a migration of the resonant frequencies associated with the previous (lower-order) model. As component ranks increase during the model synthesis process, the migrations of the low-frequency poles tends to diminish. As this migration may be important, an engineer using MODA may wish to increase the order of the models (by specifying a higher FROI) and examine changes in the low-frequency resonances. When model order is such that an increase in rank of any component produces little or nor shift in the fundamental resonant frequencies, an engineer will have better assurance that the model provides an accurate estimate of the resonant frequencies within the FROI. Ferris et al. (1994) explore the issue of pole migration and eigenvalue accuracy in detail. They have developed a new model deduction algorithm, EXTENDED-MODA, that explicitly accounts for accuracy by tracking pole migration.

5 Summary

The recent publication of several collections of research papers and symposia (de Kleer and Williams, 1991; Stein, 1991; Falkenhainer and Stein, 1992; Stein, 1993) on various aspects of automated modeling highlights a growing awareness and interest in this area. One of the needs of this area is the development of algorithms that are able to coordinate the model synthesis process so that engineers are able to quickly obtain models of varying levels of complexity, so that they meet different user-specified criteria. One such algorithm, MODA, is described in this paper. This algorithm, in conjunction with submodel synthesis and submodel assembly algorithms, synthesizes a Proper Model. A Proper Model model has the minimal ρ_{model} for a given model complexity level, and whose ρ_{model} will exceed a given FROI for any increase in model rank. MODA synthesizes this model by efficiently searching for the proper combination of component submodels. While the FROI model has many practical uses, minor modifications to MODA enable it to synthesize models with other applications.

The authors believe that it is worthwhile to develop an automated-modeling software program to test the implementation of the original MODA and its modified versions and to identify some of the issues developing this type of software (cf. Wilson and Stein, 1993).

Acknowledgment

The authors gratefully acknowledge the support of the National Science Foundation Grant #DMC 8657480. In addition, the authors wish to acknowledge the Manufacturing Fellowships provided by the Ford Motor Company.

References

De Kleer, J., and Williams, B. C., eds., 1991, *Artificial Intelligence: Special Volume Qualitative Reasoning about Physical Systems II,* vol. 51. North-Holland Publishing Company.

Doebelin, E. O., 1980, *System Modeling and Response,* Wiley, New York, NY.

Falkenhainer, B., and Stein, J. L., eds., 1992, *Automated Modeling, DSC-Vol. 41,* Anaheim, CA, American Society of Mechanical Engineers, The Winter Annual Meeting.

Ferris, J. B., Stein, J. L., and Bernitsas, M. M., 1994, "Development of proper models of hybrid systems," ASME WAM *Symposium on Automated Modeling,* C. J. Radcliffe, ed., *Dynamic Systems and Control 1994,* Vol. 2, Chicago, IL.

Franklin, G. F., and Powell, J. D., 1991, *Feedback Control of Dynamic Systems,* Addison-Wesley, Reading, MA.

Hogan, N., 1991, "Teaching Physical System Modeling and Modern Control Together: The Need for an Integrated Approach," Jeffrey L. Stein and Ronald C. Rosenberg, eds., *The Art of Physical System Modeling: Can It Be Taught?,* DSC-Vol. 36, New York, NY, The American Society of Mechanical Engineers, The Winter Annual Meeting.

Karnopp, D., Margolis, D., and Rosenberg, R. C., 1990, *System Dynamics: A Unified Approach,* Wiley-Interscience, New York, NY.

Rao, S. S., 1990, *Mechanical Vibrations,* Addison-Wesley Publishing Company, second edition.

Rich, E., and Knight, K., 1991, *Artificial Intelligence,* McGraw-Hill, New York.

Rinderle, J. R., and Subramaniam, B. L., 1991, "Automated Bond Graph Modeling Simplification to Support Design," Jeffrey L. Stein, ed., *Automated Modeling 1991,* DSC-Vol. 34, Atlanta, GA, American Society of Mechanical Engineers, The Winter Annual Meeting.

Rosenberg, R. C., 1989, "Modeling for the Masses: Simulation as a Soft Lab," Jeffrey L. Stein and Ronald C. Rosenberg, eds., *The Art of Physical System Modeling: Can It Be Taught?,* DSC-Vol. 36, San Francisco, CA, The American Society of Mechanical Engineers, The Winter Annual Meeting.

Stein, J. L., ed., 1991, *Automated Modeling,* DSC-Vol. 34, The American Society of Mechanical Engineers.

Stein, J. L., ed., 1993, *Automated Modeling for Design,* DSC-Vol. 47, New Orleans, LA, The American Society of Mechanical Engineers, The Winter Annual Meeting.

Stein, J. L., and Tseng, Y. T., 1991, "Strategies for Automating the Modeling Process," Jeffrey L. Stein, ed., *Automated Modeling*, DSC-Vol. 34, Atlanta, GA, The American Society of Mechanical Engineers, The Winter Annual Meeting.
Wilson, B. H., 1992, "Model-Building Assistant: An Automated Modeling Tool for Machine-Tool Drive-Trains," PhD thesis, The University of Michigan.
Wilson, B. H., and Stein, J. L., 1993, "Model-Building Assistant: An Automated Modeling Tool for Machine-Tool Drive Systems," Jose J. Granda and Francois E. Cellier, eds., *Proceedings of the 1993 SCS Western Multiconference*, San Diego, CA, Society for Computer Simulation.

APPENDIX

Dimensions and Constants for Example

Parameter Set A:

1. DC Motor: motor constant, Kt = 0.06 N-m/amp = 0.06 v/rad/s; winding resistance, Rm = 0.9 ohm; winding inductance, Lm = 0.002 henry; rotor inertia, $Jm = 3.8 \times 10^{-5}$ kg-m^2.
2. Belt Drive: diameter of pulley 1, 0.1 m; width of pulley 1, 0.01 m; density of pulley 1, 7755 kg/m^3; diameter of pulley 2, 0.2 m; width of pulley 2, 0.01 m; density of pulley 2, 7755 kg/m^3; belt stiffness, 10000 N/m.
3. Shaft: diameter, 0.0125 m; length, 1 m; shear modulus, 7.31×10^{10} N/m^2; density, 7755 kg/m^3.
4. Flywheel: diameter, 0.3 m; length, 0.025 m; density, 7755 kg/m^3.

Parameter Set B:
as above, except: Belt stiffness = 1000 N/m, shaft diameter = 0.025 m.