A Frequency Domain Model-Order-Deduction Algorithm for Linear Systems

Bruce H. Wilson
Department of MIME*
Northeastern University
Boston, MA

James H. Taylor
Department of ECE†
University of New Brunswick
Fredericton, NB CANADA

Abstract

Physical system modeling, whether automated or manual, requires a systematic procedure to choose the appropriate order of a system model. This can be most readily accomplished by determining the appropriate order of the component submodels. With previous schemes, the primary means to determine the required order of component submodels has been to focus on the eigenvalues of the model, specifically, the behavior of the eigenvalues inside of a circle in the complex plane defining a given spectral radius. In this paper we develop a new algorithm, FD-MODA, that uses changes in a model's frequency response as an algorithm-stopping-criterion. The model's frequency response provides a more comprehensive indication of model performance and adequacy than just the eigenvalues, and is more meaningful in the context of frequency-domain controller design methods.

1 Introduction

Modelers, both human and automated, must make a number of decisions when developing a mathematical model of a physical system. The ultimate onus is, of course, on the human modeler to make these decisions wisely. However, it would be highly desirable for automated-modeling software to support this decision-making process to the extent possible. Such software is primarily in the demonstration-of-concept stage, e.g. MBA, (Wilson and Stein 1993) and CAMBAS, (Stein and Louca 1995); however, one commercial program, MAX, is available (Breunese et al. 1995). For almost a decade, researchers have been investigating issues and algorithms relating to automated modeling. The most recent results can be found in the automated modeling sessions of the following conference volumes: (Radcliffe 1994, Alberts 1995, Danai 1996).

1.1 Background

Key issues that confront a modeler of a given system include the *modeling approach* and the *model complexity*. For physical systems the principal choices for the modeling approach are continuum-based models (CBMs), finite-element models (FEMs), and lumped-parameter models (LPMs). These models have wide and often overlapping application. In this paper, a *system* is defined as a collection of interconnected components, which may be passive or active or a combination thereof.

The most exact approach to modeling a distributed system, i.e. a system with spatially distributed inertia, compliance, and dissipation, is to employ a CBM expressed using a partial differential equation (PDE). When using CBMs, the external boundary conditions must be specified and inter-component compatibility conditions must be specified and maintained. To analyze a system expressed using this approach in the frequency-domain, either the normal modes of the model must be identified using a root finding algorithm

*Mechanical, Industrial, and Manufacturing Engineering
†Electrical and Computer Engineering

or the response to a sinusoidal input at frequency ω_{in} can be found by substituting $\omega = \omega_{\text{in}}$ throughout the model, removing the temporal component from the PDE, and solving the remaining spatial component of the equation. While obtaining an exact solution is often trivial for a single uniform component, it may be quite difficult to obtain for a system of disparate components. A thorough discussion of the various analysis techniques is provided in (Graff 1975).

The well known finite-element modeling method is also a viable way to model a physical system. With this method, a continuum is approximated with a collection of elements whose physical behavior (e.g. displacements and stresses) is represented by simple mathematical series such as a polynomial. As in CBMs, external boundary conditions must be specified and inter-component compatibility conditions must be met. FEMs are well suited to modeling continuous physical systems, in that a fine mesh will lead to a model whose behavior will approach that of the corresponding CBM.

Although LPMs are more approximate than both CBMs and FEMs, they are often useful for control system analysis and design purposes. As its name implies, this model requires that spatially distributed physical phenomena be aggregated or *lumped* into a finite number (often quite low) of discrete inertial, compliant, and dissipative elements. The models are generally simple to synthesize and manipulate. A common procedure for obtaining the first-order state equations (which can then be used for analysis and simulation) is to synthesize a bond graph from the lumped model. Once a bond graph is available, state equations can be synthesized either manually or with bond graph software, e.g. (Rosenberg 1996). In addition to their ease of use, a second reason for LPMs utility in a control context is that closed-loop bandwidths are generally limited by the fundamental resonance(s) in a system. Inertial and compliant elements cause these resonances, and the latter are often accurately predicted by low-order LPMs.

The selection of a modeling approach is a complex topic and beyond the scope of this paper. Insightful discussions are available in the article (Margolis 1985), chapter 11 of (Huston 1990), and chapter 10 of (Karnopp et al. 1990). For purposes of exposition and their general utility, this paper will use lumped-parameter component models in the development and demonstration of FD-MODA.

Regardless of the modeling approach, appropriate model order is a central issue in system modeling. A low-order model that is easy to analyze, simulate, and use in controller design may not provide an accurate prediction of system performance. Conversely, a high-order model that provides a good prediction of system performance may be cumbersome to use and lead to a high-order controller if certain controller design techniques are applied. The algorithm presented here is intended to assist a designer in choosing the appropriate order of component submodels that comprise a system model. As a byproduct, the algorithm will also choose the appropriate order of the system model.

In addition to the order of the model, the representation in the model of nonlinear behavior found in the real world system also affects model complexity. Although preliminary results for treating nonlinear behavior, e.g. Coulomb friction or saturation, in an automated modeling context are available (Taylor and Wilson 1995), the focus in this paper will be on linear systems.

1.2 Model-order deduction

This paper provides a new algorithm for model order *deduction*. To distinguish this process from the more established model order *reduction*, some background on the latter is useful. Model order reduction begins with a full order model, $G(s)$ of order n, and seeks a reduced order model, $G_r(s)$ of order r with $r < n$, that closely approximates the behavior of the full order model. A number of criteria exist for determining how well $G_r(s)$ approximates $G(s)$; a comprehensive review of these criteria is found in (Aguirre 1994) and references therein. Standard model order reduction approaches include the Padé method (Bultheel and van Barel 1986), the balanced realization (Pernebo and Silverman 1982), the Hankel norm approximation (Glover 1984), and selecting the dominant poles (Aguiree 1993). The H_∞-norm of the model error

$$||E(\mathrm{j}\omega)||_\infty \doteq \sup_{\omega \in R} |G(\mathrm{j}\omega) - G_r(\mathrm{j}\omega)| \tag{1}$$

is a commonly used measure to quantity the error between full and reduced order models. Both the balanced realization and the Hankel norm approximation enable bounds of (1) to be determined *a priori*.

Although model order reduction finds considerable utility and application in dynamic system analysis and control design, its use presupposes the existence of a high order model as a starting point. However, such a model, especially when the system is in the design stage, is often unavailable. In contrast, the starting point for model order deduction is a very low order model. The model order deduction process adds state variables to this model, through the inclusion of generalized inductive and capacitive elements (in the bond graph sense), until the behavior predicted by the model no longer changes appreciably as model order increases.

Existing model order deduction algorithms have focused solely on system eigenvalues,

$$\{\lambda_i \mid \det(\lambda_i I - A) = 0,\ i = 1, \ldots, n\},$$

where A is the system matrix and n is the order of the model, as a means to assess model behavior. The seminal algorithm is named *model order deduction algorithm* (MODA) (Wilson and Stein 1995). MODA systematically increases model order until any further increase in the order will cause the spectral radius (the largest Euclidean norm of the model eigenvalues) to become greater than the upper bound, $\omega_{\max}$, of some predefined frequency range of interest (FROI). In the case of a LPM, model order is increased by a finer division of compliant elements, adding compliance to gear pairs, and adding inductive lag to a DC motor. MODA selects the combination of component submodels that (i) minimizes the spectral radius and (ii) guarantees that any increase in model order will result in a spectral radius beyond $\omega_{\max}$. This implies that a model synthesized using MODA contains only eigenvalues whose Euclidean norm is less than or equal to $\omega_{\max}$. Wilson and Stein (Wilson and Stein 1995) use the term *Proper Model* to refer to a model with physically based parameters and state variables and that meets this model performance criterion.

A limitation of MODA is the lack of a guarantee regarding the accuracy of the eigenvalues in a model synthesized using this algorithm (Ferris and Stein 1995). It is well known that different lumpings of continuous components in a larger system model will result in different eigenvalues. In the case of MODA, an increase in the order of a Proper Model generally results in some shifting of $|\lambda_i| \in [0, \omega_{\max}]$, which is understandable, as the location of the low frequency eigenvalues of a model will approach those of a continuum-based model as more degrees of freedom are added to the model (Huston 1990, p. 334-337). This suggests that a more accurate estimation of the true location of the eigenvalues can be obtained by increasing the order of the Proper Model. Ferris and Stein (Ferris and Stein 1995) address the issue of eigenvalue accuracy in a model synthesis algorithm that monitors the migration of the $|\lambda_i| \in [0, \omega_{\max}]$, which they refer to as the *critical system eigenvalues*. Their algorithm, EXTENDED-MODA, synthesizes a Proper Model in the same manner as MODA; it then continues increasing model order until the critical system eigenvalues remain in approximately the same location as the model order is increased further. The degree of approximation can be controlled by a user-specified tolerance defining the acceptable percentage change in the critical system eigenvalues. The claim by Ferris and Stein is that EXTENDED-MODA will synthesize a model of appropriate complexity that provides estimates of $|\lambda_i| \in [0, \omega_{\max}]$ that have converged (presumably to the location predicted by a CBM) within some user-specified percentage. The algorithm IO-MODA (Walker et al. 1996) refines EXTENDED-MODA by using controllability and observability criteria to pare the quantity of models examined. Other than this extension, the algorithms both employ the same convergence criteria.

As the discussion in the last two paragraphs indicates, existing deduction algorithms use eigenvalues whose Euclidean norm is less than a given frequency and their convergence as a means of evaluating the accuracy of a model. At best, this criteria provides a limited means to evaluate a model; at worst, it may lead to completely false confidence in the response predicted by a model. The criteria is limited in that the frequency response of a model, a well known analysis and design tool, is a function of both the poles (i.e. the eigenvalues) and the zeros of a model. By ignoring the zeros during model synthesis, potentially useful information on model performance is being ignored.

As for the issue of false confidence, consider the two models of a flywheel-shaft-flywheel system shown in Fig. 1.

Insert Fig. 1 here.

The driving-point admittance transfer function of the first model equals $1/(J_1 + J_2)s$, and the admittance of the second model equals

$$\frac{\omega(s)}{T(s)} = \frac{J_2 s^2 + K}{s(J_1 J_2 s^2 + K(J_1 + J_2))}. \tag{2}$$

Let us assume that the FROI is 0–9 rad/sec, and that the model parameters are $J_1 = J_2 = 1$ and $K = 50$. The rigid body model in Fig. 1, the initial model used by the algorithm discussed above, has a single eigenvalue at the origin. The frequency response predicted by such a model would be a straight line extending to $\omega = \infty$, with a slope of -20 dB/decade. The second model has eigenvalues at the origin and at $\pm 10\mathrm{j}$; it also has a zero at $\pm\sqrt{50}\mathrm{j}$. The frequency response of this model would reflect the system's anti-resonance and resonance pair.

All the deduction algorithms discussed above would determine that the first model in Fig. 1 is adequate, as subsequent increases in model order would result in a spectral radius beyond $\omega_{\max}$ (MODA) and the single eigenvalue within the FROI has obviously converged (EXTENDED-MODA, IO-MODA). Yet the use of the frequency response obtained from the first model to design a controller with a crossover frequency even at 0.1 rad/sec would likely lead to an unstable closed-loop system. The frequency response of the second model would likely lead to a successful controller design effort. While this example may appear somewhat contrived, it does indeed portray the behavior of existing deduction algorithms and highlights the need for a more comprehensive indication of model performance than just the eigenvalues.

1.3 Approach

The example in the previous section demonstrates the need for a model performance metric that is more comprehensive than model eigenvalues. This paper proposes the use of the frequency response (specifically, changes in the frequency response) as a model performance metric. The frequency response of a model is obtained by substituting $s = \mathrm{j}\omega$ into the model transfer function and evaluating the resulting rational function over a range of frequencies, i.e.

$$G(\mathrm{j}\omega) = K\frac{\prod_{i=1}^{m}(\mathrm{j}\omega + z_i)}{\prod_{i=1}^{n}(\mathrm{j}\omega + p_i)}, \tag{3}$$

where K is a constant, $-z_i$ are the zeros, $-p_i$ are the poles, and $m \leq n$.

In the context of the deduction algorithms described in the previous section, comparing the frequency response (3) over a FROI $[\omega_{\min}, \omega_{\max}]$ provides a more meaningful basis for evaluating model performance than the eigenvalues. Using eigenvalue convergence as a criterion for setting model order may have little bearing on the convergence of the frequency response. As the example above shows, an eigenvalue just beyond the $\omega_{\max}$ will not be considered while evaluating eigenvalue convergence, but may have considerable bearing on the frequency response within the FROI. Furthermore, in the context of control system analysis and design, the frequency response provides a more comprehensive prediction than eigenvalues in measures such a gain and phase margin, loop-shaping requirements (for compensator design), and the like. In the final analysis, we want the system model to provide a reliable prediction of system performance for a given FROI. For this to occur, both the zeros and the poles of the transfer function must have converged sufficiently such that increases in model order do not cause appreciable changes in the frequency response. Approaches based solely on eigenvalues may not achieve this convergence.

1.4 Overview of paper

The deduction algorithm developed in this paper coordinates the synthesis of a set of component submodels with the property that subsequent increases in the order of the submodels will not cause the maximum relative change in a model's frequency response over a given FROI to be greater than a specified tolerance. The approach to synthesizing such a model entails a systematic selection of component (passive and active) submodels, using changes in the frequency response both as a means of selecting intermediate models and as an *algorithm stopping criterion* in the frequency-domain.

This paper is organized as follows. Section 2 presents the modeling framework for the new deduction algorithm. Section 3 develops the algorithm, and an example of algorithm operation is provided in Sect. 4. The final section includes a summary and discussions of model validation and extensions.

2 Modeling Framework

The context of the present research in model-order deduction is as follows. We assume at the outset that the modeler is dealing with a physical system that is assembled using a combination of zero or more passive components and zero or more active components that have their own dynamic responses. We also assume that a modeling environment exists that can be used to generate a system model and frequency responses once the components are specified. This technique can produce a number of dynamic models, which differ in the level of detail used to model each component. In the linear case, the *level of detail* of each component submodel translates into its order; e.g., a more detailed motor model may include inductive lag (one additional state variable), or a more detailed shaft model might include one or more compliant elements. The problem addressed is deciding what *minimal* level of detail to use in each component submodel so that the response of the resulting system model no longer changes appreciably with increasing model order.

The model order deduction algorithm described in this paper, FD-MODA, is built within a framework of components, model synthesis procedures, and model evaluation techniques. To define the framework for this algorithm, we begin by specifying the set of components

$$C = \{\mathrm{c}_1, \mathrm{c}_2, \ldots, \mathrm{c}_n\}$$

from which systems can be constructed. In the domain of electro-mechanical systems, C will include components such as DC motors, rotational shafts, and gear pairs. In addition, each component c_i has a corresponding *model generating function* (MGF) that maps the component into one of its submodels. The set of MGFs for C is

$$F = \{\mathrm{f}_1, \mathrm{f}_2, \ldots, \mathrm{f}_n\}.$$

Each f_i is capable of synthesizing one or more candidate submodels of the component c_i to which it corresponds. A non-negative integer, referred to as the *rank* of a component submodel, is used to specify its complexity. Depending on the component, the maximum rank is either unbounded or bounded. *Unbounded rank* components can be mapped into an infinite number of submodels, whereas *bounded rank* components can be mapped into only a finite (typically small) number of submodels. A flexible shaft is an example of the former case (as it can be infinitely divided), while a motor would generally be of bounded rank (as in the context of typical control engineering, it has only two models available).

The set of submodels corresponding to the i^{th} component is denoted as:

$$M_i = \{\mathrm{m}_i^{(0)}, \mathrm{m}_i^{(1)}, \ldots, \mathrm{m}_i^{(j)}\},$$

where the superscript of each m_i corresponds to the rank of the submodel. Without loss of generality, we assume the submodels in M_i are ordered by increasing rank, which corresponds to increased model order. Note that this set can also include active elements such as amplifiers and controllers with their own dynamic responses. For example, a rank zero amplifier model would be a DC gain, and a rank one model would include some rolloff with increasing frequency. The submodels, be they derived by parameter lumping or other means, are assumed to be valid. Furthermore, higher order submodel are assumed to provide a more accurate prediction of submodel behavior than a lower order submodel; in the limit as the rank approaches infinity, the predictions provided by the submodels approach those provided by CBMs.

Finally, we would like to emphasize that models synthesized using FD-MODA have *physical parameters.* The state variables are generalized displacements and momenta, and inertial, compliant, and dissipative elements appear as coefficients of these state variables. This is in contrast to a system identification approach to modeling, in which information internal to a physical system is generally not part of the model and in which abstract coefficients serve as the model parameters.

The framework of FD-MODA also includes a means to assemble the component submodels into a system model and to synthesize state equations. Submodel assembly of mechanical components is accomplished by kinematically coupling adjacent inertial elements of the component submodels and summing two or more inertias into a single inertia. This is standard, and is discussed in texts such as (Karnopp et al. 1990, Rosenberg and Karnopp 1983). In these same texts, algorithms are provided for synthesizing the state equations of a model expressed as a bond graph. An insightful discussion on the use of bond graphs in the context of model order deduction is available in (Ferris and Stein 1995), where it is noted that bond graphs allow the effect of increasing a component's rank to be visualized more readily than a solely equation-based modeling formulation. For future reference, we denote such a system model as:

$$G^R(s) = A(\mathrm{m}_i^{(r_i)})\,; \;\; R = \sum_i r_i$$

where A informally denotes "assembly" in the above sense, i ranges over the set of components and R denotes the rank of the model, which is related to its order.

The FROI is of considerable importance in systems engineering and provides a context within which to formulate requirements on model performance. The FROI is the frequency band $[\omega_{\min}, \omega_{\max}]$ over which a model, in terms of steady-state input-output prediction, should give a reliable indication of system response. Zero, or a small positive number, is often the lower bound of the FROI. The upper bound may be determined from an input specification, e.g., if the frequency content of commanded inputs or disturbance inputs is known then the model should be accurate to frequencies 2–5 times the highest input frequency (ω_{in}) (Karnopp et al. 1990); accordingly, $\omega_{max} = 5 \times \omega_{in}$. In the case of closed-loop system design, the open-loop crossover frequency drives the FROI in that the model should provide a reliable response at frequencies 1 to 2 decades beyond the crossover frequency ω_{co}, that is, $10 \times \omega_{co} \leq \omega_{max} \leq 100 \times \omega_{co}$. Both of these approaches are merely heuristics; the engineer must temper these suggestions with knowledge of the particulars of the problem under consideration and be prepared to revise them as the overall design process proceeds.

Returning to the topic of the relative change in a model's frequency response, we will adopt a metric similar to that employed in model order reduction. The metrics will differ in that model order deduction will employ a normalized relative comparison of model performance. The current model will be deemed sufficiently accurate when

$$\delta G_n^{R_m} \doteq \max_{\omega \in \text{FROI}} \left| \frac{G^{R+m}(\mathrm{j}\omega)}{G^R(\mathrm{j}\omega)} - 1 \right| < \text{TOL}, \tag{4}$$

where TOL is a given convergence tolerance, $G^R(\mathrm{j}\omega)$ refers to the frequency response of the current model, and $G^{R+m}(\mathrm{j}\omega)$ denotes the frequency response of a model with all m component ranks (where possible) increased by unity. Furthermore, R should be a minimum for a given set of components, FROI, and TOL. Equivalently, we seek a set of component ranks

$$\{r_1,\ r_2,\ \ldots,\ r_m\},$$

where m is the quantity of components in the system, such that $R = \sum r_i$ is minimized while meeting the condition in (4). This alters the Proper Model definition provided in (Wilson and Stein 1995). Here a Proper Model is defined as the minimal-order, lumped, physical model that adequately represents the response of a configuration of passive and active components to excitations in the FROI $[\omega_{\min}, \omega_{\max}]$. FD-MODA is intended to choose component submodel ranks that will result in a model with these properties.

3 Frequency-Domain Deduction Algorithm

The deduction algorithm developed in this paper uses the change in the normalized frequency response as a termination condition. Once the model has an order such that subsequent normalized changes in the frequency response (with increasing model order) are less than a given tolerance, the algorithm stops increasing model order. The use of the frequency response distinguishes the new algorithm from previous deduction algorithms, which consider convergence of the eigenvalues only. This section discusses the requirements of the algorithm in terms of input and output data, the choice of a suitable search strategy for the algorithm, and the information flow in the new algorithm.

The input and the desired output specify the requirements of the algorithm. The input to FD-MODA includes:

1. a system, $\mathcal{S}$, of serially-connected components, e.g. $\mathcal{S} = \{c_1, c_3, c_2, c_8\}$
2. a FROI, $[\omega_{\min}, \omega_{\max}]$, and a set of frequencies, Ω, discretized over this range
3. a frequency response convergence tolerance, TOL.

We assume the following are available: (i) a set of model generating functions that synthesize the submodels of the components in $\mathcal{S}$, (ii) a routine that assembles the submodels in a system model, and (iii) analysis software that calculates the frequency response of the system model.

The output of FD-MODA is the set of ranks of the components of $\mathcal{S}$, e.g. $\mathcal{R} = \{r_1, r_3, r_2, r_8\}$. This set must satisfy two conditions:

1. Increasing all r_i will not cause a significant change in the normalized frequency response over the FROI.
2. The sum of the ranks, $\sum_i r_i$, is minimum.

This provides a Proper Model of the system, i.e. a model of appropriate complexity given the specification that the frequency response should be accurate over the FROI.

3.1 Search strategy

The search strategy employed by FD-MODA is characterized as a *greedy*, or modified *hill-climbing* search (Rowe 1988). The initial *state* of the algorithm is a set of zero-rank components submodels. The final, or goal, state of the algorithm is the set of ranks with the minimum sum that meets the condition in (4). The algorithm employs inner and outer iterative loops. The inner loop determines which rank increase causes the largest normalized change in frequency response over the FROI. The component whose rank increase causes this change is referred to as the *most sensitive component* for a particular iteration of the outer loop. The outer loop increases the rank of this component on a trial basis and evaluates the need to keep the rank increased. By searching for the component whose rank increase causes the *maximum* normalized change in the frequency response, which can be assumed to lead to the biggest improvement in model fidelity, we can characterize FD-MODA's search strategy as a greedy search. This strategy modifies the usual hill-climbing search, in that only one variable (rank) is changed at each step where that single variable is selected to maximize the normalized change in the frequency response, as opposed to the conventional multi-variable hill-climbing approach.

The greedy search is adopted for several reasons. First, in the current context this search will always find *a* set of component ranks that meet the desired error criterion. As increasing component ranks adds more degrees of freedom to a model, the accuracy with which the model predicts a frequency response improves. Eventually, sufficient degrees of freedom will be included in the model so that the error criterion is met. While a non-optimal local minimum is an issue in many applications of the greedy search, it is not an issue for FD-MODA. When using this strategy to make decisions in a search *tree*, choosing a certain *branch* normally means that other branches are no longer available. In the FD-MODA context, however, choosing a branch implies increasing the rank of a given component. A rank increase in one component does not preclude subsequent rank increase of another component. Increasing a component's rank does imply a commitment in that any increase in a component's rank is permanent, but this is independent of the local minimum issue.

A second reason for choosing this search strategy is that the greedy search is suited to the problem of selecting which degrees of freedom to add to a model. Consider a torsional system with three identical inertias separated by two shafts: one which is very stiff and the other which is very flexible. The shafts provide the only means of adding degrees of freedom to the model of this system. For any realistic FROI, adding the compliance of the more flexible shaft to the model will have a larger effect on the frequency response than would adding the compliance of the stiffer shaft. Given the choice between which compliance to add, i.e. which component's rank should be increased, it is clearly desirable to add the compliance of the more flexible shaft as this will provide a more realistic description of the system behavior. This logic can be generalized to any number of components: At any state of the algorithm, the component that causes that largest change in the system behavior (as measured by the maximum change in the frequency response) when its rank is increased should be used to add more complexity to the model. The greedy search will lead to this behavior.

A final reason for adopting the greedy search strategy is that it requires fewer models to be evaluated than a strategy such as *breadth-first* (Rowe 1988). As discussed in (Wilson and Stein 1995), a greedy strategy requires that the following quantity of models be evaluated

$$1 + N(R + 1), \tag{5}$$

where N is the number of components and R is the final rank of the model. In contrast, an exhaustive breadth-first strategy introduces a combinatoric explosion, requiring that

$$\sum_{i=0}^{R+1} \frac{(N + i - 1)!}{i!(N - 1)!} \tag{6}$$

models be synthesized and evaluated. To identify a Proper Model, the greedy strategy will obviously require fewer models to be evaluated than a breadth-first strategy.

3.2 Information flow

The overall information flow of the deduction algorithm is shown in Fig. 2.

Insert Fig. 2 here.

As the flowchart illustrates, the deduction algorithm is a multi-loop iterative process of testing the effect on model performance of increasing the rank of the most sensitive component. Note that the component with this distinction will normally change during each iteration. A change in model performance is deemed *significant* when

$$\delta G_n^{R_1} \doteq \max_{\omega \in \text{FROI}} \left| \frac{G^{R+1}(\mathrm{j}\omega)}{G^{R}(\mathrm{j}\omega)} - 1 \right| \geq \text{TOL}, \tag{7}$$

where $G^{R+1}(\mathrm{j}\omega)$ corresponds to increasing the rank of the most sensitive component. When a change in model performance is significant, the more complex submodel corresponding the most sensitive component is retained in the model, and the iterative process repeats. Should the change in model performance be insignificant, i.e. the LHS of (7) is less than TOL, an additional check on model performance is performed. The ranks of the remaining components are increased and the condition in (4) is checked. The reason for this check is that normalized changes in model performance may not necessarily be monotonically decreasing with increasing model rank, and a more thorough check on model convergence is necessary. (This lack of monotonicity is demonstrated in the example in Sect. 4.) Should (4) hold, i.e. the change in model performance is insignificant after increasing the remaining ranks, all ranks are decreased and the algorithm stops. Should (4) not hold, all ranks are decreased except that of the most sensitive component and the algorithm continues searching for the new most sensitive component. This extra step adds one additional model to the quantity determined by (5).

Determining the most sensitive component requires a separate iterative procedure. The process for this step involves increasing the rank of each component on an individual basis and determining the component that causes the largest normalized change in the frequency response over the FROI. The flowchart for this process is shown in Fig. 3.

Insert Fig. 3 here.

Note that the quantity δG in the figure is determined by

$$\delta G = \max_{\omega \in \Omega} \left| \frac{G_{\mathrm{new}}(\mathrm{j}\omega)}{G_{\mathrm{current}}(\mathrm{j}\omega)} - 1 \right| \tag{8}$$

where $G_{\mathrm{new}}(\mathrm{j}\omega)$ and $G_{\mathrm{current}}(\mathrm{j}\omega)$ correspond to the frequency responses of the new (rank increased) and current models.

4 Algorithm Demonstration

The operation of FD-MODA is demonstrated in this section. First, a physical system and component submodels are described. This is followed by the synthesis of a system model with a convergence tolerance of 0.1.

4.1 Physical system and submodels

We will demonstrate FD-MODA with an example that involves synthesizing a model of a DC motor driving a chain of five flywheels separated by four torsional shafts. This system is illustrated in Fig. 4.

Insert Fig. 4 here.

The parameters for this system are given in the appendix.

The component models for the motor, flywheels, and torsional shafts are as follows:

DC Motor Two models are available for this component. The rank-0 model assumes no inductive lag in the armature windings, whereas the rank-1 model includes this lag. A bond graph illustrating these models is shown in Fig. 5.

Insert Fig. 5 here.

The symbols in the figure are defined as: L_{a} – armature winding inductance, R_{a} – armature winding resistance, J_{a} – armature torsional inertia, K_{t} – torque constant, and B_{a} – armature viscous friction coefficient.

Flywheel Only a single model of rank-0 is available for a flywheel, a rotational inertia. The bond graph for an inertia consists of a 1-junction with an inertial port. This model is standard (Rosenberg and Karnopp 1983). We also assume that each flywheel in Fig. 4 will have some viscous friction due to the bearings that support it. For convenience, the viscous friction coefficients are scaled so that the resulting time constant of the flywheel in free rotation is 10 seconds.

Torsional Shaft An infinite quantity of models is available for this component. The rank-0 model assumes no compliance, the rank-1 model assumes a single compliance, and so on. For this example, torsional damping in the shaft has been ignored.

The configuration in Fig. 4 has five components whose rank can be increased. The notation $ijklm$ will be used to denote the particular combination of ranks that correspond to a given model. For example, the combination 01001 implies three rank-zero components (the DC motor, the second shaft, and the third shaft) and two rank-one components (the first and fourth shafts). This translates into a system model with compliant elements for the first and fourth shafts and no inductive lag.

4.2 Model synthesis

For this example, the range 0.1–100 rad/sec defines the FROI and the convergence tolerance is 0.1. We begin with the 00000 model and search for the most sensitive component of this model, i.e. the component whose rank increase will cause the largest change in

$$\delta G_n^R = \max_{\omega \in \text{FROI}} \left| \frac{G^{R+1}(\mathrm{j}\omega)}{G^R(\mathrm{j}\omega)} - 1 \right| \tag{9}$$

for $R = 0$. A plot of (9) as a function of increasing the ranks of the various components is shown in Fig. 6, where 00000 is the initial set of ranks.

Insert Fig. 6 here.

The maximum change shown in the figure, 2.957, is greater than the convergence tolerance. This change was caused by increasing the rank of the fourth shaft, which is the fifth "rank-increasable" component. Thus the 00001 set of ranks produces the initial model for the second iteration of FD-MODA.

Six iterations are required to synthesize a system model that has converged. Initially, five iterations are required before the LHS of (7) is less than TOL. A sixth iteration corresponds to the final check in (4). To illustrate the incremental change in the normalized frequency response during the model synthesis process, a plot of (9) as a function of algorithm iteration number and the total rank is shown in Fig. 7. It is interesting to note that the plot in Fig. 7 does not demonstrate monotonicity. That is, the maximum incremental change in the normalized frequency response as model order increases does not necessarily decrease with increasing model order. Commentary on this topic will be provided in Sect. 5.

Insert Fig. 7 here.

We will summarize the model synthesis process for this example. Beginning with the 00000 set of ranks, the following rank sets (and corresponding model changes) were identified in synthesizing a Proper Model of the system in Fig. 4:

1. 00001 (add compliance of fourth shaft)
2. 00101 (add compliance of second shaft)
3. 00111 (add compliance of third shaft)
4. 01111 (add compliance of first shaft)
5. 11111 (add DC motor armature inductance)
6. 12222 (augment all possible ranks, ranks subsequently decreased)

Note that the last set of rank increases in the above list was needed to check model convergence. These rank increases were not actually incorporated into the final Proper Model. The frequency responses corresponding to this list of models are shown in Fig. 8. These plots illustrate the apparent convergence of the response as model order is increased using FD-MODA.

Insert Fig. 8 here.

5 Discussion

This paper develops an order deduction algorithm for linear systems. The algorithm begins with a very low order model of a system and uses a greedy search strategy to identify which submodel's order is to be increased and substituted into the overall model. The algorithm, FD-MODA, uses the normalized change in the frequency response over some specified frequency range of interest as the criterion for determining which higher order component submodel will have the most effect on model performance. If the maximum normalized change, as determined by testing all components, is greater than some convergence tolerance, the order of the appropriate submodel is increased and the model order deduction process continues. The process terminates when the maximum normalized change in the frequency response with all ranks increased is less than the convergence tolerance.

The frequency-domain performance criterion is comprehensive in that it reflects the input/output behavior of the entire system, not just the behavior of its poles. This observation is particularly important in the context of modeling plants for control system design. Note that this criterion can also be used for closed-loop systems, since a change in closed-loop frequency response due to an increase in model order can be evaluated as easily as open-loop frequency response. Finally, the frequency response reflects system performance by concisely depicting input-output behavior over $\omega \in [\omega_{min}, \omega_{max}]$. Engineers have found this type of information meaningful for over half a century and continue to employ it in modern techniques such as H_∞-based design.

The algorithm improves upon existing algorithms in that it considers normalized changes in the frequency response as the algorithm-stopping-criterion, as opposed to the (assumed) convergence of the model eigenvalues. As shown in Sect. 1.2, using the eigenvalues as the sole measure of model performance can lead to an erroneous assessment of system performance. In the remainder of this section, the validity of the models synthesized using FD-MODA, the generality of FD-MODA, and algorithm extensions will be discussed.

5.1 Model validation

The deduction algorithm developed in this paper, when used either manually or as a module of an automated modeling software program, will produce a model that is assumed to have converged to a given tolerance. This presupposes that the parameter lumping strategy being employed is acceptable. Let us examine the validity of the convergence assumption. Although the engineering community in general and the control engineering community in particular relies extensively on mathematical models of physical phenomena, absolute validation of these models is impossible without the actual hardware in place. In the design stage, when models are used extensively, the hardware is *not* be available. Lacking the hardware, the alternatives for checking a given lumped-parameter model include synthesizing and analyzing either a continuum-based model or a finite-element model. Although not demonstrated here, either of these techniques can indeed provide valuable information regarding the performance of a lumped-parameter model, whether synthesized using FD-MODA or otherwise. For the designer who wishes to use a lumped-parameter model for controller synthesis and requires a model with guaranteed accuracy, either of these modeling techniques can be quite useful. However, the designer must weigh the additional information against the effort required to obtain it. Particularly in the case of a continuum-based model, obtaining a solution can be very tedious. Although it is known that lumped-parameter models of simple continuous systems will converge to their continuum-based equivalents (Huston 1990, p. 334–337), a guarantee on the absolute error bounds of a general lumped-parameter model, including one synthesized using FD-MODA, cannot be provided. Measured data, a continuum-based model, or a finite-element model can all be used to help estimate these bounds.

As discussed in Sect. 2, FD-MODA is intended to synthesize a Proper Model, i.e. a minimum-order model that has convergence to a final frequency response within some tolerance. As FD-MODA uses the relative change in a model's frequency response as an algorithm-stopping-criterion, its models cannot be guaranteed to meet the convergence criterion. We thus restrict our claim regarding the performance of FD-MODA to the following:

> FD-MODA provides an effective heuristic for coordinating the synthesis of a model that has converged, relative to other lower-rank models of the same system, to a user-specified tolerance over a user-specified frequency range $[\omega_{\min}, \omega_{\max}]$.

The density of the frequency grid used during model synthesis is an important consideration. The spacing between adjacent grid points must be sufficiently close so that an accurate estimation of a model's frequency response results. The spacing is especially critical when the model has lightly damped poles and zeros. The example in Sect. 4 uses 200 points, logarithmically-spaced over three decades. While this spacing is sufficient for the example, some applications may require a finer grid.

5.2 Generality of algorithm

The procedure provided in this paper is applicable to systems other than electromechanical systems. The approach is suited to physical phenomena that can be characterized by distributed-parameter behavior or with parasitic effects similar to DC motor lag that may or may not need to be included in the model to achieve the desired level of accuracy. In either case, there are phenomena that may be approximated by finite-dimensional models whose order can be increased to better approximate the original dynamics. For example, model synthesis for pneumatic, hydraulic or electro-magnetic (e.g. wave-guide) phenomena can also be coordinated using FD-MODA, as can systems with parasitic capacitances.

Although demonstrated using a single-input / single-output system, FD-MODA can also be applied to a multiple-input / multiple-output model. This may be done by studying the effect of component rank increases on each input / output variable pair's transfer function and selecting the most sensitive component based on the maximum change in $\delta G_n^R(j\omega)$ (7) over the input / output pairs. In this way, *any* input / output pair that results in some component rank being increased for a given convergence tolerance will result in this rank increase being incorporated in the final model.

FD-MODA can also be used in a finite-element or finite mode bond graph (Karnopp et al. 1990, ch. 10) context. In the former, FD-MODA would provide a finer mesh for the most sensitive components until the change in the model's frequency response is less than some tolerance. With finite mode bond graphs, the algorithm would include additional modes for the most sensitive components, in an analogous manner, until the frequency response settled.

5.3 Extensions

A more thorough analysis regarding the accuracy of a lumped-parameter model against its continuum-based equivalent would improve the understanding of the former. As stated above, results regarding model accuracy are available for single elements and model accuracy is a major topic in the finite-element literature. The accuracy of LPMs does not appear to have received the same attention. In previous work with deduction algorithms, it was assumed (and observed) that the early rank increases caused the largest changes in a model and that the frequency response soon converged to its (assumed) final shape. The algorithm demonstration in Sect. 4, in particular the plot in Fig. 7, shows that model change is *not* monotonically decreasing with increasing model rank. This lack of monotonicity and the ubiquity of simple lumped-parameter models in the application of controller design suggests that an estimation of the error bounds associated with a given lumped-parameter model is needed and would be useful to a broad audience.

The utility of FD-MODA can be broadened by extending it to encompass nonlinear systems and other complicating factors. For nonlinear systems, describing-function methods provide a direct approach to achieving this goal, and frequency-domain performance criteria are even more appropriate than in the linear case. Preliminary research in extending MODA for nonlinear systems is described in (Taylor and Wilson 1995). Finally, since RHP poles and zeros may severely limit the achievable performance of a system (Middleton 1991), any rank increase that produces a RHP pole or zero should automatically be kept in the model. As such poles and zeros were not an issue in the example provided in Sect. 4, this criteria was not included previously. However, feedback can introduce RHP poles; thus when feedback is present, FD-MODA should be modified as described here.

References

Aguiree, L. A. (1993), Quantitative measure of modal dominance for continuous systems, *in* '32nd IEEE Conference on Decision and Control', San Antonio, TX, pp. 2405–2410.

Aguirre, L. A. (1994), 'Computer-aided analysis and design of control systems using model approximation techniques', *Computer methods in applied mechanics and engineering* **114**, 273–294.

Alberts, T. E., ed. (1995), *Proceedings of the ASME Dynamic Systems and Control Division*, Vol. 1, American Society of Mechanical Engineers, San Francisco, CA. The International Mechanical Engineering Congress and Exposition.

Breunese, A. P. J., de Vries, T. J. A., van Amerongen, J. and Breedveld, P. C. (1995), Maximizing impact of automation on modeling and design, *in* T. E. Alberts, ed., 'Proceedings of the ASME Dynamic Systems and Control Division', Vol. 57-1, American Society of Mechanical Engineers, San Francisco, CA, pp. 421–430. The International Mechanical Engineering Congress and Exposition.

Bultheel, A. and van Barel, M. (1986), 'Padé techniques for model reduction in linear system theory: a survey', *J. Computational and Applied Mathematics* **14**, 401–438.

Danai, K., ed. (1996), *Proceedings of the ASME Dynamic Systems and Control Division*, Vol. DSC-Vol. 58 of *1996 ASME International Mechanical Engineering Congress and Exposition*, Atlanta, GA.

Ferris, J. B. and Stein, J. L. (1995), Development of proper models of hybrid systems: A bond graph formulation, *in* F. Cellier and J. J. Granda, eds, 'International Conference on Bond Graph Modeling and Simulation', Simulation Series, Volume 27, Number 1, The Society for Computer Simulation, Las Vegas, NV, pp. 43–48. Proceedings of the 1995 Western Multiconference.

Glover, K. (1984), 'All optimal Hankel-norm approximations of linear multivariable systems and their L-∞ error bounds', *Int. J. Control* **39**, 1115–1193.

Graff, K. F. (1975), *Wave motion in elastic solids*, Ohio State University Press.

Huston, R. L. (1990), *Multibody Dynamics*, Butterworth-Heinemann, Boston, MA.

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

Margolis, D. L. (1985), 'A survey of bond graph modeling for interacting lumped and distributed systems', *Journal of the Franklin Institute* **319**(1/2), 125–135.

Middleton, R. H. (1991), 'Trade-offs in linear control system design', *Automatica* **27**, 281–292.

Pernebo, L. and Silverman, L. M. (1982), 'Model reduction via balanced state space realization', *IEEE Transactions on Automatic Control* **27**, 382–387.

Radcliffe, C. J., ed. (1994), *Dynamic Systems and Control 1994 Volume 2*, American Society of Mechanical Engineers, Chicago, IL. The International Mechanical Engineering Congress and Exposition,.

Rosenberg, R. (1996), *The Enport User's Manual*, v. 5.4 edn, Rosencode Associates, Inc., Okemos, MI 48864 USA.

Rosenberg, R. C. and Karnopp, D. (1983), *Introduction to Physical System Dynamics*, McGraw-Hill, New York, NY.

Rowe, N. C. (1988), *Artificial Intelligence through Prolog*, Prentice Hall.

Stein, J. L. and Louca, L. S. (1995), A component-based modeling approach for system design: Theory and implementation, *in* F. Cellier and J. J. Granda, eds, 'International Conference on Bond Graph Modeling and Simulation', Simulation Series, Volume 27, Number 1, The Society for Computer Simulation, Las Vegas, NV, pp. 109–115. Proceedings of the 1995 Western Multiconference.

Taylor, J. H. and Wilson, B. H. (1995), A frequency domain model-order-deduction algorithm for nonlinear systems, *in* 'IEEE Conference on Control Applications', Albany, NY, pp. 1053–1058.

Walker, D. G., Stein, J. L. and Ulsoy, A. G. (1996), Input-output criterion for linear model deduction, *in* K. Danai, ed., 'Proceedings of the ASME Dynamic Systems and Control Division', Atlanta, GA, pp. 673–681. 1996 ASME International Mechanical Engineering Congress and Exposition.

Wilson, B. H. and Stein, J. L. (1993), Model-building assistant: An automated modeling tool for machine-tool drive systems, *in* J. J. Granda and F. E. Cellier, eds, 'Proceedings of the 1993 SCS Western Multiconference', Simulation Series, Volume 25, Number 2, Society for Computer Simulation, San Diego, CA, pp. 169–178.

Wilson, B. H. and Stein, J. L. (1995), 'An algorithm for obtaining minimum-order models of distributed and discrete systems', *Journal of Dynamic Systems, Measurement, and Control* **117**(4), 534–540.

Appendix: Component Dimensions and Material Constants for Example in Sect. 4

Standard SI units are used; units are only given in the first usage.

DC Motor : motor constant $K_{\mathrm{t}} = 0.06$ n-m/amp = 0.06 v/rad/sec, armature resistance $R_{\mathrm{a}} = 0.9$ ohms, armature inductance $L_{\mathrm{a}} = 0.002$ Henry, rotor torsional inertia $J_{\mathrm{a}} = 3.8\text{e-}5$ kg-m^2

First flywheel : $J_1 = 4.937\text{e-}4$, viscous friction coefficient $B_1 = (J_{\mathrm{a}} + J_1)/10$ n-m/rad/sec

First shaft : $J_{S_1} = 6.053\text{e-}6$, stiffness $K_{S_1} = 612.5$ n-m/rad

Second flywheel : $J_2 = 0.0046$, $B_2 = J_2/10$

Second shaft : $J_{S_2} = 3.783\text{e-}7$, $K_{S_2} = 38.28$

Third flywheel : $J_3 = 0.0046$, $B_3 = J_3/10$

Third shaft : $J_{S_3} = 3.783\text{e-}7$, $K_{S_3} = 38.28$

Fourth flywheel : $J_4 = 7.748\text{e-}4$, $B_4 = J_4/10$

Fourth shaft : $J_{S_4} = 2.364\text{e-}8$, $K_{S_4} = 2.393$

Fifth flywheel : $J_5 = 4.937\text{e-}4$, $B_5 = J_5/10$

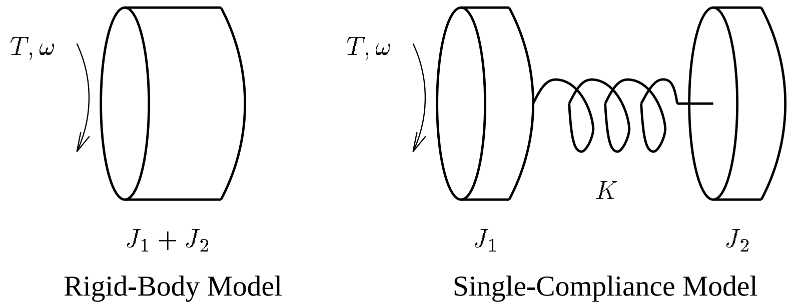

Figure 1: Models of a flywheel-shaft-flywheel system

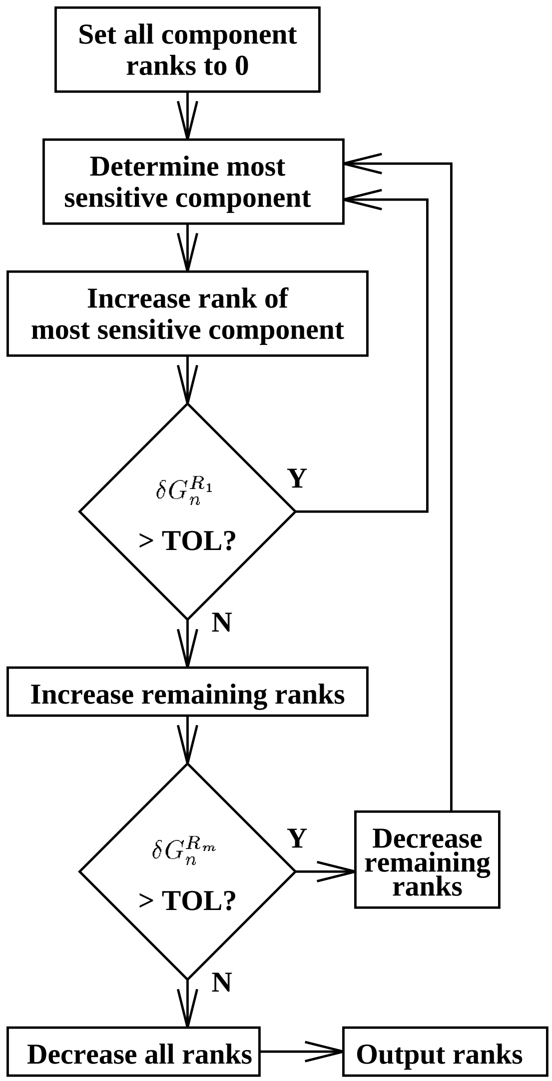

Figure 2: Flowchart for deduction algorithm

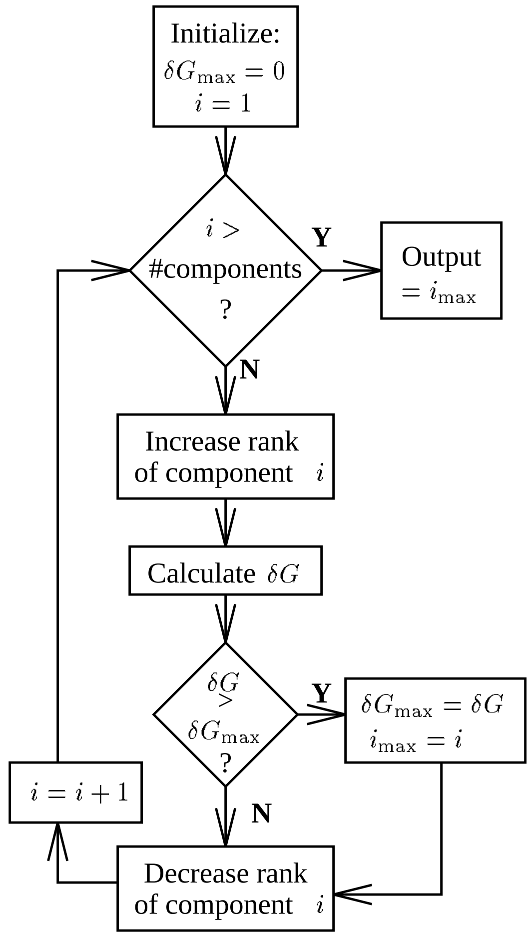

Figure 3: Flowchart for identifying most sensitive component

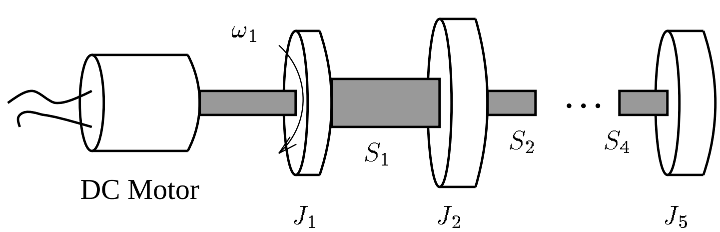

Figure 4: DC motor drive train

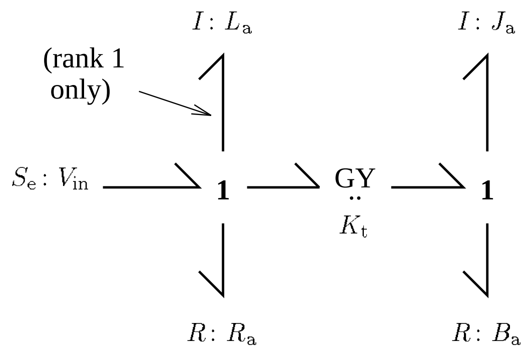

Figure 5: DC motor

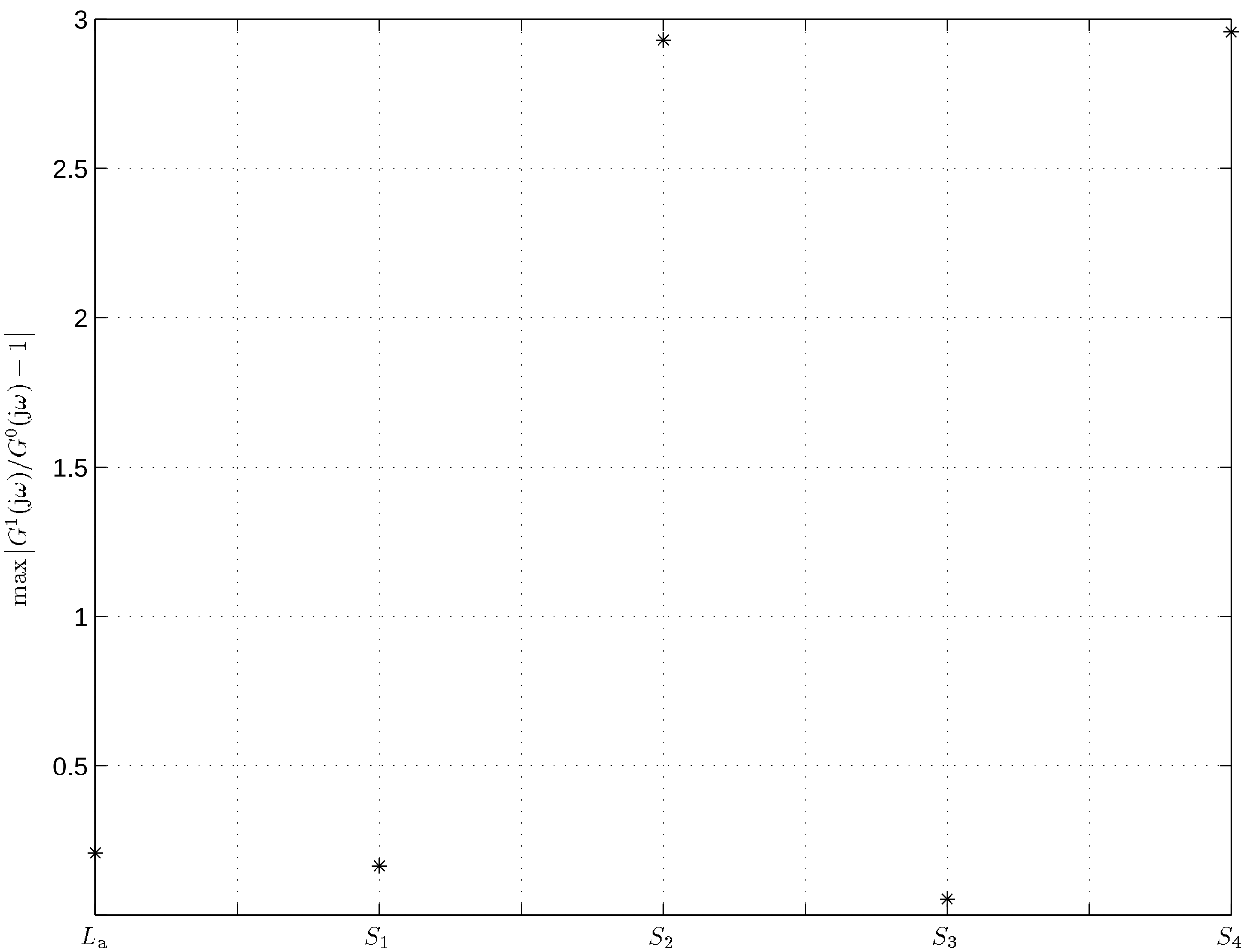

Figure 6: Maximum change in frequency response for 00000 model

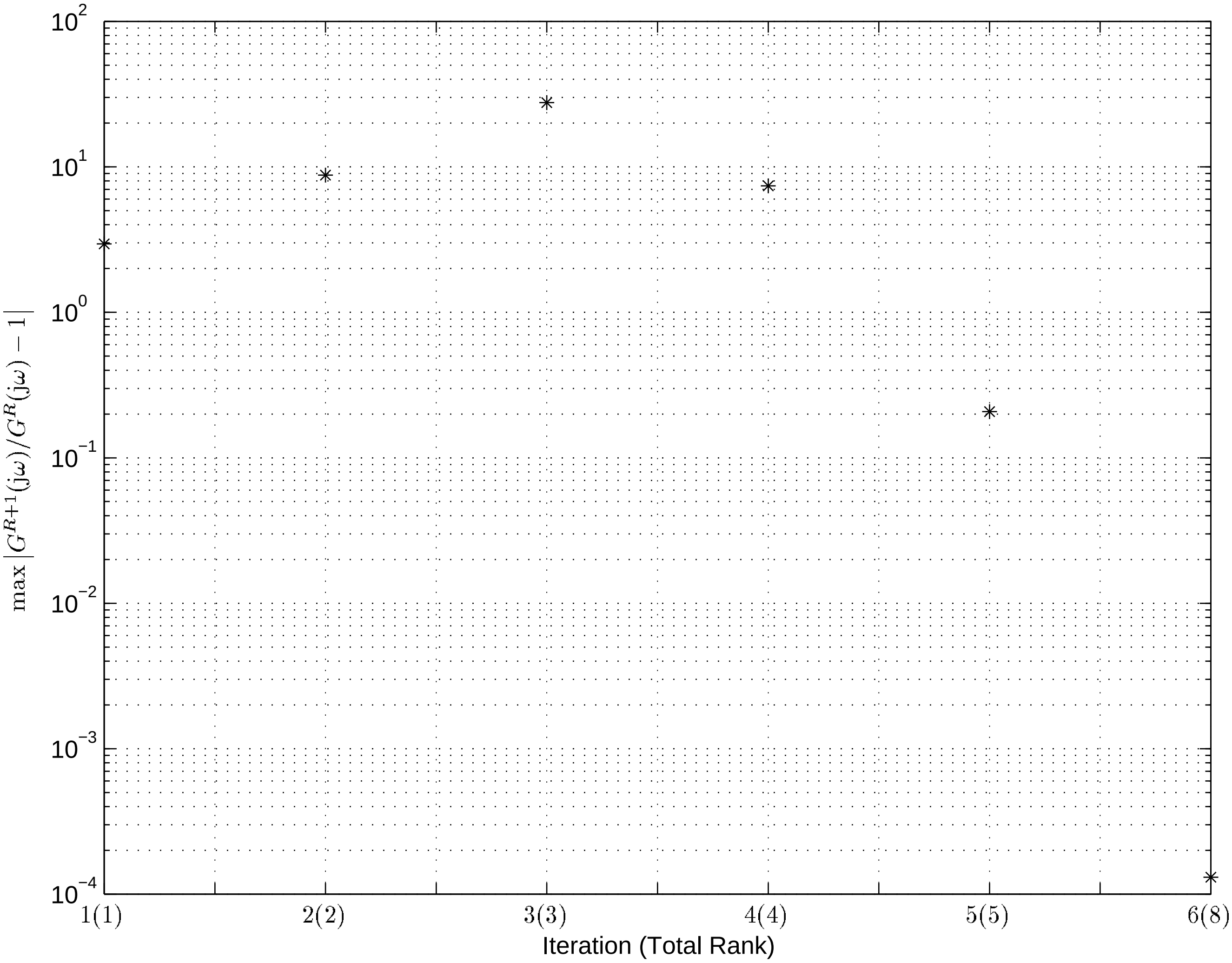

Figure 7: Change in frequency response vs. iteration

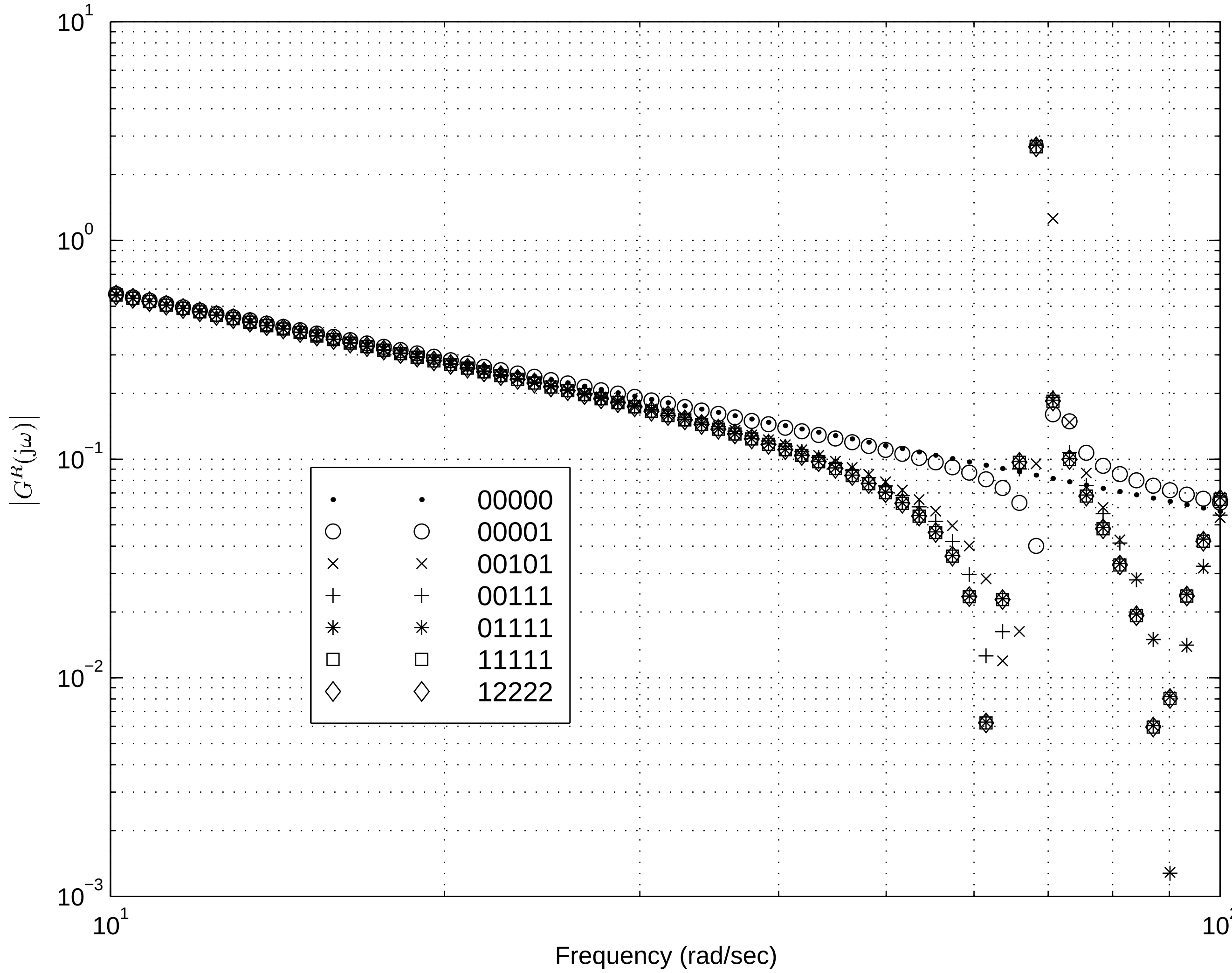

Figure 8: Frequency response during model synthesis