Multi-Configuration Model Tuning for Precision Opto-Mechanical Systems

by

Deborah Jane Howell

B.S., Rensselaer Polytechnic Institute, 2001

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science in Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2004

Author . . . .
Department of Aeronautics and Astronautics
January 29, 2004

Certified by . . . .
Olivier L. de Weck
Assistant Professor
Thesis Supervisor

Read by . . . .
Carl Blaurock, Ph.D.
Thesis Reader

Accepted by . . . .
Edward M. Greitzer
H.N. Slater Professor of Aeronautics and Astronautics
Chair, Committee on Graduate Students

Multi-Configuration Model Tuning for Precision Opto-Mechanical Systems

by

Deborah Jane Howell

Submitted to the Department of Aeronautics and Astronautics
on January 29, 2004, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

It is important for the design of future space-based observatories that simulation models physically represent the designed system and are able to track along configuration changes. This thesis outlines a three-step procedure for model tuning of complex opto-mechanical systems in the presence of measured experimental data. It is the hypothesis of this thesis that this procedure will produce a model that effectively tracks along configuration changes. The first step, engineering insight, applies model heuristics to the simulation model in an effort to produce a simulation model that includes all physical effects in the experiment. The next step, model updating, is an automated procedure whereby an optimization problem is formed in order to set uncertain model parameters. The final step is model tracking across configurations. Configuration changes include, but are not limited to, changes in mass, input/output locations, changes in geometric properties and relative placements. A new metric is provided which helps to gauge the level of experimental/model mismatch in the new configuration (using the updated model) by using the objective function from the optimization in Step 2. Using this metric, one can determine how the model changes with respect to specific configuration changes. Finally, this three-step tuning procedure is compared against traditional model tuning on a testbed at the MIT Space Systems Lab (SSL) in order to gauge its usefulness. The traditional model tuning will be performed by a colleague in the SSL who will use such methods as trial-and-error parameter updating to match the simulation model to the experimental data. Using the multi-configuration metric presented in this thesis, it is shown that the model produced using the three step method does track configurations better than the model produced using traditional model tuning.

Thesis Supervisor: Olivier L. de Weck
Title: Assistant Professor

Acknowledgments

Thank you Mom and Dad and Marty for the unconditional support and encouragement throughout my graduate career. Thank you Olivier de Weck for your support and guidance on all my research work. You allowed me the independence to address research that is important to me and provided an open environment where various research questions could be thoughtfully examined. Thanks to Carl Blaurock for taking the time to read through this thesis and providing excellent comments. Thanks to the DOCS team at MIT's SSL, especially Dave Miller, Becky Masterson, Scott Uebelhart, Soon-Jo Chung and Dave Lobosco for helping to analyze data and for guiding my research goals. Special thanks so Scott Uebelhart for helping out with the model updating in Chapter 5, and for being a great guy. Thanks to Ipek Basdogan and Scott Ploen at JPL for your encouragement on the importance of this work. This work was supported by the National Defense Science and Engineering Graduate Fellowship program.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

COMAC	Coordinate Modal Assurance Criterion
DOCS	Disturbance-Optics-Controls-Structures
FSM	fast steering mirrors
I/O	input/output
IMOS	Integrated Modeling of Optical Systems
MAC	Modal Assurance Criterion
MACE	Middeck Active Control Experiment
MIMO	multiple input/multiple output
SISO	single input/single output
SVD	singular value decomposition

Symbols

i	modal number
α	amplitude scale factor
δ_B	beam tip displacement (m)
$\frac{\mathrm{dy}}{\mathrm{dx}}$	slope of centroidal axis
γ_0	slope of beam due to shear effects
$\hat{\omega}$	measured modal frequency (Hz or rad/sec)
$\hat{G}$	complex model transfer function matrix
κ	Timoshenko shear coefficient
λ	intermediate variable for axial beam mode determination
$[\phi]$	matrix of mass normalized eigenvectors
$[I]$	identity matrix
μ	sensor noise parameter
Ω	natural frequency matrix
ω	analytical modal frequency (Hz or rad/sec)
ω	frequency (Hz or rad/sec)
Φ	modeshape matrix
ϕ	modeshape vector
ϕ	phase shift (degrees or radians)
ψ	slope of beam due to bending
ρ	density (kg/m^3)
σ_i^2	singular values
ξ	mismatch of modal frequency

a	major ellipse axis (m)
A, B, C, D	state space matrices
b	minor ellipse axis (m)
$G(\omega)$	complex transfer function
I	moment of inertia (m^4)
J	cost function
J	polar moment of inertia (m^4)
K	stiffness matrix
k	spring constant (N/m)
L	length of the beam (m)
M	mass matrix
m_p	OPTSIM bus mass
P	force due to the concentrated masses (N)
p_{LB}	lower parameter bound
p_{UB}	upper parameter bound
U,V,Σ	SVD matrices
V_S	OPTSIM input disturbance signal
W	weight of the uniform beam (N)
W	weighting matrix
$G + \mu$	experimental transfer function matrix
$\mathbf{I}$	inertia matrix
q(x,t)	force per unit length (N/m)
A	amplitude
E	Young's modulus (Pa)
G	shear modulus (Pa)
K	spring constant
K6ROT	NASTRAN rotational parameter
NL	noise level
p	parameter
R	eigenvalue sensitivity matrix
r	rank number
Y	validation matrix

Units are in SI

Chapter 1

Introduction

The next generation of space telescopes being developed by NASA are extremely complex multidisciplinary systems that often require models with predictive capability in the nanometer range (e.g. James Webb Space Telescope, Terrestrial Planet Finder, Space Interferometry Mission). In the planning and design phases, developers routinely rely on simulation models to determine hardware type and configuration and to estimate performance and cost. Also, developers would like to change aspects of these simulation models in order to improve performance or lower cost. When it is possible for a hardware prototype model to be built [1], developers use them to confirm the simulation model predictions. Even if these models match, the system would not necessarily perform or behave in the same way as it would on-orbit. The difference between the model and the actual system behaviors is called model error. Model error can affect all of the subsystems (controls, structures, optics etc.), which, in turn, affects cost and return estimates. Model error affecting the controls subsystem is more critical than the other subsystems since it could cause the system to become unstable in a very short amount of time. This problem is addressed either by designing a controller that would provide the minimum robust performance for a wide range of disturbances, or by performing on-orbit system identification and control design. It is necessary, for control design and for making changes in the simulation model,

[1] Some of the reasons why a prototype model cannot be built is that construction is prohibitively expensive, time consuming, or not possible in a 1-g environment

Figure 1-1: Artist's Concept of SIM [29]

that an accurate simulation model exists. This thesis focuses on the development of physically accurate open loop simulation models that are valid for large design changes either in the pre-hardware stage, or after a prototype hardware model has been built. A physically accurate simulation model will enable modelers to evaluate the system over a larger tradespace and to perform optimization, uncertainty, and sensitivity analyses.

1.1 Motivation

As part of the Origins program, NASA is developing a spacecraft interferometer called the Space Interferometry Mission (SIM), to look for distant evidence of planetary bodies (projected launch date 2009). SIM is an extremely complex opto-mechanical system that is relying on several subsystem models [29] for mission performance prediction. The accuracy of these models is key to the construction of an effective mission. Each subsystem is simulated and is built using hardware. The simulation is then validated and tuned using data from the hardware model. Next, these models will be integrated together, and the final performance will be predicted using this conglomerate model in the presence of a modelled disturbance environment. Unfortunately, there is no guarantee that this approach will produce an on-orbit system which is controllable to within the performance requirement, even though the performance predicted by the model is acceptable. First, the testbeds are not made using the actual flight hardware, so the structural properties may not be the same. Secondly, the level of fidelity of each subsystem model could be different. Thirdly, these models do not undergo the same environmental conditions in test as they do in flight. Other inaccuracies include the incorrect modeling of pre-stress conditions, boundary conditions, disturbance sources, and subsystem interfaces. This is not an exhaustive list by any means, but it does illustrate the need to address several issues during the modeling process. In addition, the chosen performance metrics could be highly coupled (i.e. coupled over a large bandwidth encompassing many modes), therefore the performance metrics could be affected by more than one design metric. In that case, two or more sets of unique design metrics (or variables) could produce the same performance value (see Section 4.1). It is for this reason that it is crucial not only to have prediction accuracy, but also to have the correct combination of design metrics.

Recently, a mishap involving the X-43A was attributed to modeling inaccuracies [1]. The X-43A was a flight system that was developed to demonstrate the benefit of hypersonic air-breathing propulsion. On June 2, 2001, the X43-A was released from a B-52 carrier. Approximately 11 seconds into the flight, the X-43A experienced divergent roll oscillations. Subsequently, the control surfaces failed to respond, and the flight system departed greatly from the planned trajectory, and was then terminated by ground control. The root cause of this event was stated as: "The X-43A HXLV failed because the vehicle control system design was deficient for the trajectory flown due to inaccurate analytical models (Pegasus heritage and HXLV specific), which overestimated the system margins". The entire flight system was lost, as well as any performance data on the propulsion system. This incident emphasizes the importance of accurate simulation models for complex flight systems.

This thesis strives to better understand and improve the relationship between simulation and hardware models with the purpose of being able to better predict on-orbit performance. **The focus of this thesis is on building simulation models of integrity which not only match (to some level) the hardware model, but also match the hardware model throughout configuration changes.**

Configuration changes include, but are not limited to, mass, material, and geometric changes. These changes do not alter the overall structure or the functionality of the system. Specific examples of structural configuration changes are changes in bus mass or other concentrated masses, truss beam length, disturbance input location, and performance metrics. Thus, the scope of this thesis does not include "reconfigurable spacecraft" as discussed by Jackson [27]. The changes discussed in this thesis are the types of changes that are common once an architecture for the spacecraft has been chosen, but before the details are filled out. At this stage, the system undergoes configuration changes in order to optimize the system or make it more robust. Models that are accurate across configuration changes have immense value. The developer is able to change the simulation model relatively easily and immediately see performance and cost changes, which in turn reduces performance risk. In this way meaningful optimization and sensitivity studies can be performed.

Here, a "bottom-up" approach is presented in which subsystems are broken into disciplines (e.g. disturbance, structures, optics, controls). The simulation model starts with the modeling of one discipline subsystem. This model is then updated using testbed data and a three step procedure that is outlined in Chapters 2, 3 and 4. Once the developer is satisfied that this model captures the behavior of the system in a physically meaningful way, another discipline is added to the model, and the process is repeated. Figure 1-2 depicts a system that contains the subsystems of disturbance, structures, optics, and control (typical subsystems for space telescopes). The model for this system starts with the disturbance and structure models (dotted line), and then after the model is sufficiently updated (via the three step process), the optical subsystem is included and updated (dashed line). Then the final subsystem, control, is added and the entire system is tuned. In this way, a parallel, progressive building of simulation and hardware models, the designer is better able to pinpoint modeling errors and correct them. Note that these errors include the error within each subsystem model and errors in the interface model. Compliance (stiffness, damping, etc.) at interfaces is very important when connecting subsystem models. This procedure focuses on the specific interconnections of each subsystem

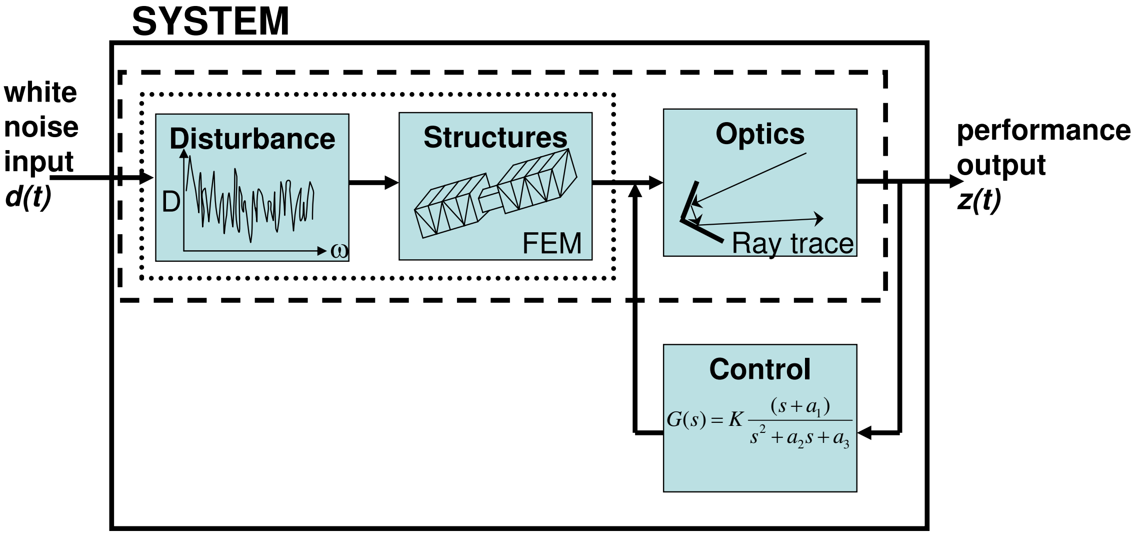


Figure 1-2: Flow Chart of “Bottom-Up” Approach

at each additive step, thereby allowing the modeler to examine the model error and interaction effects at each step. There still does exist the problem of achieving the same or the required level of fidelity of each subsystem model. This thesis will present the theory associated with model updating at each design iteration step. It will also apply this theory to an open loop structure, and set the stage for application to a more complex system.

1.2 Previous Work

Several papers and theses have been written on the process and application of integrated modeling of space structures Melody [38], Gutierrez [23], Glaese [19], de Weck [12] Shaklan [45]. An overview is provided here. In integrated modeling, multiple aspects of the system (i.e. structures, controls, optics) are combined into a single linear model, generally using the state space representation. The scope of this thesis and its application is limited to this linear modeling process. Using an integrated model is especially important for distributed optical systems, like SIM, since the flexibility of the system greatly impacts the optical performance. This impact can be measured and predicted quickly and easily using a integrated model, since laborious communication between the separate models does not need to take place.

Much of the research done on the topic of model updating for complex space structures has been in the realm of automated tuning Gutierrez [23], Natke [40], Flanigan [16], Caesar [9], Link [34], Maia [36], whereby both a simulation model and experimental data are available. In general for complex systems, the experimental results do not match the initial simulation, and model updating is required. When looking at the frequency response functions the pole zero sequences often do not line up. Also, the low and high frequency asymptotes of the dereverberated transfer function (backbone) might not match. Low frequency asymptote mismatch can be attributed to incorrect mass or static stiffnesses. High frequency asymptote mismatch can be due to incorrectly modelled dynamics. In automated tuning, parameters in the simulation model are first selected to be tuned, based on some criteria [31] [32], [33]. These parameters can be values for stiffness, length, mass, damping etc. Next bounds are chosen for these parameters. Then an optimization is run where the objective is to minimize a measure of mismatch between experimental and simulation results by altering the chosen parameters within their bounds [46], [55], [20], [15]. Several things can happen during automated tuning which creates a model that is non-physical, and does not track across configurations. First, error can exist in the model which is not captured by model parameters alone; this is called non-parametric error. Non-parametric error occurs when the physics of the system are not properly modelled, and can consist of such things as incorrectly modelled interfaces and incorrect boundary conditions. Table 1.1 shows possible sources of non-parametric error across three disciplines. Second, if several parameters are able to be adjusted in the model, there is no guarantee that these parameters are changed properly or in a physically valid manner as a result of automated tuning. This is because there are many solutions (combinations of parameter values) which can cause the model and experimental results to match, and only one combination physically represents the system. Thirdly, the correct parameters might not be included in the automated updating. This causes the algorithm to force a model/experimental match by setting the included parameters to non-physical values. In all three of these cases,

Structures	**Optics**	**Controls**
boundary conditions	mirror misalignment	sensor & actuator misalignment
gravity effects	source intensity fluctuations	coding and logic errors
suspension effects (pendulum modes)	sensor pixelization (CCD)	unmodelled sensor error & feedthrough
insufficient discretization	finite laser bandwidth (non-monochromatic)	actuator non-linearity
pre-stress from assembly	diffraction	sensor drift, hysteresis
compliance		delay and timing problems
freeplay in joints	imperfections in optics	A/D and D/A errors (round-off) and non-linearities

Table 1.1: A Survey of Non-Parametric Error Sources Across Various Disciplines

the tuned model will be necessarily valid only for that configuration. A configuration change is considered a change that does not add or subtract major components, but a change that moves or adjusts these components. Therefore, using classical automated tuning alone there is no guarantee that the model would reflect the behavior of the system with a configuration change accurately.

An excellent process for developing zero-gravity structural control models is detailed in Glaese [19]. In this work, both a simulation model and the Middeck Active Control Experiment (MACE) test article are constructed in the 1-g gravity field. MACE is a space shuttle flight experiment, and comprises of a long tubular Lexan bus, a disturbance source supported by a gimbal on one end, and a pointing/scanning payload on the other end (also attached by a gimbal). The gravity field on MACE is offloaded using suspension wires, and a reaction wheel assembly is mounted on the center of the Lexan tube. There is a section in Glaese on finite element model updating that includes using both engineering insight (to fix non-parametric error) and automated tuning (to fix parametric error using test data). The simulation model is then tuned using data collected from the testbed. Next the gravity field and the suspension wires are removed from the simulation model in order to predict behavior

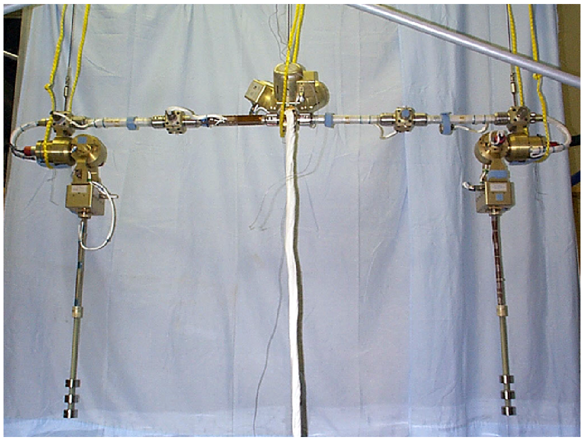


Figure 1-3: MACE Test Article

in zero gravity.

Another work by Campbell [10] does address the concept of multiple configurations but does not propose a tuning process. It also provides an extensive assessment of modeling uncertainties.

1.3 Thesis Objectives

This thesis strives to provide a method of constructing a proper, physical model, and then, if a test article exists, to tune the model using test data such that it tracks across configurations. This work will provide a comprehensive list of model checks based on engineering insight for application to the initial model (Chapter 2). Though not a part of this thesis, the ultimate goal is to incorporate these model checks into an expert system to guide the analyst during the first phase of development. Once these checks have been applied, the initial model will be sufficiently improved so that automatic tuning can be applied. If test data is not available, this simulation model will then be better, and architectural and major design changes can be evaluated. Such a model would be extremely useful during early design phases, where a test article is not immediately available. If test data is available, the multi-configuration tuning process can begin, so that the developer can perform optimization and sensitivity studies. This work will not produce an expert system, but it will lay out a procedure for open loop updating. Contributions include a comprehensive list for producing a proper model, and then a procedure for multi-configuration tuning/verification. The overall goal is to provide modelers without testbeds a checklist for proper modeling methods, and to ensure model tracking in the tuning algorithm for those with testbeds.

Thesis Objectives Summary

1. Comprehensive non-parametric model error checklist

2. Multi-configuration tuning procedure

3. Laying the groundwork for adding additional disciplines into this methodology

1.4 Three Step Procedure (Thesis Roadmap)

The procedure here for open-loop finite element model updating is laid out in three steps (see Figure 1-4). Step 1, Coarse Tuning/Model Topology (Chapter 2), makes use of engineering insight to take out non-parametric (and some parametric) error from the model, such that all the important components and interfaces are properly represented in the model. The goal of this step is to capture all the important physics of the system, and to match the modal properties of the testbed. If a transfer function is experimentally acquired, then its low and high frequency asymptotes, slope of dereverberated backbone, and pole/zero sequence should match the model's transfer function at the end of this step. In addition the frequency of each mode should match to within a certain tolerance. Steps should also be taken to make sure that each modeshape described in the simulation model does indeed match the mode of the testbed. The next step, Automated Tuning (Chapter 3), employs classical model tuning, but in a way such that the model is ensured to track across configurations. Here parameters of the system (Young's modulus, mass, inertias, etc.) are changed according to an optimization algorithm that minimizes model/experiment mismatch.

The third and final step, Model Validation and Tracking (Chapter 4), verifies that the simulation model produced in Step 2 tracks across configuration changes. If not,

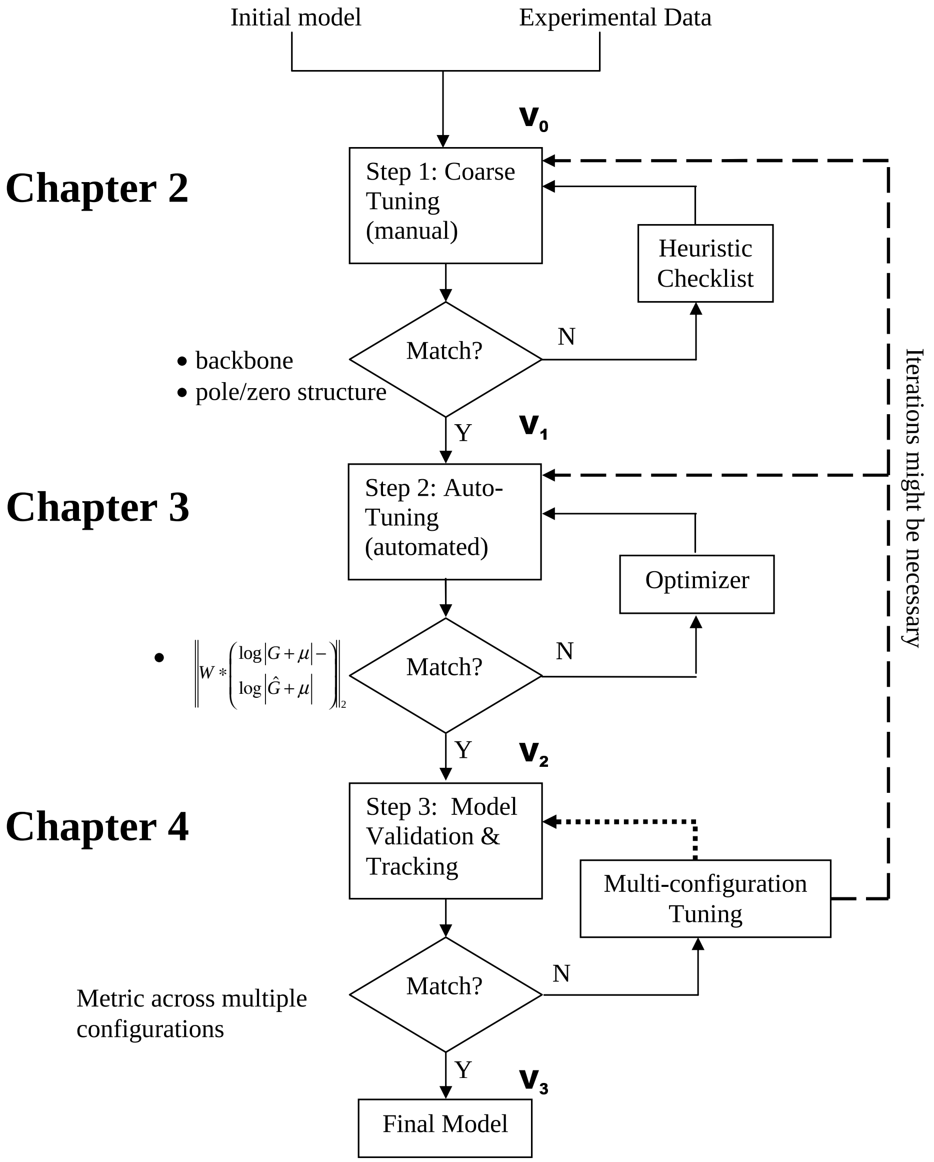

Initial model
Experimental Data
V_0
Chapter 2
Step 1: Coarse Tuning (manual)
Heuristic Checklist
• backbone
• pole/zero structure
Match?
N
Y
V_1
Iterations might be necessary
Chapter 3
Step 2: Auto-Tuning (automated)
Optimizer
• $\left\| W * \begin{pmatrix} \log\left|G+\mu\right| - \\ \log\left|\hat{G}+\mu\right| \end{pmatrix} \right\|_2$
Match?
N
Y
V_2
Chapter 4
Step 3: Model Validation & Tracking
Multi-configuration Tuning
Metric across multiple configurations
Match?
N
Y
V_3
Final Model

Figure 1-4: Thesis Roadmap

the modeler must return to Step 2, and repeat the automated tuning process with the added data. If so, then the model is ready for optimization and sensitivity analysis. In Step 3 there is a distinction between tuning data and validation data. The tuning data was used to tune the model, so the model should match this data well. The validation data consists of different system configurations and was not previously used in the tuning procedure. If the tuned model (with the appropriate configuration changes) matches the validation data well, then it is an indication that the model's physical representation and parameter values are realistic.

Iterations between Steps 1, 2, and 3 might be necessary. Often modelers become aware of non-parametric error in Steps 2 or 3 when the parameter values need to be changed an unreasonable amount in order to match the model to the experimental data. An iteration back to Step 1, and a re-examination of modeling assumptions is then required.

In Chapter 5, the three step process will be applied using a complex structural test article built at the Space Systems Lab at MIT called OPTSIM. This testbed structurally represents a spacecraft interferometer complete with a bus area, and a truss for mounting optics. Chapter 6 summarizes the contributions of this thesis, and the main conclusions reached during implementation. It also presents some recommendations for application of this work to more complex systems (e.g. MPI).

1.5 Methods and Programs

1.5.1 Software

There are several software packages available to create simulation models of complex opto-mechanical systems. Generally these specialize in a certain discipline. ZEMAX [54] is an optical prescription simulator which is capable of both geometric ray tracing and diffraction analysis. MACOS (Modeling and Analysis for Controlled Optical Systems [43]) also simulates optical systems, and provides an optical sensitivity matrix which incorporates easily into a state space representation. NASTRAN [2] is a finite element code that is used mainly to model the structural behavior of a complex system. FLUENT is generally accepted as the most state of the art CAE tool in computational fluid dynamics.

There are also integrated modeling environments in which the user can model the entire system using a single software programs. Two examples are IMOS and DOCS which are both based in the MATLAB$^{®}$ environment. IMOS (Integrated Modeling of Optical Systems) was developed at NASA JPL as a user-friendly finite element code that also contains optical ray-tracing capabilities. IMOS was developed especially to analyze distributed optical systems. DOCS (Disturbance-Optics-Controls-Structures [6]) originated at the Space Systems Lab at MIT, and is now being developed commercially at Midé Corporation in Medford, MA. DOCS does not include finite element code, but it does interface with other finite element codes, so that existing models can be easily integrated. DOCS is able to assemble separate discipline models into an integrated model which can be explicitly defined as a function of design variables. This enables tradespace exploration and sensitivity analysis in an analytical environment, which is useful for model updating. In addition, DOCS is extremely versatile because it can model and optimize very different kinds of systems.

1.5.2 Hardware

Hardware varies greatly between systems. Besides material and geometric differences in the actual structure, there are also differences in sensor and actuator types, and also in types of mechanisms used (see Figure 1.2). In every case, the modeler must be aware of calibration factors, time delays, and conversion (roundoff) errors when working with new equipment. Before the system is realized in hardware, the model is based on parameter values from literature, manufacturer's data sheets, and CAD geometry. Once the system is built, many of these values can be measured (mass, stiffness, sensor bandwidth, etc.). Since each component will be unique and will most likely not conform exactly to the published reports, the modeler should measure as many values as possible, as accurately as possible. This will help to create a more accurate model (further discussion is provided in Chapter 2). Values such as mass and dimensions can, in most cases, be determined to a much higher accuracy than values such as stiffnesses and damping coefficients. Stiffness and damping values, while measurable, will generally have a higher amount of uncertainty; it is for this reason that these values are set through an automated updating procedure, as further discussed in Chapter 3.

1.6 Chapter Summary

In this chapter motivation for the three step tuning procedure was presented, along with an outline of previous work performed in the development of integrated modeling for spacecraft. Thesis objectives and a thesis roadmap were shown. The hardware and software modeling methods were also summarized.

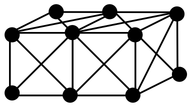
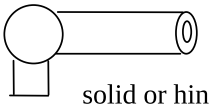
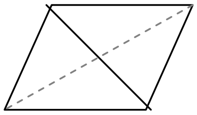
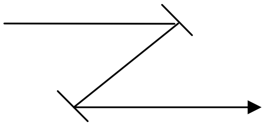
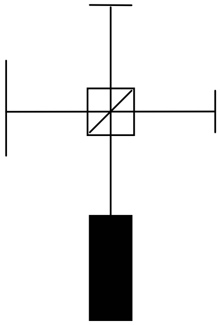
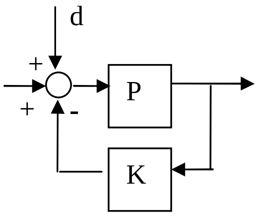
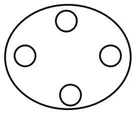
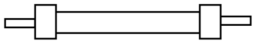
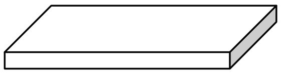
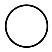
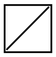
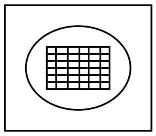
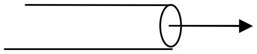
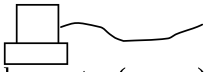
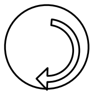
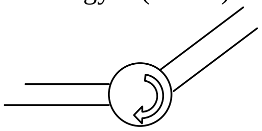
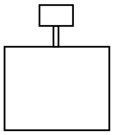
	Structures	Optics	Controls
Assemblies	 trusses  solid or hinged structures (e.g. robotic arm, MACE)  membrane structure	 optical delay line  Michelson interferometer	 PID controller
Single Components	 ball (node)  strut  plate	 flat mirror  beamsplitter  CCD camera  Laser (source)	 accelerometer (sensor)  rate gyro (sensor)  gimbal (actuator)  shaker (actuator)

Table 1.2: Hardware Overview

Chapter 2

Step 1: Coarse Tuning

When a complex system is being designed, an engineer usually must reconcile two things: an initial simulation model, and a measurement model. The initial simulation model (a priori model) can represent the entire system, only a set of subsystems, or components using a finite element model (structures), a ray-tracing program (optics), and/or a controller. The simulation model is generally created first, and is a somewhat idealized version of the system design. The measurement model is generated from data taken from a testbed that is built to the system specifications. Unfortunately, the testbed cannot be built *exactly* to the system specifications. Also the operating conditions of the real system and the testbed might not be the same (0-g in a vacuum vs. 1-g in an atmosphere). Once the initial simulation model of the complex system is developed, parametric and non-parametric model error will generally become apparent, either through inspection/intuition of the model, or via direct comparison with measurement data. If the data is collected properly, the measurements will be taken to be the truth model for the system. Non-parametric error cannot be resolved by simply changing the values of uncertain parameters in the model, but rather results from elements of the model that are improperly represented (incorrect boundary conditions, element connectivity, unmodeled physical effects, etc). Non-parametric errors often manifests themselves in initial models as large mismatches in modal frequencies and in incorrect backbone roll-off and pole/zero sequence in transfer function plots. This chapter will enumerate heuristic rules that the modeler can invoke to ensure that the non-parametric error is significantly reduced so that the basic/important dynamics are captured in the model. Even though these heuristics are listed in order to reduce non-parametric error, they also, to an extent, help to reduce parametric error. A section on overall implementation of these heuristics is provided. Once these checks are complete, the modeler should have a model with little or no non-parametric error, or at least a model that is a better representation the system. At the end of this chapter, a sample problem is presented in order to illustrate the consequences of leaving sources of non-parametric error in the simulation model.

2.1 Non-Parametric Error Heuristics

The following list of model heuristics is a compilation of engineering insights usually performed by an engineer using his experience and intuition. This is based on engineering insight from Glaese [19], Campbell [10], and Stockwell [49]. These checks should be applied to simulation models after the initial model is completed, but before automatic tuning is performed. Even if there is no measurement data available, these checks remain important to perform in order to get the most error-free model possible. This work recommends a "bottom-up" approach for assembling the test article, as the list below will reflect (Figure 2-1). Parts of the system should first be tested at a component level. Then, if needed, the components are assembled into different subsystem modules, and tested. Finally the entire system is assembled from the subsystems and tested. This approach ensures that parameters such as mass and inertia are accurate in the model. The model checks in this section are listed in the order in which they should be performed. This corresponds to an ordering where checks that are easy to perform, yet provide a relatively large amount of error reduction are ordered first. Checks that are difficult to carry out and only reduce the error by a relatively small amount are towards the end of the list. This ordering is based on the authors experience and can change given a different system. It is only a guide and it is up to the modeler to choose the appropriate order. Note that all complex systems are different, and probably have different sensitivities to each item on the

Model Heuristic	**Description**
Typographical Errors	Reality check for all parameter values and consistent units of measurement
Mass and Inertia	Total and component weights & inertias
Component properties	Stiffness, inertial, geometric and placement properties (only once assembled in subsystem)
Finite element code	Valid choice of FE program
Element types	Loading and behavior different for different elements (e.g membrane vs. shell element)
Boundary conditions	Fixed, pinned, free conditions at model boundaries
Sensors and actuators	Position, orientation, and calibration
Gravity effects	Presence of gravity or off-loading conditions
Discretization	Fineness of finite element meshing or number of states
Linearity	Identify linear and non-linear system regimes
Modal Survey	Match simulated modes with measured modes
Damping	Realistic damping values
Interfaces	Boundary conditions and compliance at important interfaces, impedance matching

Table 2.1: Summary of Model Heuristics

list, therefore it is important to consider addressing each item, keeping in mind the probable impact it has on the system. Table 2.1 is a summary of these checks.

2.1.1 List of Model Heuristics

1. **Typographical errors.** Even though this item might seem trivial, these types of errors are bound to arise in models of complex systems. These errors can be tedious to comb out of a simulation code, but can be extremely damaging to the model. For example, a misplaced decimal point influencing the mass of the bus (usually the heaviest part of the spacecraft), can shift the center of mass of the system, affecting low frequency mode shapes. A good way of finding these errors is to split the code up into sections, and address each section on a different day, that way the eye will not be familiar with the code, and will be less likely to pass over the errors. Another method is to have an independent individual or group double check the code, and perform a “test run”. In addition, the

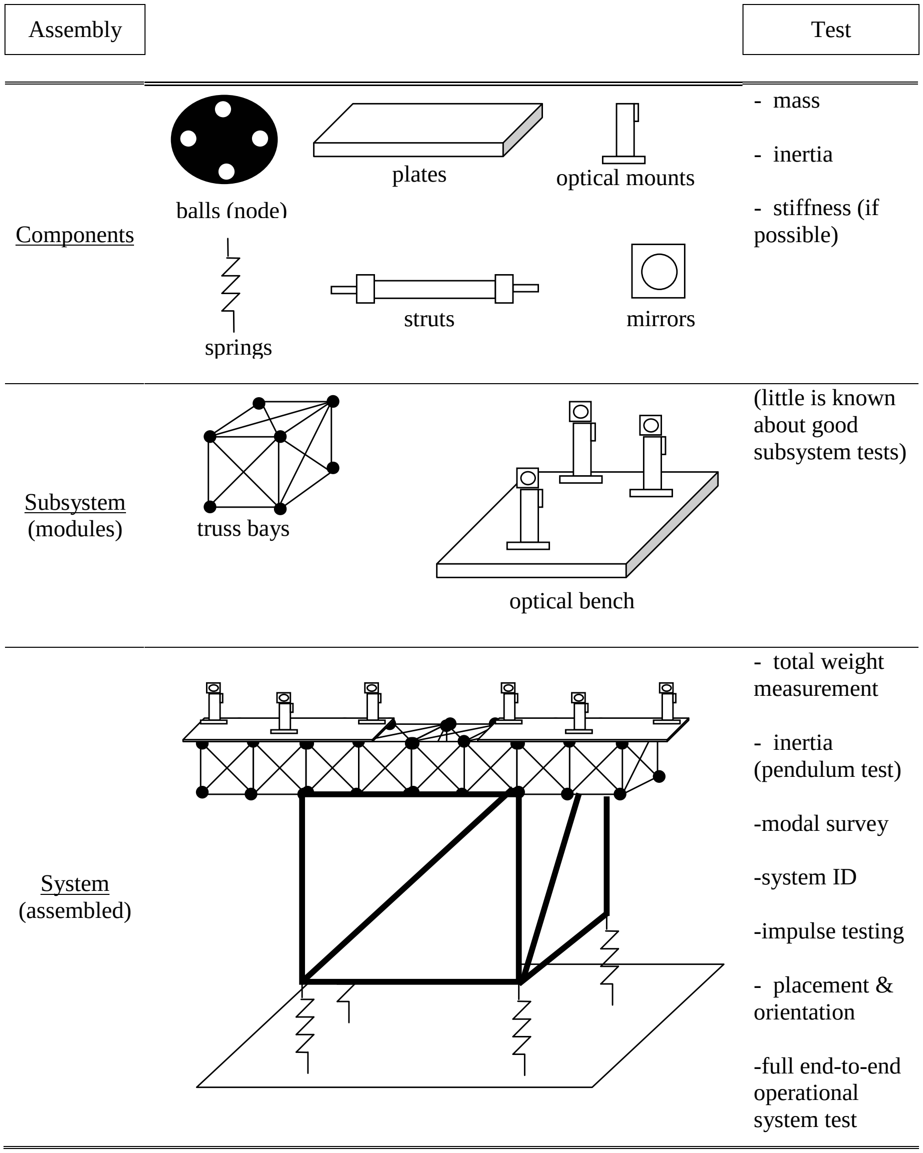


Figure 2-1: Hardware Assembly Sequence and Tests

component subsystem models are sometimes made by different people or teams, and therefore things such as coordinate systems and units might be dissimilar and need to be double checked. For example, it might make sense to model the structure in one coordinate system, but the optics might best be modelled in a coordinate system that is rotated and translated from the structural coordinate system.

2. **Mass and inertia.** In the structural model, the entire mass of the system should be compared against the total measured mass. If a discrepancy is found, it might be required to disassemble the testbed and weigh individual component masses. A good practice is to build a testbed from the ground up by weighing individual components and keeping a meticulous mass budget. For point masses, the mass matrix can be examined in order to locate the inconsistency. For components with distributed mass, the measured weight should be checked against the components' volumes and densities in the simulation model (since this is how finite element models compute component masses). Values of inertia, however, are very difficult to measure. The components' inertial properties about the system's principal axes (x, y, and z) are represented using a symmetric 3x3 inertia matrix (see Equation 2.1). Finite element software and some CAD models provide this matrix. The values in this matrix are usually based on an idealization of the component's form and therefore can be in considerable error. For example, the cross inertias (off-diagonal terms in Equation 2.1) of a beam element are often idealized as zero. Since these elements can never be exactly symmetrical due to component tolerances, the cross inertias are, in practice, non-zero.

$$\underset{3x3}{\mathbf{I}} = \begin{bmatrix} I_{XX} & I_{XY} & I_{XZ} \\ I_{YX} & I_{YY} & I_{YZ} \\ I_{ZX} & I_{ZY} & I_{ZZ} \end{bmatrix} \tag{2.1}$$

Inertia can be measured using a pendulum test [48] for small angle movement or a gravitational pendulum [28] for larger angles.

3. **Component properties.** Aside from mass considerations, other properties that can be measured in the lab should be checked. These might include stiffness properties (Young's modulus, shear modulus), placement, and orientation. Component stiffnesses are generally obtained from material handbooks, but can vary on individual pieces. The variation of individual parameters and their propagation through analysis is further described in Section 2.3. It is possible to measure stiffness in the lab for rigid components like aluminum beams (Tension testing using an Instron machine [53]), but is usually difficult and time-consuming. Elastic components (e.g. springs) can be measured in the lab using extension and rotation tests. The component's geometric properties (length, diameter, etc.) are easy to measure in the lab, and should always be measured to the best of the modeler's ability. Once the system is assembled, the placement and orientation of each component should also be measured carefully.

4. **Finite element code.** The choice of different commercial finite element programs has been shown to effect the model-experiment mismatch [19]. One solution is to model the system using at least two different programs, and compare the results. If they do not match either a third program should be employed or a comparison against measured data should be made. In either case it should become apparent which code provides the best representation. Another method is to use two different independent modelers for the same project (e.g. the U.S. government and Northrop Grumman), and then compare results. Some commercially available FE codes are NASTRAN [2] and ANSYS [39].

5. **Element types.** Some modeling assumptions that enter into finite element analysis have to do with the type of element the component represents. For example, some components might be modelled as lumped or concentrated masses, whereas a consistent mass representation is more appropriate. Similarly, in finite element analysis, beams may be represented using Bernoulli-Euler or Timoshenko dynamic beam equations [21] (Equations 2.2 and 2.3, respectively), where $q(x,t)$ is the external force per unit length. The slope of the beam's cen-

troidal axis, $\frac{\mathrm{dy}}{\mathrm{dx}}$, is made up of the slope due to bending, ψ, and the slope due to shear effects, γ_0. It can be seen that the coupled Timoshenko equations incorporate the shear modulus, G, and the Timoshenko shear coefficient, κ, in order to take into account the shear forces. The term ρI in the Timoshenko equations takes into account the rotational inertia. Therefore, the loading type and beam geometry can and should influence the choice of element type.

$$\frac{\partial^2}{\partial x^2}\left[EI\frac{\partial^2 y}{\partial x^2}\right] + \rho A\frac{\partial^2 y}{\partial t^2} = q\left(x,t\right) \tag{2.2}$$

$$\begin{aligned} GA\kappa\left(\tfrac{\partial\psi}{\partial x} - \tfrac{\partial^2 y}{\partial x^2}\right) + \rho A\tfrac{\partial^2 y}{\partial t^2} &= q\left(x,t\right) \\ GA\kappa\left(\tfrac{\partial y}{\partial x} - \psi\right) + EI\tfrac{\partial^2\psi}{\partial x^2} &= \rho I\tfrac{\partial^2\psi}{\partial t^2} \end{aligned} \tag{2.3}$$

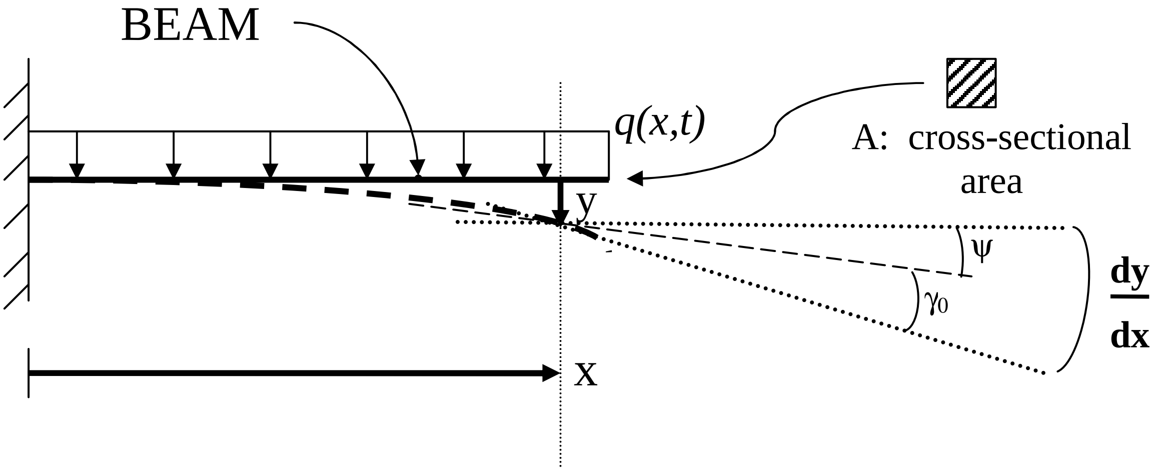


Figure 2-2: Nomenclature for Beam Equations

6. **Boundary conditions.** Boundary conditions are extremely important since they define nodal point constraints in the structural model and therefore determine reaction forces. For example, if there is some play in a connection which is modelled as rigid, unmodeled dynamics will be present due to that joint, and the dynamic behavior might be affected. Also, idealized boundary conditions do not perfectly represent the system. There may be some translation in a pinned connection, some rotation in a fixed condition or friction in a

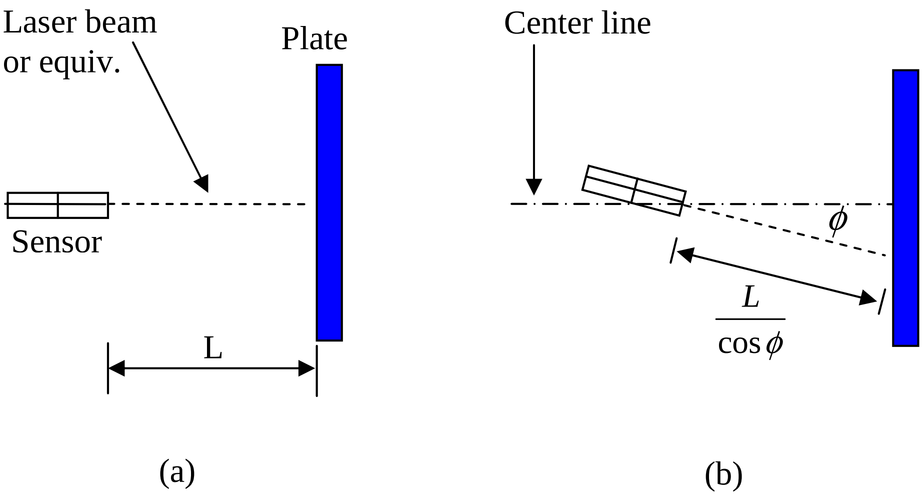


Figure 2-3: Error Due to Sensor Angle

connection that restricts movement in a degree of freedom. These effects are usually visible by the presence of rigid body modes in the model or by examining the low-frequency asymptote in a frequency response function.

7. **Sensors and actuators.** Sensors can measure position, velocity, acceleration, light (brightness, spectra), temperature, and pressure. It is extremely important that the actuators and sensors are not only positioned and oriented correctly, but also that they are calibrated. A sensor mounted at an angle will produce a cosine error in the measurement. A simple example is a displacement sensor, shown in Figure 2-3(a), which measures the displacement of the plate. Figure 2-3(b) shows the error induced in the measurement when the sensor is slightly rotated. Similarly, an actuator which is modelled as producing a force in only one direction, might actually have several directional components. It is important to understand how the force is produced and transmitted. Sensors and actuators which are linear in one regime, might be non-linear in another, so its important to calibrate them accurately. One example of this is the Bentley XL eddy current gap sensor, which can only be modelled as linear in the center of its stroke. The calibration plot for the gap sensor is shown in Figure 2-4. The gap sensor was placed underneath an aluminum plate, which was mounted

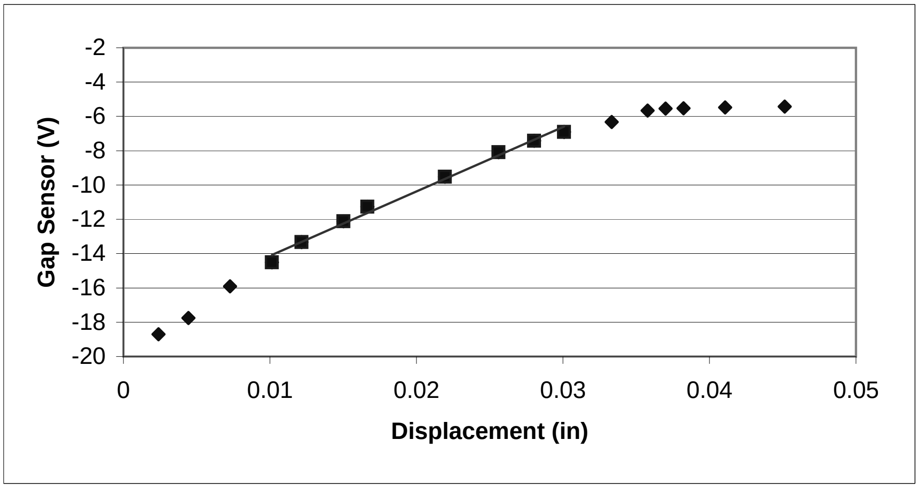


Figure 2-4: Gap Sensor Voltage vs. Displacement

at the four corners on springs. A series of weights was added to the plate, which caused the plate to lower. Displacement in inches and gap sensor voltage were recorded (data points in figure). This is the calibration procedure used in Chapter 5 for OPTSIM. It can be see at the extremes of the displacements, the trend becomes non-linear. The non-linearity at the upper end is due to saturation. For this reason, the gap sensor needed to be readjusted to the center of its stroke whenever mass was added in order to use the linear calibration factor (0.348 V/mil). This factor was obtained by fitting a linear regression line (shown in figure) to the center data points. For controlled systems, it is crucial that the sensor and actuator dynamics (such as time delay and hysteresis), are properly modelled.

8. **Gravity effects.** The force of Earth's gravity is a 3-D vector field and can be relatively easily modelled using finite element software using a gravity loading condition. The force of gravity can couple with existing modes to change the modal frequency and mode shape. This is of special interest to spacecraft systems, since these gravity coupling effects are present on ground-based hardware, but are negligible on-orbit. Suspending ground-based hardware in 1-g such as MACE [19] introduces pendulum modes; placing the testbed on springs

introduces suspension modes. The OPTSIM testbed presented in Chapter 5 is mounted from beneath on springs, and this introduces a relatively low frequency mode. The simulation model is extremely helpful in predicting how the system would behave in zero-g since these loads are easily removed in software. This strategy is more thoroughly developed in Glaese [19]. Another effect is the change in damping that occurs between 1-g and 0-g models. A comprehensive discussion of gravity effects can be found in Rey [44]. .

9. **Discretization.** The level of the model discretization can be described as the fineness of the meshing in the finite element model components or as the number of states in the dynamic system model. If the discretization is too coarse, important modes may be missed. If the discretization is too fine, computational costs may become prohibitive. An assessment of the status and use of high fidelity methods in multidisciplinary systems is provided by Guruswamy [22]. There are several bodies of work that describe model reduction, a technique whereby the model size is reduced, but important modes are retained. A very good overview of model balancing and reduction techniques for large order systems can be found in Uebelhart [51].

10. **Linearity.** Linearity can be a major problem for complex multidisciplinary systems. Sources of non-linearity include stiction and friction, sensor and actuator dynamics, hysteresis, and buckling. Oftentimes, these complex systems make use of integrated modeling. The benefit of integrated modeling is that all of the discipline models are combined into one general state space model where coupling between disciplines can be observed. This poses a problem for highly non-linear systems, because those dynamics cannot be represented using a state space representation of the form

$$\begin{aligned} \dot{x} &= Ax + Bu \\ z &= Cx + Du. \end{aligned} \tag{2.4}$$

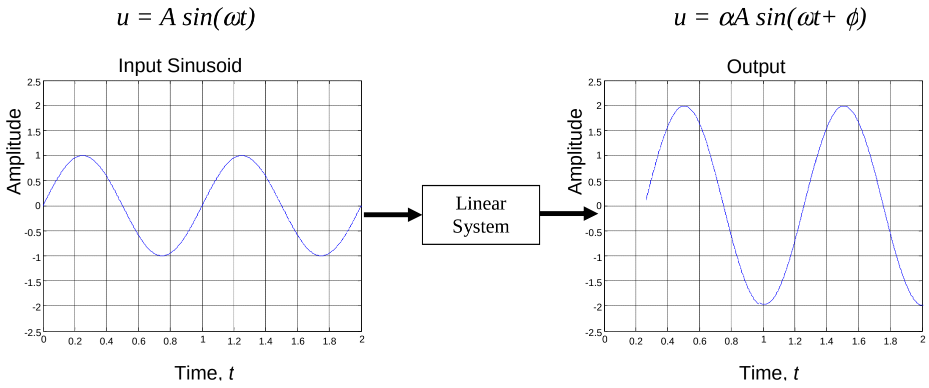


Figure 2-5: Example of Linear system output

A simple example is a non-linear spring, where the spring displacement is not always proportional to the force axially applied to it. Linearity of a system can be checked by increasing the input magnitude of the system (usually disturbance forces) and observing the measured performance (e.g. displacement or acceleration). The transfer function frequency response (magnitude and phase) should not change, since the increase in the disturbance force should only produce a proportional increase in the magnitude performance. If this is not the case, then the system is non-linear. Similarly, if the input is a sinusoid with amplitude A and frequency ω, then the measured output of a linear system will return a sinusoid. The output sinusoid could be scaled in amplitude (scaled by α) or shifted in phase (shifted by ϕ), see Figure 2-5. One can retain the linear model, and assume linearity to within a small region of the nominal operating point (which is generally done) or develop a non-linear model. Non-linear differential equations of motion are notoriously difficult to solve, and this approach (e.g. phase plane approach) is not covered here. Fortunately, most space-based precision opto-mechanical systems behave linearly during their main mode of operation since the applied forces are small once the system is deployed and on-orbit.

11. **Modal surveys.** When initially comparing model and experimental data, it is not always immediately clear which measured modes match which simulated

modes. In this case, a modal survey should be performed whereby accelerometers are distributed on the testbed in order to obtain experimental modeshapes. It is difficult to mount accelerometers on all degrees of freedom of the system, so only a partial modal surveys may be performed. A full frequency sweep can be performed, but it will only be able to be partially observed, therefore the full state vector cannot, in general, be measured for a complex system. If only a limited number of accelerometers are available, the relative magnitude of each measured degree of freedom can be compared to the simulated modeshape in order to match simulated and measured modes. The metrics used for this type of analysis are the Modal Assurance Criterion (MAC) and the Coordinate Modal Assurance Criterion (COMAC). Further discussion of matching modeshapes is in section 2.2.

12. **Damping.** Damping is important, yet difficult to measure on structural systems. One way to measure damping is to apply an impulse disturbance to the system. From this decay envelope of the response, the damping ratio can be estimated. This is difficult because in order to measure damping completely, it must be measured for each input-output pair. For that reason, several bodies of work present damping estimation methods. Ewins [14] presents a frequency dependent damping schedule. Since damping in complex linear systems is usually introduced as modal damping, and since this affects primarily the modal magnitudes, it will be addressed in Chapter 3, where it can be set using optimization.

13. **Interfaces.** Energy transmission is often idealized in the simulation model. For example, bars in a truss are generally modelled as rigidly connected to each other at the nodal points. However, these connections often involve nuts and bolts, which have their own stiffness and other material properties. For controlled systems, the impedance of the global structure can be matched to the controller using the controller hardware. This is further described in Blaurock [5].

2.2 Metrics for Model Matching

The extent of model/experiment matching will primarily be evaluated using frequency response or transfer functions because they are easy to obtain using data acquisition systems. Some alternatives include time domain output responses. The transfer function was chosen here because it incorporates the input and output dynamics in the frequency domain; this provides an easy way to analyze a large amount of data. Also, noise can be averaged out of the system easily when working in the frequency domain. Two good references on the process of obtaining frequency domain data are Ewins [14] and Maia [37]. An integrated model is generally represented in state space, and the transfer function is easily obtained from the state space representation using [17]:

$$G\left(\omega\right) = C\left(j\omega I - A\right)^{-1} B + D. \tag{2.5}$$

In the above equation, G represents the complex transfer function and A, B, C and D are the state space matrices. Before moving on to automated model tuning, there should be some degree of matching between the data and the analytical model. First of all, the low and high frequency asymptotes of the transfer function(s) should align. This check can be performed on visual inspection. Secondly the slope of the transfer function backbone should match. This is called the dereverberated transfer function backbone, and it is the transfer function of the local effects (underlying trend) while ignoring the reverberant (far-field) effects. The backbone can be calculated using wave theory, by cepstral analysis, or by smoothing the transfer function [11]. In wave theory, ignoring the far-field effects means that the wave is not reflected. Cepestral analysis finds the impulse response before the reflections occur. The easiest way to obtain the experimental and simulation transfer function backbone is smoothing the transfer functions using MATLAB® or by increasing the damping in the simulation model. By increasing the modal damping in the frequency response model to a high value (close or equal to unity), the resonances are effectively removed, leaving only the backbone. Next the sequence of poles and zeros should be compared. This is an easy task for a single-input/single-output (SISO) system, but for a MIMO system, examining the pole/zero sequence involves examining the transfer function of each input/output pair, a very involved job. Figure 2-6 shows the pole/zero maps for a SISO system and for a MIMO system. In the SISO case, a expanding circle with radius of ω will capture the pole/zero sequence as ω goes from the frequency lower bound to the frequency upper bound. In the MIMO case, however, as ω increases, the circle captures different input/output (I/O) pairs (corresponding to different transfer functions). The poles and zeros in a MIMO system must be identified to a specific I/O pair in order to identify the correct pole/zero sequence.

If the pole/zero sequences line up, then a comparison of the modal frequencies can take place. A metric that measures the degree of mismatch for these modal frequencies is:

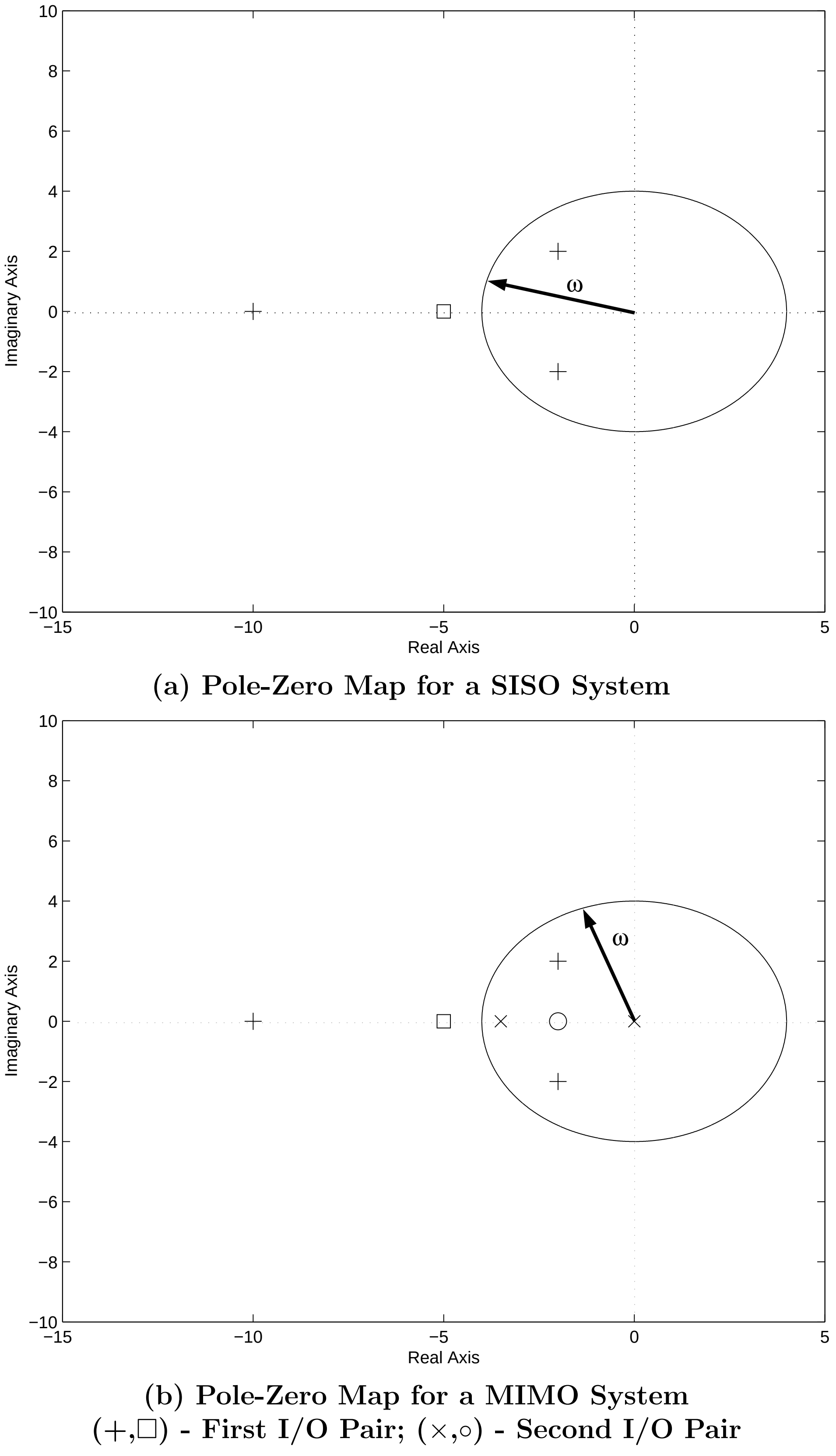
$$\left|\xi = \frac{\hat{\omega}_i - \omega_i}{\omega_i}\right|. \tag{2.6}$$

Here i is the modal number, $\hat{\omega}$ is the measured modal frequency, and ω is the analytical modal frequency. This metric should be small compared to one (below about 0.10 is a good rule of thumb) for every mode in the frequency range of interest.

The magnitude of each mode depends highly on the damping properties specified in the model. Since it is very difficult to model this damping physically, we will instead use modal damping, and address this aspect further in Step 2: Automated Tuning.

One could also employ time-domain data or modeshape information to evaluate model/experimental matching. Raw time-domain data, however, is difficult to compare to finite element matrices. Complete modeshape data is near impossible to collect on large systems in the practical sense, especially for rotational degrees of freedom. There are modal expansion techniques that make use of the available and analytical modeshapes in order to produces a complete set of modeshape data (Lipkins and Vanderurzen [35] and Harder [24]). A popular approach is called Kidder's method, described in detail in Kidder [30], and outlined here. The general eigenvalue problem,

$$\left\{[K] - \omega_i^2 [M]\right\} [\phi_i] = [0], \tag{2.7}$$



(a) Pole-Zero Map for a SISO System

(b) Pole-Zero Map for a MIMO System
(+,□) - First I/O Pair; (×,○) - Second I/O Pair

Figure 2-6: Example Pole-Zero Maps

(where K and M are the analytical mass and stiffness matrices, ω is the modal frequency, ϕ is the modeshape vector, and i is the mode number) can be partitioned into primary and secondary degrees of freedom.

$$\left\{ \begin{bmatrix} K_{pp} & K_{ps} \\ K_{sp} & K_{ss} \end{bmatrix} - \omega_i^2 \begin{bmatrix} M_{pp} & M_{ps} \\ M_{sp} & M_{ss} \end{bmatrix} \right\} \begin{bmatrix} \phi_p \\ \phi_s \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{2.8}$$

The primary degrees of freedom (denoted by subscript p) are the measured modes, and they direct the motion of the secondary degrees of freedom (denoted by the subscript s) which are the unmeasured modes. The secondary modes are then determined using the second equation:

$$[\phi_s]_i = -\left[[K_{ss}] - \omega_i^2 [M_{ss}] \right]^{-1} \left[[K_{sp}] - \omega_i^2 [M_{sp}] \right] [\phi_p]_i \tag{2.9}$$

The second equation is used because the first equation has a rank deficient coefficient matrix for the case where there are more unmeasured degrees of freedom than there are measured degrees of freedom (which is generally the case).

Once the complete modeshape data is assembled (measured or expanded), the experimental and analytical modeshapes can be compared. One method is to check the orthogonality condition:

$$[\phi]^{\mathrm{T}} [M] [\phi] = [I] \tag{2.10}$$

where $[I]$ is the identity matrix and $[\phi]$ is the matrix of mass normalized eigenvectors. The experimental eigenvectors are substituted in 2.10, and the deviation from the identity matrix is observed. If the diagonal values are near unity for a particular mode, then that mode can be considered correlated to the respective analytical mode.

Another method of comparing modeshapes is called the Modal Assurance Criterion (MAC) [3]:

$$\mathrm{MAC}\left([\phi_X]_i [\phi]_j \right) = \frac{\left| [\phi_X]_i^{\mathrm{T}} [\phi]_j \right|^2}{\left([\phi_X]_i^{\mathrm{T}} [\phi_X]_i \right) \left([\phi]_j^{\mathrm{T}} [\phi]_j \right)} \tag{2.11}$$

where the subscript X identifies the experimental modeshape. MAC values near unity identify highly matched modes. If the values for MAC are poor (near zero), then the error can be physically localized using the Coordinate Modal Assurance Criterion (COMAC):

$$\mathrm{COMAC}\left(i\right) = \frac{\left(\sum\limits_{j=1}^{nm} \left|\phi\left(i,j\right)\phi_X\left(i,j\right)\right|\right)^2}{\sum\limits_{j=1}^{nm} \left|\phi\left(i,j\right)\right|^2 \sum\limits_{j=1}^{nm} \left|\phi_X\left(i,j\right)\right|^2} \tag{2.12}$$

The correlation at each degree of freedom is averaged over each set of correlated mode pairs. Degrees of freedom with a low value of COMAC indicate regions in the system where there is poor experimental/analytical correlation.

2.3 Role of Parameter Uncertainty Analysis

There are several sources in the literature that deal with parameter uncertainty. Hasselman [25] provides uncertainties on mass and stiffness uncertainties, while Simonian [47] provides damping uncertainties. Two that apply particularly to the space structure problem are Bourgault [8] and Campbell [10]. The process of parameter uncertainty propagation is to provide ranges of uncertainties for input parameters in order to predict a range of performance values. Besides the identifiable sources of error presented already in this chapter and in the next chapter, more uncertainty can occur in the system due to unmodeled disturbances and non-linearities. It is assumed that all initially unmodeled dynamics will be added to the model, and that the uncertain parameters will be identified and tuned in Step 2. Uncertainty analysis is extremely useful when there is no testbed available, and a performance estimate must be made in order to evaluate a particular design. However, when a testbed is available, this three step process can be applied to significantly reduce non-parametric and parametric error.

2.4 Variability Considerations

The day to day variations present particular problems when dealing with hardware. Temperature variation, outside disturbances, and gradual loosening of bolts are some things that contribute to variations in the test data and thereby experiment/model mismatch. For these reasons, data is stochastic in nature, and it might be required to take multiple data sets, and to average them. In general the mean of the experimental data should be predicted by the model. To combat the loosening of the structure, a glue like Loc-Tite can be used to set the nuts and bolts into place. A good practice is to tighten all the connections on the structure before taking data. This might take a long time and be tedious, but it will produce a much cleaner signal.

2.5 Overall Implementation of Coarse Tuning

The corrections required to eradicate non-parametric error will differ from system to system, and it is important that the modeler understand the changes in the model that result from each of these adjustments. In addition, coarse tuning is fundamentally an iterative process. After the modeler implements one of the listed items and a change is made to the simulation model, the performance and behavior of the system should be compared to the previously unaltered model. The changes and the results should be recorded, since the evolution of the model will become important in evaluating model fidelity, and how close the model comes to approximating the actual physical characteristics of the system. This becomes important in Step 3, model tracking, because a model that represents the system in a physical way will be useful in optimization analysis. If one of the changes described in this chapter is applied to the model, and this change worsens the model/experimental mismatch in the short-term, it should not be immediately rejected. The change should be checked to make sure it is physically correct, and if so, then the change should be retained. There is, most likely, another mechanism in the model which is causing the mismatch. A small amount of investigative work in the beginning could save the modeler time over blindly going through these model checks one by one. For example, an incorrect low frequency asymptote most probably indicates incorrectly modelled boundary conditions, and this is the modeling error that should be addressed first.

The overall goal and output of this step should be a system that has a matching transfer function roll-off, pole/zero sequence, and high and low frequency asymptotes. The frequencies of the important modes should be similar. A ten percent margin of error is a good, but undocumented number to use here. Only then is the system suitable for Step 2, automated tuning.

2.6 Application to a sample problem

2.6.1 Description of Nominal Model

The sample problem presented here consists of one beam, one spring, and two concentrated mass elements. They are arranged as in Figure 2-7. The beam is a 0.2 meter long solid tube (3 cm o.d.) and is made from the Al-1100 alloy ($\rho = 2710 {}^{kg}/_{m^3}$, E = 70.0 GPa, G = 26.0 GPa). The beam is securely fastened to a wall at node 101 (cantilevered). The spring is 0.10 m long with a linear spring constant of 30.0 kN/m. The concentrated mass at nodes 102 and 103 are 10.0 kg and 2.0 kg, respectively. The system is constrained so that there are twelve degrees of freedom: all six degrees of freedom at nodes 102 and 103. Node 101 is completely constrained since is is fastened to the wall. The NASTRAN bulk data deck can be seen in Appendix A.1. Non-parametric error will be introduced to this model in the following sections to illustrate the effects of such errors. A summary can be seen at the end of this section in Table 2.3.

2.6.2 Nominal Statics

This system was first modelled using the NASTRAN finite element code. The NASTRAN data deck for this model can be seen in Appendix A.1. The total mass of the system is calculated as the mass at nodes 102 and 103 plus the mass of the beam (the

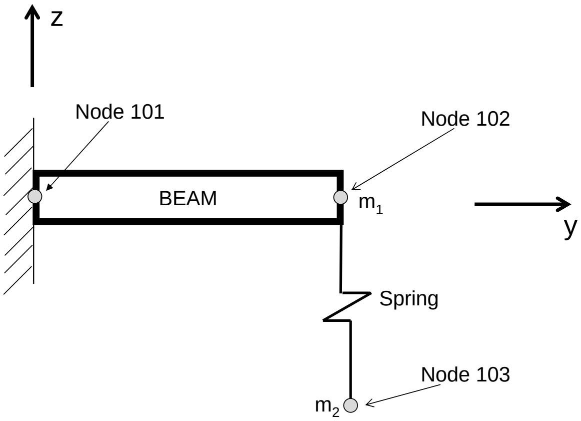


Figure 2-7: Sample Problem Configuration

spring is assumed to be massless). The mass of the beam is just the volume of the beam times the density. The calculation is shown below. The total mass reported by NASTRAN was 12.38 kg, which agrees with the design calculation.

$$\begin{aligned} Mass_{total} &= mass_{node102} + mass_{node103} + mass_{beam} \\ &= 10kg + 2kg + A_{X-\sec} L_{beam} \rho_{Al} = 12.38kg \end{aligned}$$

Due to gravity, the beam bends slightly. The spring elongates due to gravity acting on the mass at node 103. The deformed result can be seen in Figure 2-8. The dashed and the solid lines are the undeformed and deformed systems, respectively. The node numbers are also shown in this figure. The spring elongation reported by NASTRAN is $6.54 * 10^{-4} m$. This agrees with the analytical solution:

$$\begin{aligned} & k = 30,000 {}^{N}\!/_{m} \\ & F = 2kg * 9.81 {}^{m}\!/_{s^2} = 19.62N \\ & elongation = F/k = 6.54 * 10^{-4} m \end{aligned}$$

The beam tip displacement due to the static gravity load was calculated by NASTRAN to be $1.1459 * 10^{-4}$m. The closed form solution for the beam tip displacement [18] is:

$$\delta_B = \frac{WL^3}{8EI} + \frac{PL^3}{3EI} = 1.1414 * 10^{-4}\text{m}$$

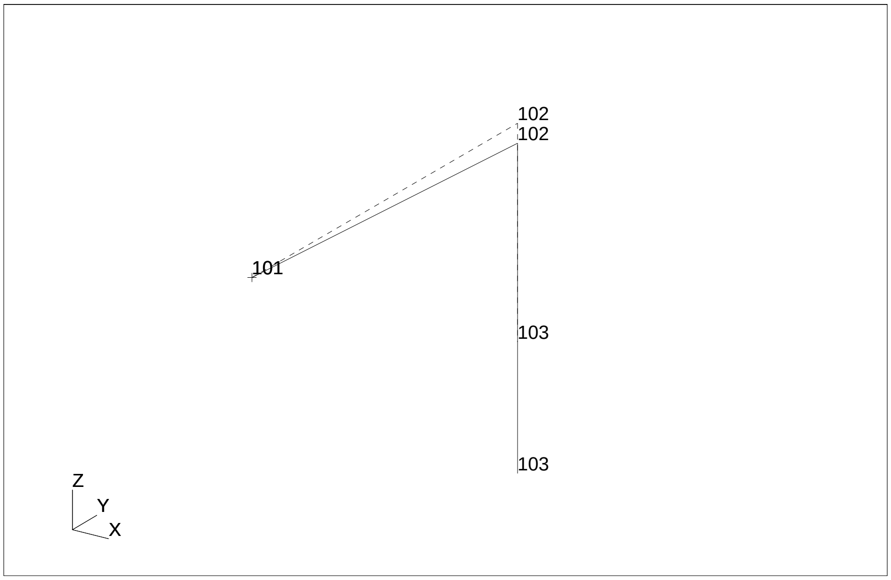


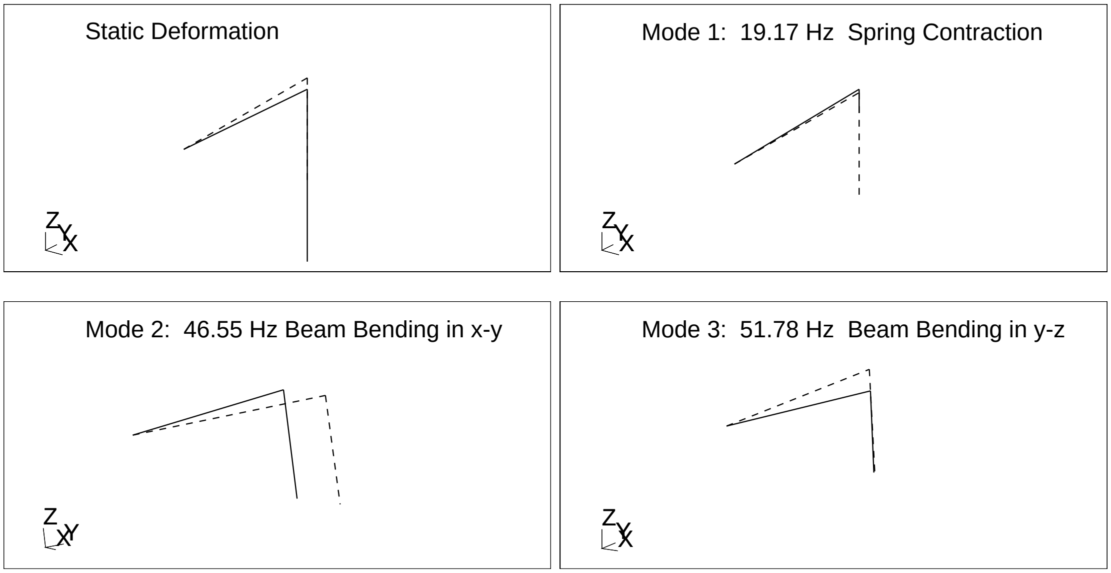
Figure 2-8: Static Deformation of Sample Problem

where W is the weight of the uniform beam, P is the force due to the concentrated masses, and L is the length of the beam. It can readily be seen that the analytical and finite element solution match well for the beam displacement (to within 0.40%).

2.6.3 Nominal Dynamics

There are initially many modes in the continuous structure. Since, however, the beam and the spring were modelled as finite elements with a certain discretization and since the eigenanalysis only identified modes below 500 Hz, only a certain amount of modes are available in the analysis. The number of eigenvalues in the model is the same as the size of the K and M matrices, and when the eigensolution is run only in a specified bandwidth is evaluated (0.1 to 500 Hz in this case). The eigenanalysis reveals three normal modes (19.17 Hz, 46.55 Hz, 51.78 Hz). The first mode (19.17 Hz), shows the extension and contraction of the spring. Most (96.24%) of the strain energy lies in the spring element for this mode. This modeshape for the second mode (46.55 Hz) shows the beam bending in the *x*-*y* plane. At this frequency 99.91% of the strain energy is in the beam element. The third mode (51.78 Hz) involves the beam bending in the

y-z plane (96.24% of the strain energy is in the beam element). These modeshapes can be seen in Figure 2-9. The dashed and the solid lines are the undeformed and deformed systems, respectively.


Static Deformation

Mode 1: 19.17 Hz Spring Contraction

Mode 2: 46.55 Hz Beam Bending in x-y

Mode 3: 51.78 Hz Beam Bending in y-z

Figure 2-9: Nominal Modeshapes of Sample Problem

A comparison to the finite element eigenanalysis can be made by finding the analytical modal frequencies and modeshapes. Blevins [7] provides formulas for modal frequencies and modeshapes for a large variety of simple systems. An exact replication of this sample system is not available in Blevins, but the modes described above can be approximated using a combination of similar systems. The first mode (spring contraction) can be approximated by a simple two degree of freedom mass spring system which can be seen in Figure 2-10(a). The first spring ($k_1 = 1031.37{}^{kN}\!/_{m}$) represents the linear beam, and is found by dividing the weight of the concentrated masses by the static tip deflection. The mass at Node 102 is shown as m_1, m_2 is the mass at Node 103, and k_2 is the spring constant. The equation for the natural frequency of this system is:

$$f(Hz) = \frac{1}{2^{3/2}\pi} \left\{ \frac{k_1}{m_1} + \frac{k_2}{m_1} + \frac{k_2}{m_2} - \left[\left(\frac{k_1}{m_1} + \frac{k_2}{m_1} + \frac{k_2}{m_2} \right)^2 - \frac{4k_1k_2}{m_1m_2} \right]^{1/2} \right\}^{1/2} \tag{2.13}$$

The equation for the corresponding modeshape is:

$$\begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 + \frac{k_1}{k_2} - \frac{m_1}{k_2}(2\pi f) \end{bmatrix} \tag{2.14}$$

Substituting in the appropriate values gives the natural frequency as 19.17 Hz, and the modeshape as $\begin{bmatrix} 1 \\ 30.54 \end{bmatrix}$. The natural frequency is the same value (accurate to two decimal places) as was found in the finite element eigenanalysis. Since this modeshape describes the movement of the masses in the z-direction only, it is appropriate to compare them with the z-direction translational degree of freedom (T3) in the nominal model. Scaling this modeshape to the value of T3 for Mode 1 and Node 102 (2.29E-2), the analytical and nominal scaled modeshapes of T3 for Mode 1, Nodes 102 and 103 can now be compared, and they are:

$$\begin{bmatrix} 2.29\text{E}-2 \\ 6.99\text{E}-1 \end{bmatrix}_{\text{Analytical}} ; \begin{bmatrix} 2.29\text{E}-2 \\ 7.05\text{E}-1 \end{bmatrix}_{\text{Nominal}} . \tag{2.15}$$

These modeshapes are very close to each other (0.8% difference).

The second and third modes describe beam bending in axes perpendicular to the beam axis, and can be approximated as a cantilevered beam with a concentrated mass at the free end (see Figure 2-10(b)). The equation for this natural frequency is

$$f(Hz) = \frac{1}{2\pi}\sqrt{\frac{3EI}{L^3(m + 0.24m_b.)}} \tag{2.16}$$

When substituting in the sum of the concentrated masses (12 kg) at nodes 102 and 103 in the above equation for m, the natural frequency is calculated to be 46.76 Hz.

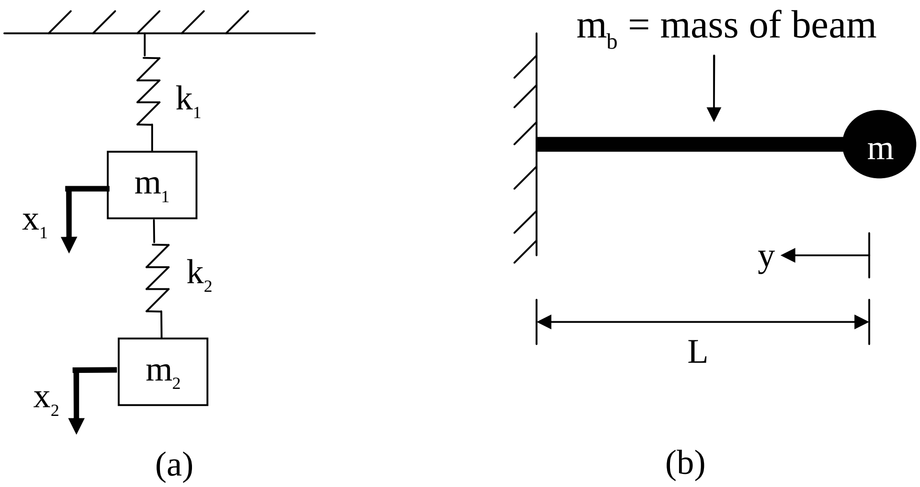


Figure 2-10: Sample Problem Approximations

When only using the mass at node 102 (10 kg), the frequency is 51.18. These are very close to the modal frequencies from the finite element model. The modeshape, $\tilde{x}(y)$, is described by the function

$$\tilde{x}(y) = \left(\frac{y}{L}\right)^3 - 3\left(\frac{y}{L}\right) + 2 \tag{2.17}$$

This modeshape describes beam movement in the x-y and y-z planes, so the modeshape could be labelled as $\tilde{x}(y)$ or $\tilde{z}(y)$. Since Node 102 is at the tip of the beam, $y = 0$ in this case, and the modeshape value for T1 and T3 for Modes 2 and 3, respectively are both 2 (scaled by 10, it can be 0.2). If it is assumed that the spring moves a negligible amount in these modes, then the values are the same for Node 103. These values can be seen in Table 2.4. A major discrepancy occurs between the analytical and the nominal modeshapes for T3 in Mode 3 for both Nodes 102 and 103. The nominal value for Node 103 is 5.16E-2, about 16% of the Node 102 nominal value. In the analytical case, they are the same. This discrepancy is most likely due to the spring moving a non-negligible amount in Mode 3. Unfortunately, there is not an obvious way to calculate analytically the value for the x-direction rotation (R1) for Modes 1 and 3 or the the z-direction rotation (R1) for Mode 2.

The axial beam mode can be found via the equation

$$\begin{aligned} &f(Hz) = \tfrac{\lambda}{2\pi L}\sqrt{\tfrac{E}{\rho}} \\ &\text{where } \cot\lambda{=}\tfrac{\mathrm{m}}{\mathrm{m_b}}\lambda. \end{aligned} \tag{2.18}$$

For this system, it is calculated to be 808.9 Hz, which is outside of the bandwidth of interest (0.1 to 500 Hz).

2.6.4 Sample Problem Errors

In this section, a series of errors will be introduced to the sample problem in order to demonstrate the importance of applying engineering insight to a newly modelled system. Only a few of the previously enumerated model heuristics will be shown on this sample problem. Please refer to Summary Table 2.3 that contains the relevant data for the sample problem and for all the subsequent models with induced error.

1. **Typographical errors.** It is very easy to make an error when inputting values into a model, especially for complex systems. One error that can be made is the misplacement of a decimal point. By decreasing the spring constant by a factor of ten, the static elongation of the spring due to the gravity load increased by a factor of ten (since the force, that is the weight of the concentrated mass at node 103 remained constant) . The first mode (spring contraction mode) was reduced by 67.9% and the third mode was reduced by only 1.5%. The second mode was unchanged. The large amount of change observed in the first mode is consistent with intuition since most of the strain energy for that mode resided in the spring element.

2. **Mass and Inertia.** The total mass of the system calculated by NASTRAN was compared against the separate hand calculation in Section 2.6.2 and was found to agree. The moments of inertia that were input into this system to describe the inertia of the beam about its axes were determined using the ana-

lytical formula for circular cross-sections [4]:

$$\begin{aligned} I_X = I_Y = \tfrac{1}{4}\pi r^4 \\ J = \tfrac{1}{2}\pi r^4 \end{aligned} \tag{2.19}$$

If, however, the cross section of the beam were an ellipse, the moments of inertia would be calculated as

$$\begin{aligned} I_X &= \tfrac{1}{4}\pi a b^3 \\ I_Y &= \tfrac{1}{4}\pi a^3 b \\ J &= \tfrac{1}{4}\pi a b(a^2 + b^2) \end{aligned} \tag{2.20}$$

where a and b are the major and minor axes of the ellipse. Keeping the cross sectional area and the length of the beam the same, the moments of inertia in the beam were changed to represent an elliptical cross section (a = 2 cm, b = 1.125 cm). The results of this change show that the spring elongation remained exactly the same, whereas the static tip displacement increased by about 17%. The first modal frequency and modeshape remained relatively unchanged. There was, however, a significant shift in the second and third modes and modeshapes. The modeshapes and the corresponding modal frequencies can be seen in Figure 2-11. The dashed line shows the undeformed system, and the solid line shows the modeshape. Instead of the beam bending being exclusively in the x-y plane for the second mode and in the y-z planes for the third mode as in the nominal model, it can be seen that the beam is now bending in both planes for both modes. The change to an elliptical cross section introduced modal coupling.

3. **Boundary Conditions** In order to demonstrate the importance of proper boundary condition modeling, the rotational degrees of freedom are removed from node 102 in the nominal model. Because the identified modeshapes in the nominal model have rotational components, there will be reaction forces located at this degree of freedom. For that reason, the static beam tip displacement due to gravity is reduced to $2.8647 * 10^{-5}$, an error of about 75%. Also, only

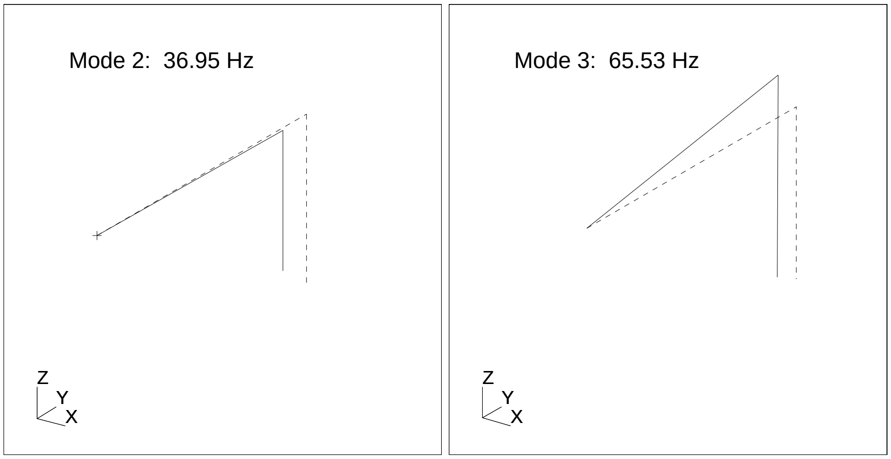


Figure 2-11: Second and Third Modes Using Elliptical Cross-Section

two modes are reported in the finite element analysis, one at 19.42 Hz and one at 102.24 Hz. It can be seen from Figure 2-12 that the first mode is the same shape as the first mode in the nominal model, that of spring contraction. The second mode involves beam bending exclusively in the y-z plane (corresponding to the third mode in the nominal model). Furthermore, its modal frequency has almost doubled. The modeshape that had bending only in the x-y plane has completely disappeared.

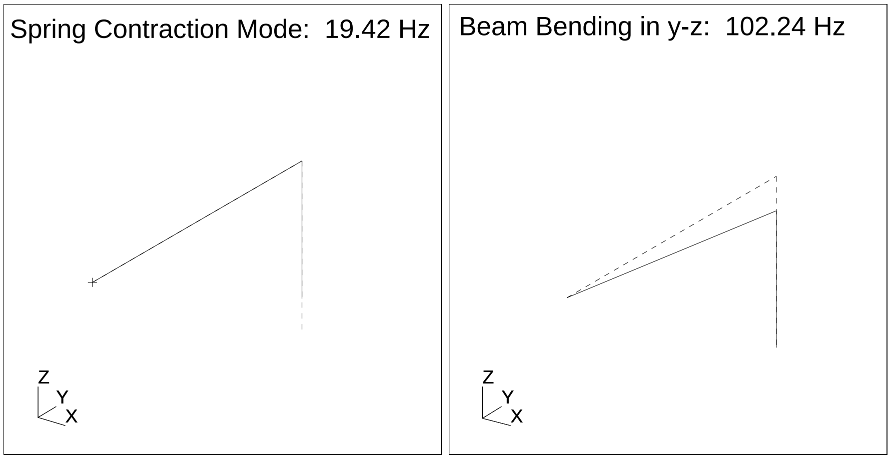

Spring Contraction Mode: 19.42 Hz
Z
Y
X
Beam Bending in y-z: 102.24 Hz
Z
Y
X

Figure 2-12: First and Third Modes Imposing New Boundary Conditions

4. **Gravity effects.** Since there is a relatively large weight at the end of this cantilevered beam (10.0 kg at node 102 and 2.0 kg at node 103) and since it is known that this beam has a static tip deflection (0.06% of the beam length), one can assume that the force of gravity plays a part in the static and dynamic properties of the system. When the gravity load is removed, there is no initial static displacement of the beam or elongation of the spring. The modal frequencies stay about the same, however, the modeshapes do change slightly. Table 2.2) shows the eigenvector for node 102 for all three modes (the eigenvector does not change significantly for nodes 101 and 103). Only slight changes in the second translational and rotational degrees of freedom (y-axis) are observed. Since there is a non-zero term for the second rotational degree of freedom (R2) in the second mode for the nominal case, this indicates that the beam twists to some extent due to the presence of gravity (this entry is zero for the no gravity case). Similarly, the values for the second translational degree of freedom (T2) in the first and third nominal case modes are non-zero, indicating a contraction of the beam (these entries are zero in the no gravity case). Therefore, the presence of loads such as gravity can affect the properties of the system. Further investigation of the effect of gravity on other types of systems can be found in Glaese [19] and Rey [44].

System	T1(m)	T2(m)	T3(m)	R1(rad)	R2(rad)	R3(rad)
Mode 1: Nominal	0.0	**-5.53E-08**	0.0229	0.171	0.0	0.0
Mode 1: No Gravity	0.0	**0.0**	0.0229	0.171	0.0	0.0
Mode 2: Nominal	0.286	0.0	0.0	0.0	**-8.28E-04**	-2.15
Mode 2: No Gravity	0.286	0.0	0.0	0.0	**0.0**	-2.15
Mode 3: Nominal	0.0	**-7.59E-07**	-0.312	-2.34	0.0	0.0
Mode 3: No Gravity	0.0	**0.0**	0.312	2.34	0.0	0.0

Table 2.2: Comparison of Eigenvector at Node 102

5. **Discretization** One way to gauge the fidelity of a FE model is via the size and number of the finite elements (its discretization level). In this sample problem, a relatively long beam is represented by only one element. Therefore,

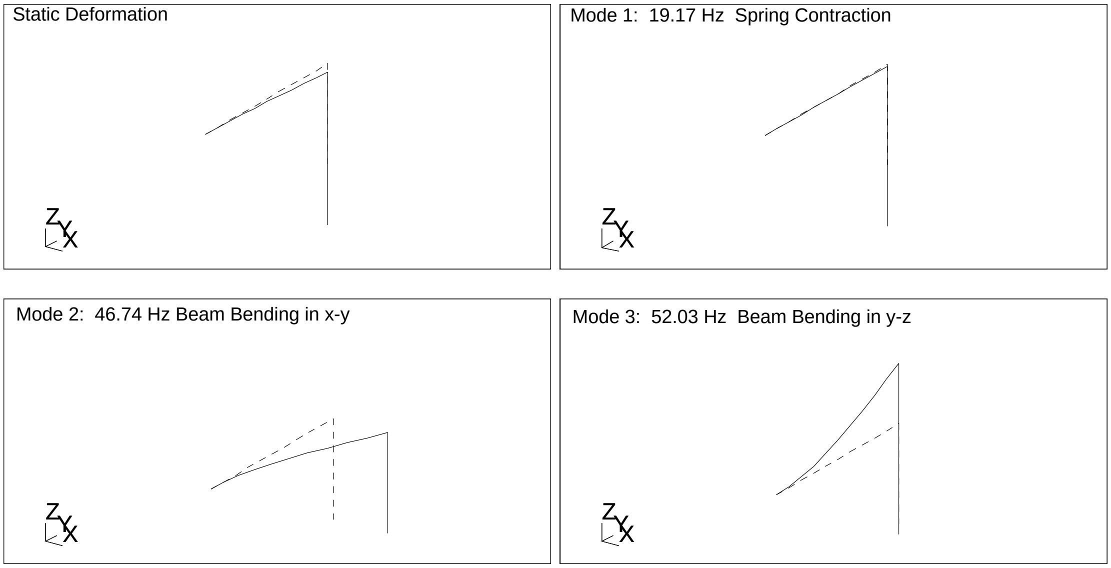

Static Deformation
Mode 1: 19.17 Hz Spring Contraction
Mode 2: 46.74 Hz Beam Bending in x-y
Mode 3: 52.03 Hz Beam Bending in y-z
Z
Y
X

Figure 2-13: Curved Modeshapes for Discretized System

the beam itself cannot display curvature of any type. Once the beam is split up into ten sections, the curvature of the beam in the second and third modes becomes apparent (see Figure 2-13). Again, the dashed and the solid lines are the undeformed and deformed systems, respectively. This curvature provides much more information about the behavior of the beam over the bandwidth of the eigensolution (0.1 to 500 Hz). Also, the discretized model produced a static beam tip displacement value that was closer to the analytical result. Moreover, the modal frequencies were further refined.

2.7 Chapter Summary

When developing initial models for physically existing systems it is crucial that the modeler applies engineering insight (also called coarse tuning) to the model. This chapter has listed the major areas where engineering insight can be applied and has described heuristically how to implement them. Also presented in this chapter was the description and analysis of a simple sample problem. This system was modified in

Model	Analytical	Nominal	Typo	Inertia	B.C.'s	No Gravity	Discretized
Static Beam Tip Disp. (m)	$1.1414 * 10^{-4}$	$1.1458 * 10^{-4}$	$1.1459 * 10^{-4}$	$1.3409 * 10^{-4}$	$2.8647 * 10^{-5}$	0.0	$1.1414 * 10^{-4}$
Static Spring Elong. (m)	$6.5400 * 10^{-4}$	$6.5400 * 10^{-4}$	$6.5400 * 10^{-3}$	$6.5400 * 10^{-4}$	$6.5400 * 10^{-4}$	0.0	$6.5400 * 10^{-4}$
Normal Modes (Hz)	19.49 46.76 51.18	19.17 46.55 51.78	6.16 46.55 51.01	19.09 36.95 65.53	19.42 - 102.24	19.17 46.55 51.78	19.17 46.74 52.03

Table 2.3: Summary Table for Nominal and Error-Induced Models

Modes	Node	T1(m)	T2(m)	T3(m)	R1(rad)	R2(rad)	R3(rad)
Mode 1	102	0.0	0.0	2.29E-2	-	0.0	0.0
19.17 Hz	103	0.0	0.0	6.99E-1	-	0.0	0.0
Mode 2	102	0.2	0.0	0.0	0.0	0.0	-
46.76 Hz	103	0.2	0.0	0.0	0.0	0.0	-
Mode 3	102	0.0	0.0	0.2	-	0.0	0.0
51.18 Hz	103	0.0	0.0	0.2	-	0.0	0.0

Table 2.4: Analytical Modeshapes

five different ways corresponding to five different model heuristics. The consequences of these changes were explained and displayed. On more complex systems these effects can only be magnified. The sample problem demonstrated the need for the modeler to apply engineering insight to an initial model. In the next chapter, experimental and model transfer functions will be compared.

Modes	Node	T1(m)	T2(m)	T3(m)	R1(rad)	R2(rad)	R3(rad)
Mode 1	102	0.0	-5.53E-8	2.29E-2	1.71E-1	0.0	0.0
19.17 Hz	103	0.0	-5.53E-8	7.05E-1	1.71E-1	0.0	0.0
Mode 2	102	-2.86E-1	0.0	0.0	0.0	-8.28E-4	2.145963
46.55 Hz	103	-2.88E-1	0.0	0.0	0.0	-8.28E-4	2.145963
Mode 3	102	0.0	7.59E-7	-3.12E-1	-2.343053	0.0	0.0
51.78 Hz	103	0.0	7.64E-7	5.16E-2	-2.343053	0.0	0.0

Table 2.5: Nominal Modeshapes

Modes	Node	T1(m)	T2(m)	T3(m)	R1(rad)	R2(rad)	R3(rad)
Mode 1	102	0.0	-4.97E-9	2.06E-3	1.54E-2	0.0	0.0
6.16 Hz	103	0.0	-4.97E-9	7.07E-1	1.54E-2	0.0	0.0
Mode 2	102	-2.86E-1	0.0	0.0	0.0	-8.28E-4	2.15
46.55 Hz	103	-2.88E-1	0.0	0.0	0.0	-8.285E-4	2.15
Mode 3	102	0.0	7.61E-7	-3.13E-1	-2.35	0.0	0.0
51.01 Hz	103	0.0	7.66E-7	4.642E-3	-2.35	-0.0	0.0

Table 2.6: Modeshapes with Typographical Error

Modes	Node	T1(m)	T2(m)	T3(m)	R1(rad)	R2(rad)	R3(rad)
Mode 1	102	1.77E-2	-7.08E-8	2.93E-2	2.20E-1	4.39E-5	-1.33E-1
19.09 Hz	103	1.78E-2	-7.09E-8	7.037E-1	2.20E-1	4.39E-5	-1.33E-1
Mode 2	102	-2.21E-1	4.76E-7	-1.96E-1	-1.47	-5.46E-4	1.66
36.95 Hz	103	-2.22E-1	4.78E-7	7.57E-2	-1.47	-5.46E-4	1.66
Mode 3	102	-1.81E-1	-5.91E-7	2.42E-1	1.82	-4.472E-4	1.36
65.53 Hz	103	-1.832E-1	-5.97E-7	-2.35E-2	1.82	-4.472E-4	1.36

Table 2.7: Modeshapes with Inertia Change

Modes	Node	T1(m)	T2(m)	T3(m)	R1(rad)	R2(rad)	R3(rad)
Mode 1	102	0.0	-1.27E-8	5.23E-3	0.0	0.0	0.0
19.42 Hz	103	0.0	-1.27E-8	7.07E-1	0.0	0.0	0.0
Mode 3	102	0.0	7.73E-7	-3.13E-1	0.0	0.0	0.0
102.24 Hz	103	0.0 7.946894E-7	1.18E-2	0.0	0.0	0.0	

Table 2.8: Modeshapes with Changed B.C.

Modes	Node	T1(m)	T2(m)	T3(m)	R1(rad)	R2(rad)	R3(rad)
Mode 1	102	0.0	0.0	2.29E-2	1.71E-1	0.0	0.0
19.17 Hz	103	0.0	0.0	7.05E-1	1.71E-1	0.0	0.0
Mode 2	102	2.86E-1	0.0	0.0	0.0	0.0	-2.15
46.55 Hz	103	2.88E-1	0.0	0.0	0.0	0.0	-2.15
Mode 3	102	0.0	0.0	3.12E-1	2.34	0.0	0.0
51.78 Hz	103	0.0	0.0	-5.16E-2	2.34	0.0	0.0

Table 2.9: Modeshapes without Gravity

Modes	Node	T1(m)	T2(m)	T3(m)	R1(rad)	R2(rad)	R3(rad)
Mode 1	102	0.0	5.50E-8	-2.28E-2	-1.71E-1	0.0	0.0
19.17 Hz	103	0.0	5.50E-8	-7.05E-1	-1.71E-1	0.0	0.0
Mode 2	102	2.87E-1	0.0	0.0	0.0	8.25E-4	-2.15
46.74 Hz	103	2.89E-1	0.0	0.0	0.0	8.25E-4	-2.15
Mode 3	102	0.0	-7.60E-7	3.14E-1	2.35	0.0	0.0
52.03 Hz	103	0.0	-7.65E-7	-5.13E-2	2.35	0.0	0.0

Table 2.10: Modeshapes of Discretized Beam

Chapter 3

Step 2: Automated Tuning

By going through the model checks in Step 1, it is assumed that all (or most) of the error left in the system is due to parametric error. Parametric error can be eliminated by changing the values of parameters already present in the system model within reasonable upper and lower bounds. All non-parametric error should have been eliminated in Step 1. The precise material and structural values of each component in the system will generally not be known a priori, and therefore the values that are set in the model are not necessarily correct. Typically the nominal catalogue values for such parameters will be used in the initial model. In the case where experimental data is available, these values are determined via an optimization problem; this process is called automated tuning. In this optimization problem the cost is a measure of experimental/model mismatch. The component parameters are varied (within upper and lower bounds) until the cost metric (amount of mismatch) is minimized. The selection of these parameters, the selection of the cost function, and the implementation of the optimization problem (A.K.A. tuning algorithm) are things that need to be carefully considered before proceeding. As in Chapter 2, the preferred data used to measure mismatch is the transfer function matrix (assuming a linear system).

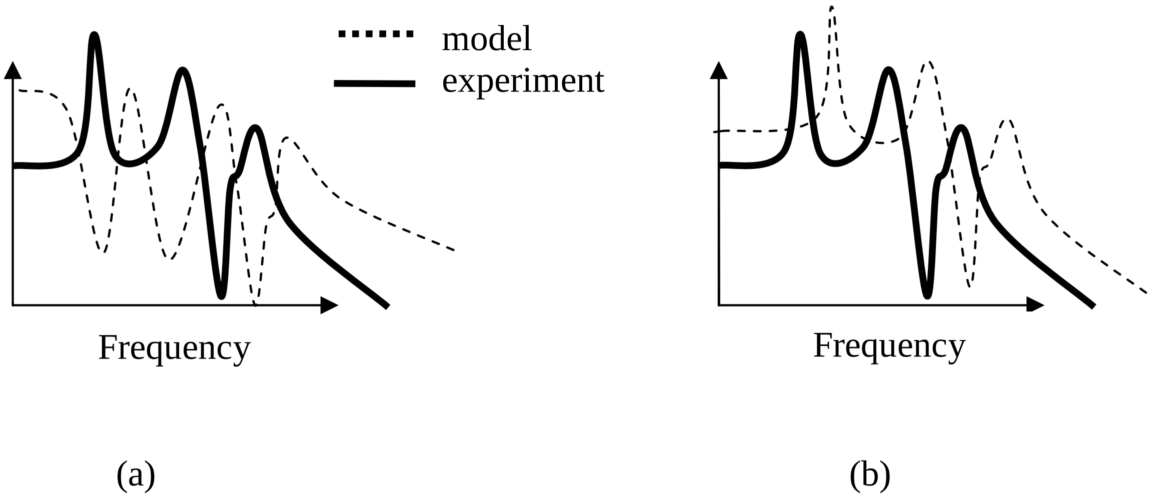


Figure 3-1: Transfer Function Starting Conditions

3.1 Model Initial Conditions

As mentioned in Section 2.2, before automated tuning is considered there must be some degree of matching between experimental data and the simulation model. For example, the high and low frequency asymptotes on the frequency response or transfer functions should match well. In addition, the frequency response roll-off (slope of the transfer function), the pole/zero sequence, and the frequencies of important modes should match well. Figure 3-1 shows a notional case of unacceptable (a) and acceptable (b) starting conditions.

3.2 Parameter Selection

In Step 1 it was mentioned that several quantities (weight, dimension, etc) should be measured to the best of the modeler's ability. Two quantities that are not very easy to measure are stiffness and damping coefficients. Therefore, these parameters are uncertain, and are usually selected for automated tuning. Other parameters that are sometimes selected are density and moments of inertia. However, density values should be considered constant because weight and volume are easily measured properties. For large, complex systems, choosing all the stiffness and damping parameters presents an unwieldy problem that is not practically solvable. Instead, a smaller subset of these parameters must be chosen. The modeler can look at the system components that contribute the most to each modal response, and select their stiffness properties. One easily obtainable measure of modal contribution of elements through finite element analysis is the strain energy. By identifying the major strain and kinetic energy contributors for each mode, elements that are likely candidates for updating can be identified. Then one can scrutinize the modeshapes and decide which elements' properties should be selected. For example, if it is known that 90% of the strain energy for, say, mode #1 resides in the longerons of a truss, then those longeron elements should be chosen. If it is determined that these longerons are mostly undergoing bending (as opposed to torsion), then the Young's modulus of the longerons should be the parameter chosen for automatic updating. If, after using these techniques, there is a large number of parameters selected for updating, a sensitivity analysis can be performed whereby only the parameters that have a significant impact on the cost (i.e. the mismatch objective function) are retained. The cost function (J) can be a vector or a scalar and is further described in Section 3.3. The sensitivity values can be obtained using finite differencing, whereby each chosen parameter is changed a small amount(Δp), and the change in the cost is observed(ΔJ). The sensitivity using forward differencing can then be written as

$$\frac{\partial J}{\partial p} \simeq \frac{J(p_0 + \Delta p) - J(p_0)}{\Delta p} = \frac{\Delta J}{\Delta p}. \tag{3.1}$$

This value can be evaluated at for each parameter, p. Since the parameters are not necessarily in the same units, the sensitivities must be normalized in order to compare them to each other. The normalized sensitivity can be written as

$$\frac{\Delta J/_J}{\Delta p/_p} \simeq \frac{p_0}{J(p_0)} \cdot \frac{\Delta J}{\Delta p}. \tag{3.2}$$

This can be interpreted as a percent change in cost per percent change in the parameter. The finite differencing approach is the most practical way of obtaining sensitivities, since the integrated system might be too complex to obtain analytical results. However, this approach becomes computationally expensive when many parameters are present. The results from finite differencing can also only be applied appropriately in a small range around the nominal parameter values.

A more rigorous and more favorable approach to sensitivity is to solve for the analytical sensitivity since the cost function can be non-linear. This is sometimes difficult to do for large, complex systems that are most easily modelled using finite elements. The analytical sensitivities of the eigenvalues and eigenvectors with respect to the chosen k parameters, p, are given by Equations 3.3 and 3.4, respectively [36].

$$\frac{\partial\left(\omega\right)_i^2}{\partial p_k} = \left[\phi\right]_i^{\mathrm{T}}\left[\frac{\partial\left[K\right]}{\partial p_k} - \left(\omega\right)_i^2\frac{\partial\left[M\right]}{\partial p_k}\right]\left[\phi\right]_i \tag{3.3}$$

$$\frac{\partial\left[\phi\right]_i}{\partial p_k} = \sum_{j=1,j\neq i}^{N}\frac{\left[\phi\right]_j\left[\phi\right]_j^{\mathrm{T}}}{\left(\omega\right)_i^2-\left(\omega\right)_j^2}\left[\frac{\partial\left[K\right]}{\partial p_k} - \left(\omega\right)_i^2\frac{\partial\left[M\right]}{\partial p_k}\right]\left[\phi\right] - \frac{1}{2}\left[\phi\right]_i\left[\phi\right]_i^{\mathrm{T}}\frac{\partial\left[M\right]}{\partial p_k}\left[\phi\right]_i \tag{3.4}$$

Using Equation 3.5, $\frac{\partial\omega_i^2}{\partial p_k}$ can be converted to $\frac{\partial\omega_i}{\partial p_k}$.

$$\frac{\partial\omega_i}{\partial p_k} = \frac{1}{2\omega_i}\frac{\partial\omega_i^2}{\partial p_k} \tag{3.5}$$

These analytical sensitivities can be easily found by supplying the system and the appropriate parameters to the DOCS function, grad_eig. The implementation of this function can be seen in Appendix B.

A problem occurs when parameters are chosen that have a similar impact on the cost. Using the eigenvalue sensitivity values calculated above, one can decompose the sensitivity matrix using singular value decomposition (SVD) in order to select parameters that have orthogonal influence on the the elements in the cost function.

First, the eigenvalue sensitivity matrix, R, is calculated.

$$\mathrm{R} = \begin{bmatrix} \frac{\partial\omega_1}{\partial p_1} & \frac{\partial\omega_1}{\partial p_2} & \cdots & \frac{\partial\omega_1}{\partial p_k} \\ \frac{\partial\omega_2}{\partial p_1} & \frac{\partial\omega_2}{\partial p_2} & & \vdots \\ \vdots & & \ddots & \\ \frac{\partial\omega_{nm}}{\partial p_1} & \cdots & & \frac{\partial\omega_{nm}}{\partial p_k} \end{bmatrix} \tag{3.6}$$

R is assumed to be rectangular matrix of size nm x k since the number of chosen parameters is generally greater than the number of modes ($k > nm$). Next the singular value decomposition is performed.

$$\mathrm{R} = U\Sigma V^T \tag{3.7}$$

This produces two orthogonal matrices, U (nm x nm) and V (k x k), and a diagonal matrix Σ (nm x k) that contains the singular values.

$$U = \begin{bmatrix} \vdots & & \vdots \\ u_1 & \cdots & u_r \\ \vdots & & \vdots \end{bmatrix}, \Sigma = \left[\begin{array}{ccc|ccc} \sigma_1^2 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_r^2 & 0 & \cdots & 0 \end{array}\right], \quad V = \left[\begin{array}{ccc|ccc} \vdots & & \vdots & \vdots & & \vdots \\ v_1 & \cdots & v_r & v_{r+1} & \cdots & v_k \\ \vdots & & \vdots & \vdots & & \vdots \end{array}\right] \tag{3.8}$$

It can be shown [50] that

$$\mathrm{R}v_i = \sigma_i^2 u_i \tag{3.9}$$

The first r vectors of V $(v_1, v_2, ...)$ contain the perturbations of the parameters in the proper order.

$$v_i = \begin{bmatrix} \Delta p_1 \\ \Delta p_2 \\ \vdots \\ \Delta p_k \end{bmatrix} \tag{3.10}$$

The product $\sigma_i^2 u_i$ describes the resultant change in the modal frequencies.

$$\mathrm{R}v_i = \underset{nm\mathrm{x}k}{\begin{bmatrix} \frac{\partial \omega_1}{\partial p_1} & \frac{\partial \omega_1}{\partial p_2} & \cdots & \frac{\partial \omega_1}{\partial p_k} \\ \frac{\partial \omega_2}{\partial p_1} & \frac{\partial \omega_2}{\partial p_2} & & \vdots \\ \vdots & & \ddots & \\ \frac{\partial \omega_{nm}}{\partial p_1} & \cdots & & \frac{\partial \omega_{nm}}{\partial p_k} \end{bmatrix}} \underset{k\mathrm{x}1}{\begin{bmatrix} \Delta p_1 \\ \Delta p_2 \\ \vdots \\ \Delta p_k \end{bmatrix}} = \sigma_i^2 u_i = \underset{nm\mathrm{x}1}{\begin{bmatrix} \Delta \omega_1 \\ \vdots \\ \Delta \omega_{nm} \end{bmatrix}} \tag{3.11}$$

The vectors u_i are unit vectors, and they are multiplied by the corresponding singular values to describe how the modes change with the perturbation changes present in the v_i vectors. Therefore the singular values (σ^2) indicate the influence that a certain combination of perturbations in the parameters have on the chosen modes (the singular values are ordered along the diagonal from greatest to least). The amount of these perturbations are in the first r vectors of V $(v_1, v_2, ...)$, where r is the rank number. The reason these vectors are used is because they comprise the row space of R. The last $k - r$ vectors in V are in the null space of R and indicate how the parameters can be changed in order to keep the chosen modal frequencies the same. The null space is not used in this analysis. The object of the SVD is to compile a set of parameters that have the influence values on the same order. If, for example, one of the singular values was very close to zero, the influence of that corresponding change in parameters (v_i) would be very small. It would be inappropriate to include that parameter change combination with one that has a singular value several orders of magnitude larger. Also important is choosing parameters in such a way that maximum row value of the U matrix is much larger than the other row entries for each row. In this way we can be sure that each linear combination of u_i vectors influences different modes uniquely. Using this SVD procedure helps to choose parameters that are influential to the system, and discards parameters that are not.

This SVD procedure aids in choosing effective parameters for automated tuning. If parameters were chosen that were not orthogonal, then the parameters could be changed in a non-unique way in order to produce the same result in the cost function. This could result in a system that was tuned in a non-physical way. A non-physical system is not useful when optimizing a system, since perturbations on that system will not be valid. This concept is further explained in Chapter 4.

Once the parameters are selected, bounds must be put on those parameters to ensure that the parameters are updated in a physically plausible way. These bounds can be chosen based on prior experience of the modeler, testing of the components, or material studies found in literature. For stiffness parameters, these bounds should generally be within $+/-10\%$ of the nominal value.

3.3 Cost/Objective Function

In the frequency domain there are two primary types of objective functions that can be employed. The first involves the modal frequencies only and the second reduces the amount of mismatch in the transfer function in the frequency range of interest. In both cases and as in Step 1, the experimental data will be taken to be the truth model. The updating functions mentioned below are available in the DOCS software package [6], and are easily implemented in the MATLAB® environment. As a first pass the modal frequency cost function should be evaluated (coarse mismatch metric). Then, once the frequencies are better lined up, the frequency response objective function can be used (finer tuning).

Other types of cost functions can include those in the time domain. Impulse response functions can give a good description of the dynamics of the system, including an estimate for damping. Time domain data can be converted into an overall RMS value for performance, and this can be compared against RMS requirements.

3.3.1 Modal Frequencies

The modal frequencies in the simulation model are easily obtained using finite element software. Experimental modal frequencies can be found by examining frequency-domain data. Due to the frequency resolution of the data acquisition, some uncertainty in the experimental frequencies can exist. The cost function can then be written as [6]:

$$J_{\text{modal}} = \sum_{i=1}^{nm} \left[\left(\omega_{\text{model}} \left(i \right) - \omega_{\text{experimental}} \left(i \right) \right) \right]^2 \tag{3.12}$$

where i is the index to the modal frequency, ω, and nm is the number of modes. The modeler should be cautious, since errors can occur when there are missing or repeated modes or when the modes cross over each other during the tuning process. Making use of the MAC or COMAC (described in Section 2.2) metrics can help to determine which experimental modes line up with which model modes, when experimental eigenvector data is available. Future work will include tracking the modes (using eigenvectors) throughout the optimization process. Using this function is an iterative process whereby the lower frequency modes are updated first, followed by higher frequency modes. Using this function only aligns the modes with each other, but does not take into account the amount of magnitude mismatch between modes or the modal damping coefficients. It also does not take into account the location of zeros. Even though this metric might cause the optimization to converge for models with larger parameter errors than the frequency response error metric (described in the next section), using this cost function could result in a non-unique parameter set. In effect, the mass and stiffness values can be traded in order to get the same modal frequency (if the mass is known to a high degree of accuracy this can be avoided). Therefore it should be noted that this matching metric is only good for early model studies, and should only be used for coarse automated tuning. This metric could be normalized in order to counteract the heavier weighting it applies to higher frequency modes, however since this is only for coarse automated tuning, it does not need to be normalized. Finer tuning is described in Section 3.3.2. The modal damping ratios can be obtained by trial and error or by using the frequency response error cost function, described in the next section.

3.3.2 Frequency Response Error

This function takes the difference of the magnitude of the transfer function at each frequency:

$$J_{phase} = \left\| W * \left(\log\left(G + \mu\right) - \log\left(\hat{G} + \mu\right) \right) \right\|_2 \tag{3.13}$$

where $\hat{G}$ is the model complex transfer function matrix, and $G + \mu$ is the corresponding measured transfer function matrix. The weighting matrix, W, allows chosen frequency regions to be emphasized, and the sensor noise parameter, μ approximates the noise floor. Note that the noise is already included in the measurement, but that it needs to be added to the model. This metric uses the logarithmic difference so that the zeros are weighted as heavily as the poles. The zeros need to be captured accurately since the updated model could eventually be used for control design. Since G is a complex vector, phase information is also included in this costing function. If the experimental phase information is corrupted, then the cost function:

$$J_{magnitude} = \left\| W * \left(\log\left|G + \mu\right| - \log\left|\hat{G} + \mu\right| \right) \right\|_2 \tag{3.14}$$

can be used, which only includes magnitude. The frequency response error metric is much more powerful than the modal frequencies metric since it takes into account not only the modal frequency, but also the residue and damping information. Furthermore, using this metric in the optimization will find a unique combination of mass and stiffness values to match the experimental data. If there exists non-parametric error in the model, however, the optimization routine will assign non-physical mass and/or stiffness values in order to minimize this cost metric. Recall that non-parametric error was addressed in Chapter 2. The main disadvantage between this type of cost function and the modal frequencies function is that the modes do not necessarily need to match in order for this optimization to be minimized. In order to guard against modal confusion, one needs to be careful that the model transfer function is sufficiently close to that of the experiment at the beginning of the optimization.

3.4 Tuning Algorithm/Optimization Problem

This optimization problem can be stated as:

$$\begin{gathered}\min_{p} J(x) \\ p_{LB} \leq p \leq p_{UB}\end{gathered} \tag{3.15}$$

which minimizes the cost function (*J*) by varying the chosen parameters (p), subject to lower and upper bounds (p_{LB} and p_{UB} respectively) on those parameters. The x vector contains all the parameters required of the function, *J*, a subset of which is p. The basic algorithm for this optimization problem starts with a nominal model. This is the model that is produced in Step 1, where the modal frequencies in the simulation model are matched to within approximately 10% of the experimentally identified modal frequencies. The cost function is first evaluated for the nominal model. Then, the chosen parameters are changed in the nominal model, and the perturbed cost function is evaluated. The parameters are changed again, and the process is repeated. The optimization routine determines how the parameters are changed at each iteration. The optimization then determines the "optimal" configuration which minimizes the cost function and produces an optimal set of parameters, p_{opt}. There are several optimization routines from which to choose (Least-squares, Simulated annealing, Genetic algorithms). The selection should be made based on the number of parameters, the continuity and linearity of the design space, the computation time associated with the evaluation of the cost function at each iteration, and whether the parameters take on discrete or continuous values. Optimization is a very mature and involved topic; two good references on optimization are Vanderplaats [52] and Papalambros [42]. MATLAB® and iSIGHT are two software packages that allow the user to easily implement the optimization routine given the cost function, the parameters, parameter bounds, gradient information and termination criteria.

For the two cost functions described in the previous sections, it is required to perform a finite element (or eigenvalue) analysis on the system before evaluating the cost function. In general, the larger and more complex the model, the longer the analysis will take. There are techniques available to shorten the computation time (modal reanalysis and modal sensitivities), but these either require storage of large matrices (M and K) or several calculations of modal sensitivities [19].

3.5 Damping

Since material damping is very difficult to model, modal damping is the practical solution. Applying modal damping is a non-physical process, and is performed after test data is available. The modal damping coefficients can be set using the frequency response error cost function. The modal frequency cost does not take modal damping into account, and must be set after the optimization, either by trial and error or using the frequency response error cost functions (Equation 3.13 and 3.14).

3.6 Application to Sample Problem

This section will demonstrate the automated tuning procedure on the same sample problem presented in Chapter 2. The discretized beam model will be used as the simulation model. Synthetic experimental data will be created as well. Both frequency domain cost models presented in this chapter will be employed, and the results will be shown.

3.6.1 Initial Model

Since it was found that the discretized beam model produced a beam tip displacement that was closer to the analytical result in Chapter 2, the discretized beam system will be used for this automated tuning demonstration (see Figure 3-2). The numbers 301-309 in this figure denoted the extra nodes that were added to the beam model. The modes of the discretized model were found in Chapter 2 and are given here in Table 3.1. By selecting the input force to occur at the end of the beam (Node 102) in the positive z direction, and the output to be the displacement of the end of the beam also in the positive z direction, a transfer function can be created. Figure 3-3

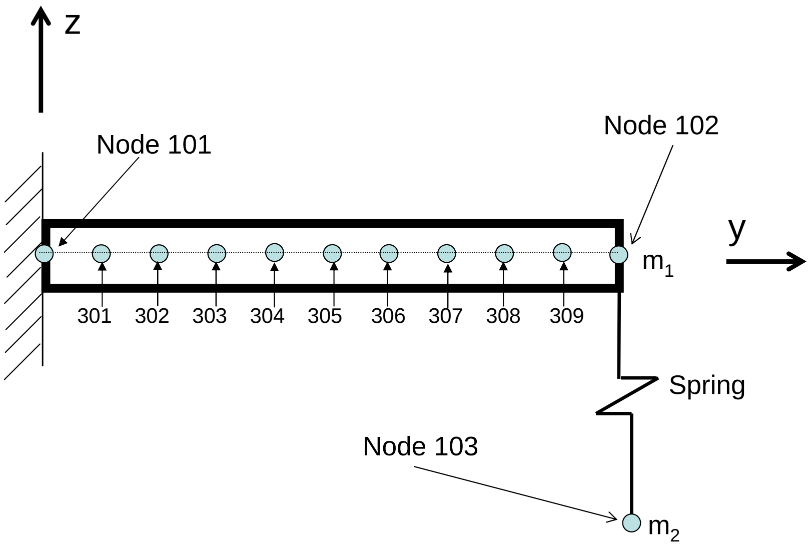


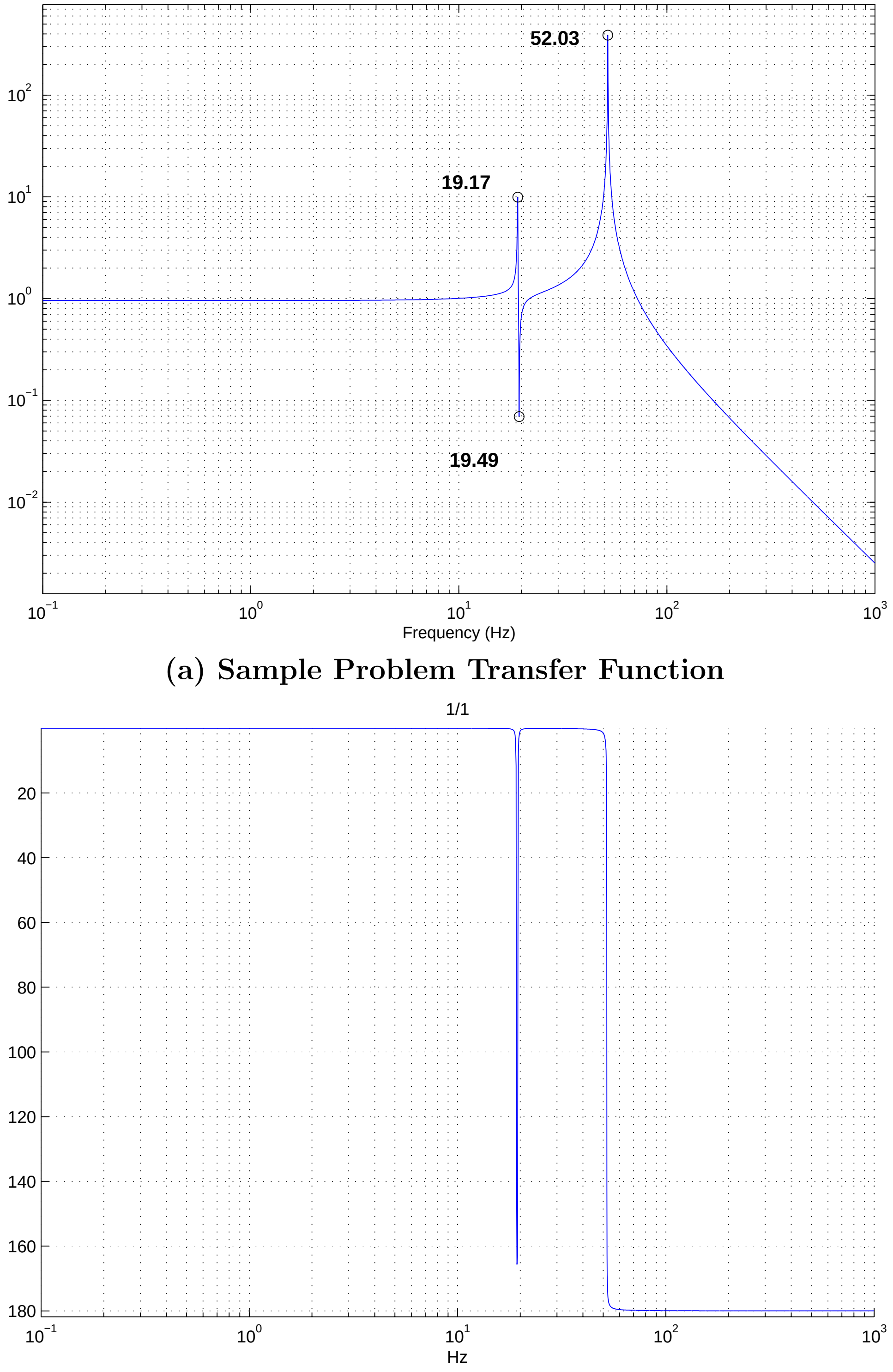
Figure 3-2: Sample Problem Discretized Beam Model

Normal Modes (Hz)
19.17
46.70
52.03

Table 3.1: Normal Modes for Discretized Model

shows this transfer function, which was created using DOCS (the code can be seen in Appendix B). The modal damping coefficient was assumed to be $\zeta = 0.001$. The numbers beside the circles in the plot indicate the frequency values for those modes in Hertz.

First of all, the high and low frequency asymptotes should be examined. The low frequency asymptote should approximate the response of the system due to static loading. The value of the transfer function at 0.1 Hz is 0.9581. From the static analysis in Chapter 2, the static value of output displacement to input force (weight of concentrated masses) is 0.9696 μm/N). The reason for the slight discrepancy here is that the transfer function of the system is created from the discretized model, and does not exactly represent a continuous model; the output displacement, however, is calculated using a closed form solution. The high frequency asymptote is falling off at a slope of -42.7 dB/dec; this was calculated by finding the slope of the line connecting



(a) Sample Problem Transfer Function

(b) Corresponding Phase (deg)

Figure 3-3: Sample Problem Transfer Function: Beam Tip Force Input (N) to End of Beam Z Displacement Output (μm)

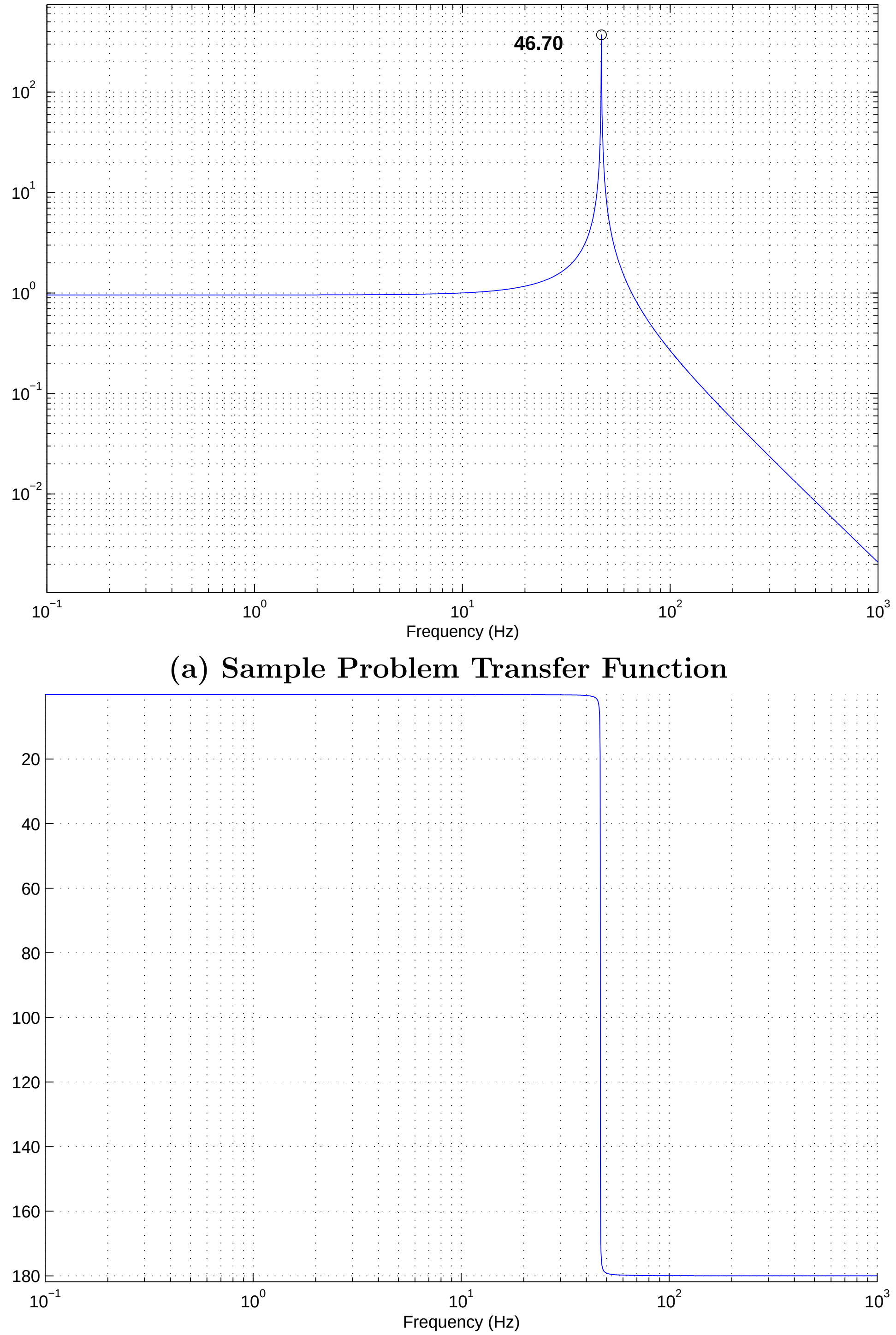
the transfer function values at 100 Hz and 1000 Hz. The expected high frequency asymptote is -40 dB/dec, since there are three poles (3 normal modes) and one zero in this system below about 1000 Hz (see [41] for asymptotic approximation rules). The slight discrepancy here is due to the fact that transfer functions do not exactly lie upon asymptotes, but only follow them. The high and low frequency asymptotes observed here, then, are reasonable.

Notice that only the first and third modes are visible on this plot (Figure 3-3). This is due to the choice of input and output degrees of freedom for this system. If instead the force input and the displacement output were applied in the positive x direction (for the same nodes), the second mode would become visible, because the second mode is associated with bending in the x-y plane (see Figure 3-4). Notice also the existence of a zero between the first and third modes (which increases the phase at that frequency). This is due to the fact that the input and output are collocated. A zero will aid in demonstrating the usefulness of the logarithmic cost function.

3.6.2 Synthetic Data

Since this is a purely analytical model, there is no true experimental data with which to compare. The automated tuning procedure can be effectively demonstrated by creating synthetic experimental data. The synthetic data will include noise, off-nominal spring stiffness, and off-nominal beam stiffness to the transfer function.

The beam and spring stiffness changes were introduced by simply changing those values in the NASTRAN model. The Young's modulus of the beam was changed by +7.14% (to 75 GPa) and the spring constant was changed by -7% (to 27.9 kN/m). The sensor noise was added to the transfer function by first choosing a reasonable noise level (NL). A value of 0.08 will be chosen for demonstration purposes. This is about 10% of the zero-frequency transfer function value and it is slightly larger than the transfer function value at 19.5 Hz (a zero), thereby approximating the noise floor. The zero shown at 19.49 Hz in Figure 3-3 dips below this value and the overall transfer function goes through this value at about 200 Hz. Setting the noise level at 0.08 will help demonstrate the effect of noise using the frequency response error



(a) Sample Problem Transfer Function

(b) Corresponding Phase (deg)

Figure 3-4: Sample Problem Transfer Function: Beam Tip Force Input (N) to End of Beam Z Displacement Output (μm)

metric. This noise level should, in real systems, approximate the noise floor. The actual noise signal was approximated using a random normal distribution with mean of zero, and a standard deviation of NL/3. This distribution ensures that 99.7% of the discretized noise signal falls between -NL and +NL (see the histogram in Figure 3-5). This distribution of noise levels at each frequency point was then added to the existing transfer function. The synthetic data and the original (discretized) model transfer function can be seen in Figure 3-6; for this sample problem these transfer functions will be taken as the experimental and model transfer functions, respectively. Notice that the first modelled modal frequency is high compared to the experimental data, and that the third modal frequency (52.03 Hz model mode) is low. Once the experimental and model data are assembled, the automated tuning optimization problem can be set up.

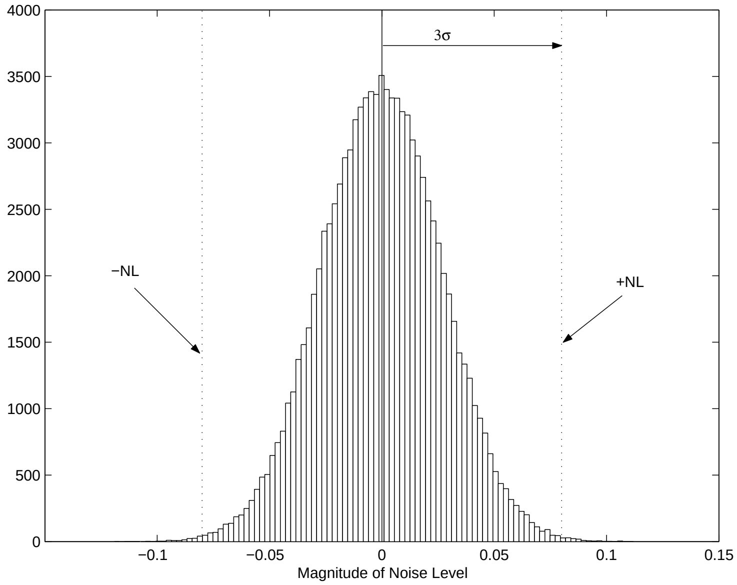


Figure 3-5: Noise Level Distribution

3.6.3 Optimization Set-Up

The experimental frequencies for the first and third modes are 18.53 Hz and 53.73 Hz, respectively. The corresponding model frequencies are 19.17 Hz and 52.03 Hz.

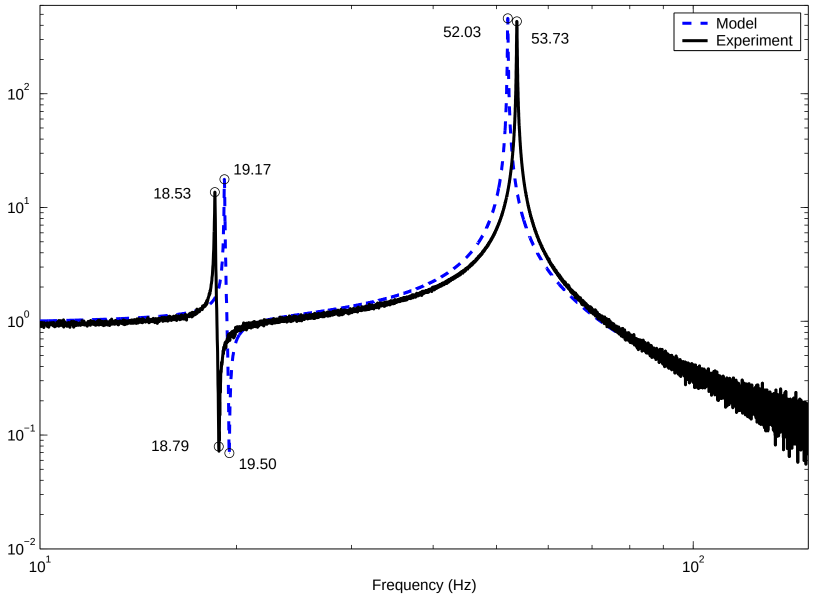


Figure 3-6: Model and "Experimental" Transfer Function for Sample Problem (μm/N)

Since fairly complete "experimental" data exists for this sample problem, and since the beam and spring stiffnesses need to be fine tuned in order to match the model transfer function to the experimental transfer function, the frequency response error metric (Equation 3.13) will be used.

The next step is to select the parameter(s) for updating. Since beam stiffness is generally an uncertain parameter, and since it influences the third mode greatly (recall that the third modeshape primarily describes beam bending in the y-z plane), the stiffness of the beam in bending (Young's modulus) will be selected for updating. The nominal value for the Young's modulus in this problem is 70.0 GPa (an aluminum beam). The beam bending parameter in each bar section (ten total) will be varied as one parameter. Even though stiffness can vary slightly across a long beam, if the stiffnesses in each beam section were allowed to vary independently, one could end up with abutting beam sections with widely varying stiffnesses. Varying the stiffnesses independently would therefore require the inclusion of boundary conditions in the optimization process, and are beyond the scope of this thesis. The second parameter that will be varied is the stiffness of the spring (recall that the first mode mainly describes spring contraction). Its nominal value is 30.0 kN/m. In this problem, another type of parameter to include in the optimization would be the modal damping. This value can be set easily using optimization, and provides an easy means of matching model and experimental magnitudes once the frequency of the modes are altered using stiffness parameters.

The SVD decomposition described in Section 3.2 will now be applied. This will identify parameters that have orthogonal influences on the eigenvalues. The eigenvalue sensitivity matrix of the first three modes with respect to the beam's Young's modulus and the spring constant is:

$$\mathrm{R} = \begin{bmatrix} \frac{\partial\omega_1}{\partial \mathrm{E}} & \frac{\partial\omega_1}{\partial \mathrm{k}} \\ \frac{\partial\omega_2}{\partial \mathrm{E}} & \frac{\partial\omega_2}{\partial \mathrm{k}} \\ \frac{\partial\omega_3}{\partial \mathrm{E}} & \frac{\partial\omega_3}{\partial \mathrm{k}} \end{bmatrix} = \begin{bmatrix} \text{- 2.25} & \text{- 57.98} \\ \text{- 146.36} & 0 \\ -157.00 & -6.12 \end{bmatrix} \tag{3.16}$$

Notice that the modal damping is not included here as a parameter since modal damping does not effect the eigenvalues (but might affect the eigenvectors). The singular values are

$$\sigma_1^2 = 214.71 \; \sigma_2^2 = 58.0658. \tag{3.17}$$

These singular values indicate that both parameters are influential, but not equally so. The singular values are not very close to one another numerically, but since they are both sufficiently large, both parameters will be retained in the optimization problem. The U matrix is

$$U = \begin{bmatrix} \text{- 0.0174} & 0.9973 & \text{- 0.0718} \\ \text{- 0.6814} & \text{- 0.0644} & \text{- 0.7290} \\ \text{- 0.7317} & 0.0363 & 0.6807 \end{bmatrix}. \tag{3.18}$$

It can be seen that the maximum row value for each row greatly exceeds the other row values for the first row only. Therefore it is possible that these parameters can be combined in different ways to achieve the same result in the cost function. This is something to keep in mind throughout the updating process.

The upper and lower parameter bounds should be decided upon next. The Young's modulus of the aluminum beam will be varied by [-3%, +10%] (63 GPa to 77 GPa). A listing of common engineering materials can be found in [4]. The highest value listed for aluminum is nominally 75 GPa (2014 series) which represents about a +7% change in the stiffness; the lowest value is 70 GPa (6061 series), identical to the nominal value in the sample problem. An extra 3% padding was then added to the upper and lower bounds as a buffer. The upper and lower bounds of the spring will be + 10% of the nominal value, following the general rule of thumb. The absolute bounds for the modal damping are zero to one (only available values for modal damping). But for demonstration purposes, the bounds for the modal damping parameters on all modes will be + 10% of the nominal value (0.001).

The costing function is the next thing to decide. Since there is full transfer function information, the frequency response error function is chosen. Furthermore the phase version of the frequency response error (Equation 3.13) will be chosen since the phase of this experimental data is only slightly degraded by the noise addition. The noise level in the optimization, μ, was set to 0.08, as discussed before. Therefore, the optimization statement for this problem is as follows:

$$\begin{aligned} &\min_{p} J(x) \quad \text{subject to } p_{LB} \le p \le p_{UB} \\ &\text{where } J = \left\| W * \left(\log\left(G + \mu\right) - \log\left(\hat{G} + \mu\right) \right) \right\|_2 \end{aligned} \tag{3.19}$$

The weighting matrix, W, was also used in order to focus on the frequency region between 10 and 70 Hz. The DOCS [6] updating function *update_frf* is used, which makes use of MATLAB®'s constrained non-linear least squares optimization algorithm and implements the above optimization. The DOCS run code can be seen in Appendix B.

3.6.4 Results

The automated tuning computation took only seven minutes and 17 iterations using *update_frf* on a Pentium IV, 1.7GHz PC. The initial value of the objective function was 3144.08 and the final value after tuning is 0.0209. The fitted stiffness values are extremely close to the“experimental” values (see Table 3.2). Notice also that the modal damping coefficients were not originally changed in order to produce the “experimental” data, and they were also left unchanged in the automated updating. Changing the stiffness of a component while keeping the modal damping coefficient constant can, in general, change the magnitude of the response, so for real systems, this parameter is a good candidate for automated updating. The results can be seen graphically in Figure 3-7. The experimental and updated model transfer functions match so closely between 10 and 100 Hz that it is hard to distinguish them in the figure. A listing of the original, experimental and fitted parameters can be seen in Table 3.2.

	Nominal Model	“Experiment”	Updated Model
Beam Young’s modulus(GPa)	70.000	75.000	75.012
Spring Constant (kN/m)	30.000	27.900	27.898
Modal Damping	0.001	0.001	0.001

Table 3.2: Sample Problem Parameter Values for Automated Tuning

Another case can be run where the experimental modal damping values are set to be low on the first mode and high on the third mode, while keeping the initial model modal damping values the same. The optimization is run again and the results can be seen in Figure 3-8 and Table 3.3. Notice that the model updates well for the third mode, but hardly moves the position of the first mode. This is due to the addition of the significantly altered damping parameters. Therefore, when performing an automated optimization, it is usually more effective to first update the physical parameters (such as Young’s modulus and spring constant) and then go back and set the modal damping parameters. This is proper since modal frequency is more important than modal damping and since modal damping does not affect the modal frequency values. In this way the modeler can be sure that the modal damping parameters are not driving the tuning algorithm optimization.

This section has demonstrated the automated tuning procedure on a single input/single output (SISO) system. For multiple input/multiple output (MIMO) sys-

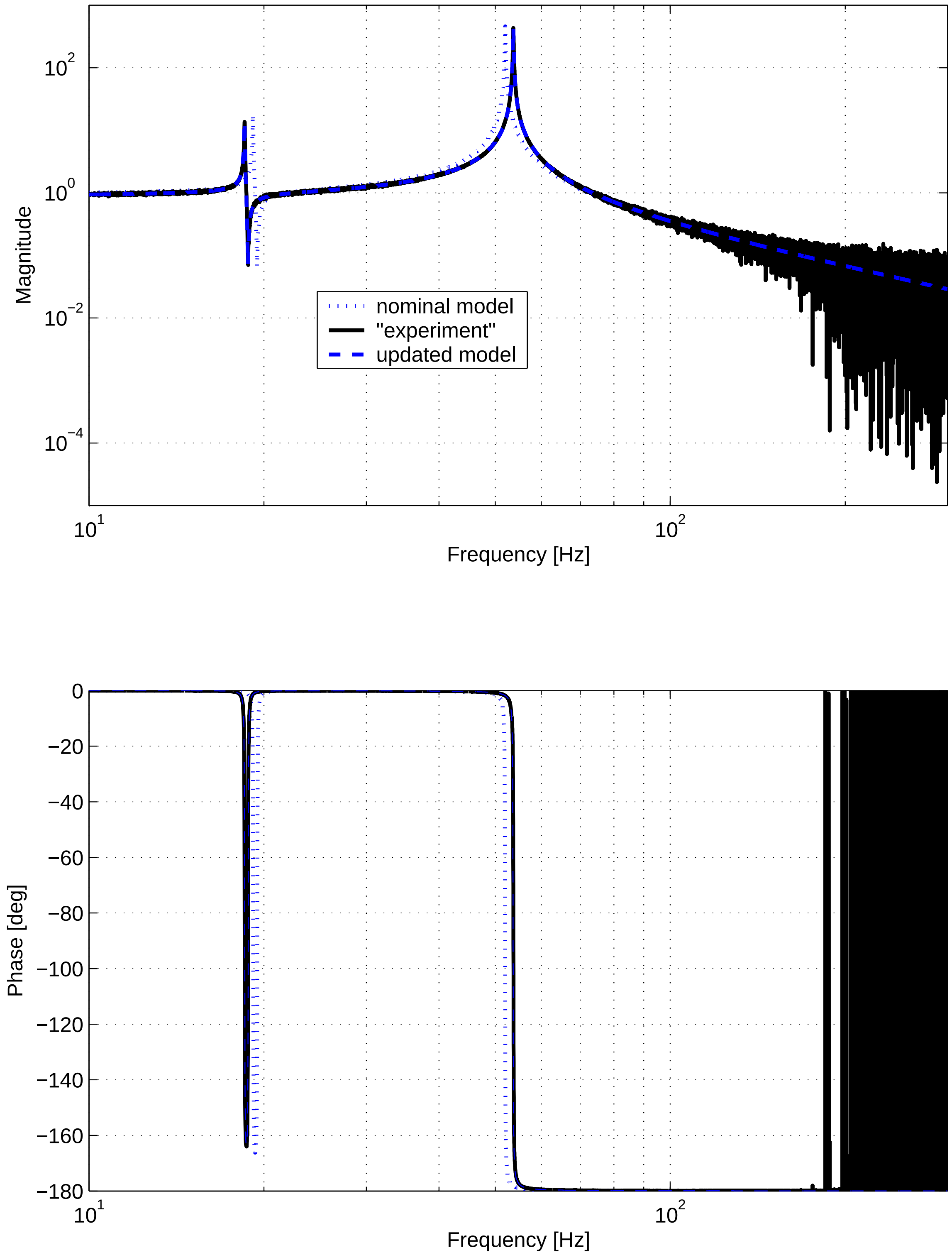


Figure 3-7: Automated Tuning Results Transfer Functions (μm/N)

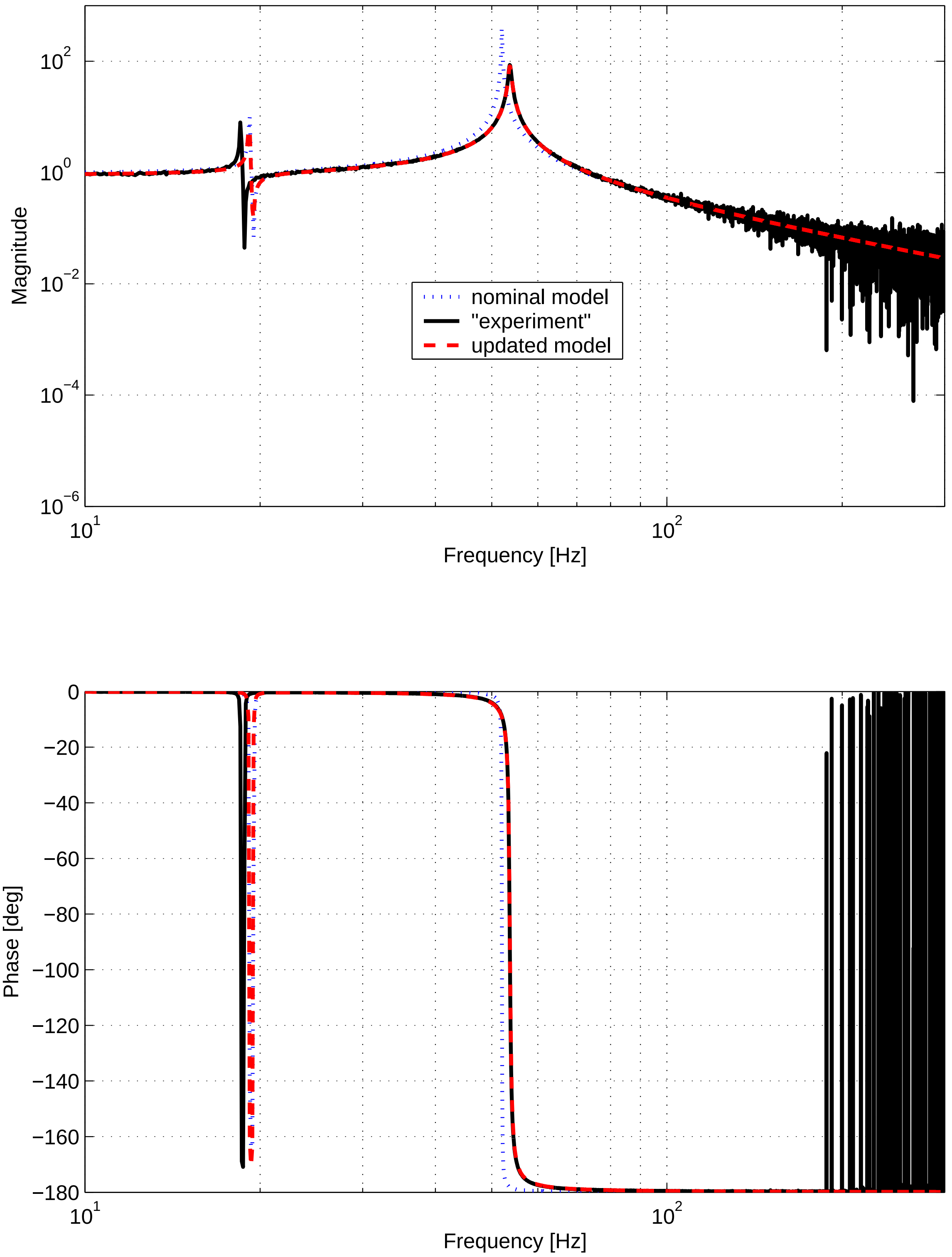


Figure 3-8: Automated Tuning Results Transfer Functions (μm/N) with Modal Damping

	Nominal Model	“Experiment”	Updated Model
Beam Young’s modulus(GPa)	70.000	75.000	74.8790
Spring Constant (kN/m)	30.000	27.900	29.8740
Modal Damping Mode 1	0.001	0.0005	0.0006
Modal Damping Mode 3	0.001	0.005	0.0054

Table 3.3: Sample Problem Parameter Values for Automated Tuning with Modal Damping

tems the procedure would be the same except the transfer function matrix used in the objective function would be a two-dimensional array instead of a one-dimensional array.

3.7 Chapter Summary

After the initial model for some existing system is created, experimental data might then be taken. This chapter outlines two main metrics which can be used to compare the experimental and model data in the frequency domain. The first metric compares only the modal frequencies. The second metric employs the entire transfer function in order to measure the model/experiment mismatch. The process of automated tuning is to alter chosen parameters (usually stiffness parameters) so that the mismatch metric is minimized subject to bounds on the parameters. The same sample problem presented in Chapter 2 was extended here to demonstrate the application of one of the mismatch metrics, and the implementation of the optimization. In the absence of experimental data for the sample problem, synthetic experimental data was created. The optimization was shown to be successful, with the fitted parameter values and transfer functions very close to the “experimental” values.

Chapter 4

Step 3: Model Validation and Tracking

4.1 Purpose

A chief purpose of developing a simulation model is to be able to evaluate the performance and cost associated with configuration changes without the expense of building a new hardware model. Configuration changes are the types of changes that are essential to optimization and redesign analyses, such as changing the placement or the weight of the component. These are generally not the types of parameter changes that are made in Step 2, nor the types of changes that were important in Step 1 (boundary conditions, etc). Configuration changes imply component redistribution (changes of form) rather than changes in functionality. Once a simulation model is built, it is far less expensive to evaluate the perturbed system with the simulation model than to reassemble hardware. It is usually using a simulation model that optimization of performance and cost is performed. However, there must be some degree of confidence that this model is valid across these configurations changes in order for the optimization to have validity. One of the dangers of automated tuning is that if there is more than one parameter to be tuned there is no guarantee that the parameters are tuned in a way that produces a physical model. For example, increasing the Young's modulus of the beam in the sample problem might produce the same effect on the frequency response as changing the beam's cross-sectional area, however changes in one or both of these parameters might be required in order to produce a system model that reflects the actual physical system parameters. A model tuned in a non-physical way might match experimental data well for the current configuration, but it is not guaranteed to match for configuration changes, whereas a model tuned properly will track across configurations. Howell [26] presents a model that represents the testbed well for the original model, but that provides poor performance predictions when the configuration is changed (disturbance magnitude is increased). It is essential to validate that the parameters were tuned in a physical way in Step 2. This can be verified by perturbing the testbed and the simulation model together to see if they track along configuration changes. If the model does track, then there is greater confidence that the model physically represents the system and is appropriate for optimization.

Recall from Chapter 2 that the first modeshape of the sample problem mainly involved spring elongation and contraction, and that it was approximated by a simple mass-spring system (Figure 2-10(a)). Figure 4-1 and Table 4.1 show two new evaluations of the sample problem (using the discretized system). One evaluation increases the spring constant (K), and another decreases the concentrated mass at Node 103 (m). Both produce the same transfer function and the same modal frequencies (see Figure 4-1), but only one combination of K and m is physically correct. This illustrates the concept on non-physical tuning.

	Nominal Model	Changed K	Changed M
Spring Constant (kN/m)	30.0	36.3	30.0
Node 103 Concentrated Mass (kg)	2.0	2.0	1.653
Mode 1 (Hz)	19.17	21.01	21.08
Mode 3 (Hz)	52.03	52.24	52.06

Table 4.1: Sample Problem Parameter Values for Automated Tuning

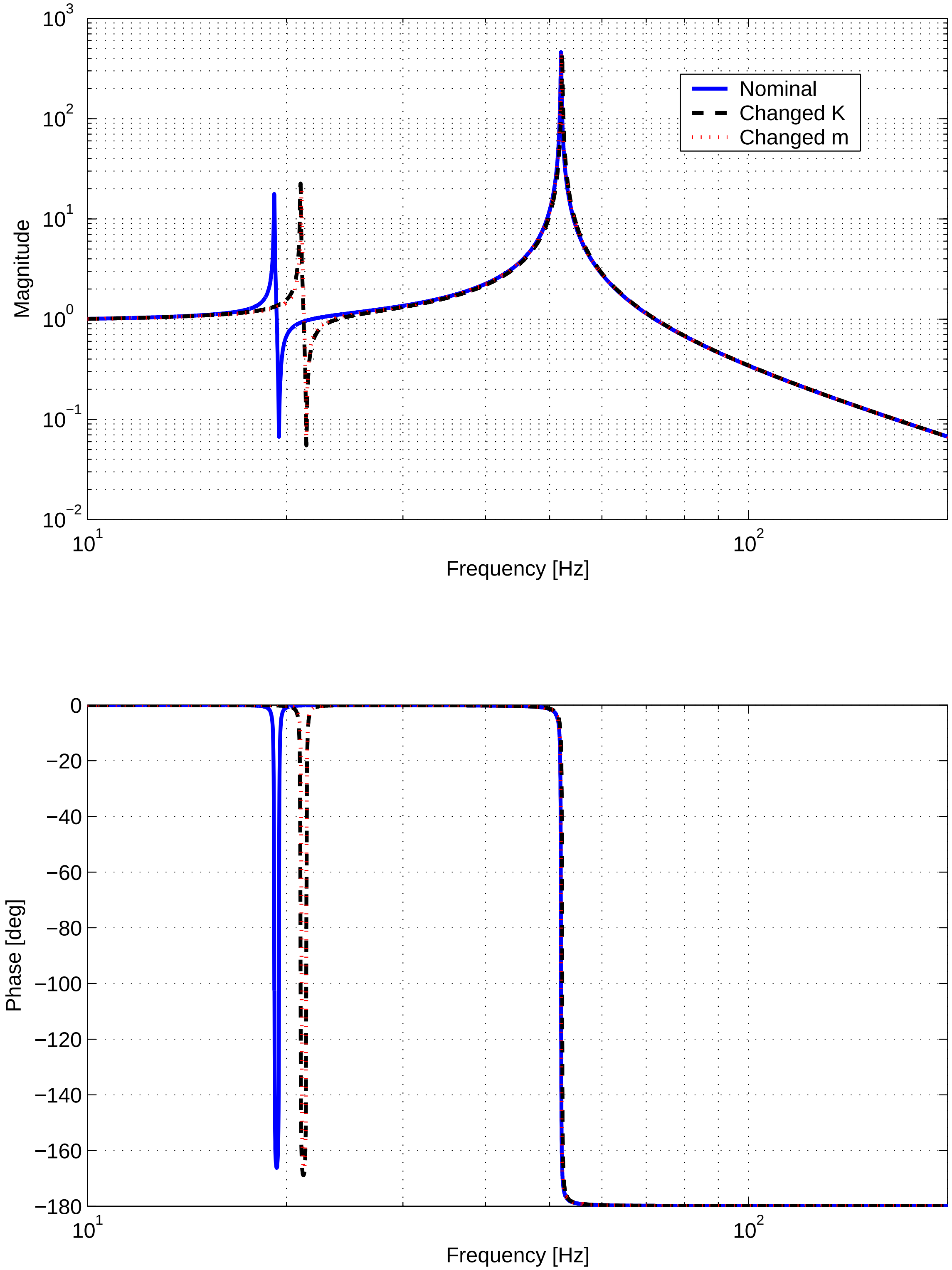


Figure 4-1: Transfer Functions of Nominal and Altered Sample Problem (μm/N)

4.2 Method

The model that is used at this stage is a mature model that has undergone the coarse and automated tuning steps. A configuration change is implemented in the hardware model, and a corresponding change is made in the updated simulation model. One or several new configuration systems can be constructed this way. The new hardware and simulation models are then compared in the same ways as in Sections 3.3.1 and 3.3.2. The new configurations are evaluated using the optimal parameter values p_{opt} found in Step 2 for the original configuration (the new configurations are not tuned). For each configuration, then, there will be an associated objective function cost value. These cost values are evaluated as described in Section 4.3. If this evaluation is found to be deficient, it can be assumed that the parameters were not changed properly in Step 2 for the original configuration. Consequently, Step 2 needs to be repeated with an altered formulation (see Figure 1-2). One way to alter the algorithm presented in Step 2 is to narrow the bounds on the chosen parameters. Another way is to change the number of parameters by including some that were initially discarded. Ultimately the formulation in Step 2 is dependent upon the modeler's understanding of the system. It is assumed that Step 1 was implemented properly, since Step 1 consisted of applying engineering insight and systematic error checking. However, further non-parametric error checking might be required in order to produce a physically appropriate system.

4.3 Multi-Configuration Metrics

The matching metrics provided in Chapter 3 are reproduced here for convenience.

$$\begin{aligned}
J_{\text{modal}} &= \sum_{i=1}^{nm} \left[\left(\omega_{\text{model}}\left(i\right) - \omega_{\text{experimental}}\left(i\right) \right) \right]^2 \\
J_{\text{phase}} &= \left\| W * \left(\log\left(G+\mu\right) - \log\left(\hat{G}+\mu\right) \right) \right\|_2 \\
J_{\text{magnitude}} &= \left\| W * \left(\log\left|G+\mu\right| - \log\left|\hat{G}+\mu\right| \right) \right\|_2
\end{aligned} \tag{4.1}$$

As previously mentioned, these metrics will be applied to the new configuration models. The new metrics, Z, will then be defined as

$$\begin{aligned}
\mathrm{Z}_{\mathrm{modal}} &= \max_{k} \left[(J_{\mathrm{modal}})_1 \,, (J_{\mathrm{modal}})_2 \,, (J_{\mathrm{modal}})_3 \,, ..., (J_{\mathrm{modal}})_k \right] \\
\mathrm{Z}_{\mathrm{phase}} &= \max_{k} \left[(J_{\mathrm{phase}})_1 \,, (J_{\mathrm{phase}})_2 \,, (J_{\mathrm{phase}})_3 \,, ..., (J_{\mathrm{phase}})_k \right] \\
\mathrm{Z}_{\mathrm{magnitude}} &= \max_{k} \left[(J_{\mathrm{magnitude}})_1 \,, (J_{\mathrm{magnitude}})_2 \,, (J_{\mathrm{magnitude}})_3 \,, ..., (J_{\mathrm{magnitude}})_k \right]
\end{aligned} \tag{4.2}$$

where k denotes the number of new configurations constructed. These metrics are simply the maximum values of the new configuration costs. They will then be compared against the costs of the original configuration. Success of the algorithm will be defined by one of the following conditions (depending on which cost was ultimately used in Step 2),

$$\begin{aligned}
\mathrm{Z}_{\mathrm{modal}} &\leq (J_{\mathrm{modal}})_t + \varepsilon \\
\mathrm{Z}_{\mathrm{phase}} &\leq (J_{\mathrm{phase}})_t + \varepsilon \\
\mathrm{Z}_{\mathrm{magnitude}} &\leq (J_{\mathrm{magnitude}})_t + \varepsilon
\end{aligned} \tag{4.3}$$

The t subscript indicates the costs associated with the original tuned model. The ε term allows the new configuration costs to be slightly higher than the original configuration costs. This term must be set by the modeler. These conditions merely determine if the costs associated with the new configurations are no worse than the cost of the original plus some small slack. If this is the case then it can be assumed that the tuned model physically represents the actual system.

An alternate way of assessing the validity of the model (and also by allowing the new costs to exceed the original costs by a small amount) is to first construct the following matrix, Y, which is symmetric positive definite.

$$\underset{\mathrm{k+1 \times k+1}}{\mathrm{Y}} = \frac{1}{\mathrm{J_t^2}} \begin{bmatrix} \mathrm{J_t^2} & \mathrm{J_t J_1} & \mathrm{J_t J_2} & \cdots & \mathrm{J_t J_k} \\ & \mathrm{J_1^2} & \mathrm{J_1 J_2} & \cdots & \mathrm{J_1 J_k} \\ & & \mathrm{J_2^2} & \cdots & \mathrm{J_2 J_k} \\ & \text{Sym.} & & \ddots & \vdots \\ & & & & \mathrm{J_k^2} \end{bmatrix} \tag{4.4}$$

The infinity norm of this matrix (maximum absolute row sum) is then evaluated and divided by the number of configurations (original plus new). This value, called the multi-configuration metric, contains information about the distribution of the costs as well as the distance between the new configuration costs and the original cost. It is then compared against a threshold number, T (see Equation 4.5). If it exceeds the threshold number, then the model is rejected; if the multi-configuration metric is below the threshold number, the model is accepted.

$$\frac{\|\mathrm{Y}\|_\infty}{(\mathrm{k}+1)} \leq \mathrm{T} \tag{4.5}$$

The threshold number is defined by the modeler and can be greater than or equal to 1.0. A threshold number of 1.0 approximately corresponds to the success criteria defined in Equation 4.3, with a small allowable distribution around the original cost. Increasing the threshold number increases the range in which new configuration costs are allowed to exceed the original cost. It is not recommended that the threshold number exceed 2.0 since this would indicate configuration costs that far exceed the original cost.

4.4 Limits and Perturbation Size

For every model there is a limit as to the size of the perturbation that can be made to a model, and still retain “good” results. For example, non-linear buckling modes will eventually become apparent in beams and in plates for increasingly large loads. This is a fundamentally new type of behavior that cannot be modelled well using a strictly linear model. Also, modal overlap can cause large changes in the response for very small changes in parameters.

4.5 Sample Problem Application

In the previously described sample problem, a configuration change could be (1) attaching the spring to a different point on the beam, (2) changing the length of

	Tuned Model	Tuned with New Configuration
Mode 1	18.54	52.47
Mode 2	48.38	52.47
Mode 3	53.73	46.00
Mode 4	-	46.00

Table 4.2: Changes in Modal Frequencies with New Configuration

beam or spring, or (3) changing the values of the concentrated masses. Changing the mass and the lengths are easy changes to implement in the model. However, changing the placement of the spring is more difficult; the beam must first be discretized in order to provide a new spring attachment node. Therefore the discretized system used in Chapter 3 is more appropriate to use in this situation. In addition, a new data acquisition setup is required on the testbed (since the output must now be measured in a different place). This illustrates the point that configuration changes are different for every system, so the user must be diligent to note the possible effects of these changes.

Recall that in Chapter 3 a nominal model and “experimental” data were presented for the original configuration, and a tuned model was created. The resulting transfer functions can be seen in Figure 3-7, and a list of changed parameters can be seen in Table 3.2 (the “true” parameters are listed under “Experiment”). Let us suppose that the configuration change was to attach the spring at the center of the beam (Node 305 in Figure 3-2) instead of at the end. The tuned finite element model was used to model this new configuration,. The resulting transfer function for the same input and output locations can be seen in Figure 4-2. Table 4.5 shows the changes in the modal frequencies. Note the striking dissimilarities between the previous transfer function and the one for the new configuration. The objective function cost after the automated tuning in Chapter 3 was $\mathrm{J_t} = 0.0012$. Applying the same process as in Chapter 3 for producing the “experimental” data, the new value of the objective function for the new configuration is $\mathrm{J_1} = 0.0010$, which is less than the cost in Chapter 3. This satisfies the requirements as laid out in Equation 4.3 for positive

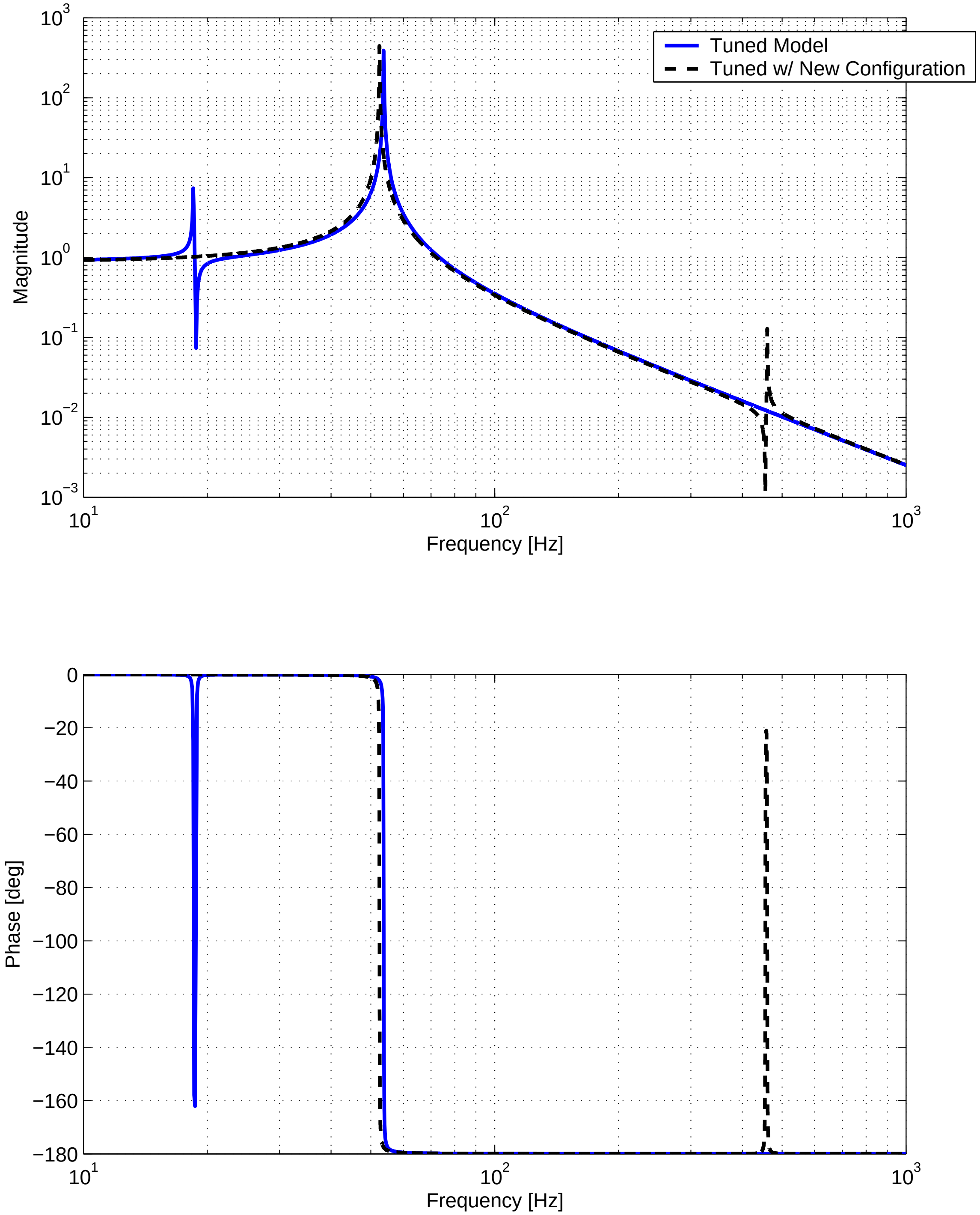


Figure 4-2: Sample Problem Transfer Function: Beam Tip Force Input (N) to End of Beam Z Displacement Output (μm)

	Nominal Model	Non-Physical Model
Mass at Node 102 (kg)	10.0	9.4
Mass at Node 103 (kg)	2.0	2.15
Modal Damping for Mode 3	0.001	0.0025

Table 4.3: Changed Parameters for Non-Physical Model

values of ε. The Y matrix would be

$$\underset{2\text{x}2}{\mathrm{Y}} = \frac{1}{1.4837\text{e - }6}\begin{bmatrix} 1.4837\text{e - }6 & 1.235\text{e - }6 \\ 1.235\text{e - }6 & 1.0279\text{e - }6 \end{bmatrix}. \tag{4.6}$$

And the multi-configuration metric is

$$\frac{\|\mathrm{Y}\|_\infty}{(\mathrm{k}+1)} = \frac{1.8323}{2} = 0.9162. \tag{4.7}$$

This value (0.9162) is less than unity; this was expected since the new configuration cost was less than the original cost. This value indicates, but does not prove, that the model was tuned in a physical manner.

Now let us examine a model of the sample problem which was tuned in a non-physical manner. The masses at Nodes 102 and 103 and the damping of mode 1 were changed in order to produce the non-physical model in the original configuration. We know that this is a non-physical model since the concentrated masses can usually be measured to within 0.05 kg using a standard laboratory scale. Figure 4-3 shows the transfer functions of the nominal model, the “experimental” data, and the non-physical model. It can be seen that the non-physical transfer function matches well with the experimental data for the nominal configuration. However, the value of the objective function for the non-physical model when compared to the ”experimental” data is 0.0292, which is almost 30 times larger than the cost associated with the originally tuned model (0.0012). Table 4.5 contains the changed parameter values. Now, when the configuration is changed so that the spring is attached to the center of the beam (Node 305) using the non-physical model, the new objective function cost is 0.410, which is, as expected, larger than the cost using the tuned model for the same

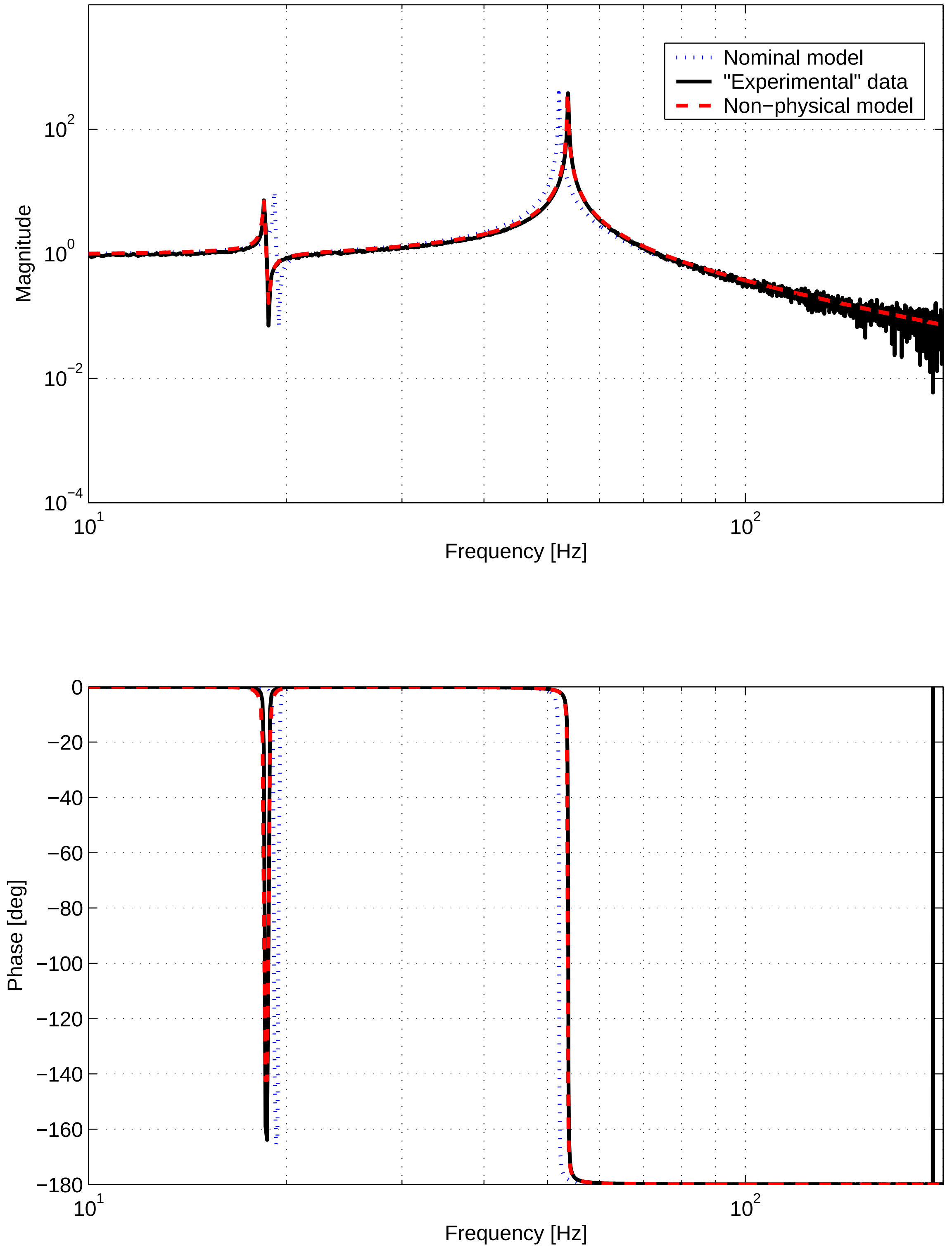


Figure 4-3: Sample Problem Transfer Functions: Beam Tip Force Input (N) to End of Beam Z Displacement Output (μm)

configuration. Applying the Y metric to this scenario ($\mathrm{J_t} = 0.0292$, $\mathrm{J_1} = 0.0410$) gives

$$\begin{aligned} \underset{2x2}{\mathrm{Y}} &= \frac{1}{8.499\mathrm{e}\text{ - }4}\begin{bmatrix} 8.499\mathrm{e}\text{ - }4 & 0.0012 \\ 0.0012 & 0.0017 \end{bmatrix} \\ \frac{\|\mathrm{Y}\|_\infty}{(\mathrm{k}+1)} &= \frac{3.3838}{2} = 1.6919 \end{aligned} \tag{4.8}$$

The value of the multi-configuration metric is rather large for this test, and may indicate that this model was tuned in a non-physical way. Therefore an iteration back to Step 1 or Step 2 is prescribed in order to create a model that is physically representative. Some alterations that can be made when iterating back to Step 2 could be the changing the choice of the objective cost function, changing the choice of updateable parameters, and changing the allowable parameter bounds.

4.6 Chapter Summary

The motivation for evaluating the system was presented: that if the tuned model tracks well over different configurations, then there is a greater likelihood that this simulation model is tuned physically and is therefore more useful for tradespace exploration or optimization studies using a simulation model. It is therefore recommended to perturb the system's configuration one or more times in order to evaluate the validity of the model. Also presented in this chapter is a new type of metric for multi-configuration tuning which aids the modeler in evaluating the model across configurations.

Chapter 5

Experimental Validation

In the previous chapters, the three step process was illustrated using a simple beam model. Flight systems, however, are much more complicated. For that reason, the testbed OPTSIM (Optical Performance Testbed for Space Interferometry Models) was developed. This system is complex enough to illustrate the problems encountered on space flight systems, yet it is simple enough to model and to understand the basic dynamics. OPTSIM was built at the Space Systems Laboratory at MIT. The central structure (bus area, disturbance source, and one truss bay) was built for experimental validation in de Weck [13]. The truss boom and optical mounting plates were subsequently added in order to include optics and controls.

The goal of the experiment is to illustrate that the three step procedure described in this thesis can indeed be applied successfully to an actual system, and to support the hypothesis of this thesis. The hypothesis of this thesis is that this three step procedure will produce a model that effectively tracks along configuration changes. The final tuned model will be compared with a model that is tuned by a third party using trial and error parameter changing, a traditional way of tuning a model (see Section 5.3). By performing experimental validation, the hypothesis that the proposed three step procedure is useful in discerning a physically appropriate simulation model will be supported.

OPTSIM was designed to structurally represent a spacecraft interferometer. It has a central area where a large fraction of the mass could be located (bus), a disturbance

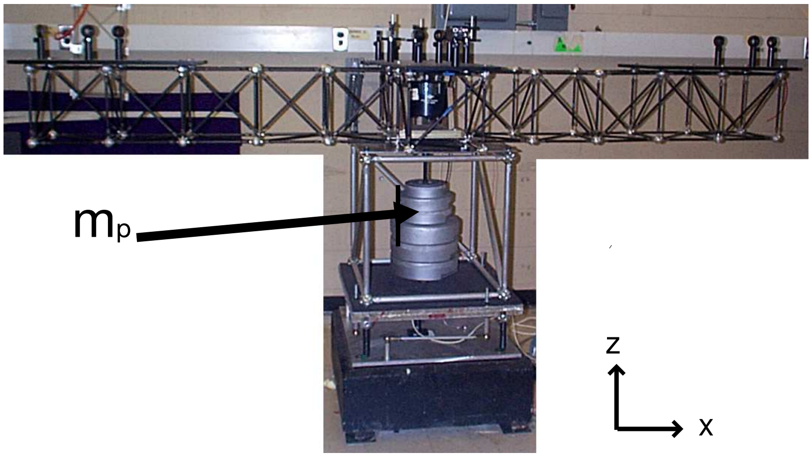


Figure 5-1: OPTSIM: General View

source located above the bus, and a long truss consisting of 11 truss bays. Stationary optical elements are mounted on the top of the truss, and some of those elements can be easily replaced with fast steering mirrors (FSM's). This structure has the potential for a multi-disciplinary analysis, but only the structural model will be address in this work.

5.1 Testbed Description

5.1.1 Hardware

The entire testbed sits on a large rectangular cement block (300 lb). At each of the four corners, identical springs are mounted vertically. The springs are all of the same length (4.5"), and have a shortened bolt running through the center to keep them upright. On top of the springs sits a three layer composite aluminum plate (610 mm x 610 mm x 36.35 mm). The outer two layers are made of solid aluminum. The central layer is made of aluminum honeycomb. Rotational inhibitors are connected to the bottom of the plate and to the cement block. These mechanisms prevent the plate from twisting about its normal axis, ensuring near-vertical motion on the z-axis. These elements comprise the base of OPTSIM (see Figure 5-2). One large cubic truss bay (0.5 m on a side) is bolted on top of the composite plate. The beams (22.12 mm

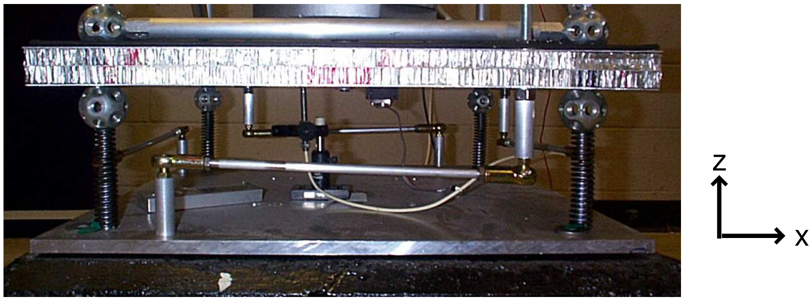


Figure 5-2: Base of OPTSIM

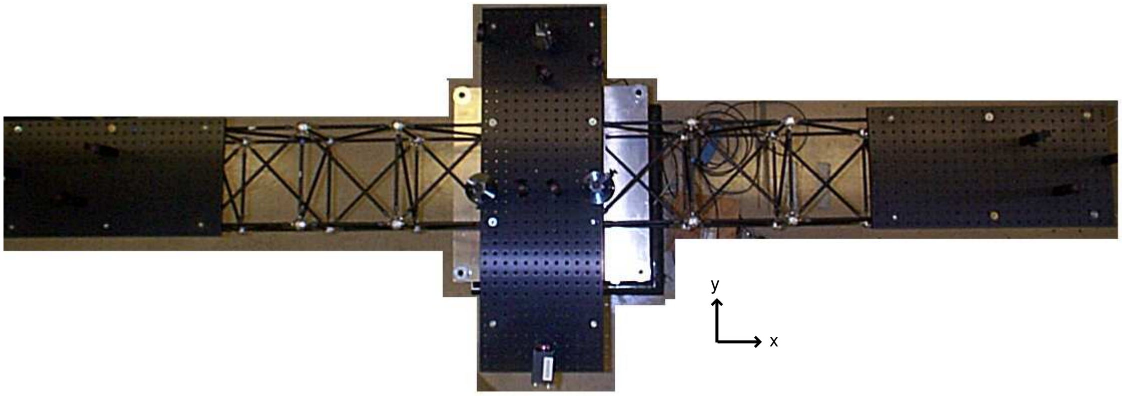


Figure 5-3: Optical Truss: Top View

o.d.) are made of aluminum tubes that are bolted into eight large aluminum nodes (49.48 mm o.d.). Three lateral sides of the bay also contain diagonal beam elements. A thin aluminum plate (540 mm x 540 mm x 4.572 mm) is bolted to the top of the bay. A steel rod (5mm o.d.) connects the center of the composite plate to the center of the aluminum plate. The steel rod allows the easy addition of additional bus weight (m_p) using standard exercise weights (see Figure 5-1). The additional weight of the bus can range from zero to 200 lbs. in increments of 10 lbs. The optical truss is bolted on top of the aluminum plate. The overall layout is in a cross pattern with 11 bays in one direction, and two bays on either side of the central bay in the perpendicular direction (see Figure 5-3). The beams of the truss are made of tubular aluminum (9.398 mm o.d.). They are fastened to small aluminum nodes (31.75 mm o.d.) using a nut cap and a socket cap with two washers. Each truss bay is a cube (0.25 m on

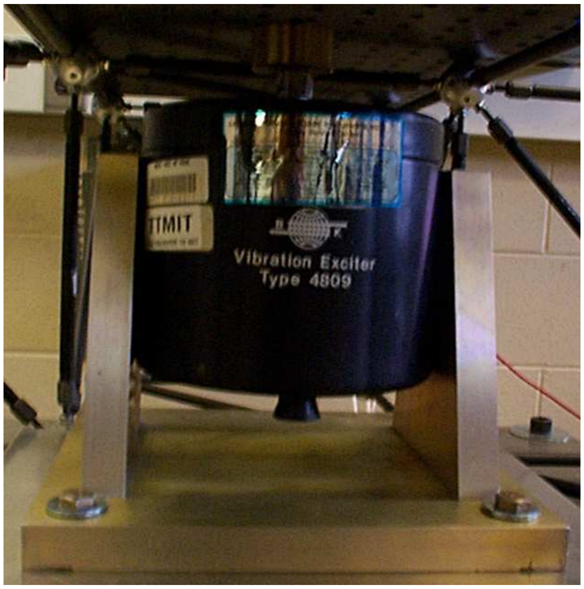


Figure 5-4: Disturbance Shaker

a side). The truss was built by bolting the nodes to plate templates, then attaching the beams between the nodes to ensure uniform side length. There are three optical aluminum plates that are bolted to the top of the truss. Two plates (2' x 1' x 0.5") lie on each outer section of the truss. The third plate (3' x 2' x 0.5") is mounted on the three central bays (perpendicular to the rest of the truss). Optical elements are mounted on top of these plates. The optical elements comprise of four beamsplitters, five mirrors, and a CCD camera. They are set up so that interferometric fringes can be captured. Controls can be added to this testbed by replacing some or all of the stationary mirrors with FSM's.

5.1.2 Sensors and Actuator

Currently this is a SISO (single input single output) system, but it has the potential for adding more input and output locations such as imaging sensors and control actuators. It would be a much more complicated system if there were controlled actuators, since factors such as time delay would need to be taken into account. The input to the system is a disturbance force provided by a shaker (Brüel & Kjaer Vibration Exciter Type 4809, see Figure 5-4) which is located inside the small central bay. It can provide up to 44.5 N of force in the vertical direction, with a frequency range of 10 Hz to 20 kHz. In addition, there is a load cell attached to the shaker that records the force input to the system (Model 208A02, labelled PCB in Figure 5-5). The force provided by the shaker depends on the input voltage and the proof mass (small mass on top of the shaker that is attached to the load cell. The proof mass in this case is kept constant at 339.75 grams. The load cell conditioner provides a voltage, and the calibration factor was experimentally determined to be 94.872 N/V. The output of the system is the vertical (z-direction) displacement of the composite base plate as measured by an eddy current gap sensor (Bentley XL 5mm). The calibration factor is 0.348 V/mil, which was calibrated using a LB-11/70 Laser Displacement Sensor (see Figure 2-4).

5.1.3 Data Acquisition

This system was assumed to be steady state, that is the loading force changes over time (*F(t)*) but its statistics (density function and mean) remain constant. The data acquisition system for the gap sensor signal is a Tektronix Tekbox 4690. The Tekbox has coaxial input and output connections, and can provide various high and low frequency electrical signals (sine wave, chirp, sawtooth). The connection setup for the Tekbox data acquisition can be seen in Figure 5-5. The output channel of the Tekbox produces the input signal to the shaker. The input signal to the shaker (V_S) is a banded white noise (0Hz-100Hz) spectrum with a variable root mean squared voltage value of $0.1V_{RMS}$ to $1.0V_{RMS}$ (to be set by the user). These input settings translate into the input noise spectrum shown in Figure 5-6 for a voltage RMS value of $0.2V_{RMS}$. The output signals of the gap sensor and the load cell on the shaker were set by the Tekbox to record data up to 200 Hz. The Tekbox automatically adjusts the sampling rate in order to acquire anti-aliased data in a set frequency range, in this case that range is 0-200 Hz. There were 2046 frames taken and these frames were taken 40 times sequentially and were averaged. In addition to time traces of both the gap sensor and load sensor, transfer and coherence functions were also taken. The transfer functions recorded the load cell signal as the input and the gap sensor signal

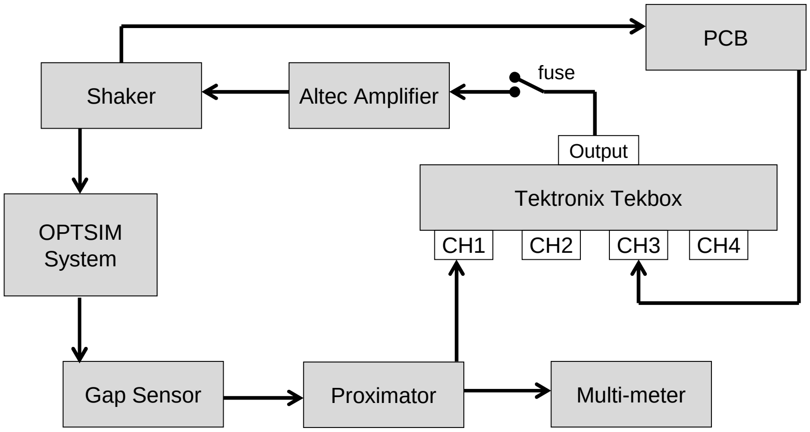


Figure 5-5: Tekbox Data Acquisition Setup

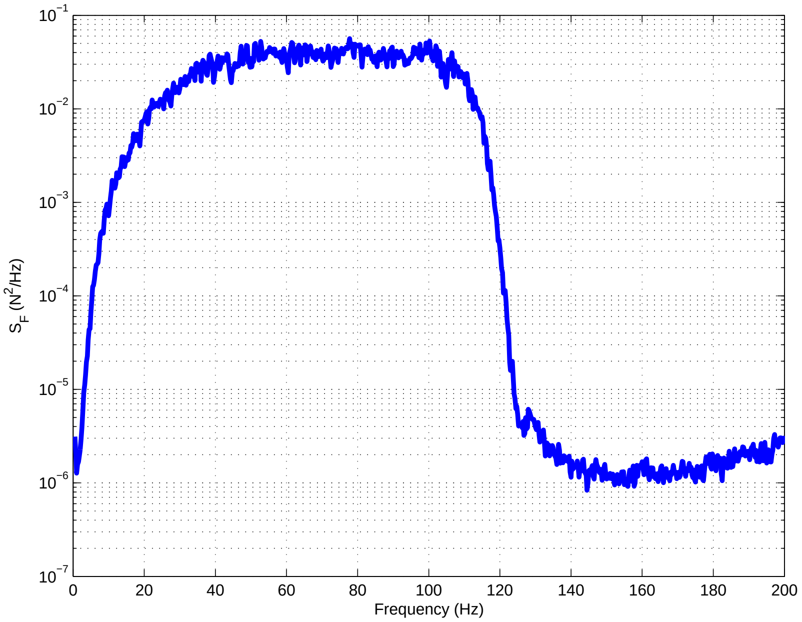


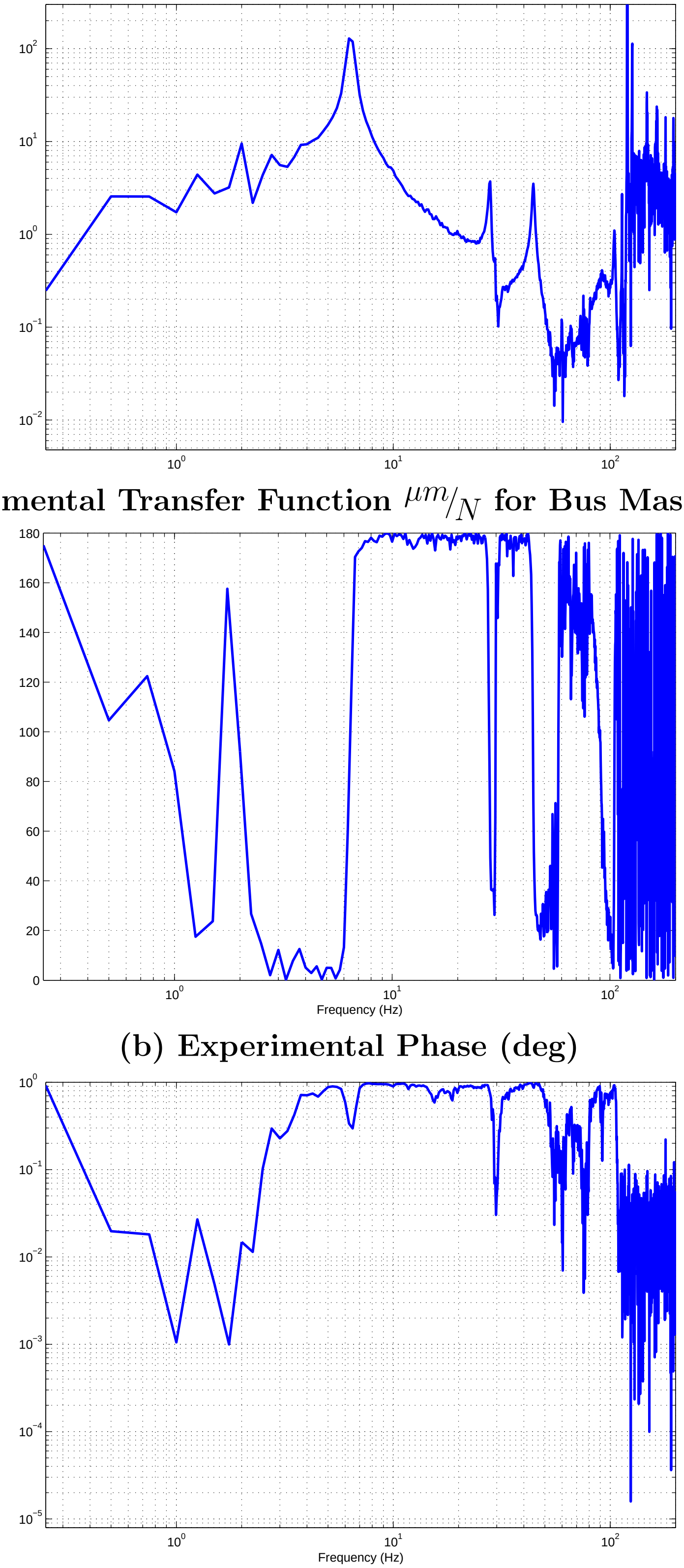
Figure 5-6: Input Noise Spectrum (N^2/Hz)

as the output. Calibration factors were used to change the units from Volts/Volts to ${}^{\mu m}/_{N}$. The experimental data shown in Figure 5-7 correspond to an OPTSIM configuration of $V_S = 0.2V_{RMS}$ and $m_p = 100$ lbs. Before each test run, the gap sensor is centered using the multi-meter to be in the center of its range. This needs to be done, because the variable weight of the spacecraft bus changes the absolute vertical position of the composite base plate, and the gap sensor tip needs to be within about 0.05 mm of the object it is sensing (see the gap sensor calibration curve in Figure 2-4) in order to be able to use a constant calibration factor. All subsequent transfer functions shown in this Chapter will relate the gap sensor output to the load cell input with the units $\frac{\mu m}{N}$.

5.2 Application of 3-Step Tuning Procedure

5.2.1 Initial Simulation Model

Since this system is purely structural (no controls or optical metrics), it can be completely modelled using the finite element method (FEM). This is assumed to be a steady state problem, with no controlled actuators, and therefore no sensor and actuator dynamics. The initial simulation model was defined using MSC/NASTRAN 2001. A wireframe model produced using PATRAN can be seen in Figure 5-8. The overall structure is made up of beam, plate, spring, and concentrated mass elements. The truss model is comprised of single beam elements with concentrated masses at the nodal points (representing the tapped aluminum nodes that hold the truss bars together). The optical plates, composite base plate and the thin top aluminum plate are all modelled as plate elements. NASTRAN allows for composite plate elements, and the base plate was modelled as such. The bases of the four springs supporting the structure were constrained in all six degrees of freedom since they were connected to the heavy cement block (clamped boundary conditions). Initially the gravity load was not applied, its application will be discussed in the next section. NASTRAN provides the eigenvalues (modal frequencies), eigenvectors (mode shapes), and strain



(a) Experimental Transfer Function $^{\mu m}/_{N}$ for Bus Mass = 100 lbs

(b) Experimental Phase (deg)

(b) Corresponding Experimental Coherence Function

Figure 5-7: Initial Experimental Transfer Function

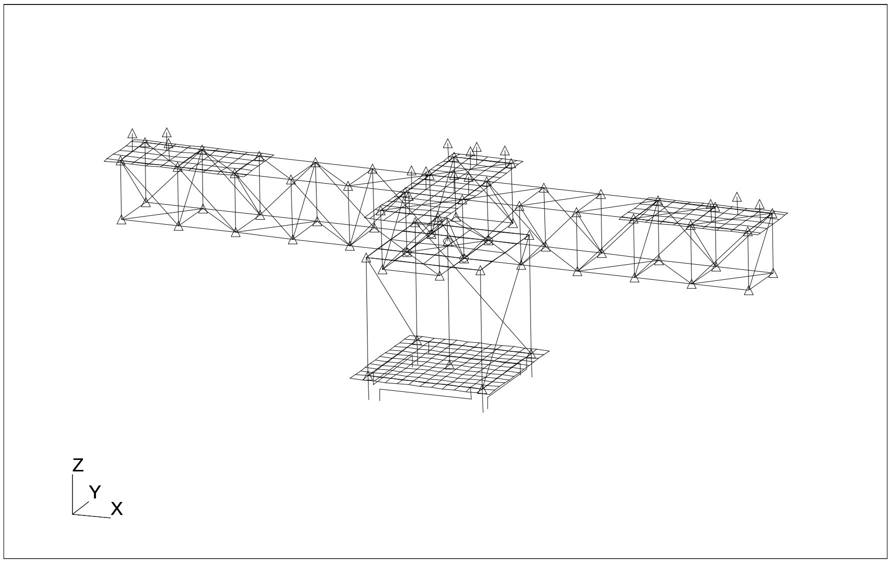


Figure 5-8: OPTSIM wireframe model

energies, which are then converted into MATLAB® data structures using the DOCS code [6]. The modal damping was initially set to 0.01 for all modes. By selecting one input node and one output node, a state space system is built, and the corresponding transfer function (shaker force input to gap sensor output) can be extracted. The selection of the input and output nodes determine the structure of the state space input and output matrices, B_0 and C_0. The finite element analysis provides the matrix of natural frequencies, Ω ($\Omega = \text{diag}(\omega_1, \omega_2, \ldots, \omega_n)$), and the modal matrix, Φ ($\Phi = [\phi_1, \phi_2, \ldots, \phi_n]$). The modal matrix is generally mass normalized, so that $\Phi^{\mathrm{T}} M \Phi = 1$ where M is the mass matrix. The state space matrices are then formed as in Equation 5.1.

$$\mathrm{A} = \begin{bmatrix} 0 & \mathrm{I} \\ -\Omega^2 & -2\mathrm{Z}\Omega \end{bmatrix}, \mathrm{B} = \begin{bmatrix} 0 \\ \Phi^{\mathrm{T}}\mathrm{B}_0 \end{bmatrix}, \mathrm{C} = [\mathrm{C}_0\Phi] \tag{5.1}$$

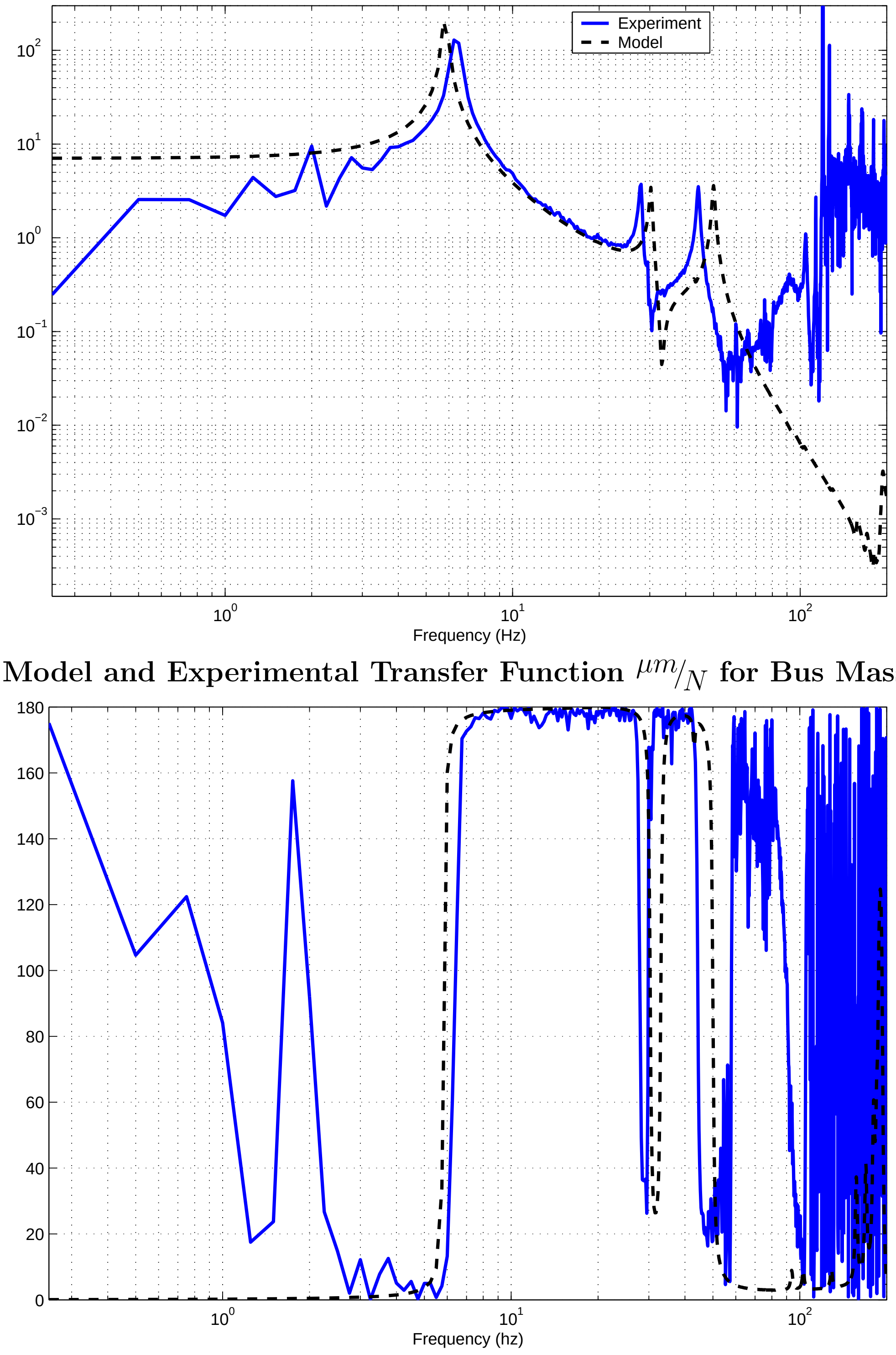
The transfer functions can then be obtained from

$$G(s) = \mathrm{C}(s\mathrm{I} - \mathrm{A})^{-1}\mathrm{B}. \tag{5.2}$$

The model transfer function can then be compared directly with the experimentally determined transfer function, which can be seen in Figure 5-9. Several things can be observed from Figures 5-7 and 5-9. First of all, the coherence function shows that the experimental data is only reliable from about 5 Hz to about 100 Hz. This region shall be referred to as the valid region. Since only the first three visible modes (shown in transfer function) will be used for tuning, and since all these modes fall within the valid region, this data is acceptable. The lower bound is limited by the low frequency power output of the shaker, and the upper bound (100 Hz) is limited by the upper corner frequency of the input noise spectrum (see Figure 5-6). Within the valid region, the transfer functions seem to match well. First, the roll-off (slope of the transfer function) from the first mode to about 60 Hz matches well at approximately -40 dB/dec. Secondly, the low frequency asymptote (at 4 Hz) matches well. It is not immediately clear that the high frequency asymptote matches, since there is a definite roll-up of the experimental transfer function after about 60 Hz. However, at 100Hz, unidentified modes might be obscuring the asymptote line. The pole/zero sequence within the valid region seem to match as well (pole-pole-zero-pole). While the first model mode (at about 6 Hz) under-predicts the experimental frequency, the second and third model modes (30 Hz and 50 Hz) are both high with respect to the frequency of the experimental modes. Table 5.1 shows all the modes from the simulation model under 100 Hz and the author's labelling. It should be noted that the author is numbering only the modes immediately visible on this transfer function, and not the modes identified through a finite element analysis (these modes exist, but are not observable by the select type, location and orientation of the sensor). Please refer to Table 5.3 for a listing of modal frequencies.

5.2.2 Step 1: Coarse Tuning

This experimental data described in the previous section will now serve as the "truth" model. It is assumed that the experimental data was taken in an appropriate manner, and that it accurately describes the testbed dynamics (according to the type and placement of the actuators and sensors used). All subsequent changes will be made



(a) Initial Model and Experimental Transfer Function $^{\mu m}/_{N}$ for Bus Mass = 100 lbs

(b) Initial Model and Experimental Phase (deg)

Figure 5-9: Comparison of Initial Model and Experimental Transfer Functions

Initial Model Modes (Hz)	Labelling
1.9125	
2.7470	
5.8362	First
7.6362	
13.9133	
19.4963	
30.2968	Second
43.0362	
50.0060	Third
63.3664	
64.2923	
93.8634	

Table 5.1: Normal Modes for Initial Model

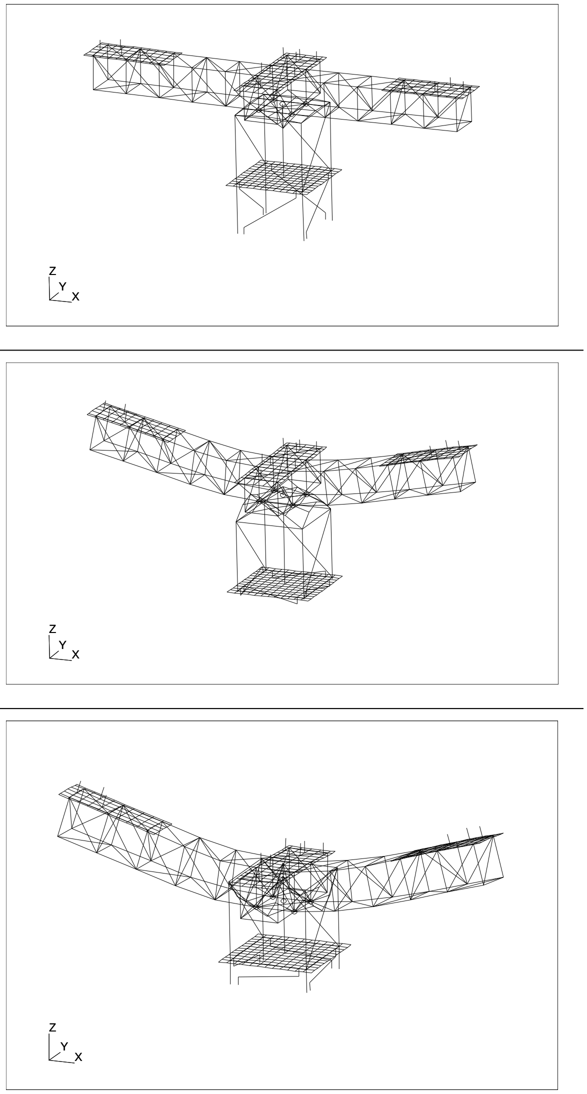
Element Group	First Mode	Second Mode	Third Mode
Springs	79.90	2.00	0.27
Composite Plate	0.11	0.05	0.05
Lower Bay	17.36	4.04	4.23
Top Al Plate	2.09	41.56	46.37
Truss Bars	0.54	51.89	48.30
Optical Elements	0.0	0.46	0.78

Table 5.2: Strain Energy Percentages of Total for Initial Model

to the simulation model in order to reproduce the experimental data. The amount of initial mismatch can be seen graphically in Figure 5-9.

The modeshapes of the first, second, and third modes can be seen in Figure 5-10. The first modeshape shows the testbed from the baseplate up moving up and down on the supporting springs. This resembles the motion of a simple mass-spring system, and will therefore be called the fundamental mode. This is supported by the strain energy analysis: 79.9% of the strain energy for this mode is in the springs. The second and third modeshapes shows the first and second bending modes of the truss in the x-z plane (respectively). The strain energy for these modes resides largely in the truss bars, but a large portion also lies in the top aluminum plate (see Table 5.2).

Now we will step through the checklist provided in Table 2.1. A linear model



(a) **First Mode : 5.83 Hz**

(b) **Second Mode : 30.30 Hz**

(c) **Third Mode : 50.01 Hz**

Figure 5-10: Modeshapes for Initial Model

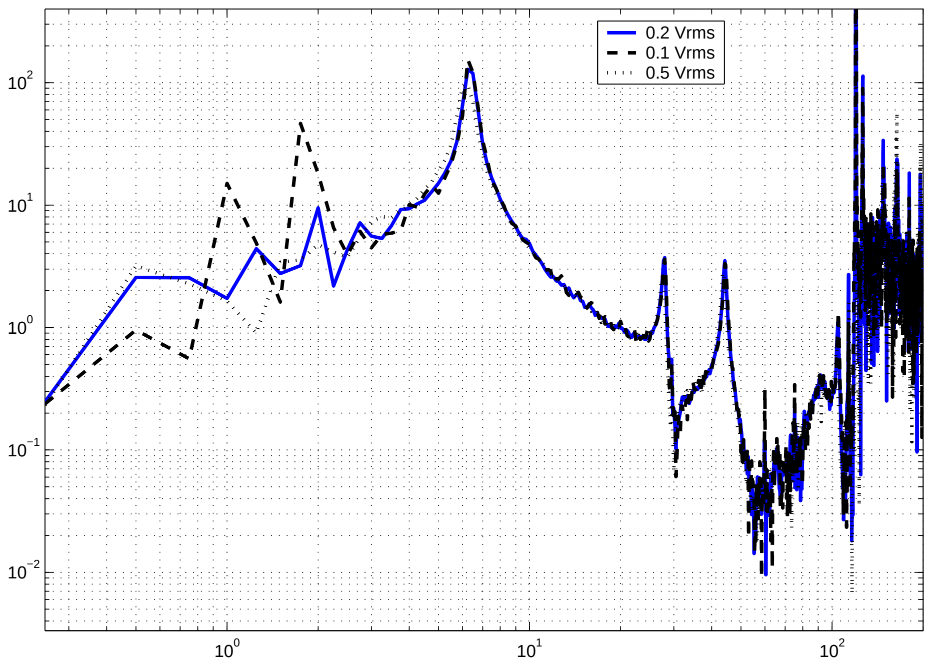
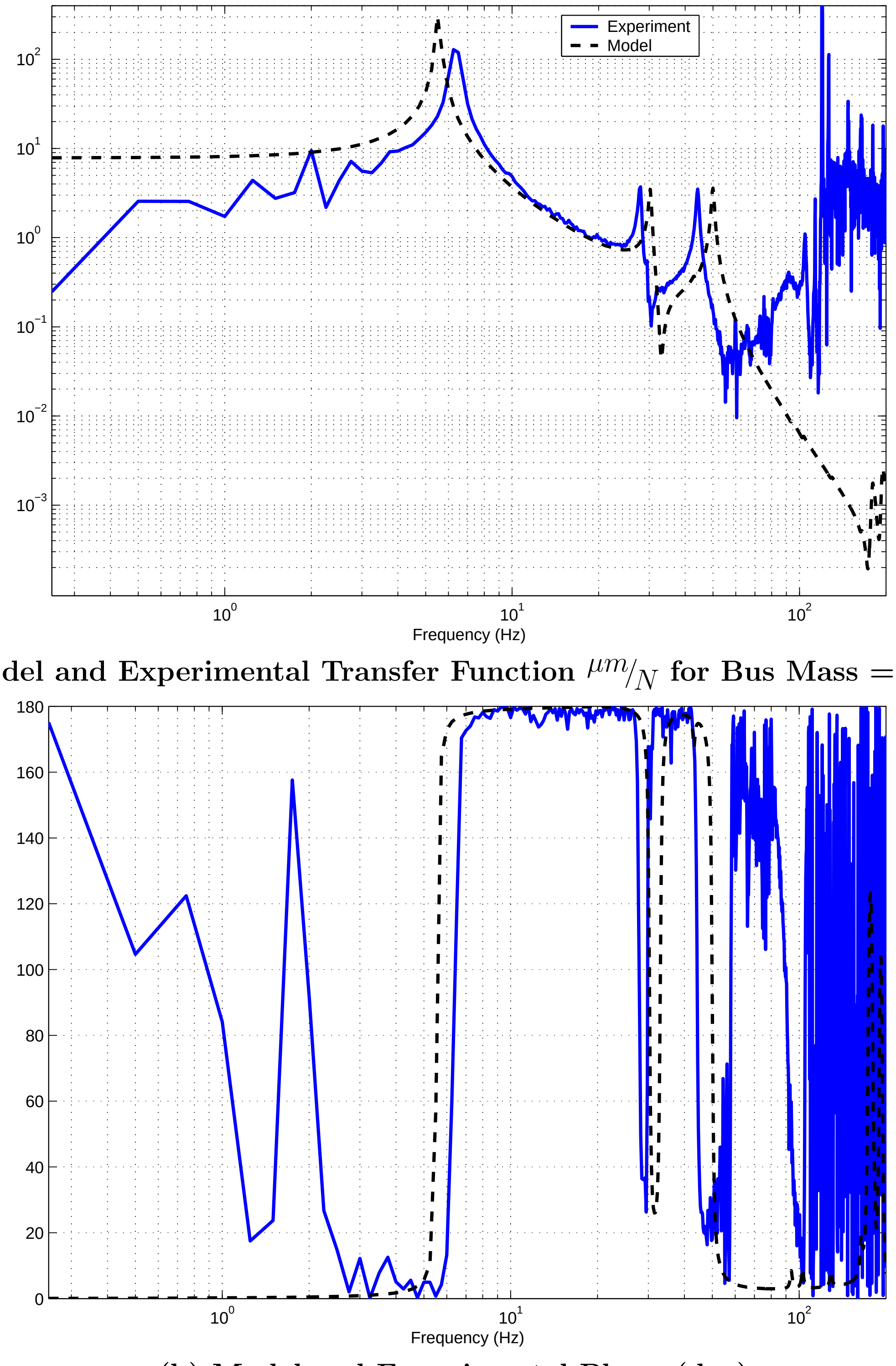


Figure 5-11: Transfer Function for Varying Input Voltages $\frac{\mu m}{N}$ Testbed Configuration: $m_p = 100$ lbs.

is being used, so a good first step is to check the linearity of the system. This can be done using by changing the magnitude of the force input into the system. The magnitude of the output should change in direct proportion to the change in input magnitude. An advantage of using a transfer function is that it should not change appreciably when increasing the force input for a linear system up to a certain limit. In Figure 5-11 the transfer functions for three different forcing levels are plotted. For the valid range (here 5-100Hz), they match extremely well. It will therefore be assumed that this system is sufficiently linear in our region of interest to continue modeling using linear models.

Now that the system is verified to be linear and we are confident that the modes match properly, we can focus on aligning the transfer functions in Figure 5-9. This will be done in two steps: coarse tuning (manual) and fine tuning (automated). The fundamental mode contributes the most to the output displacement (highest magnitude mode in the transfer function), therefore it will be addressed first. It is common practice to address mismatch of this type either in decreasing order of magnitude or increasing order of frequency. For a simple single degree of freedom mass-spring system (which the fundamental modeshape resembles), the natural frequency is $\sqrt{\frac{k}{m}}$. Therefore, the fundamental frequency heavily depends on the properties of the spring and the total mass of the testbed. The value for the spring constant in the model was taken directly from the product specifications (168 lbs/in). The four springs were set up underneath a plate, on which varying mass (0 to 200 lbs. in increments of 10 lbs.) was placed. The resulting change in length was then recorded for each mass value. The average experimental value was determined to be 145.25 lbs/in in the region of interest ($m_p = 0$ to 200 lbs). It is clear that the effective spring stiffness is much less than the catalogue value. This value was changed in the model, the new transfer function can be seen in Figure 5-12. It can be seen on comparison of Figures 5-12 and 5-9 that the fundamental mode decreases in frequency with the smaller spring constant, as expected (from 5.75 Hz to 5.5 Hz). Also, the frequencies of the second and third modes remain unchanged. Since the frequency of the fundamental mode depends so heavily on the spring constant and total mass of the system, this transfer function implies that there might be extra mass present in the simulation model. The mass of the components was measured to best of the author's ability, and the total mass of the testbed was experimentally determined to be 105.9 kg. The weight of the modelled system is 102.7 kg, a difference of only 3%. Upon careful consideration of the possible sources of this error, the optical breadboards and components were re-measured. They were determined to be slightly heavier than the model was predicting. This was changed in the model, and the total model mass increased to 103.3 kg, lowering the error to 2.5%. A possible source of error here is the compounding effect of small errors in the mass of the truss bars or truss nodes. There are 184 truss bars and 64 nodes, so a small error in the mass of one truss bar (5 g), produces a large overall error (0.92 kg). Another possible source of error is the measurement error. It should be noted that not all of the strain energy of the fundamental mode belongs to the springs, therefore the modal frequency is affected by the properties of the rest of the system (see Table 5.2). This could account for much of the error between the model and experimental fundamental frequency.



(a) Model and Experimental Transfer Function $^{\mu m}/_{N}$ for Bus Mass = 100 lbs

(b) Model and Experimental Phase (deg)

Figure 5-12: Transfer Function and Phase Plot with New Spring Constant

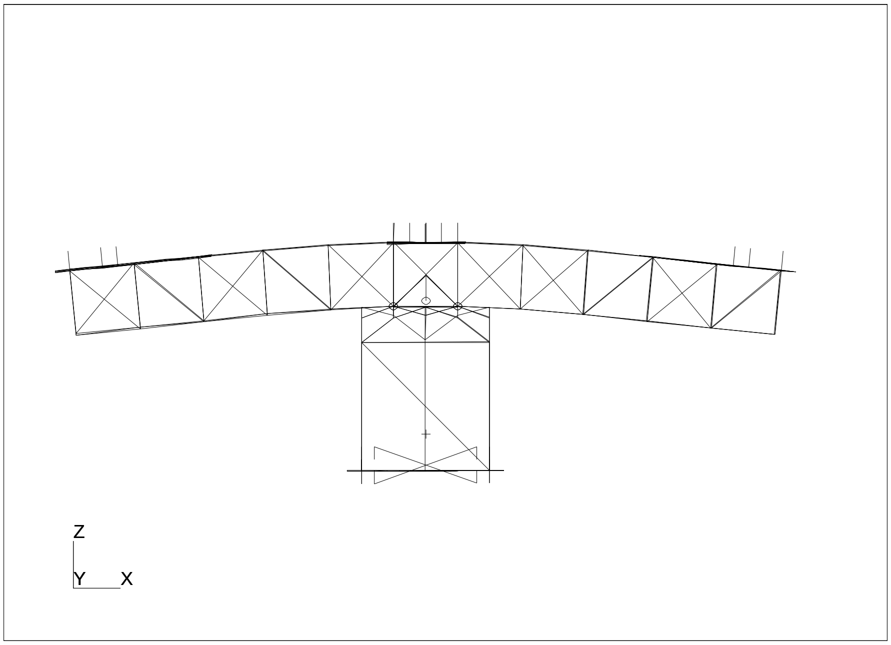


Figure 5-13: Static Effect of Gravity on OPTSIM

Next, a gravity field was added to this model, but produces no appreciable difference (see Table 5.3). The initial condition for the finite element dynamic analysis, however changed. This condition consisted of the the compression of the base springs and a drooping of the truss arms, as seen in Figure 5-13. The deformations shown in this figure are not to the scale of the structure.

A known problem in finite element software has to do with the rotation of the plate elements about their normal axis. In the initial model all the plate z rotations were constrained using single point constraints. However, NASTRAN provides a parameter (K6ROT) that resists motion in that degree of freedom, and eliminates the singularities that form in the K matrix (due to excessive pivot ratios). By removing the single point constraints, and introducing this new parameter, the boundary conditions were amended, and the singularities were avoided. There was no appreciable difference in the results when K6ROT was set to 1 or to 100. The value of 100 was used. Table 5.3 shows the new modal frequencies using the K6ROT parameter.

The location of the gap sensor was originally modelled to be in the direct center of

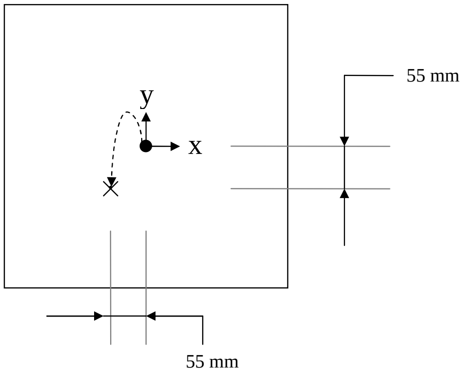


Figure 5-14: Location of Gap Sensor on Base Plate

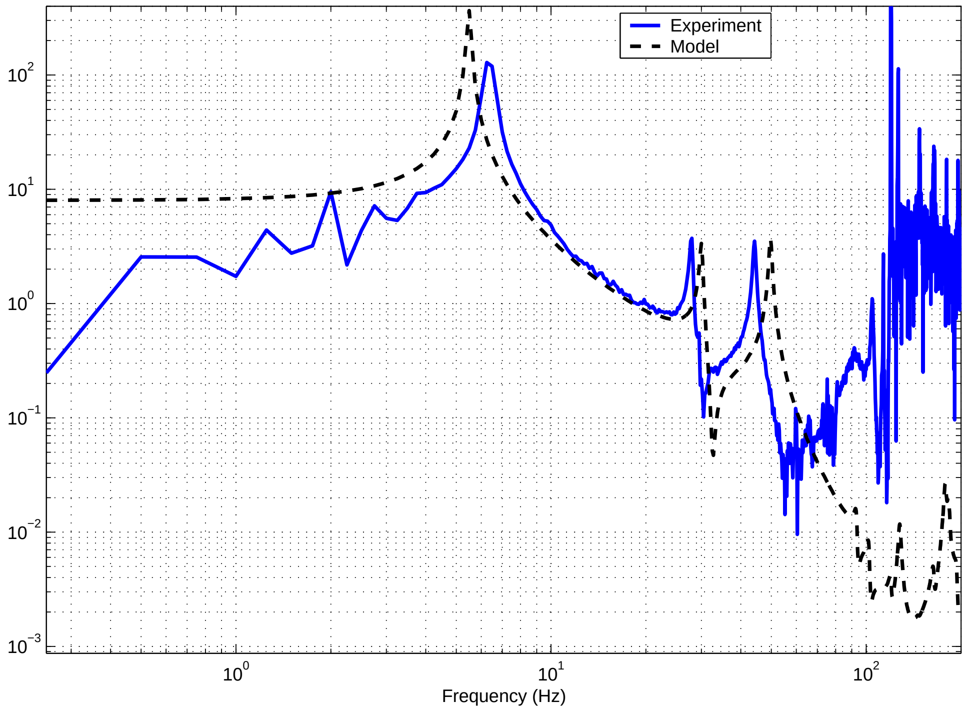
Mode No.	First	Second	Third
Experimental Modes	6.25	28.0	44.25
Initial Model	5.84	30.30	50.00
Changed Spring Constant	5.55	30.26	50.00
Changed Breadboard Mass	5.53	30.03	49.75
Added Gravity Field	5.53	30.03	49.75
K6ROT	5.48	30.02	49.73
Moved Sensor	5.48	30.02	49.73

Table 5.3: Modal Frequency (Hz)

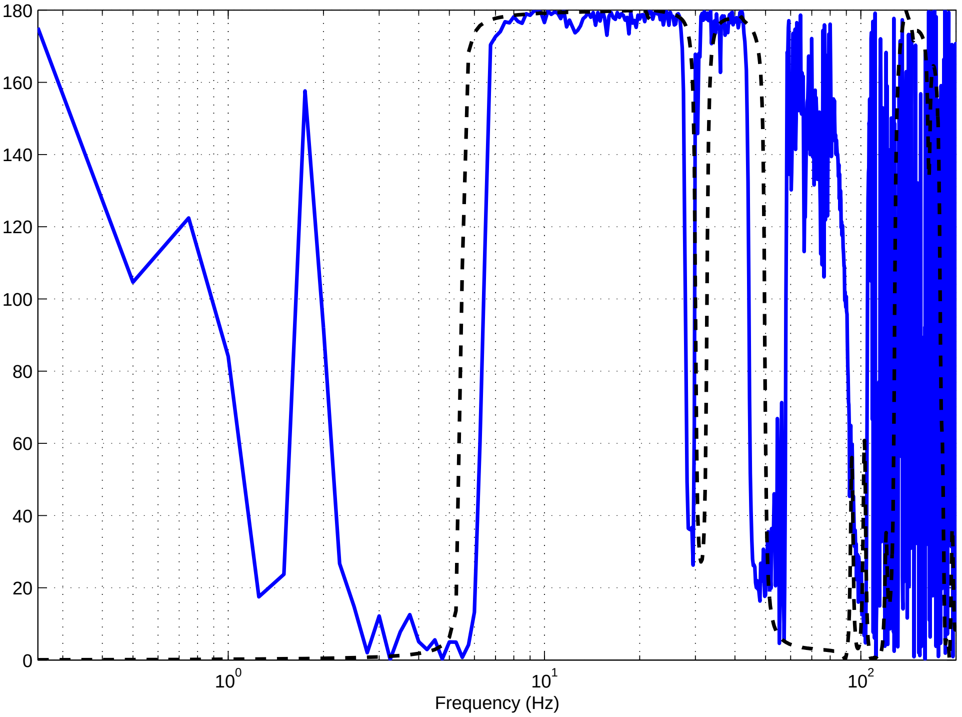
the base plate. Upon re-measuring, this position was found to be slightly incorrect. The sensor placement on the model was then moved the appropriate amount (see Figure 5-14). The black circle represents the original gap sensor location, and the X is the new sensor position (figure not to scale). Again, this does not have an appreciable difference on the modal frequencies, but it nevertheless is a better representation of the system.

Figure 5-15 shows the results of the altered model incorporating all of the changes described in this section. It can be seen that the experimental and model transfer functions not only have the same pole/zero sequence and the same roll-off, but also are very similar in terms of modal frequency. The modal frequency matching metric from Chapter 2 is applied here. The experimental modes were picked off from the plot.

Table 5.4 shows that the frequency error is below ten percent for the second mode,



(a) Model and Experimental Transfer Function $^{\mu m}/_{N}$ for Bus Mass = 100 lbs



(b) Model and Experimental Phase (deg)

Figure 5-15: Transfer Function and Phase Plot at the End of Coarse Tuning

Mode No.	Experimental Modes	Model Modes	$\frac{F_{mod}-f_{exp}}{f_{exp}}$
First	6.25	5.48	-0.1232
Second	28.00	30.02	0.0721
Third	44.25	49.73	-0.1238

Table 5.4: Percent Frequency Errors

but is about 12% for the first and third modes. Furthermore, the objective function cost, as given by Equation 3.13 (log-cost) can also be computed for the initial and fitted models. The cost for the initial model is 3.7662 and the cost for the new model is 4.4368, which indicates a slight worsening in the model-experiment matching. Even though the model has been worsened in terms of the objective function cost, it has been improved since non-parametric error has been reduced. Since all of the non-parametric errors identified in this section were implemented in the model properly, this model will be retained. This will be considered sufficient for the present problem and proceed to the next step: Automated Model Updating.

5.2.3 Step 2: Automated Model Updating

A sensitivity analysis was performed on this system with respect to its structural elements. First the model was represented in a parameter-dependent state space form,

$$S(p) = \begin{bmatrix} A(p) & B(p) \\ C(p) & D(p) \end{bmatrix}. \tag{5.3}$$

Then the analytical sensitivities were calculated by taking the partial derivatives of the state space system, $S(p)$, with respect to several different parameters,

$$dS(p) = \begin{bmatrix} \frac{\partial A(p)}{\partial p} & \frac{\partial B(p)}{\partial p} \\ \frac{\partial C(p)}{\partial p} & \frac{\partial D(p)}{\partial p} \end{bmatrix}. \tag{5.4}$$

The resulting state space system is then evaluated at the nominal parameter values. The sensitivity result is measured in units of root mean square output performance (in this case, displacement of the base plate in μm as measured by the gap sensor), and can be seen graphically in Figure 5-16. These sensitivity values show the relative impact of a similar percentage change of each parameter on the base plate displacement. The sensitivity analysis shows that the most sensitive components are the Young's modulus of the top aluminum plate and the rotational links, and the spring constants. These are the parameters that will be chosen for the automated updating. The Young's modulus parameters are both for aluminum. The various aluminum alloys found in [4] have a variation of about +/- 7%. These values (plus a buffer amount of 3%) are used as a guide for the upper and lower bounds. The spring constant was experimentally determined earlier, but this could be incorrect due to measurement error. If it is assumed that the distance measurement was accurate only to 0.1 inches, and that the weight measurement was accurate only to 1 lbs., then the corresponding variation in the spring constant would be +/-2.0%. A buffer of 3% is also added. The modal damping coefficients are also added as parameters to be updated.

The cost function that will be used is the logarithmic frequency response error, Equation 3.13. The frequency response will be weighted in the cost function according to the valid range. From 5 - 100 Hz, the weighting will be 1.0, and outside that range, the weighting will be 0.1; this effectively discounts the regions where the coherence is low. Transfer function values below 0.07 will be ignored by this algorithm; this simulates the noise floor. The automated updating was performed using DOCS updating code [6]. The updated parameter values can be seen in Table 5.5, and the updated transfer function can be seen in Figure 5-17. Substantial improvement can be seen over the the initial model and experimental transfer functions, however, there is still an appreciable error in each mode. Note that from Table 5.5 the spring constant and Young's modulus of the plate are both at their extremes (+5% and -10.0%, respectively), this would indicate that increasing the spring constant and/or decreasing the plates Young's modulus further would cause the model to match better to the experiment. However, these further changes would not produce a physically meaningful model. It is still not clear if these parameters in the automated tuning where changed in the proper manner, or even if they were the correct parameters to change. The objective function values for the initial model, the model after Step 1,

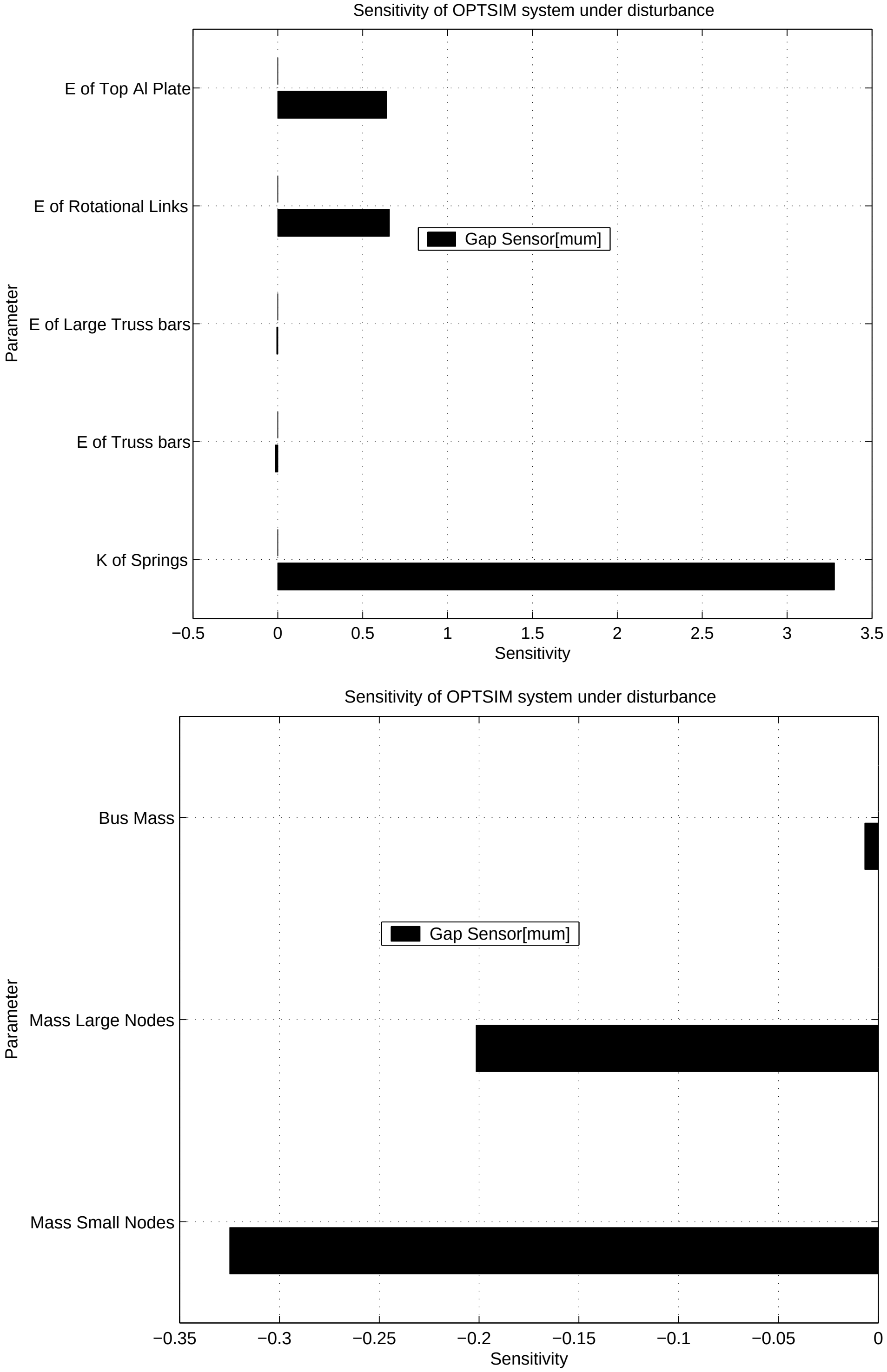


Figure 5-16: Sensitivity Analysis Results

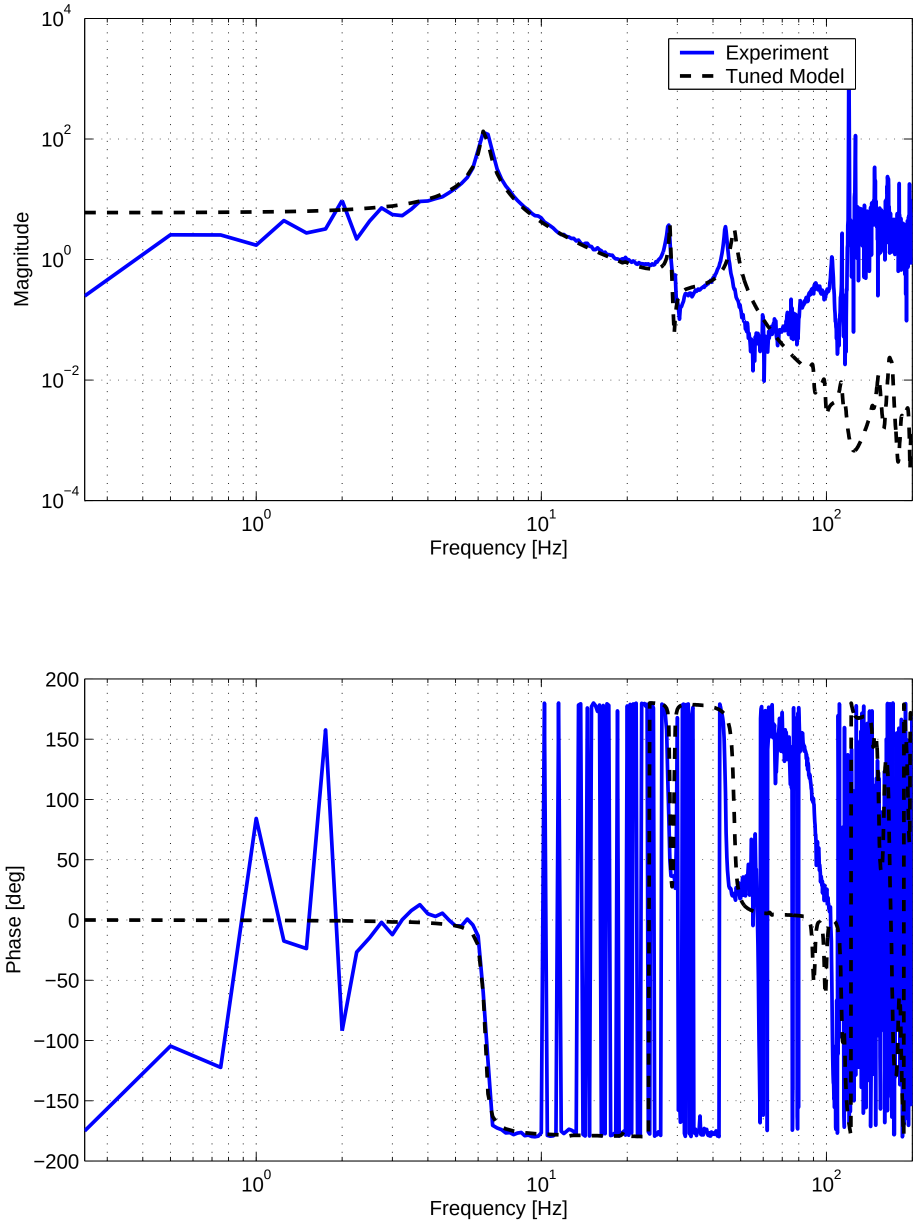


Figure 5-17: Experimental and Tuned Transfer Functions $\frac{\mu m}{N}$

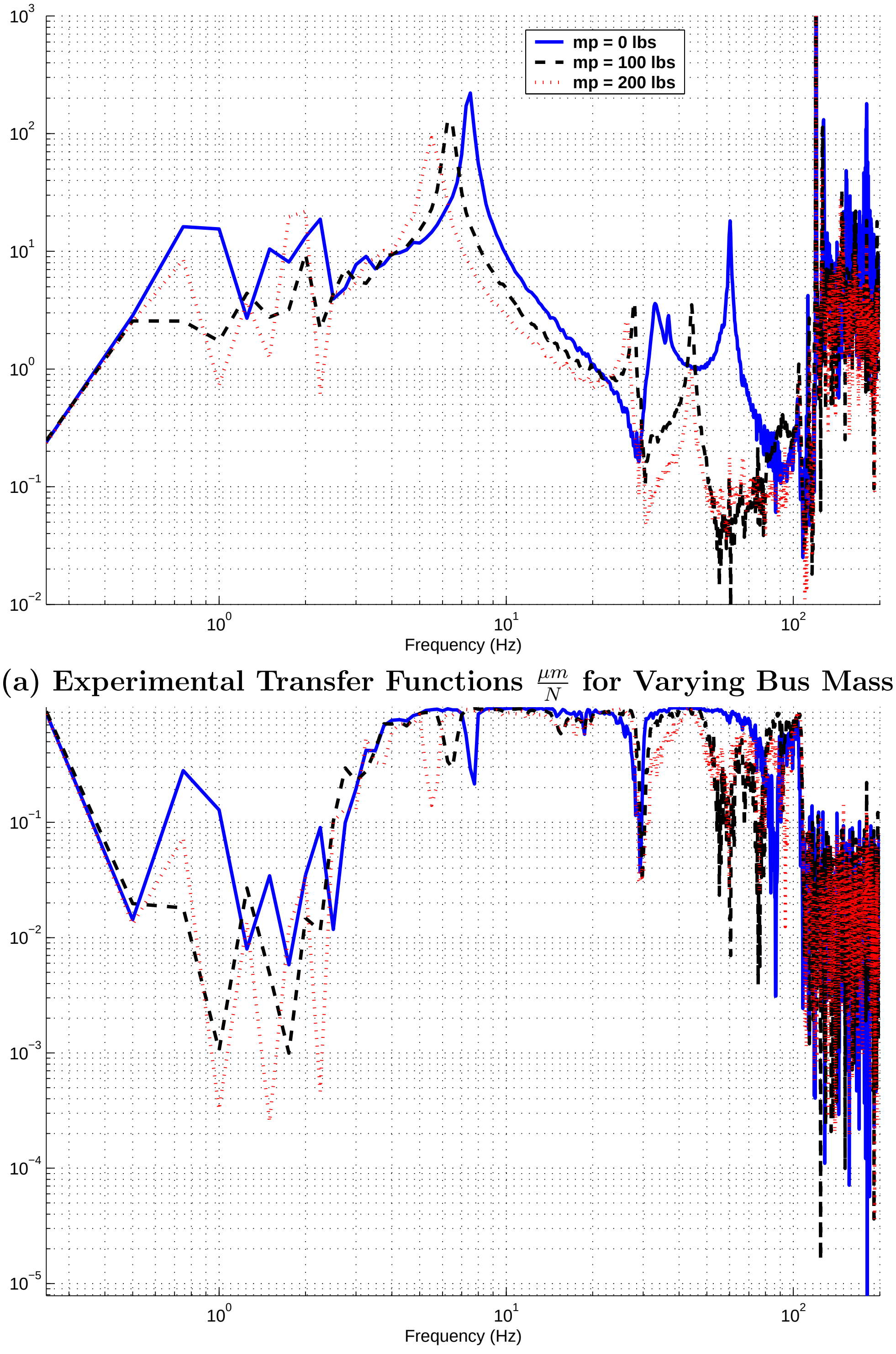
Parameters	Original Value	Updated Value	Percent Change
Spring Constant	29198.27	3.0658e+4	5.0
Young's Modulus (Plate)	7.20+10	6.4800e+10	-10.0
Young's Modulus (Links)	7.20+10	7.7767e+10	8.0
Damping ratio 1^{st}	0.01	0.0876	876.0
Damping ratio 2^{nd}	0.01	0.0487	487.0
Damping ratio 3^{rd}	0.01	0.0672	672.0

Table 5.5: Updated Parameter Values

and the model after Step 2 (tuned model) are 3.7662, 4.4368, and 1.3942 respectively. The objective function cost has decreased by more than 60% due to the changes made using automated tuning.

5.2.4 Step 3: Model Validation and Tracking

The next step is to change the configuration of the system, and to observe the change in the updated simulation model and in the testbed response. These altered systems will then be compared to determine if the correct parameters where changed in the proper way in Step 2. The change implemented was to alter the mass of the bus in the center of OPTSIM. It was altered from 100 lbs to 0 lbs and to 200 lbs. The measured transfer and coherence functions are shown in Figure 5-18. Again, it can be observed that the coherence function is high for the same valid interval of 5-100 Hz. The fundamental modal frequency decreases with increasing bus mass. This follows the modeshape analysis which shows that the fundamental mode resembles a single degree of freedom mass-spring system. The natural frequency is then $\sqrt{\frac{k}{m}}$. Therefore, as the bus mass increases, so does the mass of the system, and the natural frequency decreases. It can also be observed that the second and third modes are fundamentally different in the case of zero bus mass versus non-zero bus mass. The pole/zero sequence changes, as well as the modal frequencies. The second and third mode in the massless case increase by a large amount (to about 34 Hz and 60 Hz). Checking the coherence function for the massless case, it is shown to be high until approximately 70 Hz.



(a) Experimental Transfer Functions $\frac{\mu m}{N}$ for Varying Bus Mass

(b) Corresponding Experimental Coherence Functions

Figure 5-18: Experimental Transfer and Coherence Functions for Varying Bus Mass

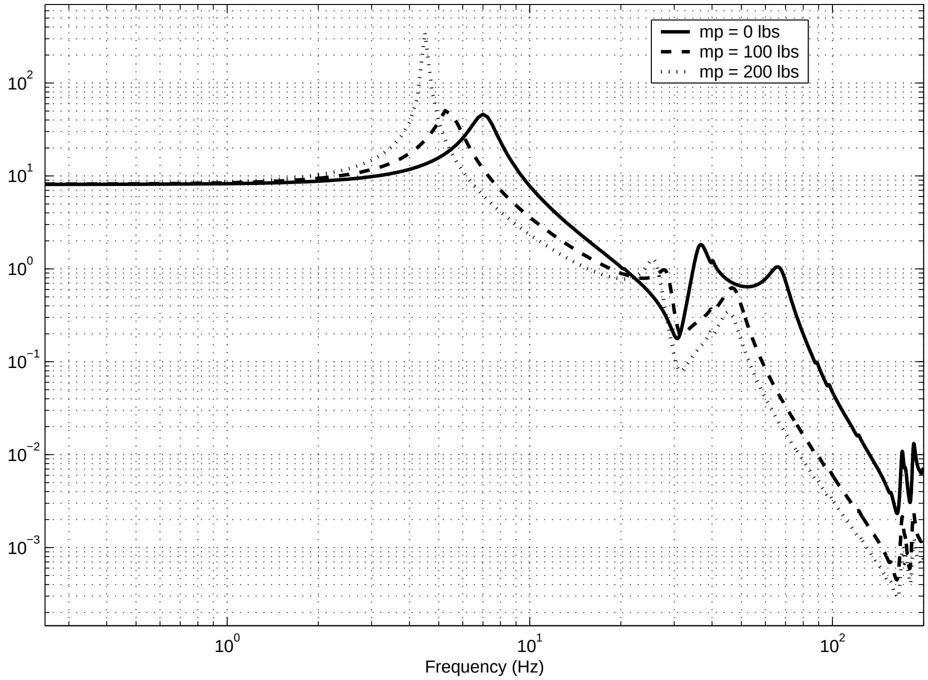
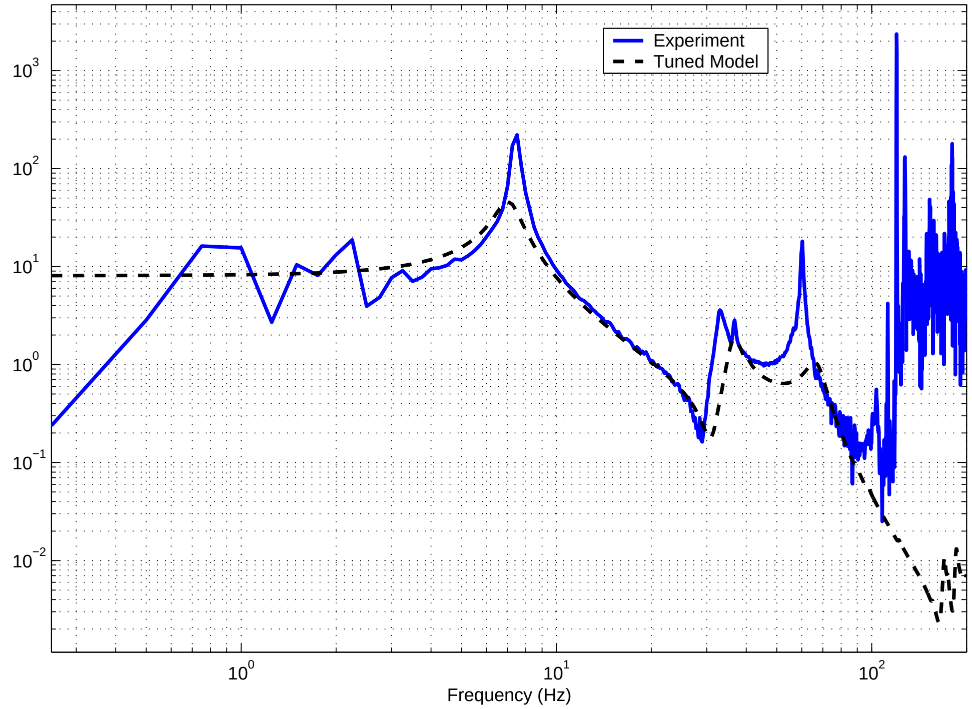
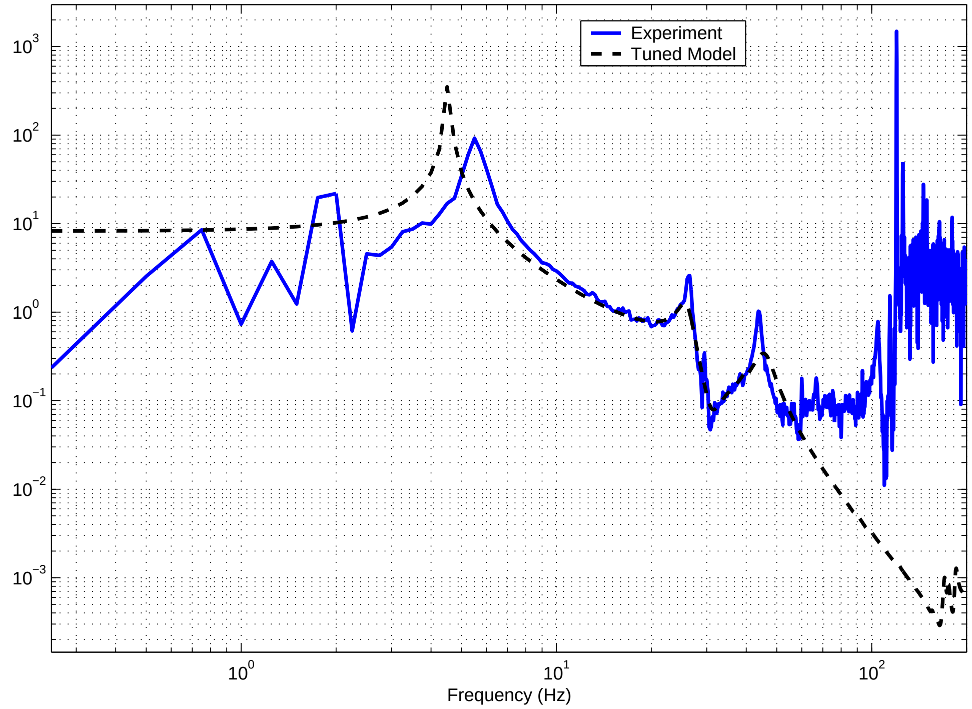


Figure 5-19: Simulation Model Transfer Functions $\frac{\mu m}{N}$ for Varying Bus Mass

The concentrated mass entry was changed in the updated NASTRAN model, and the resulting transfer functions can be seen in Figure 5-19. The features that were present in the experimental transfer function look to be repeated fairly well in the simulation model. It remains to compare model and experiment for the new configurations (see Figure 5-20) using the metrics introduced in Chapter 4. While the new pole/zero sequence for the $m_p = 0$ lbs configuration tracks well, the modal damping ratios for all three modes seem to be too large in the model. Also the second and third modes are slightly over-predicted. It is common for model modes to be higher in frequency than experimentally determined modes, it shows that the model is representing the system as too stiff, indicating missing compliance. The model for the $m_p = 200$ lbs configuration, tracks extremely well to the experimental, in terms of modal frequency for the second and third mode. The frequency for the first mode, however, is increased. One could conclude then, that the presence of bus mass presents a uniquely different problem, most likely due to the behavior of the base plate under loading. Table 5.6 shows the modal frequencies in the three



(a) Model Transfer Function $\frac{\mu m}{N}$ for $m_p = 0$ lbs



(a) Model Transfer Function $\frac{\mu m}{N}$ for $m_p = 200$ lbs

Figure 5-20: Experimental Transfer and Coherence Functions for Varying Bus Mass

Configuration: $m_p = 0$ lbs		
Mode No.	Experimental Modes	Model Modes
First	7.50	7.07
Second	33.00	36.37
Third	60.25	66.97
Configuration: $m_p = 100$ lbs		
Mode No.	Experimental Modes	Model Modes
First	6.25	5.48
Second	28.00	30.02
Third	44.25	49.73
Configuration: $m_p = 200$ lbs		
Mode No.	Experimental Modes	Model Modes
First	5.50	5.26
Second	26.50	25.94
Third	43.75	45.83

Table 5.6: Model and Experimental Modes for Three Configurations

configurations.

The objective cost function for the 0 lbs, 100 lbs, and 200 lbs configurations using the tuned model are $J_1 = 1.6968$, $J_t = 1.3942$, and $J_2 = 3.3091$, respectively. The 200 lbs configuration seems to deviate significantly from the original configuration. Applying Equations 4.4 and 4.5, to OPTSIM give,

$$\mathrm{Y} = \frac{1}{J_t^2} \begin{bmatrix} J_t^2 & J_1 J_t & J_2 J_t \\ J_1 J_t & J_1^2 & J_1 J_2 \\ J_2 J_t & J_1 J_2 & J_2^2 \end{bmatrix} = \begin{bmatrix} 1.0 & 1.2170 & 2.3734 \\ 1.2170 & 1.4810 & 2.8883 \\ 2.3734 & 2.8883 & 5.6329 \end{bmatrix} \tag{5.5}$$

and

$$\frac{\|\mathrm{Y}\|_\infty}{(k+1)} = \frac{10.8946}{3} = 3.6315 \tag{5.6}$$

The multi-configuration metric is extremely large, exceeding by a wide margin the maximum value recommended in Chapter 4 (2.0). This value indicates that there is still some non-parametric error in the model, so the modeling assumptions should be re-examined; this corresponds to the iteration steps detailed in Figure 1-4.

Parameters	Original Value	Updated Value	Percent Change
Spring Constant	29198.27	3.0658e+4	5.0
Young's Modulus (Plate)	7.20+10	6.4800e+10	-10.0
Young's Modulus (Links)	7.20+10	7.7767e+10	8.0
Damping ratio 1^{st}	0.01	0.0876	876.0
Damping ratio 2^{nd}	0.01	0.0487	487.0
Damping ratio 3^{rd}	0.01	0.0672	672.0

Parameters	Alternately Updated Values	Percent Change
Spring Constant	3.7e+4	+23.3
Young's Modulus 1st Material Card	6.00+10	-16.7
Damping ratio 1^{st}	0.01	0.0
Damping ratio 2^{nd}	0.01	0.0
Damping ratio 3^{rd}	0.01	0.0

Table 5.7: Comparison of Updated Parameter Values

5.3 Comparison to Alternately Tuned Model

As was mentioned in the beginning of this chapter, the initial model, the experimental data (all three configurations), and the DOCS plotting tools were presented to a colleague, Scott Uebelhart, who then tuned the model in the best way he saw fit. Scott investigated the model for changeable parameters, examined the testbed for possible sources of parametric variation, examined the data supplied for the material cards, and then used trial and error to adjust the Young's modulus of the material cards and the spring stiffness. He assumed that there was no non-parametric error present in the model, but he also noted that non-parametric error could still be a large contributor to model error. The result of the tuning was to increase (from the initial model) the Young's modulus of the first material card 23.3% from 30 kN/m to 37 kN/m and to decrease the spring stiffness from 72 GPa to 60 GPa, a change of -16.7%. Table 5.7 shows these changes and compares them with the previously tuned model. The first material card corresponds to the aluminum used for the small and large truss beams, the central bar, the top plate and the outer sandwich plates. Scott also noted that he had severe reservations about the appropriateness of the changed parameters, especially since the new Young's modulus appeared to be non-physical when consulting material tables for aluminum. The transfer function presented in

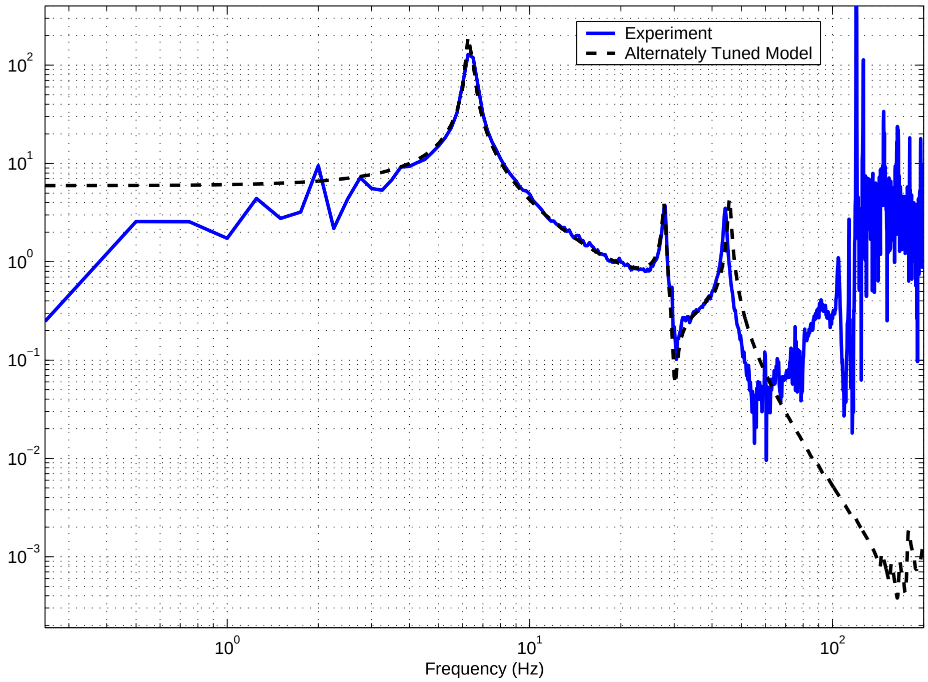


Figure 5-21: Transfer Function Comparison for mp = 100 lbs $\frac{\mu m}{N}$

Configuration	After Step 1	Updated Model	Alt. Updated Model
m_p = 0 lbs	-	1.6968	2.8001
m_p = 100 lbs	4.4368	1.3942	1.0745
m_p = 200 lbs	-	3.3091	1.4736
$\frac{\|\mathrm{Y}\|_\infty}{(k+1)} = \frac{12.9702}{3}$	-	3.6315	4.3234

Table 5.8: Table of Objective Function Costs and Multi-Configuration Metrics

Figure 5-21 illustrates the effects of these changes in the original configuration. Note that the experimental and alternately tuned model seem extremely well matched for this configuration. The associated objective function cost is 1.0745, even lower than the cost associated with the tuned model at the end of Step 2. When the configuration is changed, however, the costs go to 2.8001 and 1.4736 for 0 lbs and 200 lbs, respectively. Figures 5-22 and 5-23 shows the transfer functions for these new configurations and Table 5.8 shows these costs for the updated and alternately updated models. The associated Y matrix is

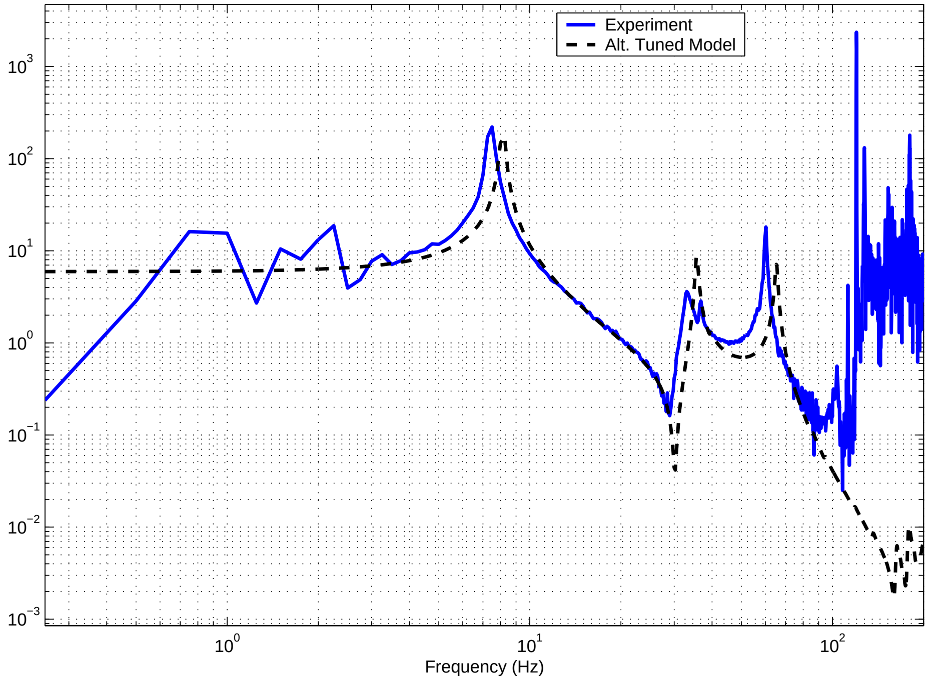


Figure 5-22: Alternately Tuned Transfer Function for mp = 000 lbs $\frac{\mu m}{N}$

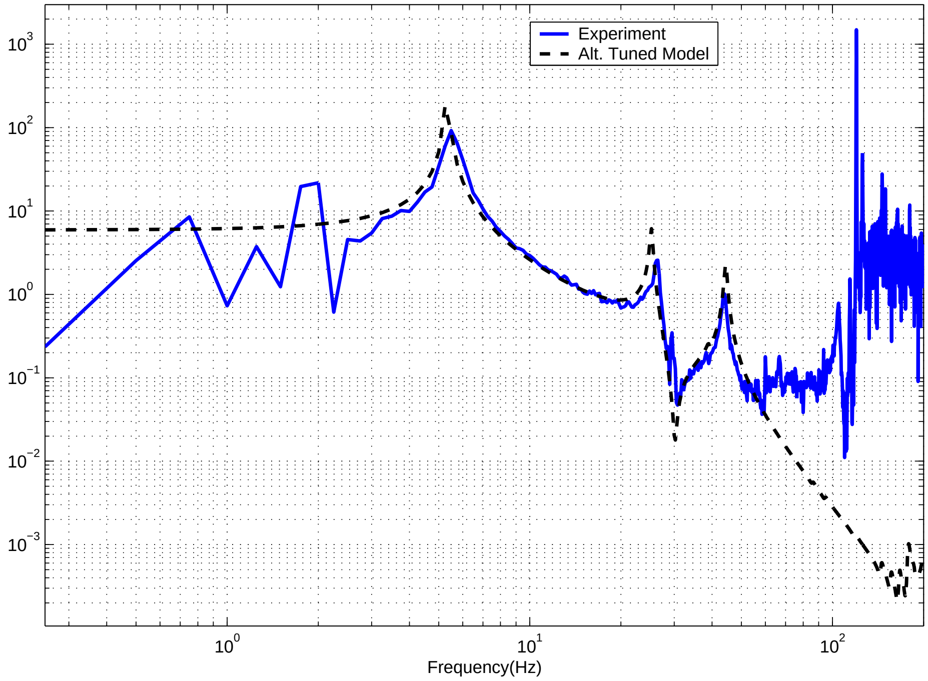


Figure 5-23: Alternately Tuned Transfer Function for mp = 200 lbs $\frac{\mu m}{N}$

$$\mathrm{Y} = \begin{bmatrix} 1.0 & 2.6059 & 1.3714 \\ 2.6059 & 6.7906 & 3.5738 \\ 1.3714 & 3.5738 & 1.8808 \end{bmatrix}, \tag{5.7}$$

and the multi-configuration metric is

$$\frac{\|\mathrm{Y}\|_\infty}{(k+1)} = \frac{12.9702}{3} = 4.3234. \tag{5.8}$$

The fact that the multi-configuration metric is lower for the model tuned using the three step procedure as opposed to the alternately tuned model indicates that the first model tracks better across configurations. Furthermore, if a model tracks well across configurations, then it also indicates that the model is a good representation of the physical system.

5.4 Chapter Summary

This chapter illustrated the three-step process on a structural testbed at MIT. A description of the testbed was provided, and its purpose was outlined. Engineering insight was applied for Step 1. Next, parameters were selected, and an automated model update was performed. Then the configuration of the system was changed in the experiment and the model, and the resulting data was examined using metrics presented in Chapter 4. This model was benchmarked against a traditionally tuned model (using trial and error to change parameter values). It was found that the model produced using the three-step procedure had a lower (i.e. better) multi-configuration metric than the one produced using traditional methods. This indicates that the model produced using the three-step procedure was more physically representative of the system, and therefore a more useful model.

Chapter 6

Results and Conclusions

6.1 Thesis Summary

This work presented a step-by-step process for model updating of large complex opto-mechanical systems. The focus was on open-loop structural systems, but by integrating the model into one state-space system, these methods can be applied to multi-disciplinary systems. The first step (Chapter 2) was to apply engineering insight to an initial simulation model. This consisted of several different types of model heuristics intended to create a model in which all the physical systems were represented. The second step, automated tuning in Chapter 3, outlined a procedure for parameter selection, objective function selection, and optimization problem setup. The third step, Chapter 4, provided a new metric which evaluated how well the tuned model from Step 2 tracked across configurations. This three step process was applied to a complex structural testbed at MIT, and the results were presented in Chapter 5. The hypothesis that the three step procedure will produce a model that effectively tracks along configuration changes was supported by empirical data in Chapter 5. Specifically, the multi-configuration metric (described in Chapter 4), which measures how well a particular model tracks the experimental model across configurations, was found to be better in the model that was tuned using the three-step method as opposed to a model using traditional methods (trial and error parameter changing).

6.2 Future Work

After applying Step 1 to OPTSIM, it was found that in order to match the experimental data well, the parameters (spring constant and Young's modulus of the top aluminum plate and rotational links) need to be altered well beyond their bounds. Future work would include a detailed procedure for going back to Steps 1 and 2 and changing the model tuning. Another aspect which should be addressed is the tracking of the modeshapes (eigenvectors) throughout the optimization process in order to make sure that the experimental and model modes are matched properly. Also, this procedure should be applied to a truly multi-disciplinary system, such as one that includes optics and controls. This will require careful incorporation, because controlled systems usually either design a controller that can address a wide variety of disturbances, or make use of on-orbit system identification. In order to build such a controller, an accurate model of the system is needed. If optics were added before the controls, then steps should be taken to ensure the model's accuracy. A useful validation of this work would be to add more inputs and outputs, and apply the three step procedure to a MIMO system. One could also investigate combining Steps 2 and 3, and perform multi-configuration tuning at the same time, in the presence of all the experimental data from all the configurations. This would reduce the iteration effort between Step 2 and Step 3 procedures. Subsystem assembly and the interfaces between subsystems is an important topic for multidisciplinary systems and should be addressed in the future.

Appendix A

NASTRAN code

A.1 Nominal Sample Problem

```
$ begin executive control
SOL 103
TIME 300
CEND
$ end executive control
$ begin case control
DISP=ALL
ECHO=NONE
ESE=ALL
GROUNDCHECK=YES
$
SUBCASE=10
        LABEL = GRAV load
       SPC = 2
       LOAD = 100

SUBCASE=20
        LABEL = normal modes with GRAV load
        STATSUB = 10
        SPC = 2
        METHOD = 100

METHOD=100
SPC=2
$ end case control
$ begin bulk data
BEGIN BULK
PARAM,POST,-1
```

```
PARAM,GRDPNT,0
PARAM,K6ROT,50.0
$1111111222222223333333344444444555555556666666677777777888888889999999
GRAV         999       0      1.      0.      0.   -9.81
LOAD         100     1.0     1.0     999
$------------------ Nodal Coordinates ------------------------------------
$1111111222222223333333344444444555555556666666677777777888888889999999
GRID         101          0.0000  0.0000  0.0000
GRID         102          0.0000  0.2000  0.0000
GRID         103          0.0000  0.2000  -0.100
$---------------- Boundary Conditions -----------------------------------
$1111111222222223333333344444444555555556666666677777777888888889999999
SPC1           2  123456     101
$------------------ Beam Element ----------------------------------------
$MAT1 is isotropic material
$MAT1   MID     E       G       NU      RHO     A(therm)TREF    GE
$1111111222222223333333344444444555555556666666677777777888888889999999
MAT1           1 7.00+10 2.60+10         2710.00 2.210-5 20.0000
$
$PBAR is a beam
$PBAR   PID     MID     Xsec    I1      I2      J       NSM
$1111111222222223333333344444444555555556666666677777777888888889999999
PBAR           2       1 7.0686-43.9761-83.9761-87.9522-8     0.0        +
$+       -3.00-2     0.0     0.0 -3.00-2 3.000-2     0.0     0.0 3.000-2 +
$+       0.53304 0.53304
$CBAR   EID     PID     GA      GB      X1      X2      X3
CBAR         201       2     101     102 1.00000 1.00000 1.00000
$-------------- Spring Element ------------------------------------------
$1111111222222223333333344444444555555556666666677777777888888889999999
$CELAS2 EID     K       G1      C1      G2      C2(component #)
CELAS2  202     30.0+6  102     1       103     1
CELAS2  203     30.0+6  102     2       103     2
CELAS2  204     30.0+3  102     3       103     3
CELAS2  205     30.0+2  102     4       103     4
CELAS2  206     30.0+2  102     5       103     5
CELAS2  207     30.0+0  102     6       103     6
$---------------- Concentrated Masses -----------------------------------
$1111111222222223333333344444444555555556666666677777777888888889999999
$CONM2  EID     G       CID     M
CONM2        203     102         10.0
CONM2        204     103         02.0
$1111111222222223333333344444444555555556666666677777777888888889999999
EIGRL   100          0.1      500.
ENDDATA
$ end bulk data
```

Appendix B

DOCS Tuning Code for Sample Problem

B.1 Analytical Calculations

```
%-----------------------Constants---------------------------
kgToNewtons =9.81; %m/s^2
%---------------------- Mass --------------------------------
%3 cm o.d. - 0.2 meters long
Asmall=((0.03/2)^2)*pi; %$m^2
density = 2710;  %kg/m^3
length = 0.2;    %m
mass_endBeam = 10;  %10kg
mass_endSpring = 2; %2kg
mass_beam = Asmall*density*length;
%------------------Spring elongation -----------------------
k = 30e+3;%30 kN/m
spring_length = 0.1;
F = mass_endSpring*kgToNewtons; %N
elongation= F/k; %m
NASTRAN_elongation = (-1.145887E-04 )-(-7.685887E-04 );
%----------------------------Tip displacement--------------------------------
%Inertias
%Ibsmall=(pi*(0.009398/2)^4/4)-(pi*((0.009398/2)-0.001524)^4/4); %m^4
I = pi/4*((0.03/2)^4);%X-sec inertia m^4
Jsmall=2*I; %approximation
E = 7.00e+10;     %Young's Modulus N/m^2
G = 2.6e+10;
%I = Ibsmall;
```

```
W = Asmall*density*length*kgToNewtons;
L = length;
P = (mass_endBeam + mass_endSpring)*kgToNewtons;  %Tip force N;
%Closed form solution
delta_b = (W*L^3)/(8*E*I) + (P*L^3)/(3*E*I); % in meters
%------------ Analytical Low Frequency Asymptote for TF-------------
TF_lowFreq = delta_b/(P)*1e6;  %output displacement/input force (mum/N)
```

B.2 Loading Nominal Model

```
%** LOAD MESHED SAMPLE PROBLEM SIMUALTION MODEL ****************
 derivative = 0;
%---------- Get the results from the .f06 file ------------------
fe_nastran('D:\SampleProblem\NASTRAN\Meshed\sampleproblemmeshed');
load D:\SampleProblem\NASTRAN\Meshed\sampleproblemmeshed;
phi = modesh;  %array of modeshapes
om = freq;       %modal frequencies
%Need to read in the bulk data deck
argin = {'D:\SampleProblem\NASTRAN\Meshed\sampleproblemmeshed.dat'};
SP_bulk_data= read_bulk_data(argin{:});
%Stick it into a DOCS model object - contains grid and element data
zeta_vec = 0.001*ones(size(om,1),1);
SP_docs_model = docs_femodel(phi,om,gid,'bulk_data',...
SP_bulk_data,'zeta',zeta_vec);
%Parameters to stick it into state space
zeta = 0.001;
input_list = {'Beam Tip Force','N',[102],[0 0 1 0 0 0]};
output_list = {'End of Beam Displacement','m',[102],...
[0 0 1 0 0 0],derivative};
%Convert to State Space
SP_docs_ss= docs_modes2ss(SP_docs_model,zeta_vec,...
input_list,output_list);
%Change units from meters to micrometers
io='output';
SP_docs_ss = convert_units(SP_docs_ss,'m','mum',1000000,io);
%Get a DOCS frequency response matrix
freq = 0.1:0.1:1000;
frmat_SP = docs_frmatrix(SP_docs_ss,freq);
```

B.3 "Experimental" Data

```
% This function adds noise, and off-nominal beam
% stiffness to the meshed model to create synthetic experimental data.
%** LOAD MESHED_E SAMPLE PROBLEM SIMUALTION MODEL ****************
```

```
% This meshed_E simulation model is the same as the meshed model
% except that the beam stiffness is increased 7.14% and the spring
% constant is changed by -7%.
%----------Get the results from the .f06 file--------------------
fe_nastran('D:\SampleProblem\NASTRAN\MeshedE\sampleproblemmeshede');
load D:\SampleProblem\NASTRAN\MeshedE\sampleproblemmeshede;
phi = modesh;  %array of modeshapes
om = freq;       %modal frequencies
%Need to read in the bulk data deck
argin = {'D:\SampleProblem\NASTRAN\MeshedE\sampleproblemmeshede.dat'};
SP_bulk_data_E= read_bulk_data(argin{:});
%Stick it into a DOCS model object - contains grid and element data
zeta_vec = 0.001*ones(size(om,1),1);
SP_docs_model_E = docs_femodel(phi,om,gid,'bulk_data',...
SP_bulk_data_E,'zeta',zeta_vec);
%Parameters to stick it into state space
zeta = 0.001;
derivative = 0;      % This says its displacement
input_list = {'Beam Tip Force','N',[102],[0 0 1 0 0 0]};
output_list = {'End of Beam Displacement','m',[102],...
[0 0 1 0 0 0],derivative};
%Convert to State Space
SP_docs_ss_E= docs_modes2ss(SP_docs_model_E,zeta_vec,...
input_list,output_list);
%Change units from meters to micrometers
io='output';
SP_docs_ss_E = convert_units(SP_docs_ss_E,'m','mum',1000000,io);
%Get a DOCS frequency response matrix
freq = 0.1:0.1:1000;
frmat_SP_E = docs_frmatrix(SP_docs_ss_E,freq);
%ADD SENSOR NOISE
noise_level = 8e-2;
RandNoise = randn(1,10000);
RandNoise1 = RandNoise./3.*noise_level;
bins = 100;
Noise = docs_frmatrix(RandNoise1',freq);
frmat_SP_E_Noise = frmat_SP_E + Noise;
```

B.4 Automated Tuning

```
%** CREATE PARAMETER DEPENDENT DOCS SYSTEM **********************
%The chosen parameters here are the Young's modulus of the bar, spring
%stiffness, and modal damping
label = {'E(Pa)','K(N/m)','zeta1','zeta2','zeta3'};
ID = [1 204 ]; %entity ID number - so in this case its the mat1 ID#
groups = {};
```

```
groups{end+1} = {'mat1(E)',ID(1),label{1}};
groups{end+1} = {'celas2',ID(2),label{2}};
groups{end+1} = {'zeta',1,label{3}};
groups{end+1} = {'zeta',2,label{4}};
groups{end+1} = {'zeta',3,label{5}};
%Parameter-dependent model
SP_param = fem_uncertainty2(SP_docs_ss,SP_docs_model,groups);
%The eigenvalue sensitivity matrix
p0 = SP_param.param_nom;
[om, dom] = grad_eig(p0,SP_param,[1 2 3],SP_docs_model);
%**************** DO THE DOCS OPTIMIZATION ************
%Necessary Inputs to DOCS function update_modes
nparam = length(SP_param.paramlist);
Enom = 7.0e10;        %Nominal parameter value
Knom = 30e3;
%---------------
p0 = SP_param.param_nom;
LB = [-0.03 -0.1 -.1 -.1 -.1]';
UB =  [0.10 0.1  .1 .1 .1]';
mu = 0.08; %Noise level
W = ones(size(freq,2),1);
W(700:end) =0.01;  %de-weighting above 70Hz
W(1:100) = 0.01; %de-weighting below 10 Hz
cost = 'log'; %include phase
%===============Use DOCS function============================
[p_fit] = update_frf(frmat_SP_E_Noise,SP_param,p0,LB,UB,mu,W,cost);
%===========================================================
%New system
SP_fit = SP_param('param',p_fit);
frmat_SPfit = docs_frmatrix(SP_fit,freq);
```

Bibliography

[1] Report of Findings X-43 Mishap. Technical report, NASA, May 2003. Conducted by the X-43 Mishap Investigation Board.

[2] *MSC.Nastran Quick Reference Guide*, volume 1. Springer-Verlag Berlin and Heidelberg GmbH and Co. KG, 2004.

[3] R.J. Allemang and D.L. Brown. A Correlation Coefficient for Modal Vector Analysis. *Proceedings of the 1st International Modal Analysis Conference*, pages 110–116, 1982.

[4] Ferdinand Beer and E. Russel Johnston Jr. *Mechanics of Materials.* McGraw-Hill, Inc., 2nd edition, 1992.

[5] Carl Blaurock. Impedence Matching and Energy Shunting for Nonpositive Real Structures. Master's thesis, Massachusetts Institute of Technology, February 1994.

[6] Carl Blaurock. *DOCS User's Guide.* MIDE Technology Corporation, Medford, MA, 2003.

[7] Robert D. Blevins. *Formulas for Natural Frequency and Mode Shape.* Robert E. Krieger Publishing Co., 1979.

[8] Frederic Bourgault and D.W. Miller. Model Uncertainty and Performance Analysis for Precision Controlled Space Structures. Master's thesis, Massachusetts Institute of Technology, December 2000.

[9] B. Caesar. Updating System Matrices Using Model Test Data. International Modal Analysis Conference (IMAC), 1987.

[10] Mark E. Campbell and E.F. Crawley. *Uncertainty Modeling for Structural Control Analysis and Synthesis*. PhD thesis, Massachusetts Institute of Technology, January 1996.

[11] Edward F. Crawley and Steven Hall. Course notes: Dynamics of controlled structures, 1991.

[12] Olivier L. de Weck. Integrated Modeling and Dynamics Simulation for the Next Generation Space Telescope. Master's thesis, Massachusetts Institute of Technology, June 1999.

[13] Olivier L. de Weck. *Mulitvariable Isoperformance Methodology for Precision Opto-Mechanical Systems*. PhD thesis, Massachusetts Institute of Technology, August 2001.

[14] D. J. Ewins. *Modal Testing: Theory, Practice, and Application*. Research Studies Press Ltd., second edition, 2000.

[15] E. Fissette, C. Stavrinidis, and S. Ibrahim. Error Location and Updating of Analytical Dynamic Models Using a Force Balance Method. pages 1063–1070. International Modal Analysis Conference, 1989.

[16] C. C. Flanigan. Test/Analysis Correlation Using Design Sensitivity and Optimization. SAE, 1987. Paper 871743.

[17] Wodek Gawronski. *Dynamics and Control of Structures, A Modal Approach*. Mechanical Engineering Series. Springer-Verlag, 1998.

[18] James Gere and S. Timoshenko. *Mechanics of Materials*. PWS Publishing Co., 4th ed. edition, 1997.

[19] Roger M. Glaese and D.W. Miller. Development of Zero-Gravity Structural Control Models from Analysis and Ground Experimentation. Master's thesis, Massachusetts Institute of Technology, January 1994.

[20] J.H. Gordis. An Exact Formulation for Structural Dynamic Model Error Localization. pages 159–167. International Modal Analysis Conference, 1993.

[21] Karl Graff. *Wave Motion in Elastic Solids.* Oxford University Press, 1975.

[22] Guru P. Guruswamy. Use of High Fidelity Methods in Multidisciplinary Optimization - A Preliminary Study. *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, September 2002. AIAA-5638.

[23] Homero L. Gutierrez. *Performance Assessment and Enhancement of Precision Controlled Structures During Conceptual Design.* PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, 1999.

[24] R.L. Harder and R.N. Desmarais. Interpolation Using Surface Splines. *Journal of Aircraft*, 9:189–191, 1972.

[25] T. K. Hasselman and J. D. Chrostowski. Methods for Evaluating the Predictive Accuracy of Structural Dynamic Models. Technical Report NASA-CR-191337, NASA, December 1991.

[26] Deborah Howell and Olivier de Weck. Experimental Validation of Multidisciplinary Isoperformance Methodology. In *Proceedings of the 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Atlanta, Georgia*, 2002.

[27] Epstein K. Jackson, B. A Reconfigurable Multifunctional Architecture Approach for Next-generation Nanosatellite Design. *IEEE Aerospace Conference Proceedings*, 7, 2000.

[28] Gregory Jones. The Rock and Roll Telescope (using a gravitational pendulum to determine moment of inertia). *53nd Annual International Conference of the Society of Allied Weight Engineers, Inc*, May 1994. SAWE Paper No. 2191.

[29] NASA JPL. Space Interferometry Mission. Internet: http://sim.jpl.nasa.gov, accessed 5-4-2003 2003.

[30] R. L. Kidder. Reduction of Structural Frequency Equations. *AIAA Journal*, 11(6), 1973.

[31] N.A.J. Lieven and D. J. Ewins. Correlation of Mode Shapes, the Coordinate Modal Assurance Criterion. pages 690–698. International Modal Analysis Confernce, 1988.

[32] N.A.J. Lieven and T.P. Walters. Error Location Using Normalised Orthogonality. pages 761–764. International Modal Analysis Conference, 1994.

[33] C.S. Lin. Location of Modeling Errors Using Modal Test Data. *AIAA Journal*, 28(9), September 1990.

[34] M. Link. Identification and Correction of Errors in Analytical Models Using Test Data - Theoretical and Practical Bounds. pages 570–578. International Modal Analysis Conference, 1990.

[35] J. Lipkins and U. Vandeurzen. The Use of Smoothing Techniques for Structural Modification Applications. In *Proceedings of the 12th International Seminar on Modal Analysis*, 1987.

[36] N. Maia, M. Reynier, and P. Ladeveze. Error Localization for Updating Finite Element Models Using Frequency Response Functions. *International Modal Analysis Conference*, pages 1299–1308, 1994.

[37] N.M.M. Maia. *Theoretical and Experimental Modal Analysis.* John Wiley and Sons, 1997.

[38] J.W. Melody and G. W. Neat. Integrated Modeling Metholdology Validation Using the Micro-prescision Interferometer Testbed: Assessment of Closed-loop Performance Prediction Capability. In *Proceedings of the American Control Conference.* American Control Conference, June 1997. Albuquerque, NM.

[39] Saeed Moaveni. *Finite Element Analysis: Theory and Applications with ANSYS*. Prentice Hall, 2nd edition, January 2003.

[40] H. Natke. Updating Computational Models in the Frequency Domain Based on Measured Data: A Survey. *Probabilistic Engineering Mechanics*, 3(1):28–35, 1988.

[41] Norman S. Nise. *Control Systems Engineering*. Addison-Wesley Publishing Company, 2nd edition, 1995.

[42] Panos Y. Papalambros and Douglass J. Wilde. *Principles of Optimal Design - Modeling and Computation*. Cambridge University Press, 2nd edition, 2000.

[43] D.C. Redding and W.G. Breckenridge. Optical Modeling for Dynamics and Control Analysis. *Journal of Guidance, Control, and Analysis*, 14:1021–1032, 1991.

[44] D. Rey. Gravity and Suspension Effects on Controlled Flexible Spacecraft. Master's thesis, Massachusetts Institute of Technology, 1993.

[45] S.B Shaklan, J.W. Yu., and H.C. Briggs. Integrated Structural and Optical Modeling of the Orbiting Stellar Interferometer. volume 1945, pages 123–132. SPIE Conference on Space Astronomical Telescopes and Instruments II., April 1993.

[46] J. Sidhu and D.J. Ewins. Correlation of Finite Element and Modal Test Studies of a Practical Structure. pages 756–762. International Modal Analysis Conference, 1984.

[47] S. S. Simonian. Survey of Spacecraft Damping Measurements: Applications to Electro-Optic Jitter Problems. In *The Role of Damping in Vibration and Noise Control*, pages 287–292. ASME, September 1987.

[48] Society of Allied Weight Engineers, Inc. *SAWE Weight Engineers Handbook*, 1999.

[49] Alan E. Stockwell. A Verification Procedure for MSC/NASTRAN Finite Element models. NASA Contractor Report 4675, NASA Langley Research Center, May 1995.

[50] Gilbert Strang. *Introduction to Linear Algebra*. Wellesley-Cambridge Press, 2nd edition, 1998.

[51] Scott Uebelhart. Conditioning, Reduction, and Disturbance Analysis of Large Order Integrated Models for Space-Based Telescopes. Master's thesis, Massachusetts Institute of Technology, February 2001.

[52] Garret N. Vanderplaats. *Numerical Optimization Techniques for Engineering Design*. Vanderplaats Research and Development Inc, 3rd edition, 2001.

[53] Chris Wilson and Patty Hartzell. Accurate Spring Testing. Using Instron Machine, June 2003.

[54] Zemax Development Corporation. *ZEMAX Manual*, 2003.

[55] Q. et al. Zhang. A Complete Procedure for the Adjustment of a Mathematical Model from Identified Complex Modes. pages 1183–1190. International Modal Analysis Conference, 1987.