Integrated Modeling Tools for Precision Multidisciplinary Systems

Mark Milman and Marie Levine

Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, California 91109 USA

ABSTRACT *An integrated modeling tool that couples structural, optical, thermal and control disciplines is discussed. Details of the core modules that comprise the tool are provided, and recent applications of the tool are given. Particular emphasis is on the* modeling *of precision optomechanical systems and how IMOS has* been *used to perform multidisciplinary analysis of such systems.*

1. INTRODUCTION. This paper describes the integrated modeling tool, IMOS (Integrated Modeling of Optical Systems) that has been under development at the Jet Propulsion Laboratory over the past several years. IMOS was originally created as a modeling tool to assist in the early design phases of multidisciplinary systems. The need for such a tool became evident when a design team at JPL confronted the problem of modeling a space based interferometer [3]. This interferometer was comprised of a complex optical train with numerous collecting and actively steered optics. The optics were' mounted on a lightweight truss structure subject to dynamic, quasistatic and static disturbances. The precise tolerance required by the interferometer forced the design team to investigate many design options and trades, including disturbance isolation, vibration suppression, structural quieting and methods for minimizing interact ions between the actively controlled optics and the structure. An end to end simulation demonstrating the proof of concept was also necessary. The simulation had to integrate the structural, optical and control models into a single model to analyze and quantify the effects of mechanical disturbances on the path of starlight passing through the interferometer. This particular problem spurred the growth of IMOS and the optics modeling package MACOS [9].

Over the course of time both IMOS and MACOS have matured and have been used to analyze many optomechanical systems, (e.g. Space Infrared Telescope Facility (SIRTF), Next Generation Space Telescope (NGST)). The initial space based interferometer concept that led to the development of these tools has also matured quite nicely to become the SIM (Space Interferometry Mission) preproject. IMOS and MACOS are critical modeling tools for SIM and the accompanying ground and flight test beds used to validate SIM technologies. (See related papers [5] and [8].)

This paper presents an overview of IMOS. In addition to providing a description of the IMOS modules, a few sample problems that serve to illustrate the breadth of applications will also be sketched,

2. OVERVIEW. IMOS is a collection of functions that operates in the MATLAB [10] environment. Currently these functions perform structural modeling and analysis, thermal analysis and optical analysis (when used in conjunction with MACOS). IMOS also incorporates several graphics functions that enable viewing of strucutral assembly operations, structural deformations, and element optical layouts. The core programs are easily coupled in MATLAB, and can be extended by the user by writing his/her own MATLAB functions. Additional capabilities offered by the MATLAB toolboxes for control design, signal processing and optimization further enhance the versatility of IMOS. Several interface programs have also been developed for optical analysis (MACOS), thermal analysis (TRASYS [13] and SINDA [12]), and finite element modeling (NASTRAN [11]), IMOS has a limited internal optical analysis capability, and as an alternative to using the SINDA program, there is also an internal function for solving the heat balance equation.

With these modules, IMOS enables the user to perform the following tasks in developing an integrated model:

- Define the system geometry
- Build the structural finite element model
- Define the system optical prescription in the coordinate system of the finite element geometry
- Add control elements to the model/design
- Add mechanical and thermal disturbances to the model
- Evaluate open and closed loop performance of the integrated model using linear or nonlinear functions of the system states

Because each of these processes share common system states and design variables, they are easily iterated, allowing flexibility for design, optimization and identification.

The typical IMOS model is of the form of a mechanical system with generalized force inputs and outputs that can be used to represent a wide variety of phenomena:

$$M(\alpha)\ddot{x} + D(\alpha)\dot{x} + K(\alpha)x = Bu + d \tag{2.1a}$$

$$y = C_1(\alpha)x + C_2(\alpha)\dot{x} \tag{2.1b}$$

$$z = F_1(\alpha)x + F_2(\alpha)\dot{x} \tag{2.1c}$$

Here x denotes the vector of generalized coordinates, α is a vector of structural parameters, M, D, and K are the mass, damping and stiffness matrices, respective y; u represents a control input, d is a disturbance input, y is the vector of observed outputs, and z is the vector of controlled outputs, and B, C and F are the influence matrices for the the control inputs, measurement outputs and controlled variable outputs, respectively. Although we have written (2. lb) and (2.1c) as linear output models, this is not an IMOS constraint, but for many problems the linear formulation is appropriate.

Equation (2.1) reflects many elements of modeling in the IMOS environment. The user builds the M, D, and K matrices as a finite element model from the geometry and structural properties. These matrices define the static and dynamic behavior of the underlying structure. The other elements of the analysis depend fundamentally on these structural elements because they define the generalized coordinates from which all the quantities of interest are defined. The interdisciplinary nature of IMOS comes from the role of the influence matrices B, C and D. The B matrix specifies how control signals are injected into the system. The C matrix defines what signals from the system are observed. And the F matrix defines the combination of st ate variables that form the variables to be controlled; or the variables by which performance is measured. For example, using IMOS functions, F can define a perturbation of a ray bundle through an optical system, After specifying a feedback mechanism to define u as a function of y (which can be designed in MATLAB), (2. 1) then represents a controlled optomechanical system. The system description is completed by adding dynamic and static disturbances that represent gravity or thermal loads, for example.

Equations (2. 1a)–(2.1c) embody the end–to–end nature of the IMOS model; the modeling and analysis of the (open or closed loop) system from the input disturbance d to the the controlled output variable z. The power of IMOS is derived from the fact that this model resides entirely within MATLAB, This enables the manipulation of all of the variables building on MATLAB functions to, design, model, analyze, optimize and identify these systems.

3. **IMOS MODULES.** IMOS is comprised of core modules written in the MATLAB environment, and of several interfaces to discipline specific programs. The core programs are written as either MATLAB .m files or as MEX interface files, The latter is especially useful for functions that have nested do–loops. Figures 3.1 and 3,2 illustrate the general IMOS architecture. Figure 3.1 represents the core modules of IMOS and their present/near term capabilities,

Among the emerging capabilities of IMOS is the use of parallel computing. A parallel version of the standard Bode analysis function and the QR algorithm have been implemented. These two algorithms were targeted because they are frequently used and are both extremely time consuming on large problems, Theyt have been implemented within the IMOS/MATLAB environment to run on parallel machines via a PVM interface.

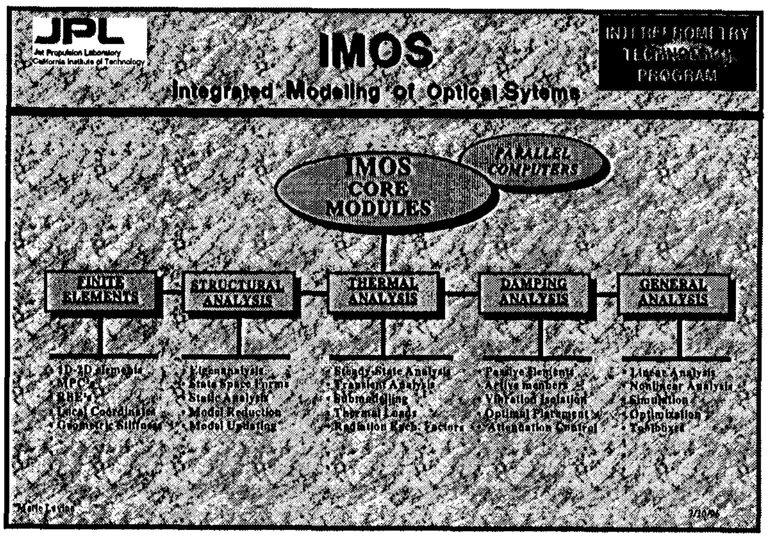


Figure **3.1.** IMOS CORE MODULES

Figure 3.2 represents the interfaces between IMOS and discipline specific programs.

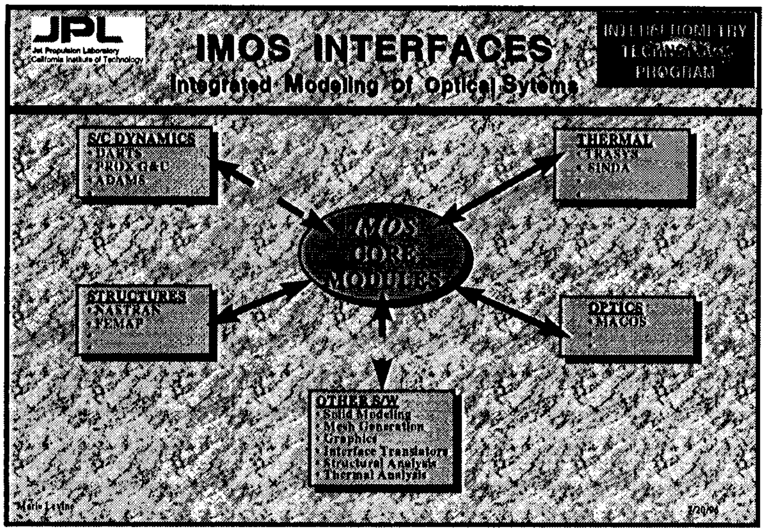


Figure 3.2. IMOS INTERFACES

We will briefly outline the structural, optical and thermal analysis IMOS modules and interfaces. Some rudimentary uses of these modules will also be given.

3.1. Structural Module. The IMOS structural analysis module provides a capability for finite element modeling and analysis of structural statics and dynamics. The ingredients of the IMOS finite element model include the location of the the element nodes, the number of degrees of freedom per node (this is a function of the type of element and constraints), the boundary conditions, the type of elements that connect the nodes, and the material and mass properties of these elements,

Finite elements are based on using low order piecewise polynomial functions to approximate the continuum structure, Interpolating the values of these polynomial functions between the nodes defines the deflection of the entire structure, The output of the finite element model is the system mass and stiffness matrices. These are the M and K matrices in (2,1), and each degree of freedom represents a generalized coordinate corresponding to a translation or rotation at a node of the model, Damping models are usually developed (or perhaps guessed at) after further analysis of the structure, but certainly if the user has one, it can be included at this time,

Having the M and K matrices, even in the preliminary design phase, is very useful, It gives subst ante to the design process, Eigenanalysis provides insight into the fundamental dynamics of the structure. And the stiffness matrix K can be used to carry out preliminary static analysis, Deformations of the structure produced by thermal effects can also be analyzed using the elements in the structural module. Temperature distributions are converted to thermal load vectors which are in turn “inverted” through the stiffness matrix to calculate the deformations,

The IMOS structural elements include rods, beams (with axial, torsional, and two transverse bending stiffnesses), COSMIC NASTRAN plate elementst, as well as rigid body elements, multipoint constraints, concentrated masses (with and without inertia), springs. There is a variety of standard model reduction and conversion routines, and IMOS also supports the use of local coordinates. The structural analysis portion of IMOS allows the user to build relatively elaborate spacecraft models with its library of elements. Using sparse matrices supported by MATLAB, models as large as 7300 dof have been successfully built and analyzed using IMOS. For these large problems, eigenanalysis is done by an implicity restarted Arnoldi scheme [4] that computes selected eigen–pairs, and is substantially more efficient than the resident MATLAB eigensolver. (The QZ algorithm that is used in MATLAB computes all of the eigenpairs, and takes no advantage of the sparsity of finite element systems, Just loading the matrices for these problems presents serious memory difficulties since they are not stored as sparse systems.)

Interfaces with FEMAP and NASTRAN have been developed, This enables the user to transport and convert models developed with other finite element programs into IMOS. FEMAP is also useful as a preprocessor to prepare IMOS files in addition to its translation role between IMOS and other finite element programs, The 7300 dof NGST alluded to above was originally developed in NASTRAN and then ported into IMOS to perform interdisciplinary analysis using these interface functions.

3.2 **Optics Modeling.** IMOS optics modeling is designed to be used in conjunction with the IMOS strucutral model. This allows for great flexibility in the design and analysis of controlled opt ical systems that are at t ached to structures, Many aspects of optomechanical systems can be understood by characterizing the linear sensitivities of motion of the optical elements. For these applications linear optical models can be developed and then analyzed using powerful state space methods.

The defining feature of the IMOS optomechanical model is that the generalized coordinates that describe the structural model also describe the position and orientation of the optical elements, i.e., the optical prescription is defined in the same coordinate system as the finite element model, In addition to the coordinates, the optical prescription must also contain the descriptions, called element types in the IMOS optics module, of the individual surfaces that form the optical train. Although IMOS has the capability for defining optical prescriptions, the element types are somewhat limited, and the MACOS program is generally used with IMOS for obtaining optical prescriptions and conducting physical optics analysis,

Once the finite element model and optical model have been married, we can characterize the effects of mechanical motions on the optomechanical system. For many applications it is sufficient to trace rays through the optical train and see where they pierce the focal plane, The locus of such points is called the spot diagram, For example a point source on the optical axis located at infinity will produce a spot very tightly centered around the focal point. If the source is shifted slightly off center, again a small spot will be produced, but this time it will shift proportionally to how far off axis the source has moved,

When aberrations of the optical system are large their effects are captured by geometric ray tracing, However when the aberrations produce small changes, on the scale of a wavelength of the light, geometric optics and ray tracing no longer adequately capture the behavior of light. In these situations it is necessary to model the wave behavior of light. Geometric optics models the distribution of light for an unaberrated optical system as a delta function at the focus. A more accurate distribution of the light is contained in the figure below where the effects due to diffraction have been modeled, The energy is actually spread out over an infinite extent with approximately 84% contained in the main lobe.

The distribution [illegible] tion (PSF) of the system, It is very an [illegible] ne diameter of the aperture increases, t [illegible] tric optics,

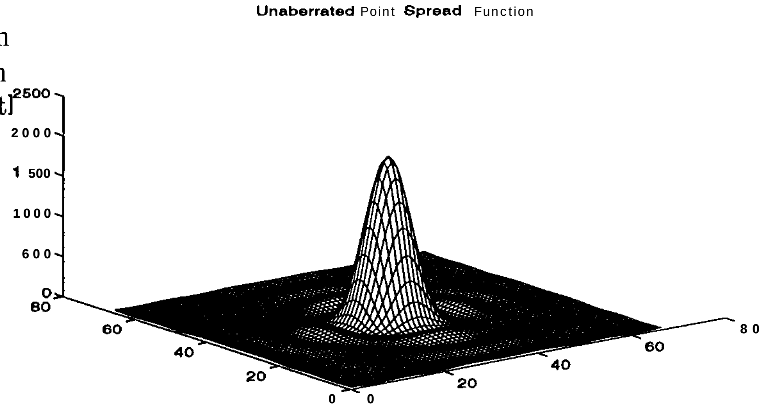
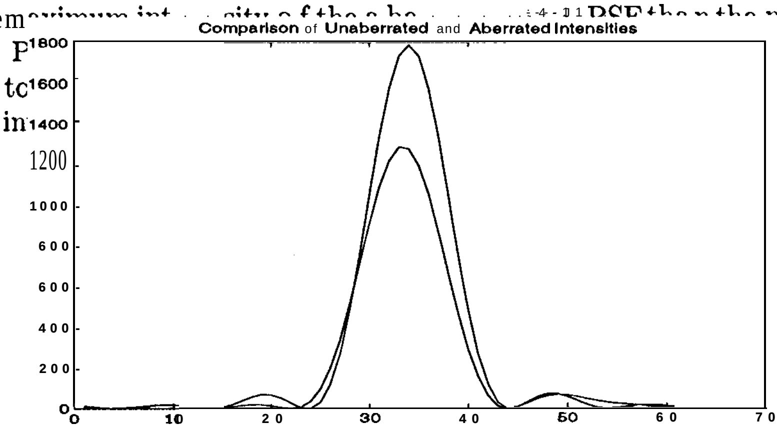


Figure 3.3 Point Spread Function

The diffraction effects in IMOS are typically modeled with a combination of ray trace and scalar diffraction theory, Rays are traced through the optical train onto a spherical reference surface with center located at the focal point. If there are no deformations in the optical train, the optical pathlength of each ray to the reference surface is the same. In this case the converging wavefront and the reference surface coincide, or equivalently the light has uniform phase over the reference surface, When there are perturbations of the optical elements the optical pathlength of the rays change with respect to one another, producing a nonuniform phase over the reference surface.

In general the [illegible] maximum intensity of the unaberrated P [illegible] atio, and is always less than or equal t [illegible] imaging systems, and fits quite well in [illegible]



Fig, 3,4 Comparison of PSF Profiles for an Aberrated and Unaberrated Optical System

3.3, **Thermal Module.** The IMOS thermal module provides a set of software tools for calculating temperatures and heat flows using a previously created IMOS structural model of the system. This is completely analogous with how the optics model is created, The IMOS functions allow the user to create input files for the TRASYS and SINDA programs, as well as using the internal IMOS heat balance equation solve as an alternative to SINDA. An important feature of the IMOS generated thermal model is that the thermal nodes are completely compatible with the structural nodes, Thus there is no need for interpolating temperature values to conduct structural analysis, and consequently the transition from thermal to structural analysis is quite seamlesss.

The discretization method for modeling thermal systems in IMOS (when either using SINDA or the internal solver) is based on the thermal network approach. The network approach is derived from the energy balance equations and is equivalent to a particular finite difference discretization of the underlying heat transfer equation. The thermal network is defined by a set of nodes and conduct antes, and is analogous to an electrical network, Thus there is a correspondence between potential, flow, resistance, capacitance, etc., between these networks, and basic laws such as Ohm's Law and Kirchoff's Laws can then be applied to balancing the network.

Each thermal node represents an isothermal component of the system. The temperature of the node is the average temperature over the subvolume it encloses. In addition to temperature, a node also has a capacitance which represents the thermal mass of the node that dictates how quickly the node can change temperature. The capacitance is computed from the properties of the material comprising the subvolume attached to the node, and arises in the analysis of the transient characteristics of the system. The nodes for the steady–st ate problem can be considered to have a zero time constant, and are sometimes referred to as arithmetic nodes,

The conductors describe the energy transport between nodes. They have the form of either a conduct ion conductor, a convection conductor, or a radiation conductor. The underlying physical mechanisms for each oft hese modes of energy transfer is different. Within the context of the resulting mathematical model, conduct ion and convection conductors transfer energy as the difference between the temperatures between connected nodes, while the radiator conductors are nonlinear and involve the difference in the fourth power of the nodal temperatures, Conductors normally have the same value in both directions between nodes; however, one–way nodes are often used to model fluid flow,

The resulting transient equations have the form

$$\bar{C}_i \frac{dT_i}{dt} = Q_i + \sum_{j=1}^{N} C_{ij}(T_j - T_i) + \sum_{j=1}^{N} R_{ij}(T_j^4 - T_i^4); \quad i = 1, ..., N \tag{3.1}$$

Here T_i denotes the temperature at node $i, \bar{C}_i$ is the heat capacitance of node i, C_{ij} are the conduction coefficients, R_{ij} are the radiation coefficients, and the Q_i are heat sources. The associated steady state equation we study in this note is

$$Q_i + \sum_{j=1}^{N} C_{ij}(T_j - T_i) + \sum_{j=1}^{N} R_{ij}(T_j^4 - T_i^4) = 0; \; i = 1, ..., N \tag{3.2}$$

The conduction coefficients are computed as a function of the thermal conductivity of the material, the cross–sectional area through which the heat flows, and the length between nodes. For a rectangular discretization Fourier's Law yields

$$C_{ij} = \frac{kA}{L} \tag{3.3}$$

where k denotes the thermal conductivity of the material, A is the cross–sectional area, and L is the length between nodes i and j. IMOS computes these conduction coefficients using the structural model and geometry. These coefficients are either prepared as an input file for use in SINDA, or they can be used directly with the internal solver.

For convection conductors,

$$C_{ij} = hA, \tag{3.4}$$

where h **is the** thermal convective conductance and A **is** the nodal surface area in contact with the fluid. The thermal conductivity is in general temperature dependent, so that the C matrix above is also, The radiation interchange matrix R is a function of the the surface geometries, the orientation of the elements of the system with respect to one another (viewfactors), their radiative properties, and temperature. TRASYS is used to calculate the radiation conductor values. TRASYS requires a specification of the system geometry, surface optical properties and orbit/orientation with respect to the sun and or planets. IMOS has the capability of generating a TRASYS model from the finite element geometry.

Typically, $C_{ij} = C_{ji}$, and $R_{ij} = R_{ji}$, although when modeling fluid flow, the use of one-way nodes causes asymmetry in the matrix C. When temperatures are not expected to deviate greatly from a nominal value, the assumption that the coefficients of C and R are temperature independent, i.e. constant, is often made. At cryogenic temperatures, where small variations in temperature can lead to significant changes in the thermal conductivity of materials this is not a valid assumption.

Solution techniques resident in SINDA exploit the network structure of the equations. A nonlinear Gauss–Seidel iteration in conjunction with acceleration techniques is used. The internal IMOS steady state solver uses a Newton method with steplength control. This algorithm has been shown to be globally convergent,

4. **APPLICATIONS). Much of the control/dynamic analysis of optomechanical systems that has been performed using** IMOS relies on placing the model into the form of equations (2.1). We will describe this procedure for interferometric and filled-aperture systems with emphasis on the generation of the optical output models.

The optical elements of the systems of interest are defined with respect to nodes that exist in the finite element model. For example, a nondeforming reflective or refractive surface is defined by a location (vertex point), directionality of the element (the principal axis), focal length and eccentricity, and index of refraction. The vertex point is either a structural node or slaved to a structural node via a rigid body element. Any motion of a node causes a translation/rotation of the optical element. The opt ical train of the system is comprised of a concatenation of such elements. Consider one ray traced through the system, and and the resulting effect caused by a small change in a single degree of freedom of one node that supports the optical train. Fix a reference surface of interest at which to keep track of the pathlength of this ray as it traverses all of the optical elements in its path, The differential change in the optical pathlength of the ray is computed in MACOS as a function of the perturbed degree of freedom. This can be done either analytically or numerically. A row vector can be built up by perturbing other degrees of freedom. And finally an entire matrix can be constructed by following the paths of many rays. An important function of MACOS in developing IMOS output models is to compute this matrix. Thus MACOS provides a matrix, C such that

$$y = Cx \tag{4.1},$$

where y is a vector of pathlength errors as a function of the structural deformation x. For precision optical systems that require tight tolerances on the mechanical deformations, the differential pathlength approximation in (4.1) is an excellent approximation to the actual pathlength change.

The C matrix in (4.1) is the departure point for linear output models for interferometric and filled aperture systems.

The objective in interferometry is to combine (i.e., interfere) two beams of light with the same phase, Each beam traverses a different path through the interferometer, The SIM interferometer uses actively controlled opt ics to change the pathlength of one of the beams as a means for adjusting its phase, The main output of interest is the difference in the mean pathlength for each beam, The goal is to regulate this quantity to a small fraction of a wavelength (approximately to $\lambda/20$). In addition to controlling the mean pathlength, it is also important to control the variation of the pathlength of the rays about the mean pathlength. The first order moment of this quantity is the "tilt" of the wavefront. The OPD (optical pathlength delay – another name for the mean pathlength difference) and wavefront tilt are the two fundamental quantities that must be controlled for the operation of the interferometer, Both of these output variables can be derived from the C matrix in (4.1). (We refer the interested reader to [6].)

For filled aperture systems, e.g. telescopes, a commonly used figure of performance is the Strehl ratio. We will next develop how the C matrix enters into defining this quantity in the context of (2.1). Let x denote the generalized coordinates in (2. la), and let $h(x)$ denote the intensity at focus of the optical system as a function of the deformation x. We assume that the system is unaberrated when $x = 0$. The Strehl ratio, denoted as SR is by definition

$$SR = \frac{h(x)}{h(0)} \tag{4.2}$$

Expand h **to** second order to obtain

$$h(x) = h(0) + \nabla h(0) \cdot x + \frac{1}{2} < Hx, x > + o(|x|^2) \tag{4.3},$$

where H is the hessian and o(u) is the standard mathematical notation: $o(u)/u \rightarrow 0$ as $u \rightarrow 0$. Since h is optimized at x = O, $\nabla h(0)$ = O, and consequently

$$h(x) = h(0) + \frac{1}{2} < Hx, x > + o(|x|^2) \tag{4.4}$$

Hence,

$$SR = 1 + \frac{< Hx, x >}{2h(0)} \tag{4.5},$$

and optimizing to second order the Strehl ratio is equivalent to minimizing the quadratic form $< -Hx, x >$. **Since** $-H$ **is** nonnegative definite it has a nonnegative square root, $(-H)^{1/2}$. Thus defining the output variable z,

$$z = (-H)^{1/2} x, \tag{4.6}$$

leads to an output model that approximates the Strehl ratio. Standard state space analysis (e.g., bode plots, Lyapunov analysis) can then be used to characterize the Strehl ratio of the system as a function of input disturbances and perturbations of the mechanical states. Moreover, it can be shown that $(-H)^{1/2}$ can be approximated by the C matrix in (4.1) [7]. The study in [7] considered the question of selecting positions for damping mechanisms in the backup truss structure of a space based segmented reflector telescope to optimize the Strehl ratio. The damping mechanisms were modeled as velocity feedback loops on the displacements of the ends of struts in the truss. Changing their location is affected by appropriately modifying the B matrix in (2. la). Using the definition of z in (4.6), the objective functional is simply

$$J = E(|z|^2), \tag{4.7}$$

where E denotes the expectation operator, J is computed in the standard way as the solution to a Lyapunov equation involving the dynamics (2. la) and output equation (4,6). Optimizing the damper locations with respect to J was done using a simulated annealing strategy. The ability to directly optimize the merit function of interest is a tremendous advantage. In this application the locations are selected to essentially damp the modes that degrade optical performance the most.

Although this discussion has focused on IMOS'S ability to model optomechanical systems, there are many other applications for which the coupling of MATLAB, structural, optical and thermal models offers a powerful combination of tools, techniques, and analysis possibilities that would otherwise be unpractical or impossible.

Acknowledgements

This research was performed at the Jet Propulsion Laboratory, California Institute of Technology, under contract from the National Aeronautics and Space Administration.

References

[1] M. Born and E. Wolf, *Principles of Optics,* Pergamon Press, Oxford, 1989.

[2] J. W. Goodman *Introduction to Fourier Optics*, McGraw-Hill, New York, 1968.

[3] R. A. Laskin and M. San Martin, Control/Structure system design of a spaceborne optical interferometer, AAS/AIAA Astrodynamics Specialist Conference, Stowe, Vermont, August, 1989.

[4] R. B. Lehoucq, D. C. Sorensen, P. Vu, and C. Yang, *ARPACK*: An implementation of the implicitly restarted Arnoldi iteration that computes some of the eigenvalues and eigenvectors of a large sparse matrix, 1995, Available from ftp.caam.rice.edu under the directory pub/software/ARPA *CK*

[5] M. B. Levine, J. W. Melody, and M. Milman, Experimental verification of integrated opto–structural model predictions, Space Microdynamics and Accurate Control Symposium, Tolouse, France, May, 1997.

[6] J. W. Melody and G. W. Neat, Integrated modeling methodology validation using the Micro-Precision Interferometer Testbed, Conference *on* Control and Decision, Kobe, Japan, December, 1996.

[7] M. Milman and L. Needels, Modeling and optimization of a segmented reflector telescope, SPIE 1917, Albuquerque, New Mexico, 1992

[8] G. W. Neat and J. W. Melody, Control technology readiness for future spaceborne interferometer missions, Space Microdynamics and Accurate Control Symposium, Tolouse, France, May, 1997.

[9] D. Redding and W. Breckenridge, Optical modeling for dynamics and control analysis, J. Guidance, Control, and Dynamics, 14, September 1991.

[10] MATLAB, The MathWorks, Inc.

[11] NASTRAN, NASTRAN User's Manual, NASA SP-222.

[12] SINDA, Systems Improved Numerical Differencing Analyzer and fluid Integrato (SINDA) Version 2.3", prepared by Martin Marietta Corporation Denver Aerospace, March, 1990.

[13] TRASYS, Thermal Radiation Analyzer System (TRASYS) User's Manual", prepared by Lockheed Engineering and Management Services Companuy, April, 1988.