An Object Oriented Framework for the Synthesis of Robust Multi-rate Controllers for Hard Disk Drives (HDD)

Leonardo Contreras, Richard Conway, Sarah Felix, Jongeun Choi and Roberto Horowitz

Computer Mechanics laboratory
Department of Mechanical Engineering
University of California
Berkeley, CA 94720-1740

ABSTRACT

This paper proposes an object oriented framework to model and synthesize robust multi-rate controllers for many of the track-following servo configurations encountered in the hard disk drive (HDD) industry, including traditional single-stage and new dual-stage servos. The proposed object oriented software will be used to synthesize and evaluate the tracking performance of different single-stage and dual-stage servo designs, given realistic Track-Mis-Registration (TMR) budgets.

Keywords: Object Oriented Design, Hard Disk Drive, Robust Control

1. INTRODUCTION

In this object oriented approach to control design, the goal was to program a user-friendly, adaptable software package to handle a broad range of HDD servo problems. The algorithms used to construct the plant model must be as flexible and generic as possible to handle various types of actuation and sensing schemes. At the same time, the plant model must have a particular structure to be used in multi-input-multi-output (MIMO) robust control design techniques. Once a mathematical model of the HDD dual-stage servo system is defined, a robust track-following problem can be formulated as a time-varying robust H_2 synthesis problem. Then optimal robust controller design techniques can be used to solve the H_2 problem, depending on the assumptions made about plant uncertainties.

An operational prototype GUI has been developed, and programs have been written to construct generalized plant models and facilitate control design iterations. A preliminary study has been conducted to validate use of the interface for MIMO controller design in a MEMS-actuated dual stage HDD. Section 2 describes how the hard disk drive system is modeled. Section 3 presents the object oriented design framework. Section 4 describes the robust control design methods used. Finally, Section 5 includes a preliminary GUI model example.

2. HARD DISK DRIVE SERVO SYSTEM MODEL

In order to characterize the scope of possible systems to be modeled, we anticipated various hard disk drive plant configurations that may be encountered. These configurations are based on actual dual stage designs and models that have been developed and tested in academic laboratories and in industry. They include different actuation schemes, sensing schemes, and disturbance models.

Dual Stage Model Configurations

A key feature of the software framework is the ability to construct plant models of several possible hard disk drive configurations. Possible actuation schemes include: Translational MEMS actuated slider [1], rotational MEMS actuated slider [2], PZT actuated suspension [3] and PZT actuated slider [4]. These configurations are summarized in Figure 1.

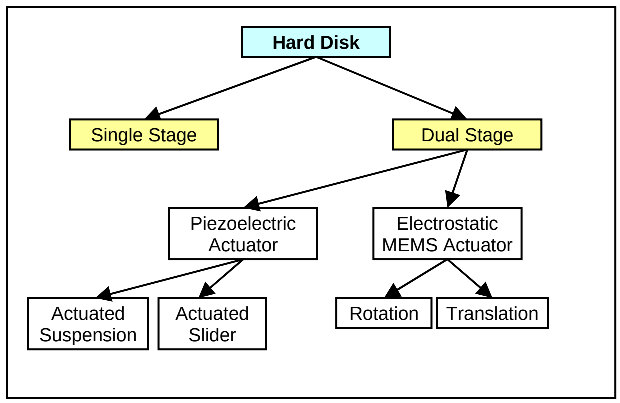

Figure 1: Configurations supported by control design interface

A hard disk drive system with a MEMS-actuated slider includes an electrostatic microactuator installed between the suspension gimbal and the magnetic read-write head, or slider. The actuator serves to displace the slider relative to the suspension for fine tune positioning of the magnetic head. The actuator may provide translational or rotational motion. A schematic is shown in Figure 2a. Another kind of hard disk drive system uses PZT actuation. This configuration uses piezoelectric strips at a location along the suspension arm assembly that, when actuated, expand or contract to cause a yaw displacement in the structure.

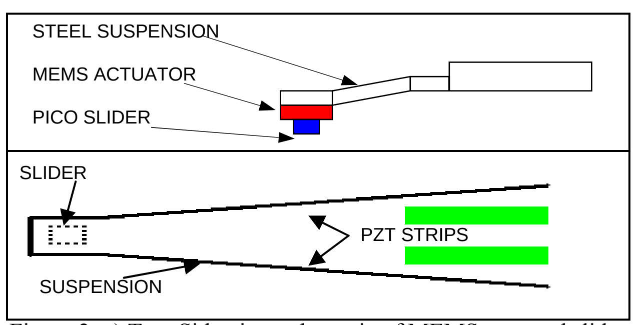

Figure 2: a) Top, Side view schematic of MEMS-actuated slider
b) Bottom, Top view schematic of PZT actuation

The PZT strips may be located near the suspension hinge in the case of a PZT-actuated suspension, or at the tip, in the case of a PZT-actuated slider. A schematic is shown in Figure 2b.

Several sensing schemes have been developed to enhance the sampling rate and amount of information contained in displacement signals used for feedback. A typical single-stage disk drive system relies on position error information from servo sectors on the disk. In dual stage actuation systems described above, it is necessary to have additional sensor information for relative position sensing and higher rate signals. Position sensing of the slider relative to the tip may be extracted from onboard sensing in the actuator. In addition, piezoelectric strain gages fabricated directly onto the suspension can sense relevant vibrations and are independent of the actuators [5].

Different assumptions and models may be used to describe the disturbances to the system. All sensors will have some sensor noise, which may be assumed white or filtered. In a disk drive system, structural components of the E-block/suspension assembly will be subject to airflow disturbances, or windage. Again, this may be assumed to be white, or may have a more realistic spectral model. Finally, the track runout model is important for simulating realistic track mis-registration. Companies and institutions often have proprietary models of windage and runout, so these components should be a generic as possible in the plant construction algorithms.

Generalized Model Structure and Assumptions

The goal for the plant modeling algorithms was to develop a generic model structure that can handle all the possible configurations mentioned above, while avoiding excessive complexity. Considering the model structure and assumptions, we constructed a theoretical global model that captures all possible components, represented by the block diagram in Figure 3.

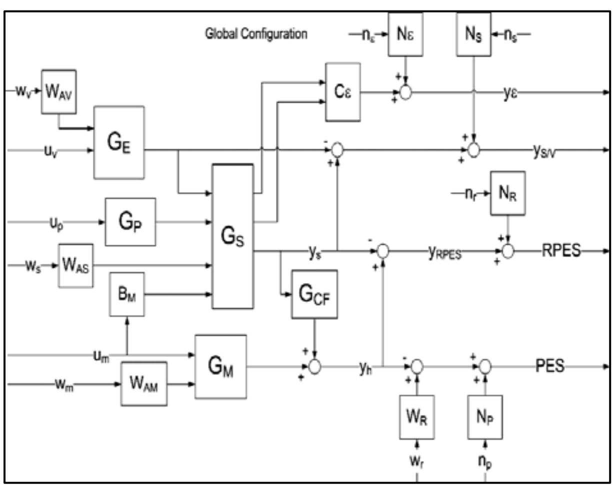

Figure 3: Block diagram representing generalized configuration

While such a system will likely not exist, this global model was used as a framework for constructing models of the four actuator configurations discussed. Thus we ensure that all the plant models have a common format.

Some simplifying assumptions were made based on common modeling and design assumptions. We will describe the assumptions while pointing out the features of the block diagram. In the notation, an 'x' represents an unspecified subscript. First, the E-block (G_E), suspension (G_S), and slider actuator (G_M) components are each modeled as vibrational systems with uncoupled second order modes. The inputs of each system include a control input (u_X), a windage input (w_X), and inputs from interaction with other components. It was assumed that all parameter uncertainties of interest would be in one or all of these modal systems. This is reasonable, as the assembly and fabrication of the E-block, suspension, and MEMS microactuator are subject to the most significant manufacturing variations. The PZT driver for an actuated suspension (G_P) is assumed to have first order dynamics and has control input u_P. Sensor dynamics were neglected so that sensor ouputs are algebraic relationships between the measured signals. Possible measurement outputs include the position error signal from the servo sectors (PES), relative position between the slider and suspension (RPES), relative position between the suspension and the VCM ($y_{S/V}$), and the PZT strain gage measurements (y_ε). Input disturbances (w_X and n_X) are represented by white noise inputs going through filters or spectral transfer functions (W_X and N_X).

Dynamic coupling between structural components differs significantly among the configurations described above. Therefore, the coupling dynamics are represented by separate components (G_{CF} and B_M) in the system model, with their inputs and outputs flowing between the interacting components. This modular approach to coupling allows us to construct a global model that encompasses all possible configurations. Figure 4 shows the block diagrams of the two MEMS actuator configurations. Figure 5 shows the block diagrams of the two PZT actuator configurations.

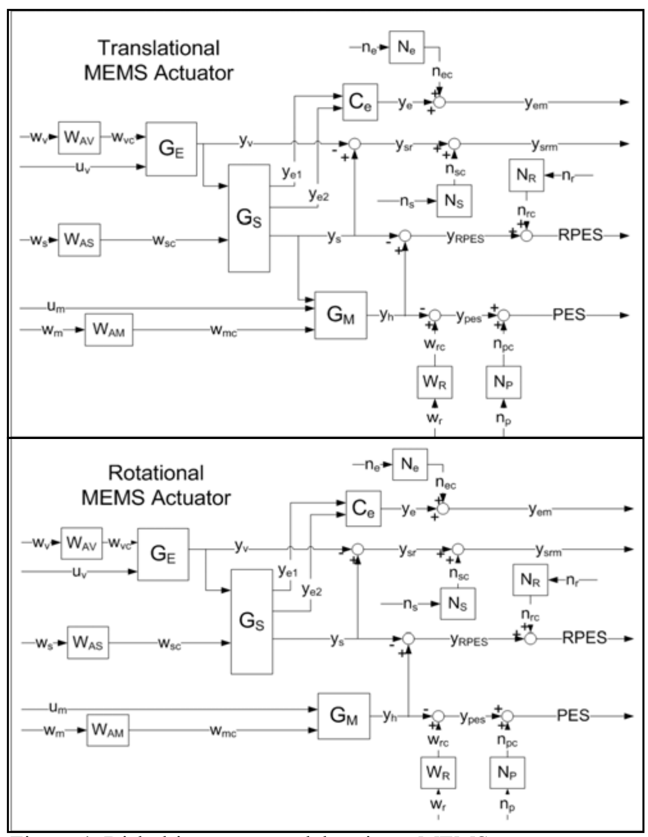

Figure 4: Disk drive servo models using a MEMS actuator

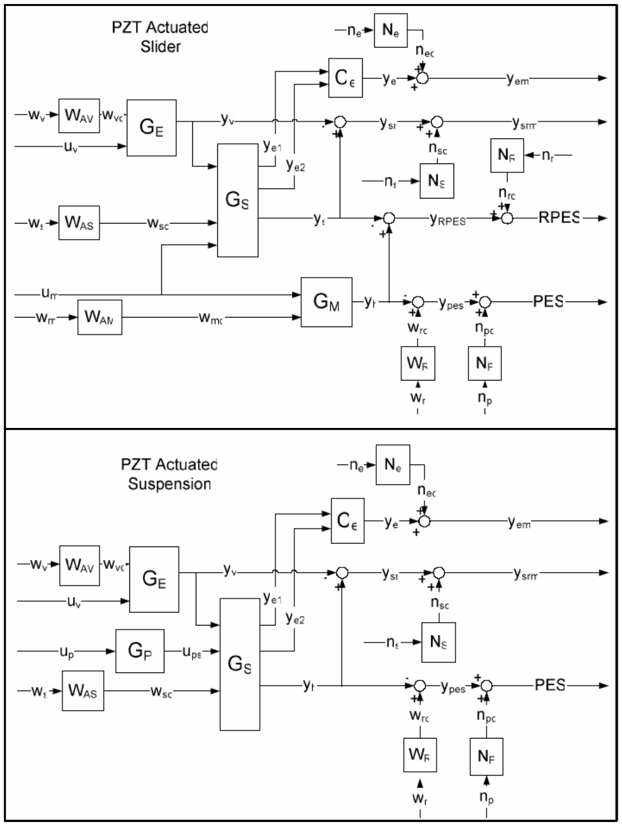

Figure 5: Disk drive servo models using PZT actuators

Linear Fractional Transformation (LFT)
The final model realization will be in the form of an LFT, shown in Figure 6. This model realization is frequently used in robust control design. The signals w_Δ and z_Δ are inputs and outputs describing the perturbation of the uncertain parameters in G_E, G_S, and G_M. The signal w_2 captures the disturbance inputs including sensor noise, windage and runout. z_2 represents the performance outputs to be minimized and can be selected by the user.

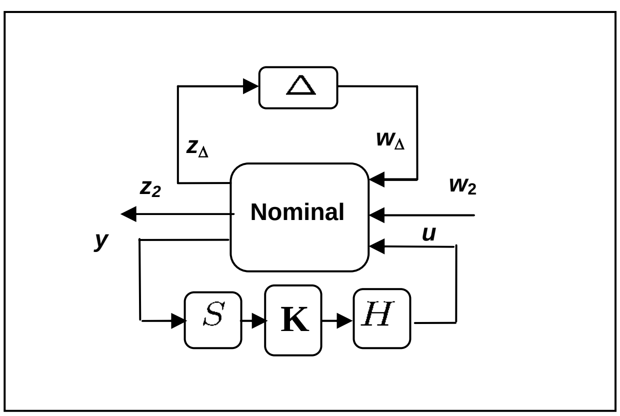

Figure 6: Linear fractional transformation (LFT) block diagram

The signals *u* and *y* are the control inputs and measurement outputs respectively and can also be selected by the user from among the inputs and measurements described above. The Δ block represents the normalized uncertainties, the K block is the controller, and 'S' and 'H' represent a sample and hole, respectively.

3. OBJECT ORIENTED MODEL FRAMEWORK

Object-oriented programming techniques are used to develop flexible and user-friendly interfaces. The structural pattern that the framework characterizes is carried out with the unified modeling language (UML) [6]. In the pattern shown in Figure 7, the four configurations of HDDs correspond to different classes in this methodology. The framework is based on a multi-layered abstraction of model configurations represented using several package applications.

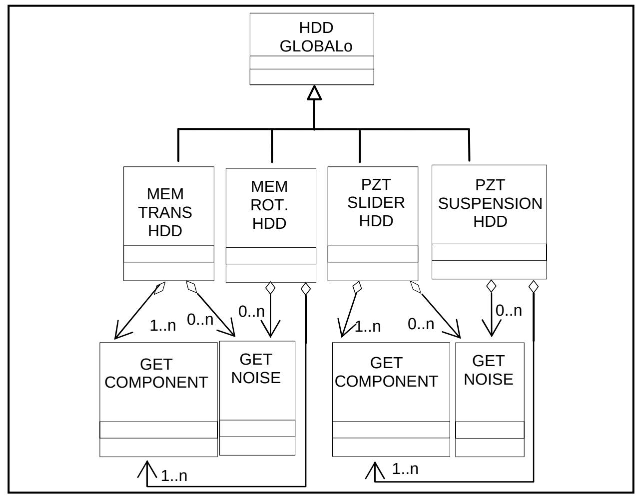

Figure 7: Framework HDDs Object Oriented Diagram Class

In particular, a special class of the type Association is in charge of registering the different components and noise models for each of the four HDD configurations.

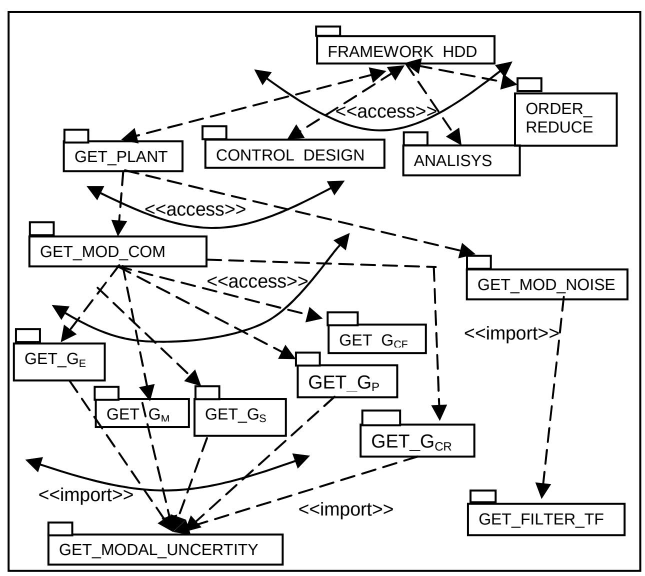

Figure 8: Framework HDDs Object Oriented Diagram Packages

The Object Oriented Diagram Packages of the HDD framework are shown in Figure 8. This diagram illustrates the processes for plant modeling, control design, analysis, and order reduction. The diagram shows the functionality of the different subsystems that compose the application. For space reasons, some of the details of the get_plant package have been omitted, but each one of the classes should define its properties and operations, to support the behavior pattern of HDD framework.

4. ROBUST CONTROL DESIGN

We consider a discrete-time linear time-invariant generalized plant with an uncertainty block, illustrated in Figure 6:

$$\begin{bmatrix} z_\Delta \\ z_2 \\ y \end{bmatrix} = \begin{bmatrix} A & B_\Delta & B_2 & B_u \\ C_\Delta & 0 & D_{\Delta 2} & D_{\Delta u} \\ C_2 & D_{2\Delta} & D_{22} & D_{2u} \\ C_y & 0 & D_{y2} & 0 \end{bmatrix} \cdot \begin{bmatrix} w_\Delta \\ w_2 \\ u \end{bmatrix}$$

$$w_\Delta = z_\Delta \,, \qquad (1)$$

where we have used the standard state space notation:

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] := D + C(zI - A)^{-1}B. \qquad (2)$$

Let us define a set B_p of parametric uncertainties as

$$B_p = \{\Delta := diag[\delta_{1,\dots,}\delta p], \delta_j \in B\Re,\, j = 1,\dots, p\}, \qquad (3)$$

where $B\Re := \{r \in \Re, |r| \le 1\}$ We are interested in robust H_2 performance in the presence parametric uncertainties. Our problem is formulated as follows:

Problem: For given multi-rate sampler S and hold *H* with fixed sampling and hold rates, design a controller *K* that stabilizes the closed-loop system for all $\Delta \in$ Bp, and minimizes the worst -case RMS value of z_2 against Gaussian white noise w_2 for all $\Delta \in$ Bp, or equivalently, solve an optimization problem:

$$\min_{K \in \kappa(B_p)} \max_{\Delta \in B_p} \|T_{z2w2}(HKS,\Delta)\|_2 \qquad (4)$$

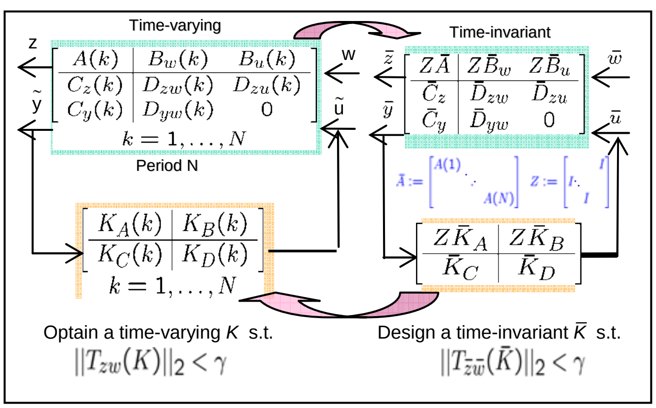

Figure 9: Flow-chart for designing multi-rate controllers

It is important note that the structure in (1) is quite common in most of the track-following problems in HDDs. It guarantees that the closed loop system matrices depend on Δ affinely and ensures the well-posedness of the closed-loop system..To cope with the multirate nature of the controller, we used the procedure in Section III in [7]. The procedure is summarized as follows:

1) Derive a linear time invariant (LTI) systems from the periodically time-varying system.
2) Design a controller for the LTI system.
3) Recover a periodic time-varying controller by decomposing the LTI controller matrices.

For more details, see [7] and references therein. This procedure is illustrated in Figure. 9. In step 2), the coordinate descent method is used [7] to handle coupling between the controller variable, Θ, and the parameter for a Lyapunov function, P. By fixing Θ, the optimization problem becomes convex in P, and viceversa. The descent algorithm iterates between these two convex optimization problems until a robust controller with a reasonable worst-case H_2 performance cost is achieved.

In nonconvex problems, the selection of an initial point is essential. Reasonable initial controllers are obtained by the procedure given in [8]. For future work, we plan to investigate alternative Bilinear Matrix Inequality (BMI) optimization solvers, which are not based on this dual iterative approach.

5. PROTOTYPE INTERFACE AND MODEL EXAMPLE

The prototype interface consists of a plant modeling interface, a control design specification interface, and their parent interface, shown in Figure 10.

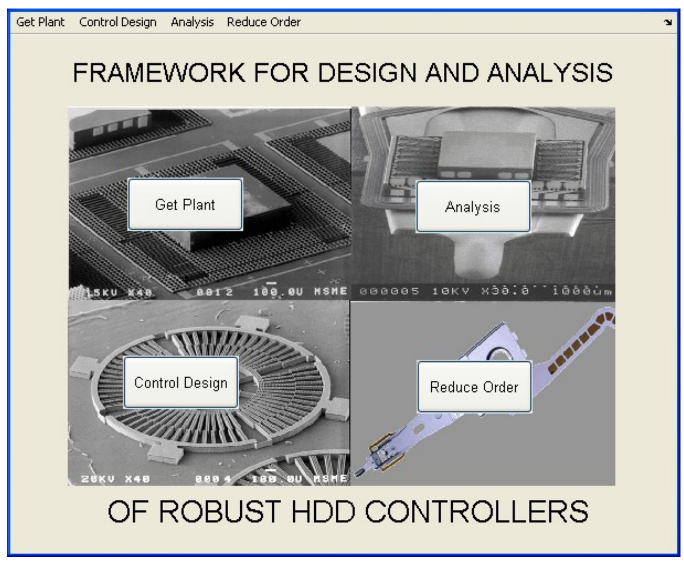

Figure 10: Parent GUI for mediation between the control design specification and plant modeling interfaces

The 'Get Plant' menu allows the user to select the form of the plant model. After the user has selected the form of the plant model, the plant modeling interface will open. The 'Control Design' menu in the parent interface will open the control design specification interface.

Plant Modeling Interface

Each plant configuration has its own modeling interface, such as the one shown in Figure 11. In the block diagram which constitutes each modeling interface, each block is a pushbutton which opens a GUI for specifying the continuous time model corresponding to that block.

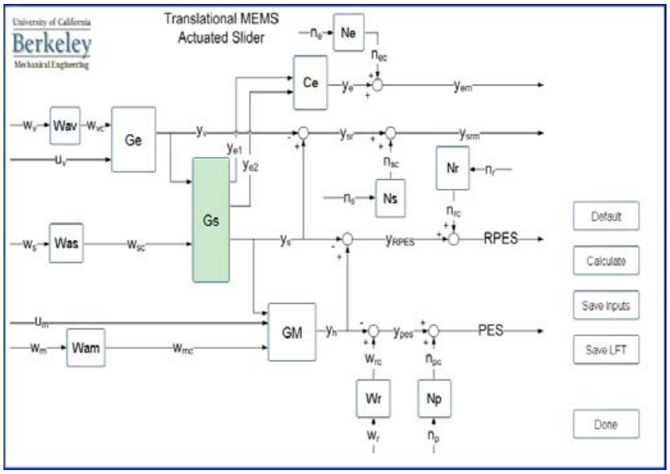

Figure 11. Plant modeling interface for servo with translational MEMS actuated slider

When the model for a particular block has been specified, the background color of that block darkens for ease in determining which models have not yet been specified. The user can also choose to set all of the model parameters to a set of default values.

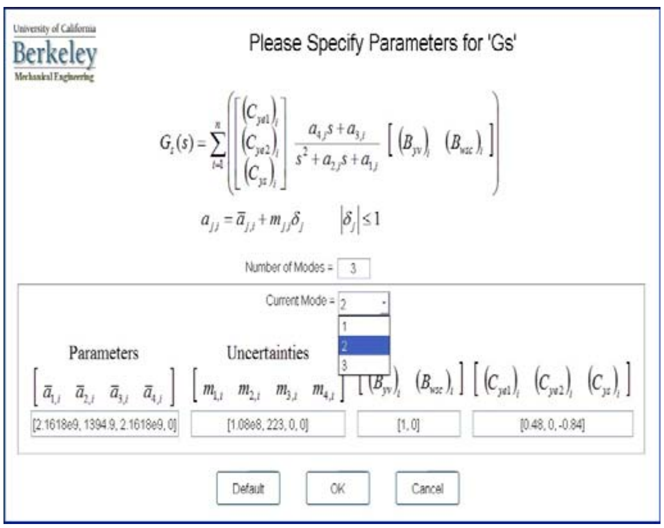

Figure 12: GUI for specifying the suspension mode model for a servo with translational MEMS actuated slider

Once the model parameters for all of the blocks have been specified, the user must combine all of the block models into a single LFT for the plant by using the 'Calculate' pushbutton. At this point, the user can save the model parameters and/or the plant LFT to a data file.

The E-block, suspension, and slider actuator models are all formed from sums of SISO transfer functions, each of which is multiplied on the left and right by static gain matrices of the appropriate size. Each transfer function parameter is assumed to be uncertain with known bounds whereas the static gain matrices are assumed to be known. In the E-block and suspension models, the user is allowed to use an arbitrary number of SISO transfer functions to model their respective dynamics, as shown in Figure 12.

The measurement noise, windage, and runout characteristics of the system can be specified using any one of several formats. For each signal, the user can choose to specify its variance, the filtering transfer function, the spectral density transfer function, or the spectral density frequency response. In the case of spectral density frequency response specification, a stable, minimum phase transfer function of the desired order is fit to the frequency response magnitude using a version of log-Chebychev magnitude design [9].

Control Design Specification Interface

The control design specification interface, shown in Figure 13, has a structure very similar to the plant modeling interface. Like the plant modeling interface, the control design specification interface has a graphical form with pushbuttons in relevant locations to specify control design options. After the options corresponding to a particular pushbutton have been specified, the background color of that pushbutton darkens. There is also a pushbutton to set all design options to their default values and a pushbutton to save all of the design options to a data file.

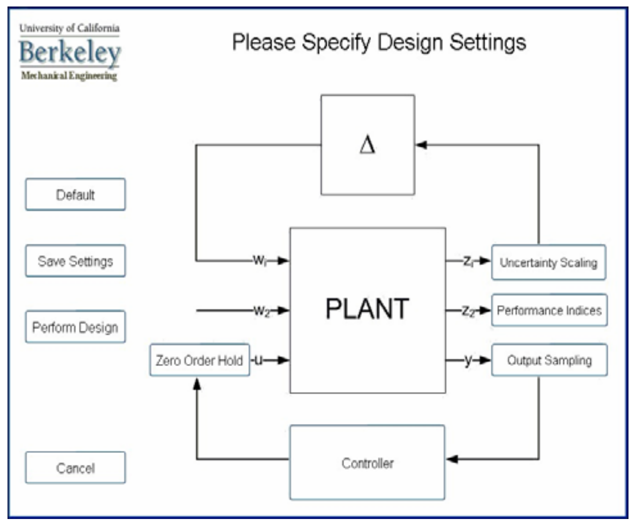

Figure 13: GUI for specifying controller design options

There are three sets of design options that must be set up in this interface. The first set of design options specifies which measurements are available to the controller, the sampling rates, and actuation rates. The sampling and actuation rates are assumed to be integer multiples of a common base rate. The second set of design options specifies the performance output. The performance output is a vector of signals which includes multiples of PES, RPES, and the two control inputs. The user can select which ratios of these signals will be used in the performance output. The third set of design options specifies which methodology to use in designing the controller—mixed H_2/H_∞ synthesis, mixed H_2/μ synthesis, or robust H_2 synthesis.

Model Example

Using the prototype interface, a plant model for a representative model of a servo with translational MEMS actuated slider was constructed and a controller was designed using robust H_2 synthesis. This model took into account parametric variations of up to 8% in the natural frequency and up to 20% in the damping coefficient for the butterfly and sway modes. For comparison, an equivalent plant model was created using offline modeling techniques and a controller was designed using robust H_2 synthesis.

Design Method	H_2 Norm to PES (nm)	
	Nominal	Worst Case
Interface	7.61	9.38
Offline	7.04	7.72

Table 1: Closed loop norms for both design methods

To analyze the resulting controllers, 400 samples of each plant were formed by randomly choosing sets of parameter values from their respective variation ranges. For each sample of the closed loop system, stability was checked and the H_2 norm of the closed loop system to the PES signal was found. All 400 samples of both cases were found to be stable and the H_2 norms are as summarized in Table 1. The interface design had slightly worse overall H_2 performance than the offline design. Figure 14 shows the nominal closed loop sensitivity functions for the system designed using the prototype interface and the system designed using offline modeling. The sensitivity function is defined as the transfer function from the track runout to the PES. Note that in the interface design, track runout is more attenuated than in the offline model and the sensitivity bandwidth is higher. Thus, although the interface design is more sensitive to windage and measurement noise, it has better track runout rejection properties.

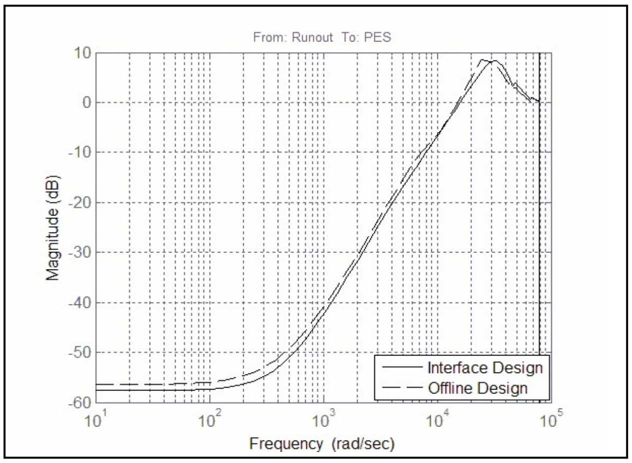

Figure 14: Nominal closed loop sensitivity plots for both design methods

6. CONCLUSIONS

In this paper we presented an object oriented framework for modeling and robust control design of hard disk drives. The framework adopts existing modeling design algorithms for source code and defines new applications to represent higher-level program abstractions. The framework is extensible; new representations and tools can be added to facilitate different generic tasks, such as analysis, controller order reduction, and system identification.

The sizes of the intermediate representations and the time required to generate them are reasonable. A preliminary operational framework has been developed and efficient programs have been written to construct generalized plant models. The preliminary framework software has been validated with an example of controller design for a translational MEMS actuated slider configuration.

ACKNOWLEDGEMENTS

We would like to thank Dr. Ryozo Nagamune for letting us to use his algorithms and for his valuable suggestions for this project. The work in this paper was supported by the Information Storage Industry Consortium (INSIC), the Computer Mechanics Laboratory of U.C. Berkeley, and grants from the National Science Foundation (NSF).The work of Dr Contreras was supported by University Simon Bolivar Sabbatical Grand.

7. REFERENCES

[1] R. Horowitz, T.L. Chen, K. Oldham and Y. Li, "Design, Fabrication and Control of Micro-Actuators for Dual-Stage Servo Systems in Magnetic Disk Files," **Springer Handbook of Nanotechnology,** Edit. Bharat Bushan, January 2004.

[2] M. White, T. Hirano, H. Yang, K. Scott, S. Pattanaik, and F-Y Huang, “High-bandwidth Hard Disk Drive Tracking Using a Moving-slider MEMS Microactuator,” **IEEE 8th IEEE Int’l Workshop Advanced Motion Control,** 2004, pp. 299–304.

[3] Y. Li and R. Horowitz, “Design and testing of track following controllers for dual-stage servo systems with PZT actuated suspensions,” **Microsystems Technologies**, vol. 8, 2002, pp. 194-205.

[4] Y. Li, F. Marcassa, R. Horowitz, R. Oboe, and R. Evans, “Track-following control with active vibration damping of a PZT-actuated suspension dual-stage servo system,” in **Proc. American Control Conf.**, 2003, pp. 2553–2559.

[5] K. Oldham, S. Kon, and R. Horowitz, “Fabrication and optimal strain sensor placement in an instrumented disk drive suspension for vibration suppression,” **Computer Mechanics Laboratory Blue Report**, Nov 2003.

[6] L. Contreras, J. Molero, and A. Sucre, “Object Oriented Metamodel of Systems Administrative Production Process,” in **Proc. of the World Multiconference on Systemics, Cybernetics and Informatics: Information Systems Development-Volume I,** July 22 - 25, 2001, IIIS, pp. 601-606.

[7] R. Nagamune, X. Huang and R. Horowitz, “Multi-rate track-following control with robust stability for a dual-stage multi-sensing servo system in HDDs,” in **Proc. of the Joint 44th IEEE Conference on Decision and Control and European Control Conference ECC 2005**, Seville, Spain, December 12-15, 2005, pp. 3886-3891.

[8] S. Kanev, C. Scherer, M. Verhaegen, and B. De Shutter, “Robust output feedback controller design via local BMI optimization,” **Automatica,** 40, 2004, pp. 1115–1127.

[9] A.V. Oppenheim and R.W. Schaffer, **Digital Signal Processing**, Prentice Hall, New Jersey, 1975, pp. 513.