SIDF-Based Methods for Control System Synthesis

Prof. James H. Taylor
Department of Electrical & Computer Engineering
University of New Brunswick
Fredericton, NB CANADA E3B 5A3
telephone: +506.453.5101
internet: jtaylor@unb.ca
web site: www.ee.unb.ca/jtaylor

29 July 2002

JUV RESCCE'02 Summer School
Danang, Viet Nam
05-09 August 2002

Topic Outline: SIDF-Based Nonlinear Control Synthesis

- Introduction: Motivation and Background
- Design Philosophy
- Basic Considerations and Concepts
- SIDF Modelling Approaches
- Overview of Nonlinear Controller Synthesis
- One Degree of Freedom Nonlinear Controller Synthesis
- Three Degree of Freedom Nonlinear Controller Synthesis
- Fuzzy Logic Controller Approach
- Nonlinear PID Autotuning

References:

1. J. H. Taylor and K. L. Strobel, "Applications of a Nonlinear Controller Design Approach Based on Quasilinear System Models", *Proc. American Control Conference,* San Diego, CA, June 1984.
2. J. H. Taylor and K. L. Strobel, "Nonlinear Compensator Synthesis Via Sinusoidal-Input Describing Functions", *Proc. American Control Conference,* pp. 1242-1247, Boston, MA, June 1985.
3. J. H. Taylor and J. R. O'Donnell, "Synthesis of Nonlinear Controllers with Rate Feedback via SIDF Methods", *Proc. American Control Conference,* pp. 2217-2222, San Diego, CA, May 1990.
4. J. H. Taylor and J. Lu, "Robust Nonlinear Control System Synthesis Method for Electro-Mechanical Pointing Systems with Flexible Modes", *Journal of Systems Engineering* (Special issue on motion control systems), January, 1995.
5. J. H. Taylor, *Describing Functions*, an article in the *Electrical Engineering Encyclopedia*, John Wiley & Sons, Inc., New York, 1999.

Introduction

- **Basic Problem:** How to design **effective** controllers for **nonlinear systems**
- **Considerations:**
  1. Small-signal linearization often yields a system model that is inadequate for designing a good controller
  2. Many approaches deal with designing **linear** controllers, which may not provide acceptable performance
  3. Too much engineering time is spent modifying and "tuning" control system (due to the first two points)
  4. Few **systematic** methods exist for synthesizing nonlinear controllers to achieve traditional performance criteria
- **Effective, robust control is essential in many mechatronic systems**

Introduction (Cont'd)

Considerations in robust nonlinear control:

- Nonlinear systems behavior is sensitive to both **operating point** and **input amplitude**
- Nonlinear systems behavior is sensitive to **modeling uncertainty** (how you model various effects, as well as parametric uncertainty)
- Engineering models of nonlinear effects are often not "nice" in mathematical terms (e.g., may involve discontinuous or multivalued functions)
- Intuitive methods involving natural performance criteria are very desirable

These issues motivated a re-examination of describing function methods as a basis for robust nonlinear control

A Robust Nonlinear Control Problem

Problem Statement: Synthesize a **nonlinear controller** so that the behavior of the resulting control system is as **insensitive to input amplitude as possible**

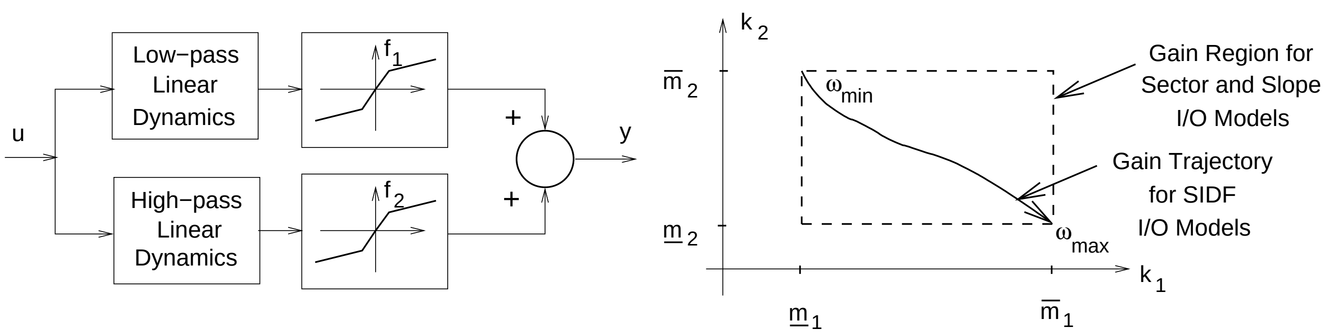
- Failure to deal with input amplitude sensitivity is likely to lead to a control system that is **not robust**[1]
- Operating point sensitivity may also be handled using describing function methods, but this is beyond the scope of this presentation
- Many other nonlinear control problems have been considered and have utility; different problems require different solution methods

This problem definition may be called **robust performance**

[1]E.g., which may behave very differently for small *versus* large input excitation, or perhaps exhibit limit cycles or instability

Linearized Models for Design

- **No** linearization method is rigorously robust:
  - Small-signal linearization
  - Small-signal models based on sector bounds
  - Small-signal models based on slope bounds
  - Sinusoidal-input describing function (SIDF) models
  - Random-input describing function (RIDF) models
- Many nonlinear effects depend on both **frequency** as well as amplitude:



- SIDF models account for both factors; none of the others do
- SIDFs provide the best compromise in terms of **safety** and **lack of conservatism**

Key Concept of SIDF-based Synthesis

- **Basic concept: the SIDF I/O model**

$$
\begin{aligned}
u(t) &\triangleq u_0 + a\cos(\omega t) \\
y &\cong y_0 + \mathrm{Re}[\, b\exp(j\omega t)\,] \\
b &= G(j\omega; u_0, a)\cdot a
\end{aligned}
$$

- **Methods for obtaining SIDF I/O models:**
  - Solve harmonic balance relations
  - Perform direct simulation plus Fourier analysis
  - Take lab measurements

Outline of SIDF Approach for Controller Synthesis

1. Choose three or more operating ranges ("small, medium, large" input amplitudes)

2. Obtain SIDF I/O models for the plant for each amplitude

3. Choose a fixed controller configuration

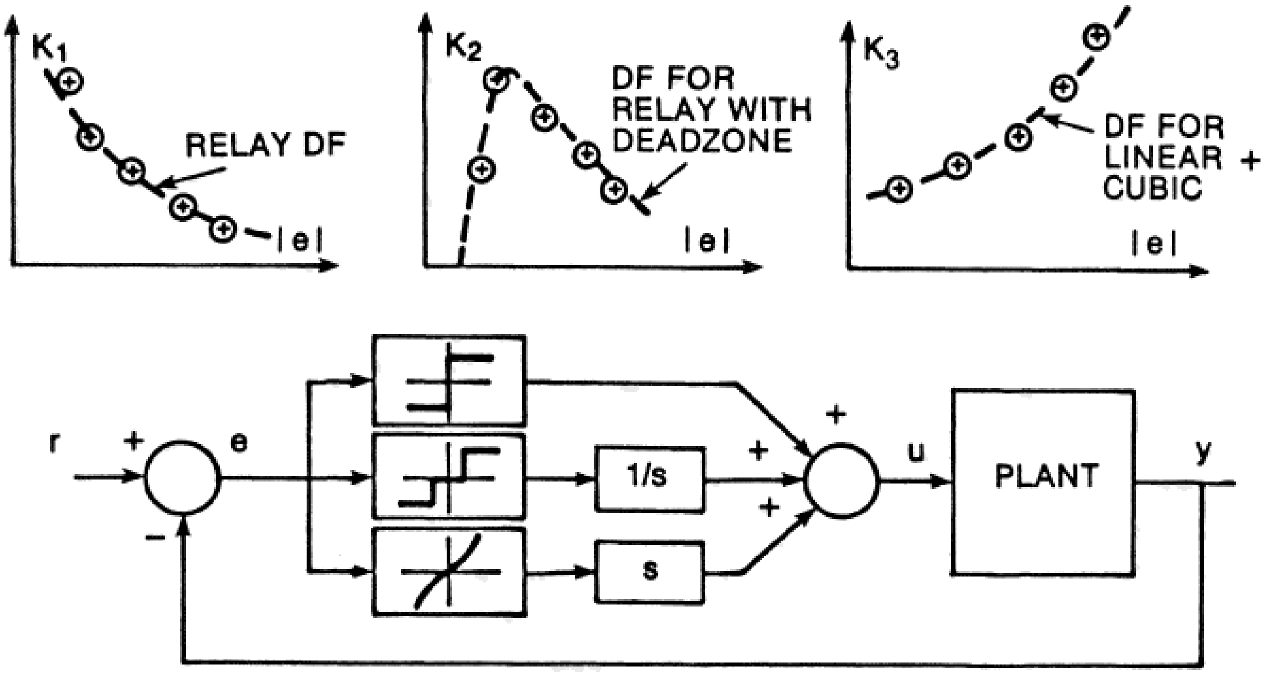
4. Use the set of SIDF models to synthesize a set of *linear* controllers, based on an open-loop frequency-domain specification

5. Interpret the amplitude-dependent gains as SIDFs for controller nonlinearities; invert the SIDFs to obtain the nonlinearities

6. Validate the nonlinear controller design via simulation

Controller Synthesis (Cont'd)

Example: PID controller configuration, $K_p + K_I/s + K_D s$

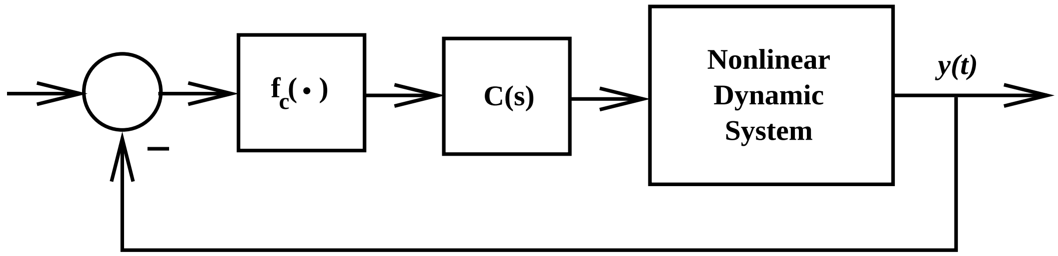


Justification:

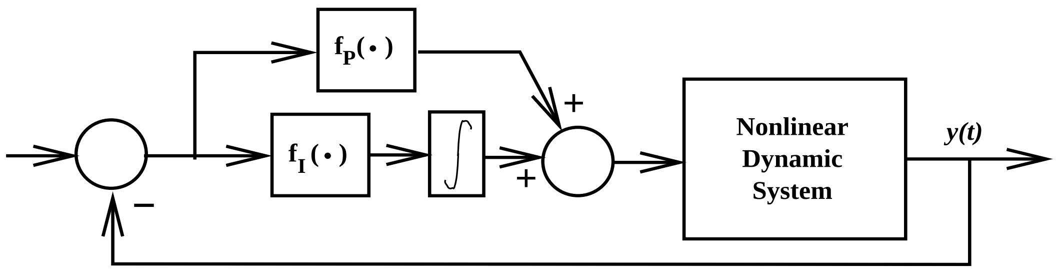
- This synthesis approach is completely systematic
- There is no direct nonlinearity cancellation
- Performance is as uniform as possible over the operating range of input amplitudes

Nonlinear Controller "Degrees of Freedom" (DoF)

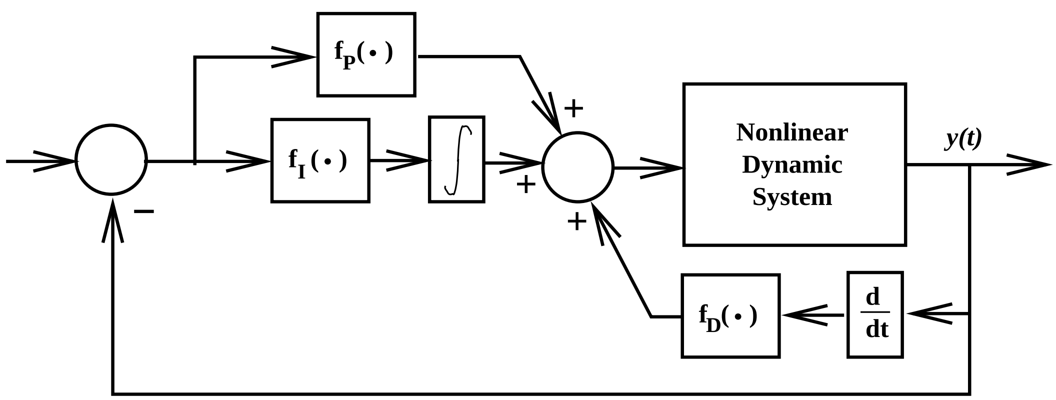
- One degree of freedom (1-DoF):



- Two degrees of freedom (2-DoF):



- Three degrees of freedom (3-DoF):

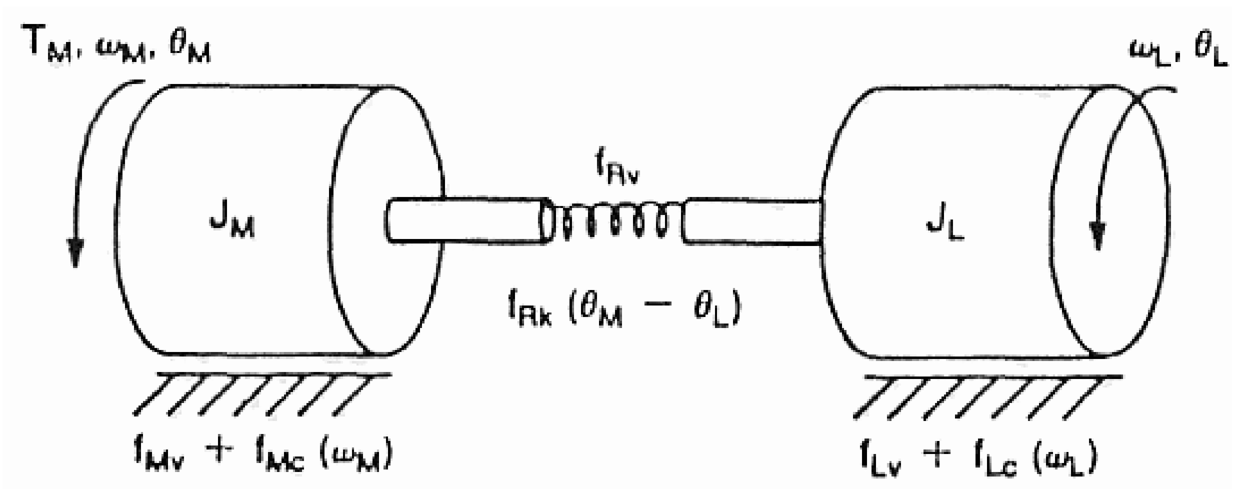


(there is also **PID in the forward path**)

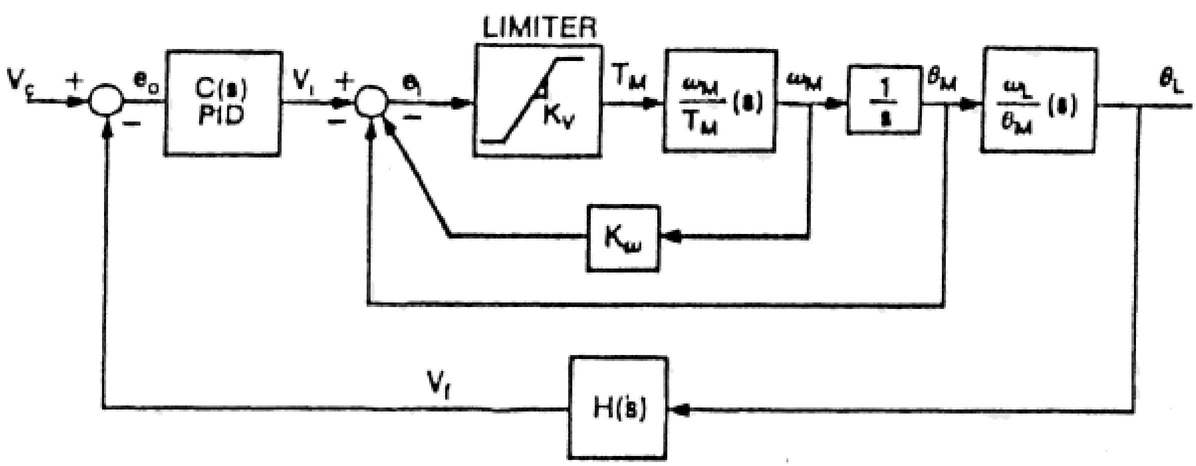
Choice is based on the **degree** and **type of variability** in the nonlinear plant's SIDF I/O models

One Degree of Freedom Synthesis

Problem: design an effective controller for the base unit of an industrial robot with a stiff harmonic drive and servo saturation



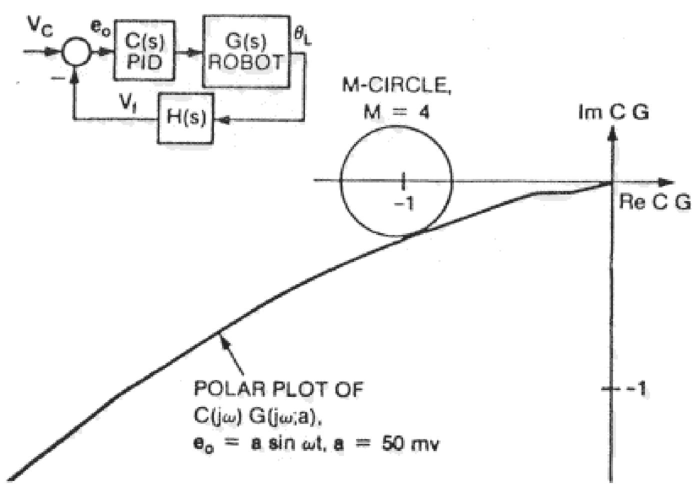
Single-axis robot model schematic

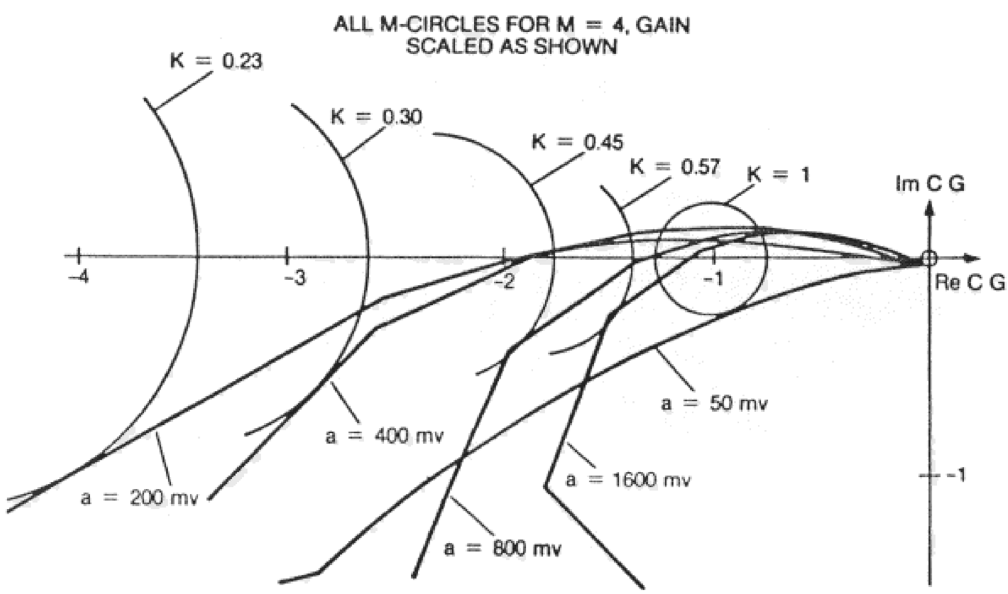


Robot model components identified in the lab

1-DoF Synthesis (Cont'd)

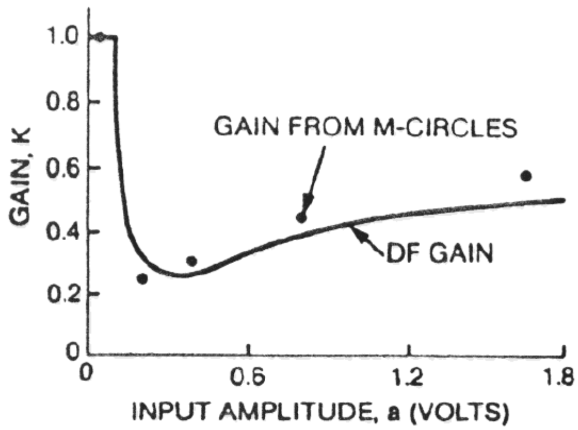
- Design a controller $C(s)$ (PID) based on 50 mv input amplitude, using the M = 4 criterion
- Find gains that bring other input amplitudes into line with the design specification

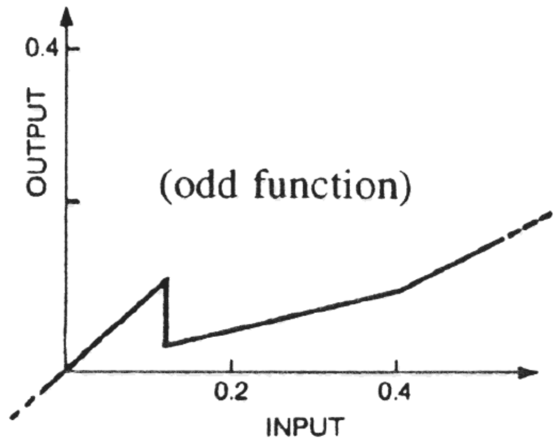




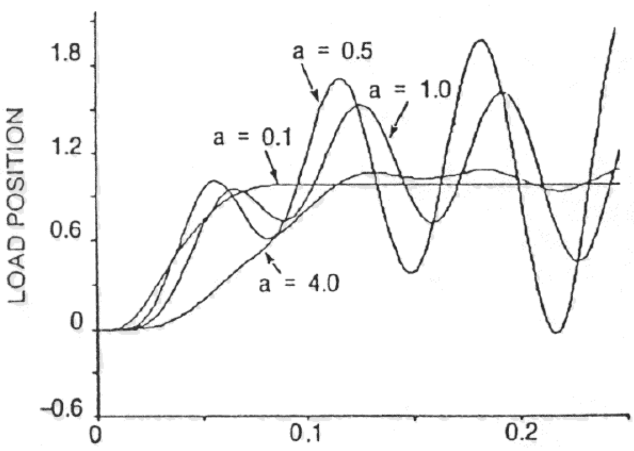
1-DoF Synthesis (Cont'd)

- The M-circle criterion provides $K(a)$; the nonlinearity is obtained by SIDF inversion:

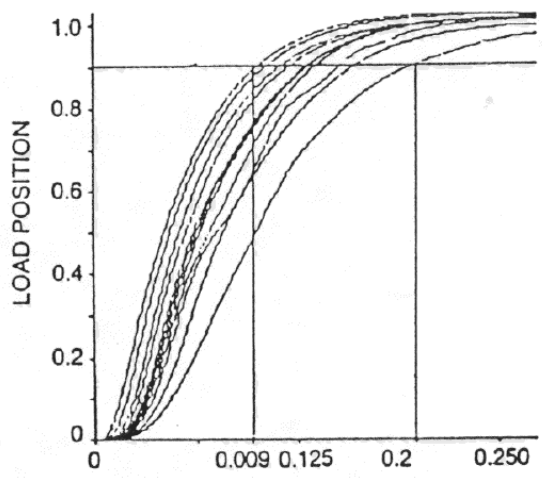




- Step responses with linear and nonlinear control:



(a) Linear Control



(b) Nonlinear Control

Comments on SIDF Inversion

- The term "SIDF inversion" refers to optimizing the parameters of a nonlinearity of a specified form to fit the SIDF of that nonlinearity to a desired gain/amplitude relation with minimum mean square error

- Piece-wise linear (PWL) functions are particularly well suited to this use:

  - Unlike polynomial fitting, the behavior of a PWL function is robust over interpolation and extrapolation

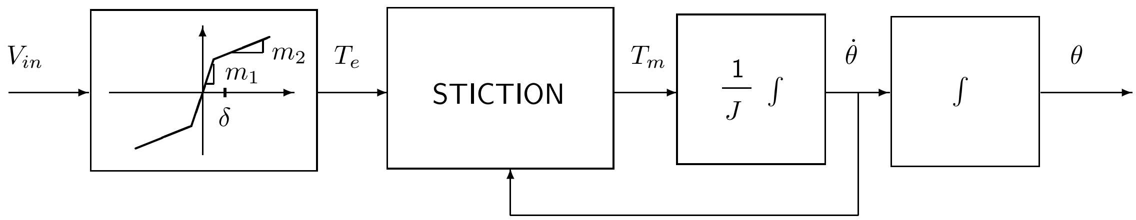
  - A variety of behaviors can be obtained with simple PWL functions

  - Simplicity translates into efficient parametrization and easy optimization

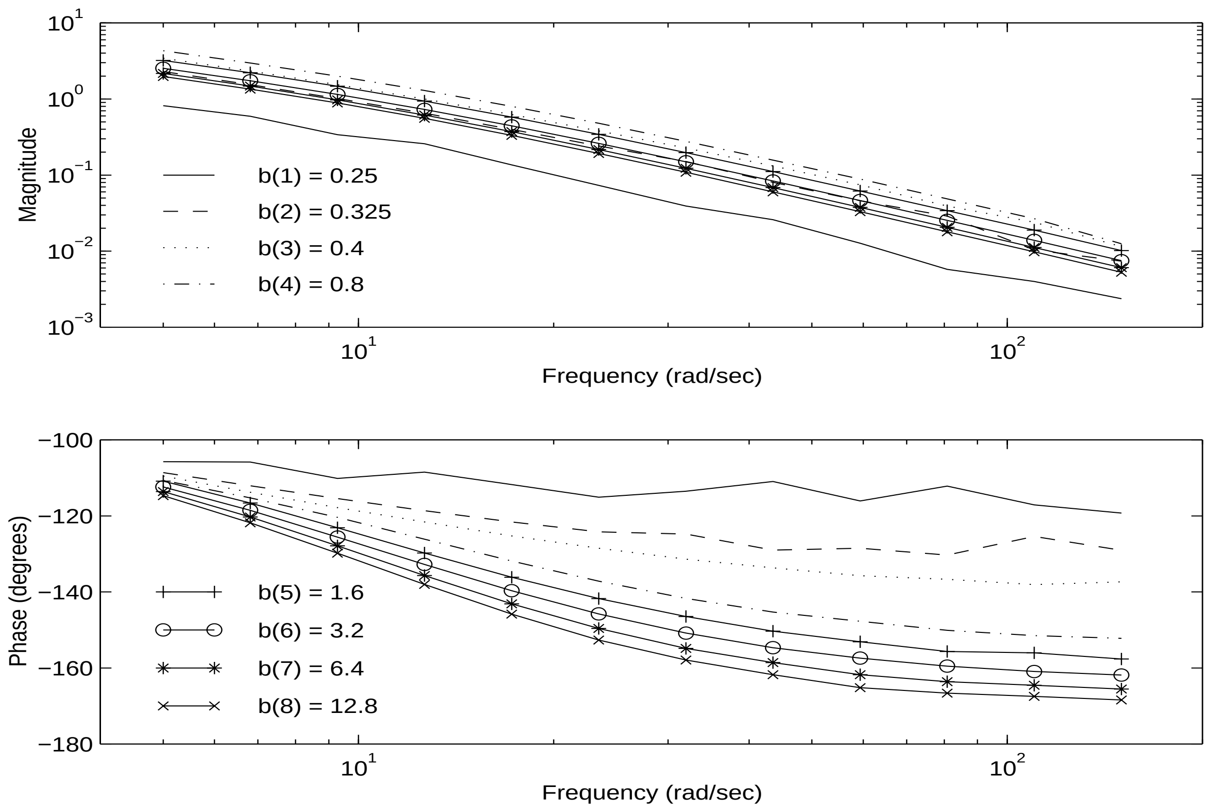
  - PWL functions are easy to implement in hardware or software

Three Degree of Freedom Synthesis

Recall the electromechanical system with stiction:

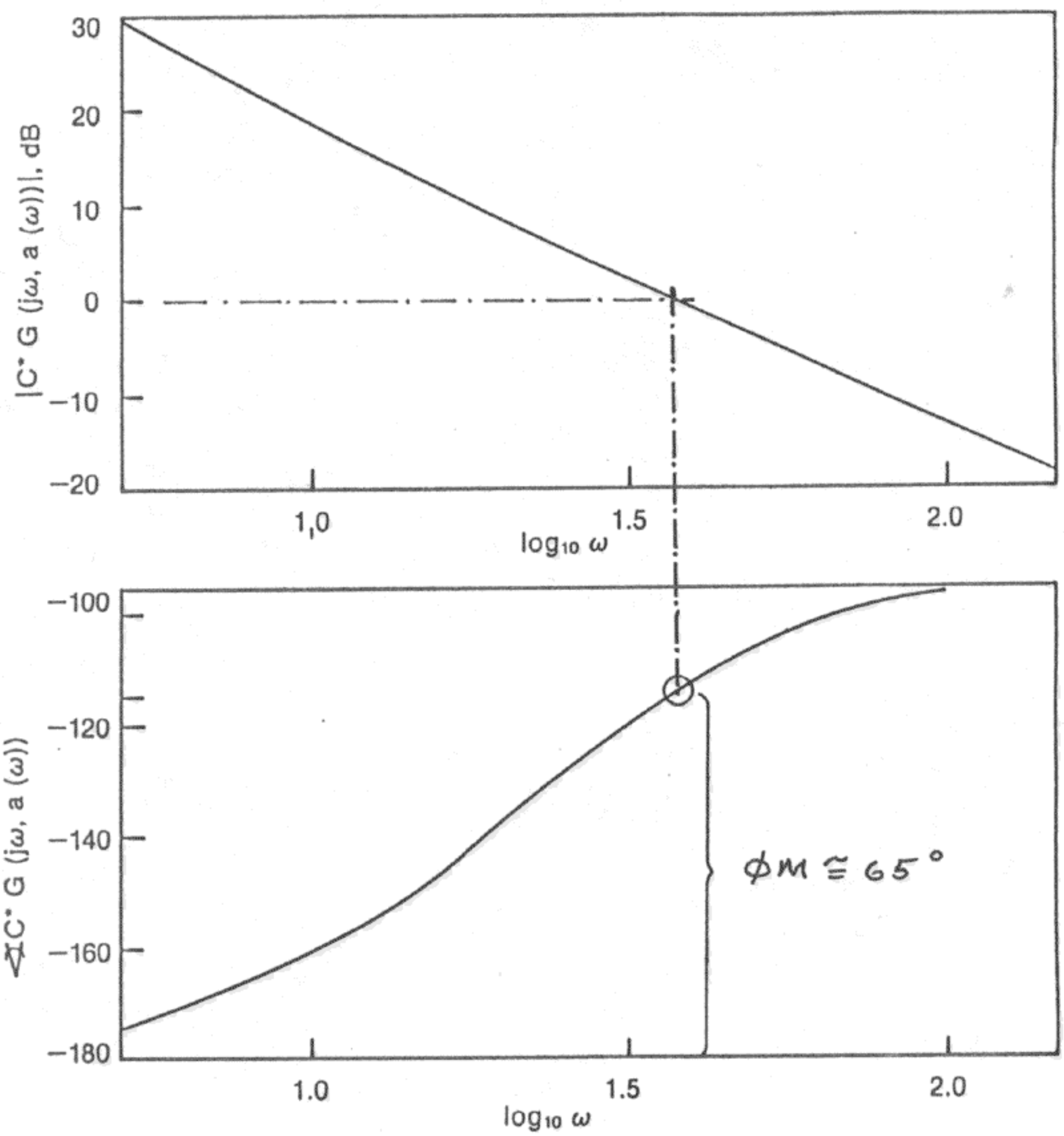


for which we obtained the following SIDF I/O model:



3-DoF Synthesis (Cont'd)

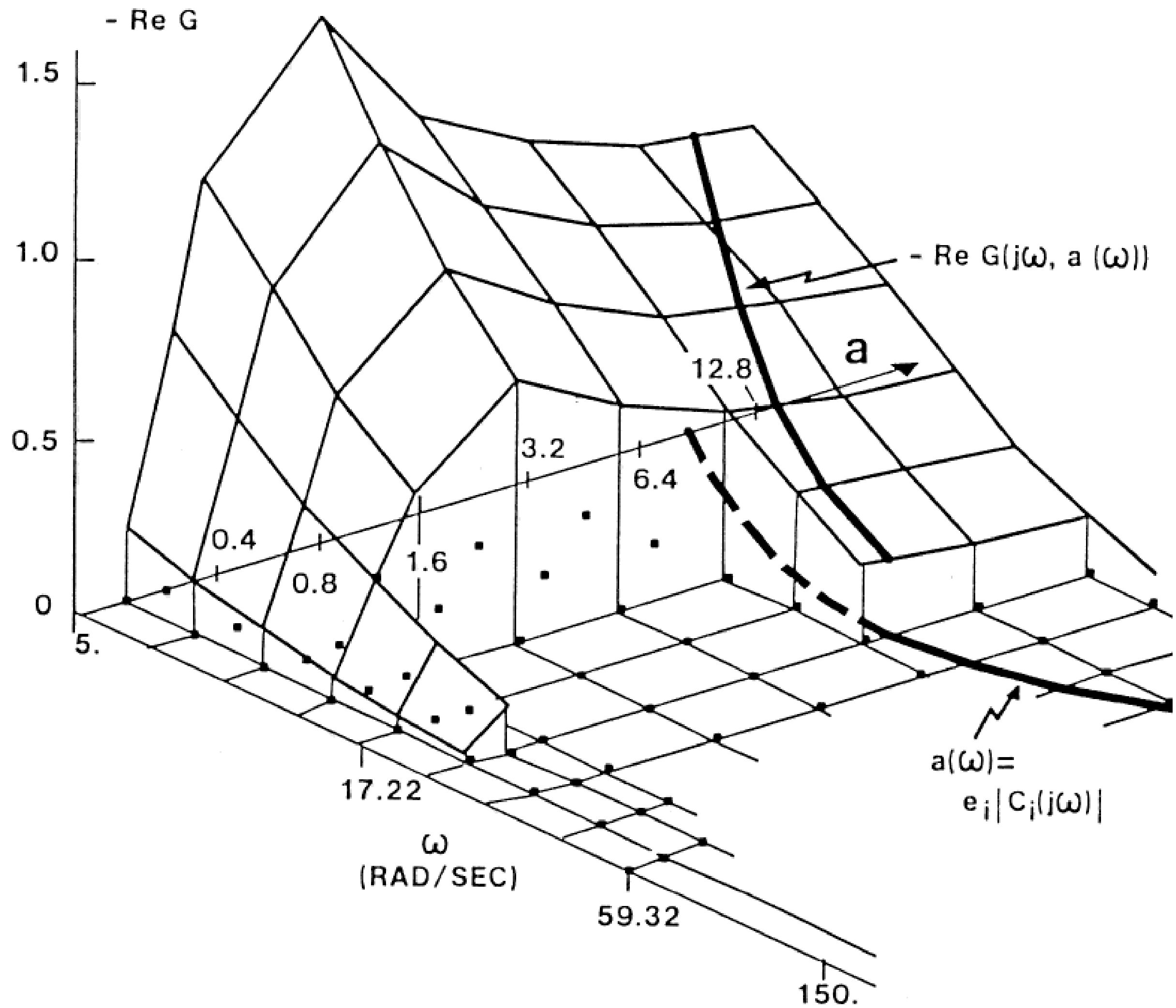
1. Design a linear PID, with a comfortable phase-margin spec, based on one SIDF I/O model to obtain the desired open-loop frequency response called $CG^{\star}(j\omega)$:



3-DoF Synthesis (Cont'd)

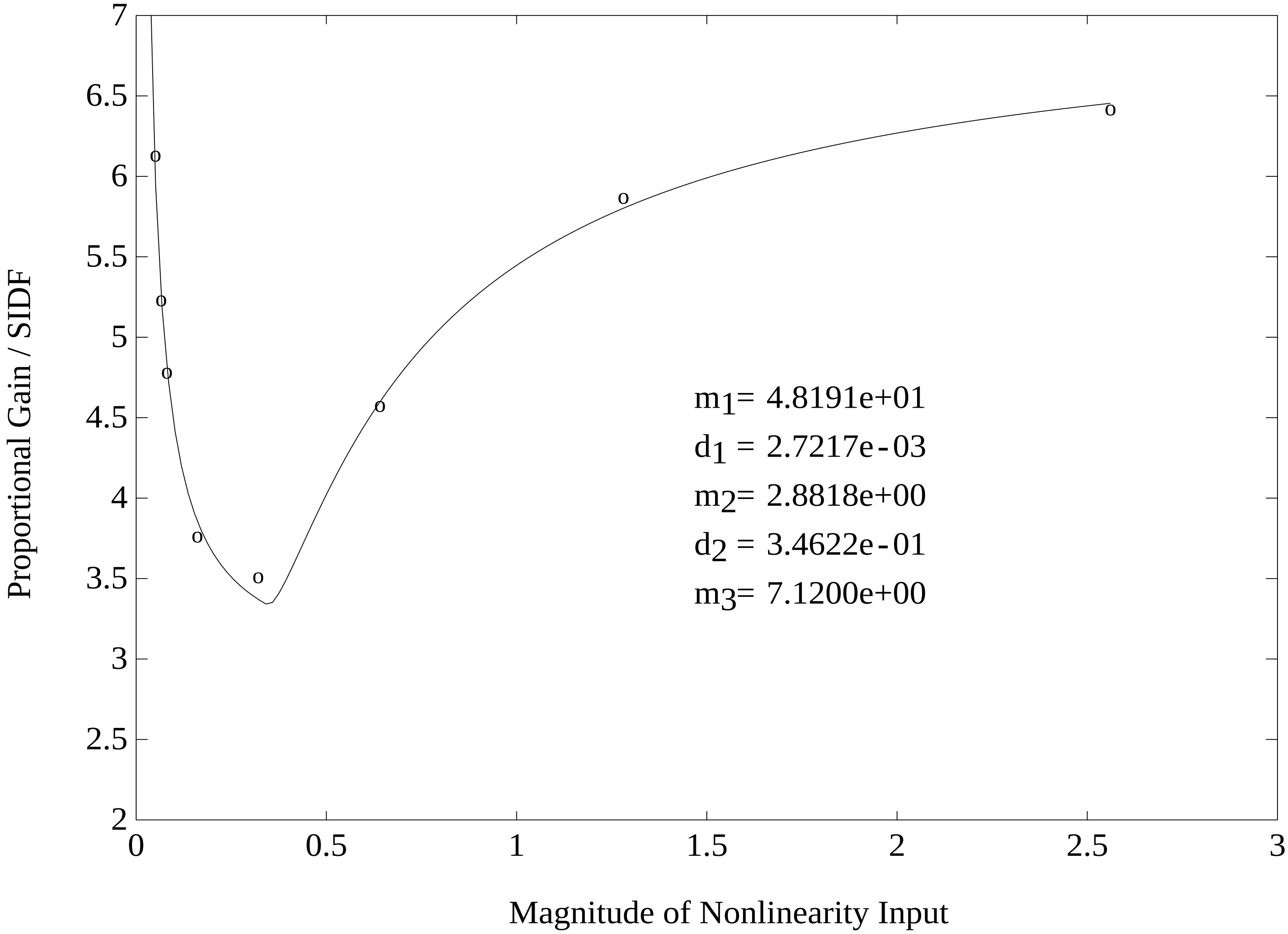
2. Optimize the PID controller gains for different input amplitudes:

Select error signal amplitude $e_i \in \{ e_i \}$ and use the current controller gains to obtain the corresponding plant input amplitude $a_i(\omega_j)$ over a set of frequencies $\omega_j \in \{ \omega_j \}$, interpolate to find $G(j\omega; a_i)$, iterate the controller gains to fit the desired $CG\star(j\omega)$



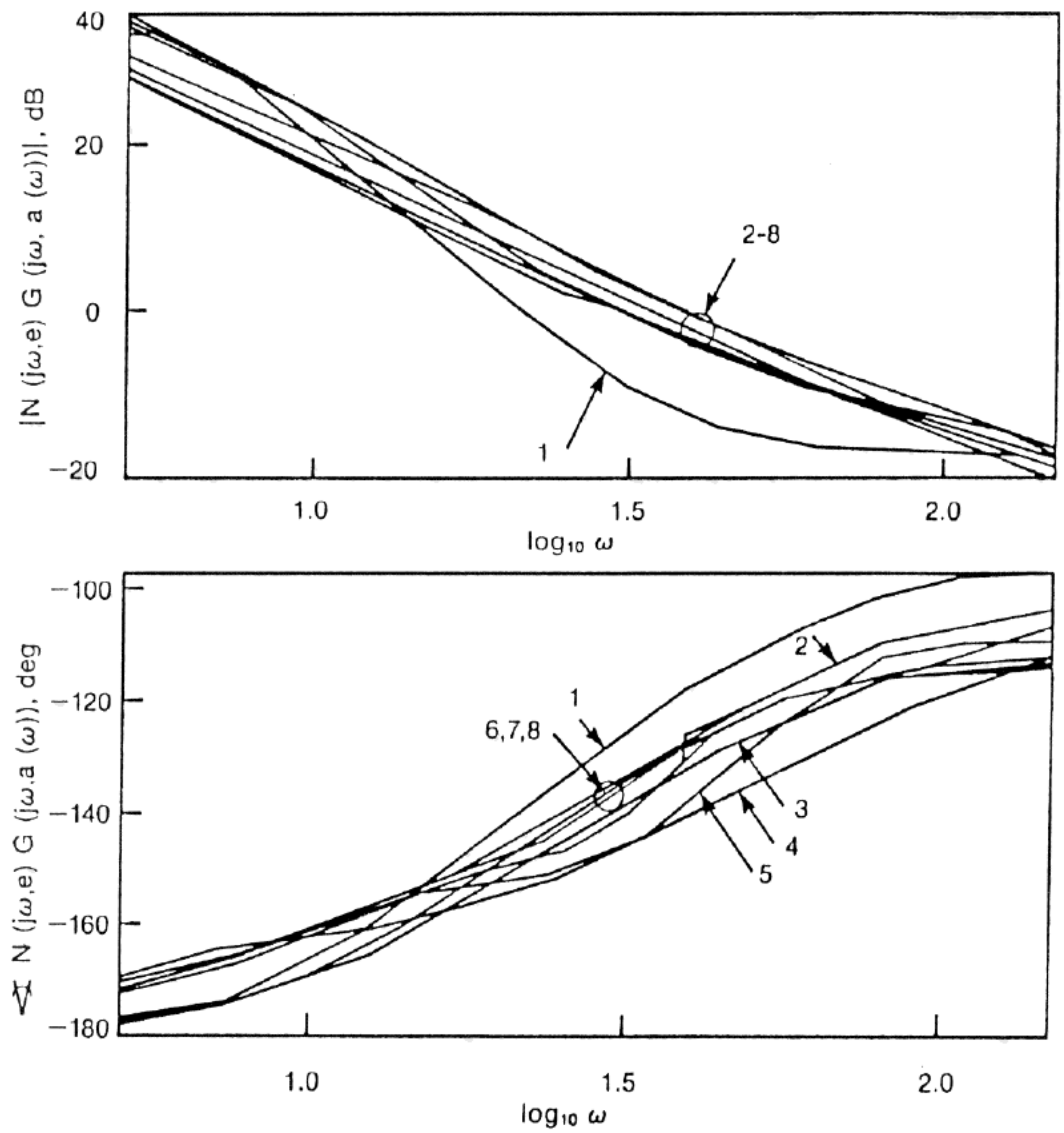
3-DoF Synthesis (Cont'd)

3. Take the controller gains $K_{P,i}(e_i)$ etc. from the previous step and use SIDF inversion to synthesize the controller nonlinearities; for example,



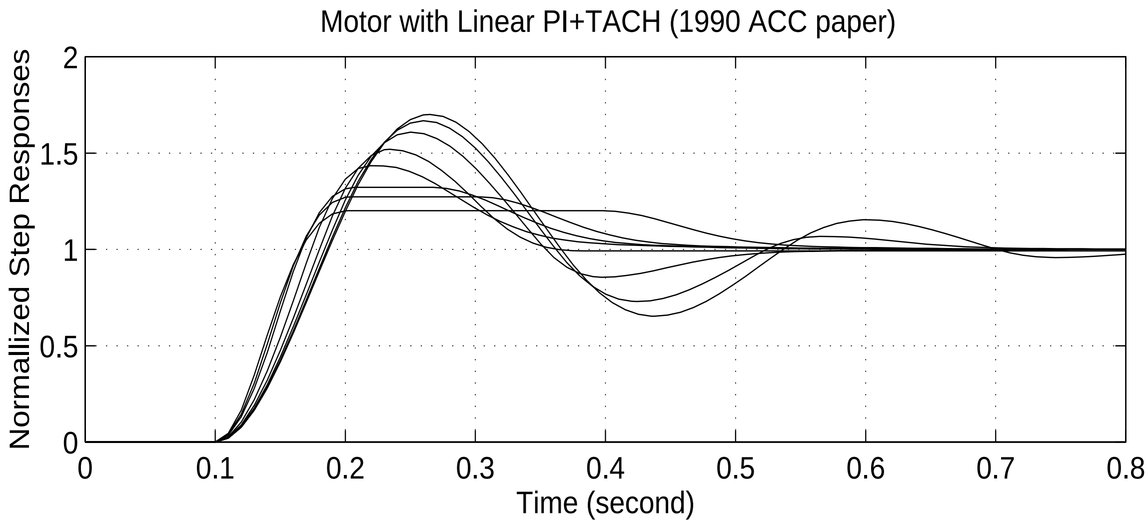
3-DoF Synthesis (Cont'd)

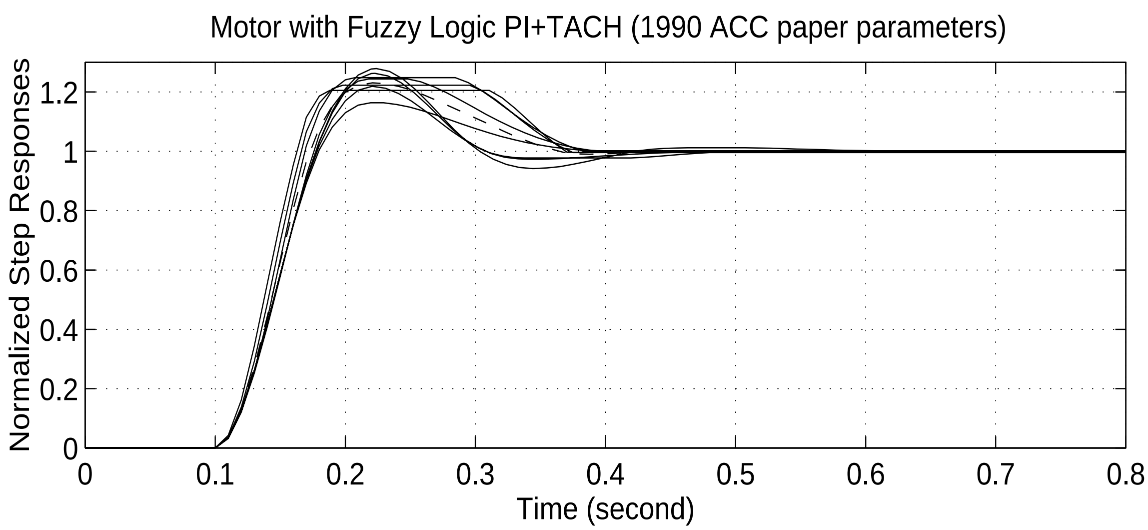
4. Check sensitivity of the open-loop frequency responses with nonlinear compensation:



3-DoF Synthesis (Cont'd)

Finally: check the step responses of the linear and nonlinear compensated systems:





(In fact, this 3 degree-of-freedom controller incorporates rate feedback and was implemented using fuzzy logic.)

Other SIDF-Based Design Methods

A number of other approaches were developed from the basic SIDF-based design method:

- **Fuzzy-logic control**, including time-domain optimization – to refine SIDF-based design by directly minimizing the sensitivity of the closed-loop step response

- **Nonlinear controller autotuning** – to automatically synthesize a controller nonlinearity for 1-DoF robust control

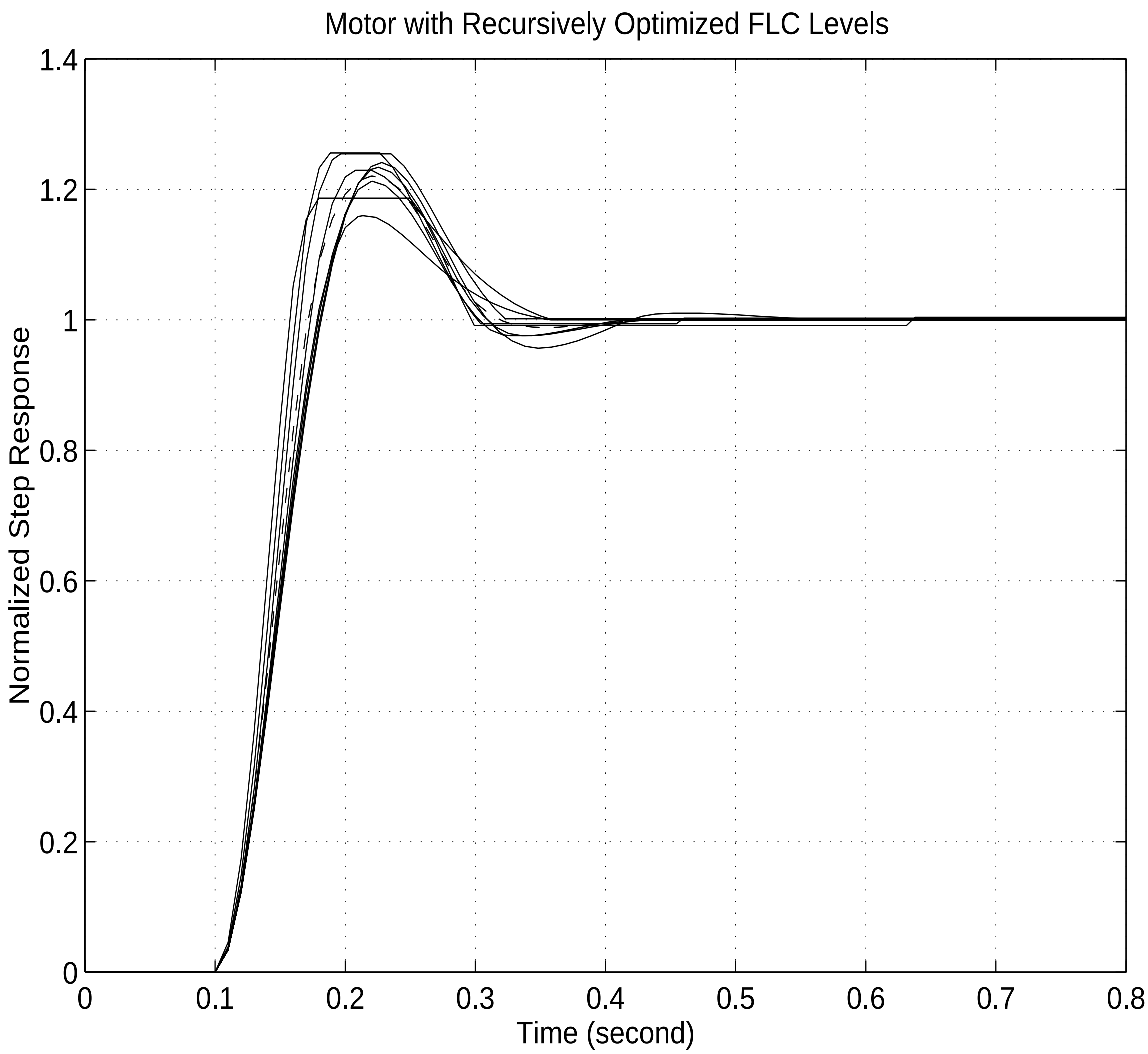
These extensions and others are due to the fact that SIDF methods fit in with known linear techniques in an intuitive and simple manner

References:

1. J. H. Taylor and L. Sheng, "Recursive Optimization Procedure for Fuzzy-Logic Controller Synthesis", *Proc. American Control Conference,* Philadelphia, PA, pp. 2286-2291, June 1998.
2. J. H. Taylor and K. J. Åström, "A Nonlinear PID Autotuning Algorithm", *Proc. American Control Conference,* pp. 2118-2123, Seattle, WA, 18-20 June 1986.

Synthesis Using Time-Domain Ooptimization

Here is a typical result using time-domain optimization **after** a preliminary SIDF-based design:



This **could not be achieved** using direct/blind time-domain optimization!

SIDF-Based Synthesis: Conclusions

- Several nonlinear compensator synthesis methods based on SIDF I/O models have been developed and applied to **difficult electromechanical systems** (real and modeled)

- **Validation** in both the time and frequency domain has been **highly successful**

- SIDF-based modeling and synthesis is **broadly applicable**, regardless of system order, number or type of nonlinearity, or configuration – **"If you can simulate the system you can use SIDF-based synthesis."**

- **Excellent robustness** has been achieved in every case