Fuzzy-Logic Controller Synthesis for Electro-mechanical Systems with Nonlinear Friction

James H. Taylor & Lan Sheng
Department of Electrical Engineering
University of New Brunswick
Fredericton, NB CANADA E3B 5A3
Internet: *jtaylor@unb.ca*

Abstract

We report on preliminary results in developing a new method for the synthesis of fuzzy-logic controllers for amplitude-sensitive nonlinear plants based on sinusoidal-input describing-function (SIDF) methods. This technique exploits the fact that two traditional classes of fuzzy-logic controllers are, in functional terms, of the proportional-plus-derivative (PD) and proportional-plus-integral (PI) types. It involves the direct generation of the membership functions and output levels based on the "frequency response" of the nonlinear plant in the describing-function sense ($G(j\omega, a)$ where a is the input amplitude). The resulting fuzzy-logic controller obtained in this paper includes derivative action in an inner-loop feedback path (nonlinear rate feedback) and nonlinear PI compensation in the forward path; the performance of the closed-loop system is, by design, quite insensitive to reference-input amplitude.

An illustration of the method and its effectiveness is provided, based on a prototypical position control problem where a servo motor plus mechanical load are characterized by torque saturation and nonlinear friction (stiction). We emphasize, however, that this approach is capable of treating nonlinear systems of a very general nature, with no restrictions as to system order, number of nonlinearities, configuration, or nonlinearity type.

1   INTRODUCTION

We present new research in the development of nonlinear control system design techniques based on sinusoidal-input describing-function (SIDF) methods. In this work, we combine an SIDF technique with a fuzzy logic methodology to create a direct fuzzy-logic controller synthesis procedure. The background for this work is established by focussing on these two fundamental foundations, namely SIDF methods and fuzzy control.

1.1   SIDF Methods

The SIDF basis of this work has been established previously: Taylor [1] outlined the motivation for using a modern[1] SIDF approach for control system design and established a systematic plan of attack, and both Taylor and Strobel [2, 3] and Taylor and O'Donnell [4] presented several implementations of these concepts. More specifically, this modern SIDF approach was first applied in [2], where a linear PID (proportional-integral-derivative) compensator in series with a single static nonlinearity was designed, and it was subsequently extended to the design of a fully nonlinear PID compensator, i.e., a PID controller with a nonlinearity in each of the three channels [3]. Then the extension to independent rate-feedback and PI forward-path compensation was realized [4], and finally the use of the SIDF approach for control of nonlinear electro-mechanical systems with flexible members was accomplished [7], by combining an SIDF-based inner-loop controller (to take care of drive-train nonlinearities) with an outer-loop linear controller (to deal with the flexible modes). In every case, it was shown that the resulting nonlinear control scheme is capable of reducing amplitude sensitivity and even correcting instabilities in nonlinear control systems without unnecessarily sacrificing performance.

All of these techniques use a set of SIDF models of the nonlinear plant as the basis for nonlinear compensator synthesis. SIDF models are used because they provide an excellent characterization of the major nonlinear effect with which we are concerned: the sensitivity of the nonlinear plant's input/output (I/O) behavior to the amplitude of the input signal; this issue has been discussed in detail in [1, 2, 8, 9]. In summary, given an input in the form $u(t) = a\cos(\omega t)$ the I/O model is of the form

$$y(t) = Re[G(j\omega, a)ae^{j\omega t}]$$

---

[1] "Modern" refers to an algebraic approach [5, 6] that permits one to deal with a broad class of nonlinear systems wherein the configuration and type and number of nonlinearities are not constrained.

where higher harmonics are neglected in this representation. Generally we consider a <u>set</u> of SIDF models; corresponding to an amplitude set $\{a_i\}$ these are denoted $\{G(j\omega, a_i)\} = \{G_i\}$.

Once a set of SIDF plant models is available, the synthesis of a nonlinear controller proceeds as follows: first, a *linear compensator set* is designed based on these models, with the objective of making the overall open-loop control system as insensitive to input amplitude as possible for a set of error signal amplitudes $\{e_i\}$. This yields a parametrized set of compensators $\{C_i(e_i)\}$, where the configuration of each compensator is the same (e.g., PID) but the parameters differ (e.g., $\{K_{P,i}(e_i)\}$ etc.). Final synthesis of the nonlinear control system is then accomplished by SIDF inversion to determine the required compensator nonlinearities.

The particular approach presented here involves the design of fuzzy-logic rulebases that, in effect, realize a nonlinear rate-feedback inner-loop compensator plus a nonlinear PI compensator placed in cascade with the rate-feedback-compensated plant; hereafter this is referred to as a PI+RF (proportional, integral plus rate-feedback) fuzzy-logic controller. The unique aspect of this new method is that it leads to the direct synthesis of a fuzzy-logic controller that produces desired amplitude-insensitive performance in the same general way as in past SIDF-based approaches that led to conventional nonlinear controllers.

1.2   Fuzzy Control

Fuzzy-logic-based or fuzzy control has a long record of development and application. This is described well in Tong [10], Sugeno [11] and Berenji [12]. For the purposes of this presentation, we merely point out the well-established connection between one standard form of fuzzy control – where the decision variables are "error" and "error-rate-of-change" – and proportional plus derivative (PD) control [10]. In many applications, this idea is implemented by using operator heuristics to describe what the linguistic variables "large-negative", "small-negative", "zero", "small-positive", "large-positive" mean in terms of membership functions for error and error-rate-of-change, and then to describe the appropriate control action to take (level of plant input to apply) under various circumstances, e.g., "if error is large-negative and error rate is small-positive". Of course, the details such as the number of linguistic variables etc. vary from application to application; however, this general idea is the basis for an important class of fuzzy-logic controllers. A second type of fuzzy-logic controller corresponds in the same analogous way with PI control.

As mentioned, in many applications the set of fuzzy rules is based on operator heuristics. In other cases, the

control engineer may be the source of this knowledge. While these approaches have proven to be effective in numerous applications, there are many circumstances where the fuzzy logic rulebases are difficult or impossible to generate in this fashion. In these cases, a model-based approach for generating the set of fuzzy rules may be more appropriate. Situations where the system dynamics are too fast for a human operator to be able to cope and too nonlinear for the engineer to write a rulebase by intuition would seem to be two cases in point, and electromechanical systems with nonlinear friction would be a prime example.

1.3   Paper Overview

The remainder of this paper describes the steps involved in this new fuzzy-logic controller synthesis approach and its application to a motor + load model with saturation and stiction. Specifically, Section 2 outlines the approach, including (i) SIDF modeling, (ii) the generation of "amplitude-sensitive proportional, integral and rate-feedback gains" needed to control the nonlinear plant, and (iii) the conversion of this model-based information into fuzzy-logic rulebases. Section 3 presents the application of this approach to an electromechanical systems with nonlinear friction, and Section 4 provides a summary and conclusions of this work along with suggestions for future research.

2   NONLINEAR PI+RF DESIGN

First, it is important to state the premises of the SIDF design approaches that we have been developing:

1. The nonlinear system design problem being addressed is the synthesis of controllers that are effective for plants having frequency-domain I/O models that are *sensitive* to input amplitude (e.g., for plants that behave very differently for "small" and "large" input signals).

2. The primary objective of nonlinear compensator design is to arrive at a closed-loop system that is as *insensitive* to input amplitude as possible.

This encompasses a limited but important set of problems, for which gain-scheduled compensators cannot be used and for which other approaches (e.g., variable structure systems, model-reference adaptive control, global linearization) do not apply because their objectives are different (e.g., their objectives deal with *asymptotic* solution properties rather than *transient* behavior, or they deal with the behavior of *transformed* variables rather than physical variables).

An outline of the synthesis algorithm for the nonlinear PI+RF fuzzy-logic controller is as follows:

1. Select a set of input amplitudes and frequencies that characterizes the behavior of the plant in the operating regimes of interest.

2. Generate SIDF models of the plant corresponding to the input amplitudes and frequencies of interest.

3. Design amplitude-dependent rate-feedback gains using the extended D'Azzo and Houpis algorithm [4] (see below).

4. Convert this design into a fuzzy-logic rulebase.

5. Find SIDF models for the nonlinear plant plus nonlinear rate-feedback compensation (RF fuzzy-logic controller).

6. Design PI compensator gains using the technique described in [4].

7. Convert this design into a fuzzy-logic rulebase (PI fuzzy-logic controller).

8. Develop a simulation model of the plant with PI+RF fuzzy-logic controller.

9. Validate the design through simulation.

Only items 4 and 7 above are discussed in much detail. The selection of the input amplitudes and frequencies for SIDF generation (items 1 and 5) and the use of modeling and simulation to validate the design (items 8, 9) depend on the familiarity of the designer with the system in question. The actual generation of SIDF models (items 2 and 5) is discussed briefly below; see [14, 15] for more details. Generating amplitude-dependent gains (items 3 and 6) is done as in [4].

2.1 SIDF Modeling

The generation of sinusoidal-input describing function models that provide an amplitude-dependent I/O characterization for a nonlinear plant has been dealt with in detail in [2, 14, 15]. There are two basic approaches: solving the nonlinear algebraic equations derived from the principle of harmonic balance, and simulation coupled with Fourier analysis.

The first method is not easy to apply, especially if it is desired to develop a general package that substitutes the appropriate SIDFs into the nonlinear algebraic equations and solves them. Also, it is assumed that the input to each nonlinearity is approximately sinusoidal [8, 9], which may leave the analysis open to question. However, there is an advantage to this approach: the SIDF model is obtained in a form that lends itself to further analysis such as predicting the existence of limit cycles.

The second technique is easier to implement, given a good package for integrating nonlinear differential equations, and it avoids the controversial assumption that the inputs to every nonlinearity are nearly sinusoidal. The only qualification is that a frequency-domain amplitude-dependent I/O model must provide a good representation of the behavior of a nonlinear plant for control system design; that issue has been discussed in [1, 2]. In our opinion, while SIDF models are not exact, a *set* of SIDF models covering the range of input amplitudes that will be encountered provides an excellent basis for "robust design", in the sense that the sensitivity of a nonlinear plant's behavior to input amplitude is one of the most important issues in robustness, and the SIDF I/O model is the least conservative model that accurately accounts for this factor.

A MATLAB-based tool has been developed to perform SIDF I/O model generation for nonlinear system models. The basic idea is to drive the nonlinear plant with a sinusoid of the desired amplitude for a number of frequencies of interest, and evaluate Fourier integrals as the simulation proceeds. The Fourier integrals are appended to the nonlinear system differential equation set, and they are integrated simultaneously. The simulation for a given frequency is sampled after each cycle and stopped when the Fourier integrals have converged; then the I/O model is evaluated, as described in Taylor [14]. Hereafter, we use the shorthand notation $G_i = G(j\omega; a_i)$ to designate the SIDF I/O model generated by driving a nonlinear system with the input $u(t) = a_i \cos(\omega t)$.

2.2 Generating Rate-feedback Gains

The general objective when designing the inner-loop rate-feedback controller is to give the same benefits expected in the linear case, namely stabilizing and damping the system, if necessary, and reducing the sensitivity of the system to disturbances and plant nonlinearities (see Thaler [16]). To accomplish this, we design a nonlinearity to be incorporated in the controller that desensitizes the inner-loop as much as possible to different input amplitudes.

Taylor and O'Donnell [4] achieved this result by extending a method from D'Azzo and Houpis [13], based on inverse Nyquist plots of the plant I/O model. This inner-loop rate feedback design algorithm (referred to as *Case 2* [13]) uses a construction amenable to extension to nonlinear systems. For linear systems, this algorithm generates fixed rate-feedback and forward-path gains that adjust the inverse Nyquist plot to be tangent to a given M-Circle at a selected frequency. The algorithm is extended to the nonlinear case by applying it to each SIDF frequency-response model G_i. Then for each input amplitude a rate-feedback or "tachometer" gain, $K_{T,i}$, and external (to the inner loop) gain $K_{E,i}$ is found. The $K_{E,i}$ values are discarded, since the external gain will be subsumed in the cascade PI compensator that is synthesized in the next step.

2.3 Generating PI Controller Gains

Referring back to the design strategy outlined above, another major step in the complete controller design is generating the nonlinear cascade PI compensator gains.

The general idea is to first generate SIDFs for the nonlinear plant (which, at this stage, is actually the nonlinear plant with nonlinear rate feedback) over the range of input amplitudes and frequencies of interest, denoted $G_{rf}(j\omega, a_i)$. This information plus a "frequency-domain open-loop objective function" $C^\star G^\star_{rf}(j\omega)$ forms a frequency-response map as a function of both input amplitude and frequency. The approach of [4] is then used to generate the gains $K_{P,i}(a_i)$ and $K_{I,i}(a_i)$ such that the overall open-loop frequency response fits $C^\star G^\star_{rf}(j\omega)$ as closely as possible for each a_i, thereby rendering it insensitive to input amplitude.

2.4 Conversion of Controller Gains into Fuzzy Rulebases

The set of desired rate-feedback gains $K_{T,i}(a_i)$ and PI compensator gains $K_{P,i}(a_i)$, $K_{I,i}(a_i)$ are then used to synthesize the rate-feedback and PI fuzzy-logic controllers. As first described in [1], this gain/amplitude data is interpreted as SIDF information for unknown static nonlinearities. Since simple fuzzy-logic functions define a class of static nonlinearities, the same approach can be used here. A least-squares routine is used to adjust the parameters of a fuzzy-logic function so that the SIDF of that nonlinearity fits this gain/amplitude data with minimum mean square error; this generates the desired fuzzy-logic controller. While the concept is identical to [2, 3, 14], the fuzzy-logic function used in this study is of greater generality and interest.

3 DESIGN EXAMPLE

A fuzzy-logic controller is designed for the same motor + load model used in [3, 4]. This model uses a substantial gain reduction to represent motor saturation, and a standard stiction nonlinearity. Following the general algorithm presented above, the first step is to perform a SIDF analysis of the motor. This was done using the MATLAB-based tool mentioned above, and the resulting I/O model is portrayed in Bode magnitude and phase response plots for the different input amplitudes considered (Fig. 1). As seen from the plots, the magnitude differs by over 14 dB and the phase response has significantly different shapes. Additionally, an examination of the higher harmonics of the SIDF response (not included) showed them to be several orders of magnitude lower than the fundamental. These two facts indicate the the SIDF approach is both valid and appropriate for this problem. The observation that the phase responses are very different in shape over the frequency range of interest indicates that one static compensator nonlinearity cannot desensitize the open-loop frequency response very well, so we choose to design a fully nonlinear PI+RF controller.
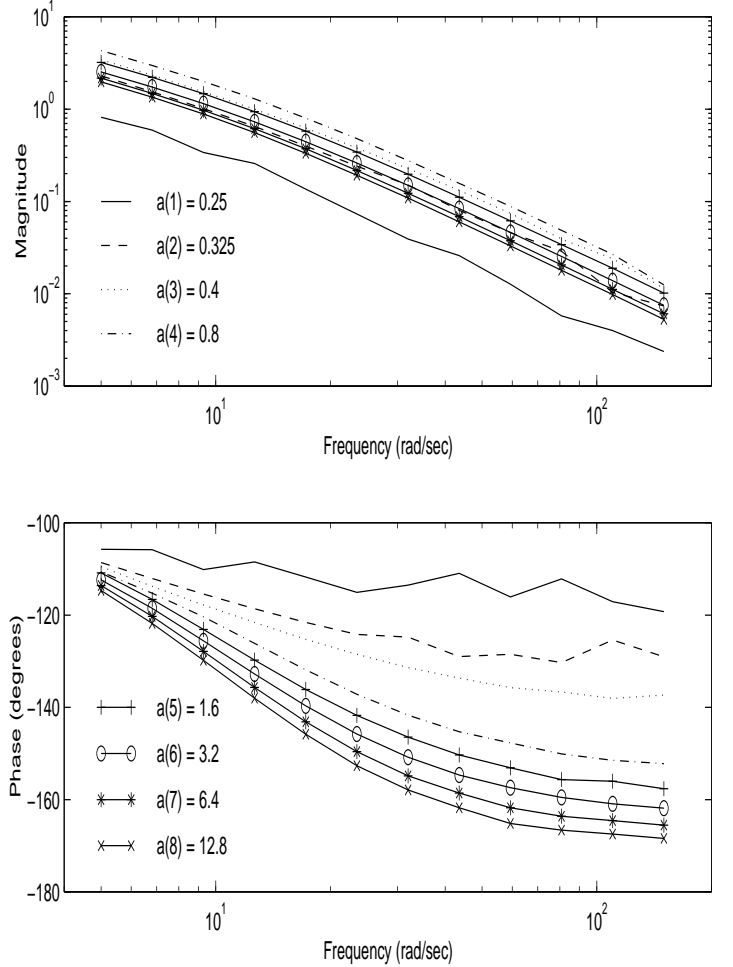

Figure 1: Motor + Load SIDF Models

Steps 3 and 6 involve the synthesis of the nonlinear inner-loop rate-feedback and PI compensators. The D'Azzo and Houpis algorithm is used for the rate-feedback gain, and the cascade PI design was done on the basis of SIDF I/O models of the plant with nonlinear rate feedback, since this represents the "plant" for the cascade design. A linear PI controller was designed at a nominal, mid-range, input amplitude, and the SIDF for the linear PI in cascade with the closed-inner-loop plant at that amplitude was generated and used as the objective function $C^\star G^\star_{rf}(j\omega)$. Using this nominal linear controller as a starting point, the parameters $K_P(a_i)$, $K_I(a_i)$ were fitted at each input amplitude to match $C^\star G^\star_{rf}(j\omega)$ as closely as possible. The resulting gain/amplitude relationships for $K_{T,i}(a_i)$, $K_{P,i}(a_i)$ and $K_{I,i}(a_i)$ are identical to those in [4], as shown in Table 1. Note that this design calls for slight *positive* rate feedback at low input amplitudes – this is due to the algorithm's objective which is to counteract the high effective damping provided by the stiction nonlinearity.

Rate Feedback		PI Controller		
e	K_T	e	K_P	K_I
0.4150	-0.3413	0.050	6.1266	34.0292
1.8091	-0.0385	0.065	5.2276	33.5021
3.5008	-0.0088	0.080	4.7781	32.8287
8.7778	0.0108	0.160	3.7593	30.0063
12.5293	0.0263	0.320	3.5064	28.8968
19.0920	0.0461	0.640	4.5711	22.5852
31.6697	0.0675	1.280	5.8645	8.0451
56.3283	0.0867	2.560	6.4135	0.0000

Table 1: Gain / Amplitude Relationships

The remaining steps in the fuzzy-logic controller design technique is taking the gain/amplitude information and generating the fuzzy-logic controller rulebases. This is illustrated for the proportional part of the fuzzy PI controller as follows: First, the rule was taken to have eight membership functions, PZ, NZ ("positive-zero, negative-zero"), PS, NS ("positive-small, negative-small"), PM, NM ("positive-medium, negative-medium") and PB, NB ("positive-big, negative-big"). The membership function "corners" and output levels were taken to be adjustable parameters defining the "fuzzy proportional nonlinearity" $f_{FL,P}$; they were determined by SIDF inversion, i.e., adjusting them to minimize the fitting error between the SIDF of $f_{FL,P}$ and the gain/amplitude data in Table 1; this resulted in the membership functions and nonlinear behavior depicted in Fig. 2. Note that the five-segment piece-wise-linear function exhibits very high gain for small signals, moderately low gain in the middle range, then a somewhat higher gain; ultimately the function saturates.
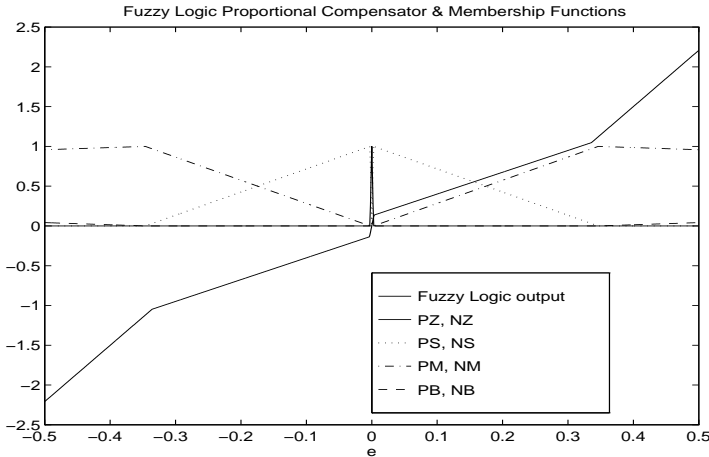

Figure 2: $f_{FL,P}$ and Membership Functions

The final stage of the design was simulation to validate the resulting closed-loop system. Two cases were studied: The nominal linear PID used in generating the

"frequency-domain objective function", and the nonlinear fuzzy-logic controller. The resulting time histories, with input amplitudes ranging from $a_1 = 0.20$ to $a_8 = 10.2$ in both cases, are shown in Fig. 3. Comparing these two sets of time histories, it is evident that the linear PID yields rather poor performance. (At small and large input amplitudes stiction and integral windup, respectively, tend to spoil the transient response.) The nonlinear fuzzy-logic controller, on the other hand, has produced a set of step responses that is remarkably insensitive to input amplitude. (The performance of the fuzzy-logic controller is slightly different from that shown in [4], although the nonlinear controller is functionally very similar, due to the fact that we are using a better numerical integration method [17, 18] which accurately and effectively manages the discontinuous behavior caused by stiction.)
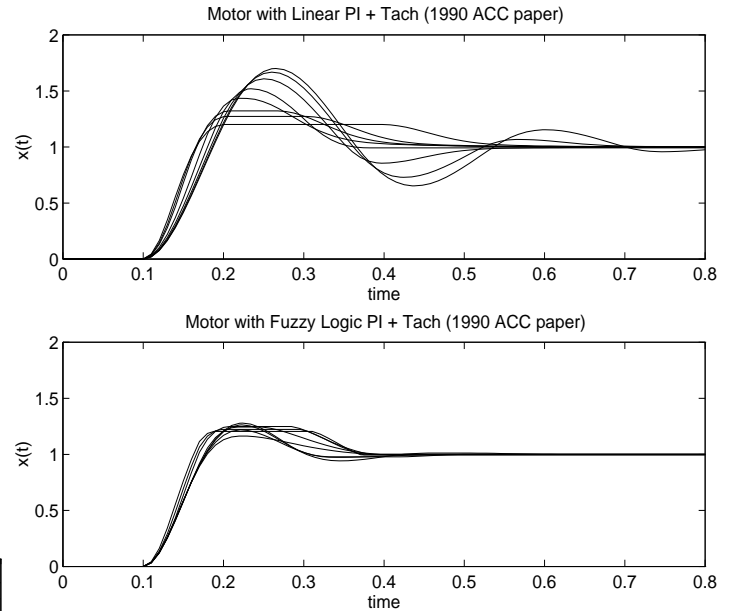

Figure 3: Linear and Nonlinear Fuzzy-logic Controller Step Responses

4 SUMMARY AND CONCLUSIONS

The method outlined in Section 2 and applied in Section 3 is a specific realization of the basic concept of using SIDF I/O models as the basis for nonlinear compensator design proposed in [1] and developed for the synthesis of several specific controller configurations [2] – [7]. The contribution of this paper is to extend these ideas to fill an important gap in fuzzy control, namely, the generation of fuzzy-logic controllers for plants that are too fast and/or too complicated for heuristic methods to be effective. Based on the example shown in Section 3, we feel that this model-based approach shows considerable promise in dealing with one of the more difficult problems in nonlinear systems design: the design

of controllers to correct for the amplitude-dependence of nonlinear plants.

This approach is capable of treating nonlinear plants of a very general type, with no restrictions as to system order, number of nonlinearities, configuration, or nonlinearity type. These results make the use of SIDF-based nonlinear controller design methods substantially more effective. It is also believed that this design approach will provide a framework for further developments in the realm of fuzzy-logic controller design for nonlinear systems.

References

[1] J. H. Taylor, "A Systematic Nonlinear Controller Design Approach Based on Quasilinear System Models", *Proc. American Control Conference,* San Francisco, CA, pp. 141-145, June 1983.

[2] 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, pp. 817-824, June 1984.

[3] J. H. Taylor and K. L. Strobel, "Nonlinear Compensator Synthesis via Sinusoidal-Input Describing Functions," *Proc. American Control Conference*, Boston MA, pp. 1242-1247, June 1985.

[4] J. H. Taylor and J. R. O'Donnell, "Synthesis of Nonlinear Controllers with Rate Feedback via SIDF Methods", *Proc. American Control Conference,* San Diego, CA, pp. 2217-2222, May 1990.

[5] J. H. Taylor, "A General Limit Cycle Analysis Method for Multivariable Systems", a chapter in *New Approaches to Nonlinear Problems in Dynamics,* pp. 521-529, Ed. by P. J. Holmes, SIAM (Society of Industrial and Applied Mathematics), 1980.

[6] J. H. Taylor, "Describing Function Methods for Analyzing High-Order Highly Nonlinear Systems", *Proc. Intl. Congress on Applied Systems Research and Cybernetics,* Acapulco, Mexico, December 1980.

[7] J. H. Taylor and J. Lu, "Robust Nonlinear Control System Synthesis Method for Electro-Mechanical Pointing Systems with Flexible Modes", *J. of Systems Engineering, Vol. 5* (special issue on motion control systems), pp. 192-204, January 1995.

[8] D. P. Atherton, *Nonlinear Control Engineering,* Van Nostrand Reinhold Co., London & New York; full edition 1975, student edition 1982.

[9] A. Gelb and W. E. Vander Velde, *Multiple-Input Describing Functions and Nonlinear System Design*, McGraw-Hill Book Co., New York, NY, 1968.

[10] R. M. Tong, "A Control Engineering Review of Fuzzy Systems", *Automatica, 13,* pp./559-569, 1977.

[11] M. Sugeno, "An Introductory Survey of Fuzzy Control", *Information Sciences, 36,*, pp. 59-83, 1985.

[12] H. R. Berenji, "Fuzzy Logic Controllers", in *An Introduction to Fuzzy Logic Applications and Intelligent Systems,* Ed. R. R. Yager and L. A. Zadeh, Kluwer Academic Publishing, Boston, MA, 1992.

[13] J. J. D'Azzo and C. H. Houpis, *Feedback Control System Analysis and Synthesis*, McGraw-Hill Book Co., New York, NY, 1960.

[14] J. H. Taylor, "Computer-aided Control Engineering Environment for Nonlinear Systems Analysis and Design", *Proc. Third IFAC Symposium on Computer-Aided Design in Control and Engineering Systems,* Lyngby, Denmark, pp. 38-43, August 1985.

[15] J. H. Taylor and J. Lu, "Computer-Aided Control Engineering Environment for the Synthesis of Nonlinear Control Systems," *Proc. American Control Conference*, San Francisco, CA, pp. 2557-2561, June 1993.

[16] G. J. Thaler, *Design of Feedback Systems*, Dowden, Hutchinson, and Ross, Inc., Stroudsburg, PA, 1973.

[17] J. H. Taylor and D. Kebede, "Modeling and Simulation of Hybrid Systems", *Proc. IEEE Conference on Decision and Control,* New Orleans, LA, December 1995.

[18] J. H. Taylor and D. Kebede, "Modeling and Simulation of Hybrid Systems in MATLAB", *Proc. IFAC World Congress,* San Francisco, California, July 1996.