Qualitative Robust Fuzzy Control with Applications to 1992 ACC Benchmark

Stephen Paul Linder, *Member, IEEE* and Bahram Shafai, *Member, IEEE*

Abstract--Robust control has long been the purview of quantitative linear control techniques, while qualitative, symbolic control has been deemed more suitable to obtaining complex control objectives that require only low output precision. The intelligent techniques of Fuzzy control have, however, shown promise in obtaining results comparable to those obtained from H_∞ and H_2 robust control techniques. Often though, these fuzzy control techniques ignore the original intent of fuzzy logic: implementation of symbolic, linguistic control laws based on qualitative models of the plant, and control behaviors. We will show that robust control objectives, even for simple plants, can be achieved by first developing qualitative behaviors that stabilize the plant, and then superimposing tracking behaviors that achieve control objectives. Specifically, by superimposing qualitative stability and tracking behaviors, we can achieve robustness and tracking stability comparable to the best published linear compensators for the 1992 ACC Robust Control Benchmark.

***Index terms*-- fuzzy control, robustness, stability, vibration control.**

I. Introduction

A control system should be designed to provide both stability and tracking performance not only for the nominal plant, but robustly over a range of possible plant configurations. Controller robustness is commonly benchmarked with the American Control Conference Robust Control Benchmark [22]. In over 30 papers, including papers in special issues of the *Journal of Guidance, Control and Dynamics* and the *International Journal of Robust and Nonlinear Control*, the robustness of various design methodologies has been evaluated with the Benchmark. These evaluated methodologies have exclusively used conventional quantitative techniques, while ignoring the promise of qualitative intelligent control techniques in developing robust controllers. Our work, however, extends the "weakly intelligent" techniques of fuzzy control [2, 20] to a design methodology for **robust fuzzy control** based on qualitative models and superimposed qualitative behaviors. Our methodology, Qualitative Robust Control (QRC), allows the designer to make the tradeoffs between the often contradictory goals of stability and tracking robustness by changing the relative weighting between qualitative behaviors that promote stability and those that promote tracking. Even the relative importance of these behaviors can be made to vary with the state of the plant and changing objectives.

Often, conventional quantitative robust control design methodologies make tradeoffs between the conflicting goals of stability and tracking by minimizing a robustness cost function that includes metrics that measure compensator performance, including settling time, actuator saturation, and stability. Published methodologies include H_∞ [25], H_2 [10], H_∞/H_2 hybrids [6], generalized LQG [4], nonlinear constraint optimization [16], LMI [8], loop shaping [23], and a hybrid design approach in which genetic algorithms optimize a stochastic robustness function [17]. All these conventional control techniques rely on a quantitative differential/difference equation model of the plant [2]. A set of controllers that stabilize the plant is defined, and then the set is searched for a controller that minimizes the cost function. Once the optimal controller is found, incremental tuning of the initial design is difficult. The reason for this difficulty is that compensator parameters do not necessarily have a linguistic interpretation to guide the refinement, and the designer does not know the precision that must be preserved in the individual parameter in order to preserve the specified performance.

Fuzzy control, however, lends itself to a different design methodology, in which a compensator is designed incrementally by superimposing qualitative behaviors that achieve the underlying performance goals. A controller is developed using the following steps:

1. rules are developed to realize localized qualitative behaviors,
2. global behavior caused by the interpolated localized rules is tested, and
3. behavior is refined by tuning localized behavior and superimposing additional localized and global behaviors.

Ruspini, Saffiotti and Konolige [21] used these steps to develop a controller with only 15 rules, representing 6 elemental behaviors that can navigate a simple maze. These rules reason about three sonar sensor outputs and direction and implement a "reactive navigation" behavior. Fuzzy logic achieves a consistent global behavior by interpolating and superimposing the elemental behaviors.

This methodology for designing fuzzy controllers transcends the quantitative methods that are utilized by the majority of fuzzy controllers described in [7, 12]. Instead of using these fuzzified versions of quantitative controllers, our methodology relies on qualitative models of the plant behavior. As in

Stephen Paul Linder was with Northeastern University, but is now at Applied Research Labs, Penn State, State College PA, spl@alum.mit.edu, and Bahram Shafai is at the CDSP Center, Northeastern University, Boston, MA, shafai@cdsp.neu.edu. This work was funded by school loans and a very generous grant from my parents; may dad rest in peace.

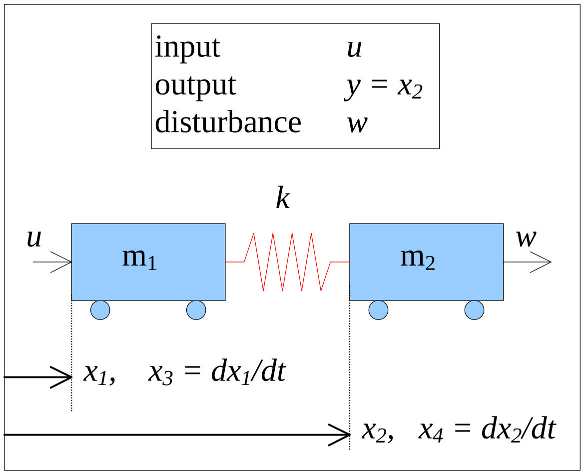


Figure 1. The dual mass, single spring plant used in the ACC benchmark.

conventional control where the designer must make tradeoffs between different performance metrics with the fuzzy controller, the designer must weigh the effects of different localized qualitative behaviors on the overall performance of the controller.

Our paper shows that fuzzy setpoint control can be expanded to provide robust setpoint control if a qualitative plant model that subsumes all specified plant perturbations is utilized in designing the controller. A rule-based fuzzy controller is then incrementally designed based on the "qualitatively robust" plant model. First, stability behaviors are developed and characterized. Then tracking behaviors are developed to augment the controller. Because these behaviors are based on a qualitatively robust model of the plant, the stability and tracking behaviors are robust over the extent of plant configurations that are subsumed by the qualitative plant model. The resulting fuzzy controller supports both stability and performance robustness and allows for simple compensator tuning by changing linguistically interpretable rule parameters.

The problem of tuning and precision is addressed by the use of linguistic rules. The rule-based fuzzy compensators can be augmented directly by the addition of new rules and tuned by adjusting parameters with clear linguistic interpretations. Initially, a compensator is developed to stabilize the plant and then incrementally augmented with rules until the final performance objectives are met. This robust control design process, based on a qualitative plant model, and the hierarchical development of stability and tracking behaviors is called **Qualitative Robust Control** (**QRC**).

The 1992 ACC Benchmark is used to benchmark our QRC compensator. By superimposing vibration suppression stability behaviors with tracking behaviors, we achieve stability robustness and tracking comparable to the best published compensators for the Benchmark. Two designs of the QRC compensator are developed: one stressing stability robustness and the other stressing tracking performance robustness. They are tested for both in a Full State Feedback (FSFB) and output feedback configuration. With output feedback, the QRC compensator is clearly more robust than a LQR with FSFB and comparable actuator usage. With only output feedback, the QRC compensator was used in conjunction with a robust linear state estimator and performed as well as or better than the best published compensators for the Benchmark.

This paper continues with a description of the Benchmark plant and a description of the two of the four Benchmark control scenarios considered in this paper. A brief overview of fuzzy control is given along with a short description of our QRC methodology. The QRC design process for the Benchmark compensator is described. Our two designs are described and then compared against two similarly tuned conventional controllers from Marrison and Stengel. Unlike in our previous results [14], we have successfully solved the Benchmark problem for reasonable levels of noise and have used a simple first-order actuator model.

II. Problem Formulation

This section describes the ACC Robust Control Benchmark and several published conventional compensators for the benchmark.

A. Benchmark Problem for Robust Control

At the 1990 American Control Conference (ACC) a set of three benchmark scenarios for robust control was suggested by Wie and Bernstein [28] and then augmented by a fourth scenario at the 1992 ACC [29]. These scenarios require the designer to make tradeoffs between maximizing stability and tracking robustness while minimizing actuator effort. This paper considers only Scenario 1 and Scenario 4; they are described after the Benchmark plant description.

The Benchmark plant shown in Figure 1 is a simple flexible structure consisting of two masses connected with a single spring. This dynamic system has a **noncollocated** sensor and actuator; the sensor senses the position of $\mathbf{m_2}$ while the actuator accelerates $\mathbf{m_1}$, making control of the plant difficult. The state space model for the plant is

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -k/m_1 & k/m_1 & 0 & 0 \\ k/m_2 & -k/m_2 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1/m_1 \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1/m_2 \end{bmatrix} w \qquad \textbf{(1)}$$

$$y = x_2$$

where

$\mathbf{x_1}$ is the position of $\mathbf{m_1}$,
$\mathbf{x_2}$ is the position of $\mathbf{m_2}$,
$\mathbf{x_3}$ is the velocity of $\mathbf{m_1}$,
$\mathbf{x_4}$ is the velocity of $\mathbf{m_2}$,
$\mathbf{y}$ is the plant output $\mathbf{x_2}$,
$\mathbf{w}$ is an acceleration disturbance on $\mathbf{m_2}$ and
$\mathbf{u}$ is the control acceleration on $\mathbf{m_1}$.

The transfer function representation is

$$T_{uy} = \frac{(k/m_1 m_2)}{s^2\left[s^2 + k(m_1+m_2)/m_1 m_2\right]} \quad \textbf{(2)}$$

and the corresponding transfer function between a disturbance to m_2 and plant output is

$$T_{wy} = \frac{(1/m_2)(s^2 + k/m_1)}{s^2\left[s^2 + k(m_1+m_2)/m_1 m_2\right]} \quad \textbf{(3)}$$

Control Scenario 1 has the following design requirements for the compensated system:

i. The closed-loop system is stable for $\mathbf{m_1} = \mathbf{m_2} = 1$ and $0.5 < \mathbf{k} < 2.0$.

ii. For the disturbance **w(t)** = unit impulse at **t** = 0, **y** has a settling time of 15 seconds for the nominal plant parameters $\mathbf{m_1} = \mathbf{m_2} = \mathbf{k} = 1$.

Control Scenario 4 requires the compensated plant to track a unit-step with the following design requirements:

i. The compensator output is limited to $|\boldsymbol{u}| \leq 1$.

ii. Settling time and overshoot are both minimized.

iii. Robustness to perturbation in $\mathbf{m_1}$, $\mathbf{m_2}$ and **k** with the nominal plant parameters being $\mathbf{m_1} = \mathbf{m_2} = \mathbf{k} = 1$.

Settling is achieved for both scenarios when **y** is bounded by ±0.1 units.

B. Conventional Compensators for the Benchmark

The simple structure of the benchmark initially suggests that it can be controlled by a conventional Proportional-Integral-Derivative (PID) compensator. The PID control law is appealing because the parameters of the transfer function

$$u(s) = K_P + \frac{K_I}{s} + K_D s \quad \textbf{(4)}$$

have simple linguistic interpretations. Indeed a PID controller can stabilize a mass-spring-mass plant with a **collocated** sensor and actuator. However, a PID compensator, conventional or fuzzy, can not stabilize the Benchmark; a more complex controller is required. An example of a more complex controller is Uy-Loi Ly's [16] 2^{nd}-order compensator:

$$T = \frac{5.3697s^2 - 1.3702s - 0.17375}{s^2 + 3.6767s + 4.8840} \quad \textbf{(5)}$$

Here the interpretation of parameters is no longer straight forward; interpretation can be given in terms of the zeros and poles of the system (e.g. the compensator models the plant spring as a simple oscillator with a frequency of 0.34813 rad/sec). Any further improvement in performance requires the use of a more complex fifth-order compensator [24].

This paper will benchmark our fuzzy controller against two higher order compensator developed by Marrison and Stengel using the linear quadratic Gaussian regulator methodology [18]. The compensator, **Comp1,** stresses stability robustness:

$$T = \frac{-79.3(s-0.8)(s+5.7)(s+0.11)}{(s^2+3.84s+10.24)(s^2+6.882s+13.69)(s+0.46)} \quad \textbf{(6)}$$

The compensator, **Comp3**, stresses settling time 9e.g. tracking) robustness:

$$T = \frac{-8.2(s-4.7)(s+3.9)(s+0.24)}{(s^2+4.662s+13.69)(s^2+3.132s+7.29)(s+1.6)} \quad \textbf{(7)}$$

These compensators contain numerous numerical constants that do not have linguistic interpretations. Tuning of the compensators is difficult since no clear coupling between individual parameters and individual design constraints exists, and the parameter precision required to achieve the specified performance is unknown. The next section of the paper describes rule based qualitative robust control where every constant has a clear linguistic interpretation and requires only low numerical precision. Tuning can be performed by either augmenting the design with additional rules or by adjusting the appropriate compensator parameters.

III. Robust Fuzzy Control

Fuzzy control has gained a wide practical acceptance, providing a simple, intuitive and qualitative methodology for control [11, 30, 31]. Currently, the typical implementation of a fuzzy controller consists of a set of **if-then** rules, where the controller input ***e*** is used to evaluate the rules' antecedents and the controller output ***u*** is the combined output of all the rules evaluated in parallel. This simple logical system, a **Fuzzy Inference System** (**FIS**) [9], does not implement inference chaining and can only evaluate a simplified qualitative model of a plant.

Fuzzy controllers described in the literature fall into three main categories [7, 12]:

1. fuzzy PID controllers,
2. fuzzy sliding mode controllers, and
3. fuzzy gain scheduling.

All three categories realize closed-loop control action and are based on quantitative control techniques. The fuzzy PID controllers and fuzzy slide mode controllers are fuzzy implementations of the linear quantitative PID controller and a nonlinear quantitative sliding mode controller. Both controllers use the error term, and its derivatives and integrals, as input into a FIS [3]. The fuzzy gain scheduler utilizes Sugeno fuzzy rules to interpolate between several control strategies [12]. This methodology is useful for controlling nonlinear plants that are piecewise linear or for linear plants that have a time varying parameter. An example of this is a controller built for an inverted pendulum with variable length. Measurements of the pendulum length are used as inputs into a FIS that interpolates the output of a small number of controllers that are optimized for controlling short, medium and long pendulums [9].

In addition to these compensators which realize closed-loop control action, a FIS can be used as a supervisor of a conventional closed-loop controller, including systems where a FIS adaptively tunes a closed-loop PID controller [5].

Our Qualitative Robust Control (QRC) methodology uses an incremental design approach, based on the work of [21], to implement robust fuzzy controllers. First, rules are developed to implement a global stabilizing behavior and are then augmented with behaviors that implement performance goals. Additionally, the compensator achieves robustness by being designed for a sound**,** or consistent, qualitative plant model that encompasses all the specified plant perturbations. If the rules developed for the compensator are consistent with the plant model, then the resulting compensator will provide

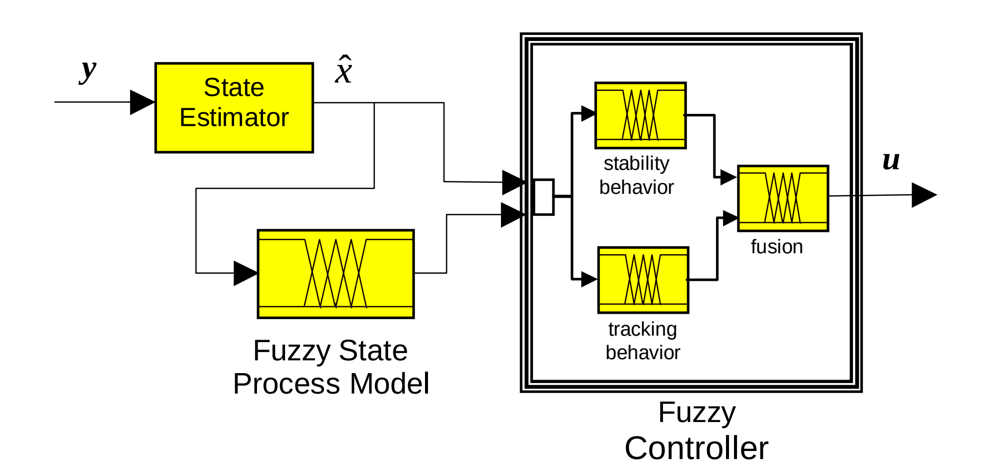


Figure 2. Fuzzy Controller consisting of a linear state estimator, fuzzy process model and fuzzy compensator which superimposes stability and tracking behaviors.

robust control. The QRC methodology condenses to a four step procedure:

1. creation of a qualitative plant model
2. creation of a stabilizing compensator
3. augmentation of the initial compensator design with behaviors that achieve robust performance, and
4. tuning of the linguistically interpretable compensator parameters.

At the moment, the QRC methodology is only a guide to developing a controller. The authors are working on making the methodology more rigorous and supporting automatic generation of rules. The following sections provide additional details.

A. Selecting a Proper Qualitative Plant Model

One of the strengths of intelligent control is that it is **not** based on a quantitative model of the plant, but rather a simplified qualitative model. Model resolution is sacrificed so that the qualitative model contains only a finite number of interpretable states and state transitions. Each qualitative state is based on a finite partition of the plant parameters; exact numerical values of plant parameters are abstracted to interpretable qualitative terms.

This qualitative partition of plant states and parameters has the benefit of making the qualitative model more robust to changes in plant parameters. Qualitative abstraction enhances stability robustness because the qualitative model of the plant does not rely on the exact values of the plant parameters, but on a finite qualitative partitioning of the plant's states and parameters. Abstracting away details of the plant reduces the complexity of the resulting fuzzy controller, but if too much detail is abstracted away the qualitative model will no longer be complete and consistent with respect to the control requirements; no stabilizing controller can be developed.

B. Designing a Fuzzy Controller

The fuzzy controller design begins with the qualitative plant model. First state information must be extracted from the plant output by using a combination of linear methods and fuzzy process models. The crisp outputs of linear operators and fuzzy process models are used as input to a Sugeno FIS controller (see Figure 2). The design of the rules for the FIS are suggested by the plant's qualitative state transition diagram. Given the connection between qualitative input events and changes in qualitative states, as shown by the plant's state transition diagram, the FIS controller is constructed to generate the qualitative events that will result in the qualitative state transitions required to realize the desired control actions.

The first control objective is the stabilization of the plant. Stability for the Benchmark entails the dampening of vibrations after an external disturbance is applied. After stabilization, the FIS is augmented with rules to achieve performance objectives. Fuzzy logic lends itself to this methodology, because fuzzy logic deals with possibilities, and reasoning remains consistent even when conflicting requirements generate conflicting rules. Further tuning of the composite compensator can be made by adjusting the relative weighting of the outputs of the different behaviors and by gating behaviors, allowing them to be performed only during specified states.

IV. Robust Fuzzy Controller for Benchmark

The Benchmark problem requires the achievement both of stability and tracking performance. The physical intuition is that the spring oscillations caused by disturbances must first be dampened to achieve stability, and then after stability is achieved the goal of tracking can become paramount. This nonlinear dichotomy is easily supported by fuzzy logic.

QRC allows the separate development of stability and tracking behaviors; the superimposition of these behaviors then achieves the final control objective. An analysis of the transfer function of the mass-spring-mass plant indicates that these behaviors should exploit the rigid-body-mode of the plant, where the plant behaves as if the masses are rigidly connected. If the stability behavior can be made to achieve this rigid-body-mode, then the tracking behavior can treat the mass-spring-mass system as a simple single mass. This allows a simpler tracking behavior to be effective.

The stability behavior is derived from the heuristic that a control action is most effective in suppressing plant vibration if it is applied when the spring is neutral, and the control action opposes the motion of the spring. As an extreme example of this effect, an impulse to a stationary plant can be rejected with just one complementary impulse of equal magnitude, if the impulse occurs exactly as the spring relaxes.

This section continues with the development of a fuzzy process model that provides a qualitative partition of the spring state necessary to suppress vibration. Next, the ramification of this spring process state on the estimation of the plant is discussed. The Stability Behavior that suppresses vibration is then described. Finally, we present the Tracking Behaviors and discuss the superimposition of the stability and tracking behaviors.

A. Fuzzy Spring Process Model

A fuzzy process model of the spring needs to provide the qualitative state information necessary to dampen the vibrations of the plant and achieve stability. This can be achieved by abstracting the quantitative state of the Benchmark plant to just one qualitative state that indicates whether the spring is at its neutral length and whether the spring is in the process of compressing or elongating. This fuzzy process model requires that any estimate provided by a linear filter of the plant's quantitative state effectively captures

1. the timing of the spring relaxation and
2. the direction of motion of the spring (e.g. compressing and stretching).

Because the fuzzy process model only requires accurate estimates of state with respect to these two metrics, the linear filter does not need be optimal (e.g. Kalman filter), but can be a sub-optimal filter derived with robust filter methodologies.

The fuzzy process model utilizes a qualitative spring state that is specified by a qualitative partition of the spring length, $L = x_2 - x_1$, and the spring length velocity, $\dot{L} = \dot{x}_2 - \dot{x}_1$. A Mamdani Fuzzy Inference System (FIS) applies seventeen rules, shown in Table 1, to infer the qualitative spring state, Q_L from the inputs L and $\dot{L}$. The output Q_L is partitioned into the following qualitative states mapped with triangular and trapezoidal membership functions to the interval [-1, 1]:

1. spring is neutral and compressing rapidly,
2. spring is neutral and compressing ,
3. spring is neutral and not in State 1, 2, 4 or 5,
4. spring is neutral and stretching, and
5. spring is neutral and stretching.

The input and output membership functions for the process model are shown graphically in Figure 3, where L, $\dot{L}$ and Q_L are partitioned by five membership functions. The stability behavior is enabled during output states 1, 2, 4 and 5; these states indicate the spring is vibrating and transitioning through `zero`. Noise immunity was improved by requiring that when L is not `zero`, but rather `small_positive` or `small_negative`, that the level of $\dot{L}$ be large, either `negative` or `positive`, before Q_L is set to a state indicate a vibration. Only for `zero` values of L will `small_positive` or `small_negative` of $\dot{L}$ result in an output Q_L which indicates that a vibration needs to be suppressed. This is shown clearly in the decision surface shown in Figure 4.

B. Estimating Plant State

Previous versions of this fuzzy controller used the acceleration and jerk of $\boldsymbol{m_2}$ as estimates of $\dot{L}$ and Q_L [14]. This method works extremely well, but fails when measurement noise is present. The differentiation which is required to obtain the acceleration and jerk amplifies any high frequency noise associated with the position measurement. In order to promote noise immunity the current controller implementation utilizes a model based filter.

Ideally, in an attempt to maintain technological consistency, a filter based on a fuzzy model of the plant should be used. However, existing fuzzy estimators are hybrid systems, with a Tanaka observer using fuzzy reasoning about quantitative linear plant models to infer a quantitative state [27]. Until a full fuzzy filter is developed the state must be estimated by a linear estimator. Consider a linear filter for the dynamic system of order n described by the following equations

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{Bu} \\ \mathbf{y} &= \mathbf{Cx}\end{aligned} \tag{8}$$

where $\boldsymbol{u}(t)$ are the p inputs and $\boldsymbol{y}(t)$ the m outputs, $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ are $(n \times n)$, $(n \times p)$ and $(m \times n)$ matrices respectively. The filter has the form

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{L}(\mathbf{y} - \mathbf{C}\hat{\mathbf{x}}) + \mathbf{Bu} \tag{9}$$

where $\hat{\mathbf{x}}$ is the estimate of the state $\boldsymbol{x}$. If the system ($\mathbf{A}$, $\mathbf{C}$) is observable then the constant $\mathbf{L}$ can be selected so that ($\mathbf{A}$ - $\mathbf{LC}$) is asymptotically stable and $\hat{\mathbf{x}}(t)$ asymptotically approaches $\mathbf{x}(t)$. A Kalman filter uses $\mathbf{L}$ that minimizes the mean square error of the filter whereas H_∞ filters use $\mathbf{L}$ which minimizes the supremum of the estimation error. However, we use a form of the common Proportional-Integral (PI) [14, 19] Kalman filter, the Proportional Fading-Integral (PFI) Kalman filter [13, 15], where the integral action allows the filter to estimate and robustly reject plant perturbation. The PFI Kalman filter is described by

$$\begin{aligned}\dot{\hat{\mathbf{x}}} &= \mathbf{A}\hat{\mathbf{x}} + \mathbf{L}(\mathbf{y} - \mathbf{C}\hat{\mathbf{x}}) + \mathbf{Bu} + \mathbf{B_I v} \\ \dot{\mathbf{v}} &= \mathbf{K_I}(\mathbf{y} - \mathbf{C}\hat{\mathbf{x}}) + \mathbf{K_F v}\end{aligned} \tag{10}$$

where the constant $\mathbf{K_I}$ determines the time constant of the integral action while $\mathbf{B_I}$ reflects the form of the plant perturbations or disturbance injection points.

The structure of the spring process model allows the use of a linear estimator that is non-optimal with respect to measurement noise, so long as the estimator can predict accurately the times when the spring is relaxed and the direction of vibration at these periods of spring relaxation. Because of this relaxation in performance, it is easier to find a combination $\mathbf{L}$, $\mathbf{B_I}$, $\mathbf{K_F}$ and $\mathbf{K_I}$ which performs robustly in conjunction with the spring process model over a wide range of spring constants. Simula-

Table 1. Rules for determining spring state.

If (spring_length_estimate **is** negative) **then** (spring_state **is** not_stretching_or_compressing_with_neutral_spring)
If (spring_length_estimate **is** small_negative) **and** (delta_spring_length_estimate **is** negative) **then** (spring_state **is** compressing_fast_with_neutral_spring) **If** (spring_length_estimate **is** small_negative) **and** (delta_spring_length_estimate **is** sm_neg) **then** (spring_state **is** not_stretching_or_compressing_with_neutral_spring) **If** (spring_length_estimate **is** small_negative) **and** (delta_spring_length_estimate **is** zero) **then** (spring_state **is** not_stretching_or_compressing_with_neutral_spring) **If** (spring_length_estimate **is** small_negative) **and** (delta_spring_length_estimate **is** sm_pos) **then** (spring_state **is** not_stretching_or_compressing_with_neutral_spring) **If** (spring_length_estimate **is** small_negative) **and** (delta_spring_length_estimate **is** positive) **then** (spring_state **is** stretching_fast_with_neutral_spring)
If (spring_length_estimate **is** zero) **and** (delta_spring_length_estimate **is** negative) **then** (spring_state **is** compressing_fast_with_neutral_spring) **If** (spring_length_estimate **is** zero) **and** (delta_spring_length_estimate **is** sm_neg) **then** (spring_state **is** compressing_fast_with_neutral_spring) **If** (spring_length_estimate **is** zero) **and** (delta_spring_length_estimate **is** zero) **then** (spring_state **is** not_stretching_or_compressing_with_neutral_spring) **If** (spring_length_estimate **is** zero) **and** (delta_spring_length_estimate **is** sm_pos) **then** (spring_state **is** stretching_fast_with_neutral_spring) **If** (spring_length_estimate **is** zero) **and** (delta_spring_length_estimate **is** positive) **then** (spring_state **is** stretching_fast_with_neutral_spring)
If (spring_length_estimate **is** small_positive) **and** (delta_spring_length_estimate **is** negative) **then** (spring_state **is** compressing_fast_with_neutral_spring) **If** (spring_length_estimate **is** small_positive) **and** (delta_spring_length_estimate **is** sm_neg) **then** (spring_state **is** not_stretching_or_compressing_with_neutral_spring) **If** (spring_length_estimate **is** small_positive) **and** (delta_spring_length_estimate **is** zero) **then** (spring_state **is** not_stretching_or_compressing_with_neutral_spring) **If** (spring_length_estimate **is** small_positive) **and** (delta_spring_length_estimate **is** sm_pos) **then** (spring_state **is** not_stretching_or_compressing_with_neutral_spring) **If** (spring_length_estimate **is** small_positive) **and** (delta_spring_length_estimate **is** positive) **then** (spring_state **is** stretching_fast_with_neutral_spring)
If (spring_length_estimate **is** positive) **then** (spring_state **is** not_stretching_or_compressing_with_neutral_spring)

tions show that the use of the robust PFI Kalman filter, rather than the standard P Kalman filter, doubles the stability range of the QRC fuzzy compensator.

C. Fuzzy Compensator

The fuzzy compensator integrates both the stability and tracking behavior. First, a single Stability Behavior was developed and qualified, and then two tracking behaviors were developed. As when designing linear controllers, a tradeoff is required between maintaining the stability robustness of the Stability Behavior and increased tracking performance. Tracking Behavior A limits its control action to when the spring length is small, minimizing the interference with the Stability behavior, while Tracking Behavior B adds additional rules to improve the settling time performance, but at the expense of reducing stability robustness.

The Stability Behavior requires five rules while Tracking Behavior A required an additional three rules and Tracking Behavior B requires five rules. These rules used a combination of six (seven when Tracking Behavior B) Sugeno output functions [26] where the output functions are a linear combination of the position and velocity inputs to the FIS.

1) Stability Behavior

Stability behavior is realized with 5 fuzzy rules that map the 5 fuzzy partitions of Q_L to five output membership functions. These rules are shown at the beginning of Table 4. Vibrations are suppressed using the following five output functions:

`big_stop_spring_stretching`	$y = b_{pos}^{big} x + b_{vel}^{big} \dot{x} + b_{accel}^{big}$	
`small_stop_spring_stretching`	$y = b_{accel}^{small}$	
`zero`	$y = 0$	**(11)**
`small_stop_spring_compressing`	$y_{sssc} = b_{accel}^{small}$	
`big_stop_spring_compressing`	$y_{bssc} = b_{pos}^{big} x + b_{vel}^{big} \dot{x} - b_{accel}^{big}$	

These output equations use four constants whose values are given in Table 2. Actual vibration suppression is achieved only by the bias terms b_{accel}^{big} and b_{accel}^{small}. These terms generate a pulse which opposes the stretching or compression of the spring. The remaining terms bias the amplitude of the pulse, so that the pulse aids in zeroing both the position and velocity of the masses.

2) Tracking Behavior

Tracking behavior has the conflicting goal of settling the plant output so that $|y| < 0.1$ after 15 seconds and interfering as little as possible with the stability robustness provided by the stability behavior. Two versions of the tracking behavior were developed: Tracking Behavior A which attempts to minimizes any detrimental interaction with the Stability

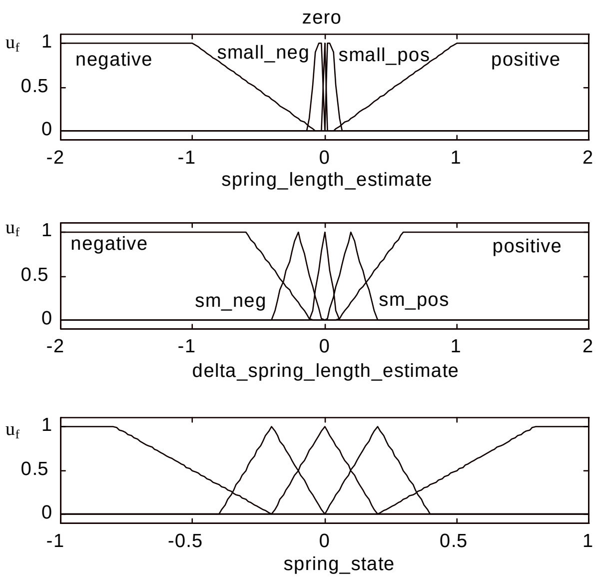


Figure 3. Input and output membership function of the Spring Observer.

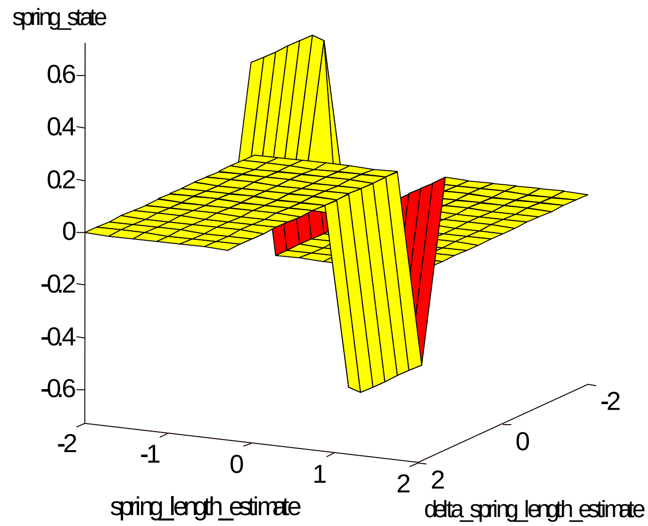


Figure 4. Output surface of the Spring Fuzzy Process Model .

Behavior and Version B which sacrifices stability, but attempts to minimize the peak overshoot and settling time for y for the nominal plant with $k = 1$.

Tracking Behavior A is implemented with Rule 7 through 9, from Table 4. Behavior A uses these three rules to implement a Sugeno Fuzzy PD controller which becomes active only when the spring length becomes small. This gating of the tracking behavior minimizes any detrimental affect the tracking behavior may have on the stability behavior. A smooth gating is obtained by using a smooth bell curve, the `tinyBell` membership function, instead of a simpler triangle membership function. The width of `tinyBell` at half-height is smaller than the equivalent `zero` membership function used by the spring process model, but unlike a triangle membership function decays smoothly to zero.

Tracking Behavior B was designed so that when there are large position errors the tracking behavior is superimposed directly on the stability behavior, with the assumption that stability and vibration suppression are less important for large errors. This modification of Behavior A requires an additional two rules that are activated when y is large. These additional two rules are used to increase the PD controller gains when the output error is large, irrespective of whether the spring length is small, but with the detrimental effect of reducing the efficacy of the stability behavior.

A Sugeno PD controller was used because it has the advantage of providing a more compact representation than an equivalent Mamdani fuzzy PD controller. The controller was implemented with an output membership function which consists of a linear combination of the position and velocity of m_2. Tracking Behavior A uses the single output membership function

`zero_small_position` $$y_{sssc} = p_{small}x + d_{small}\dot{x} \tag{12}$$

while Tracking Behavior B uses the additional output membership function

`zero_large_position` $$y_{sssc} = p_{large}x + d_{large}\dot{x} \tag{13}$$

Table 3 gives the values of the constants used in the output equation for both Tracking Behavior A and B.

V. Simulation Results

The performance of our QRC fuzzy controller was investigated using computer simulations for two scenarios: when complete state information was available and when a state observer was required to estimate the plant state. The QRC controller using Full State Feedback (FSFB) was benchmarked against the Linear Quadratic Regulator (LQR). The LQR was selected because it shows the optimal robustness to plant perturbations of any linear controller [1]. Equivalently, the QRC controller using output feedback and a robust Kalman Filter to estimate state was compared to the H_2 compensators **Comp1** and **Comp3.** The **Comp1** design stresses stability robustness while **Comp3** stresses performance robustness.

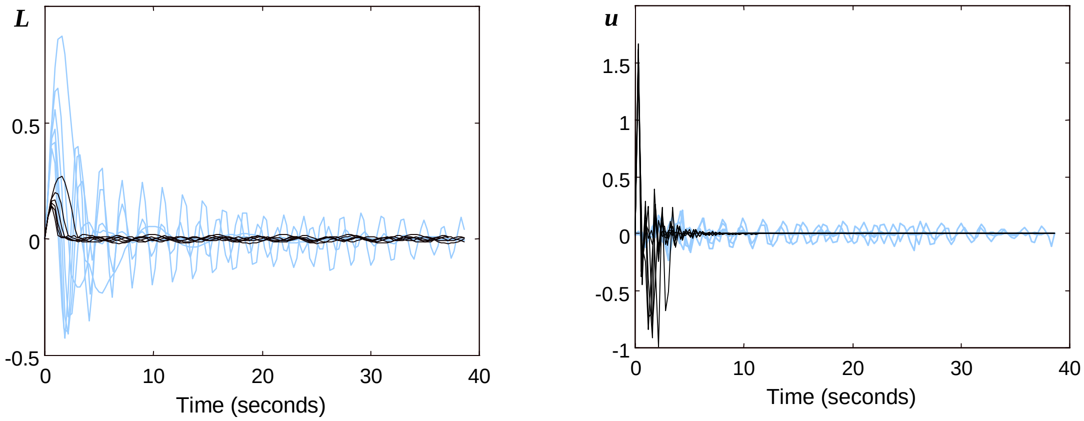
The compensators were evaluated for their ability to reject

Table 2. Coefficients for output equations used by vibration suppression behavior.

b^{big}_{pos}	b^{big}_{vel}	b^{big}_{accel}	b^{small}_{accel}
-1	-3	8	-0.6

Table 3. Coefficients for output equations used by Tracking Behavior A and B.

	p_{small}	d_{small}	p_{large}	d_{large}
Behavior A	-0.4	-2.1	—	—
Behavior B	-0.25	-2.5	-0.75	-1.2



(a) **(b)**

Figure 5. Performance is compared between the stability behavior using FSFB (black) and state estimates derived from a PFI Kalman filter (gray). The spring length L (a) and Actuator output u (b) are shown after a unit impulse disturbance to m_2 for k = 0.5 to 3.0 in steps of 0.5.

an impulse disturbance to m_2 using several metrics. $\boldsymbol{t}_{|L| \leq 0.05}$, the time it takes L to settle within ±0.05 units of the final value, is used to measure the stability of a compensated plant. The effect of vibration suppression on stability robustness was evaluated by comparing the range of spring constants k for which L settles within ±0.05 units of the final value in less than 15 seconds to the stability radius of the compensated plant. Tracking performance was measured by comparing the metric $\boldsymbol{y}_{max}$, the maximum value of the plant output, and $\boldsymbol{t}_{|y| \leq 0.1}$, the settling time of plant output y to within ±0.1 units of the final value. Since stability robustness and performance is enhanced by increased compensator output, $\boldsymbol{\Sigma}_{u15.0}$, the total actuator output for the first 15 seconds of the simulation was measured to insure that comparable levels of effort were used by all compensators.

All measurements, of both plant outputs and states, were corrupted with zero mean Gaussian noise with a period of 0.01 seconds and a standard deviation of 0.02. The LQR and LQG controllers use the same LQR gain matrix $\mathbf{K}_{\mathbf{LQR}}$ = [0.8997 3.6077 0.5634 6.5076]'. Note that full loop transfer recovery can not be achieved when designing a LQG for the Benchmark because the Plant's first Markov parameter, $\mathbf{C}\cdot\mathbf{B} = 0$ [22]. The PFI Kalman filters used a model based on the nominal plant, with $m_1 = m_2 = k = 1$ and process and measurement noise covariance matrices

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \text{and } V = 0.01, \qquad (14)$$

which gives a Kalman gain $\mathbf{K}$ = [41.3051 6.4508 29.1938 20.8062]. Integral action, which compensates for spring constant perturbations, used a distribution matrix $\mathbf{B_I}$ = [0 0 -1 1]' and an integral gain $K_I = -20$.

Both the Kalman gain and the LQR gain were calculated with the standard Matlab functions. All the fuzzy compensator configurations modeled the actuator as first order system with T = 40/(s+40). These simulation were performed with Simulink®.

A. Fuzzy Stability Behavior

The robustness of the stability behavior used in the full fuzzy controller was characterized for both FSFB and output feedback. Figure 5 shows the response to a unit impulse

Table 4. Fuzzy Rules for Controlling Plant

Rules to supress vibrations

1. **If** (spring state **is** compressing_fast_with_neutral_spring) **then** (control_output **is** big_stop_spring_stretching)
2. **If** (spring state **is** compressing_slowly_with_neutral_spring) **then** (control_output **is** small_stop_spring_stretching)
3. **If** (spring state **is** not_stretching_or_compressing_with_neutral_spring) **then** (control_output **is** zero)
4. **If** (spring state **is** stretching_slowly_at_zero_accel) **then** (control_output **is** small_stop_ spring_compressing)
5. **If** (spring state **is** stretching_fast_with_neutral_spring) **then** (control_output **is** big_stop_ spring_compressing)

Additional rules to achieve tracking of $\mathbf{m_2}$.

6. **If** (position_error **is** BigNegative) **then** (control_output **is** zero_large_position)
7. **If** (position_error **is** negative) and (spring_length is tinyBell) **then** (control_output is zero_small_position)
8. **If** (position_error **is** zero) and (velocity **is** zero) **then** (control_output **is** zero)
9. **If** (position_error is positive) and (spring_length is tinyBell) **then** (control_output **is** zero_small_position)
10. **If** (position_error **is** BigPositive) **then** (control_output **is** zero_large_position)

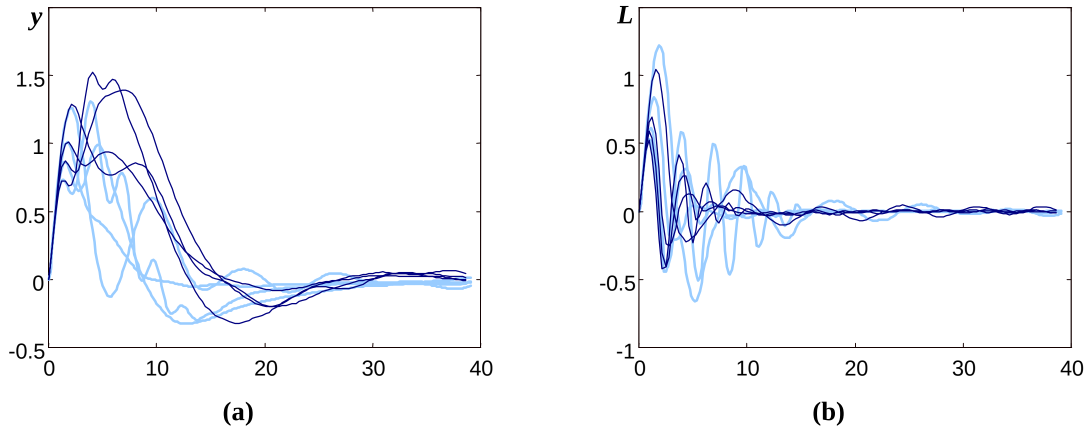


Figure 6. Comparison of the two tracking behaviors using only output feedback, where the black lines are from the more robust Tracking Behavior A and the gray lines are from the better performing Tracking Behavior B . The plant output *y* (a) and spring length *L* (b) are shown after a unit impulse disturbance to m_2 for k = 0.6 to 2.0 in steps of 0.4.

disturbance to m_2 for a range of spring constants: $k = 0.5$ to 3.0 in steps of 0.5. Both compensator configurations show excellent vibration suppression properties. The range of spring constants for which $|L| < 0.05$ after 15 seconds is $0.1 \le k \le 1000$ for FSFB and $0.4 \le k \le 2.3$ for the output feedback configuration. As expected, FSFB suppresses vibrations faster, over a wider range of plant perturbations and with less compensator effort, than output feedback. The fuzzy stability behavior is so effective in suppressing vibrations that even the output feedback configuration suppresses vibrations over a wider range of spring constants than a LQR using FSFB with equivalent actuator effort; the LQR regulator settles *L* only in the range $0.5 \le k \le 1.6$.

B. *Effect of Tracking Behavior on Stability*

The addition of tracking behavior reduces the stability robustness of the compensator. However, if designed correctly a compensator with the less evasive Tracking Behavior A should be more robust than the compensator incorporating Tracking Behavior B. The effect of these two tracking behaviors is compared for the output feedback case in Figure 6. While Behavior A shows higher peak responses and longer settling times, it is faster and more robust in settling *L*. This corresponds to a 50% larger stability radius for Behavior A than Behavior B.

The Stability Behavior settles *L* within 15 seconds to $|L| \le 0.05$ for $0.4 \le k \le 2.3$. The addition of Tracking Behavior A only marginally reduces the range of *k* to $0.7 \le k \le 2.3$, where as the performance oriented Tracking Behavior B reduces the range even further to $0.8 \le k \le 2.0$.

C. *Fuzzy Control with Full State Feedback*

The stability robustness and tracking performance of the full QRC controller with Tracking Behavior B (**QRC B**) using FSFB was compared to LQR. Figure 7 superimposes the output of these two controllers for a range of spring constants after a unit impulse disturbance to m_2. In order to insure a fair comparison, the LQR gain $\mathbf{K_{LQR}}$ was selected so that the peak LQR output was about the same as for the fuzzy controller; in fact, Figure 7 (c) shows that in the range $1 \le k \le 4$ the LQR produces a larger peak compensator output.

While the LQR over this range of *k* suppresses the effect of disturbance on the plant output *y* more quickly, shown in Figure 7 (a), Figure 7 (b) shows that the LQR compensated Plant has significantly larger oscillations in *L*. The superior vibration suppression properties of the QRC compensator contributes to the significantly larger stability margin of the QRC compensator: $0.2 \le k \le 1000$ for the **QRC A**, $0.4 \le k \le 1000$ for the **QRC B**, versus $1 \le k \le 4$ for the LQR. Figure 7 (d) shows that the superior performance of the QRC compensators is achieved while having a total compensator effort that is significantly smaller than that for the LQR controller, except when $k = 0.5$.

D. *Fuzzy Control with Output Feedback*

The stability robustness and tracking performance of the full QRC controller with output feedback was evaluated using a PFI Kalman filter to estimate plant states. The QRC controller performance was compared to LQG using an identical PFI Kalman filter and to the two Marrison and Stengel Compensators: **Comp1** given in equation (6) and **Comp2** given in equation (7) .

The stability robustness of the two QRC controllers proves to be far superior to that of the PFI Kalman filter-based LQG, but less robust than **Comp1** and **Comp3**. The QRC A is stable for $0.1 \le k \le 3.0$ and QRC B is stable for $0.5 \le k \le 2.1$, while the LQG is stable for $0.8 \le k \le 1.4$ and **Comp1** is stable for $0.5 \le k \le 5.5$. However **QRC B** has superior tracking performance than **Comp1** in the range of spring constants specified by the Benchmark problem, $0.5 \le k \le 2.0$. As shown in Figure 8 (a) **QRC B** has smaller peak output when perturbed by an impulse to m_2, and as shown Figure 8 (d) has smaller overshoot when tracking a unit step. Additionally, Figure 8 (b) and (c) show that **QRC B** has superior tracking performance when m_1 and m_2 are perturbed. When comparing metrics for the **QRC A**, **QRC B**, **Comp1** and **Comp3** compensators at $k =$

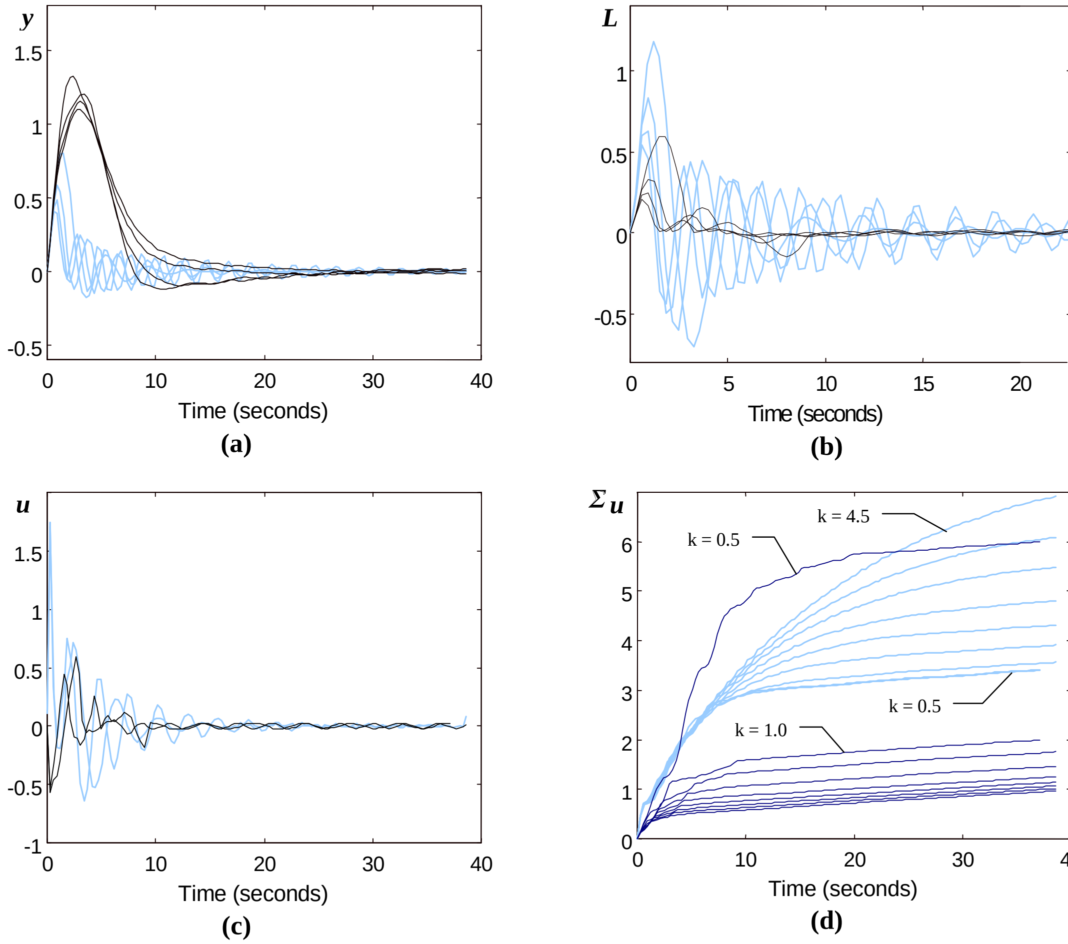


Figure 7. A comparison of the impulse rejection performance of two full state feedback controllers, the fuzzy controller (black lines) and a conventional Linear Quadratic Regulator (gray lines), for a unit impulse disturbance to m_2 with k = 1, 2, 3, 4. Figure (a) shows plant output y, (b) the spring length L, (c) the compensator output u, and (d) the cumulative compensator output Σu (for k = 0.5 to 4.5 in steps of 0.5).

0.5, 1.0 and 2.0, the QRC compensators show better vibration suppression behavior and comparable compensator output. The following tabulates simulation results for an impulse disturbance on $\boldsymbol{m_2}$:

	Comp1			**Fuzzy Controller A**		
k	**0.5**	**1.0**	**2.0**	**0.5**	**1.0**	**2.0**
t|y|≤	26.9	14.4	14.3	25.3	15.0	24.2
t|L|≤0.	>40	14.5	7.5	26.1	6.7	8.8
$\mathbf{y_{max}}$	1.8	2.1	2.0	1.45	1.05	1.58
$\mathbf{\Sigma_{u15.0}}$	2.6	2.8	2.7	3.1	2.2	3.9

	Comp3			**Fuzzy Controller B**		
k	**0.5**	**1.0**	**2.0**	**0.5**	**1.0**	**2.0**
t|y|≤	∞[1]	10.3	10.2	>40	8.0	19.4
t|L|≤		33.4	9.5	>40	8.8	14.9
$\mathbf{y_{max}}$		0.87	1.17	1.38	1.02	1.33
$\mathbf{\Sigma_{u15.0}}$		2.3	2.8	5.71	2.2	4.96

The table cells with the best results for the nominal plant, k = 1, are highlighted. **QRC B** has the best settling time for x_2, but takes longer to dampen vibrations in the flexible structure than **QRC A**. **Comp3** has the smallest peak, but takes longer to settle than **QRC B** and is not stable in the range $0.5 \le k \le 2.0$ specified in the Benchmark problem**.** Figure 9 compares the

[1] k = 0.95 gives a settling time of 19.4 seconds.

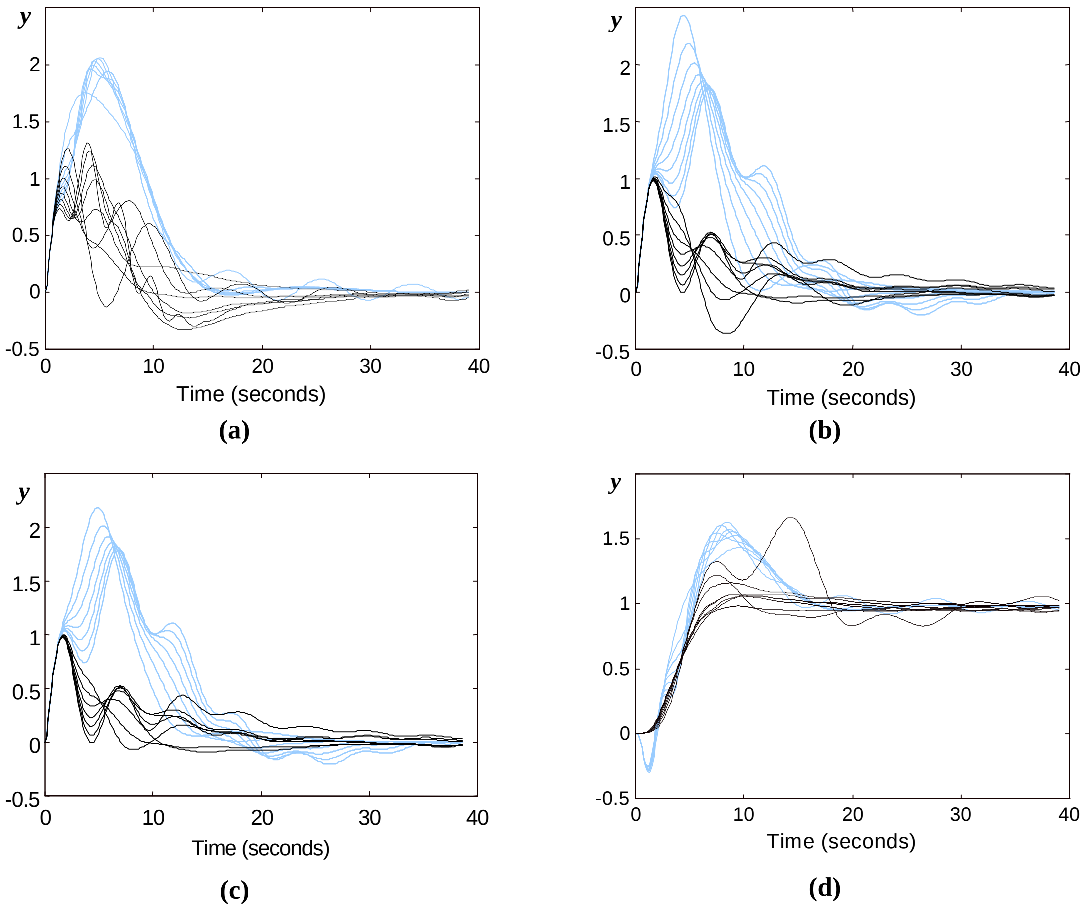


Figure 8. Performance comparison of fuzzy controller using state estimates from a PFI Kalman filter (black lines) and Comp1, a 5th order H^2 compensator from Marrison and Stengel (gray lines). Figure (a) shows the output response to a unit impulse disturbance to m_2 for k = 0.6 to k = 2.0 in steps of 0.2, (b) for m_1 = 0.6 to m_1 = 2.0 in steps of 0.2, and (c) for m_2 = 0.8 to m_2 = 2.0 in steps of 0.2. Figure (d) shows the tracking of a unit step command for k = 0.6 to k = 2.0 in steps of 0.2.

nominal plant($m_1 = m_2 = k = 1$) output y and the spring length L response to a unit impulse disturbance to m_2.

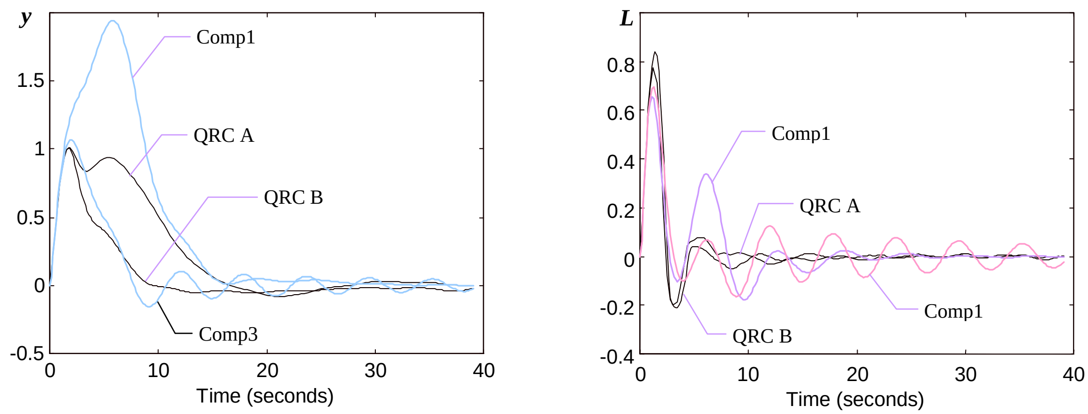
E. Stability Robustness Comparison

Stability robustness varies widely for the compensators evaluated in this paper, as shown graphically in **Figure 10**. The stability margins are given by the range of spring constants for which the compensated plants are stable. The best stability margins are offered by the fuzzy controllers with FSFB while the worse stability margin is for the PFI Kalman filter-based LQG. **Figure 10** also shows that stability margins correspond closely to the range of spring constants for which L settles quickly.

The limited robustness of the PFI Kalman filter-based compensator shows that the robustness of the QRC compensators using output feedback is limited by the accuracy of the state estimation; the limited robustness of the Kalman filter decreases the stability of the fuzzy compensators by several orders of magnitude. Note, however, that the combination of the QRC controller and PFI Kalman Filter is more robust than the LQG.

VI. Conclusion

This paper shows that fuzzy control, based on qualitative behaviors, can achieve performance comparable or superior to that achieved by linear control when used in the domain of set point control for simple plants. Our QRC methodology uses a superimposition of qualitative stability and tracking behaviors instantiated with fuzzy rules which have clear linguistic interpretations. Using the ACC Robust Control Benchmark, we successfully demonstrate that QRC compensators are able to perform comparable to, if not better than, compensators designed with H_2 and H_∞ robust control techniques. The QRC methodology allows the fuzzy compensator to be designed incrementally, with stability behaviors being developed and evaluated initially and then supplemented with tracking behaviors. As with linear controllers, the additional tracking requirements degrade the stability robustness of the initial stability behavior. However, by properly gating the tracking behavior, we are able to develop a tracking behavior that has minimal adverse impact on stability robustness. A second more aggressive tracking behavior, while improving tracking



(a) **(b)**

Figure 9. Responses to an impulse at m_2 for the nominal plant (k = 1.0) for Fuzzy controllers QRC A and QRC B and the linear controllers Comp1 and Comp3. Figure (a) shows that the fuzzy compensators have a lower overshoot in the output while Figure (b) shows that the fuzzy controllers also dampening vibrations faster.

performance, degrades the stability robustness of the full QRC compensator by over 50%.

The limited robustness of the QRC compensator is due in part to the limited robustness of the PI Kalman filter which is incorporated into our compensator. In fact, we show that fuzzy

Stability Behavior, with its relaxed requirements for state information, extends the robustness of the Kalman filter-based compensator. The Kalman filter-based QRC compensator more than quadruples the robustness of a conventional Kalman filter-based LQG. The limitations induced by the Kalman filter are shown with additional simulations using a full state feedback version of the Benchmark, where a PI Kalman filter is no longer needed to estimate hidden states. Direct state access increased the robustness of the QRC compensator dramatically. The fuzzy controller has substantially greater stability robustness than even an optimally robust LQR, when using the same level of actuator effort.

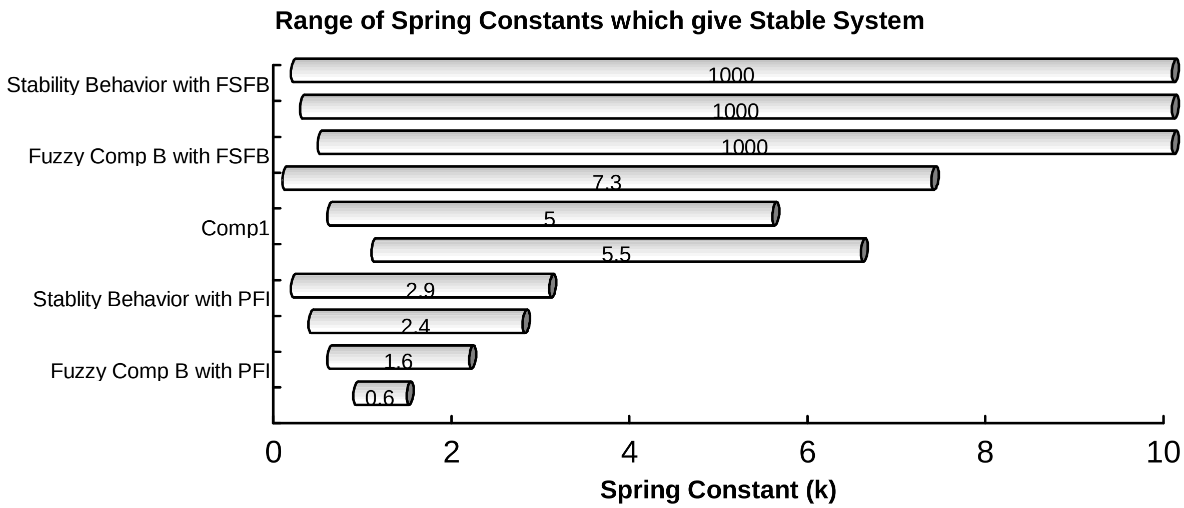
Future versions of our QRC compensators will utilize a fuzzy state estimator, where qualitative fuzzy plant models will allow the direct estimation of qualitative states. The Kalman filter and fuzzy process model can then be replaced by a single fuzzy state estimator, allowing for the development of completely fuzzy QRC compensators.

VII. Acknowledgements

The authors would like to acknowledge the outstanding work editing this publication by David Custer of the Writing Program at M.I.T. and Catherine Lavelle of M.I.T. David also provided invaluable insight into the physics of the Benchmark.

VIII. References

[1] B. D. O. Anderson and J. B. Moore, *Optimal Control: Linear Quadratic Methods*. Englewood Cliffs, NJ: Prentice Hall, 1989.
[2] P. Antsaklis, “Defining Intelligent Control: Report of the Task Force on Intelligent Control,” *IEEE Control Systems*, pp. 4, 1994.
[3] C. P. Coleman and D. Godbole, *A Comparison of Robustness: Fuzzy Logic, PID and Sliding Mode control*, Proceedings of the American Control Conference, pp. 1654-9, 1994.
[4] E. G. Collins, Jr.; Bernstein, D.S., *Robust control design for a benchmark problem using a structured covariance approach*, Proceedings of the American Control Conference, San Diego, CA, USA, pp. p. 970-1 vol.1, 1990.
[5] R. P. Copeland and K. S. Rattan, *A Fuzzy Logic Supervisor for PID Control of Unknown Systems*, Proceedings of the IEEE International Symposium on Intelligent Control, Columbus, Ohio, pp. 22-6, 1994.
[6] L. D. Davis, E. G. C. Jr., and W. M. Haddad, *Discrete-time mixed-norm H/sub 2//H/sub infinity / controller synthesis*, Proceedings of the American Control Conference, Baltimore, MD, USA, pp. 1148-52, 1994.
[7] D. Driankov, H. Hellendoorn, and M. Reinfrank, *An Introduction to Fuzzy Control*, 2 ed. New York, NY, USA: Springer, 1996.
[8] J. P. Folcher and L. E. Ghaoui, *State-feedback design via linear matrix inequalities: application to a benchmark problem*, Proceedings of the American Control Conference, Baltimore, MD, USA, pp. 1217-22, 1994.
[9] N. Gulley and J.-S. R. Jang, *Fuzzy Logic Toolbox for use with Matlab*: The Mathworks, Inc., 1995.
[10] J. P. How, W. M. Haddad, and S. R. Hall, “Application of Popov controller synthesis to benchmark problems with real parameter uncertainty,” *Journal of Guidance, Control, and Dynamics*, vol. 17, pp. 759-68, 1994.
[11] M. Jamshidi, N. Vadiee, and T. Ross, “Fuzzy Logic and Control: Software and Hardware Applications,” . Englewood Cliffs: Prentice Hall, 1993.
[12] J.-S. R. Jang and C.-T. Sun, “Neuro-Fuzzy Modeling and Control,” *Proceedings of the IEEE*, 1995.
[13] S. P. Linder and B. Shafai, *Rejection Disturbances to Flexible Structures using PI Kalman Filters*, Proceedings of the Conference on Control Applications, Hartford, CT, USA, pp. 475-7, 1997.
[14] S. P. Linder and B. Shafai, *A Systematic Approach to Fuzzy Control Design for the 1992 ACC Robust Control Benchmark*, Proceedings of the American Control Conference, Albuquerque, New Mexico, USA, pp. 1258-63, 1997.
[15] S. P. Linder and B. Shafai, *Robust PFI Kalman Filters*, Proceedings of the American Control Conference, Philadelphia, USA, pp. 3163-4, 1998.
[16] U.-L. Ly, *Robust Control Design using Nonlinear Constrained Optimization*, Proceedings of the American Control Conference, San Diego, CA, pp. 968-9, 1990.
[17] C. I. Marrison and R. F. Stengel, *The use of random search and genetic algorithms to optimize stochastic robustness functions*, Proceedings of the American Control Conference, Baltimore, MD, USA, pp. 1484-9, 1994.
[18] C. I. Marrison and R. F. Stengel, “Stochastic Robustness Synthesis Applied to a Benchmark Control Problem,” *International Journal of Robust and Nonlinear Control*, vol. 5, pp. 13-31, 1995.
[19] H. H. Niemann, J. Stoustrup, B. Shafai, and S. Beale, “LTR Design of Proportional-Integral Observers,” *International Journal of Robust and Nonlinear Control*, vol. 5, pp. 671-93, 1995.
[20] K. M. Passino, “Toward Bridging the Perceived Gap between Conventional and Intelligent Control,” in *Intelligent Control Systems: Theory and*



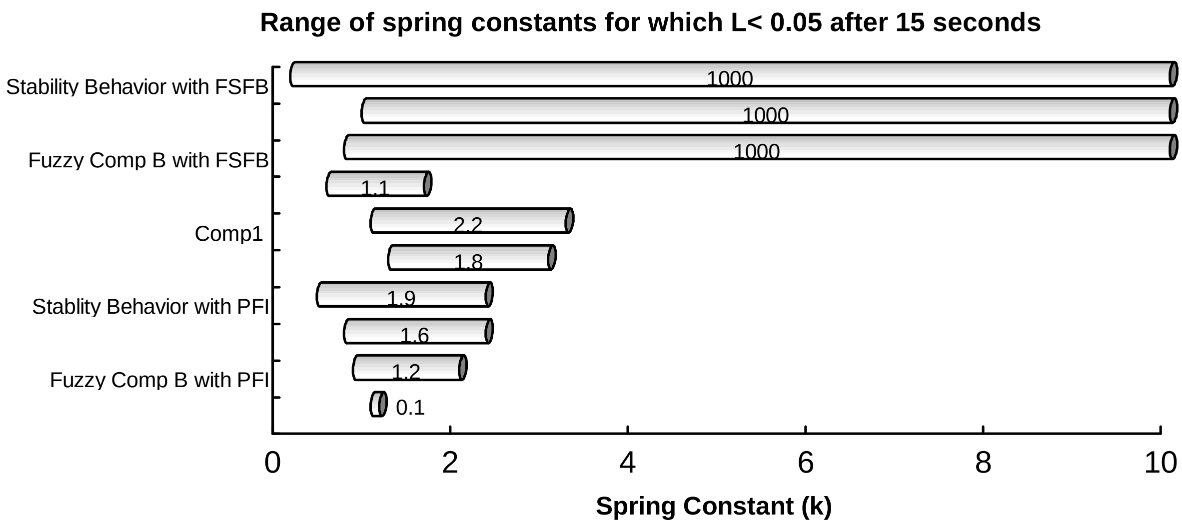


Figure 10. A comparison of stability robustness and vibration suppression robustness for various Benchmark compensators is made by showing the range of spring constants for which the compensated system is stable in response to a unit impulse to m_2 and has $|L| < 0.05$ after 15 seconds. Numeric labels give extent of k.

Application, M. M. Gupta and N. K. Sinha, Eds. New York: IEEE Press, pp. 3-27, 1996.

[21] E. H. Ruspini, A. Saffiotti, and K. Konolige, "Progress in Research on Autonomous Vehicle Motion Planning," in *Industrial Applications of Fuzzy Logic and Intelligent Systems*, J. Yen, R. Langari, and L. A. Zadeh, Eds. New York: IEEE Press, pp. 157-74, 1995.

[22] A. Saberi, B. M. Chen, and P. Sannuti, *Loop Transfer Recovery: Analysis and Design*. New York: Springer-Verlag, 1993.

[23] D. K. Schmidt and D. Benson, "Loop shaping and a zero-placement technique as applied to the benchmark problem," *International Journal of Robust and Nonlinear Control*, vol. 5, pp. 33-51, 1995.

[24] R. F. Stengel and C. I. Marrison, "Robustness of Solutions to a Benchmark Control Problem," *Journal of Guidance, Control and Dynamics*, vol. 15, pp. 1060-7, 1992.

[25] M. Sznaier and H. Rotstein, " Robust controller design for a non-collocated spring-mass system via mixed l∞/H∞ optimization," *International Journal of Robust and Nonlinear Control*, vol. 5, pp. 53-65, 1995.

[26] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-15, pp. 116-32, 1985.

[27] K. Tanaka and M. Sano, *On the concepts of regulator and observer of fuzzy control systems*, Proceedings of the Proceedings of the Third IEEE Conference on Fuzzy Systems. IEEE World Congress on Computational Intelligence, Orlando, FL, USA, pp. 767-72, 1994.

[28] B. Wie and D. S. Bernstein, *A Benchmark Problem for Robust Control Design*, Proceedings of the American Control Conference, San Diego, CA, pp. 961-2, 1990.

[29] B. Wie and D. S. Bernstein, *A Benchmark Problems for Robust Control Design (1992 ACC version)*, Proceedings of the American Control Conference, Chicago, IL, USA, pp. 2047-8, 1992.

[30] J. Yen, R. Langari, and L. Zadeh, *Proceeding of the Second International Workshop on Industrial Fuzzy Control and Intelligent Systems, ,* College Station, TX, 1992.

[31] L. A. Zadeh, "Fuzzy logic, Neural networks, and soft computing," *Communications of the ACM*, vol. 37, pp. 77-84, 1994.

Stephen Paul Linder attended Stuyvesant High School in New York and then attended the Massachusetts Institute of Technology, Cambridge, receiving a Mechanical Engineering B.S. in 1982. After working for ten years in the area of embedded control systems and software engineering he reentered school at Northeastern University, Boston, receiving a M.S. in Computer Systems Engineering in 1996 and a Ph.D. in Electrical and Computer Systems Engineering in 1998. Currently, he is a member of the research faculty at Penn State University and an associate of the Intelligent Control group at the Applied Research Laboratory. Current projects include the application of sensor data fusion and intelligent system principles to the tracking of weaving underwater vehicles, and the intelligent cooperation of underwater vehicles with only intermittent and limited data communication capabilities. When not working Steve would rather be on a mountain or in a sea kayak far away from any computer or Internet connection. See pictures at http://www.coe.neu.edu/~spl.

Bahram Shafai received his Diploma in Electrical Engineering from the Swiss Federal Institute of Technology (ETH) Zurich, Switzerland in 1978 and his Ph.D. degree in Electrical Engineering from George Washington University, Washington D.C. in 1985. He joined the Northeastern University faculty in 1985, served as the Director of the Communications and Digital Signal Processing (CDSP) Center from 1990 to 1992, and was a Visiting Professor at the Technical University of Denmark, Lyngby. He serves on the organizing committees of the CDC, ACC and the World Congresses of IFAC and WAC and has often appeared as an invited speaker in the area of Robust Control. Dr. Shafai is active within the IEEE control systems and signal processing societies. He is the co-author of *Computer-Aided Analysis and Design of Linear Control Systems*, a book published by Prentice Hall in 1992. His research interests include robust multivariable control systems, optimal estimation and control, fuzzy and intelligent control, and digital signal processing with an emphasis on filter design, adaptive filtering and system identification. He conducted research in the application of these concepts to magnetic bearing control systems, multiple sensor data fusion and tracking of targets, electromechanical systems, and DC-DC power converters.