Multiobjective Algebraic Synthesis of Neural Control Systems by Implicit Model Following

Silvia Ferrari, *Senior Member, IEEE*

***Abstract*—The advantages brought about by using classical linear control theory in conjunction with neural approximators have long been recognized in the literature. In particular, using linear controllers to obtain the starting neural control design has been shown to be a key step for the successful development and implementation of adaptive-critic neural controllers. Despite their adaptive capabilities, neural controllers are often criticized for not providing the same performance and stability guarantees as classical linear designs. Therefore, this paper develops an algebraic synthesis procedure for designing dynamic output-feedback neural controllers that are closed-loop stable and meet the same performance objectives as any classical linear design. The performance synthesis problem is addressed by deriving implicit model-following algebraic relationships between model matrices, obtained from the classical design, and the neural control parameters. Additional linear matrix inequalities (LMIs) conditions for closed-loop exponential stability of the neural controller are derived using existing integral quadratic constraints (IQCs) for operators with repeated slope-restricted nonlinearities. The approach is demonstrated by designing a recurrent neural network controller for a highly maneuverable tailfin-controlled missile that meets multiple design objectives, including pole placement for transient tuning, H_∞ and H_2 performance in the presence of parameter uncertainty, and command-input tracking.**

***Index Terms*—Closed-loop stability, dynamic control systems, linear matrix inequalities, neural control, output-feedback control, recurrent neural networks.**

I. Introduction

MANY practical applications require control systems that can meet multiple objectives including closed-loop stability and robustness, for safe operation in the presence of uncertainties, as well as performance requirements pertaining to noise, disturbance rejection, and transient response. A very useful result from linear control theory is that a multiobjective synthesis problem that combines quadratic closed-loop stability with other objectives, such as H_∞ and H_2 performance and pole placement, can be approached by means of linear matrix inequalities (LMIs) [1]. In recent years, considerable theoretical advancements have been made regarding the stability and performance of feedback neural systems (e.g., [2]–[16]). But, there is yet no simple way to extend the insights and applicability afforded by linear control methods, such as multiobjective synthesis, to the specification and preliminary design of neural controllers. The advantages brought about by using linear controllers to design and train neural network controllers have long been recognized in the literature [17]–[25]. In particular, using linear controllers to obtain the starting neural control design has been shown to be a key step in the development of several highly effective adaptive neural controllers (e.g., see [25]–[29] and references therein). One reason is that the starting design provides adequate performance while the adaptation compensates for nonlinearities and unmodeled dynamics. Another reason is that many popular adaptation schemes, such as adaptive critics, improve iteratively upon the existing approximation of the control law. Therefore, the starting neural control design provides a performance baseline for the adaptive controller [25], [30].

Prior research [27] showed that, by adapting to online aircraft dynamics through adaptive critics, a neural flight controller initialized with a classical gain-scheduled linear design [24] can prevent stall and loss of control during unexpected control failures and maneuvers. Despite exhibiting excellent performance in numerical tests, neural controllers that are either augmented or initialized by a linear design, such as [20] and [23]–[26], typically are not accompanied by guarantees of closed-loop stability or performance away from the operating set points that are used for training. This limiting factor may prevent the replacement of linear controllers by neural networks in applications, such as flight and power systems control, where a high level of safety and performance must be guaranteed everywhere in the operating envelope. Therefore, this paper addresses the specification of the starting neural control law, and shows that given *any* dynamic linear controller, a recurrent neural control system that meets the same performance objectives and is closed-loop stable can be synthesized by means of algebraic equations and LMIs.

As reviewed in [31], earlier synthesis procedures applicable to the design of a starting neural control system based on linear control methods rely on optimization-based training algorithms that offer performance guarantees only in the vicinity of sample operating points. Also, due to the presence of repeated nonlinearities, the closed-loop stability and performance guarantees of the linear methods often cannot be extended to the trained neural controller. Synthesis procedures that offer closed-loop stability guarantees, on the other hand, rely on the assumption of a small reconstruction error. But this assumption is not easily met in solving the performance synthesis problem, due to the lack of training algorithms that can determine the appropriate neural network size and parameters required to ensure good dynamic approximation over the entire operating envelope [31], [32].

Manuscript received April 26, 2007; revised March 20, 2008; accepted July 30, 2008. First published February 06, 2009; current version published February 27, 2009. This work was supported by the National Science Foundation under Grant ECS 0300236 and CAREER award ECS 0448906.

The author is with the Department of Mechanical Engineering & Materials Science, Duke University, Durham, NC 27708-0005 USA (e-mail: sferrari@duke.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNN.2008.2008332

Recently, an algebraic technique for training neural networks without relying on optimization algorithms was presented in [33]. By using basic linear algebra, this technique determines the size of the network that is necessary to achieve exact matching of a given training set and computes the neural parameters by solving linear systems of equations. In this paper, this algebraic training technique is combined with an implicit model-following framework to solve the performance synthesis problem (Section IV). In implicit model following (IMF), the control system is synthesized by determining an algebraic relationship between the controller's parameters and the known parameters of an ideal model that specifies the desired closed-loop response [34], [35]. In this paper, IMF is applied to the neural network controller to guarantee that the origin of the reconstruction error dynamics (where the error is zero) is met at *any* set point of the closed-loop system and any deviations from it decay exponentially fast. Then, in Section V, stability LMIs obtained from the integral quadratic constraints (IQCs) developed in [36] are used to guarantee that the resulting closed-loop system is exponentially stable. In Section VII, the approach is illustrated for a tailfin-controlled missile that must meet a set of specifications including exponential stability, tracking of a reference signal with minimal overshoot, and adequate transient behavior, rolloff, and $\mathcal{L}_2$ performance for normalized parameter uncertainties.

II. Problem Statement

The advantages brought about by embedding prior linear control knowledge in the starting design of adaptive neural controllers have been demonstrated by several authors [17]–[27]. For example, in [27], an adaptive flight controller initialized with linear multivariable controllers was shown to adapt rapidly to unexpected control failures and nonlinear regimes, in some cases preventing stall and loss of control. However, neural controllers that are either augmented or initialized by a linear design, such as [20], [23], [26], and [27], typically are not accompanied by guarantees of closed-loop stability or performance away from the operating set points that are used for training. This is a hindering limitation in the replacement of linear controllers by neural controllers in applications such as flight and power control systems, where safety is a key concern and the system to be controlled is expected to operate in the linear regime most of the time. Consequently, this paper addresses the synthesis of an output-feedback dynamic neural controller that meets the following objectives:

1) it meets multiple performance objectives synthesized by a set of dynamic linear controllers;
2) it is closed-loop exponentially stable.

As a result, neural network controllers are proven capable of providing the same performance and safety guarantees as *any* classical linear designs. By this approach, a set of linear controllers is replaced by one global nonlinear controller comprised entirely by a sigmoidal neural network that interpolates automatically over the entire operating envelope. Since the neural network controller is adaptive, this synthesis approach can be used to obtain the starting action-network design in adaptive-critic architectures [25]–[27].

III. Background

Consider the class of affine nonlinear parameter-varying systems

$$P(s,\zeta):\begin{cases}\dot{x}=A\left(\zeta(t)\right)x+B_w\left(\zeta(t)\right)w+B_u\left(\zeta(t)\right)u\\ y=C\left(\zeta(t)\right)x+D\left(\zeta(t)\right)u\end{cases}\quad(1)$$

which can be used to model a variety of plants, including missiles, airplanes, robots, and electrical circuits [37], where $x\in\mathcal{X}\subset\mathbb{R}^{n\times 1}$ is the state, $u\in\mathbb{R}^{m\times 1}$ is the control, and $y\in\mathbb{R}^{r\times 1}$ is the output. $w\in\mathbb{R}^{s\times 1}$ represents a disturbance and, in this paper, it is used to capture parameter uncertainty over the state space $\mathcal{X}$, as in [1], [6], and [38]. The state–space matrices are known affine functions of a time-varying *scheduling vector* $\zeta(t)\in\mathbb{R}^{z\times 1}$, which contains physical parameters and state variables that significantly influence the system dynamics. As shown in [39] and [40], given any convex decomposition $\zeta(t)=\sum_{j=1}^{p}c_j\zeta_j$ in a box of $\mathbb{R}^{z\times 1}$, with a set of vertices $\mathcal{Z}=\{\zeta_1,\ldots,\zeta_p\}$, $c_j\geq 0$, and $\sum_{j=1}^{p}c_j=1$, the affine parameter-dependent system (1) ranges in a matrix polytope with vertices specified by $A(\zeta_j)$, $B_w(\zeta_j)$, $B_u(\zeta_j)$, $C(\zeta_j)$, and $D(\zeta_j)$, $j=1,\ldots,p$. Therefore, (1) can be represented by p state–space models

$$P_j(s):\begin{cases}\dot{x}=A(\zeta_j)x+B_w(\zeta_j)w+B_u(\zeta_j)u\\ y=C(\zeta_j)x+D(\zeta_j)u,\end{cases},\quad j=1,\ldots,p\quad(2)$$

and if $S(\zeta)$ denotes the system matrix of (1), then $S(\zeta)=\sum_{j=1}^{p}c_jS(\zeta_j)$, where $S(\zeta_j)$ is the system matrix of $P_j(s)$ in (2), and the coefficients c_j are given by the decomposition of ζ [39]–[41]. From hereon, the shorthand notation $A_j=A(\zeta_j)$, $B_{w_j}=B_w(\zeta_j)$, $B_{u_j}=B_u(\zeta_j)$, $C_j=C(\zeta_j)$, and $D_j=D(\zeta_j)$ is adopted for brevity.

Several techniques including multivariable control [42], linear fractional transformations [37], and H_∞ [40], [43] have been developed to design gain-scheduled linear dynamic controllers for (1) in the form

$$K(s,\zeta):\begin{cases}\dot{x}_K=A_K\left(\zeta(t)\right)x_K+B_K\left(\zeta(t)\right)y\\ u=u_K:=C_K\left(\zeta(t)\right)x_K+D_K\left(\zeta(t)\right)y.\end{cases}\quad(3)$$

In particular, a gain-scheduled controller that is closed-loop stable and meets multiple design objectives, including H_∞ and H_2 performance, and pole placement, can be obtained by solving a multiobjective synthesis problem that results in sets of LMIs, as shown in [1] and [41]. In this classical approach, given a convex decomposition of $\zeta(t)$, the state–space matrices of the parameter-dependent controller (3) at any measured value of ζ in $\mathcal{X}$ are obtained by convex interpolation of a set of linear time-invariant (LTI) vertex controllers $\mathcal{K}=\{K_1,\ldots,K_p\}$ [37], [44], where

$$K_j:=\begin{bmatrix}A_{K_j} & B_{K_j}\\ C_{K_j} & D_{K_j}\end{bmatrix}:=\begin{bmatrix}A_K(\zeta_j) & B_K(\zeta_j)\\ C_K(\zeta_j) & D_K(\zeta_j)\end{bmatrix},\quad j=1,\ldots,p.\quad(4)$$

In this paper, it is assumed that a set $\mathcal{K}$ that achieves the aforementioned design objectives is given, or may be obtained via convex optimization. Then, $\mathcal{K}$ is used to synthesize a

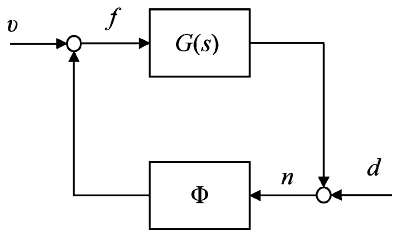

Fig. 1. Basic feedback interconnection for IQC analysis.

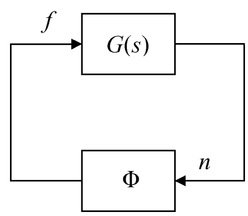

Fig. 2. Feedback configuration for stability analysis.

global neural network controller that provides satisfactory performance in $\mathcal{X}$. The main advantages of the neural network controller over a classical gain-scheduled design, such as convex interpolation, are that a single controller performs the interpolation automatically over $\mathcal{X}$, and that, thanks to its adaptive capabilities, the neural network can be immediately implemented as the starting action-network design in an adaptive-critic architecture [25]–[27].

A. Background on IQCs for Systems With Repeated Nonlinearities

This section is a review of the IQCs obtained in [36], which are used in this paper to guarantee the closed-loop stability of the neural network control system. IQC methods can be used to analyze the stability and robustness of a feedback interconnection (Fig. 1) between a linear operator G and a bounded causal operator Φ that is, possibly, nonlinear

$$\begin{aligned} n &= Gf + d \\ f &= \Phi(n) + v \end{aligned} \tag{5}$$

where $n, d \in \mathcal{L}_{2e}^l$ and $f, v \in \mathcal{L}_{2e}^m$. $\mathcal{L}_{2e}^n$ denotes the linear space of all functions $\phi : (0, \infty) \to \mathbb{R}^n$, which are square integrable on any finite interval, and the signals d and v represent the interconnection noise.

For stability analysis, consider the configuration in Fig. 2, where $G(s)$ is the transfer function of a stable linear system with state–space realization

$$G(s) : \begin{cases} \dot{\chi} = A\chi + Bf \\ n = C\chi + Ef \end{cases} \tag{6}$$

and suppose Φ can be described by the IQC inequality

$$\int_{-\infty}^{\infty} \begin{bmatrix} \hat{n}(j\omega) \\ \hat{f}(j\omega) \end{bmatrix}^* \Pi(j\omega) \begin{bmatrix} \hat{n}(j\omega) \\ \hat{f}(j\omega) \end{bmatrix} d\omega \geq 0 \tag{7}$$

where $(\hat{\cdot})$ denotes the Fourier transform at frequency ω, A^* denotes the adjoint of A, and Π is a Hermitian matrix function. Then, under appropriate assumptions [45, Th. 1], the IQC stability theorem states that if there exists $\epsilon > 0$ such that

$$\begin{bmatrix} G(j\omega) \\ I \end{bmatrix}^* \Pi(j\omega) \begin{bmatrix} G(j\omega) \\ I \end{bmatrix} \leq -\epsilon I \qquad \forall \omega \in \mathbb{R} \tag{8}$$

the feedback interconnection of G and Φ is stable.

The class $\mathbf{\Pi}_\Phi$ of all Π that define a valid IQC for a particular operator Φ is convex and, in some cases, can be readily found in the literature [45]. Following the approach proposed in [46], the search for a suitable $\Pi \in \mathbf{\Pi}_\Phi$ can be restricted to a finite-dimensional subset and carried out by numerical optimization. By applying the Kalman–Yakubovich–Popov lemma [47], the inequality in (8) can be transformed into an LMI feasibility problem. In the case of monotonic and slope-restricted diagonal operators with repeated nonlinearities, this LMI feasibility problem consists of showing that there exist constant matrices $T \in \mathbb{R}^{l \times l}$ and $Q \in \mathbb{R}^{\nu \times \nu}$ such that

$$\begin{aligned} &T = T^T \quad Q = Q^T \\ &T_{ii} \geq \sum_{j=1, j \neq i}^{l} |T_{ij}|, \qquad i = 1, \ldots, l \\ &\begin{bmatrix} A^T Q + QA & QB + C^T T \\ (QB)^T + TC & -2T + E^T T + TE \end{bmatrix} < 0 \end{aligned} \tag{9}$$

where ν is the dimension of the state vector χ in (6), as proven in [36]. In Section V, this result is used to guarantee the closed-loop stability of the neural network controller by combining (9) with the performance synthesis equations derived in the next section.

IV. Performance Synthesis Problem

The dynamic neural control law is formulated in terms of a sector-bounded operator Φ with repeated sigmoidal nonlinearities

$$\begin{cases} \dot{x}_K = A_K(\zeta) x_K + B_K(\zeta) y \\ u = u_N := v\Phi(W\chi) \end{cases} \tag{10}$$

where $\chi := [x^T \; x_K^T]^T \in \mathbb{R}^{\nu \times 1}$, and the internal controller state $x_K \in \mathbb{R}^{\kappa \times 1}$ is a function of u_N (as illustrated in Fig. 3). Therefore, (10) is a recurrent neural network. The control is assumed to be scalar without loss of generality. The adjustable parameters $v \in \mathbb{R}^{1 \times l}$ and $W \in \mathbb{R}^{l \times (\nu + z)}$, and the matrix functions $A_K \in \mathbb{R}^{\kappa \times \kappa}$ and $B_K \in \mathbb{R}^{\kappa \times r}$ that meet objectives 1) and 2) in Section II are to be determined (Section VI). Φ is a diagonal operator with repeated sigmoidal functions

$$\Phi(n) := [\sigma(n_1) \cdots \sigma(n_l)]^T \tag{11}$$

where n_i denotes the ith component of a signal $n \in \mathbb{R}^{l \times 1}$. The sigmoidal function $\sigma : \mathbb{R} \to \mathbb{R}$ is a bounded measurable function on $\mathbb{R}$, for which $\sigma(n_i) \to 1$ as $n_i \to \infty$, and $\sigma(n_i) \to 0$ as $n_i \to -\infty$, and, in this paper, it takes the form

$$\sigma(n_i) := \frac{e^{n_i} - 1}{e^{n_i} + 1}. \tag{12}$$

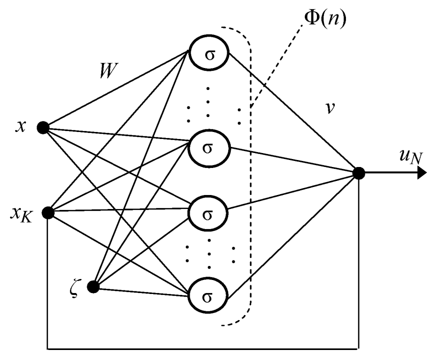

Fig. 3. State-feedback dynamic neural network controller.

It can be easily shown that (12) is monotonically nondecreasing and slope restricted, and that it belongs to the sector $[\alpha, \beta]$, with $\alpha = 0$ and $\beta = 1/2$. Therefore, the stability of a feedback interconnection between a linear operator and the sigmoidal operator Φ can be approached using the IQCs reviewed in Section III-A.

The performance synthesis problem is formulated using IMF. Based on the problem statement in Section II, given the plant (1) and the set (4), the objective is to design a controller with repeated sigmoidal nonlinearities (10), such that the resulting closed-loop dynamics given by

$$\dot{x}_{cl} = A(\zeta)x_{cl} + B_w(\zeta)w + B_u(\zeta)v\Phi(W\chi_{cl}) \tag{13}$$

follow the model obtained from (1) and (3)

$$\begin{cases} \dot{x}_m = A(\zeta)x_m + B_w(\zeta)w + B_u(\zeta)[C_K(\zeta)x_{K_m} + D_K(\zeta)y_m] \\ \dot{x}_{K_m} = A_K(\zeta)x_{K_m} + B_K(\zeta)y_m \\ y_m = [I - D(\zeta)D_K(\zeta)]^{-1}[C(\zeta)x_m + D(\zeta)C_K(\zeta)x_{K_m}] \end{cases} \tag{14}$$

as closely as possible, and are exponentially stable, where $[I - D(\zeta)D_K(\zeta)]$ can be assumed to be invertible, and the time argument is omitted for simplicity. The model (14) is known and specifies the desired *closed-loop* response with respect to multiple performance objectives (Section III).

Similarly to classical IMF techniques [34], [35], the model (14) is used to find a suitable relationship between the neural control parameters v and W and the known model matrices in (14) (Section III). As in [34] and [35], let the generalized error be defined as

$$e_r := \dot{\chi}_{cl} - \dot{\chi}_m \tag{15}$$

where $\chi_{cl} := [x_{cl}^T \; x_K^T]^T$, $\chi_m := [x_m^T \; x_{K_m}^T]^T$, x_K is the state of the dynamic neural network controller (10), and x_{K_m} is the state of the dynamic linear controller (3). The error e_r can also be viewed as a reconstruction error in the training of the neural control system (10). A well-known result from multivariable control theory is that linear model-following control systems can be designed by requiring $e_r(t)$ to go to zero as $t \to \infty$, as in [48] and [49], or by minimizing its $\mathcal{L}_2$ norm [42, pp. 520–524]. In this paper, the generalized error dynamics and the algebraic training approach presented in [33] are used to guarantee that the nonlinear closed-loop system (13) follows the performance specified by the model (14), and any deviations from the model decay exponentially fast. Modifying the definition in [3], we say that two systems are input–output and gradient equivalent, if they are characterized by the same output and state derivatives when $\dot{\chi} = 0$. Then, the following theorem can be used to specify the performance of a dynamic neural network controller.

Theorem 1: Given a set of LTI controllers $\mathcal{K}$, there exists a controller (10) with $l = p$ sigmoidal nonlinearities that is input–output equivalent to (3) for all $\zeta_j \in \mathcal{Z}$, i.e., satisfies the *closed-loop* requirements

$$\begin{cases} u_N(\zeta_j) = u_K(\zeta_j) \\ \dfrac{\partial u_N}{\partial \dot{\chi}}(\zeta_j) = \dfrac{\partial u_K}{\partial \dot{\chi}}(\zeta_j) \end{cases}, \qquad j = 1, \ldots, p \tag{16}$$

if there exist a matrix $W = [W_\chi \; W_\zeta]$ and a vector $v^T \in \mathbb{R}^{l \times 1}$ that satisfy the linear systems

$$\begin{cases} N = W_\zeta Z \\ Sv^T = b \\ DVW_\chi = M_2 \end{cases} \tag{17}$$

and provided the matrices $(I - D_j D_{K_j})$ are invertible, where

$$M_{1_j} := \left(I - D_j D_{K_j}\right)^{-1} \left[C_j \quad D_j C_{K_j}\right] \tag{18}$$
$$M_{2_j} := \left[0 \quad C_{K_j}\right] + D_{K_j} M_{1_j} \tag{19}$$
$$M_2 := \left[M_{2_1}^T \cdots M_{2_p}^T\right]^T \tag{20}$$
$$Z := [\zeta_1 \cdots \zeta_p] \tag{21}$$
$$N := [n^1 \cdots n^p] \tag{22}$$
$$S := \left[\Phi(n^1) \cdots \Phi(n^p)\right]^T \tag{23}$$
$$D := \left[\Phi'(n^1) \cdots \Phi'(n^p)\right]^T. \tag{24}$$

$V = \mathrm{diag}(v)$, $b := \tilde{b}\imath_p$, where $\imath_p := 1_{p \times 1}$, $\tilde{b}$ is a known constant, and the diagonal operator Φ' is defined in terms of the sigmoid derivative $\sigma' = d\sigma(n_i)/dn_i$

$$\Phi'(n) := \left[\sigma'(n_1) \cdots \sigma'(n_l)\right]^T. \tag{25}$$

A proof is provided in Appendix I.

The linear systems in (17) have unknowns N, W_ζ, v, and W_χ. Therefore, if they are approached in the order provided, they may be solved by linear algebra techniques. Theorem 1 specifies the number of sigmoidal nonlinearities l and matrices $A_K(\zeta)$ and $B_K(\zeta)$. Also, (17) specifies satisfactory parameter values for *any* matrix N (provided S and D are invertible). As shown in [33], generating a matrix N such that S and D in (23) and (24) are invertible is straightforward and effective in training neural networks to approximate smooth nonlinear functions. In this case, given a matrix Z of scheduling vectors, two matrices N and W_ζ can always be found such that the first system in (17) is satisfied, subsequently specifying both S and D. Then, the second system in (17) is linear and square and can be used to compute v and V. After both D and V are determined, the third and last system in (17) is linear and square and can be used to compute W_χ. An important consequence of Theorem 1 is that a set of linear relationships between the neural control parameters and the known vertices of (14) are established. Based on these relationships, it can be easily shown that for all $\zeta_j \in \mathcal{Z}$ the feedback interconnection of the plant (1) and the neural network controller (10), given by

$$\begin{cases} \dot{\chi_{cl}} = A_{cl}(\zeta_j)\chi_{cl} + B_{cl}(\zeta_j)vf \\ n = W\chi_{cl} \\ f = \Phi(n) \end{cases} \quad j = 1, \ldots, p \tag{26}$$

where

$$A_{cl}(\zeta_j) := \begin{bmatrix} A_j & 0 \\ B_{K_j}C_j & A_{K_j} \end{bmatrix} \quad \text{and} \quad B_{cl}(\zeta_j) := \begin{bmatrix} B_j \\ B_{K_j}D_j \end{bmatrix} \tag{27}$$

is in the same form as the IQC feedback interconnection (Fig. 2). It follows that the IQCs reviewed in Section III-A can be used to derive closed-loop stability conditions (as shown in Section V). Additionally, model-following performance guarantees away from the vertices in $\mathcal{Z}$ (the training set points) are given by the following result.

Theorem 2 (Model Following): Let $e_r(t)$ denote the generalized error (15) between the closed-loop dynamics (13) and the model (14). Assume the following.

i) The model is designed such that the matrices

$$A_{e_j} := \begin{bmatrix} A_{cl}(\zeta_j) & -B_{cl}(\zeta_j)M_{2_j} \\ 0 & A_a(\zeta_j) \end{bmatrix}, \qquad j = 1, \ldots, p \tag{28}$$

with A_a defined in (56), are Hurwitz.

ii) The controller parameters V and W satisfy the conditions in Theorem 1.

iii) The disturbances rate of change $\dot{w}$ is negligibly small.

Then, the error state equation can be written as the sum of a linear nominal system and a nonlinear time-varying perturbation

$$\dot{e} := \begin{bmatrix} \dot{e}_r \\ \ddot{\chi}_m \end{bmatrix} = A_e(\zeta)e + \begin{bmatrix} B_{cl}(\zeta)\Phi'\left(n(t)\right)^T VW \\ 0 \end{bmatrix}[I \quad -I]e, \qquad j = 1, \ldots, p \tag{29}$$

and has an equilibrium point at the origin $e = 0$. For any set point or nonzero equilibrium, one of the systems in (29) is at its zero-equilibrium point $e = 0$, leading to *perfect model following*. Let P_e be a positive–definite symmetric matrix that satisfies the Lyapunov equation

$$P_e A_e(\zeta) + A_e^T(\zeta)P_e = -Q \tag{30}$$

where $A_e(\zeta) = \sum_{j=1}^{p} c_j A_{e_j}$, $Q = \sum_{l=1}^{p} Q_l$, and $Q_l = Q_l^T > 0$, such that $V(e) = e^T P_e e$ is a Lyapunov function for the nominal system $\dot{e} = A_e(\zeta)e$. If the nonlinear control parameters V and W satisfy the bounds

$$\sqrt{2l}\beta \left\|B_{cl}(\zeta)\right\|_2 \left\|VW\right\|_2 < \frac{1}{2\lambda_{\max}(P_e)}, \qquad j = 1, \ldots, p \tag{31}$$

or the bounds

$$\left\{VW[P_{11} - P_{21}]B_{cl}(\zeta)\right\}_i < 0, \qquad i = 1, \ldots, l \tag{32}$$

where

$$B_{cl}(\zeta) = \sum_{j=1}^{p} c_j B_{cl}(\zeta_j) \quad \text{and} \quad P_e = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \tag{33}$$

are known matrices, then, the origin of the error (29) is an exponentially stable equilibrium.

A proof is provided in Appendix II.

Based on the results in this section, the performance of the dynamic neural control system can be specified through the linear systems in (17) relating the model matrices to the neural parameters. Thus, perfect model following is guaranteed at *any* set point of the closed-loop system and, if the controller parameters also satisfy (31) or (32), any response deviations from the models will decay exponentially fast. Also, using (17) and the IQCs reviewed in Section III-A, the closed-loop stability of the resulting neural network control system can be guaranteed as shown in the next section.

V. Closed-Loop Stability of Neural Controller via IQCs

By the synthesis problem formulation in Section IV, the neural-network-controlled system (26) has the same form as the IQC feedback interconnection (5). For the purpose of stability analysis, the feedback configuration in Fig. 2, with no interconnection noise, is considered. Since Φ is a diagonal operator with repeated sigmoidal nonlinearities that are monotonically nondecreasing, slope restricted, and sector bounded (Section IV), (9) can be used to analyze the closed-loop stability of (26). However, when used for neural network synthesis, (9), which is rewritten as (80), becomes a nonconvex problem due to the presence of the products between the LMI variables and the neural parameters (similarly to the problem in [6]). Therefore, in the following proposition, a new stability LMI is obtained that is less conservative than (9), and can be combined with (17) to obtain new stability conditions that are linear with respect to a matrix function of the neural parameters.

Proposition 1: Assume that for all $\zeta_j \in \mathcal{Z}$, $A_{cl}(\zeta_j) \in \mathbb{R}^{\nu\times\nu}$ is asymptotically stable, and $Q_{i_j} \in \mathbb{R}^{\nu\times\nu}$ is the positive–definite solution of $A_{cl}^T(\zeta_j)Q_{i_j} + Q_{i_j}A_{cl}(\zeta_j) = -R_{i_j}$, with positive–definite $R_{i_j} \in \mathbb{R}^{\nu\times\nu}$, and $i = 1, 2$. Then, the feedback interconnection in (26) is exponentially stable if and only if there exist p diagonally dominant constant matrices $T_j \in \mathbb{R}^{l\times l}$, such that

$$\mathcal{M}_2 := \begin{bmatrix} -R_{2_j} & \beta W^T T_j \\ \beta T_j W & -2T_j + \left[Q_j B_{cl}(\zeta_j)v\right]^T \left(R_{1_j}\right)^{-1} \left[Q_j B_{cl}(\zeta_j)v\right]^T \end{bmatrix} \leq 0 \tag{34}$$

with $Q_j := \sum_{i=1}^{2} Q_{i_j}$. Then

$$\begin{aligned} -2T_j &\leq \left[\left[Q_j B_{cl}(\zeta_j)v\right]^T + \beta T_j W\right]^T \left[A_{cl}(\zeta_j)^T Q_j + Q_j A_{cl}(\zeta_j)\right]^{-1} \\ &\quad \times \left[\left[Q_j B_{cl}(\zeta_j)v\right]^T + \beta T_j W\right] \\ &\leq \left[Q_j B_{cl}(\zeta_j)v\right]^T \left(-R_{1_j}\right)^{-1} \left[Q_j B_{cl}(\zeta_j)v\right] \\ &\quad + \beta^2 T_j W \left(-R_{2_j}\right)^{-1} W^T T_j. \end{aligned} \tag{35}$$

A proof is provided in Appendix IV.

The LMI (34) presents several advantages with respect to the original LMI (80), obtained from (9) in Appendix IV. Since (34) is required to be negative semidefinite instead of negative definite, it is sufficient to seek a feasible solution, which typically is easier than to seek a strictly feasible one [41]. Also, as shown by the inequalities in (35) and the proof in Appendix IV, the second bound in (35), which is obtained from the LMI in (34), is less conservative than the first bound, which is obtained from the original LMI in (80). Since for all $R_{i_j} > 0$ there always exists a solution $Q_{i_j} = Q_{i_j}^T > 0$, then R_{2_j} and R_{1_j} can be chosen *a priori* for all j, reducing the LMI variables to T_j. Although usually it is not desirable to eliminate LMI variables, in this case, this result is useful because it allows to separate the neural controller parameters v and W from the LMI variables. As a result, in (34), v and W appear in different partitions of $\mathcal{M}_2$. Consequently, (34) can be used to derive an LMI that is linear with respect to a matrix function of v and W, as shown by the following result.

Proposition 2: Assume that $A_{cl}(\zeta_j) \in \mathbb{R}^{\nu \times \nu}$ is asymptotically stable for all $\zeta_j \in \mathcal{Z}$, v and W satisfy the conditions in Theorem 1, and $Q_j := \sum_{i=1}^{2} Q_{i_j}$ is known from the positive–definite solutions of (81), such that $\mathcal{Q}_j := Q_j B_{cl}(\zeta_j) b^T$ also is known and constant for $j = 1, \ldots, p$. Then, if there exist p symmetric matrices $P_j \in \mathbb{R}^{p \times p}$ that each allow a factorization $P_j = S T_j S^T$, with T_j diagonally dominant, and such that

$$\begin{bmatrix} \mathcal{W} & \beta P_j^T \\ \beta P_j & -2P_j + \mathcal{Q}_j^T R_1^{-1} \mathcal{Q}_j \end{bmatrix} \leq 0, \qquad j = 1, \ldots, p \quad (36)$$

where $\mathcal{W}$ is a negative–semidefinite matrix function of v and W, given by (93), the feedback interconnection (26) is exponentially stable.

A proof is provided in Appendix V.

It can be seen that (36) is linear with respect to the variables $\mathcal{W}$ and P_j, because the neural parameters v and W are now confined to $\mathcal{W}$. Since (36) is equivalent to (34), it also provides a less conservative bound than (80). Therefore, the above proposition provides new stability conditions that can be used to synthesize a neural controller that is guaranteed *a priori* to be closed-loop stable, as explained in the next section.

VI. Synthesis Procedure

Theorem 1 provides a criterion for choosing the number of sigmoidal nonlinearities, and for computing the neural control parameters v and W, such that the neural controller is guaranteed to meet the desired performance objectives for all $\zeta_j \in \mathcal{Z}$ for any suitable choice of the design matrix N. The values of v and W can be computed from three linear systems of equations, obtained from (17), by the following procedure. Given a right-invertible matrix Z formed from the p scheduling vectors in $\mathcal{Z}$, design a matrix N such that the first system in (17) is consistent, then

$$W_\zeta = N Z^+. \quad (37)$$

Once N has been designed, the matrices S and D are computed from (23) and (24), respectively. Then, given any choice of a small scalar $\tilde{b}$ (or output bias), it follows from the second system in (17) that

$$v^T = S^{-1} b. \quad (38)$$

Letting $V = \mathrm{diag}(v)$, the last system in (17) also is linear, and can be inverted to compute the remaining parameters

$$W_\chi = (DV)^{-1} M_2 \quad (39)$$

such that $W = [W_\chi \; W_\zeta]$. Therefore, one approach to designing the nonlinear controller (10) is to choose a matrix N according to the guidelines provided above and in [33]. Subsequently, compute v and W using (37)–(39), and verify that these parameters satisfy one of the model-following bounds (31) or (32), and produce a feasible LMI problem (34). If the conditions are not all met, the procedure is repeated for a different choice of design matrices. Since all of the steps in this design process consist of solving linear systems of equations, or LMIs, they can be performed very efficiently and are easily repeated.

Another approach consists of designing N such that the model-following bounds and the stability conditions in Proposition 2 are simultaneously satisfied. In this case, the matrix N is still designed such that the first system in (17) is consistent, and such that S and D are nonsingular. But, additionally, D must satisfy the bound (32), which can be written more conveniently as

$$\left\{ D^{-1} M_2 (P_{11} - P_{21}) \sum_j c_j B_{cl}(\zeta_j) \right\}_i < 0, \qquad \text{for } i = 1, \ldots, p \quad (40)$$

where $\{\cdot\}_i$ denotes the ith element of the vector in parenthesis, and all other terms are known and constant. Finally, N must be designed such that $\mathcal{W}$ is negative semidefinite and the LMIs (36) are feasible under the conditions stated in Proposition 2. Since the relationship between $\mathcal{W}$ and N is nonlinear, as shown by (93), this approach requires solving a set of nonlinear equations numerically for the p^2 elements of N. This synthesis procedure guarantees *a priori* that both of the design objectives 1) and 2) in Section II are satisfied by the neural network controller.

VII. Tailfin-Controlled Missile Problem

The results obtained in the previous sections are used to synthesize a dynamic neural network controller for a highly maneuverable tailfin-controlled missile model, taken from [38], for which several linear control designs have been proposed in the literature [1], [38], [50], [51]. The objective is to demonstrate that a dynamic neural network controller (10) synthesized according to Section VI can meet the same multiple performance objectives and stability guarantees as the best available classical design [1]. The missile model

$$\begin{aligned} \begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} &= \begin{bmatrix} f_1(\alpha) & 1 \\ f_2(\alpha) & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} 0 \\ l_1 \end{bmatrix} w + \begin{bmatrix} g_1(\alpha) \\ g_2 \end{bmatrix} \delta \\ \begin{bmatrix} \eta \\ q \end{bmatrix} &= \begin{bmatrix} h_1(\alpha) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} h_2 \\ 0 \end{bmatrix} \delta \end{aligned} \quad (41)$$

falls into the class (1), where $f_1(\alpha)$, $f_2(\alpha)$, $g_1(\alpha)$, and $h_1(\alpha)$ are nonlinear functions of the angle of attack α that are continuous and bounded over their domain $\{\alpha \in \mathbb{R} : |\alpha| \leq 20\}$. Together with the other model parameters, these functions are defined in Appendix VI. The plant state $x = [\alpha \; q]^T$ consists of the angle of attack α and pitch rate q. The measurable output is $y = [\eta \; q]^T$, where η is the vertical acceleration, and the control is the tailfin deflection δ. w is used to capture variations in the missile aerodynamic coefficients by introducing the following auxiliary signals:

$$\begin{aligned} z &= [-1 \quad 0] \begin{bmatrix} \alpha \\ q \end{bmatrix} \\ w &= \Lambda z \end{aligned}$$

where $\Lambda \in [-1 \; 1]$ is a normalized parameter uncertainty [38].

TABLE I
POLES AND PERFORMANCE OF THE LINEAR CONTROLLER $K_2(s)$ WITH $\zeta_2 = 0°$

H_∞ **performance**	$\gamma = 0.802$
RMS gain	6.9392
$K_2(s)$ **poles**	-2.2
	$-1.3 \cdot 10^2 \pm i0.9 \cdot 10^2$
	$-5.82 \cdot 10^2$
	$-1.32 \cdot 10^3$
	$1.28 \cdot 10^3$

In addition to being closed-loop stable in order to stabilize the airframe rotational motion, the neural network controller must meet the following specifications, borrowed from [1], [38]:

I) track the step command in vertical acceleration η_c with zero steady-state error;
II) track η_c with minimal overshoot and a time constant of 0.2 s;
III) enforce adequate high-frequency rolloff, stability, and L_2 performance for all uncertainties Λ;
IV) achieve differential damping at low and high frequencies to improve transient behavior.

The following performance objectives, taken from [1], are used to meet the above specifications:

i) introduce an integrator with state $\xi = \int_0^t (\eta_c - \eta(\tau))d\tau$;
ii) use the following shaping filters for signals δ and ξ:

$$W_\delta = \frac{10^{-3}s^3 + 0.03s^2 + 0.3s + 1}{10^{-5}s^3 + 310^{-2}s^2 + 30s + 10^4} \quad W_\xi = 0.8;$$

iii) minimize the L_2 gain of the mapping from the inputs (w, η_c) to the outputs (z, ξ, δ);
iv) place the closed-loop poles within an LMI region that is the intersection of $\mathcal{D}_1 = \{z \in \mathbb{C} : \varphi_1(z) < 0\}$ and $\mathcal{D}_2 = \{z \in \mathbb{C} : \varphi_2(z) < 0\}$, where

$$\varphi_1(z) = \begin{bmatrix} -r & \bar{z} + c \\ z + c & -r \end{bmatrix}, \qquad c = 0, \qquad r = 1500$$
$$\varphi_2(z) = -2\beta \cos\theta z e^{i\theta} + \bar{z} e^{-i\theta}, \qquad \beta = 1, \qquad \theta = 30°.$$

As shown in [1], the multiple objectives listed above can be met by a linear dynamic controller obtained by solving a set of LMIs. In this paper, the approach in [1] and the LMI software in [41] are used to obtain the LTI controllers $\mathcal{K} = \{K_1, K_2, K_3\}$ for the set $\mathcal{Z} = \{-10°, 0°, 12°\}$ chosen for the given angle-of-attack range. As shown in Table I, the properties of $K_2(s)$ are equivalent to those achieved in [1]. Similar properties are also obtained for the other controllers in $\mathcal{K}$, which are used for neural network control synthesis, as explained below.

The synthesis procedure presented in Section VI is applied to design a neural network controller for (41) that meets multiple performance objectives i)–iv) above and is closed-loop stable. In the classical LMI design [1], a linear dynamic controller in the form (3) maps the tracking error signal ξ to the control δ to meet the zero-error tracking objective I). As shown in [24] and [52] and reviewed in [42, p. 522], a common and effective approach for including the integrator described in i) is to augment the plant state equation by the integral-state variable ξ. Thus, the neural network input is $\chi := [x^T \; x_K^T \; \xi]^T$. Let the transfer function $P(s, \zeta) = \mathcal{C}(\zeta)[sI - \mathcal{A}(\zeta)]^{-1}\mathcal{B}(\zeta)$ map the input signals (δ, η_c, w) to the outputs (χ, z), where matrices $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ are defined in Appendix VII. Then, the missile neural network controller, schematized in Fig. 4, takes the form in (10), with $x_K \in \mathbb{R}^{6\times1}$, and $l = p = 3$. Finally, the neural control parameters v and W are computed such that the matrix equalities (17), the inequality (32), and the LMI (34) are all satisfied.

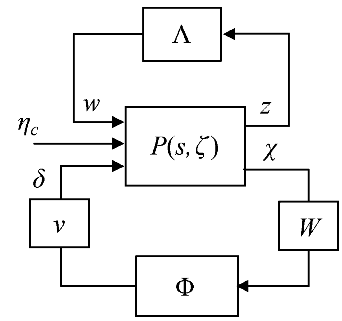

Fig. 4. Sigmoidal neural control structure for missile dynamics with parameter uncertainty Λ.

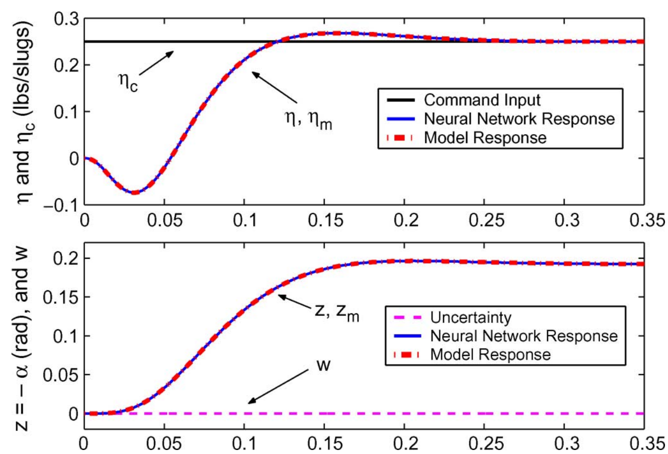

Fig. 5. Neural network controlled system identically overlaps that of the ideal model, achieving perfect model following and zero tracking error.

The ability of the neural network control system to track a step command with zero error, minimal overshoot, and a time constant of 0.2 s is illustrated in Fig. 5. Here, an increase in vertical acceleration of 0.25 lb/slugs is commanded under zero parameter uncertainty, and η_c is perfectly tracked after approximately 0.25 s, which is appropriate for a highly maneuverable missile (as shown in [1], [38], [50], and [51]). The time history of the neural network output δ and internal state rate $\dot{x}_K$ for this maneuver are plotted in Fig. 6. The dynamic neural network controller also achieves zero tracking error in the presence of parameter uncertainty, as demonstrated in Fig. 7, where a decrease in vertical acceleration of 0.3 lb/slugs is commanded in the presence of a constant parameter uncertainty $w = -0.15$ rad [normalized as shown in (42)]. Even in this case, η_c is perfectly tracked with minimal overshoot after approximately 0.25 s by the neural network controller with time evolutions plotted in Fig. 8. The time histories of the angle of attack, plotted in Figs. 5–7 (where $z = -\alpha$) also show that in both cases the neural network controller exhibits

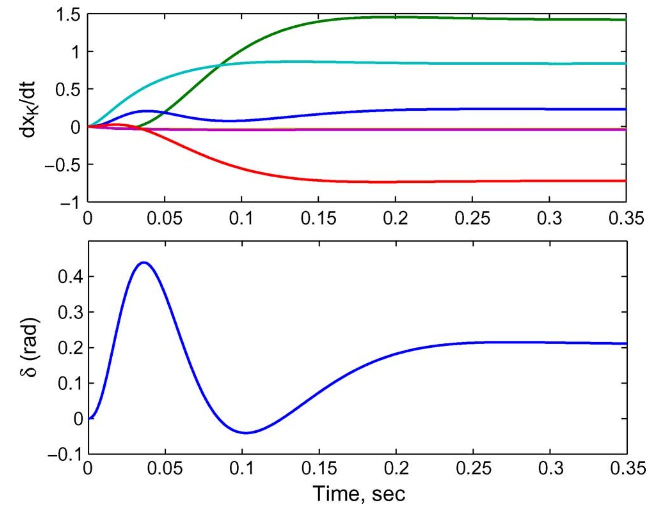

Fig. 6. Time history of the neural network controller's state derivative $\dot{x}_K$ and output δ, for the maneuver illustrated in Fig. 5.

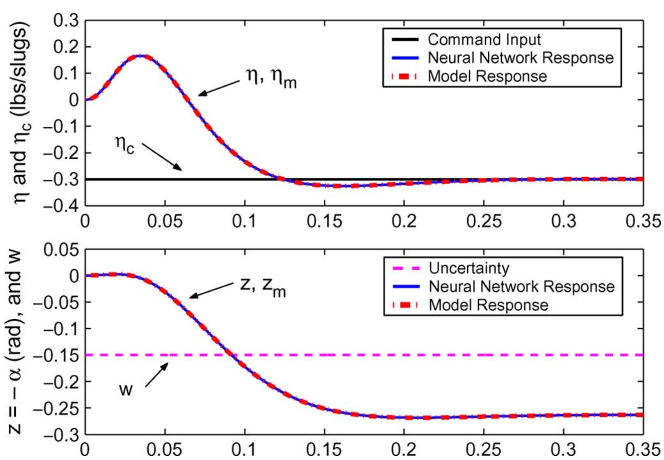

Fig. 7. Neural network controlled system achieves perfect model following and zero tracking error in the presence of parametric uncertainty.

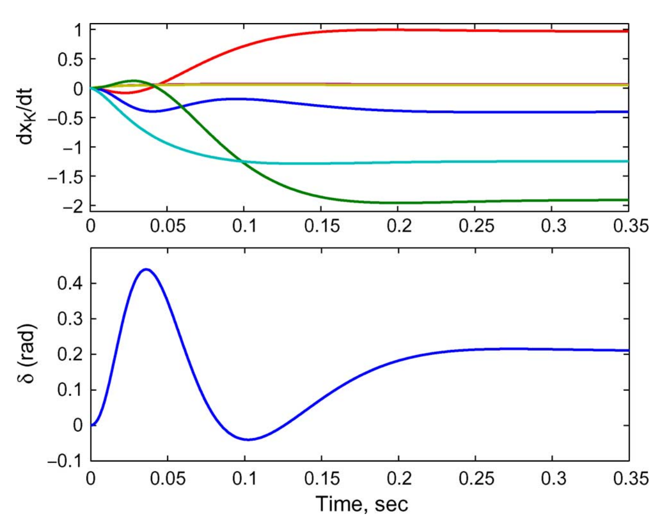

Fig. 8. Time history of the neural network controller's state derivative $\dot{x}_K$ and output δ, for the maneuver illustrated in Fig. 7.

perfect model following at all times, even when the missile operates away from the set points in $\mathcal{Z}$.

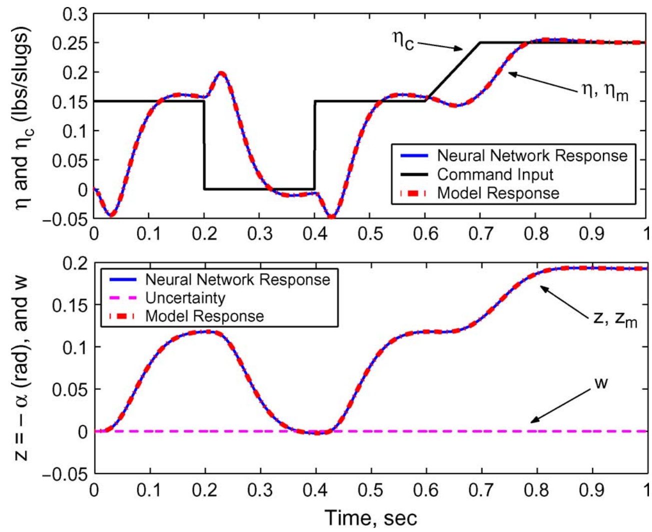

Fig. 9. Response of the neural-network-controlled system is overlapped by that of the ideal model, revealing perfect model following, for a time-varying command input (η_c) and no uncertainty ($w = 0$).

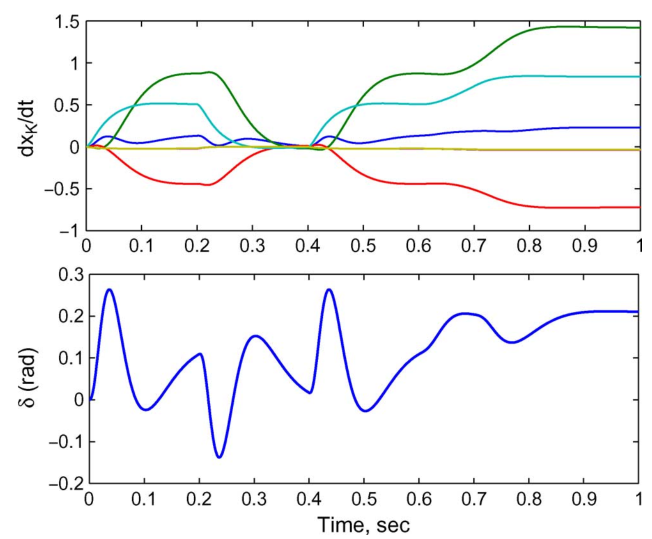

Fig. 10. Time history of the neural network controller's state derivative $\dot{x}_K$ and output δ, for the maneuver illustrated in Fig. 9.

The time-varying command input considered in Fig. 9 purposely prevents the missile from reaching a steady state. This type of command may come about during a task that involves tracking a nonlinear trajectory produced by an outer-loop guidance system. The response of the neural network controlled system with $w = 0$ is plotted in Fig. 9 and compared to that obtained from the ideal model (dashed–dotted line). The time histories of the corresponding neural control output δ and internal state rate $\dot{x}_K$ are plotted in Fig. 10. It can be observed that, even when the controller's state rates are not zero (Fig. 10) and the plant is operating away from any set point, the nonlinear controller achieves *perfect* model following, and its closed-loop state and output response identically overlaps that of the model at all times (Fig. 9). Similar results are obtained in the presence of a time-varying parametric uncertainty w plotted by a dashed line in Fig. 11. The closed-loop response comparison is plotted in Fig. 11, and the neural controller evolutions are

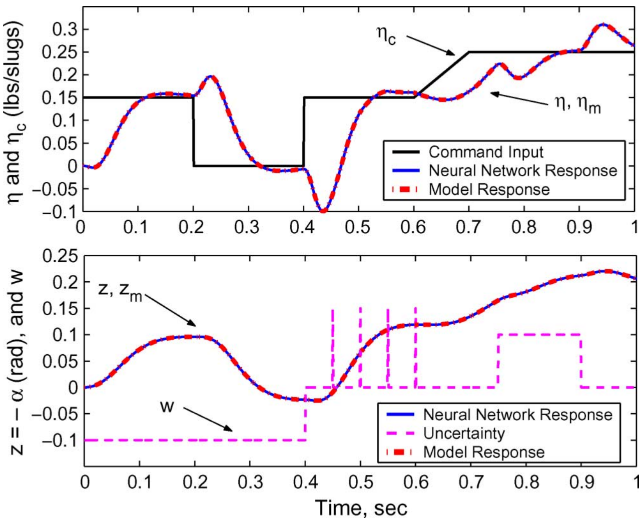

Fig. 11. Response of the neural-network-controlled system is overlapped by that of the ideal model, revealing perfect model following for time-varying command input (η_c) and uncertainty (w).

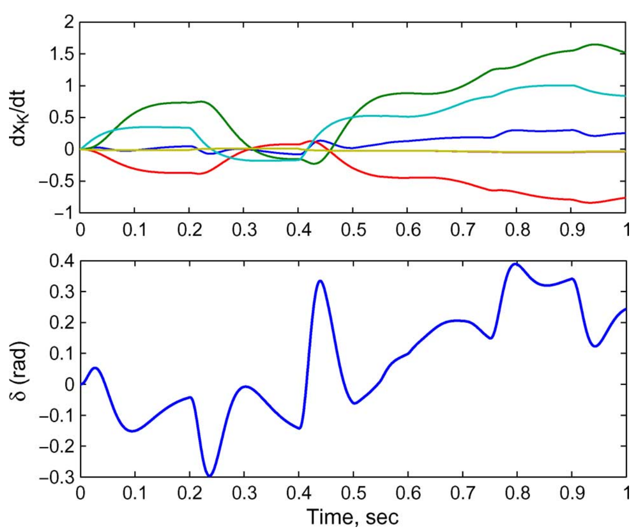

Fig. 12. Time history of the neural network controller's state derivative $\dot{x}_K$ and output δ, for the maneuver illustrated in Fig. 11.

plotted in Fig. 12. It can be observed that closed-loop stability and perfect model following are achieved both in the presence of time-varying uncertainties and command inputs that vary on a time scale, which is smaller than the missile's settling time.

VIII. Conclusion

The advantages brought about by using linear controllers in conjunction with neural approximators have long been recognized in the literature. In particular, using linear controllers to obtain the starting neural control design has been shown to be a key step in the development of very effective adaptive neural controllers. On the other hand, neural controllers are often criticized for not providing the same guarantees of closed-loop stability, performance, and robustness as classical linear designs even when the system is operating in a linear regime. This is a key-limiting factor to the replacement of linear controllers by neural controllers in applications such as the control of flight or power systems, where safety is a key concern and the system to be controlled often operates in a linear regime. Consequently, this paper addresses the specification and design of an output-feedback dynamic neural controller that meets multiple design objectives, including mixed H_2/H_∞ performance objectives and closed-loop exponential stability. Using algebraic training and nonlinear IMF, conditions are given for the origin of the generalized error dynamics to be exponentially stable. Closed-loop stability of the neural network control system is guaranteed by means of LMIs derived from existing IQCs for diagonal operators with repeated monotonic and slope-restricted nonlinearities. As demonstrated by a missile control problem, these theoretical results can be used to synthesize neural network control systems that have the same performance and stability guarantees as classical linear designs.

Appendix I
Proof of Theorem 1

Proof: The closed-loop dynamics of the linearly controlled system for all $\zeta_j \in \mathcal{Z}$ are obtained from (1) and (3)

$$\begin{cases} \dot{x}_m = A_j x_m + B_j \left[C_{K_j} x_{K_m} + D_{K_j} y\right] \\ \dot{x}_{K_m} = A_{K_j} x_{K_m} + B_{K_j} y \\ y = C_j x_m + D_j \left[C_{K_j} x_{K_m} + D_{K_j} y\right] \end{cases}, \qquad j = 1, \ldots, p. \tag{42}$$

The above closed-loop dynamic equation must be considered in computing the right-hand side of requirement (16). From (14), the closed-loop output can be expressed as

$$\begin{aligned} y &= \left(I - D_j D_{K_j}\right)^{-1} \left[C_j \quad D_j C_{K_j}\right] \chi_m \\ &:= M_{1_j} \chi_m, \qquad j = 1, \ldots, p \end{aligned} \tag{43}$$

where $\chi_m := \left[x_m^T \; x_{K_m}^T\right]^T$, provided $(I - D_j D_{K_j})$ is invertible for all j. Thus, y can be eliminated from the state and control equations, such that

$$\begin{aligned} u_K(\zeta_j) &= \left\{\left[0 \quad C_{K_j}\right] + D_{K_j} M_{1_j}\right\} \chi_m \\ &:= M_{2_j} \chi_m, \qquad j = 1, \ldots, p \end{aligned} \tag{44}$$

and

$$\begin{aligned} \dot{\chi}_m &= \begin{bmatrix} \left[A_j \quad B_j C_{K_j}\right] + B_j D_{K_j} M_{1_j} \\ \left[0 \quad A_{K_j}\right] + B_{K_j} M_{1_j} \end{bmatrix} \\ &:= A_{a_j} \chi_m, \qquad j = 1, \ldots, p. \end{aligned} \tag{45}$$

From (44) and (45), when $\zeta(t) = \zeta_j$, the derivative of the linear control law (3) with respect to the state rate is

$$\frac{\partial u_K}{\partial \dot{\chi}_m}(\zeta_j) = \frac{\partial u_K}{\partial \chi_m} \frac{\partial \chi_m}{\partial \dot{\chi}_m}\bigg|_{\zeta_j} = M_{2_j} \left(A_{a_j}\right)^{-1}, \qquad j = 1, \ldots, p \tag{46}$$

assuming A_{a_j} is invertible for all j.

From (10), the derivative of the neural network control law with respect to $\dot{\chi}$ is given by

$$\frac{\partial u_N}{\partial \dot{\chi}}(\zeta_j) = \Phi' \left[W \chi(\zeta_j)\right]^T V W_\chi \left(A_{a_j}\right)^{-1}, \qquad j = 1, \ldots, p \tag{47}$$

where $V = \mathrm{diag}(v)$, and W_χ are parameters to be determined. Now, assume that the scheduling vector can be obtained from the augmented state through a known linear transformation $\zeta = H\chi$, which can always be accomplished by properly defining χ.

Then, ζ in Fig. 3 can be assimilated into χ by postmultiplying W_ζ by H. At every one of the p set points, where $\zeta(t) = \zeta_j$, the state rate and the control vanish, i.e., $\dot{\chi}(\zeta_j) = 0_{\nu\times 1}$ and $u_K(\zeta_j) = 0$ for all j. Therefore, the derivative in (47) can be simplified to

$$\frac{\partial u_N}{\partial \dot{\chi}}(\zeta_j) = \Phi'(W_\zeta \zeta_j)^T V W_\chi \left(A_{a_j}\right)^{-1}, \qquad j = 1, \dots, p. \tag{48}$$

The neural network control law (10) satisfies the requirements (16) when $A_K(\zeta)$ and $B_K(\zeta)$ are obtained from convex interpolation of the LTI control matrices (4), as in (3), and the adjustable parameters v and W satisfy the sets of equations

$$\begin{cases} n^j = W_\zeta \zeta_j \\ u_N(\zeta_j) = v\Phi(n^j) = 0 \\ \frac{\partial u_N}{\partial \dot{\chi}}(\zeta_j) = \Phi'(n^j)^T V W_\chi = M_{2_j} \end{cases}, \qquad j = 1, \dots, p. \tag{49}$$

Through simple manipulations, the above equations can be written in a matrix form

$$\begin{bmatrix} v\Phi(n^1) \\ \vdots \\ v\Phi(n^p) \end{bmatrix} = \begin{bmatrix} \Phi(n^1)^T \\ \vdots \\ \Phi(n^p)^T \end{bmatrix} v^T := S v^T = 0 \tag{50}$$

and

$$\begin{bmatrix} \Phi'(n^1)^T V W_\chi \\ \vdots \\ \Phi'(n^p)^T V W_\chi \end{bmatrix} = D V W_\chi = M_2 \tag{51}$$

where D is defined in (24), finally obtaining the linear systems in (17). Since $l = p$, these systems have as many unknowns as there are equations and, when approached in the order provided, they also are linear. ■

Appendix II
Proof of Theorem 2

Proof: The error state equation is obtained by subtracting (14) from (13), according to the error definition (15), so that the generalized error simplifies to (52) shown at the bottom of the page, since $y_m = Cx_m + Du_m$, and u_m can be written in terms of χ_m using (44). Throughout this proof, the argument ζ is omitted from the state–space matrices for brevity. Differentiating both sides with respect to time the resulting differential equation is

$$\dot{e}_r = A_{cl}(\dot{\chi}_{cl} - \dot{\chi}_m) + B_{cl}\Phi'(W\chi_{cl})^T V W \dot{\chi}_{cl} - B_{cl} M_2 \dot{\chi}_m \tag{53}$$

where

$$A_{cl} := \begin{bmatrix} A & 0 \\ B_K C & A_K \end{bmatrix} \quad \text{and} \quad B_{cl} := \begin{bmatrix} B_u \\ B_K D \end{bmatrix} \tag{54}$$

and $\dot{\zeta} \approx 0$ [53]. Since from (15) $e_r = \dot{\chi}_{cl} - \dot{\chi}_m$, the above differential equation can be written solely with respect to the model state, with the exception of the argument of the diagonal operator Φ, i.e.,

$$\begin{aligned} \dot{e}_r = A_{cl} e_r - B_{cl} M_2 \dot{\chi}_m + B_{cl}\Phi'(W\chi_{cl})^T V W e_r \\ - B_{cl}\Phi'(W\chi_{cl})^T V W \dot{\chi}_m. \end{aligned} \tag{55}$$

Assuming that $\dot{w}$ is negligibly small, the model can be differentiated with respect to time, obtaining $\ddot{\chi}_m = A_a \dot{\chi}_m$, where

$$A_a := \begin{bmatrix} [A \quad B_u C_K] + B_u D_K M_1 \\ [0 \quad A_K] + B_K M_1 \end{bmatrix} \tag{56}$$

with vertices given by (45). Therefore, (55) can be written as the sum of a nominal linear system and a nonlinear perturbation that are both formulated with respect to the augmented error variable e, as shown in (29), whereas χ_{cl} is confined to the argument of the diagonal operator derivative Φ' in (29).

We are now ready to show that any set point of the closed-loop system is a zero-equilibrium of the error (29). Given a desired set point (defined in [42, p. 508]), the corresponding equilibrium values of the plant state and control may be nonzero, and depend on the coefficient of the plant dynamics (2), as well as on the definition of the command or reference vector. Since the error state equation can be written as the perturbed system (29), it can be seen that $\dot{e} = 0$ when $e = 0$. However, we also want to show that for any set point the nonlinear term

$$\begin{aligned} g(t,e) &:= \begin{bmatrix} B_{cl}\Phi'\left(n(t)\right)^T VW \\ 0 \end{bmatrix} [I \quad -I] e \\ &= B_{cl}\Phi'(W\chi_{cl})^T V W \dot{\chi}_{cl} - B_{cl} M_2 \dot{\chi}_m \end{aligned} \tag{57}$$

is a vanishing perturbation, i.e., $g^*(t,0) = 0$ for all t. Assume that the controller parameters satisfy the systems in (17). By design, at any set point scheduled by ζ_j, $W\chi_{cl}^* = W_\zeta \zeta_j = n^j$. Thus, from Theorem 1 and (57), the nonlinear perturbation at the set point can be simplified to

$$g^*(t,e) = B_{cl}\Phi'(n^j)^T V W_\chi \dot{\chi}_{cl} - B_{cl} M_2 \dot{\chi}_m \tag{58}$$

$$= B_{cl} M_2 \dot{\chi}_{cl} - B_{cl} M_2 \dot{\chi}_m = B_{cl} M_2 e_r. \tag{59}$$

Therefore, the set-point perturbation is said to be a vanishing perturbation because it equals zero at the zero equilibrium or origin $e = 0$, where $e_r = 0$, which corresponds to perfect model following and zero error rate.

It can be assumed that $A_e(\zeta)$ is Hurwitz for all $\zeta = \sum_{j=1}^{p} c_j \zeta_j$, since from Section III $A_e(\zeta) = \sum_{j=1}^{p} c_j A_{e_j}$, and A_{e_j} in (28) is Hurwitz by design for all j. Then, $e = 0$ is an exponentially stable equilibrium point of the nominal system $\dot{e} = A_e e$, and from [54, Th. 4.6], the Lyapunov equation

$$P_e A_e(\zeta) + A_e^T(\zeta) P_e = -Q \tag{60}$$

with $Q = \sum_{l=1}^{p} Q_l$, and $Q_l = Q_l^T > 0$, has a unique solution $P_e = P_e^T > 0$. Additionally, it can be shown that the quadratic

$$e_r = \begin{bmatrix} A(x_{cl} - x_m) + B_u v\Phi(W\chi_{cl}) - B_u M_2 \chi_m \\ A_K\left(x_K - x_{K_m}\right) + B_K C(x_{cl} - x_m) + B_K D v \Phi(W\chi_{cl}) - B_K D M_2 \chi_m \end{bmatrix} \tag{52}$$

Lyapunov function $V(e) = e^T P_e e$ satisfies the inequalities in the converse Lyapunov theorem [54, Th. 4.14], i.e.,

$$c_1\|e\|^2 \leq V(e) \leq c_2\|e\|^2 \tag{61}$$
$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial e} f_e(t,e) \leq -c_3\|e\|^2 \tag{62}$$
$$\left\|\frac{\partial V}{\partial e}\right\| \leq c_4\|e\| \tag{63}$$

by choosing the following four positive constants: $c_1 = \lambda_{\min}(P_e)$, $c_2 = \lambda_{\max}(P_e)$, $c_3 = \lambda_{\min}(Q)$, and $c_4 = 2c_2$, where f_e denotes the right-hand side of the nonlinear state (29), thus $\dot{e} = f_e(t,e)$. In this paper, we choose $Q = I$, since it can be shown that this choice always maximizes the ratio $\lambda_{\min}(Q)/\lambda_{\max}(P_e)$ (e.g., see [54, p. 372]) that is used in the upper bound below.

The bounds for exponential stability of the origin of the error state (29) are derived using [54, Corollary 9.1] on vanishing perturbations. It has been shown above that the nominal system has an exponentially stable equilibrium at origin $e = 0$, and a Lyapunov function $V(e)$ that satisfies the inequalities in (61). Then, if the vanishing perturbation $g(t,e)$ satisfies the inequality

$$\|g(t,e)\| \leq \phi\|e\| \tag{64}$$

where ϕ is a constant that satisfies the bound $\phi < c_1c_3/c_2c_4$, the origin is an exponentially stable equilibrium of the nonlinear system (29), by virtue of the comparison method [54]. In the remainder of the proof, $\|\cdot\|$ denotes the $\mathcal{L}_2$ norm. Having reduced the error state equation to (29), the norm of the nonlinear perturbation can be written as

$$\|g(t,e)\| = \left\|B_{cl}\Phi'(n(t))^T VW[I \quad -I]e(t)\right\|. \tag{65}$$

From the property of the Hölder norms [55, Corollary 9.3.4], and using $\|[I - I]\|_2 = \sqrt{2}$, the following inequalities are obtained:

$$\begin{aligned}\|g(t,e)\| &\leq \left\|B_{cl}\Phi'(n(t))^T VW[I \quad -I]\right\| \|e(t)\| \\ &\leq \left\|B_{cl}\Phi'(n(t))^T VW\right\| \|[I \quad -I]\| \|e(t)\| \\ &= \sqrt{2}\left\|B_{cl}\Phi'(n(t))^T VW\right\| \|e(t)\|.\end{aligned} \tag{66}$$

Since the norm of a matrix equals the norm of its transpose, it follows that:

$$\begin{aligned}\Psi(t) := \left\|B_{cl}\Phi'(n(t))^T VW\right\| &= \left\|W^TV^T\Phi'(n(t))B_{cl}^T\right\| \\ &\leq \left\|W^TV^T\Phi'(n(t))\right\| \left\|B_{cl}^T\right\| \\ &= \|B_{cl}\| \left\|W^TV^T\Phi'(n(t))\right\| | \\ &\leq \|B_{cl}\|\|VW\| \|\Phi'(n(t))\|.\end{aligned} \tag{67}$$

By definition, the diagonal operator Φ contains repeated slope-restricted nonlinearities, such that $0 \leq \sigma'(n(t)) \leq \beta$ for all $n(t)$, with $\beta = 1/2$ in this case (Section IV). Thus, it can be easily shown that $\|\Phi'(n(t))\|_2 \leq \beta\sqrt{l}$, where l is the number of nonlinearities. Then, it follows that:

$$\Psi(t) \leq \beta\sqrt{l}\|B_{cl}\|\|VW\| \tag{68}$$

and

$$\|g(t,e)\| \leq \sqrt{2l}\beta\|B_{cl}\|\|VW\|\|e\| =: \phi\|e\| \tag{69}$$

where ϕ is a constant. Thus, the origin $e = 0$ of the error state (29) is exponentially bounded provided

$$\phi = \sqrt{2l}\beta\|B_{cl}\|\|VW\| < \frac{1}{2\lambda_{\max}(P_e)} \tag{70}$$

since the bound can be simplified by noting $c_1/c_2 \leq 1 \rightarrow \phi < c_3/c_4$.

Although this first inequality is educational, it tends to be conservative and may be difficult to satisfy in practice. Another bound is obtained by calculating the derivative of the quadratic Lyapunov function $V(e)$ along the trajectories of the nonlinear system (29)

$$\dot{V}(e) = -\|e\|_2^2 + 2e^TP_eg(t,e) \tag{71}$$

thus exploiting the structure of the perturbation $g(t,e)$ instead of the bound $\phi < c_3/c_4$. The bound is derived by requiring $\dot{V}(e)$ to be negative definite. Then, from the theorem on the Lyapunov stability of nonautonomous systems (see [54, Th. 4.10]) and substituting (57) in (71), such that

$$\begin{aligned}\dot{V}(e) &= -\|e\|_2^2 + 2e^TP_e\begin{bmatrix}B_{cl}\Phi'(n(t))^T VW \\ 0\end{bmatrix}[I \quad -I]e \\ &= -\|e\|_2^2 + 2e^T\Upsilon(t)e\end{aligned} \tag{72}$$

it follows that the origin $e = 0$ is exponentially stable. If $e^T\Upsilon(t)e < 0$ for all t and e, then $\dot{V}(e) < 0$, where

$$\Upsilon(t) := P_e\begin{bmatrix}B_{cl}\Phi'(n(t))^T VW \\ 0\end{bmatrix}[I \quad -I] \tag{73}$$
$$= \begin{bmatrix}P_{11} \\ P_{21}\end{bmatrix}B_{cl}\Phi'(n(t))^T VW[I \quad -I] \tag{74}$$

is an outer product matrix. Using Corollary 1 in Appendix III and its proof, if the following inequality holds:

$$\begin{aligned}&\Phi'(n(t))^T VW[I \quad -I]\begin{bmatrix}P_{11} \\ P_{21}\end{bmatrix}B_{cl} \\ &\quad = \Phi'(n(t))^T VW(P_{11} - P_{21})B_{cl} < 0\end{aligned} \tag{75}$$

then $e^T\Upsilon(t)e < 0$ for all t and e. The left-hand side of the above inequality is the inner product of a vector $\Phi'(n(t))^T$, whose elements are always nonnegative, with a vector that is constant with respect to time. Thus, the above inner product is always negative provided all elements of the constant vector $VW(P_{11} - P_{21})B_{cl}$ are negative, i.e.,

$$\{VW(P_{11} - P_{21})B_{cl}\}_i < 0, \qquad \text{for} \quad i = 1, \ldots, l \tag{76}$$

where $B_{cl} = \sum_{j=1}^{p} c_jB_{cl}(\zeta_j)$ and all $B_{cl}(\zeta_j)$ are known from (54), which concludes the proof. ■

APPENDIX III

Corollary 1: Let M be an outer product matrix, such that $M = xy^T \in \mathbb{R}^{n\times n}$, and $x, y \in \mathbb{R}^n$. Then, $M \geq 0$ if and only if $y^Tx \geq 0$, and $M \leq 0$ if and only if $y^Tx \leq 0$.

Proof: Using [55, Fact 2.11.11], the equality

$$(xy^T)^r = (y^T x)^{r-1} xy^T \tag{77}$$

holds for any two vectors $x, y \in \mathbb{R}^n$. Letting $r = 2$

$$M^2 = (xy^T)^2 = (y^T x)(xy^T) = (y^T x)M. \tag{78}$$

It can be easily shown that $M^2 \geq 0$ [56, p. 258], therefore, by definition, $z^T M^2 z \geq 0$ for all $z \in \mathbb{R}^n$. Using (78) and noting that the inner product $y^T x$ is a scalar, it follows that:

$$z^T M^2 z = z^T (y^T x) M z = (y^T x) z^T M z \geq 0, \qquad \text{for } \forall z \in \mathbb{R}^n. \tag{79}$$

Then, $M \geq 0$, or $z^T M z \geq 0$ for all $z \in \mathbb{R}^n$, if and only if $y^T x \geq 0$. Conversely, $M \leq 0$, or $z^T M z \leq 0$ for all $z \in \mathbb{R}^n$, if and only if $y^T x \leq 0$ ■

APPENDIX IV
PROOF OF PROPOSITION 1

Proof: For the neural-network-controlled-system (26), the stability inequality (9) gives

$$\mathcal{M} := \begin{bmatrix} A_{cl}(\zeta_j)^T Q_j + Q_j A_{cl}(\zeta_j) & Q_j B_{cl}(\zeta_j) v + \beta W^T T_j \\ [Q_j B_{cl}(\zeta_j) v]^T + \beta T_j W & -2T_j \end{bmatrix} < 0, \quad j = 1, \ldots, p \tag{80}$$

where $T_j \in \mathbb{R}^{l \times l}$ and $Q_j \in \mathbb{R}^{\nu \times \nu}$ are LMI variables that need not be positive definite. If the network parameters v and W are known, the above inequality is linear with respect to T_j and Q_j. However, it is nonlinear if v and W are unknown as, for example, during the control synthesis and adaptation processes.

Using (17), a new and less conservative inequality is obtained from (80). If $A_{cl}(\zeta_j) \in \mathbb{R}^{\nu \times \nu}$ is asymptotically stable, then for any positive–definite matrix $R_i \in \mathbb{R}^{\nu \times \nu}$, the Lyapunov equation

$$A_{cl}(\zeta_j)^T Q_{i_j} + Q_{i_j} A_{cl}(\zeta_j) = -R_{i_j}, \qquad i = 1, 2 \tag{81}$$

has a positive–definite solution $Q_{i_j} \in \mathbb{R}^{\nu \times \nu}$, and if R_{i_j} is symmetric, then Q_{i_j} is symmetric [55, Sec. 11.8]. Let $Q_j = \sum_i Q_{i_j}$, where Q_{i_j} are solutions of (81) corresponding to symmetric and positive–definite matrices R_i. Then, it can be easily shown that Q_j also is a symmetric positive–definite matrix that satisfies (81) when the right-hand side is equal to $R_j = \sum_i R_{i_j} > 0$. It follows that (80) can be written as the sum of two matrices, shown in the first equation at the bottom of the page. $\mathcal{M}_1$ is in a special form that is always negative definite provided $-R_{1_j} < 0$ (see [55, Fact 8.9.9]). Since $R_{1_j} > 0$ by definition, $\mathcal{M}_1 < 0$ for any value of Q_j, $B_{cl}(\zeta_j)$, and v. Thus, (80) can be replaced by a less conservative inequality $\mathcal{M}_2 \leq 0$, leading to (34).

Next, since $-R_{2_j} < 0$, it also follows that $\mathcal{M}_2 \leq 0$ is equivalent to the statement

$$-2T_j \leq [QB_{cl}(\zeta_j)v]^T \left(-R_{1_j}\right)^{-1} [QB_{cl}(\zeta_j)v] + \beta^2 T_j W \left(-R_{2_j}\right)^{-1} W^T T_j \tag{82}$$

from a well-known result on positive–semidefinite matrices [55, Prop. 8.2.3]. And, similarly, since $-R_j < 0$, the statement $\mathcal{M} < 0$ in the original stability LMI (80) is equivalent to the statement

$$-2T_j \leq \left[(Q_j B_{cl}(\zeta_j)v)^T + \beta T_j W\right]^T (-R_j)^{-1} \times \left[(Q_j B_{cl}(\zeta_j)v)^T + \beta T_j W\right]. \tag{83}$$

Now, suppose the equality holds in the new bound (82), then (84) shown at the bottom of the page holds, and it can be shown that if $-R_{1_j} < 0$ and $-R_{2_j} < 0$, then $\mathcal{M} \leq 0$ (see [55, Prop. 8.9.13], and references therein). Furthermore

$$\left[(Q_j B_{cl}(\zeta_j)v)^T + \beta T_j W\right]^T (-R_j)^{-1} \left[(Q_j B_{cl}(\zeta_j)v)^T + \beta T_j W\right] \leq [Q_j B_{cl}(\zeta_j)v]^T \left(-R_{1_j}\right)^{-1} [Q_j B_{cl}(\zeta_j)v] + \beta^2 T_j W \left(-R_{2_j}\right)^{-1} W^T T_j. \tag{85}$$

Since the left-hand side of the above inequality is equal to the right-hand side of the inequality (83), which is obtained from the original LMI problem (80), it follows that the new inequality (82) is less conservative than the former, and the statement in (35) holds when $\mathcal{M}_2 \leq 0$. ■

APPENDIX V
PROOF OF PROPOSITION 2

Proof: It can be shown that for any two symmetric matrices $X, Y \in \mathbb{R}^{n \times n}$, if $SXS^T \leq SYS^T$ and $\text{rank}(S) = n$, then $X \leq Y$ [55, p. 264]. It follows that if the inequality

$$-2ST_j S^T \leq -S [Q_j B_{cl}(\zeta_j)v]^T \left(R_{1_j}\right)^{-1} Q_j B_{cl}(\zeta_j) v S^T - \beta^2 S T_j W \left(R_{2_j}\right)^{-1} W^T T_j S^T \tag{86}$$

holds for the nonsingular matrix $S \in \mathbb{R}^{p \times p}$ defined in (23), then $\mathcal{M}_2 \leq 0$ is satisfied because $-R_{2_j} < 0$. Using (17), the term vS^T can be replaced by the known vector b^T. From Theorem 1

$$W = [V^{-1} D^{-1} M_2 \quad NZ^+] \tag{87}$$

$$\mathcal{M} = \mathcal{M}_1 + \mathcal{M}_2 := \begin{bmatrix} -R_{1_j} & Q_j B_{cl}(\zeta_j) v \\ [Q_j B_{cl}(\zeta_j) v]^T & [Q_j B_{cl}(\zeta_j) v]^T (-R_1)^{-1} Q_j B_{cl}(\zeta_j) v \end{bmatrix} + \begin{bmatrix} -R_{2_j} & \beta W^T T_j \\ \beta T_j W & -2T_j + [Q_j B_{cl}(\zeta_j) v]^T (R_1)^{-1} [Q_j B_{cl}(\zeta_j) v]^T \end{bmatrix}$$

$$\mathcal{M} = \begin{bmatrix} -R_{1_j} - R_{2_j} & \left[(Q_j B_{cl}(\zeta_j) v)^T + \beta T_j W\right]^T \\ (Q_j B_{cl}(\zeta_j) v)^T + \beta T_j W & (Q_j B_{cl}(\zeta_j) v)^T (-R_1)^{-1} Q_j B_{cl}(\zeta_j) v + \beta T_j W (-R_2)^{-1} (\beta T_j W)^T \end{bmatrix} \tag{84}$$

where Z defined in (21) is a known matrix that is right-invertible by design, and Z^+ denotes its generalized inverse. For convenience, define the matrix

$$\Pi := \left[(DVS^T)^{-1}M_2 \quad S^{-T}NZ^+\right] \tag{88}$$
$$= \left[\left\{D\left[(S^{-1}b)\otimes \imath_p^T\right]\circ I_pS^T\right\}^{-1}M_2 \quad S^{-T}NZ^+\right] \tag{89}$$

where $\otimes$ and $\circ$ denote the Kronecker and Schur products, respectively, I_p is a $p\times p$ identity matrix, and $\imath_p := 1_{p\times 1}$. Then, $W = S^T\Pi$, and using the change of variables $P_j = ST_jS^T$, the inequality (86) can be simplified to

$$-2P_j \le -\left[Q_jB_{cl}(\zeta_j)b^T\right]^T R_{1_j}^{-1}Q_jB_{cl}(\zeta_j)b^T - \beta^2 P_j\Pi R_{2_j}^{-1}\Pi^T P_j^T \tag{90}$$

thereby confining the neural controller parameters to Π. Let $\mathcal{Q}_j := Q_jB_{cl}(\zeta_j)b^T$ denote in short a known and constant matrix that is independent of v and W. Then, (90) is equivalent to

$$\begin{bmatrix} -\left[\Pi R_{2_j}^{-1}\Pi^T\right]^{-1} & \beta P_j^T \\ \beta P_j & -2P_j + \mathcal{Q}_j^T R_{1_j}^{-1}\mathcal{Q}_j \end{bmatrix} \le 0 \tag{91}$$

provided the matrix

$$\mathcal{W} := -\left[\Pi R_{2_j}^{-1}\Pi^T\right]^{-1} \tag{92}$$

is always negative semidefinite, which can be accomplished, for example, by letting $R_{2_j} = I_\nu$. A few matrix manipulations yield

$$\mathcal{W} = \mathcal{I}_2 + \mathcal{I}_2(\mathcal{I}_1 - \mathcal{I}_2)^{-1}\mathcal{I}_2 \tag{93}$$

where

$$\mathcal{I}_1 := \left\{D\left[(S^{-1}b)\otimes \imath_p^T\right]\circ I_pS^T\right\}^T M_6 \times \left\{D\left[(S^{-1}b)\otimes \imath_p^T\right]\circ I_pS^T\right\} \tag{94}$$
$$\mathcal{I}_2 := SN^{-T}M_7N^{-1}S. \tag{95}$$

The matrices $M_6 := (M_2M_2^T)^{-1}$ and $M_7 := \left[Z^+(Z^+)^T\right]^{-1}$ exist and are known by design. Finally, since $Q_j = \sum_i Q_{i_j} = Q_j^T > 0$, if there also exists a diagonally dominant matrix $T_j = T_j^T$ such that $P_j = ST_jS^T$, then by definition of diagonally dominant matrix, it follows that:

$$T_{j_{(\imath\imath)}} \ge \sum_{\jmath=1,\jmath\ne\imath}^{l} \left|T_{j_{(\imath\jmath)}}\right| \quad \imath = 1,\ldots,l \tag{96}$$

where $T_{j_{(\imath\jmath)}}$ denotes the element in the $\imath$th-row and $\jmath$th-column of T_j, and $|\cdot|$ denotes the absolute value. Thus, all of the conditions in (34) are satisfied, and the feedback interconnection (26) is exponentially stable. ■

Appendix VI
Missile Model Functions and Parameters

The nonlinear functions in the missile model (41) are defined as [50]

$$f_1(\alpha) = \frac{QS}{mv}(a_1\alpha^2 + b_1\alpha + c_1)\cos^2\alpha$$
$$f_2(\alpha) = \frac{QSd}{I_y}(a_2\alpha^2 + b_2\alpha + c_2)$$
$$g_1(\alpha) = \frac{QS}{mv}d_1\cos^2\alpha$$
$$h_1(\alpha) = \frac{QS}{m}(a_1\alpha^2 + b_1\alpha + c_1) \tag{97}$$

where Q is the dynamic pressure, S is the reference area, m is the missile mass, I_y is the pitch moment of inertia, v is a constant velocity component along the missile centerline, and d is the airframe diameter. The coefficients $a_1 = 1.03e^{-4}$, $b_1 = -0.00945$, $c_1 = -0.17$, $d_1 = -0.034$, $a_2 = 2.15e^{-4}$, $b_2 = -0.0195$, $c_2 = 0.051$, and $d_2 = -0.206$ are obtained from the polynomial approximation of the missile aerodynamic normal-force and pitching-moment coefficients.

Appendix VII

$$\mathcal{A}(\zeta) := \begin{bmatrix} A(\zeta) & 0 & 0 \\ -B_K(\zeta)M_4C(\zeta) & A_K(\zeta) & B_K(\zeta)M_3 \\ -M_5C(\zeta) & 0 & 0 \end{bmatrix}$$
$$\mathcal{B}(\zeta) := \begin{bmatrix} B_u(\zeta) & 0 & B_w(\zeta) \\ -B_K(\zeta)M_4D(\zeta) & 0 & 0 \\ -M_5D(\zeta) & 1 & 0 \end{bmatrix}$$
$$\mathcal{C} := \begin{bmatrix} I \\ -M_5 \end{bmatrix} \quad M_4 := \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$
$$M_5 := [1 \quad 0] \quad M_3 := M_5^T$$

where $A(\zeta)$, $B_w(\zeta)$, $B_u(\zeta)$, $C(\zeta)$, and $D(\zeta)$ are known from the plant state (1), and $A_K(\zeta)$ and $B_K(\zeta)$ are known from $K(s,\zeta)$ in (3).

References

[1] M. Chilali, P. Ghainet, and P. Apkarian, "Robust pole placement in LMI regions," *IEEE Trans. Autom. Control*, vol. 44, no. 12, pp. 2257–2270, Dec. 1999.

[2] D. Liu and Z. Lu, "A new synthesis approach for feedback neural networks based on the perceptron training algorithm," *IEEE Trans. Neural Netw.*, vol. 8, no. 6, pp. 1468–1482, Nov. 1997.

[3] Y. C. Chu and K. Glover, "Stabilization and performance synthesis for systems with repeated scalar nonlinearities," *IEEE Trans. Autom. Control*, vol. 44, no. 3, pp. 484–496, Mar. 1999.

[4] C. Kwan and F. Lewis, "Robust backstepping control of induction motors using neural networks," *IEEE Trans. Neural Netw.*, vol. 11, no. 5, pp. 1178–1187, Sep. 2000.

[5] Y. Zhang and J. Wang, "Global exponential stability of recurrent neural networks for synthesizing linear feedback control systems via pole assignment," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 633–644, Mar. 2002.

[6] J. D. Bendtsen and K. Trangbaek, "Quasi-LPV control based on neural state-space models," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 355–368, Mar. 2002.

[7] S. Ge and J. Zhang, "Neural-network control of nonaffine nonlinear system with zero dynamics by state and output feedback," *IEEE Trans. Neural Netw.*, vol. 14, no. 4, pp. 900–918, Jul. 2003.

[8] Y. Li, S. Qiang, X. Zhuang, and O. Kaynak, "Robust and adaptive backstepping control for nonlinear systems using RBF neural networks," *IEEE Trans. Neural Netw.*, vol. 15, no. 3, pp. 693–701, May 2004.

[9] H. Gao, J. Lam, and C. Wang, "Induced l_2 and generalized H_2 filtering for systems with repeated scalar nonlinearities," *IEEE Trans. Signal Process.*, vol. 53, no. 11, pp. 4215–4226, Nov. 2005.

[10] J. Mulero-Martínez, "An improved dynamic neurocontroller based on Christoffel symbols," *IEEE Trans. Neural Netw.*, vol. 18, no. 3, pp. 865–879, May 2007.

[11] C. Cao and N. Hovakimyan, "Novel L_1 neural network adaptive control architecture with guaranteed transient performance," *IEEE Trans. Neural Netw.*, vol. 18, no. 4, pp. 1160–1171, Jul. 2007.

[12] Y. Shen and J. Wang, "An improved algebraic criterion for global exponential stability of recurrent neural networks with time-varying delays," *IEEE Trans. Neural Netw.*, vol. 19, no. 3, pp. 528–531, Mar. 2008.

[13] T. Hayakawa, W. Haddad, and N. Hovakimyan, "Neural network adaptive control for a class of nonlinear uncertain dynamical systems with asymptotic stability guarantees," *IEEE Trans. Neural Netw.*, vol. 19, no. 1, pp. 80–89, Jan. 2008.

[14] C. Hua and X. Guan, "Output feedback stabilization for time-delay nonlinear interconnected systems using neural networks," *IEEE Trans. Neural Netw.*, vol. 19, no. 4, pp. 673–688, Apr. 2008.

[15] M. Abu-Khalaf, F. Lewis, and J. Huang, "Neurodynamic programming and zero-sum games for constrained control systems," *IEEE Trans. Neural Netw.*, vol. 19, no. 7, pp. 1243–1252, Jul. 2008.

[16] W. Qiao, R. G. Harley, and G. K. Venayagamoorthy, "Fault-tolerant indirect adaptive neurocontrol for a static synchronous series compensator in a power network with missing sensor measurements," *IEEE Trans. Neural Netw.*, vol. 19, no. 7, pp. 1179–1195, Jul. 2008.

[17] K. S. Narendra and K. Parthasaranthy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Netw.*, vol. 1, no. 1, pp. 4–27, Mar. 1990.

[18] K. S. Narendra, "Adaptive control using neural networks," in *Neural Networks for Control*, W. T. Miller, R. S. Sutton, and P. W. Werbos, Eds. Cambridge, MA: MIT Press, 1990.

[19] J. Neidhoefer and K. Krishnakumar, "Nonlinear control using neural approximators with linear control theory," in *Proc. AIAA Guid. Navigat. Control Conf.*, New Orleans, LA, 1997, pp. 364–372.

[20] E. Jonckheere, G.-R. Yu, and C.-C. Chien, "Gain scheduling for lateral motion of propulsion controlled aircraft using neural networks," in *Proc. Amer. Control Conf.*, Albuquerque, NM, Jun. 1997, pp. 3321–3325.

[21] T. Iwasa, N. Morizumi, and S. Omatu, "Temperature control in batch process by neural networks," in *Proc. IEEE Int. Conf. Neural Netw.*, Houston, TX, 1997, pp. 2430–2433.

[22] M. Agarwal, "Combining neural and conventional paradigms for modelling, prediction and control," *Int. J. Syst. Sci.*, vol. 28, no. 1, pp. 65–81, 1997.

[23] R. Scott and L. E. Pado, "Active control of wind-tunnel model aeroelastic response using neural networks," *J. Guid. Control Dyn.*, vol. 23, no. 6, pp. 1100–1108, 2004.

[24] S. Ferrari and R. Stengel, "Classical/neural synthesis of nonlinear control systems," *J. Guid. Control Dyn.*, vol. 25, no. 3, pp. 442–448, 2002.

[25] G. G. Lendaris and J. C. Neidhoefer, "Guidance in the use of adaptive critics for control," in *Learning and Approximate Dynamic Programming*, J. Si, A. Barto, and W. Powell, Eds. New York: Wiley, 2004.

[26] G. K. Venayagamoorthy, R. G. Harley, and D. C. Wunsch, "Implementation of adaptive critic-based neurocontrollers for turbogenerators in a multimachine power system," *IEEE Trans. Neural Netw.*, vol. 14, no. 5, pp. 1047–1064, Sep. 2003.

[27] S. Ferrari and R. Stengel, "On-line adaptive critic flight control," *J. Guid. Control Dyn.*, vol. 27, no. 5, pp. 777–786, 2004.

[28] U. J. Pesonen, J. E. Steck, K. Rokhsaz, H. S. Bruner, and N. Duerksen, "Adaptive neural network inverse controller for general aviation safety," *J. Guid. Control Dyn.*, vol. 27, no. 3, pp. 434–443, 2004.

[29] R. Rysdyk and A. J. Calise, "Robust nonlinear adaptive flight control for consistent handling qualities," *IEEE Trans. Control Syst. Technol.*, vol. 13, no. 6, pp. 896–910, Nov. 2005.

[30] S. Ferrari and R. Stengel, "Model-based adaptive critic designs," in *Learning and Approximate Dynamic Programming*, J. Si, A. Barto, and W. Powell, Eds. New York: Wiley, 2004.

[31] S. Limanons and J. Si, "Neural-network-based control design: An LMI approach," *IEEE Trans. Neural Netw.*, vol. 9, no. 6, pp. 1422–1429, Nov. 1998.

[32] M. M. Polycarpou, "Stable adaptive neural control scheme for nonlinear systems," *IEEE Trans. Autom. Control*, vol. 14, no. 3, pp. 447–451, Mar. 1996.

[33] S. Ferrari and R. Stengel, "Smooth function approximation using neural networks," *IEEE Trans. Neural Netw.*, vol. 16, no. 1, pp. 24–38, Jan. 2005.

[34] J. S. J. Tyler, "The characteristics of model-following systems are synthesized by optimal control," *IEEE Trans. Autom. Control*, vol. AC-9, no. 4, pp. 485–498, Oct. 1964.

[35] H. Erzberger, "Analysis and design of model-following control systems by state space techniques," in *Proc. Joint Autom. Control Conf.*, Jun. 1968, pp. 572–581.

[36] F. D'Amato, M. Rotea, A. Megretski, and U. Jönsson, "New results for analysis of systems with repeated nonlinearities," *Automatica*, vol. 37, pp. 739–747, 2001.

[37] P. Apkarian and P. Gahinet, "A convex characterization of gain-scheduled H_∞ controllers," *IEEE Trans. Autom. Control*, vol. 40, no. 5, pp. 853–864, May 1995.

[38] Y. Huang and W. Lu, "An LFT approach to autopilot design for missiles," in *Proc. Amer. Control Conf.*, Jun. 1996, pp. 2990–2994.

[39] S. Boyd, L. El-Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*. Philadelphia, PA: SIAM, 1994.

[40] P. Apkarian, P. Gahinet, and G. Becker, "Self-scheduled H_∞ control of linear parameter-varying systems: A design example," *Automatica*, vol. 31, no. 9, pp. 1251–1261, 1997.

[41] P. Gahinet, A. Nemirovski, A. J. Laub, and M. Chilali, LMI Control Toolbox, for Use With MATLAB, The Mathworks, Inc., Natick, MA, 1995 [Online]. Available: http://www.mathworks.com

[42] R. F. Stengel, *Optimal Control and Estimation*. New York: Dover, 1994.

[43] Z. Lin and M. Khammash, "Robust gain-scheduled aircraft longitudinal controller design using an H_∞ approach," in *Proc. Amer. Control Conf.*, Arlington, VA, 2001, pp. 2724–2729.

[44] G. Becker and A. Packard, "Robust performance of linear parametrically varying systems using parametrically-dependent linear feedback," *Syst. Control Lett.*, vol. 23, no. 3, pp. 205–215, 1994.

[45] A. Megretski and A. Rantzer, "System analysis via integral quadratic constraints," *IEEE Trans. Autom. Control*, vol. 42, no. 6, pp. 819–830, Jun. 1997.

[46] A. Megretski and A. Rantzer, "System analysis via integral quadratic constraints: Part II," L. I. O. Technology, Tech. Rep. ISRN LUTFD2/TFRT-7559-SE, Sep. 1997.

[47] A. Rantzer, "On the Kalman-Yacubovich-Popov lemma," *Syst. Control Lett.*, vol. 28, pp. 7–10, 1996.

[48] K.-K. D. Joung, "Design of variable structure model-following control systems," *IEEE Trans. Autom. Control*, vol. AC-23, no. 6, pp. 1079–1085, Dec. 1978.

[49] I. D. Landau and B. Courtiol, "Design of model reference adaptive techniques—Theory and applications," *Automatica*, vol. 10, pp. 353–379, 1974.

[50] R. T. Reichert, "Robust autopilot design using μ-synthesis," in *Proc. Amer. Control Conf.*, May 1990, pp. 2368–2373.

[51] R. T. Reichert, "Dynamic scheduling of modern-robust control autopilot designs for missiles," *IEEE Control Syst. Mag.*, vol. 12, no. 5, pp. 35–42, Oct. 1992.

[52] C. Huang and R. F. Stengel, "Restructurable control using proportional-integral implicit model following," *J. Guidance*, vol. 13, no. 2, pp. 303–309, 1990.

[53] W. J. Rugh, "Analytical framework for gain scheduling," *IEEE Control Syst. Mag.*, vol. 11, no. 1, pp. 79–84, Jan. 1991.

[54] H. K. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ: Prentice-Hall, 2002.

[55] D. S. Bernstein, *Matrix Mathematics*. Princeton, NJ: Princeton Univ. Press, 2005.

[56] G. Strang, *Introduction to Linear Algebra*. Philadelphia, PA: SIAM, 1988.

Silvia Ferrari (S'01–M'02–SM'08) received the B.S. degree from Embry-Riddle Aeronautical University, Daytona Beach, FL, in 1997, and the M.A. and Ph.D. degrees from Princeton University, Princeton, NJ, in 1999 and 2002, respectively.

Currently, she is an Assistant Professor of Mechanical Engineering and Materials Science at Duke University, Durham, NC, where she directs the Laboratory for Intelligent Systems and Controls (LISC). Her principal research interests include robust adaptive control of aircraft, learning and approximate dynamic programming, and control of mobile sensor networks.

Dr. Ferrari is a member of the American Society of Mechanical Engineers (ASME), the International Society for Optical Engineers (SPIE), and the American Institute of Aeronautics and Astronautics (AIAA). She is the recipient of the U.S. Office of Naval Research (ONR) Young Investigator award (2004), the national Science Foundation (NSF) CAREER award (2005), and the Presidential Early Career Award for Scientists and Engineers (PECASE) award (2006).