Spring 2007

Reinforcement-learning based output-feedback controller for nonlinear discrete-time system with application to spark ignition engines operating lean and EGR

Peter Shih

REINFORCEMENT-LEARNING BASED OUTPUT-FEEDBACK CONTROLLER FOR NONLINEAR DISCRETE-TIME SYSTEM WITH APPLICATION TO SPARK IGNITION ENGINES OPERATING LEAN AND EGR

by

PETER SHIH

A THESIS

Presented to the Faculty of the Graduate School of the

UNIVERSITY OF MISSOURI-ROLLA

In Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE IN COMPUTER ENGINEERING

2007

Approved by

_____	_____
Dr. Jagannathan Sarangapani, Advisor	Dr. James A. Drallmeier

_____

Dr. Scott C. Smith

PUBLICATION THESIS OPTION

This thesis consists of the following two articles that have been submitted for publication as follows:

Pages 1 - 37 will appear in the Proceedings of the 2007 American Controls Conference and submitted for review to IEEE Transactions on Neural Networks

Pages 38 - 81 are intended for submission to 2007 IEEE Joint Conference on Neural Networks and IEEE Systems, Man and Cybernetics.

ABSTRACT

A spark ignition (SI) engine can be described by non-strict feedback nonlinear discrete-time system with the output dependent upon on the states in a nonlinear manner. The controller developed in this thesis utilizes the inherent universal approximation property of neural networks (NN) to simplify the design process and solve the non-causality problem inherent with traditional designs. It also exploits a long-term performance index called the *strategic* utility function to minimize and assist in updating of the NN weights; therefore, an *optimal* controller can be realized. Finally, through Lyapunov equations, the controller guarantees stability.

The controller allows for engine operation under two types of conditions: lean and EGR. Lean operation decreases the ratio of fuel over air below the stoichiometric levels where modern engines normally operate. In contrast, EGR introduce an inert gas, such as nitrogen in the lab or exhaust gas to displace a portion of the stoichiometric fuel air ratio. The purpose of these two operation modes are to improve fuel efficiency and more importantly, decrease the amount of harmful pollutants such as nitrous oxide (NO_x), carbon dioxide (CO_2), unburned hydrocarbon (H_xC_x), and sulfur dioxide (SO_2).

Operating under these stressful conditions increase cyclic variability, an erratic behavior where partial or no burn occurs at an increasing rate as the engine operates further away from normal condition. The controller remedies this situation by predicting future heat release and adjusting the amount of fuel for the next cycle while continuously adapting to changing conditions. Both cases decreased the unburned hydrocarbon by 8% and cyclic dispersion by 20% on average, while suffered negligible increase in average fuel input. Other pollutants also decreased with varying success.

ACKNOWLEDGMENTS

I would like to thank my Advisor Dr. Jagannathan Sarangapani for his guidance, support and patience. I thank Dr. James A. Drallmeier for helping me grasp the mechanics of spark ignition engines and suggesting improvements. I thank Brian Kaul for operating the engine and assisting in data collection. I thank John Vance for assisting me throughout this project and allowing me to use his custom hardware. I thank Dr. Scott C. Smith for his guidance and being a committee member. I thank National Science Foundation for the financial support. Also, I thank my family. I would be lost without them. Finally, I thank my wife, Xiaoou Pan, for her emotional support and being there always.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

PAPER 1

Reinforcement Learning Based Output-Feedback Controller for Nonlinear Discrete-time Systems with Application to Spark Engines

Peter Shih, B. Kaul, J. Vance, Sarangapani Jagannathan, *Sr. Member, IEEE*, and James A. Drallmeier

***Abstract*— A novel reinforcement-learning based output-adaptive neural network (NN) controller, also referred to as the adaptive-critic NN controller, is developed to deliver a desired tracking performance for a class of complex feedback nonlinear discrete-time systems in the presence of bounded and unknown disturbances. The adaptive critic NN controller consists of an observer, critic, and two action NNs. The observer estimate the states and output and the two action NNs provide virtual and actual control inputs to the nonlinear discrete-time system. The critic approximates a certain *strategic* utility function and the action NNs minimize both the *strategic* utility function and control inputs. All NN weights adapt online towards minimization of a quadratic performance index, utilizing gradient-descent based rule. Lyapunov functions are used to show the uniformly ultimate boundedness (UUB) of the closed-loop tracking error, weights, and observer estimates. Separation principle and certainty equivalence principle are relaxed, persistency of excitation condition is not required and linear in the unknown parameter assumption is not needed.**

The performance of this adaptive critic NN controller is evaluated on a spark ignition (SI) engine operating lean where the NN controller objective is to reduce cyclic dispersion in heat release while facing unknown engine dynamics. The secondary objectives are to reduce emissions. Experimental results at the equivalence ratio of 0.75 show a significant (25%) reduction in cyclic dispersion in heat release with control compared to the uncontrolled case. The average fuel input changes by less than 1% compared to uncontrolled case. Additionally, oxides of

nitrogen (NO_x) drop by 30% compared to uncontrolled. The unburned hydrocarbons (uHC) drop by 16% with control. Overall, NO_x are reduced over 80% compared to stoichiometric levels.

I. INTRODUCTION

Adaptive neural network NN backstepping control of nonlinear discrete-time systems in strict feedback form has been addressed in the literature [1-3]. The strict feedback nonlinear system is normally expressed as

$$x_i(k+1) = f_i(\bar{x}_i(k)) + g_i(\bar{x}_i(k))x_{i+1}(k) \quad (1)$$

$$x_n(k+1) = f_n(\bar{x}_n(k)) + g_n(\bar{x}_n(k))u(k) \quad (2)$$

where $x_i(k) \in \Re$ is the state, $u(k) \in \Re$ is the control input, $\bar{x}_i(k) = [x_1(k), \cdots, x_i(k)]^T \in \Re^i$ and $i = 1, ..., (n-1)$. For strict feedback nonlinear systems [1], the nonlinearities $f_i(\bar{x}_i(k))$ and $g_i(\bar{x}_i(k))$ depend only upon states $x_1(k), \ldots, x_i(k)$, i.e., $\bar{x}_i(k)$. However, for non-strict feedback nonlinear system, where $f_i(\bar{x}_i(k))$ and $g_i(\bar{x}_i(k))$ depend on both $\bar{x}_i(k)$ and $x_{i+1}(k)$, there are no control design schemes currently available. Available [1-3] methods applied to the nonlinear discrete-time systems will result in a non-causal controller (current control input depends on the future system states) even when the system is of second order and when the adaptive NN backstepping approach is utilized. Finally, no optimization is carried out in these control designs whereas simple tracking error is utilized.

In short, available NN controller designs employ either supervised training, where the user specifies a desired output, or online NN training based on classical adaptive control [1-3], where a short-term system performance measure is defined by using the tracking error. By contrast, the reinforcement-learning based adaptive critic NN approach [4] has emerged as a promising tool to develop optimal NN controllers due to its potential to find approximate solutions to dynamic programming, where a *strategic* utility function, which is considered as the long-term system performance measure, can be optimized. In supervised learning, an explicit signal is provided by the teacher to guide the learning process whereas in the case of reinforcement learning, the role of the teacher is more

evaluative than instructional in nature. The critic NN monitors the system states and approximates the *strategic* utility function, with a potential for a look-ahead and better training of the action NN which generates the control action to the system.

There are many variants of adaptive critic NN controller architectures [4-9] using state feedback even though few results [6-9] address the controller convergence and they are limited to affine nonlinear discrete-time systems. However, NN controller results are not available for the nonlinear discrete-time systems in non-strict feedback form.

In this paper, a novel adaptive critic NN-based *output* feedback controller is developed to control a class of nonlinear discrete-time systems in non-strict feedback form with bounded and unknown disturbances. Adaptive NN backstepping is utilized for the controller design with two action NNs being used to generate the virtual and actual control inputs, respectively. The two action NN weights are tuned by the critic NN signal to minimize the *strategic* utility function and their outputs. The critic NN approximates certain *strategic* utility function which is a variant of standard Bellman equation. The NN observer generates the estimates of the system states and output, which are subsequently used in the controller design. The proposed controller is *model–free* since the dynamics of the nonlinear discrete-time systems are unknown and NN weights are tuned online.

The main contributions of this paper can be summarized as follows: 1) the adaptive NN backstepping scheme is extended to non-strict feedback nonlinear systems. The non-causal problem is overcome by employing the universal NN approximation property; 2) optimization of a long-term performance index is undertaken in contrast with traditional adaptive NN back stepping schemes [1, 2] where no optimization is performed; 3) demonstration of the UUB of the overall system is shown even in the presence of NN approximation errors and bounded unknown disturbances unlike in the existing adaptive critic works [7-9] where the convergence is presented under ideal circumstances. Stability proof is inferred even with a NN observer by relaxing separation principle via novel weight updating rules and by selecting the Lyapunov function consisting of the system estimation errors, tracking and the NN weight estimation errors; 4) a well-defined controller is presented by overcoming the problem of certain nonlinear function estimate becoming zero since a single NN is used to approximate both the nonlinear

functions $f_i(\bar{x}_i(k))$ and $g_i(\bar{x}_i(k))$ compared to [10]; 5) the NN weights are tuned online instead of offline [5]; and finally 6) the assumption that $g_1(x_1(k), x_2(k))$ is bounded away from zero and its sign is known *a priori* is relaxed in contrast with [2].

The proposed controller is applied to control spark ignition (SI) engine dynamics, a practical non-strict feedback nonlinear system. The controller allows the engine to operate in a lean regime, where less than stoichiometric ratio of fuel to air is injected each cycle. The problem of operating the engine lean is cyclic dispersion in heat release which makes the engine unstable. The controller enables the engine to operate leaner compared to the uncontrolled case by reducing heat release dispersion. Consequently, the fuel conversion efficiency increases and engine out emissions decrease. Though an SI engine with a three-way catalyst cannot be operated lean, the objective is to control an SI engine used for other applications such as scooters and lawn mowers, where a three-way catalyst is not normally used. Alternatively, the proposed scheme could be used with the new generation of lean NO_x catalyst systems currently under development.

II. Non-Linear Non-Strict Feedback Discrete-Time Systems

Consider the nonlinear discrete-time system, given in the following form

$$x_1(k+1) = f_1(x_1(k), x_2(k)) + g_1(x_1(k), x_2(k))x_2(k) + d_1(k) \tag{3}$$

$$x_2(k+1) = f_2(x_1(k), x_2(k)) + g_2(x_1(k), x_2(k))u(k) + d_2(k) \tag{4}$$

$$y(k) = f_3(x_1(k), x_2(k)) \tag{5}$$

where $x_i(k) \in \Re; i = 1, 2$ are the states, $u(k) \in \Re$ is the system input, and $d_1(k) \in \Re$ and $d_2(k) \in \Re$ are unknown but bounded disturbances whose bounds are given by $|d_1(k)| < d_{1m}$ and $|d_2(k)| < d_{2m}$, where d_{1m} and d_{2m} are unknown positive scalars. Here the nonlinearities are considered unknown. The system output is a nonlinear function of states in contrast with available literature [11, 12] where the output is considered as a linear function of the states. Finally, only the output is considered measurable whereas the states are not available. An additional constraint that has to be satisfied is that the convergence of the output to its target value alone is not sufficient and the states have to be close to their respective target values.

III. OBSERVER DESIGN

To overcome the immeasurable states, $x_1(k)$ and $x_2(k)$, an observer is utilized where the current heat release output, $y(k)$, is employed to estimate the future output $\hat{y}(k+1)$ and states $\hat{x}_1(k+1)$ and $\hat{x}_2(k+1)$. The design of the observer is discussed next.

A. Observer Design

Consider equations (3) and (4). We expand the individual nonlinear functions using Taylor series expansion into linear and higher order terms as follows

$$f_1(\cdot) = f_{10} + \Delta f_1(\cdot) \tag{6}$$

$$f_2(\cdot) = f_{20} + \Delta f_2(\cdot) \tag{7}$$

$$g_1(\cdot) = g_{10} + \Delta g_1(\cdot) \tag{8}$$

$$g_2(\cdot) = g_{20} + \Delta g_2(\cdot) \tag{9}$$

where the first term in (6) through (9) are known nominal values and the second term are unknown higher order terms. We use a two-layer feed-forward NN with semi-recurrent architecture and novel weight tuning to construct the output as

$$y(k+1) = w_1^T \phi\left(v_1^T z_1(k)\right) + \varepsilon\left(z_1(k)\right), \tag{10}$$

where $z_1(k) = \left[x_1(k), x_2(k), y(k), u(k)\right]^T \in R^4$ is the network input, $y(k+1)$ and $y(k)$ are the future and current output values, $w_1 \in \Re^{n_1}$ and $v_1 \in \Re^{2\times n_1}$ denote the ideal output and constant hidden layer weights, respectively, $u(k)$ is the control input, $\phi\left(v_1^T z_1(k)\right)$ represents the hidden layer activation function, n_1 is the number of nodes in the hidden layer, and $\varepsilon\left(z_1(k)\right) \in \Re$ is the approximation error. For simplicity the two equations can be represented as

$$\phi_1(k) = \phi\left(v_1^T z_1(k)\right) \tag{11}$$

and

$$\varepsilon_1(k) = \varepsilon\left(z_1(k)\right) \tag{12}$$

Rewrite (10) using (11) and (12) to obtain

$$y(k+1) = w_1^T \phi_1(k) + \varepsilon_1(k) \tag{13}$$

The states $x_1(k)$ and $x_2(k)$ are not measurable, therefore, $z_1(k)$ is not available either. Using the estimated states and output $\hat{x}_1(k)$, $\hat{x}_2(k)$, and $\hat{y}(k)$, respectively, instead of $x_1(k)$, $x_2(k)$, and $y(k)$, the proposed observer is given as

$$\begin{aligned} \hat{y}(k+1) &= \hat{w}_1^T(k)\phi\left(v_1^T \hat{z}_1(k)\right) + l_1 \tilde{y}(k) \\ &= \hat{w}_1^T(k)\hat{\phi}_1(k) + l_1 \tilde{y}(k) \end{aligned} \tag{14}$$

where $\hat{z}_1(k) = \left[\hat{x}_1(k), \hat{x}_2(k), \hat{y}(k), u(k)\right]^T \in R^4$ is the NN input vector using estimated states, $\hat{y}(k+1)$ and $\hat{y}(k)$ are the estimated future and current outputs, $\hat{w}_1(k)$ is the actual weight matrix, $u(k)$ is the control input, $\hat{\phi}_1(k)$ is the hidden layer activation function, $l_1 \in R$ is the observer gain, and $\tilde{y}(k)$ is the heat release estimation error defined as

$$\tilde{y}(k) = \hat{y}(k) - y(k) \tag{15}$$

It is demonstrated in [13] that, if the hidden layer weights, v_1, are chosen initially at random and kept constant, and the number of hidden layer nodes is sufficiently large, then the approximation error $\varepsilon\left(z_1(k)\right)$ can be made arbitrarily small so that the bound $\left\|\varepsilon\left(z_1(k)\right)\right\| \le \varepsilon_{1m}$ holds for all $z_1(k) \in S$ in a compact set, since the activation function vector forms a basis to the nonlinear function that the NN approximates. Now we choose, at our convenience, the observer structure as a function of output estimation errors and known quantities as

$$\hat{x}_1(k+1) = f_{10} - \hat{x}_2(k) + l_2 \tilde{y}(k) \tag{16}$$

$$\hat{x}_2(k+1) = f_{20} + g_{20} u(k) + l_3 \tilde{y}(k) \tag{17}$$

where $l_2 \in R$ and $l_3 \in R$ are design constants.

B. Observer Error Dynamics

Define the state estimation and output errors as

$$\tilde{x}_i(k+1) = \hat{x}_i(k+1) - x_i(k+1), i \in \{1,2\} \tag{18}$$

$$\tilde{y}(k+1) = \hat{y}(k+1) - y(k+1) \tag{19}$$

Combine (3) through (10) and (16) through (19), to obtain the estimation and output error

$$\tilde{x}_1(k+1) = f_{10} - \hat{x}_2(k) + l_2\tilde{y}(k) - f_1(\cdot) - g_1(\cdot)x_2(k) - d_1(k) \tag{20}$$

$$\tilde{x}_2(k+1) = f_{20} + g_{20}u(k) + l_3\tilde{y}(k) - f_2(\cdot) - g_2(\cdot)u(k) - d_2(k) \tag{21}$$

$$\tilde{y}(k+1) = \hat{w}_1^T(k)\hat{\phi}_1(k) + l_1\tilde{y}(k) - w_1^T\phi_1(k) - \varepsilon_1(k) \tag{22}$$

Choose the weight tuning of the observer NN as

$$\hat{w}_1(k+1) = \hat{w}_1(k) - \alpha_1\hat{\phi}_1(k)\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right) \tag{23}$$

where $\alpha_1 \in R$, and $l_4 \in R$ are design constants. It will be shown that by using the above weight tuning, the separation principle is relaxed and the closed-loop signals will be bounded. Next we present the following theorem, where it is demonstrated that the state estimation and output estimation errors along with observer NN weight estimation errors are bounded. The following mild assumptions are required.

Assumption 1: The unknown smooth functions, $f_2(\cdot)$ and $g_2(\cdot)$, and control $u(k)$, are upper bounded within the compact set S as $f_{2\max} > |f_2(k)|$, and $g_{2\max} > |g_2(k)|$.

Theorem 1: Consider the system given by (3), (4) and (5), and the disturbance bounded by $|d_1(k)| < d_{1m}$ and $|d_2(k)| < d_{2m}$ where d_{1m} and d_{2m} are known positive scalars. Let the observer NN weight tuning be given by (23). Given bounded inputs such that $u_{\max} > |u(k)|$, the state estimation errors $\tilde{x}_1(k)$ and $\tilde{x}_2(k)$, output estimation error $\tilde{y}(k)$ and observer NN weight estimation errors $\tilde{w}_1(k)$ are UUB, with the bounds specifically given by (B.17) provided the controller design parameters are selected as

$$0 < \alpha_1\|\phi_1(k)\|^2 < 1 \tag{24}$$

$$|l_1| < \frac{1}{2} \tag{25}$$

$$|l_2| < \frac{\sqrt{3}}{3} \tag{26}$$

$$|l_3| < \frac{\sqrt{3}}{3} \tag{27}$$

$$|l_4| < \frac{\sqrt{3}}{3} \tag{28}$$

where α_1 is NN adaptation gain, l_1, l_2, l_3, and l_4 are observer parameters.

Proof: Define the Lyapunov function

$$J(k)=\sum_{i=1}^{4}J_i(k)=\frac{\gamma_1}{\alpha_1}\tilde{w}_1^T(k)\tilde{w}_1(k)+\frac{\gamma_2}{3}\tilde{x}_1^2(k)+\frac{\gamma_3}{2}\tilde{x}_2^2(k)+\frac{\gamma_4}{3}\tilde{y}^2 \tag{29}$$

where $0<\gamma_i, i\in\{1,2,3,4\}$ are auxiliary constants. Take the first difference of the first term, and substitute (23) to get

$$\begin{aligned}
J_1(k)&=\tfrac{\gamma_1}{\alpha_1}\tilde{w}_1^T(k)\tilde{w}_1(k)\\
\tfrac{\alpha_1}{\gamma_1}\Delta J_1(k)&=\tilde{w}_1^T(k+1)\tilde{w}_1(k+1)-\tilde{w}_1^T(k)\tilde{w}_1(k)\\
&=[\tilde{w}_1^T(k)-\alpha_1\left(\hat{w}_1^T(k)\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^T\hat{\phi}_1^T(k)]*\\
&\quad[\tilde{w}_1(k)-\alpha_1\hat{\phi}_1(k)\left(\hat{w}_1^T(k)\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)]-\tilde{w}_1^T(k)\tilde{w}_1(k)\\
&=\alpha_1^2\left\|\phi_1(k)\right\|^2\begin{pmatrix}\hat{w}_1^T(k)\hat{\phi}_1(k)\\+l_4\tilde{y}(k)\end{pmatrix}^2-2\alpha_1\tilde{w}_1(k)\hat{\phi}_1(k)\begin{pmatrix}\hat{w}_1^T(k)\hat{\phi}_1(k)\\+l_4\tilde{y}(k)\end{pmatrix}+\\
&\quad\alpha_1\left(\hat{w}_1^T(k)\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^2-\alpha_1\left(\hat{w}_1^T(k)\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^2\\
&=-\alpha_1\left(1-\alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\left(\hat{w}_1^T(k)\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^2+\\
&\quad\alpha_1\left(\left(\zeta_1(k)+w_1^T\hat{\phi}_1+l_4\tilde{y}(k)\right)-\zeta_1(k)\right)^2-\alpha_1\zeta_1^2(k)\\
&=-\alpha_1\left(1-\alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\begin{pmatrix}\hat{w}_1^T(k)\hat{\phi}_1(k)\\+l_4\tilde{y}(k)\end{pmatrix}^2+\alpha_1\left(w_1^T\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^2-\alpha_1\zeta_1^2(k)
\end{aligned} \tag{30}$$

Invoke Cauchy-Schwarz inequality defined as

$$\left(a_1b_1+\ldots+a_nb_n\right)^2\le\left(a_1^2+\ldots+a_n^2\right)\left(b_1^2+\ldots+b_n^2\right) \tag{31}$$

and simplify to get

$$\begin{aligned}
\Delta J_1(k)\le&-\gamma_1\left(1-\alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\left(\hat{w}_1(k)\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^2+\\
&2\gamma_1\left(w_{1\mathrm{m}}\hat{\phi}_{1\mathrm{m}}\right)^2+2\gamma_1l_4^2\tilde{y}^2(k)-\gamma_1\zeta_1^2(k)
\end{aligned} \tag{32}$$

Take the second term and substitute (20)

$$\Delta J_2(k)\le\gamma_2l_2^2\tilde{y}^2(k)+\gamma_2\tilde{x}_2^2(k)+\gamma_2\left(w_{3\mathrm{m}}\phi_{3\mathrm{m}}+f_{10}+\varepsilon_{3\mathrm{m}}+d_{1\mathrm{m}}\right)^2-\tfrac{\gamma_2}{3}\tilde{x}_1^2(k) \tag{33}$$

Take the third term and substitute (21)

$$\Delta J_3(k) \le \gamma_3 \left(f_{20} + \left(g_{20} + g_{2\max} \right) u_{\max} + f_{2\max} + d_{2\mathrm{m}} \right)^2 + \gamma_3 l_3^2 \tilde{y}^2(k) - \tfrac{\gamma_3}{2} \tilde{x}_2^2(k) \tag{34}$$

Take the fourth and final term and substitute (22)

$$\Delta J_4(k) \le \gamma_4 \zeta_1^2(k) + \gamma_4 l_1^2 \tilde{y}(k) + \gamma_4 \left(w_{1\mathrm{m}} \tilde{\phi}_{1\mathrm{m}} + \varepsilon_{1\mathrm{m}} \right) - \tfrac{\gamma_4}{3} \tilde{y}^2(k) \tag{35}$$

Combine equations (32) through (35) and simplify to get the first difference of the Lyapunov function.

$$\begin{aligned} \Delta J(k) \le & -\gamma_1 \left(1 - \alpha_1 \left\| \hat{\phi}_1(k) \right\|^2 \right) \left(\hat{w}_1(k) \hat{\phi}_1(k) + l_4 \tilde{y}(k) \right)^2 - \left(\tfrac{\gamma_3}{2} - \gamma_2 \right) \tilde{x}_2^2(k) - \tfrac{\gamma_2}{3} \tilde{x}_1^2(k) \\ & - \left(\tfrac{\gamma_4}{3} - 2\gamma_1 l_4^2 - \gamma_2 l_2^2 - \gamma_3 l_3^2 - \gamma_4 l_1^2 \right) \tilde{y}^2(k) - \left(\gamma_1 - \gamma_4 \right) \zeta_1^2(k) + D_M^2 \end{aligned} \tag{36}$$

where D_M^2 is defined as

$$\begin{aligned} D_M^2 = & 2\gamma_1 \left(w_{1\mathrm{m}} \hat{\phi}_{1\mathrm{m}} \right)^2 + \gamma_2 \left(w_{3\mathrm{m}} \phi_{3\mathrm{m}} + f_{10} + \varepsilon_{3\mathrm{m}} + d_{1\mathrm{m}} \right)^2 + \\ & \gamma_3 \left(f_{20} + \left(g_{20} + g_{2\max} \right) u_{\max} + f_{2\max} + d_{2\mathrm{m}} \right)^2 + \gamma_4 \left(w_{1\mathrm{m}} \tilde{\phi}_{1\mathrm{m}} + \varepsilon_{1\mathrm{m}} \right) \end{aligned} \tag{37}$$

Select

$$\gamma_3 > 2\gamma_2;\ \gamma_4 > 6\gamma_1 l_4^2 + 3\gamma_2 l_2^2 + 3\gamma_3 l_3^2 + 3\gamma_4 l_1^2;\ \gamma_1 > \gamma_4 \tag{38}$$

This implies $\Delta J(k) < 0$ as long as (24) through (28) hold *and* the following hold

$$\begin{aligned} & \left| \tilde{x}_1(k) \right| > \frac{D_M}{\sqrt{\frac{\gamma_2}{3}}}; or \left| \tilde{x}_2(k) \right| > \frac{D_M}{\sqrt{\frac{\gamma_3}{2} - \gamma_2}}; or \left| \tilde{y}(k) \right| > \frac{D_M}{\sqrt{\frac{\gamma_4}{3} - 2\gamma_1 l_4^2 - \gamma_2 l_2^2 - \gamma_3 l_3^2 - \gamma_4 l_1^2}}; \\ & or \left| \zeta_1(k) \right| > \frac{D_M}{\sqrt{\gamma_1 - \gamma_4}} \end{aligned} \tag{39}$$

According to a standard Lyapunov extension theorem [14], this demonstrates that the estimation errors, the output error, and the NN observer weight estimation errors are *UUB*. ***Remark***: In this above theorem, the control input is considered bounded which is an acceptable assumption (also made in all output feedback control literature) which is relaxed in the next few sections when combined with the controller design wherein the enclosed-loop system is shown to be bounded. On the other hand, the assumption that the unknown nonlinearities are bounded is valid since for many practical systems, the upper bound on the unknown nonlinearities will be known [14]. Additionally, for NN based control it is also necessary for the nonlinear functions to be on a compact set in order for the NN to approximate them.

Next we discuss the design of the adaptive critic NN controller and demonstrate that if the closed-loop system including the NN observer signals will be bounded, then the control inputs will be bounded.

IV. CRITIC DESIGN

The purpose of the critic NN is to approximate the long-term performance index (or strategic utility function) of the nonlinear system through online weight adaptation. The critic signal also tunes the two action NNs. The tuning will ultimately minimize the strategic utility function itself and NN outputs so that closed-loop stability is inferred.

A. The Strategic Utility Function

The utility function $p(k) \in \Re$ is given by

$$p(k) = \begin{cases} 0, & if \left(\left| \tilde{y}(k) \right| \right) \le c \\ 1, & otherwise \end{cases} \tag{40}$$

where $c \in \Re$ is a user-defined threshold. The utility function $p(k)$ represents the current performance index. In other words, $p(k) = 0$ and $p(k) = 1$ refer to good and unsatisfactory tracking performance at the k^{th} time step, respectively. The long-term *strategic* utility function $Q(k) \in \Re$, is defined as

$$Q(k) = \beta^N p(k+1) + \beta^{N-1} p(k+2) + \cdots + \beta^{k+1} p(N) + .., \tag{41}$$

where $\beta \in \Re$ and $0 < \beta < 1$ is the discount factor and N is the horizon index. The term $Q(k)$ is viewed here as the long system performance measure for the controller since it is the sum of all future system performance indices. Equation (41) can also be expressed as $Q(k) = \min_{u(k)} \left\{ \alpha Q(k-1) - \alpha^{N+1} p(k) \right\}$, which is similar to the standard Bellman equation.

B. Design of the Critic NN

We utilize the universal approximation property of NN to estimate the critic NN output and rewrite $\hat{Q}(k)$ as

$$\hat{Q}(k) = \hat{w}_2^T(k) \phi\left(v_2^T \hat{z}_2(k) \right) = \hat{w}_2^T(k) \hat{\phi}_2(k) \tag{42}$$

where $\hat{Q}(k) \in \Re$ is the critic signal, $\hat{w}_2(k) \in \Re^{n_2}$ is the tunable weight matrix, $v_2 \in \Re^{2 \times n_2}$ represent the constant input weight matrix selected initially at random, $\hat{\phi}_2(k) \in \Re^{n_2}$ is the activation function vector in the hidden layer, n_2 is the number of the nodes in the hidden layer, and $\hat{z}_2(k) = [\hat{x}_1(k), \hat{x}_2(k)]^T \in R^2$ is the NN input vector.

C. Critic Weight Update Law

Define the prediction error as

$$e_c(k) = \hat{Q}(k) - \beta\left(\hat{Q}(k-1) - \beta^N p(k)\right) \tag{43}$$

where the subscript "c" stands for the "critic." Define a quadratic objective function to minimize based on the prediction error

$$E_c(k) = \frac{1}{2} e_c^2(k) \tag{44}$$

The weight update rule for the critic NN is obtained using gradient adaptation, which is given by the general formula

$$\hat{w}_2(k+1) = \hat{w}_2(k) + \Delta\hat{w}_2(k) \tag{45}$$

$$\Delta\hat{w}_2(k) = \alpha_2\left[-\frac{\partial E_c(k)}{\partial \hat{w}_2(k)}\right] \tag{46}$$

or

$$\hat{w}_2(k+1) = \hat{w}_2(k) - \alpha_2\hat{\phi}_2(k)\left(\hat{Q}(k) + \beta^{N+1}p(k) - \beta\hat{Q}(k-1)\right)^T \tag{47}$$

where $\alpha_2 \in \Re$ is the NN adaptation gain.

V. Design of the Virtual Control Input

In this section, the design of the virtual control input is discussed. Before we proceed, the following mild assumption is needed. Then the systems of nonlinear equations are

Assumption 2: The unknown smooth function $g_2(\cdot)$ is bounded away from zero for all $x_1(k)$ and $x_2(k)$ within the compact set S. In other words, $0 < g_{2\min} < \left|g_2(\cdot)\right| < g_{2\max}$, $\forall x_1(k) \,\&\, x_2(k) \in S$ where $g_{2\min} \in \Re^+$ and $g_{2\max} \in \Re^+$. Without loss of generality, we will assume $g_2(\cdot)$ is positive in this paper.

A. System Simplification

First, we simplify by rewriting the state equations with the following

$$\Phi(\cdot) = f_1\left(x_1(k), x_2(k)\right) + g_1\left(x_1(k), x_2(k)\right)x_2(k) + x_2(k) \tag{48}$$

The system of equations (3) and (4) can be rewritten as

$$x_1(k+1) = \Phi(\cdot) - x_2(k) + d_1(k) \tag{49}$$

$$x_2(k+1) = f_2(\cdot) + g_2(\cdot)u(k) + d_2(k) \tag{50}$$

B. Virtual Control Input Design

Our goal is to stabilize the system output $y(k)$ around a specified target point, y_d by controlling the input. The secondary objective is to make $x_1(k)$ approach the desired bounded trajectory $x_{1d}(k)$. At the same time, all signals in systems (3) and (4) must be UUB; all the NN weights must be bounded; and a performance index must be minimized. Define the tracking error as

$$e_1(k) = x_1(k) - x_{1d}(k) \tag{51}$$

where $x_{1d}(k)$ is the desired trajectory. Using (49), (51) can be expressed as

$$\begin{aligned} e_1(k+1) &= x_1(k+1) - x_{1d}(k+1) \\ &= \left(\Phi(\cdot) - x_2(k) + d_1(k)\right) - x_{1d}(k+1) \end{aligned} \tag{52}$$

By viewing $x_2(k)$ as a virtual control input, a desired virtual control signal can be designed as

$$x_{2d}(k) = \Phi(\cdot) - x_{1d}(k+1) + l_5\hat{e}_1(k) \tag{53}$$

where l_5 is a gain constant. Since $\Phi(\cdot)$ is an unknown function, $x_{2d}(k)$ in (53) cannot be implemented in practice. We invoke the universal approximation property of NN to estimate this unknown nonlinear function.

$$\Phi(\cdot) = w_3^T\phi\left(v_3^T z_3(k)\right) + \varepsilon\left(z_3(k)\right) \tag{54}$$

where $z_3(k) = \left[x_1(k), x_2(k)\right]^T \in \Re^2$ is the input vector, $w_3^T \in \Re^{n_2}$ and $v_3^T \in \Re^{2\times n_3}$ are the ideal and constant input weight matrices, $\phi\left(v_3^T z_3(k)\right) \in \Re^{n_3}$ is the activation function vector in the hidden layer, n_3 is the number of the nodes in the hidden layer, and

$\varepsilon\left(z_3(k)\right)$ is the functional estimation error. It is demonstrated in [13] that, if the hidden layer weights, v_1, are chosen initially at random and kept constant, and the number of hidden layer nodes is sufficiently large, then the approximation error $\varepsilon\left(z_3(k)\right)$ can be made arbitrarily small so that the bound $\left\|\varepsilon\left(z_3(k)\right)\right\| \le \varepsilon_{3m}$ holds for all $z_3(k) \in S$ in a compact set since the activation function vector forms a basis to the nonlinear function that the NN approximates.

Rewriting (53) using (54), the virtual control signal can be rewritten as

$$x_{2d}(k) = w_3^T \phi\left(v_3^T z_3(k)\right) + \varepsilon\left(z_3(k)\right) - x_{1d}(k+1) + l_5 \hat{e}_1(k) \tag{55}$$

Replacing the actual with estimated states, (55) becomes

$$\begin{aligned} \hat{x}_{2d}(k) &= \hat{w}_3^T(k)\phi\left(v_3^T \hat{z}_3(k)\right) - x_{1d}(k+1) + l_5 \hat{e}_1(k) \\ &= \hat{w}_3^T(k)\hat{\phi}_3(k) - x_{1d}(k+1) + l_5 \hat{e}_1(k) \end{aligned} \tag{56}$$

where $\hat{z}_3(k) = \left[\hat{x}_1(k), \hat{x}_2(k)\right]^T \in \Re^2$ is the NN input vector using estimated states, and $\hat{e}_1(k) = \hat{x}_1(k) - x_{1d}(k)$.

define

$$e_2(k) = x_2(k) - \hat{x}_{2d}(k) \tag{57}$$

Equation (52) can be rewritten using (57)

$$\begin{aligned} e_1(k+1) &= \left(\Phi(\cdot) - x_2(k) + d_1(k)\right) - x_{1d}(k+1) \\ &= \Phi(\cdot) - \left(e_2(k) + \hat{x}_{2d}(k)\right) + d_1(k) - x_{1d}(k+1) \\ &= \Phi(\cdot) - \hat{x}_{2d}(k) - e_2(k) - x_{1d}(k+1) + d_1(k) \end{aligned} \tag{58}$$

Replace (56) into (58), then (54) into the combined equation

$$\begin{aligned} e_1(k+1) &= \Phi(\cdot) - \left(\hat{w}_3^T(k)\hat{\phi}_3(k) - x_{1d}(k+1) + l_5\hat{e}_1(k)\right) - e_2(k) - x_{1d}(k+1) + d_1(k) \\ &= \left(w_3^T \phi_3(k) + \varepsilon_3(k)\right) - \hat{w}_3^T(k)\hat{\phi}_3(k) - l_5\hat{e}_1(k) - e_2(k) + d_1(k) \\ &= w_3^T\left(\hat{\phi}_3(k) - \tilde{\phi}_3(k)\right) - \hat{w}_3^T(k)\hat{\phi}_3(k) + \varepsilon_3(k) - l_5\hat{e}_1(k) - e_2(k) + d_1(k) \\ &= w_3^T\left(\hat{\phi}_3(k) - \tilde{\phi}_3(k)\right) - \hat{w}_3^T(k)\hat{\phi}_3(k) + \varepsilon_3(k) - l_5\hat{e}_1(k) - e_2(k) + d_1(k) \\ &= -\tilde{w}_3^T\hat{\phi}_3(k) - w_3^T\tilde{\phi}_3(k) + \varepsilon_3(k) - l_5\hat{e}_1(k) - e_2(k) + d_1(k) \\ &= -\zeta_3(k) - w_3^T\tilde{\phi}_3(k) + \varepsilon_3(k) - l_5\hat{e}_1(k) - e_2(k) + d_1(k) \end{aligned} \tag{59}$$

where

$$\zeta_3(k) = \tilde{w}_3^T(k)\hat{\phi}_3(k) = \hat{w}_3^T(k)\hat{\phi}_3(k) - w_3^T\hat{\phi}_3(k) \tag{60}$$

$$\tilde{\phi}_3(k) = \phi\left(v_3\hat{z}_3(k)\right) - \phi\left(v_3 z_3(k)\right) \tag{61}$$

C. Virtual Control Weight Update

Let us define

$$e_{a1}(k) = \hat{w}_3^T(k)\hat{\phi}_3(k) + \left(\hat{Q}(k) - Q_d(k)\right) \tag{62}$$

where $\hat{Q}(k)$ is defined in (42), and the a1 subscript represents the error for the first action NN, $e_{a1}(k) \in \Re$. The desired *strategic* utility function $Q_d(k)$ is "0" to indicate perfect tracking at all steps whereas the first term in (60) is essentially the action NN output or virtual control input. Thus, (62) becomes

$$e_{a1}(k) = \hat{w}_3^T(k)\hat{\phi}_3(k) + \hat{Q}(k) \tag{63}$$

The objective function to be minimized by the first action NN is given by

$$E_{a1}(k) = \frac{1}{2}e_{a1}^2(k) \tag{64}$$

The weight update rule for the action NN is also a gradient-based adaptation, which is defined as

$$\hat{w}_3(k+1) = \hat{w}_3(k) + \Delta\hat{w}_3(k) \tag{65}$$

where

$$\Delta\hat{w}_3(k) = \alpha_3\left[-\frac{\partial E_{a1}(k)}{\partial\hat{w}_3(k)}\right] \tag{66}$$

or in other words

$$\hat{w}_3(k+1) = \hat{w}_3(k) - \alpha_3\hat{\phi}_3(k)\left(\hat{Q}(k) + \hat{w}_3^T(k)\hat{\phi}_3(k)\right) \tag{67}$$

with $\alpha_3 \in \Re$ is the NN adaptation gain.

VI. Control Input Design

Choose the following desired control input

$$u_d(k) = \frac{1}{g_2(k)}\left(-f_2(k) + \hat{x}_{2d}(k+1) + l_6 e_2(k)\right), \tag{68}$$

Note that $u_d(k)$ is non-causal since it depends upon future value of $\hat{x}_{2d}(k+1)$. We solve this problem by using a semi-recurrent NN since it can be a one step predictor. The term $\hat{x}_{2d}(k+1)$ depends on state $x(k)$, virtual control input $\hat{x}_{2d}(k)$, desired trajectory $x_{1d}(k+2)$, and system errors $e_1(k)$ and $e_2(k)$. By taking the independent variables as the input to a NN, $\hat{x}_{2d}(k+1)$ can be approximated during control input selection. Consequently, in this paper, a feed forward NN with a properly chosen weight tuning law rendering a semi-recurrent or dynamic NN can be used to predict the future value. Alternatively, the value can be obtained by employing a filter (Jagannathan 2006). The first layer of the second NN using the system errors, state estimates and past value $\hat{x}_{2d}(k)$ as inputs generates $\hat{x}_{2d}(k+1)$, which in turn is used by the second layer to generate a suitable control input. The results in the simulation section show that the overall controller performance is satisfactory. On the other hand, one can use a single layer dynamic NN to generate the future value of $\hat{x}_{2d}(k)$, which can be utilized as an input to a third control NN to generate a suitable control input. Here, these two single layer NN are combined into a single NN.

Assume the NN input to be $z_4(k) = \left[x_1(k), x_2(k), e_1(k), l_6 e_2(k), \hat{x}_{2d}(k), x_{1d}(k+2)\right]^T \in \Re^6$, then $u_d(k)$ can be approximated as

$$u_d(k) = w_4^T \phi\left(v_4^T z_4(k)\right) + \varepsilon\left(z_4(k)\right) = w_4^T \phi_4(k) + \varepsilon_4(k) \tag{69}$$

Where $w_4 \in \Re^{n_4}$ and $v_4 \in \Re^{6 \times n_4}$ denote the constant ideal output and hidden layer weight matrices, $\phi_4(k) \in \Re^{n_4}$ is the activation function vector, n_4 is the number of hidden layer nodes, and $\varepsilon\left(z_4(k)\right)$ is the estimation error so that the bound $\left\|\varepsilon\left(z_4(k)\right)\right\| \le \varepsilon_{4m}$ holds for all $z_4(k) \in S$ in a compact set. Again, we hold the input weights constant and adapt the output weights only. We also replace the actual with estimated states to design the control input as

$$u(k) = \hat{w}_4^T(k)\phi\left(v_4^T \hat{z}_4(k)\right) = \hat{w}_4^T(k)\hat{\phi}_4(k) \tag{70}$$

where $\hat{z}_4(k) = \left[\hat{x}_1(k), \hat{x}_2(k), \hat{e}_1(k), l_6 \hat{e}_2(k), \hat{x}_{2d}(k), x_{1d}(k+2)\right]^T \in \Re^6$ is the input vector. Rewriting (57) and substituting (68) through (70), to get

$$
\begin{aligned}
e_2(k+1) &= x_2(k+1) - \hat{x}_{2d}(k+1) \\
&= \left(f_2(\cdot) + g_2(\cdot)\hat{w}_4^T(k)\hat{\phi}_4(k) + d_2(k)\right) - \hat{x}_{2d}(k+1) \\
&= f_2(\cdot) + g_2(\cdot)\left(\tilde{w}_4^T(k)\hat{\phi}_4(k) + w_4^T\phi_4(k) + w_4^T\tilde{\phi}_4(k)\right) + d_2(k) - \hat{x}_{2d}(k+1) \\
&= f_2(\cdot) + g_2(\cdot)\left(w_4^T(k)\phi_4(k)\right) + g_2(\cdot)\left(\zeta_4(k) + w_4^T\tilde{\phi}_4(k)\right) + d_2(k) - \hat{x}_{2d}(k+1) \\
&= f_2(\cdot) + g_2(\cdot)\left(u_d(k) - \varepsilon_4(k)\right) + g_2(\cdot)\left(\zeta_4(k) + w_4^T\tilde{\phi}_4(k)\right) + d_2(k) - \hat{x}_{2d}(k+1) \\
&= l_6 e_2(k) - g_2(\cdot)\varepsilon_4(k) + g_2(\cdot)\zeta_4(k) + g_2(\cdot)w_4^T\tilde{\phi}_4(k) + d_2(k)
\end{aligned}
\tag{71}
$$

where

$$\zeta_4(k) = \tilde{w}_4^T(k)\hat{\phi}_4(k) = \hat{w}_4^T(k)\hat{\phi}_4(k) - w_4^T(k)\hat{\phi}_4(k) \tag{72}$$

and

$$\tilde{\phi}_4(k) = \hat{\phi}_4(k) - \phi_4(k) \tag{73}$$

Equations (59) and (71) represent the closed-loop error dynamics. Next we derive the weight update law for the second action NN. Define

$$e_{a2}(k) = \hat{w}_4^T(k)\hat{\phi}_4(k) + \hat{Q}(k) \tag{74}$$

where $e_{a2}(k) \in \Re$ is the error, the subscript a2 stands for the second action NN, and the first term in (72) is the NN output or control input to the nonlinear system. Following a similar design, choose a quadratic objective function to minimize

$$E_{a2}(k) = \frac{1}{2}e_{a2}^2(k) \tag{75}$$

Define a gradient-based adaptation where the general form is given by

$$\hat{w}_4(k+1) = \hat{w}_4(k) + \Delta\hat{w}_4(k) \tag{76}$$

where

$$\Delta\hat{w}_4(k) = \alpha_4\left[-\frac{\partial E_{a2}(k)}{\partial\hat{w}_4(k)}\right] \tag{77}$$

or

$$\hat{w}_4(k+1) = \hat{w}_4(k) - \alpha_4\hat{\phi}_4(k)\left(\hat{w}_4^T(k)\hat{\phi}_4(k) + \hat{Q}(k)\right) \tag{78}$$

The proposed controller structure is shown in Figure 1. Next, in the following theorem, the uniformly ultimately boundedness (UUB) of the closed loop system is demonstrated through the use of Lyapunov function.

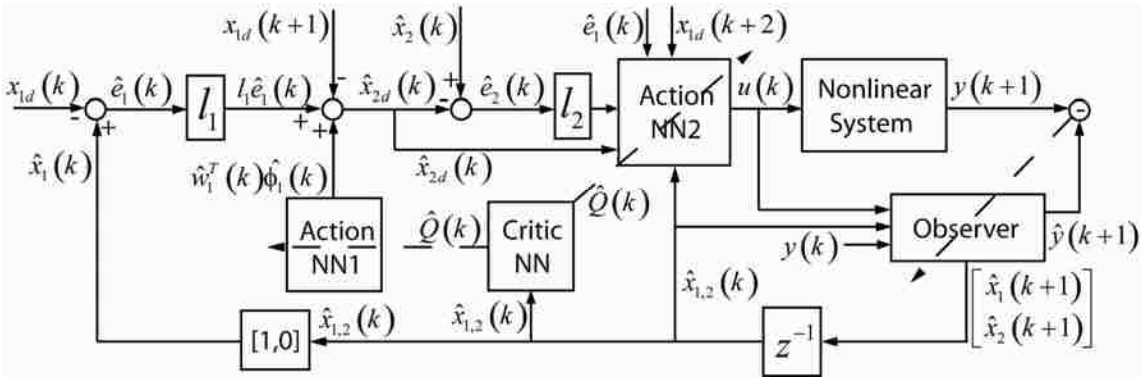

Figure 1 Adaptive-critic NN-based controller diagram.

Assumption 3 (Bounded Ideal Weights): Let w_1, w_2, w_3, and w_4 be the unknown output layer target weights for the observer, critic, and two action NNs, and assume that they are bounded above so that

$$\|w_1\| \le w_{1m},\ \|w_2\| \le w_{2m}, \|w_3\| \le w_{3m}, \text{ and } \|w_4\| \le w_{4m} \tag{79}$$

where $w_{om} \in R^+$, $w_{1m} \in R^+$ and $w_{2m} \in R^+$ represent the bounds on the unknown target weights, where the Frobenius norm [14] is used.

Fact 1: The activation functions are bounded above by known positive values so that

$$\left\|\tilde{\phi}_1(\cdot)\right\| \le \tilde{\phi}_{1m}, \left\|\tilde{\phi}_2(\cdot)\right\| \le \tilde{\phi}_{2m}, \left\|\tilde{\phi}_3(\cdot)\right\| \le \tilde{\phi}_{3m}, \text{ and} \left\|\tilde{\phi}_4(\cdot)\right\| \le \tilde{\phi}_{4m} \tag{80}$$

where $\hat{\phi}_{1m}, \tilde{\phi}_{1m} \in R^+, \hat{\phi}_{2m}, \tilde{\phi}_{2m} \in R^+, \hat{\phi}_{3m}, \tilde{\phi}_{3m} \in R^+ \text{and} \hat{\phi}_{4m}, \tilde{\phi}_{4m} \in R^+$ are the upper bounds.

Theorem 2: Consider the system given by (3) and (4), and the disturbance bounds d_{1m} and d_{2m} be known constants. Let the observer, critic, virtual control, and control input NN weight tuning be given by (23), (47), (67), and (78), respectively. Let the virtual control input and control input be given by (56) and (70), the tracking errors $e_1(k)$ and $e_2(k)$ and weight estimates $\hat{w}_1(k), \hat{w}_2(k), \hat{w}_3(k)$, and $\hat{w}_4(k)$ are UUB, with the bounds specifically given by (B.17), and with the design parameters selected as

$$0 < \alpha_1 \|\phi_1(k)\|^2 < 1 \tag{81}$$

$$0 < \alpha_2 \|\phi_2(k)\|^2 < 1 \tag{82}$$

$$0 < \alpha_3 \|\phi_3(k)\|^2 < 1 \tag{83}$$

$$0 < \alpha_4 \|\phi_4(k)\|^2 < 1 \tag{84}$$

$$|l_1| < \frac{1}{2} \tag{85}$$

$$|l_2| < \frac{\sqrt{3}}{3} \tag{86}$$

$$|l_3| < \frac{\sqrt{3}}{3} \tag{87}$$

$$|l_4| < \frac{\sqrt{3}}{3} \tag{88}$$

$$|l_5| < \frac{1}{\sqrt{5}} \tag{89}$$

$$|l_6| < \frac{\sqrt{3}}{3} \tag{90}$$

$$0 < \beta < \frac{\sqrt{2}}{2} \tag{91}$$

where α_1, α_2, α_3 and α_4 are NN adaptation gains, l_1, l_2, l_3, l_4, l_5, and l_6 are gains, and β is employed to define the *strategic* utility function.

Proof: See Appendix B. ■

Remark 1: A well-defined controller is developed in this paper since a single NN is utilized to approximate two nonlinear functions.

Remark 2: It is important to note that in this theorem there is no persistency of excitation condition (PE) and linearity in the parameters assumption condition for the NN observer and controller, in contrast with standard work in the discrete-time adaptive control since the first difference does not require the PE condition to prove the boundedness of the weights. Even though the input to the hidden-layer weight matrix is not updated and only the hidden to the output-layer weight matrix is tuned, the NN method relaxes the linear in the unknown parameter assumption. Additionally, the certainty equivalence principle is not used.

Remark 3: Generally, the separation principle used for linear systems does not hold for nonlinear systems, and hence it is relaxed in this paper for the controller design since the Lyapunov function is a quadratic function of system errors and weight estimation errors of the observer and controller NNs.

Remark 4: The NN weight tuning proposed in (23), (47), (67), and (78) renders a semi-recurrent NN due to the proposed weight tuning law even though a feedforward NN is utilized. Here the NN outputs are not fed as delayed inputs to the network whereas the outputs of each layer are fed as delayed inputs to the same layer. This semi-recurrent NN architecture renders a dynamic NN which is capable of predicting the state one step ahead.

Remark 5: It is only possible to show boundedness of all the closed–loop signals by using an extension of Lyapunov stability [14] due to the presence of approximation errors and bounded disturbances consistent with the literature.

Corollary 1: The proposed adaptive critic NN controller and the weight updating rules with parameter selection based on (81) through (91) cause the state $x_2(k)$ to approach the desired virtual control input $x_{2d}(k)$.

Proof: Combining (55) and (56), the difference between $\hat{x}_{2d}(k)$ and $x_{2d}(k)$ is given by

$$\hat{x}_{2d}(k) - x_{2d}(k) = \tilde{w}_3(k)\phi_3(k) - \varepsilon(z_3(k)) = \zeta_3(k) - \varepsilon_3(k) \tag{92}$$

where $\tilde{w}_3(k) \in \Re^{n_3}$ is the first action NN weight estimation error and $\zeta_3(k) \in \Re$ is defined in (60). Since both $\zeta_3(k) \in \Re$ and $\varepsilon_3(k)$ are bounded, $\hat{x}_{2d}(k)$ is bounded to $x_{2d}(k)$. In *Theorem 1*, we show that $e_2(k)$ is bounded, i.e., the state $x_2(k)$ is bounded to the virtual control signal $\hat{x}_{2d}(k)$. Thus the state $x_2(k)$ is bounded to the desired virtual control signal $x_{2d}(k)$.

VII. Results and Analysis

Lean operation of an SI engine allows low emissions and improved fuel efficiency. However, lean operation destabilizes the engine due to the cyclic dispersion of heat release. The controller is designed to stabilize the SI engine operating at lean conditions.

A. Daw Engine Model

Spark ignition (SI) engine dynamics can be expressed according to the Daw model as a class of nonlinear systems in non-strict feedback form [15] as

$$x_1(k+1) = AF(k) + F(k)x_1(k) - R \cdot F(k)CE(k)x_2(k) + d_1(k), \tag{93}$$

$$x_2(k+1) = (1 - CE(k))F(k)x_2(k) + (MF(k) + u(k)) + d_2(k), \tag{94}$$

$$y(k) = x_2(k)CE(k), \tag{95}$$

$$\varphi(k) = R\frac{x_2(k)}{x_1(k)}, \tag{96}$$

$$CE(k) = \frac{CE_{\max}}{1 + 100^{-(\varphi(k) - \varphi_m)/(\varphi_u - \varphi_l)}}, \tag{97}$$

$$\varphi_m = \frac{\varphi_u - \varphi_l}{2}, \tag{98}$$

Where $x_1(k)$ and $x_2(k)$ are total mass of air and fuel, respectively, in each cylinder which is unknown. The variable $y_1(k)$ is the heat release at k^{th} instance. The value of combustion efficiency $CE(k)$ is within the range of $0 < CE_{\min} < CE(k) < CE_{\max}$ which is typically unknown whereas the unknown residual gas fraction $F(k)$ is bounded by $0 < F_{\min} < F(k) < F_{\max}$. The terms $d_1(k)$ and $d_2(k)$ are unknown but bounded disturbances upper bounded by $|d_1(k)| < d_{1m}$ and $|d_2(k)| < d_{2m}$ with d_{1m} and d_{2m} being known positive scalars. To implement the observer, replace the following from the Daw model into the general case

$$\begin{aligned} f_1(\cdot) &= AF(k) + F(k)x_1(k) \\ g_1(\cdot) &= -R \cdot F(k)CE(k) \\ f_2(\cdot) &= (1 - CE(k))F(k)x_2(k) + MF(k) \\ g_2(\cdot) &= 1 \end{aligned} \tag{99}$$

and

$$\begin{aligned} f_{10} &= AF_0 + F_0\hat{x}_1(k) \\ g_{10} &= -R \cdot F_0CE_0 \\ f_{10} &= (1 - CE_0)F_0\hat{x}_2(k) + MF_0 \\ g_{10} &= 1 \end{aligned} \tag{100}$$

To implement the controller, replace the following in place of $f_1(\cdot)$ and $g_1(\cdot)$

$$\Phi(\cdot) = AF(k) + F(k)x_1(k) - R \cdot F(k)CE(k)x_2(k) + x_2(k) \tag{101}$$

To calculate the nominal values for equations (6) through (9), we run the engine at the desired equivalence ratio. That will give us the nominal fuel, air, and equivalence ratio - MF_0, AF_0 and φ_0. From those, combustion efficiency CE_0 is calculated.

B. Simulation Results

The controller is simulated in C in conjunction with the Daw model. The learning rates for the observer (81), critic (82), virtual control input (83), and control input (84) networks are 0.01, 0.01, 0.01, and 0.01, respectively. The gains l_1, l_2, l_3, l_4, l_5, and l_6 are selected as 0.05, 0.05, 0.04, 0.05, 0.2 and 0.1. The system constants CE_{max}, φ_l, and φ_u are chosen as 1, 0.66 , and 0.73. The critic constants β and N are 0.4 and 4. All NNs use 20 neurons with hyperbolic tangent sigmoid activation functions in the hidden layer.

The maximum moles a single cylinder holds is set as 0.021 to match the experimental engine constraint shown in the next section. Using this constant along with the following equations

$$\varphi = R\left(MF \diagup AF\right) \tag{102}$$

$$tm = \frac{MF}{mw_{fuel}} + \frac{AF}{mw_{air}} \tag{103}$$

Where mw_{fuel} and mw_{air} are molecular weights of fuel and air, respectively and tm is the maximum moles each cylinder is capable of holding. For each equivalence ratio set point, φ, MF and AF can be calculated.

The last two system variables: disturbances and stochastic effects are modeled as follows. First, we assume a Gaussian distribution governs the two effects. We may inject disturbances to the two states in equations (93) and (94) due to $d_1(k)$ and $d_2(k)$, but a simpler method is to perturb the equivalence ratio equation (96). This simplification is sufficient because the states are not measurable; therefore, the disturbances are increasingly complex and immeasurable. Stochastic effects alter the output, and through the combustion efficiency equation (97) and finally the output equation (95), this single perturbation effectively models the last two system variables. The final model uses a Gaussian distribution noise injected into equation (96) centered around the target equivalence ratio and deviation of 1% of the target equivalence ratio. The resulting

simulation output matches the output observed from the Ricardo engine shown in the preceding section. All simulations ran for 5000 cycles uncontrolled first, then 5000 cycles controlled.

Figure 2 shows two heat release return maps, one controlled and the other uncontrolled, for an equivalence ratio of 0.89. Each subfigure shows heat release for the next time step versus the current time step. Points centered along the 45 degree line represent heat release values that are equal to the next step heat release. Note the clustering of the points around the mean heat release of 870J. The square represents the target heat release. The relatively high equivalence ratio exhibits little dispersion, indicated by little or no stray points away from the central cluster. The left uncontrolled plot is similar to the right controlled plot, because the controller is quiescent due to the simulated engine performing well. There are no complete misfires, but the heat release variation can be clearly seen.

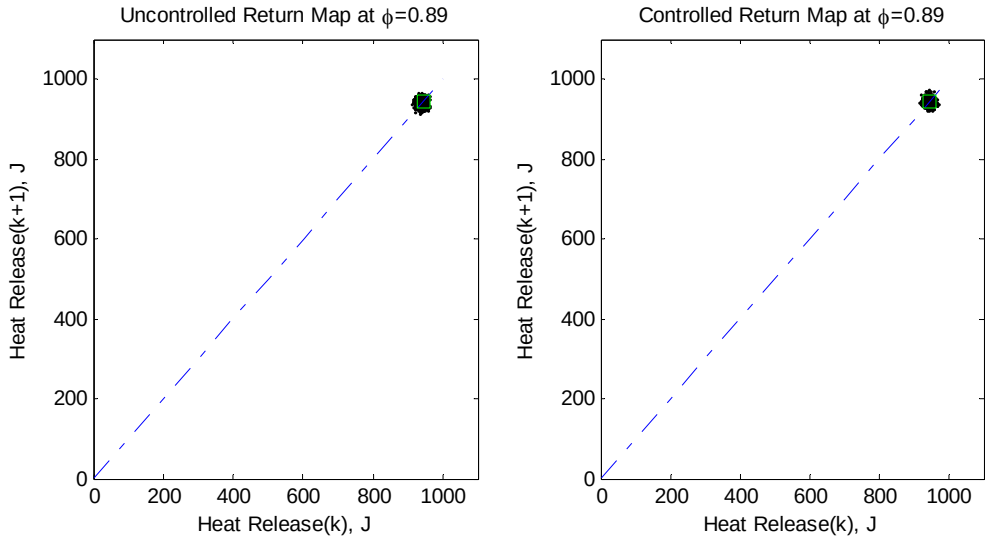

Figure 2 Uncontrolled and controlled heat release return map at φ=0.89. Heat release at k+1 instance is plotted against heat release at k instance.

Figure 3 shows the time series of the heat release and control input at the same equivalence ratio. The controller activates after several thousand cycles, indicated by the fluctuation of the control output. The controller converges quickly and to a stable operation point. The presence of spikes in the control output indicates a decline in heat release such as a misfire, translating into additional fuel control to counteract.

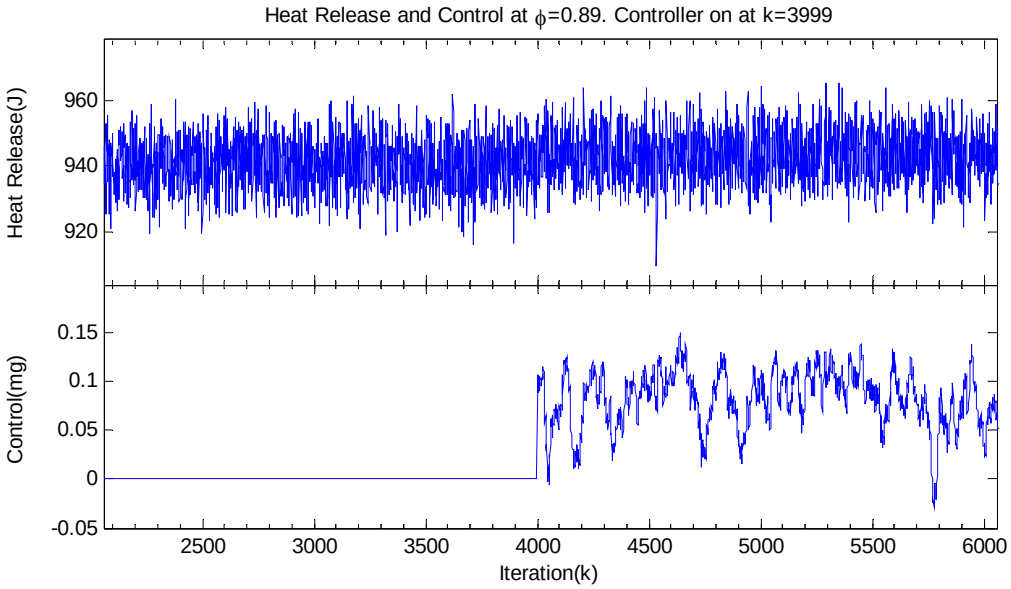

Figure 3 Heat release and control input at φ=0.89. Controller turns on at k=4000. Note the almost instant learning convergence of the controller.

Figures 4 and 5 present another set point at 0.79. Similar features appear compared to the previous equivalence ratio, except with higher frequency and amplitude of dispersion, indicated by a larger cluster of heat release data points on the 45 degree line. Improvements shown reflect the assertion of the control action. The cluster is therefore, tighter on the right subfigure.

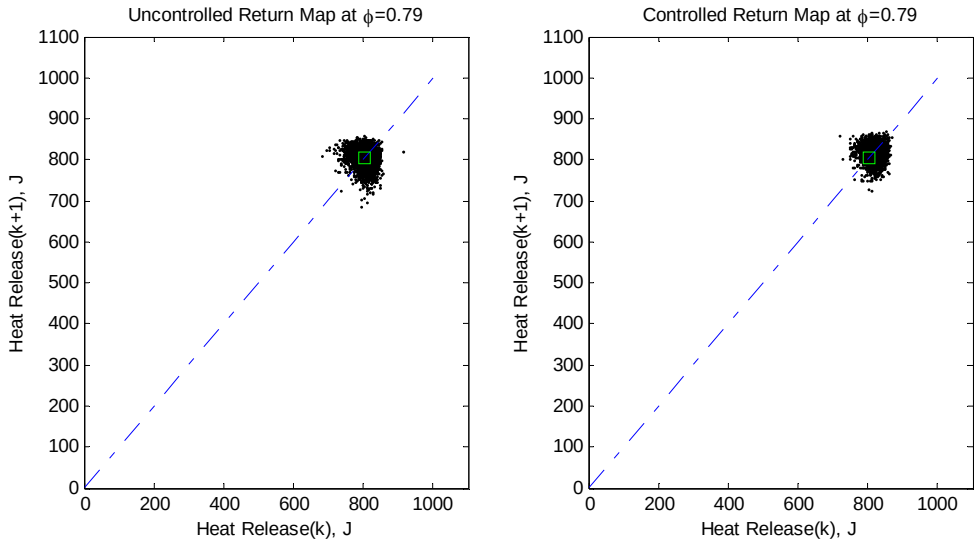

Figure 4 Uncontrolled and controlled heat release return map at φ=0.79.

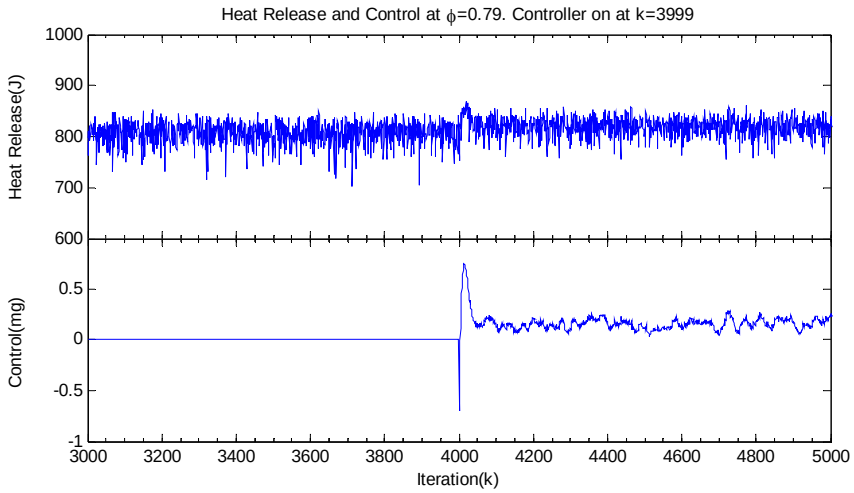

Figure 5 Heat release and control input at φ=0.79.

In order to quantify the performance of the controller, we compare the coefficient of variation (COV), which is the standard deviation normalized by dividing by the mean of the heat release. As the COV decreases, the standard deviation decreases, which indicates that the engine heat release is more stable compared to higher COV. The controller performs better, and the return map consequently should approach the target value. Table 1 tabulates all of the data from the simulation. The COV of each set point decreased drastically (shown with a negative sign) as the controller operated. The performance exceeded the improvement expected due to the slight increase in the mean fuel input. Next, we show that experimental data supports the simulation data.

Table 1 Coefficient of variation (COV) and fuel data for each of the six set points.

	COV		%COV	%Fuel
φ	**Uncontrolled**	**Controlled**	**Change**	**Change**
0.89	0.0080	0.0077	-4.1	0.29
0.84	0.0090	0.0087	-3.3	0.11
0.79	0.0267	0.0221	-17.0	0.66
0.77	0.0475	0.0435	-8.3	0.48
0.75	0.1217	0.1071	-12.0	0.56
0.72	0.2373	0.2128	-10.3	0.48

C. Ricardo Engine

The experimental results are collected from a Ricardo Hydra engine with a modern four valve Ford Zetek head. A single cylinder runs at 1000 rpm with shaft encoders to signal each crank angle degree and start of cycle. There are 720° per engine cycle.

In the cylinder, a piezoelectric pressure transducer records pressure every crank angle degree. Combustion is considered to take place between 345° to 490°, for a total of 145 pressure measurements. The cylinder pressure is integrated along with volume during the 17.7 ms calculation window. All communications are completed at this time. The output of our controller controls the fuel input. This is controlled by a TTL signal to a fuel injector driver circuit.

All signals communicate through a custom interface board using a microcontroller. The board interfaces with the PC through a parallel port and with the engine hardware through an analog signal.

D. Experimental Results

All constants given in the simulation section are used in the experiment. The first operation for an engine run is to measure the air flow and nominal fuel. The desired equivalence ratio is given by (102), where *MF* is nominal mass of fuel, *AF* is nominal mass of air, and *R* is the stoichiometric ratio.

These values are loaded into the controller. Ambient pressure is used to reference the in-cylinder pressure when the exhaust valve is fully open and subtracted from the combustion pressure measurements. Uncontrolled and controlled data were collected at equivalence ratios of 0.8, 0.78, 0.75, and 0.72. The uncontrolled engine ran for 5,000 cycles and then the controller is turned on for another 5,000 cycles. Steady state was ensured prior to data collection by measuring stable exhaust temperature.

Figure 6 shows two heat release return maps, one controlled and the other uncontrolled, for the equivalence ratio of 0.8. The target heat release is at 850J. Figure 7 shows the time series of the heat release and control input for the same equivalence ratio. Small changes indicate a quiescent controller due to the near stoichiometric set point. Define the state and output tracking errors:

$$\begin{aligned} \hat{e}_1(k) &= \hat{x}_1(k) - x_{1d}(k) \\ \hat{e}_2(k) &= \hat{x}_2(k) - \hat{x}_{2d}(k) \\ \hat{e}_y(k) &= \hat{y}(k) - y(k) \end{aligned} \tag{104}$$

where $\hat{e}_1(k)$, $\hat{e}_2(k)$, and $\hat{e}_y(k)$ are state 1, state 2, and output tracking errors, respectively.

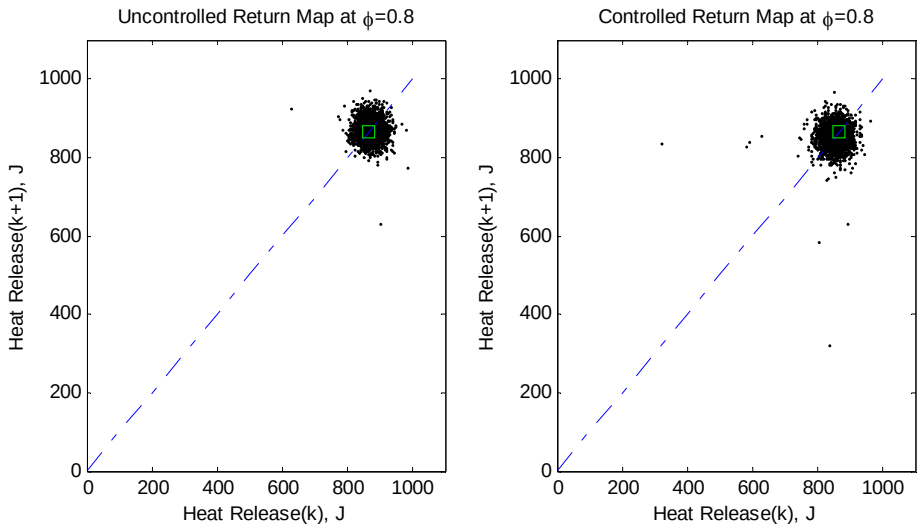

Figure 6 Uncontrolled and controlled heat release return map at φ=0.8. Heat release at $k+1$ instance is plotted against heat release at k instance.

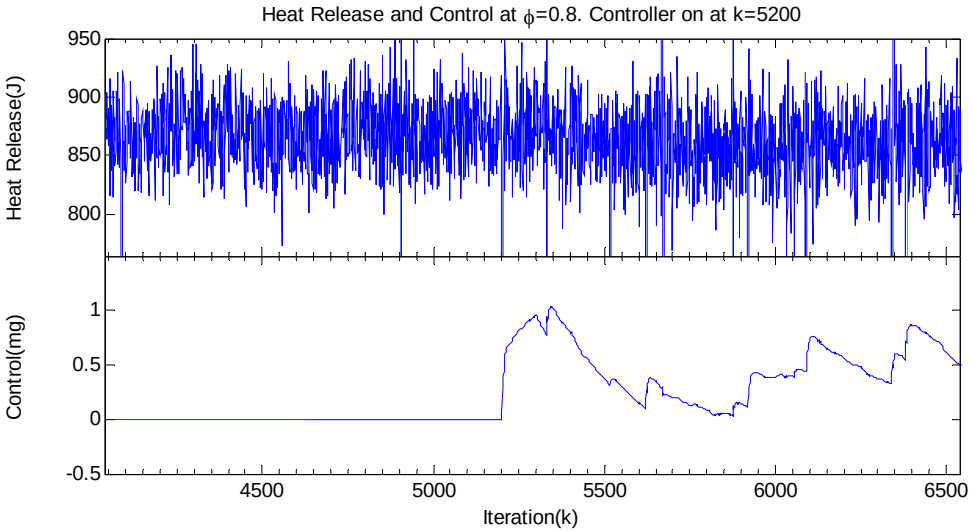

Figure 7 Heat release and control input at φ=0.8. Controller turns on at k=5200. Note the almost instant learning convergence of the controller.

Figure 8 shows the controller state tracking errors at equivalence ratio of 0.8. The range represents tracking error in percentage over and under the desired state trajectories. State one tracking error is considerably better than state two tracking. The second state tracks within 0.3%, therefore, both are performing well. The spikes indicate unsuccessful tracking. Consequently, the observer and controller converged together to the desired states and estimated states, generating a stable error system.

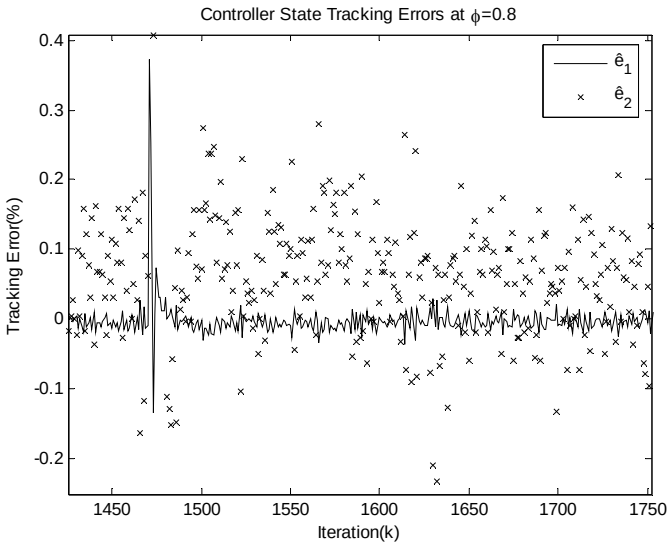

Figure 8 State tracking errors.

Figure 9 shows the output tracking error in the same form as the state tracking error. Immediate observation shows an extremely high error rate. The observer performance is abysmal. Nonetheless, this signal fed into the NN controller allows for the critical performance factor, state tracking errors, to converge and stabilize. It is not critical for one signal to track perfectly, rather the system as a whole. Moreover, theorem 1 proved the boundedness of the output estimation. In conjunction with the natural bound of the engine output, the tracking error will always be bounded. The extreme fluctuation of the observer output may be the key to the responsiveness of the controller as a whole.

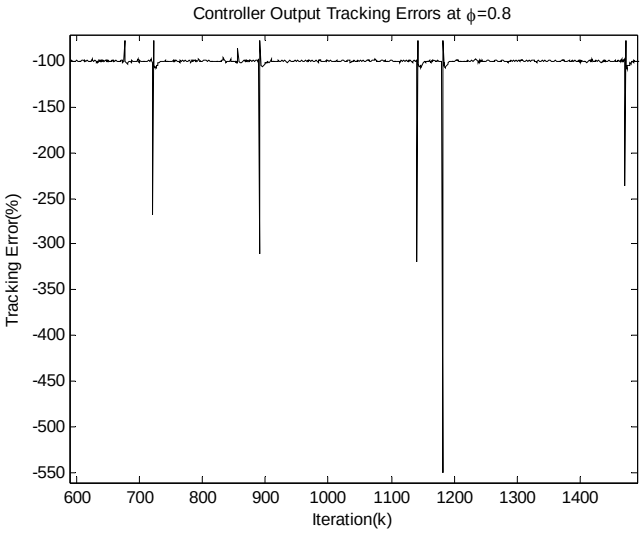

Figure 9 Output tracking error.

Figure 10 shows the return map of heat release for an equivalence ratio of 0.72. Note that as the equivalence ratio decreases, the return map spreads out and dispersion increases. Figure 11 is the corresponding heat release and time series of the control input. Misfires increase in frequency, as shown by the negative heat release spikes due to heat transfer from the cylinder to the environment without internal generation of useful work by combustion.

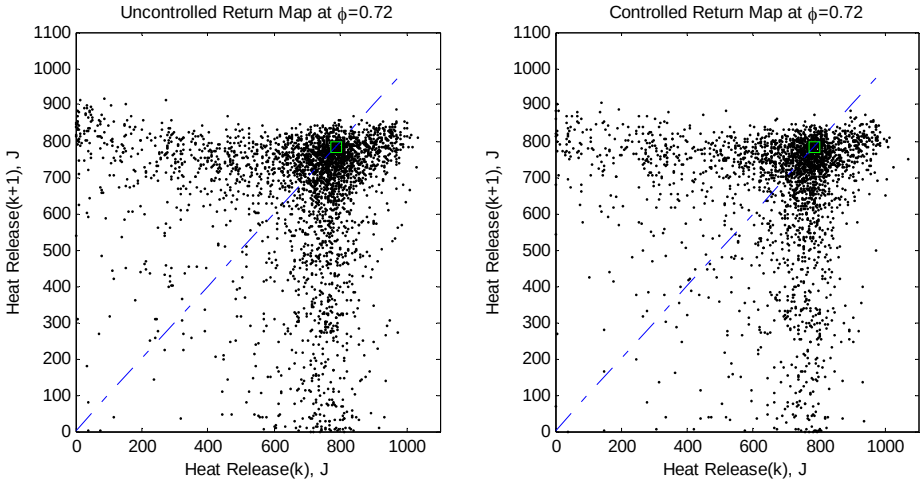

Figure 10 Uncontrolled and controlled heat release return map at φ=0.72.

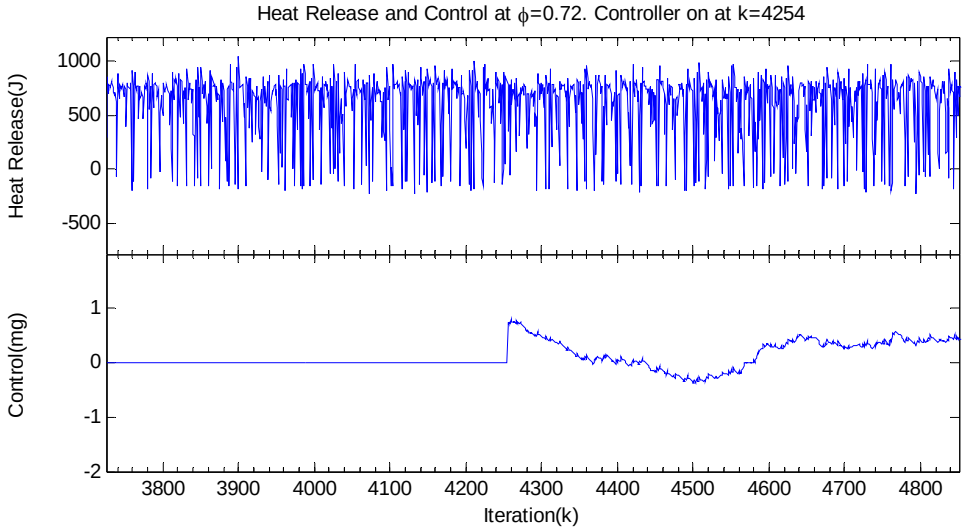

Figure 11 Heat release and control input at φ=0.72.

Figure 12 shows increasing difficulty of the observer and controller to generate a low state tracking error compared to the previous case. As the engine operates in a leaner mode, overall dispersion increases, thus degrading observer performance. Although the performance is reduced, the tracking error is well within satisfactory performance. Figure

13 shows the output tracking error. At the lower equivalence ratio, it is performing better than the previous equivalence ratio. This may be due to the memory effect of past engine cycles contributing to the residuals in the current cycle. At a near stoichiometric ratio of fuel over air, little dispersion occurs, resulting in similar cylinder chemistry content before each power cycle. Stochastic effects dominate and destroy predictability. The high observer learning rate decimates the tracking ability. On the other hand, at lower equivalence ratios, higher dispersion and misfires create patterns of predictable residuals. The observer exploits the pattern recognition power of NN to drastically improve its performance.

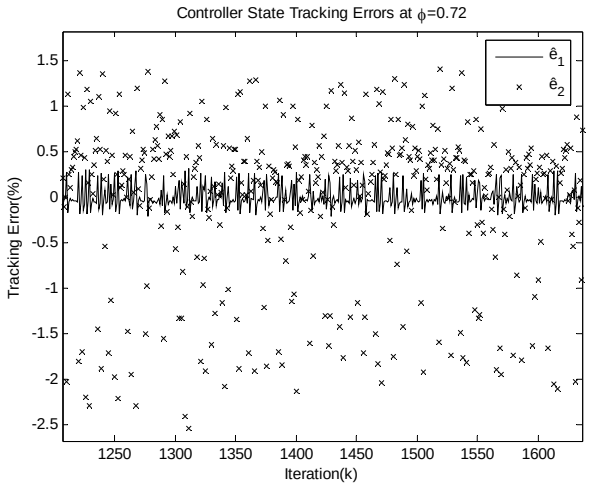

Figure 12 State tracking error with corresponding mean value.

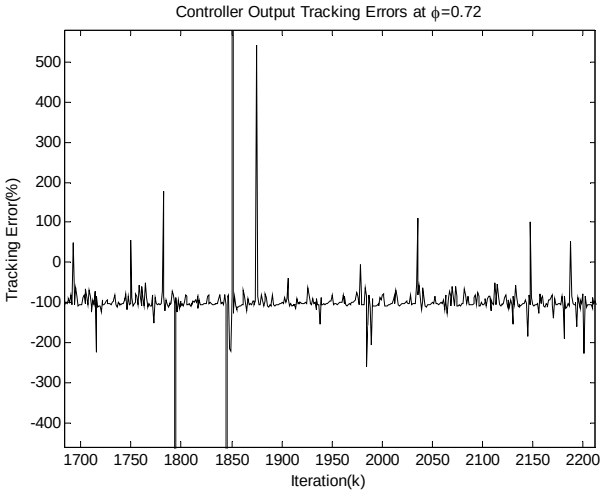

Figure 13 Output tracking error.

Figure 14 shows a detailed view of 35 controlled cycles at an equivalence ratio of 0.72. The controller generates decreasing control during cycles when the heat release is steady, indicated by cycles between 4947 to 4954 and between 4963 to 4769. However, during misfires or extreme dispersion in heat release, the controller attempts to compensate for the drop in heat release by pushing the control up, indicated by cycles 4943, 4944, 4955, etc. Note the general increase in control during sequential or near sequential misfires such as between cycles 4955 to 4962. The controller compensates after a one cycle delay in the positive direction and attempts to recover the engine heat release towards the target point. It is difficult to determine success on cycles with no misfire, because no heat release plots are available for uncontrolled case during the same cycles when the controller is operating for comparison. Overall, the controller performs to general expectation.

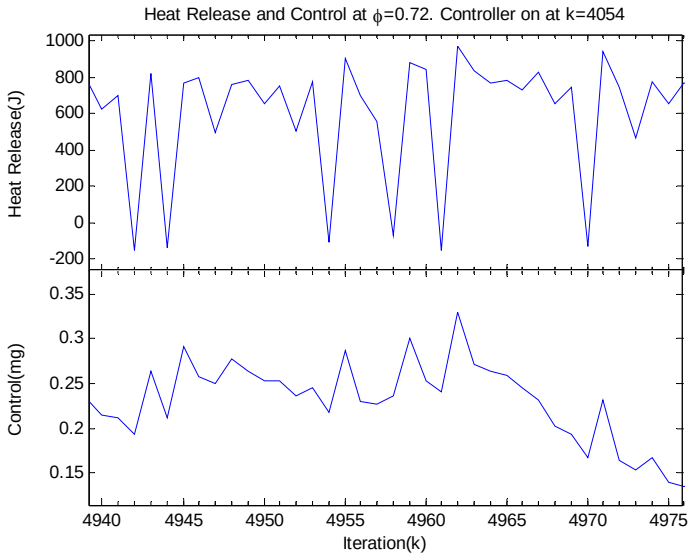

Figure 14 Detailed view of 35 controlled cycles at φ=0.72

Table 2 shows the improved COV when the controller is in operation compared to uncontrolled engine and also the corresponding change in nominal fuel. An improvement in the COV may be artificial due to an increase in fuel input. However, this is not the case for this controller. At all equivalence ratios except 0.75, the increase in fuel input is well within the tolerance of the equipment. On average, the COV decreased significantly by 16% compared to controlled case.

Table 2 Coefficient of variation (COV) and fuel data for each of the four set points.

φ	COV Uncontrolled	COV Controlled	%COV Change	%Fuel Change
0.80	0.1140	0.0913	-19.9	0.76
0.78	0.1457	0.1318	-9.5	0.65
0.75	0.3438	0.2522	-26.6	2.48
0.72	0.6088	0.5712	-6.2	1.07

Due to reduced cyclic dispersion and fewer misfires and low energy cycles, a gain of approximately 8% in indicated fuel conversion efficiency was observed for controlled engine operation.

The COV and fuel change data indicates an improved performance compared to previous controller without any optimization [16]. The average drop in COV was 30% between uncontrolled and controlled compared to 15.7% for the current controller. Although this seems to indicate a decrease in performance, we must consider the increase in average fuel input. The previous controller increased the average fuel by 2.5%. This is within the detection error. This controller, however, average below the detection error of 1%. The controller fuel increases negligibly while approaching the performance of the previous controller which in turn demonstrates the optimality of the controller.

VIII. CONCLUSIONS

The controller presented successfully controlled a SI engine to reduce cyclic dispersion under lean operation. The system is modeled under a non-strict feedback nonlinear discrete-time system. It converged upon a near *optimal* solution through the use of a long-term strategic utility function even though the exact dynamics are not known beforehand. It was shown experimentally that the COV is reduced when the controller is turned on. At the same time, the average fuel input did not change significantly; therefore, the improvements are solely due to the effects of the controller. The output is stable, as predicted by the Lyapunov proof.

We also provided the emissions data for several set points in Appendix A. It is important to note that the emissions data uncertainty may be 5% or more. Therefore, the data presented is used for indicating general trends, and not as absolute improvement.

However, lean operation in general is proven to decrease emissions compared to stoichiometric operation regardless of the data inaccuracies presented. Both NO_x and unburned hydrocarbons reduced significantly compared to the uncontrolled case. However, the most significant drop is between lean and stoichiometric equivalence ratios. This is due to the controller's ability to successfully decrease dispersion.

APPENDIX A

Tables A.1 and A.2 show the improvement in emissions for several equivalence ratios. The improvement is better than what we have seen before [16] using another controller. NO_x is reduced by around 30 to 40% from uncontrolled scenario. However CO_2 remains unchanged, whereas O_2 decreased by about 4 to 10%, as well as unburned hydrocarbons (uHC) decreasing with control by 8% due to reduced cyclic dispersion.

Table A.1 Emissions data for select equivalence ratios.

	Uncontrolled			Controlled			Change(%)		
φ	**CO_2(%)**	**CO(%)**	**O_2(%)**	**CO_2(%)**	**CO(%)**	**O_2(%)**	**CO_2**	**CO**	**O_2**
0.80	7.5	0.1	8.0	7.5	0.1	7.0	-0.5	0.0	-12.4
0.78	7.7	0.1	8.3	7.6	0.1	8.6	-0.5	0.0	3.9
0.75	8.9	0.1	7.7	9.1	0.1	7.5	1.8	0.0	-2.7

Table A.2 Unburned hydrocarbon (uHC) emission data.

	Uncontrolled		Controlled		Change(%)	
φ	**uHC (ppm C_1)**	**NO_X (ppm)**	**uHC (ppm C_1)**	**NO_X (ppm)**	**uHC**	**NO_X**
0.80	8732.00	1284	8919.00	761	2.1	-40.7
0.78	9457.00	400	9354.00	802	-1.1	100.5
0.75	11154.00	215	9347.00	150	-16.2	-30.2

APPENDIX B

Proof of Theorem 1: Define the Lyapunov function

$$J(k)=\sum_{i=1}^{10}J_i(k)=\frac{\gamma_1}{5}e_1^2(k)+\frac{\gamma_2}{3}e_2^2(k)+\sum_{j=3}^{6}\frac{\gamma_j}{\alpha_{j-2}}\tilde{w}_j^T(k)\tilde{w}_j(k)+$$
$$\gamma_7\zeta_2^2(k-1)+\frac{\gamma_8}{3}\tilde{x}_1^2(k)+\frac{\gamma_9}{3}\tilde{x}_2^2(k)+\frac{\gamma_{10}}{3}\tilde{y}^2 \quad \text{(B.1)}$$

where $0<\gamma_i, i\in\{1,...,6\}$ are auxiliary constants; the NN weights estimation errors $\tilde{w}_1^T(k+1)$, $\tilde{w}_2^T(k+1)$, $\tilde{w}_3^T(k+1)$, and $\tilde{w}_4^T(k+1)$ are defined in (23), (47), (67), and (78), by subtracting their respective ideal weights $w_i, i\in\{1,2,3,4\}$ on both sides; the observation errors $\tilde{x}_1(k+1)$, $\tilde{x}_2(k+1)$, are defined in (20) and (21), respectively; the system errors $e_1(k+1)$ and $e_2(k+1)$ are defined in (59) and (71), respectively; and $\alpha_i, i\in\{1,2,3,4\}$ are NN adaptation gains. The Lyapunov function (B.1) obviates the need for the separation principle. Take the first term and the first difference using (59) to get

$$\begin{aligned} J_1(k) &= \tfrac{\gamma_1}{5}e_1^2(k) \\ \tfrac{5}{\gamma_1}\Delta J_1(k) &= e_1^2(k+1)-e_1^2(k) \\ &= \left(-\zeta_3(k)-w_3^T\tilde{\phi}_3(k)+\varepsilon_3(k)-l_5\hat{e}_1(k)-e_2(k)+d_1(k)\right)^2-e_1^2(k) \\ &= \left(-\zeta_3(k)-w_3^T\tilde{\phi}_3(k)+\varepsilon_3(k)-l_5\left(\tilde{x}_1(k)+e_1(k)\right)-e_2(k)+d_1(k)\right)^2-e_1^2(k) \end{aligned} \tag{B.2}$$

Invoke the Cauchy-Schwarz inequality defined as

$$\left(a_1b_1+...+a_nb_n\right)^2\le\left(a_1^2+...+a_n^2\right)\left(b_1^2+...+b_n^2\right) \tag{B.3}$$

simplify to get

$$\begin{aligned} \tfrac{1}{\gamma_1}\Delta J_1(k) &\le \left(\zeta_3^2(k)+l_5^2\tilde{x}_1^2(k)+l_5^2e_1^2(k)+e_2^2(k)+\left(\varepsilon_3(k)-w_3\tilde{\phi}_3(k)+d_1(k)\right)^2\right)-\tfrac{1}{5}e_1^2(k) \\ \Delta J_1(k) &\le \gamma_1\zeta_3^2(k)+\gamma_1l_5^2\tilde{x}_1^2(k)+\gamma_1l_5^2e_1^2(k)+\gamma_1e_2^2(k)+\gamma_1\left(\varepsilon_3(k)-w_3\tilde{\phi}_3(k)+d_1(k)\right)^2-\tfrac{\gamma_1}{5}e_1^2(k) \\ &\le \gamma_1l_5^2\tilde{x}_1^2(k)+\gamma_1l_5^2e_1^2(k)+\gamma_1e_2^2(k)+\gamma_1\zeta_3^2(k)+\gamma_1\left(\varepsilon_{3\mathrm{m}}+w_{3\mathrm{m}}\tilde{\phi}_{3\mathrm{m}}+d_{1\mathrm{m}}\right)^2-\tfrac{\gamma_1}{5}e_1^2(k) \end{aligned} \tag{B.4}$$

Take the second term, substitute (71), invoke Cauchy-Schwarz inequality, and simplify

$$\Delta J_2(k)\le 3l_6^2e_2^2(k)+3g_{2\max}^2\zeta_4^2(k)+\gamma_2\left(d_{2\mathrm{m}}+g_{2\max}\varepsilon_{4\mathrm{m}}+g_{2\max}w_{4\mathrm{m}}\tilde{\phi}_{4\mathrm{m}}\right)^2-e_2^2(k) \tag{B.5}$$

Take the third term, substitute (23), invoke Cauchy-Schwarz inequality, and simplify

$$\begin{aligned} \Delta J_3(k)\le{}& -\gamma_3\left(1-\alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\left(\hat{w}_1(k)\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^2+ \\ & 2\gamma_3\left(w_{1\mathrm{m}}\hat{\phi}_{1\mathrm{m}}\right)^2+2\gamma_3l_4^2\tilde{y}^2(k)-\gamma_3\zeta_1^2(k) \end{aligned} \tag{B.6}$$

Take the fourth term, substitute (47), invoke Cauchy-Schwarz inequality, and simplify

$$\begin{aligned} \Delta J_4(k)\le{}& -\gamma_4\left(1-\alpha_2\left\|\hat{\phi}_2(k)\right\|^2\right)\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)^2-\gamma_4\zeta_2^2(k)+ \\ & 2\gamma_4\beta^2\zeta_2^2(k-1)+2\gamma_4\left(w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}}(1+\beta)+\beta^{N+1}\right)^2 \end{aligned} \tag{B.7}$$

Take the fifth term, substitute (67), invoke Cauchy-Schwarz inequality, and simplify

$$\begin{aligned}\Delta J_5(k) \le -\gamma_5\left(1-\alpha_3\left\|\hat{\phi}_3(k)\right\|^2\right)\left(\hat{Q}(k)+\hat{w}_3^T(k)\hat{\phi}_3(k)\right)^2 + \\ 2\gamma_5\zeta_2^2(k)+2\gamma_5\left(w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}}+w_{3\mathrm{m}}\hat{\phi}_{3\mathrm{m}}\right)^2-\gamma_5\zeta_3^2(k)\end{aligned} \tag{B.8}$$

Take the sixth term, substitute (78), invoke Cauchy-Schwarz inequality, and simplify

$$\begin{aligned}\Delta J_6(k) = -\gamma_6\left(1-\alpha_4\left\|\hat{\phi}_4(k)\right\|^2\right)\left(\hat{w}_4^T(k)\hat{\phi}_4(k)+\hat{Q}(k)\right)^2 + \\ 2\gamma_6\left(w_{4\mathrm{m}}\hat{\phi}_{\mathrm{m}}+w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}}\right)^2+2\gamma_6\zeta_2^2(k)-\gamma_6\zeta_4^2(k)\end{aligned} \tag{B.9}$$

Take the seventh term, set $\gamma_7 = 2\gamma_4\beta^2$

$$\Delta J_7(k) = 2\gamma_4\beta^2\zeta_2^2(k)-2\gamma_4\beta^2\zeta_2^2(k-1) \tag{B.10}$$

Take the eighth term, substitute (20), invoke Cauchy-Schwarz inequality, and simplify

$$\Delta J_8(k) \le \gamma_8 l_2^2\tilde{y}^2(k)+\gamma_8\tilde{x}_2^2(k)+\gamma_8\left(w_{3\mathrm{m}}\phi_{3\mathrm{m}}+f_{10}+\varepsilon_{3\mathrm{m}}+d_{1\mathrm{m}}\right)^2-\tfrac{\gamma_8}{3}\tilde{x}_1^2(k) \tag{B.11}$$

Take the ninth term, substitute (21), invoke the Cauchy-Schwarz inequality, and simplify

$$\begin{aligned}\Delta J_9(k) \le \gamma_9\left(f_{20}+\left(g_{20}+g_{2\max}\right)w_{4\mathrm{m}}\hat{\phi}_{4\mathrm{m}}+f_{2\max}+d_{2\mathrm{m}}\right)^2 + \\ \gamma_9\left(g_{20}+g_{2\max}\right)\zeta_4(k)+\gamma_9 l_3^2\tilde{y}^2(k)-\tfrac{\gamma_9}{3}\tilde{x}_2^2(k)\end{aligned} \tag{B.12}$$

Take the tenth term, substitute (22), invoke the Cauchy-Schwarz inequality, and simplify

$$\Delta J_{10}(k) \le \gamma_{10}\zeta_1^2(k)+\gamma_{10}l_1^2\tilde{y}(k)+\gamma_{10}\left(w_{1\mathrm{m}}\tilde{\phi}_{1\mathrm{m}}+\varepsilon_{1\mathrm{m}}\right)-\tfrac{\gamma_{10}}{3}\tilde{y}^2(k) \tag{B.13}$$

Combine (B.4) through (B.13) to get the first difference of the Lyapunov function

$$\begin{aligned}\Delta J \le &-\left(\tfrac{\gamma_1}{5}-\gamma_1 l_5^2\right)e_1^2(k)-\left(\tfrac{\gamma_2}{3}-\gamma_1-\gamma_2 l_6^2\right)e_2^2(k)-\left(\gamma_3-\gamma_{10}\right)\zeta_1^2(k)-\left(\tfrac{\gamma_9}{3}-\gamma_8\right)\tilde{x}_2^2(k) \\ &-\left(\gamma_5-\gamma_1\right)\zeta_3^2(k)-\left(\gamma_6-\gamma_2 g_{2\max}^2-\gamma_9\left(g_{20}+g_{2\max}\right)\right)\zeta_4^2(k)-\left(\tfrac{\gamma_8}{3}-\gamma_1 l_5^2\right)\tilde{x}_1^2(k) \\ &-\left(\gamma_4-2\gamma_5-2\gamma_6-2\gamma_4\beta^2\right)\zeta_2^2(k)-\left(\tfrac{\gamma_{10}}{3}-2\gamma_3 l_4^2-\gamma_8 l_2^2-\gamma_9 l_3^2-\gamma_{10}l_1^2\right)\tilde{y}^2(k)+D_M^2 \\ &-\gamma_3\left(1-\alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\left(\hat{w}_1(k)\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^2 \\ &-\gamma_4\left(1-\alpha_2\left\|\hat{\phi}_2(k)\right\|^2\right)\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)^2 \\ &-\gamma_5\left(1-\alpha_3\left\|\hat{\phi}_3(k)\right\|^2\right)\left(\hat{Q}(k)+\hat{w}_3^T(k)\hat{\phi}_3(k)\right)^2 \\ &-\gamma_6\left(1-\alpha_4\left\|\hat{\phi}_4(k)\right\|^2\right)\left(\hat{w}_4^T(k)\hat{\phi}_4(k)+\hat{Q}(k)\right)^2\end{aligned} \tag{B.14}$$

where

$$
\begin{aligned}
D_{\mathrm{m}}^2 = {} & \gamma_1\left(\varepsilon_{3\mathrm{m}} + w_{3\mathrm{m}}\tilde{\phi}_{3\mathrm{m}} + d_{1\mathrm{m}}\right)^2 + \gamma_2\left(d_{2\mathrm{m}} + g_{2\max}\varepsilon_{4\mathrm{m}} + g_{2\max}w_{4\mathrm{m}}\tilde{\phi}_{4\mathrm{m}}\right)^2 + \\
& 2\gamma_3\left(w_{1\mathrm{m}}\hat{\phi}_{1\mathrm{m}}\right)^3 + 2\gamma_4\left(w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}}(1+\beta) + \beta^{N+1}\right)^2 + 2\gamma_5\left(w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}} + w_{3\mathrm{m}}\hat{\phi}_{3\mathrm{m}}\right)^2 + \\
& 2\gamma_6\left(w_{4\mathrm{m}}\hat{\phi}_{\mathrm{m}} + w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}}\right)^2 + \gamma_8\left(w_{3\mathrm{m}}\phi_{3\mathrm{m}} + f_{10} + \varepsilon_{3\mathrm{m}} + d_{1\mathrm{m}}\right)^2 + \\
& \gamma_9\left(f_{20} + (g_{20} + g_{2\max})w_{4\mathrm{m}}\hat{\phi}_{4\mathrm{m}} + f_{2\max} + d_{2\mathrm{m}}\right)^2 + \gamma_{10}\left(w_{1\mathrm{m}}\tilde{\phi}_{1\mathrm{m}} + \varepsilon_{1\mathrm{m}}\right)^2
\end{aligned}
\tag{B.15}
$$

Select

$$
\begin{aligned}
& \gamma_1 > 5\gamma_1 l_5^2;\ \ \gamma_2 > 3\gamma_1 + 3\gamma_2 l_6^2;\ \ \gamma_3 > \gamma_{10};\ \ \gamma_4 > 2\gamma_5 + 2\gamma_6 + 2\gamma_4\beta^2;\ \ \gamma_5 > \gamma_1; \\
& \gamma_6 > \gamma_2 g_{2\max}^2 + \gamma_9(g_{20} + g_{2\max});\ \ \gamma_7 = 2\gamma_4\beta^2;\ \ \gamma_8 > 3\gamma_1 l_5^2;\ \ \gamma_9 > 3\gamma_8; \\
& \gamma_{10} > 6\gamma_3 l_4^2 + 3\gamma_8 l_2^2 + 3\gamma_9 l_3^2 + 3\gamma_{10} l_1^2;
\end{aligned}
\tag{B.16}
$$

This implies $\Delta J(k) < 0$ as long as (81) through (91) hold *and* the following hold.

$$
\begin{aligned}
& |e_1(k)| > \frac{D_M}{\sqrt{\frac{\gamma_1}{5} - \gamma_1 l_5^2}};\ or\ \ |e_2(k)| > \frac{D_M}{\sqrt{\frac{\gamma_2}{3} - \gamma_1 - \gamma_2 l_6^2}};\ or\ |\zeta_1(k)| > \frac{D_M}{\sqrt{\gamma_3 - \gamma_{10}}}; \\
& or\,|\zeta_2(k)| > \frac{D_M}{\sqrt{\gamma_4 - 2\gamma_5 - 2\gamma_6 - 2\gamma_4\beta^2}};\ or\ |\zeta_3(k)| > \frac{D_M}{\sqrt{\gamma_5 - \gamma_1}};\ \ or \\
& or\,|\zeta_4(k)| > \frac{D_M}{\sqrt{\gamma_6 - \gamma_2 g_{2\max}^2 - \gamma_9(g_{20} + g_{2\max})}};or \\
& or\,|\tilde{x}_1(k)| > \frac{D_M}{\sqrt{\frac{\gamma_8}{3} - \gamma_1 l_5^2}};\ or\ |\tilde{x}_2(k)| > \frac{D_M}{\sqrt{\frac{\gamma_9}{3} - \gamma_8}};\ \ or \\
& |\tilde{y}(k)| > \frac{D_M}{\sqrt{\frac{\gamma_{10}}{3} - 2\gamma_3 l_4^2 - \gamma_8 l_2^2 - \gamma_9 l_3^2 - \gamma_{10} l_1^2}};
\end{aligned}
\tag{B.17}
$$

REFERENCES

[1] M. Krstic, I. Kanellakopoulos, and P. Kokotovic, Nonlinear and Adaptive Control Design: John Wiley & Sons, Inc, 1995.

[2] S. S. Ge, T. H. Lee, G. Y. Li, and J. Zhang, "Adaptive NN control for a class of discrete-time nonlinear systems," Int. J. Contr., vol. 76, pp. 334-354, 2003.

[3] F. C. Chen and H. K. Khalil, "Adaptive control of a class of nonlinear discrete-time systems using neural networks," IEEE Trans. Automat. Contr, vol. 40, pp. 791-801, 1995.

[4] J. Si, in NSF Workshop on Learning and Approximate Dynamic Programming, Playacar, Mexico, 2002.

[5] P. J. Werbos, Neurocontrol and supervised learning: An overview and evaluation. New York: Van Nostrand Reinhold, 1992.

[6] J. J. Murray, C. Cox, G. G. Lendaris, and R. Saeks, "Adaptive dynamic programming," IEEE Trans. Syst., Man, Cybern., vol. 32, pp. 140-153, 2002.

[7] D. P. Bertsekas and J. N. Tsitsiklis, Neuro-Dynamic Programming. Balmont, MA: Athena Scientific, 1996.

[8] J. Si and Y. T. Wang, "On-line learning control by association and reinforcement," IEEE Trans. on Neural Networks, vol. 12, pp. 264-276, 2001.

[9] X. Lin and S. N. Balakrishnan, "Convergence analysis of adaptive ciritc based optimal control," Proceedings of the American Control Conference, vol. 12, 264-276 2000.

[10] F. L. Lewis, S. Jagannathan, and A. Yesilderek, Neural Network control of robot manipulators and nonlinear systems. UK: Taylor and Francis, 1999.

[11] N. Hovakimyan, F. Nardi, A. Calise, and N. Kim, "Adaptive output feedback control of uncertain nonlinear systems using single-hidden-layer neural networks," IEEE Trans. on Neural Networks, vol. 13, pp. 1420-1431, 2002.

[12] A. N. Atassi and H. K. Khalil, "A separation principle for the stabilization of a class of nonlinear systems," IEEE Trans. Automat. Contr, vol. 76, pp. 334-354, 2003.

[13] B. Igelruk and Y. H. Pao, "Stochastic choice of basis functions in adaptive function approximation and the functional-link net," IEEE Trans. Neural Networks, vol. 6, pp. 1320-1329, 1995.

[14] S. Jagannathan, Neural Network Control of Nonlinear Discrete-time Systems. London, UK: Taylor and Francis, 2006.

[15] C. S. Daw, C. E. A. Finney, M. B. Kennel, F. T. Connolly, "Observing and Modeling Nonlinear Dynamics in an Internal Combustion Engine," Phys. Rev. E, vol. 57, pp. 2811-2819, 1998.

[16] J. Vance, P. He, S. Jagannathan, and J. Drallmeier, "Neural Network-based Output Feedback Controller for Lean Operation of Spark Ignition Engine," in American Control Conference, Portland, OR, 2006.

PAPER 2

Reinforcement Learning Based Feedback Controller for Complex Nonlinear Discrete-time Systems with Application to Spark Engine EGR Operation

Peter Shih, Bryan Kaul, Sarangapani Jagannathan, *Sr. Member, IEEE*, and James A. Drallmeier

***Abstract*— A novel reinforcement-learning based output-adaptive neural network (NN) controller, also referred to as the adaptive-critic NN controller, is developed to deliver a desired tracking performance for a class of complex feedback nonlinear discrete-time systems in the presence of bounded and unknown disturbances. This complex nonlinear discrete-time system consists of a second order nonlinear discrete-time system in non-strict form and an affine nonlinear discrete-time system tightly coupled together. Two adaptive critic NN controllers are designed—the primary one for the non-strict feedback nonlinear discrete-time system and the secondary one for the affine nonlinear discrete-time system.**

The primary adaptive critic NN controller includes a NN observer, NN critic, and two action NNs for generating virtual control and actual control inputs for the nonstrict feedback nonlinear discrete-time system, whereas a critic NN and an action NN are included for the affine nonlinear discrete-time system. The NN observer estimates the states and output of the nonlinear discrete-time system in non-strict feedback form. The critic approximates a certain *strategic* utility function and the action NNs are used to minimize both the *strategic* utility function and action NN outputs. All NN weights adapt online towards minimization of a certain performance index, utilizing gradient-descent based rule. Using Lyapunov functions, the uniformly ultimate boundedness (UUB) of the closed-loop tracking error, weight estimates and observer estimation are shown. Separation principle and certainty equivalence principles are relaxed, persistency of excitation condition is not required, and the linear in the unknown parameter assumption is not needed.

The performance of this adaptive critic NN controller is evaluated on a spark ignition (SI) engine operating with high exhaust gas recirculation (EGR) levels where the controller objectives is to reduce cyclic dispersion in heat release. The secondary objectives are to reduce emissions. Experimental results at 20% EGR show a 34% reduction in cyclic dispersion in heat release with control while the average fuel input changes by less than 1% compared to the uncontrolled case. Additionally, the unburned hydrocarbons (uHC) drop nominally with control, and by 80% compared to levels at zero EGR. Overall, NO_x is reduced by 80% compared to levels at zero EGR.

I. INTRODUCTION

Adaptive neural network (NN) backstepping control of nonlinear discrete-time systems in strict feedback form has been addressed in the literature [1-3]. The strict feedback nonlinear system is normally expressed as

$$x_i(k+1) = f_i(\bar{x}_i(k)) + g_i(\bar{x}_i(k))x_{i+1}(k) \tag{1}$$

$$x_n(k+1) = f_n(\bar{x}_n(k)) + g_n(\bar{x}_n(k))u(k) \tag{2}$$

where $x_i(k) \in \Re$ is the state, $u(k) \in \Re$ is the control input, $\bar{x}_i(k) = [x_1(k), \cdots, x_i(k)]^T \in \Re^i$ and $i = 1, ..., (n-1)$. For strict feedback nonlinear systems [1], the nonlinearities $f_i(\bar{x}_i(k))$ and $g_i(\bar{x}_i(k))$ depend only upon states $x_1(k), \ldots, x_i(k)$, i.e., $\bar{x}_i(k)$. However, for a non-strict feedback nonlinear system, where $f_i(\bar{x}_i(k))$ and $g_i(\bar{x}_i(k))$ depend on both $\bar{x}_i(k)$ and $x_{i+1}(k)$, there are no control design schemes currently available. Available [1-3] methods applied to the nonlinear discrete-time systems will result in a non-causal controller (current control input depends on the future system states) even for second order systems using the adaptive NN backstepping approach. Finally, no optimization is carried out in these control designs, as simple tracking error is utilized.

In short, available NN controller designs employ either supervised training, where the user specifies a desired output, or online NN training based on classical adaptive control [1-3], where a short-term system performance measure is defined by using the tracking error. By contrast, the reinforcement-learning based adaptive critic NN approach [4] has emerged as a promising tool to develop optimal NN controllers due to its potential to find

approximate solutions to dynamic programming, where a *strategic* utility function, which is considered as the long-term system performance measure, can be optimized. In supervised learning, an explicit signal is provided by the teacher to guide the learning process whereas in the case of reinforcement learning, the role of the teacher is more evaluative than instructional in nature. The critic NN monitors the system states and approximates the *strategic* utility function, with a potential for a look-ahead and better training of the action NN which generates the near optimal control action to the system.

There are many variants of adaptive critic NN controller architectures [4-9] using state feedback even though few results [6-9] address the controller convergence. However, NN controller results are not available for the nonlinear discrete-time systems in non-strict feedback form. Similarly, no known results are available using adaptive critic NN control-based affine nonlinear discrete-time systems.

In this paper, a novel adaptive critic NN-based *output* feedback controller is developed to control a class of nonlinear discrete-time systems in non-strict feedback form with bounded and unknown disturbances. Since the complex nonlinear discrete-time system under consideration involves both non-strict feedback form and affine nonlinear discrete-time system, two controllers are designed one for non-strict feedback form and the other for the affine nonlinear discrete-time systems so that they can operate simultaneously.

For the case of nonlinear discrete-time system in non-strict feedback form, an adaptive NN backstepping is utilized for the controller design with two action NNs being used to generate the virtual and actual control inputs, respectively. The weights of the two action NNs are tuned by the critic NN signal to minimize the *strategic* utility function and their outputs. The critic NN approximates certain *strategic* utility function which is a variant of standard Bellman equation. The NN observer generates the estimates of the system states and output, which are subsequently used in the controller design. The proposed controller is *model–free* since the dynamics of the nonlinear discrete-time systems are unknown and NN weights are tuned online. On the other hand for the affine nonlinear discrete-time system, a separate critic NN and an action NN are utilized. The critic NN approximates the standard Bellmann equation and tunes the action NN so that the action NN generates a near optimal signal to control the affine nonlinear discrete-time system.

The main contributions of this paper can be summarized as follows: 1) the adaptive NN backstepping scheme is extended to non-strict feedback nonlinear systems. The non-causal problem is overcome by employing the universal NN approximation property; 2) optimization of a long-term performance index is undertaken in contrast with traditional adaptive NN back stepping schemes [1, 2] where no optimization is performed; 3) demonstration of the UUB of the overall system is shown even in the presence of NN approximation errors and bounded unknown disturbances unlike in the existing adaptive critic works [7-9] where the convergence is presented under ideal circumstances. Stability proof is inferred even with a NN observer by relaxing the separation principle via novel weight updating rules and by selecting the Lyapunov function consisting of the system estimation errors, tracking and the NN weight estimation errors; A single critic NN is utilized to tune two action NNs; 4) a well-defined controller is presented by overcoming the problem of certain nonlinear function estimates becoming zero since a single NN is used to approximate both the nonlinear functions $f_i(\bar{x}_i(k))$ and $g_i(\bar{x}_i(k))$ compared to [10]; 5) the NN weights are tuned online instead of offline [5]; and finally 6) the assumption that $g_1(x_1(k), x_2(k))$ is bounded away from zero and its sign is known *a priori* is relaxed in contrast with [2].

The proposed primary controller is applied to control the spark ignition (SI) engine dynamics, a practical non-strict feedback nonlinear system. The controller permits the engine to operate in high EGR mode, where an inert gas displaces the stoichiometric ratio of fuel to air. The inert gas is simulated by nitrogen in the lab, whereas exhaust gas is used after implementation. The inert gas system is modeled as an affine nonlinear discrete-time system and therefore a separate secondary controller is designed. Both controllers operate simultaneously due to the tight coupling of the systems. The controllers enable the engine to operate in higher EGR mode compared to the uncontrolled case by reducing heat release bifurcation. Consequently, the engine exhibits improved emissions and fuel efficiency compared to the uncontrolled case. Other controller designs can run a SI engine in lean mode [11]; however, engine catalysts cannot function efficiently with the lean exhaust chemistry.

EGR, on the other hand, allows for the efficient operation of standard three-way catalysts. Not only does it reduce pre-catalyst emissions, but it can improve fuel efficiency by reducing throttling losses. Therefore, the applicability of high EGR usage in the automotive engines is greater. In this paper, the secondary controller maintains a set inert gas level by allowing an appropriate amount of EGR whereas the objective of the primary controller is to minimize cyclic dispersion in heat release while optimizing the fuel intake. Dilution with EGR also has wide practical applicability in diesel engines and in spark ignition engines without three-way catalysts.

II. Non-Linear Non-Strict Feedback System

Consider the nonlinear discrete-time system, given in the following form

$$x_1(k+1) = f_1\left(x_1(k), x_2(k), x_3(k)\right) + g_1\left(x_1(k), x_2(k), x_3(k)\right)x_2(k) + d_1(k) \tag{3}$$

$$x_2(k+1) = f_2\left(x_1(k), x_2(k), x_3(k)\right) + g_2\left(x_1(k), x_2(k), x_3(k)\right)u(k) + d_2(k) \tag{4}$$

$$x_3(k+1) = f_4\left(x_1(k), x_2(k), x_3(k)\right) + g_4\left(x_1(k), x_2(k), x_3(k)\right)v(k) + d_3(k) \tag{5}$$

$$y(k+1) = f_3\left(x_1(k), x_2(k), x_3(k)\right) \tag{6}$$

where $x_i(k) \in \Re; i = 1,2,3$ are states, $u(k) \in \Re$ and $v(k) \in \Re$ are system inputs, and $d_1(k) \in \Re$, $d_2(k) \in \Re$ and $d_3(k) \in \Re$ are unknown but bounded disturbances. Bounds on the disturbances are given by $|d_1(k)| < d_{1m}$, $|d_2(k)| < d_{2m}$, and $|d_3(k)| < d_{3m}$ where d_{1m}, d_{2m}, and d_{2m} are unknown positive scalars. The output is a nonlinear function of states in contrast with available literature [12, 13] where the output is a linear function of the states. Finally, the output is measurable whereas the first two states $x_1(k)$ and $x_2(k)$ are considered not available while $x_3(k)$ is assumed to be available. For the system (3) and (4), not only should the system actual output converge to its target value, but the states should also converge to their respective desired values.

The controller development is presented separately for the two systems as the objectives are separate even though they are tightly coupled. The first part uses equations (3), (4), and (6) to develop the primary controller. The second part uses equation (5) to develop the secondary controller. Stability for both the systems is demonstrated.

III. PRIMARY CONTROLLER – OBSERVER DESIGN

To overcome the immeasurable states $x_1(k)$ and $x_2(k)$, an observer is used. It utilizes the current heat release output, $y(k)$, to estimate the future output $\hat{y}(k+1)$ and states $\hat{x}_1(k+1)$ and $\hat{x}_2(k+1)$. The design of the observer is discussed next.

A. Observer Design

Consider equations (3) and (4). We expand the individual nonlinear functions using Taylor series expansion into linear and higher order terms.

$$f_1(\cdot) = f_{10} + \Delta f_1(\cdot) \tag{7}$$

$$f_2(\cdot) = f_{20} + \Delta f_2(\cdot) \tag{8}$$

$$g_1(\cdot) = g_{10} + \Delta g_1(\cdot) \tag{9}$$

$$g_2(\cdot) = g_{20} + \Delta g_2(\cdot) \tag{10}$$

where the first term in (7) through (10) are known nominal values and the second term are unknown higher order terms. We use a two-layer feed-forward NN with semi-recurrent architecture and novel weight tuning to construct the output as

$$y(k+1) = w_1^T \phi\left(v_1^T z_1(k)\right) + \varepsilon\left(z_1(k)\right), \tag{11}$$

where $z_1(k) = \left[x_1(k), x_2(k), x_3(k), y(k), u(k)\right]^T \in R^4$ is the network input, $y(k+1)$ and $y(k)$ are the future and current outputs, $w_1 \in \Re^{n_1}$ and $v_1 \in \Re^{2\times n_1}$ denote the ideal output and constant hidden layer weight matrices, respectively, $u(k)$ is the control input, $\phi\left(v_1^T z_1(k)\right)$ represents the hidden layer activation function, n_1 is the number of nodes in the hidden layer, and $\varepsilon\left(z_1(k)\right) \in \Re$ is the approximation error. For simplicity the two equations can be represented as

$$\phi_1(k) = \phi\left(v_1^T z_1(k)\right) \tag{12}$$

$$\varepsilon_1(k) = \varepsilon\left(z_1(k)\right) \tag{13}$$

Rewrite (11) using (12) and (13) to obtain

$$y(k+1) = w_1^T \phi_1(k) + \varepsilon_1(k) \tag{14}$$

The states $x_1(k)$ and $x_2(k)$ are not measurable; therefore $z_1(k)$ is not available either. Using the estimated states and the output, $\hat{x}_1(k), \hat{x}_2(k)$, and $\hat{y}(k)$, respectively, instead of $x_1(k)$, $x_2(k)$, and $y(k)$, the proposed observer is given as

$$\begin{aligned}\hat{y}(k+1) &= \hat{w}_1^T(k)\phi\left(v_1^T\hat{z}_1(k)\right)+l_1\tilde{y}(k)\\ &= \hat{w}_1^T(k)\hat{\phi}_1(k)+l_1\tilde{y}(k)\end{aligned} \tag{15}$$

where $\hat{z}_1(k)=\left[\hat{x}_1(k),\hat{x}_2(k),x_3(k),\hat{y}(k),u(k)\right]^T \in R^5$ is the input vector using estimated states, $\hat{y}(k+1)$ and $\hat{y}(k)$ are the estimated future and current output, $\hat{w}_1(k)$ is the actual weight matrix, $u(k)$ is the estimated control input, $\hat{\phi}_1(k)$ is the hidden layer activation function, $l_1 \in R$ is the observer gain, and $\tilde{y}(k)$ is the heat release estimation error defined as

$$\tilde{y}(k)=\hat{y}(k)-y(k) \tag{16}$$

It is demonstrated in [14] that, if the hidden layer weights, v_1, are chosen initially at random and kept constant, and the number of hidden layer nodes is sufficiently large, then the approximation error $\varepsilon\left(z_1(k)\right)$ can be made arbitrarily small so that the bound $\left\|\varepsilon\left(z_1(k)\right)\right\| \le \varepsilon_{1m}$ holds for all $z_1(k) \in S$ since the activation function forms a basis to the nonlinear function that the NN approximates. Now we choose, at our convenience, the observer structure as a function of output estimation errors and known quantities as

$$\hat{x}_1(k+1)=f_{10}-\hat{x}_2(k)+l_2\tilde{y}(k) \tag{17}$$

$$\hat{x}_2(k+1)=f_{20}+g_{20}u(k)+l_3\tilde{y}(k) \tag{18}$$

where $l_2 \in R$ and $l_3 \in R$ are design constants.

B. Observer Error Dynamics

Define the state estimation and output errors as

$$\tilde{x}_i(k+1)=\hat{x}_i(k+1)-x_i(k+1), i \in \{1,2\} \tag{19}$$

$$\tilde{y}(k+1)=\hat{y}(k+1)-y(k+1) \tag{20}$$

Combine (3) through (11) and, (17) through (20), to obtain the estimation and output error dynamics as

$$\tilde{x}_1(k+1) = f_{10} - \hat{x}_2(k) + l_2\tilde{y}(k) - f_1(\cdot) - g_1(\cdot)x_2(k) - d_1(k) \tag{21}$$

$$\tilde{x}_2(k+1) = f_{20} + g_{20}u(k) + l_3\tilde{y}(k) - f_2(\cdot) - g_2(\cdot)u(k) - d_2(k) \tag{22}$$

and

$$\tilde{y}(k+1) = \hat{w}_1^T(k)\hat{\phi}_1(k) + l_1\tilde{y}(k) - w_1^T\phi_1(k) - \varepsilon_1(k) \tag{23}$$

Choose the weight tuning of the observer as

$$\hat{w}_1(k+1) = \hat{w}_1(k) - \alpha_1\hat{\phi}_1(k)\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right) \tag{24}$$

where $\alpha_1 \in R$ and $l_4 \in R$ are design constants. It will be shown in the next section that by using the above weight tuning, the separation principle is relaxed and the closed-loop signals will be bounded. Next, we present the following theorem, where it is demonstrated that the state estimation and output estimation errors along with observer NN weight estimation errors are bounded. The following mild assumptions are required.

Assumption 1: The unknown smooth functions, $f_2(\cdot)$ and $g_2(\cdot)$, and control $u(k)$, are upper bounded within the compact set S as $f_{2\max} > |f_2(k)|$, $g_{2\max} > |g_2(k)|$, and $u_{\max} > |u(k)|$.

Theorem 1: Consider the system given by (3), (4) and (6), and the disturbance bounded by $|d_1(k)| < d_{1m}$ and $|d_2(k)| < d_{2m}$ where d_{1m} and d_{2m} are known positive scalars. Let the observer NN weight tuning be given by (24). Given bounded inputs, the state estimation errors $\tilde{x}_1(k)$ and $\tilde{x}_2(k)$, output estimation errors $\tilde{y}(k)$, and NN weight estimate $\hat{w}_1(k)$ are UUB, with the bounds specifically given by (B.17), with the controller design parameters selected as

$$0 < \alpha_1\|\phi_1(k)\|^2 < 1 \tag{25}$$

$$|l_1| < \frac{1}{2} \tag{26}$$

$$|l_2| < \frac{\sqrt{3}}{3} \tag{27}$$

$$|l_3| < \frac{\sqrt{3}}{3} \tag{28}$$

$$|l_4| < \frac{\sqrt{3}}{3} \tag{29}$$

where α_1 is NN adaptation gain, l_1, l_2, l_3, and l_4 are observer parameters.

Proof: Define the Lyapunov function

$$J(k) = \sum_{i=1}^{4} J_i(k) = \frac{\gamma_1}{\alpha_1} \tilde{w}_1^T(k)\tilde{w}_1(k) + \frac{\gamma_2}{3}\tilde{x}_1^2(k) + \frac{\gamma_3}{2}\tilde{x}_2^2(k) + \frac{\gamma_4}{3}\tilde{y}^2 \tag{30}$$

where $0 < \gamma_i, i \in \{1,2,3,4\}$ are auxiliary constants. Take the first term, take the first difference, and substitute (24)

$$\begin{aligned}
J_1(k) &= \tfrac{\gamma_1}{\alpha_1}\tilde{w}_1^T(k)\tilde{w}_1(k) \\
\tfrac{\alpha_1}{\gamma_1}\Delta J_1(k) &= \tilde{w}_1^T(k+1)\tilde{w}_1(k+1) - \tilde{w}_1^T(k)\tilde{w}_1(k) \\
&= [\tilde{w}_1^T(k) - \alpha_1\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)^T\hat{\phi}_1^T(k)]* \\
&\quad [\tilde{w}_1(k) - \alpha_1\hat{\phi}_1(k)\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)] - \tilde{w}_1^T(k)\tilde{w}_1(k) \\
&= \alpha_1^2\left\|\phi_1(k)\right\|^2\begin{pmatrix}\hat{w}_1^T(k)\hat{\phi}_1(k)\\ +l_4\tilde{y}(k)\end{pmatrix}^2 - 2\alpha_1\tilde{w}_1(k)\hat{\phi}_1(k)\begin{pmatrix}\hat{w}_1^T(k)\hat{\phi}_1(k)\\ +l_4\tilde{y}(k)\end{pmatrix} + \\
&\quad \alpha_1\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)^2 - \alpha_1\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)^2 \\
&= -\alpha_1\left(1 - \alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)^2 + \\
&\quad \alpha_1\left(\left(\zeta_1(k) + w_1^T\hat{\phi}_1 + l_4\tilde{y}(k)\right) - \zeta_1(k)\right)^2 - \alpha_1\zeta_1^2(k) \\
&= -\alpha_1\left(1 - \alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\begin{pmatrix}\hat{w}_1^T(k)\hat{\phi}_1(k)\\ +l_4\tilde{y}(k)\end{pmatrix}^2 + \alpha_1\left(w_1^T\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)^2 - \alpha_1\zeta_1^2(k)
\end{aligned} \tag{31}$$

Invoke the Cauchy-Schwarz inequality, defined as

$$\left(a_1b_1 + \ldots + a_nb_n\right)^2 \le \left(a_1^2 + \ldots + a_n^2\right)\left(b_1^2 + \ldots + b_n^2\right) \tag{32}$$

and simplify to get

$$\begin{aligned}
\Delta J_1(k) \le &-\gamma_1\left(1 - \alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\left(\hat{w}_1(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)^2 + \\
&2\gamma_1\left(w_{1\mathrm{m}}\hat{\phi}_{1\mathrm{m}}\right)^2 + 2\gamma_1 l_4^2\tilde{y}^2(k) - \gamma_1\zeta_1^2(k)
\end{aligned} \tag{33}$$

Take the second term and substitute (21)

$$\Delta J_2(k) \le \gamma_2 l_2^2\tilde{y}^2(k) + \gamma_2\tilde{x}_2^2(k) + \gamma_2\left(w_{3\mathrm{m}}\phi_{3\mathrm{m}} + f_{10} + \varepsilon_{3\mathrm{m}} + d_{1\mathrm{m}}\right)^2 - \tfrac{\gamma_2}{3}\tilde{x}_1^2(k) \tag{34}$$

Take the third term and substitute (22)

$$\Delta J_3(k) \le \gamma_3 \left(f_{20} + \left(g_{20} + g_{2\max} \right) u_{\max} + f_{2\max} + d_{2\mathrm{m}} \right)^2 + \gamma_3 l_3^2 \tilde{y}^2(k) - \tfrac{\gamma_3}{2} \tilde{x}_2^2(k) \tag{35}$$

Take the fourth and final term and substitute (23)

$$\Delta J_4(k) \le \gamma_4 \zeta_1^2(k) + \gamma_4 l_1^2 \tilde{y}(k) + \gamma_4 \left(w_{1\mathrm{m}} \tilde{\phi}_{1\mathrm{m}} + \varepsilon_{1\mathrm{m}} \right) - \tfrac{\gamma_4}{3} \tilde{y}^2(k) \tag{36}$$

Combine equations (33) through (36) and simplify to get the first difference of the Lyapunov function

$$\begin{aligned} \Delta J(k) \le & -\gamma_1 \left(1 - \alpha_1 \left\| \hat{\phi}_1(k) \right\|^2 \right) \left(\hat{w}_1(k) \hat{\phi}_1(k) + l_4 \tilde{y}(k) \right)^2 - \left(\tfrac{\gamma_3}{2} - \gamma_2 \right) \tilde{x}_2^2(k) - \tfrac{\gamma_2}{3} \tilde{x}_1^2(k) \\ & - \left(\tfrac{\gamma_4}{3} - 2\gamma_1 l_4^2 - \gamma_2 l_2^2 - \gamma_3 l_3^2 - \gamma_4 l_1^2 \right) \tilde{y}^2(k) - \left(\gamma_1 - \gamma_4 \right) \zeta_1^2(k) + D_M^2 \end{aligned} \tag{37}$$

where D_M^2 is defined as

$$\begin{aligned} D_M^2 = & 2\gamma_1 \left(w_{1\mathrm{m}} \hat{\phi}_{1\mathrm{m}} \right)^2 + \gamma_2 \left(w_{3\mathrm{m}} \phi_{3\mathrm{m}} + f_{10} + \varepsilon_{3\mathrm{m}} + d_{1\mathrm{m}} \right)^2 + \\ & \gamma_3 \left(f_{20} + \left(g_{20} + g_{2\max} \right) u_{\max} + f_{2\max} + d_{2\mathrm{m}} \right)^2 + \gamma_4 \left(w_{1\mathrm{m}} \tilde{\phi}_{1\mathrm{m}} + \varepsilon_{1\mathrm{m}} \right) \end{aligned} \tag{38}$$

Select

$$\gamma_3 > 2\gamma_2;\ \gamma_4 > 6\gamma_1 l_4^2 + 3\gamma_2 l_2^2 + 3\gamma_3 l_3^2 + 3\gamma_4 l_1^2;\ \gamma_1 > \gamma_4 \tag{39}$$

This implies $\Delta J(k) < 0$ as long as (25) through (29) hold *and* any *one* the following hold

$$\begin{aligned} & \left| \tilde{x}_1(k) \right| > \frac{D_M}{\sqrt{\frac{\gamma_2}{3}}}; \\ & \left| \tilde{x}_2(k) \right| > \frac{D_M}{\sqrt{\frac{\gamma_3}{2} - \gamma_2}}; \\ & \left| \tilde{y}(k) \right| > \frac{D_M}{\sqrt{\frac{\gamma_4}{3} - 2\gamma_1 l_4^2 - \gamma_2 l_2^2 - \gamma_3 l_3^2 - \gamma_4 l_1^2}}; \\ & \left| \zeta_1(k) \right| > \frac{D_M}{\sqrt{\gamma_1 - \gamma_4}} \end{aligned} \tag{40}$$

According to a standard Lyapunov extension theorem [15], this demonstrates that the estimation errors, output error and the NN observer weight estimation errors are *UUB*. Remark: In this above theorem, the control input is considered bounded, which is an acceptable assumption (also made in all output feedback control literature) that is relaxed in the next few sections when combined with the controller design, wherein the closed-

loop system is shown to be bounded. In this above theorem, the control input is considered bounded, which is relaxed in the next few sections. On the other hand, the assumption that the unknown nonlinearities are bounded is valid, since for many practical systems, the upper bound on the unknown nonlinearities will be known [15]. Additionally, for NN based control it is also necessary that the nonlinear functions be on a compact set in order for the NN to approximate them.

Next, we discuss the design of the adaptive critic NN controller for the primary system and demonstrate that the closed-loop system, NN observer signals, and control inputs will be bounded.

IV. Primary Controller – Critic Design

The purpose of the critic NN is to approximate the long-term performance index (or strategic utility function) of the nonlinear system through online weight adaptation. The critic signal estimates the future performance and tunes the two action NNs. The tuning will ultimately minimize the strategic utility function itself and the action NN outputs or control inputs so that closed-loop stability is inferred.

A. The Strategic Utility Function

The utility function $p(k) \in \Re$ is given by

$$p(k) = \begin{cases} 0, & if \left(\left| \tilde{y}(k) \right| \right) \le c \\ 1, & otherwise \end{cases} \tag{41}$$

where $c \in \Re$ is a user-defined threshold. The utility function $p(k)$ represents the current performance index. In other words, $p(k) = 0$ and $p(k) = 1$ refer to good and unsatisfactory tracking performance at the k^{th} time step, respectively. The long-term *strategic* utility function $Q(k) \in \Re$, is defined as

$$Q(k) = \beta^N p(k+1) + \beta^{N-1} p(k+2) + \cdots + \beta^{k+1} p(N) + .., \tag{42}$$

where $\beta \in \Re$ and $0 < \beta < 1$ is the discount factor, and N is the horizon index. The term $Q(k)$ is viewed here as the long system performance measure for the controller since it is the sum of all future system performance indices. Equation (42) can also be expressed as $Q(k) = \min_{u(k)} \left\{ \alpha Q(k-1) - \alpha^{N+1} p(k) \right\}$, which is similar to the standard Bellman equation.

B. Design of the Critic NN

We utilize the universal approximation property of NN and rewrite $\hat{Q}(k)$ as

$$\hat{Q}(k) = \hat{w}_2^T(k)\phi\left(v_2^T \hat{z}_2(k)\right) = \hat{w}_2^T(k)\hat{\phi}_2(k) \tag{43}$$

where $\hat{Q}(k) \in \Re$ is the critic signal, $\hat{w}_2(k) \in \Re^{n_2}$ is the tunable weight, $v_2 \in \Re^{3\times n_2}$ represents the constant input weight matrix selected initially at random, $\hat{\phi}_2(k) \in \Re^{n_2}$ is the activation function vector in the hidden layer, n_2 is the number of the nodes in the hidden layer, and $\hat{z}_2(k) = [\hat{x}_1(k), \hat{x}_2(k), x_3(k)]^T \in R^3$ is the input vector.

C. Critic Weight Update Law

Define the prediction error as

$$e_c(k) = \hat{Q}(k) - \beta\left(\hat{Q}(k-1) - \beta^N p(k)\right) \tag{44}$$

where the subscript "c" stands for the "critic." Define a quadratic objective function to minimize

$$E_c(k) = \frac{1}{2}e_c^2(k) \tag{45}$$

The weight update rule for the critic NN is based upon gradient adaptation, which is given by the general formula

$$\hat{w}_2(k+1) = \hat{w}_2(k) + \Delta\hat{w}_2(k) \tag{46}$$

$$\Delta\hat{w}_2(k) = \alpha_2\left[-\frac{\partial E_c(k)}{\partial \hat{w}_2(k)}\right] \tag{47}$$

or

$$\hat{w}_2(k+1) = \hat{w}_2(k) - \alpha_2\hat{\phi}_2(k)\left(\hat{Q}(k) + \beta^{N+1}p(k) - \beta\hat{Q}(k-1)\right)^T \tag{48}$$

where $\alpha_2 \in \Re$ is the NN adaptation gain.

V. PRIMARY CONTROLLER – VIRTUAL CONTROL INPUT DESIGN

In this section, the design of the virtual control input is discussed. Before we proceed, the following mild assumption is needed. Then the systems of nonlinear equations are rewritten.

Assumption 2: The unknown smooth function $g_2(\cdot)$ is bounded away from zero for all $x_1(k)$ and $x_2(k)$ within the compact set S. In other words, $0 < g_{2\min} < |g_2(\cdot)| < g_{2\max}, \forall x_1(k) \& x_2(k) \in S$, where $g_{2\min} \in \Re^+$ and $g_{2\max} \in \Re^+$. Without loss of generality, we will assume that $g_2(\cdot)$ is positive in this paper.

A. System Simplification

First, we simplify by rewriting the state equations with the following

$$\Phi(\cdot) = f_1\left(x_1(k), x_2(k), x_3(k)\right) + g_1\left(x_1(k), x_2(k), x_3(k)\right) x_2(k) + x_2(k) \tag{49}$$

The system (3) and (4) can be rewritten as

$$x_1(k+1) = \Phi(\cdot) - x_2(k) + d_1(k) \tag{50}$$

$$x_2(k+1) = f_2(\cdot) + g_2(\cdot) u(k) + d_2(k) \tag{51}$$

B. Virtual Control Input Design

Our goal is to stabilize the system output, $y(k)$, around a specified target point, y_d, by controlling the input. The secondary objective is to make $x_1(k)$ approach the desired trajectory $x_{1d}(k)$. At the same time, all signals in systems (3) and (4) must be UUB, all weights must be bounded, and a performance index must be minimized. Define the tracking error as

$$e_1(k) = x_1(k) - x_{1d}(k) \tag{52}$$

where $x_{1d}(k)$ is the desired trajectory. Using (50), (52) can be expressed as the following

$$\begin{aligned} e_1(k+1) &= x_1(k+1) - x_{1d}(k+1) \\ &= \left(\Phi(\cdot) - x_2(k) + d_1(k)\right) - x_{1d}(k+1) \end{aligned} \tag{53}$$

By viewing $x_2(k)$ as a virtual control input, a desired virtual control signal can be designed as

$$x_{2d}(k) = \Phi(\cdot) - x_{1d}(k+1) + l_5 \hat{e}_1(k) \tag{54}$$

where l_5 is a gain constant. Since $\Phi(\cdot)$ is an unknown function, $x_{2d}(k)$ in (54) cannot be implemented in practice. We invoke the universal approximation property of NN to estimate this unknown function.

$$\Phi(\cdot) = w_3^T \phi\left(v_3^T z_3(k)\right) + \varepsilon\left(z_3(k)\right) \tag{55}$$

where $z_3(k) = \left[x_1(k), x_2(k), x_3(k)\right]^T \in \Re^3$ is the input vector, $w_3^T \in \Re^{n_2}$ and $v_3^T \in \Re^{3\times n_3}$ are the ideal and constant input weight matrices, $\phi\left(v_3^T z_3(k)\right) \in \Re^{n_3}$ is the activation function vector in the hidden layer, n_3 is the number of the nodes in the hidden layer, and $\varepsilon\left(z_3(k)\right)$ is the functional estimation error. It is demonstrated in [14] that, if the hidden layer weights, v_1, are chosen initially at random and kept constant, and the number of hidden layer nodes is sufficiently large, then the approximation error $\varepsilon\left(z_3(k)\right)$ can be made arbitrarily small so that the bound $\left\|\varepsilon\left(z_3(k)\right)\right\| \le \varepsilon_{3m}$ holds for all $z_3(k) \in S$ in a compact set, since the activation function vector forms a basis to the nonlinear function that the NN approximates.

Rewriting (54) using (55), the virtual control signal can be rewritten as

$$x_{2d}(k) = w_3^T \phi\left(v_3^T z_3(k)\right) + \varepsilon\left(z_3(k)\right) - x_{1d}(k+1) + l_5 \hat{e}_1(k) \tag{56}$$

Replacing actual with estimated states, (56) becomes

$$\begin{aligned} \hat{x}_{2d}(k) &= \hat{w}_3^T(k)\phi\left(v_3^T \hat{z}_3(k)\right) - x_{1d}(k+1) + l_5 \hat{e}_1(k) \\ &= \hat{w}_3^T(k)\hat{\phi}_3(k) - x_{1d}(k+1) + l_5 \hat{e}_1(k) \end{aligned} \tag{57}$$

where $\hat{z}_3(k) = \left[\hat{x}_1(k), \hat{x}_2(k), x_3(k)\right]^T \in \Re^3$ is the input vector using estimated states, and $\hat{e}_1(k) = \hat{x}_1(k) - x_{1d}(k)$.

Define

$$e_2(k) = x_2(k) - \hat{x}_{2d}(k) \tag{58}$$

Equation (53) can be rewritten using (58) as

$$\begin{aligned} e_1(k+1) &= \left(\Phi(\cdot) - x_2(k) + d_1(k)\right) - x_{1d}(k+1) \\ &= \Phi(\cdot) - \left(e_2(k) + \hat{x}_{2d}(k)\right) + d_1(k) - x_{1d}(k+1) \\ &= \Phi(\cdot) - \hat{x}_{2d}(k) - e_2(k) - x_{1d}(k+1) + d_1(k) \end{aligned} \tag{59}$$

Combine (57) into (59), then (55) into the combined equation

$$\begin{aligned} e_1(k+1) &= \Phi(\cdot) - \left(\hat{w}_3^T(k)\hat{\phi}_3(k) - x_{1d}(k+1) + l_5\hat{e}_1(k)\right) - e_2(k) - x_{1d}(k+1) + d_1(k) \\ &= \left(w_3^T\phi_3(k) + \varepsilon_3(k)\right) - \hat{w}_3^T(k)\hat{\phi}_3(k) - l_5\hat{e}_1(k) - e_2(k) + d_1(k) \\ &= w_3^T\left(\hat{\phi}_3(k) - \tilde{\phi}_3(k)\right) - \hat{w}_3^T(k)\hat{\phi}_3(k) + \varepsilon_3(k) - l_5\hat{e}_1(k) - e_2(k) + d_1(k) \\ &= w_3^T\left(\hat{\phi}_3(k) - \tilde{\phi}_3(k)\right) - \hat{w}_3^T(k)\hat{\phi}_3(k) + \varepsilon_3(k) - l_5\hat{e}_1(k) - e_2(k) + d_1(k) \\ &= -\tilde{w}_3^T\hat{\phi}_3(k) - w_3^T\tilde{\phi}_3(k) + \varepsilon_3(k) - l_5\hat{e}_1(k) - e_2(k) + d_1(k) \\ &= -\zeta_3(k) - w_3^T\tilde{\phi}_3(k) + \varepsilon_3(k) - l_5\hat{e}_1(k) - e_2(k) + d_1(k) \end{aligned} \tag{60}$$

where

$$\zeta_3(k) = \tilde{w}_3^T(k)\hat{\phi}_3(k) = \hat{w}_3^T(k)\hat{\phi}_3(k) - w_3^T\hat{\phi}_3(k) \tag{61}$$

$$\tilde{\phi}_3(k) = \phi\left(v_3\hat{z}_3(k)\right) - \phi\left(v_3 z_3(k)\right) \tag{62}$$

C. Virtual Control Weight Update

Let us define

$$e_{a1}(k) = \hat{w}_3^T(k)\hat{\phi}_3(k) + \left(\hat{Q}(k) - Q_d(k)\right) \tag{63}$$

where $\hat{Q}(k)$ is defined in (43), and the a1 subscript represents the error for the first action NN, $e_{a1}(k) \in \Re$. The desired *strategic* utility function $Q_d(k)$ is "0" to indicate perfect tracking at all steps. Thus, (63) becomes

$$e_{a1}(k) = \hat{w}_3^T(k)\hat{\phi}_3(k) + \hat{Q}(k) \tag{64}$$

The objective function to be minimized by the first action NN is given by

$$E_{a1}(k) = \frac{1}{2}e_{a1}^2(k) \tag{65}$$

The weight update rule for the action NN is also a gradient-based adaptation defined as

$$\hat{w}_3(k+1) = \hat{w}_3(k) + \Delta\hat{w}_3(k) \tag{66}$$

where

$$\Delta\hat{w}_3(k) = \alpha_3\left[-\frac{\partial E_{a1}(k)}{\partial\hat{w}_3(k)}\right] \tag{67}$$

$$\hat{w}_3(k+1) = \hat{w}_3(k) - \alpha_3\hat{\phi}_3(k)\left(\hat{Q}(k) + \hat{w}_3^T(k)\hat{\phi}_3(k)\right) \tag{68}$$

with $\alpha_3 \in \Re$ is the NN adaptation gain.

VI. PRIMARY CONTROLLER – CONTROL INPUT DESIGN

Choose the following desired control input

$$u_d(k) = \frac{1}{g_2(k)}\left(-f_2(k) + \hat{x}_{2d}(k+1) + l_6 e_2(k)\right), \tag{69}$$

Note that $u_d(k)$ is non-causal since it depends upon future value of $\hat{x}_{2d}(k+1)$. We solve this problem by using a semi-recurrent NN since it can be a one step predictor. The term $\hat{x}_{2d}(k+1)$ depends on state $x(k)$, virtual control input $\hat{x}_{2d}(k)$, desired trajectory $x_{1d}(k+2)$ and system errors $e_1(k)$ and $e_2(k)$. By taking the independent variables as the input to a NN, $\hat{x}_{2d}(k+1)$ can be approximated during control input selection. Consequently, in this paper, a feed forward NN with properly chosen weight tuning law rendering a semi-recurrent or dynamic NN can be used to predict the future value. Alternatively, the value can be obtained by employing a filter [15]. The first layer of the second NN using the system errors, state estimates and past value $\hat{x}_{2d}(k)$ as inputs generates $\hat{x}_{2d}(k+1)$, which in turn is used by the second layer to generate a suitable control input. The results in the simulation section show that the overall controller performance is satisfactory. On the other hand, one can use a single layer dynamic NN to generate the future value of $\hat{x}_{2d}(k)$, which can be utilized as an input to a third control NN to generate a suitable control input. Here, these two single layer NN are combined into a single NN.

Define input as $z_4(k) = \left[x_1(k), x_2(k), x_3(k), e_1(k), l_6 e_2(k), \hat{x}_{2d}(k), x_{1d}(k+2)\right]^T \in \Re^7$, then $u_d(k)$ can be approximated as

$$u_d(k) = w_4^T \phi\left(v_4^T z_4(k)\right) + \varepsilon\left(z_4(k)\right) = w_4^T \phi_4(k) + \varepsilon_4(k), \tag{70}$$

where $w_4 \in \Re^{n_4}$ and $v_4 \in \Re^{7\times n_4}$ denote the constant ideal output and hidden layer weight matrices, $\phi_4(k) \in \Re^{n_4}$ is the activation function vector, n_4 is the number of hidden layer nodes, and $\varepsilon\left(z_4(k)\right)$ is the estimation error. Again, we hold the input weights constant and adapt the output weights only. We also replace actual with estimated states to design the control input as

$$\hat{u}(k) = \hat{w}_4^T(k)\phi\left(v_4^T\hat{z}_4(k)\right) = \hat{w}_4^T(k)\hat{\phi}_4(k) \tag{71}$$

where $\hat{z}_4(k) = \left[\hat{x}_1(k), \hat{x}_2(k), x_3(k), \hat{e}_1(k), l_6\hat{e}_2(k), \hat{x}_{2d}(k), x_{1d}(k+2)\right]^T \in \Re^7$ is the input vector. Rewriting (58) and substituting (69) through (71), we get

$$\begin{aligned}
e_2(k+1) &= x_2(k+1) - \hat{x}_{2d}(k+1) \\
&= \left(f_2(\cdot) + g_2(\cdot)\hat{w}_4^T(k)\hat{\phi}_4(k) + d_2(k)\right) - \hat{x}_{2d}(k+1) \\
&= f_2(\cdot) + g_2(\cdot)\left(\tilde{w}_4^T(k)\hat{\phi}_4(k) + w_4^T\phi_4(k) + w_4^T\tilde{\phi}_4(k)\right) + d_2(k) - \hat{x}_{2d}(k+1) \\
&= f_2(\cdot) + g_2(\cdot)\left(w_4^T(k)\phi_4(k)\right) + g_2(\cdot)\left(\zeta_4(k) + w_4^T\tilde{\phi}_4(k)\right) + d_2(k) - \hat{x}_{2d}(k+1) \\
&= f_2(\cdot) + g_2(\cdot)\left(u_d(k) - \varepsilon_4(k)\right) + g_2(\cdot)\left(\zeta_4(k) + w_4^T\tilde{\phi}_4(k)\right) + d_2(k) - \hat{x}_{2d}(k+1) \\
&= l_6 e_2(k) - g_2(\cdot)\varepsilon_4(k) + g_2(\cdot)\zeta_4(k) + g_2(\cdot)w_4^T\tilde{\phi}_4(k) + d_2(k)
\end{aligned} \tag{72}$$

where

$$\zeta_4(k) = \tilde{w}_4^T(k)\hat{\phi}_4(k) = \hat{w}_4^T(k)\hat{\phi}_4(k) - w_4^T\hat{\phi}_4(k), \tag{73}$$

and

$$\tilde{\phi}_4(k) = \hat{\phi}_4(k) - \phi_4(k) \tag{74}$$

Equations (60) and (72) represent the closed-loop error dynamics. Next we derive the weight update law. Define

$$e_{a2}(k) = \hat{w}_4^T(k)\hat{\phi}_4(k) + \hat{Q}(k), \tag{75}$$

where $e_{a2}(k) \in \Re$ is the error and the subscript a2 stands for the second action NN. Following the similar design, choose a quadratic objective function to minimize

$$E_{a2}(k) = \frac{1}{2}e_{a2}^2(k) \tag{76}$$

Define a gradient-based adaptation where the general form is given by

$$\hat{w}_4(k+1) = \hat{w}_4(k) + \Delta\hat{w}_4(k) \tag{77}$$

$$\Delta\hat{w}_4(k) = \alpha_4\left[-\frac{\partial E_{a2}(k)}{\partial\hat{w}_4(k)}\right] \tag{78}$$

or in other words

$$\hat{w}_4(k+1) = \hat{w}_4(k) - \alpha_4\hat{\phi}_4(k)\left(\hat{w}_4^T(k)\hat{\phi}_4(k) + \hat{Q}(k)\right), \tag{79}$$

The proposed controller structure is shown in Figure 1. Next in the following theorem, it is demonstrated that the closed-loop system is uniformly ultimately bounded. Before we proceed, the following assumptions are needed.

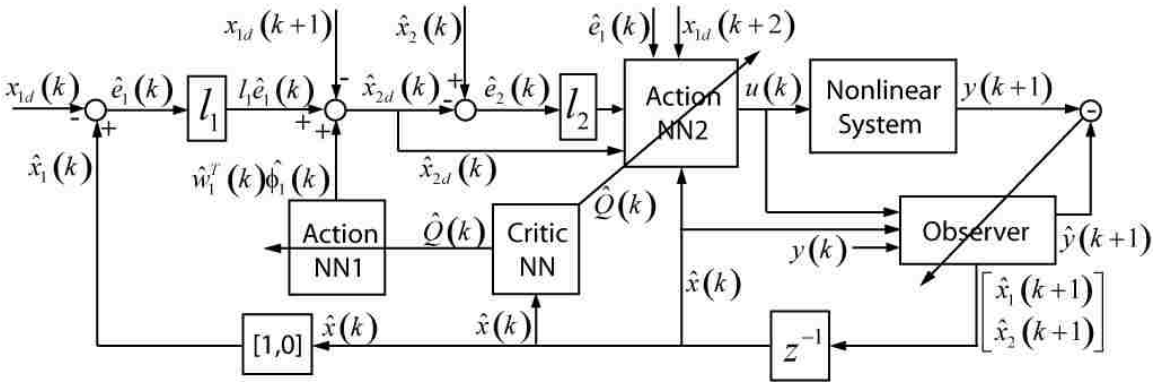

Figure 1 Adaptive-critic NN-based controller diagram.

Assumption 3 (Bounded Ideal Weights): Let w_1, w_2, w_3 and w_4 be the unknown output layer target weights for the observer, critic, and two action NNs, and assume that they are bounded above so that

$$\|w_1\| \le w_{1m},\ \|w_2\| \le w_{2m}, \|w_3\| \le w_{3m}, \text{ and } \|w_4\| \le w_{4m} \tag{80}$$

where $w_{om} \in R^+$, $w_{1m} \in R^+$ and $w_{2m} \in R^+$ represent the bounds on the unknown target weights, and where the Frobenius norm [15] is used.

Fact 1: The activation functions are bounded above by known positive values so that

$$\|\tilde{\phi}_1(\cdot)\| \le \tilde{\phi}_{1m}, \|\tilde{\phi}_2(\cdot)\| \le \tilde{\phi}_{2m}, \|\tilde{\phi}_3(\cdot)\| \le \tilde{\phi}_{3m}, \text{ and } \|\tilde{\phi}_4(\cdot)\| \le \tilde{\phi}_{4m} \tag{81}$$

where $\hat{\phi}_{1m}, \tilde{\phi}_{1m} \in R^+, \hat{\phi}_{2m}, \tilde{\phi}_{2m} \in R^+, \hat{\phi}_{3m}, \tilde{\phi}_{3m} \in R^+ \text{and} \hat{\phi}_{4m}, \tilde{\phi}_{4m} \in R^+$ are the upper bounds.

Theorem 2: Consider the system given by (3) and (4) and the disturbance bounds d_{1m} and d_{2m} to be known constants. Let the observer, critic, virtual control, and control input NN weight tuning be given by (24), (48), (68), and (79), respectively. Let the virtual control input and control input be given by (57), and (71), the estimation errors and tracking errors $e_1(k)$ and $e_2(k)$ and weight estimates $\hat{w}_1(k)$, $\hat{w}_2(k)$, $\hat{w}_3(k)$, and $\hat{w}_4(k)$ are UUB, with the bounds specifically given by (B.17) with the controller design parameters selected as

$$0 < \alpha_1 \left\| \phi_1(k) \right\|^2 < 1 \tag{82}$$

$$0 < \alpha_2 \left\| \phi_2(k) \right\|^2 < 1 \tag{83}$$

$$0 < \alpha_3 \left\| \phi_3(k) \right\|^2 < 1 \tag{84}$$

$$0 < \alpha_4 \left\| \phi_4(k) \right\|^2 < 1 \tag{85}$$

$$|l_1| < \frac{1}{2} \tag{86}$$

$$|l_2| < \frac{\sqrt{3}}{3} \tag{87}$$

$$|l_3| < \frac{\sqrt{3}}{3} \tag{88}$$

$$|l_4| < \frac{\sqrt{3}}{3} \tag{89}$$

$$|l_5| < \frac{1}{\sqrt{5}} \tag{90}$$

$$|l_6| < \frac{\sqrt{3}}{3} \tag{91}$$

$$0 < \beta < \frac{\sqrt{2}}{2} \tag{92}$$

where α_1, α_2, α_3 and α_4 are NN adaptation gains, l_1, l_2, l_3, l_4, l_5, and l_6 are controller gains, and β is employed to define the *strategic* utility function.

Proof: See Appendix B. ■

Remark 1: A well-defined controller is developed in this paper since a single NN is utilized to approximate two nonlinear functions. This avoids undefined areas when the denominator approaches zero.

Remark 2: It is important to note that in this theorem, there is no persistency of excitation condition (PE) condition and linearity in the parameters assumption for the NN observer and controller, in contrast with standard work in the discrete-time adaptive control since the first difference does not require the PE condition to prove the boundedness of the weights. Even though the input to the hidden-layer weight matrix is

not updated and only the hidden to the output-layer weight matrix is tuned, the NN method relaxes the linear in the unknown parameter assumption. Additionally, the certainty equivalence principle is not used.

Remark 3: Generally, the separation principle used for linear systems does not hold for nonlinear systems, and hence it is relaxed in this paper for the controller design since the Lyapunov function is a quadratic function of system errors and weight estimation errors of the observer and controller NNs.

Remark 4: The NN weight tuning proposed in (24), (48), (68), and (79) renders a semi-recurrent NN due to the proposed weight tuning law even though a feedforward NN is utilized. Here the NN outputs are not fed as delayed inputs to the network whereas the outputs of each layer are fed as delayed inputs to the same layer. This semi-recurrent NN architecture renders a dynamic NN which is capable of predicting the state one step ahead.

Remark 5: It is only possible to show boundedness of all the closed–loop signals by using an extension of Lyapunov stability [15] due to the presence of approximation errors and bounded disturbances consistent with the literature.

Corollary 1: The proposed adaptive critic NN controller and the weight updating rules with parameter selection based on (82) through (92) cause the state $x_2(k)$ to approach the desired virtual control input $x_{2d}(k)$.

Proof: Combining (56) and (57), the difference between $\hat{x}_{2d}(k)$ and $x_{2d}(k)$ is given by

$$\hat{x}_{2d}(k) - x_{2d}(k) = \tilde{w}_3(k)\phi_3(k) - \varepsilon(z_3(k)) = \zeta_3(k) - \varepsilon_3(k) \tag{93}$$

where $\tilde{w}_3(k) \in \Re^{n_3}$ is the first action NN weight estimation error and $\zeta_3(k) \in \Re$ is defined in (61). Since both $\zeta_3(k) \in \Re$ and $\varepsilon_3(k)$ are bounded, $\hat{x}_{2d}(k)$ is bounded near $x_{2d}(k)$. In *Theorem 1*, we show that $e_2(k)$ is bounded, i.e., the state $x_2(k)$ is bounded to the virtual control signal $\hat{x}_{2d}(k)$. Thus the state $x_2(k)$ is bounded to the desired virtual control signal $x_{2d}(k)$.

VII. SECONDARY CONTROLLER – CRITIC DESIGN

For maintaining dilution to a desired level, the third equation will be employed with EGR(k) as the control input and inert gas as an additional state. To simplify the controller development, and since the residual gas fraction is upper bounded, this third equation can be simplified as

$$x_3(k+1) = f_4\left(x(k)\right) + g_4\left(x(k)\right)v(k) + d_3(k) \tag{94}$$

where $x(k) = [x_1(k), x_2(k), x_3(k)]^T$, and the above equation can be represented as a standard affine nonlinear discrete-time system. The design of the controller is different than the non-strict feedback nonlinear discrete-time system given by (1) and (2). The design of a novel reinforcement controller is introduced here by assuming that the third state is measurable. Define

$$x(k) = [x_1(k), x_2(k), x_3(k)]^T \tag{95}$$

A. Design of the Critic

Let the long-term cost function be defined as

$$J(k) = \sum_{i=t_0}^{\infty} \gamma^i r(k+i) \tag{96}$$

where

$$r(k) = \left(x(k) - x_d(k)\right)^T Q\left(x(k) - x_d(k)\right) + v^T(k) R v(k) \tag{97}$$

where R and Q are positive definite matrices and γ is the discount factor within the range of $0 \le \gamma \le 1$. Invoke the universal approximation property of NN to estimate (96) as

$$J(k) = w_c^T \phi_c\left(v_c^T z_c(k)\right) + \varepsilon\left(z_c(k)\right) \tag{98}$$

where $\varepsilon\left(z_c(k)\right)$ is the estimation error. Replace the states with estimated states.

$$\hat{J}(k) = \hat{w}_c^T(k)\phi_c\left(v_c^T \hat{z}_c(k)\right) = \hat{w}_c^T(k)\phi_c(k) \tag{99}$$

where $\hat{w}_c \in \Re^{n_c}$ and $v_c \in \Re^{2\times n_c}$ denote the ideal output and constant hidden layer weights, $\phi_c(k) \in \Re^{n_c}$ is the activation function vector, and n_c is the number of hidden layer nodes. Again, we hold the input weights constant and adapt the output weights only. $\hat{z}_c(k) = \left[\hat{x}_1(k), \hat{x}_2(k), x_3(k)\right]^T \in \Re^3$ is the input vector.

B. Critic Weight Update Law

Define the prediction error as

$$\begin{aligned} e_c(k) &= \gamma \hat{J}(k) - [\hat{J}(k-1) - r(k)] \\ &= \gamma \zeta_c(k) + \gamma J(k) - \zeta_c(k-1) - J(k-1) + r(k) - \varepsilon_c(k) + \varepsilon_c(k-1) \end{aligned} \tag{100}$$

where

$$\zeta_c(k) = \tilde{w}_c^T(k)\hat{\phi}_c(k) = \hat{w}_c^T(k)\hat{\phi}_c(k) - w_c^T\hat{\phi}_c(k) \tag{101}$$

Use a quadratic minimizing function

$$E_c(k) = \frac{1}{2}e_c^2(k) \tag{102}$$

Use a standard gradient-based adaptation method, the general formula is given by

$$\hat{w}_c(k+1) = \hat{w}_c(k) + \Delta\hat{w}_c(k) \tag{103}$$

where

$$\hat{w}_c(k) = \alpha_c\left[-\frac{\partial E_c(k)}{\partial \hat{w}_c(k)}\right] \tag{104}$$

therefore

$$\begin{aligned} \hat{w}_c(k+1) &= \hat{w}_c(k) + \alpha_c\left[-\frac{\partial E_c(k)}{\partial \hat{w}_c(k)}\right] \\ &= \hat{w}_c(k) - \alpha_c\gamma\phi_c(k)e_c(k) \\ &= \hat{w}_c(k) - \alpha_c\gamma\phi_c(k)\left(\gamma\hat{J}(k) + r(k) - \hat{J}(k-1)\right) \end{aligned} \tag{105}$$

VIII. Secondary Controller – Control Input Design

A. Design of the Control Input

The tracking error is defined as

$$\begin{aligned} e_4(k) &= x_3(k) - x_{3d}(k) \\ e_4(k+1) &= f_4(\cdot) + g_4(\cdot)v(k) + d_3(k) - x_d(k+1) \end{aligned} \tag{106}$$

where $x_{3d}(k)$ is the target bounded trajectory. Define the desired control signal as

$$v_d(k) = g_4^{-1}(\cdot)\left(-f_4(\cdot) + x_{3d}(k+1) + l_7 e_4(k)\right) \tag{107}$$

Using the universal approximation property of NN and the approximate states

$$\hat{v}_d(k) = \hat{w}_a^T(k)\phi_a\left(v_a^T\hat{z}_a(k)\right) = \hat{w}_a^T(k)\hat{\phi}_a(k) \tag{108}$$

where $\hat{w}_a \in \Re^{n_a}$ and $v_a \in \Re^{2\times n_a}$ denote the ideal output and constant hidden layer weight matrices, $\phi_a(k) \in \Re^{n_a}$ is the activation function vector, n_a is the number of hidden layer nodes, and $\hat{z}_a(k) = \left[\hat{x}_1(k), \hat{x}_2(k), x_3(k)\right]^T \in \Re^3$ is the input vector. Again, we hold the input weights constant and adapt the output weights only. Rewrite (106) as

$$\begin{aligned} e_4(k+1) &= l_7 e_4(k) + g(\cdot)\left(v(k) - v_d(k)\right) + d_3(k) \\ &= l_7 e_4(k) + g(\cdot)\left(\tilde{w}_a^T(k)\phi_a(k) - \varepsilon_a(k)\right) + d_3(k) \\ &= l_7 e_4(k) + g(\cdot)\zeta_a(k) + d_a(k) \end{aligned} \tag{109}$$

where

$$d_a(k) = -g(\cdot)\varepsilon_a(k) + d_a(k) \tag{110}$$

and

$$\zeta_a(k) = \tilde{w}_a^T(k)\hat{\phi}_a(k) \tag{111}$$

B. Control Input Weight Update Law

Define the control input cost function

$$\begin{aligned} e_a(k) &= \sqrt{g_4(\cdot)}\zeta_a(k) + \left(\sqrt{g_4(\cdot)}\right)^{-1}\left(J(k) - J_d(k)\right) \\ &= \sqrt{g_4(\cdot)}\zeta_a(k) + \left(\sqrt{g_4(\cdot)}\right)^{-1} J(k) \end{aligned} \tag{112}$$

where $J_d(k)$ is the desired long-term cost function and is equal to zero. Define a quadratic error to minimize

$$E_a(k) = \frac{1}{2}e_a^2(k) \tag{113}$$

Utilizing a gradient decent minimization strategy

$$\begin{aligned} \hat{w}_a(k+1) &= \hat{w}_a(k) + \alpha_a\left[-\frac{\partial E_a(k)}{\partial \hat{w}_a(k)}\right] \\ &= \hat{w}_a(k) - \alpha_a\gamma\phi_a(k)\left(g_4(\cdot)\zeta_a(k) + J(k)\right)^T \\ &= \hat{w}_a(k) - \alpha_a\gamma\phi_a(k)\left(e_4(k+1) - l_7 e_4(k) - d_a(k) + J(k)\right)^T \end{aligned} \tag{114}$$

Figure 2 shows the overall controller structure including both controllers.

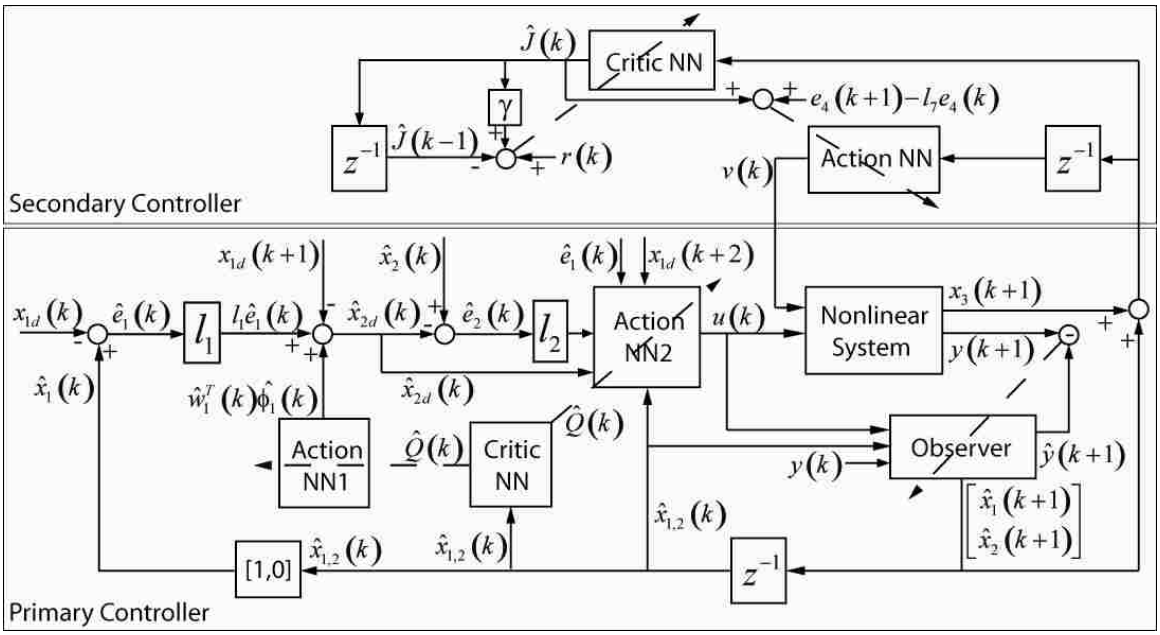

Figure 2 Combined primary and secondary controller structure.

Theorem 2: Consider the system given by (5), and the disturbance bound d_{3m} to be known constants. Let the observer and control input NN weight tuning be given by (105) and (114), respectively. Let the control input be given by (108); the tracking error $e_4(k)$ and weight estimates $\hat{w}_a(k)$ and $\hat{w}_c(k)$ are UUB, with the bounds specifically given by (C.1) with the controller design parameters selected as:

$$0 < \alpha_a \left\| \phi_a(k) \right\|^2 < 1 \tag{115}$$

$$0 < \alpha_c \left\| \phi_c(k) \right\|^2 < 1 \tag{116}$$

$$\left| l_7 \right| < \frac{1}{2} \tag{117}$$

where α_a and α_c are NN adaptation gains, and l_7 is the controller gain.

Proof: See Appendix C. ■

IX. Results and Analysis

EGR operation of an SI engine allows lower emissions and improved fuel efficiency. However, EGR operation destabilizes the engine due to the cyclic dispersion of heat release. The adaptive critic NN controller is designed to stabilize the SI engine operating at EGR conditions.

A. Daw Engine Model

Spark ignition (SI) engine dynamics can be expressed according to the Daw model as a class of nonlinear systems in non-strict feedback form [16]. At high EGR levels, the engine can be expressed as the following [17].

$$x_1(k+1) = AF(k) + F(k)x_1(k) - R \cdot F(k)CE(k)x_2(k) + F(k)\left(r_{O_2}(k) + r_{N_2}(k)\right) + d_1(k) \tag{118}$$

$$x_2(k+1) = \left(1 - CE(k)\right)F(k)x_2(k) + \left(MF(k) + u(k)\right) + d_2(k) \tag{119}$$

$$x_3(k+1) = F(k)\left(r_{CO_2}(k) + r_{H_2O}(k) + r_{N_2}(k) + x_3(k) + EGR(k)\right) \tag{120}$$

$$y(k) = x_2(k)CE(k) \tag{121}$$

$$\varphi(k) = R\frac{x_2(k)}{x_1(k)}\left[1 - \gamma\frac{x_3(k) + EGR(k)}{\left(x_3(k) + x_1(k) + x_3(k) + EGR(k)\right)}\right] \tag{122}$$

$$CE(k) = \frac{CE_{\max}}{1 + 100^{-(\varphi(k) - \varphi_m)/(\varphi_u - \varphi_l)}} \tag{123}$$

$$\varphi_m = \frac{\varphi_u - \varphi_l}{2} \tag{124}$$

$$r_{H_2O}(k) = \gamma_{H_2O}x_2(k)CE(k) \tag{125}$$

$$r_{O_2}(k) = \gamma_{O_2}x_2(k)CE(k) \tag{126}$$

$$r_{N_2}(k) = \gamma_{N_2}R \cdot x_2(k)CE(k) \tag{127}$$

$$r_{CO_2}(k) = \gamma_{CO_2}x_2(k)CE(k) \tag{128}$$

where $x_1(k)$, $x_2(k)$, and $x_3(k)$ are total mass of air, fuel, and inert gas, respectively. $y_1(k)$ is the heat release at k^{th} instance. The value of $CE(k)$ is within the range of $0 < CE_{\min} < CE(k) < CE_{\max}$. $F(k)$ is bounded by $0 < F_{\min} < F(k) < F_{\max}$. $d_1(k)$ and $d_2(k)$ are unknown but bounded disturbances bounded by $|d_1(k)| < d_{1m}$ and $|d_2(k)| < d_{2m}$ with d_{1m} and d_{2m} being unknown positive scalars. $\varphi_m, \varphi_l, \varphi_u$ are equivalence ratio system parameters. The terms $r_{H_2O}(k)$, $r_{O_2}(k)$, $r_{N_2}(k)$, and $r_{CO_2}(k)$

are the mass of water, oxygen, nitrogen, and carbon dioxide, respectively whereas γ, γ_{H_2O}, γ_{O_2}, γ_{N_2}, and γ_{CO_2} are design constants, and constants associated with their respective chemicals.

Equation (5) can be controlled by the secondary controller; however, in this case, for convenience, we assume that it provides a bounded input to the primary system. We set it to a constant which will simplify the controller implementation, as the third state is considered to be a fixed value. Note that this deterministic model accounts for stochastic effects by randomly fluctuating parameters such as injected air-fuel ratio or residual fraction. Other complex processes like temperature variation, turbulence, and fuel vaporization are not modeled but are assumed to add additional noise to the engine output. To implement the observer, replace the following from the Daw model.

$$\begin{aligned} f_1(\cdot) &= AF(k) + F(k)x_1(k) + F\left(r_{O_2}(k) + r_{N_2}(k)\right) \\ g_1(\cdot) &= -R \cdot F(k)CE(k) \\ f_2(\cdot) &= \left(1 - CE(k)\right)F(k)x_2(k) + MF(k) \\ g_2(\cdot) &= 1 \end{aligned} \tag{129}$$

and

$$\begin{aligned} f_{10} &= AF_0 + F_0\hat{x}_1(k) \\ g_{10} &= -R \cdot F_0CE_0 \\ f_{10} &= \left(1 - CE_0\right)F_0\hat{x}_2(k) + MF_0 \\ g_{10} &= 1 \end{aligned} \tag{130}$$

Note that we omitted the residuals in f_{10}, because they are not available. The error introduced by this is accounted for in the air estimation error. To implement the controller, replace the following in place of $f_1(\cdot)$ and $g_1(\cdot)$

$$\begin{aligned} \Phi(\cdot) = {} & AF(k) + F(k)x_1(k) - R \cdot F(k)CE(k)x_2(k) + x_2(k) + \\ & F(k)\left(r_{O_2}(k) + r_{N_2}(k)\right) \end{aligned} \tag{131}$$

B. Simulation Data

The controller is easily simulated in C in conjunction with the Daw model. The learning rates for the observer (82), critic (83), virtual control input (84), and control input (85) networks are 0.01, 0.01, 0.01, and 0.01, respectively. The gains l_1, l_2, l_3, l_4, l_5,

and l_6 are selected as 0.05, 0.05, 0.04, 0.05, 0.2 and 0.1. The system constants CE_{max}, φ_l, and φ_u are chosen as 1, 0.54, and 0.58. The critic constants β and N are 0.4 and 4 for all EGR levels. All NNs use 20 hidden neurons with hyperbolic tangent sigmoid activation functions in the hidden layer.

The maximum moles a single cylinder holds is set as 0.021 to match the experimental engine constraint shown in the next section. The last two system variables: disturbances and stochastic effects are modeled as follows. First, we assume a Gaussian distribution governs the two effects. We may inject disturbances to the two states in equations (118) and (119) due to $d_1(k)$ and $d_2(k)$, but a simpler method is to perturb the equivalence ratio equation (122). This simplification is sufficient because the states are not measurable; therefore, the disturbances are increasingly complex and immeasurable. Stochastic effects alter the output, and through the combustion efficiency equation (123) and finally the output equation (121), this single perturbation effectively models the last two system variables. The final model uses a Gaussian distribution noise injected into equation (122) centered around the target equivalence ratio and deviation of 0.007. The resulting simulation output matches to the output observed from the Ricardo engine. All simulations ran for 5000 cycles uncontrolled first, then 5000 cycles controlled.

Figure 3 shows two heat release return maps, one controlled and the other uncontrolled, for the set point at 13% EGR. Each subfigure shows the next time step versus the current time step heat release. Points centered along the 45 degree line represent heat release values that are equal to the next step heat release. Note the clustering of the points around the mean heat release of 850J. The square represents the target heat release. At this set point, the heat release dispersion starts to affect the engine performance, indicated by the stray points away from the central cluster. There are no complete misfires, but the heat release variation can be clearly seen. Figure 4 shows the time series of the heat release and control input at the same EGR level. The controller activates after several thousand cycles, indicated by the fluctuation of the control output. The controller converges quickly, and to a stable operation point. The presence of spikes in the control output indicates a decline in heat release such as a misfire, translating into additional fuel control to counteract.

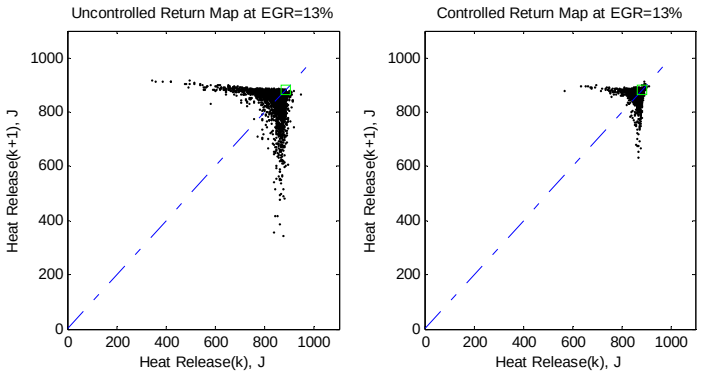

Figure 3 Uncontrolled and controlled heat release return map at 13% EGR. Heat release at k+1 instance is plotted against heat release at k instance.

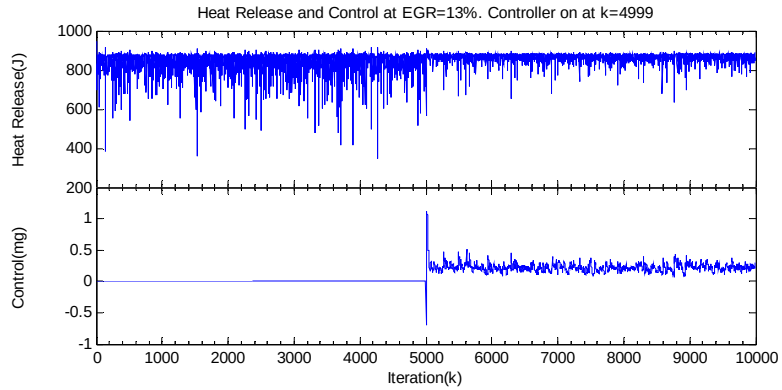

Figure 4 Heat release vs iteration number at 13% EGR. Controller turns on at k=4000. Note the almost instant learning convergence of the controller.

Figures 5 and 6 show another set point at 19% EGR. Similar features appear compared to the previous EGR level, except with higher frequency and amplitude of dispersion. Improvements shown reflect the assertion of the control action.

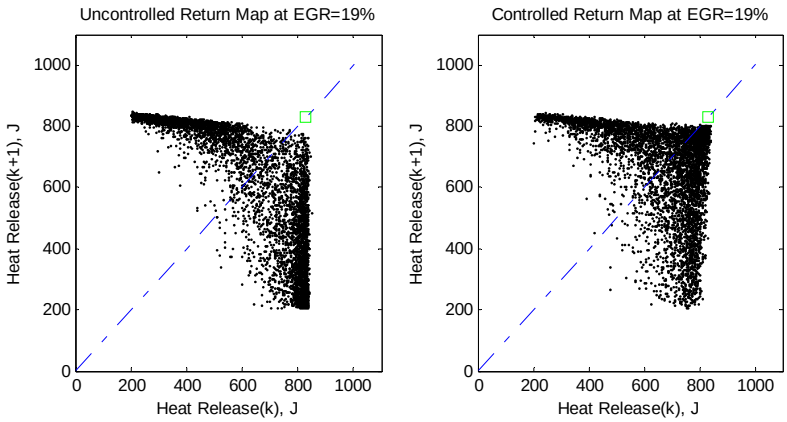

Figure 5 Uncontrolled and controlled heat release return map at 19% EGR.

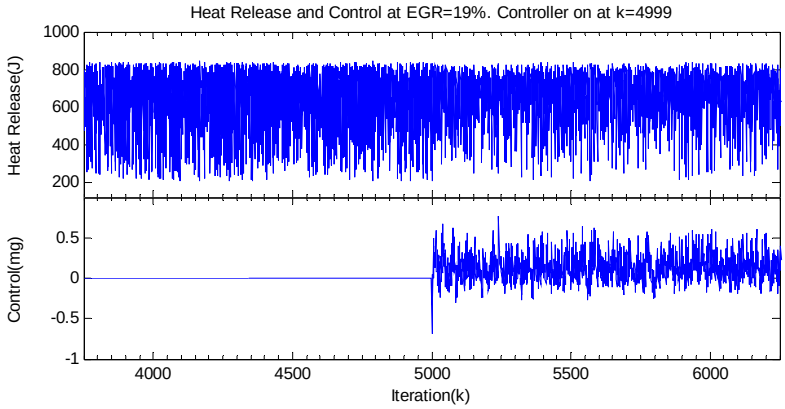

Figure 6 Heat release and control input at 19% EGR.

In order to quantify the performance of the controller, we compare the coefficient of variation (COV), which is the standard deviation normalized by dividing by the mean of the heat release. As the COV decreases, the standard deviation decreases, which indicates that the engine heat release is more stable compared to higher COV. The controller performs better, and the return map consequently should approach the target value. Table 1 tabulates all of the data from the simulation. The COV of each set point decreased drastically (shown with a negative sign) as the controller operated. The performance exceeded the improvement due to the slight increase in the mean fuel input. Next, we show that experimental data supports the simulation data.

Table 1 Coefficient of variation (COV) and fuel data for each of the four set points.

EGR Fraction	COV Uncontrolled	COV Controlled	%COV Change	%Fuel Change
0.00	0.0058	0.0057	-0.75	0.00
0.13	0.0548	0.0384	-29.94	0.40
0.15	0.1387	0.0773	-44.30	0.71
0.19	0.3421	0.2383	-30.34	0.42

C. Ricardo Engine

The experimental results are collected from a Ricardo Hydra engine with a modern four valve Ford Zetek head. It contains a single cylinder running at 1000 rpm with shaft encoders to signal each crank angle degree and start of cycle. There are 720° per engine cycle.

In the cylinder, a piezoelectric pressure transducer records pressure every crank angle degree. Combustion is considered to take place between 345° to 490°, for a total of 145 pressure measurements. The cylinder pressure is integrated along with volume during the 17.7 ms calculation window. All communications are completed at this time. The output of our controller controls the fuel input. This is controlled by a TTL signal to a fuel injector driver circuit.

All signals communicate through a custom interface board using a microcontroller. The board interfaces with the PC through a parallel port and with the engine hardware through an analog signal.

D. Experimental Data

All constants given in the simulation section are used in the experiment. The first operation for an engine run is to measure the air flow and nominal fuel. The desired EGR set point equation is given by (132). m_{EGR} is the mass of inert gas introduced at each cycle, which is nitrogen in the lab and exhaust gas in production applications. m_f and m_a are mass of fuel and mass of air, respectively.

$$\%EGR = 100 \times \left(\frac{m_{EGR}}{m_f + m_a + m_{EGR}} \right) \tag{132}$$

These values are loaded into the controller. Ambient pressure is used to reference the in-cylinder pressures when the exhaust valve is fully open and subtracted from the combustion pressure measurements. Uncontrolled and controlled data were collected at EGR percentages of 18, 20, and 23. The uncontrolled engine ran for 5,000 cycles and then the controller is turned on for another 5,000 cycles. Steady state was ensured prior to data collection by measuring stable exhaust temperatures.

Figure 7 shows two heat release return maps, one controlled and the other uncontrolled, for the 18% EGR set point. The target heat release is at 870J. At this EGR level, cyclic dispersion can clearly be seen, indicated by deviation of the points away from the main cluster on the 45 degree line. Figure 8 shows the time series of the heat release and control input for the same set point.

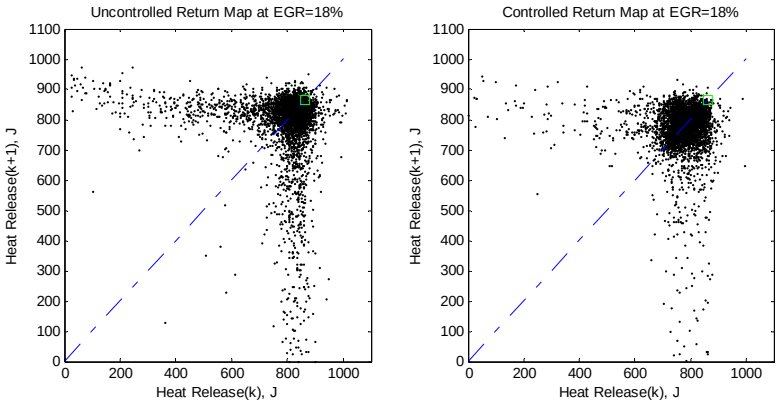

Figure 7 Uncontrolled and controlled heat release return map at EGR=18%. Heat release at *k*+1 instance is plotted against heat release at *k* instance.

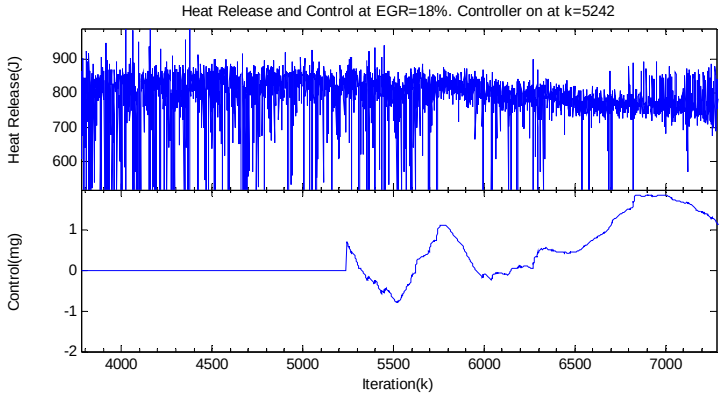

Figure 8 Heat release and control input at EGR=18%. Controller turns on at k=5200.

Define the state and output tracking errors.

$$\begin{aligned}\hat{e}_1(k) &= \hat{x}_1(k) - x_{1d}(k)\\ \hat{e}_2(k) &= \hat{x}_2(k) - \hat{x}_{2d}(k)\\ \hat{e}_y(k) &= \hat{y}(k) - y(k)\end{aligned} \tag{133}$$

Where $\hat{e}_1(k)$, $\hat{e}_2(k)$, and $\hat{e}_y(k)$ are state 1, state 2, and output tracking errors, respectively. Figure 9 shows the controller state tracking errors at a set point of 18% EGR. The range represents tracking error in percentage over and under the desired state trajectories. State one tracking error is considerably better than state two tracking. The second state tracks within 0.5%; therefore, both are performing well. The spikes indicate unsuccessful tracking. Consequently, the observer and controller converged together to the desired states and estimated states, generating a stable error system. Figure 10 shows

the output tracking error in the same form as the state tracking error. Immediate observation shows an extremely high error rate. The observer performance is abysmal. Nonetheless, this signal fed into the NN controller allows for the critical performance factor, state tracking errors, to converge and stabilize. It is not critical for one signal to track perfectly, rather the system as a whole. Moreover, theorem 1 proved the boundedness of the output estimation. In conjunction with the natural bound of the engine output, the tracking error will always be bounded. The extreme fluctuation of the observer output may be the key to the responsiveness of the controller as a whole.

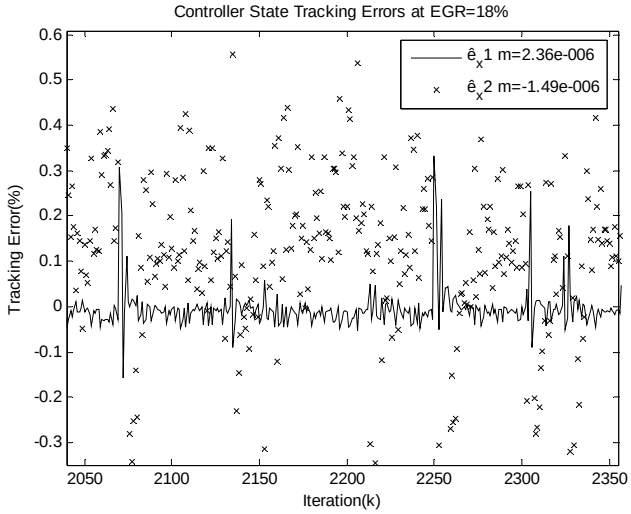

Figure 9 State tracking errors.

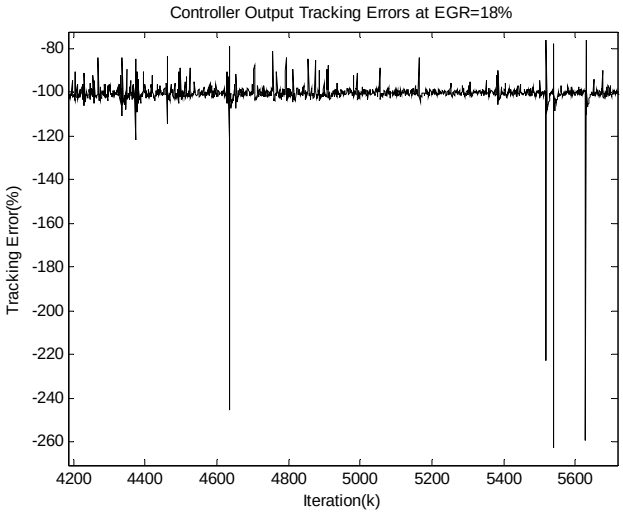

Figure 10 Output tracking error.

Figure 11 shows the return map of the heat release for 20% EGR. Note that as the equivalence ratio decreases, the return map spreads out and dispersion increases. Figure 12 is the corresponding heat release and control time series. Misfires increase in frequency, as shown by the negative heat release spikes due to heat transfer from the cylinder to the environment without internal generation of useful work by combustion.

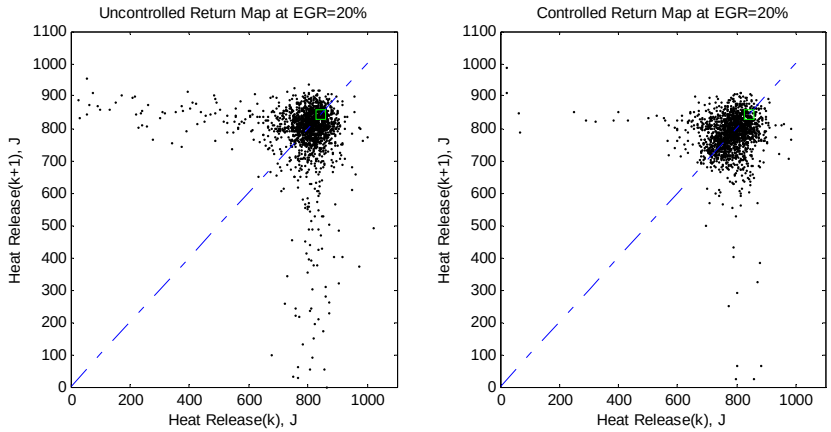

Figure 11 Uncontrolled and controlled heat release return map at 20% EGR.

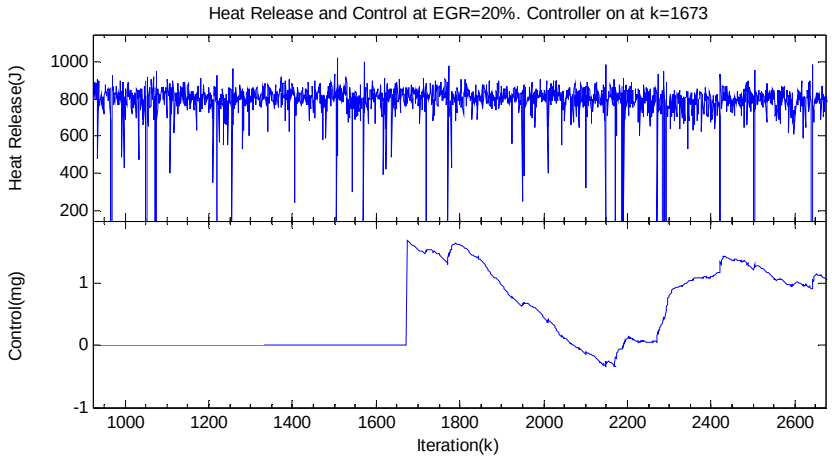

Figure 12 Heat release and control input at 20% EGR.

Figure 13 shows increasing difficulty of the observer and controller to generate a low state tracking error compared to the previous case. As the engine operates in higher EGR modes, overall dispersion increases, thus degrading observer performance. Although the performance is reduced, the tracking error is well within satisfactory performance. Figure 14 shows the output tracking error. At the higher EGR set point, it is performing better than in the previous case. This may be due to the memory effect of past engine cycles

contributing to the residuals in the current cycle. At near zero EGR, little dispersion occurs, resulting in similar cylinder chemistry content before each power cycle. Stochastic effects dominate and destroy predictability. The high observer learning rate decimates the tracking ability. On the other hand, at lower EGR levels, higher dispersion and misfires create patterns of predictable residuals. The observer exploits the pattern recognition power of NN to drastically improve its performance.

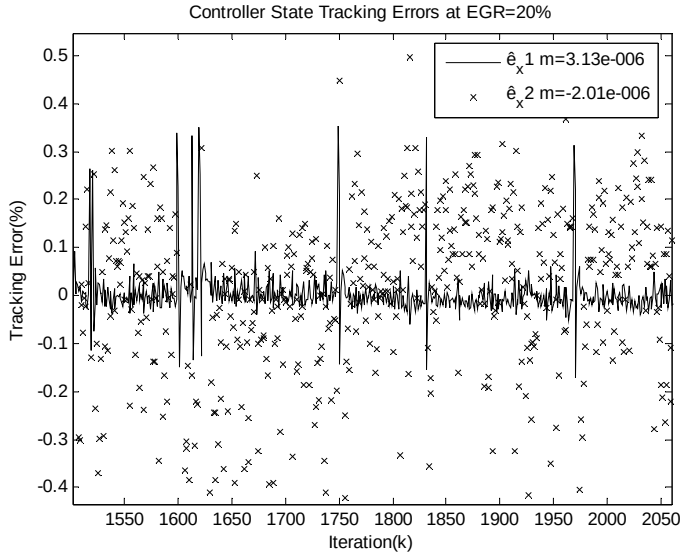

Figure 13 State tracking errors.

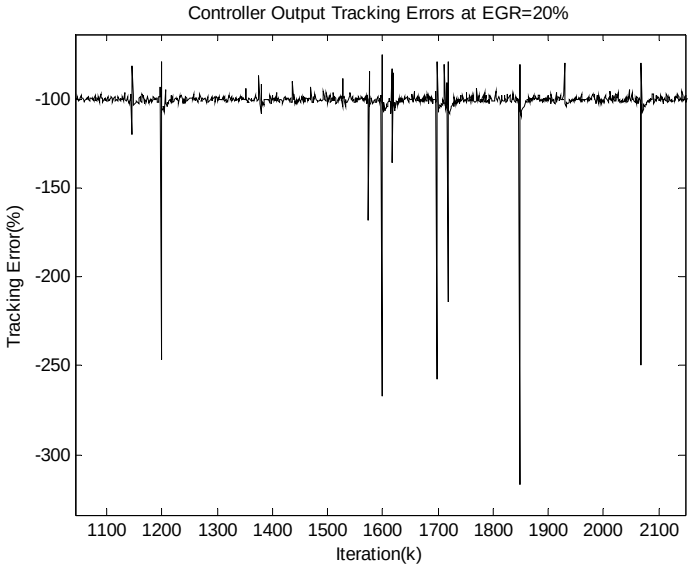

Figure 14 Output tracking error.

Figure 15 shows a detailed view of 70 controlled cycles at 20% EGR. The controller generates decreasing control during cycles when the heat release is steady, indicated by cycles between 4805 to 4818 and between 4822 to 4836. However, during misfires or extreme dispersion in heat release, the controller attempts to compensate for the drop in heat release by pushing the control up, indicated by cycles 4819, 4847, etc. The controller compensates after a one cycle delay in the positive direction and attempts to recover the engine heat release towards the target point. It is difficult to determine success on cycles with no misfire, because no heat release plots are available for uncontrolled case during the same cycles when the controller is operating for comparison. Overall, the controller performs to general expectation.

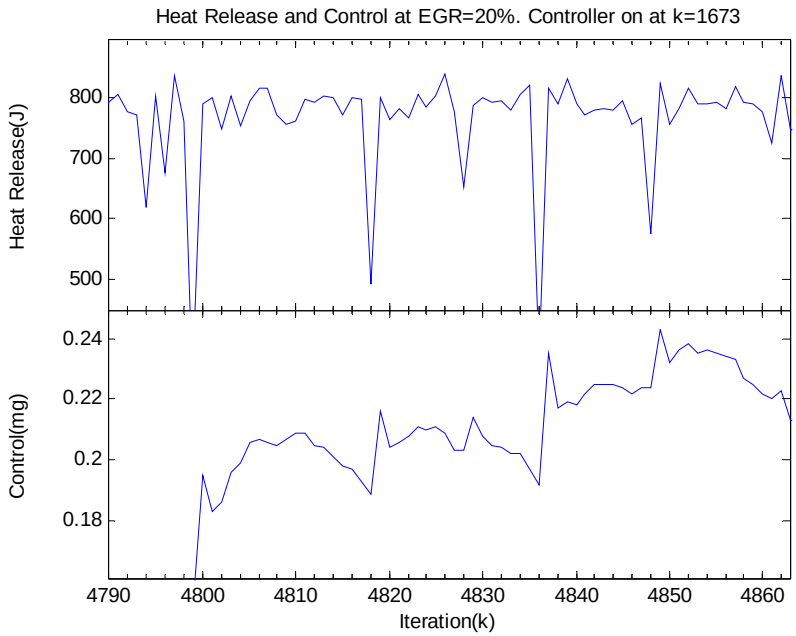

Figure 15 Detailed view of 70 controlled cycles at 20% EGR

Table 2 shows the improved COV when the controller is in operation compared to an uncontrolled engine and also the corresponding change in nominal fuel. An improvement in the COV may be artificial due to an increase in fuel input. However, this is not the case for this controller. At all EGR set points except 23%, the increase in fuel input is well within the tolerance of the equipment. On average, the COV decreased significantly by 25% compared to the controlled case.

Table 2 Coefficient of variation (COV) and fuel data for each of the three set points.

EGR	COV Uncontrolled	COV Controlled	%COV Change	%Fuel Change
0.18	0.2112	0.1511	-28.4	1.36
0.20	0.2139	0.1400	-34.6	0.77
0.23	0.5777	0.5066	-12.3	2.11

The COV and fuel change data indicates an improved performance compared to the previous controller [18]. The average drop in COV was 17% between uncontrolled and controlled, compared to 25% for the current controller. Although this seems to indicate an increase in performance, we must also consider the increase in average fuel input in conjunction. The previous controller increased the average fuel to 2.4%. This is well beyond the detection error. This controller, however, averages less than 1%, safely below the detection error. The controller fuel increases negligibly while approach the performance of the previous controller. Therefore, this controller performs better and at the same time exerts less impact on the fuel.

X. Conclusions

The controller presented successfully controlled a SI engine to reduce cyclic dispersion under higher EGR conditions. The system is modeled under a non-strict feedback nonlinear descrete-time system. It converged upon a near *optimal* solution through the use of a long-term strategic utility function even though the exact dynamics are not known beforehand. It was shown through simulation that the controller is stable under a variety of set points. In experimental results, the COV was reduced when the controller was turned on. At the same time, the average fuel input did not change significantly; therefore, the improvements are solely due to the effects of the controller. The output is stable, as predicted by the Lyapunov proof. There was also a significant reduction in unburned hydrocarbon between controlled and uncontrolled.

Appendix A

Tables A.1 and A.2 show the improvement in emissions for several equivalence ratios. The improvement is superior than what we have seen before [18] using another controller. NO_x is reduced between 2% to 7.4% from uncontrolled scenario. However,

CO_2 remains unchanged, whereas O_2 decreases by about 20%. The unburned hydrocarbons (uHC) decreased nominally compared with uncontrolled. This is mainly due to reduced cyclic dispersion.

Table A.1 Emissions data for select EGR set points.

	Uncontrolled			Controlled			Change(%)		
EGR	**CO_2(%)**	**CO(%)**	**O_2(%)**	**CO_2(%)**	**CO(%)**	**O_2(%)**	**CO_2**	**CO**	**O_2**
0.13	7.4	0.1	6.9	7.4	0.1	2.2	-0.4	0.0	-68.1
0.18	7.3	0.1	2.3	7.3	0.1	2.4	0.0	0.0	5.2
0.20	7.3	0.1	3.2	7.3	0.0	3.6	0.5	-20.0	10.8
0.23	7.3	0.1	5.4	7.3	0.1	5.8	-0.8	0.0	8.2

Table A.2 Unburned hydrocarbon emission data.

	Uncontrolled		Controlled		Change(%)	
EGR	**NO_X (ppm)**	**uHC (ppm C_1)**	**NO_X (ppm)**	**uHC (ppm C_1)**	**NO_X**	**uHC**
0.13	554.0	10619	478	10677	-13.7	0.5
0.18	82.0	13610	92	12605	12.2	-7.4
0.20	51.0	14450	51	14108	0.0	-2.4
0.23	50.0	23928	55	22345	10.0	-6.6

APPENDIX B

Proof of Theorem 1: Define the Lyapunov function

$$J(k)=\sum_{i=1}^{10}J_i(k)=\frac{\gamma_1}{5}e_1^2(k)+\frac{\gamma_2}{3}e_2^2(k)+\sum_{j=3}^{6}\frac{\gamma_j}{\alpha_{j-2}}\tilde{w}_j^T(k)\tilde{w}_j(k)+ \\ \gamma_7\zeta_2^2(k-1)+\frac{\gamma_8}{3}\tilde{x}_1^2(k)+\frac{\gamma_9}{3}\tilde{x}_2^2(k)+\frac{\gamma_{10}}{3}\tilde{y}^2 \tag{B.1}$$

where $0<\gamma_i, i\in\{1,...,6\}$ are auxiliary constants; the NN weights estimation errors $\tilde{w}_1^T(k+1)$, $\tilde{w}_2^T(k+1)$, $\tilde{w}_3^T(k+1)$, and $\tilde{w}_4^T(k+1)$ are defined in (24), (48), (68), and (79), by subtracting their respective ideal weights $w_i, i\in\{1,2,3,4\}$ on both sides; the observation errors $\tilde{x}_1(k+1)$, $\tilde{x}_2(k+1)$, are defined in (21) and (22), respectively; the system errors $e_1(k+1)$ and $e_2(k+1)$ are defined in (60) and (72), respectively; and $\alpha_i, i\in\{1,2,3,4\}$ are NN adaptation gains. The Lyapunov function (B.1) obviates the need for the CE condition. Taking the first term and the first difference using (60) to get

$$
\begin{aligned}
J_1(k) &= \tfrac{\gamma_1}{5} e_1^2(k) \\
\tfrac{5}{\gamma_1}\Delta J_1(k) &= e_1^2(k+1) - e_1^2(k) \\
&= \left(-\zeta_3(k) - w_3^T\tilde{\phi}_3(k) + \varepsilon_3(k) - l_5\hat{e}_1(k) - e_2(k) + d_1(k)\right)^2 - e_1^2(k) \\
&= \left(-\zeta_3(k) - w_3^T\tilde{\phi}_3(k) + \varepsilon_3(k) - l_5\left(\tilde{x}_1(k) + e_1(k)\right) - e_2(k) + d_1(k)\right)^2 - e_1^2(k)
\end{aligned}
\tag{B.2}
$$

Invoke the Cauchy-Schwarz inequality defined as

$$
\left(a_1b_1 + \ldots + a_nb_n\right)^2 \le \left(a_1^2 + \ldots + a_n^2\right)\left(b_1^2 + \ldots + b_n^2\right) \tag{B.3}
$$

Simplify to get

$$
\begin{aligned}
\tfrac{1}{\gamma_1}\Delta J_1(k) &\le \left(\zeta_3^2(k) + l_5^2\tilde{x}_1^2(k) + l_5^2e_1^2(k) + e_2^2(k) + \left(\varepsilon_3(k) - w_3\tilde{\phi}_3(k) + d_1(k)\right)^2\right) - \tfrac{1}{5}e_1^2(k) \\
\Delta J_1(k) &\le \gamma_1\zeta_3^2(k) + \gamma_1 l_5^2\tilde{x}_1^2(k) + \gamma_1 l_5^2e_1^2(k) + \gamma_1 e_2^2(k) + \\
&\quad \gamma_1\left(\varepsilon_3(k) - w_3\tilde{\phi}_3(k) + d_1(k)\right)^2 - \tfrac{\gamma_1}{5}e_1^2(k) \\
&\le \gamma_1 l_5^2\tilde{x}_1^2(k) + \gamma_1 l_5^2e_1^2(k) + \gamma_1 e_2^2(k) + \gamma_1\zeta_3^2(k) + \\
&\quad \gamma_1\left(\varepsilon_{3\mathrm{m}} + w_{3\mathrm{m}}\tilde{\phi}_{3\mathrm{m}} + d_{1\mathrm{m}}\right)^2 - \tfrac{\gamma_1}{5}e_1^2(k)
\end{aligned}
\tag{B.4}
$$

Take the second term, substitute (72), invoke Cauchy-Schwarz inequality, and simplify

$$
\Delta J_2(k) \le 3l_6^2e_2^2(k) + 3g_{2\max}^2\zeta_4^2(k) + \gamma_2\left(d_{2\mathrm{m}} + g_{2\max}\varepsilon_{4\mathrm{m}} + g_{2\max}w_{4\mathrm{m}}\tilde{\phi}_{4\mathrm{m}}\right)^2 - e_2^2(k) \tag{B.5}
$$

Take the third term, substitute (24), invoke Cauchy-Schwarz inequality, and simplify

$$
\begin{aligned}
\Delta J_3(k) &\le -\gamma_3\left(1 - \alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\left(\hat{w}_1(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)^2 + \\
&\quad 2\gamma_3\left(w_{1\mathrm{m}}\hat{\phi}_{1\mathrm{m}}\right)^2 + 2\gamma_3 l_4^2\tilde{y}^2(k) - \gamma_3\zeta_1^2(k)
\end{aligned}
\tag{B.6}
$$

Take the fourth term, substitute (48), invoke Cauchy-Schwarz inequality, and simplify

$$
\begin{aligned}
\Delta J_4(k) &\le -\gamma_4\left(1 - \alpha_2\left\|\hat{\phi}_2(k)\right\|^2\right)\left(\hat{Q}(k) + \beta^{N+1}p(k) - \beta\hat{Q}(k-1)\right)^2 - \gamma_4\zeta_2^2(k) + \\
&\quad 2\gamma_4\beta^2\zeta_2^2(k-1) + 2\gamma_4\left(w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}}(1+\beta) + \beta^{N+1}\right)^2
\end{aligned}
\tag{B.7}
$$

Take the fifth term, substitute (68), invoke Cauchy-Schwarz inequality, and simplify

$$
\begin{aligned}
\Delta J_5(k) &\le -\gamma_5\left(1 - \alpha_3\left\|\hat{\phi}_3(k)\right\|^2\right)\left(\hat{Q}(k) + \hat{w}_3^T(k)\hat{\phi}_3(k)\right)^2 + \\
&\quad 2\gamma_5\zeta_2^2(k) + 2\gamma_5\left(w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}} + w_{3\mathrm{m}}\hat{\phi}_{3\mathrm{m}}\right)^2 - \gamma_5\zeta_3^2(k)
\end{aligned}
\tag{B.8}
$$

Take the sixth term, substitute (79), invoke Cauchy-Schwarz inequality, and simplify

$$\Delta J_6(k) = -\gamma_6\left(1-\alpha_4\left\|\hat{\phi}_4(k)\right\|^2\right)\left(\hat{w}_4^T(k)\hat{\phi}_4(k)+\hat{Q}(k)\right)^2 + 2\gamma_6\left(w_{4\mathrm{m}}\hat{\phi}_{\mathrm{m}}+w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}}\right)^2 + 2\gamma_6\zeta_2^2(k)-\gamma_6\zeta_4^2(k) \tag{B.9}$$

Take the seventh term, set $\gamma_7 = 2\gamma_4\beta^2$

$$\Delta J_7(k) = 2\gamma_4\beta^2\zeta_2^2(k) - 2\gamma_4\beta^2\zeta_2^2(k-1) \tag{B.10}$$

Take the eighth term, substitute (21), invoke Cauchy-Schwarz inequality, and simplify

$$\Delta J_8(k) \le \gamma_8 l_2^2\tilde{y}^2(k) + \gamma_8\tilde{x}_2^2(k) + \gamma_8\left(w_{3\mathrm{m}}\phi_{3\mathrm{m}} + f_{10} + \varepsilon_{3\mathrm{m}} + d_{1\mathrm{m}}\right)^2 - \tfrac{\gamma_8}{3}\tilde{x}_1^2(k) \tag{B.11}$$

Take the ninth term, substitute (22), invoke Cauchy-Schwarz inequality, and simplify

$$\Delta J_9(k) \le \gamma_9\left(f_{20}+\left(g_{20}+g_{2\max}\right)w_{4\mathrm{m}}\hat{\phi}_{4\mathrm{m}} + f_{2\max} + d_{2\mathrm{m}}\right)^2 + \gamma_9\left(g_{20}+g_{2\max}\right)\zeta_4(k) + \gamma_9 l_3^2\tilde{y}^2(k) - \tfrac{\gamma_9}{3}\tilde{x}_2^2(k) \tag{B.12}$$

Take the tenth and final term, substitute (23), invoke Cauchy-Schwarz inequality, and simplify

$$\Delta J_{10}(k) \le \gamma_{10}\zeta_1^2(k) + \gamma_{10}l_1^2\tilde{y}(k) + \gamma_{10}\left(w_{1\mathrm{m}}\tilde{\phi}_{1\mathrm{m}} + \varepsilon_{1\mathrm{m}}\right) - \tfrac{\gamma_{10}}{3}\tilde{y}^2(k) \tag{B.13}$$

Combine (B.4) through (B.13) and simplify to get the first difference of the Lyapunov function

$$\begin{aligned}
\Delta J \le\ & -\left(\tfrac{\gamma_1}{5}-\gamma_1 l_5^2\right)e_1^2(k) - \left(\tfrac{\gamma_2}{3}-\gamma_1-\gamma_2 l_6^2\right)e_2^2(k) \\
& -\left(\gamma_3-\gamma_{10}\right)\zeta_1^2(k) - \left(\tfrac{\gamma_9}{3}-\gamma_8\right)\tilde{x}_2^2(k) - \left(\gamma_5-\gamma_1\right)\zeta_3^2(k) \\
& -\left(\gamma_6-\gamma_2 g_{2\max}^2-\gamma_9\left(g_{20}+g_{2\max}\right)\right)\zeta_4^2(k) - \left(\tfrac{\gamma_8}{3}-\gamma_1 l_5^2\right)\tilde{x}_1^2(k) \\
& -\left(\gamma_4-2\gamma_5-2\gamma_6-2\gamma_4\beta^2\right)\zeta_2^2(k) \\
& -\left(\tfrac{\gamma_{10}}{3}-2\gamma_3 l_4^2-\gamma_8 l_2^2-\gamma_9 l_3^2-\gamma_{10}l_1^2\right)\tilde{y}^2(k) + D_M^2 \\
& -\gamma_3\left(1-\alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\left(\hat{w}_1(k)\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^2 \\
& -\gamma_4\left(1-\alpha_2\left\|\hat{\phi}_2(k)\right\|^2\right)\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)^2 \\
& -\gamma_5\left(1-\alpha_3\left\|\hat{\phi}_3(k)\right\|^2\right)\left(\hat{Q}(k)+\hat{w}_3^T(k)\hat{\phi}_3(k)\right)^2 \\
& -\gamma_6\left(1-\alpha_4\left\|\hat{\phi}_4(k)\right\|^2\right)\left(\hat{w}_4^T(k)\hat{\phi}_4(k)+\hat{Q}(k)\right)^2
\end{aligned} \tag{B.14}$$

where

$$
\begin{aligned}
D_{\mathrm{m}}^2 = {} & \gamma_1\left(\varepsilon_{3\mathrm{m}} + w_{3\mathrm{m}}\tilde{\phi}_{3\mathrm{m}} + d_{1\mathrm{m}}\right)^2 + \gamma_2\left(d_{2\mathrm{m}} + g_{2\max}\varepsilon_{4\mathrm{m}} + g_{2\max}w_{4\mathrm{m}}\tilde{\phi}_{4\mathrm{m}}\right)^2 + \\
& 2\gamma_3\left(w_{1\mathrm{m}}\hat{\phi}_{1\mathrm{m}}\right)^3 + 2\gamma_4\left(w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}}\left(1+\beta\right) + \beta^{N+1}\right)^2 + 2\gamma_5\left(w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}} + w_{3\mathrm{m}}\hat{\phi}_{3\mathrm{m}}\right)^2 + \\
& 2\gamma_6\left(w_{4\mathrm{m}}\hat{\phi}_{\mathrm{m}} + w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}}\right)^2 + \gamma_8\left(w_{3\mathrm{m}}\phi_{3\mathrm{m}} + f_{10} + \varepsilon_{3\mathrm{m}} + d_{1\mathrm{m}}\right)^2 + \\
& \gamma_9\left(f_{20} + \left(g_{20} + g_{2\max}\right)w_{4\mathrm{m}}\hat{\phi}_{4\mathrm{m}} + f_{2\max} + d_{2\mathrm{m}}\right)^2 + \gamma_{10}\left(w_{1\mathrm{m}}\tilde{\phi}_{1\mathrm{m}} + \varepsilon_{1\mathrm{m}}\right)^2
\end{aligned}
\tag{B.15}
$$

Select

$$
\begin{aligned}
& \gamma_1 > 5\gamma_1 l_5^2;\ \ \gamma_2 > 3\gamma_1 + 3\gamma_2 l_6^2;\ \ \gamma_3 > \gamma_{10};\ \ \gamma_4 > 2\gamma_5 + 2\gamma_6 + 2\gamma_4\beta^2;\ \ \gamma_5 > \gamma_1; \\
& \gamma_6 > \gamma_2 g_{2\max}^2 + \gamma_9\left(g_{20} + g_{2\max}\right);\ \ \gamma_7 = 2\gamma_4\beta^2;\ \ \gamma_8 > 3\gamma_1 l_5^2;\ \ \gamma_9 > 3\gamma_8; \\
& \gamma_{10} > 6\gamma_3 l_4^2 + 3\gamma_8 l_2^2 + 3\gamma_9 l_3^2 + 3\gamma_{10} l_1^2;
\end{aligned}
\tag{B.16}
$$

This implies $\Delta J(k) < 0$ as long as (82) through (92) hold *and* any *one* of the following hold

$$
\begin{aligned}
& \left|e_1(k)\right| > \frac{D_M}{\sqrt{\frac{\gamma_1}{5} - \gamma_1 l_5^2}};\ \ \left|e_2(k)\right| > \frac{D_M}{\sqrt{\frac{\gamma_2}{3} - \gamma_1 - \gamma_2 l_6^2}};\ \ \left|\zeta_1(k)\right| > \frac{D_M}{\sqrt{\gamma_3 - \gamma_{10}}}; \\
& \left|\zeta_3(k)\right| > \frac{D_M}{\sqrt{\gamma_5 - \gamma_1}};\ \left|\zeta_2(k)\right| > \frac{D_M}{\sqrt{\gamma_4 - 2\gamma_5 - 2\gamma_6 - 2\gamma_4\beta^2}}; \\
& \left|\zeta_4(k)\right| > \frac{D_M}{\sqrt{\gamma_6 - \gamma_2 g_{2\max}^2 - \gamma_9\left(g_{20} + g_{2\max}\right)}};\ \left|\tilde{x}_1(k)\right| > \frac{D_M}{\sqrt{\frac{\gamma_8}{3} - \gamma_1 l_5^2}}; \\
& \left|\tilde{x}_2(k)\right| > \frac{D_M}{\sqrt{\frac{\gamma_9}{3} - \gamma_8}};\ \ \left|\tilde{y}(k)\right| > \frac{D_M}{\sqrt{\frac{\gamma_{10}}{3} - 2\gamma_3 l_4^2 - \gamma_8 l_2^2 - \gamma_9 l_3^2 - \gamma_{10} l_1^2}};
\end{aligned}
\tag{B.17}
$$

APPENDIX C

Proof of Theorem 2: Define the Lyapunov function

$$
J(k) = \sum_{k=11}^{14} J_k = \frac{\gamma_{11}}{3}\left\|e_4(k)\right\|^2 + \frac{\gamma_{12}}{\alpha_4}tr\left(\tilde{w}_a^T(k)\tilde{w}_a(k)\right) + \frac{\gamma_{13}}{\alpha_c}tr\left(\tilde{w}_c^T(k)\tilde{w}_c(k)\right) + \gamma_{14}\left\|\zeta_c(k-1)\right\|^2 \tag{C.1}
$$

where $0 < \gamma_k, k \in \{11,...,14\}$ are auxiliary constants; the NN weights estimation errors $\tilde{w}_a^T(k+1)$ and $\tilde{w}_c^T(k+1)$ are defined in (114) and (105), by subtracting their respective ideal weights $w_i, i \in \{a,c\}$ on both sides; the system error $e_4(k+1)$ is defined in (109); and $\alpha_i, i \in \{a,c\}$ are NN adaptation gains. The Lyapunov function (C.1) obviates the need for the CE condition. Take the first term and replace (109).

$$\begin{aligned}\Delta J_{11}(k) &= \tfrac{\gamma_{11}}{3}\left(\left\|e_4(k+1)\right\|^2 - \left\|e_4(k)\right\|^2\right) \\ &\le -\tfrac{\gamma_{11}}{3}\left(1-3l_{\max}^2\right)\left\|e_4(k)\right\|^2 + \gamma_{11}g_{\max}^2\left\|\zeta_a(k)\right\|^2 + \gamma_{11}\left\|d_a(k)\right\|^2\end{aligned} \tag{C.2}$$

Take the second term and replace (114)

$$\begin{aligned}\Delta J_{12}(k) &= \tfrac{\gamma_{12}}{\alpha_a} tr\left(\tilde{w}_a^T(k+1)\tilde{w}_a(k+1) - \tilde{w}_a^T(k)\tilde{w}_a(k)\right) \\ &\le \gamma_{12}\begin{pmatrix} -g_{\min}\left\|\zeta_a(k)\right\|^2 + \dfrac{1-\alpha_a\left\|\phi_a(k)\right\|^2 g_{\min}}{g_{\min}-\alpha_a\left\|\phi_a(k)\right\|^2 g_{\max}^2}\left\|J(k)+d_a(k)\right\|^2 \\ -\left(g_{\min}-\alpha_a\left\|\phi_a(k)\right\|^2 g_{\max}^2\right)\left(\zeta_a(k)+\dfrac{I-\alpha_\alpha\left\|\phi_a(k)\right\|^2 g_4(\cdot)}{g_{\min}-\alpha_a\left\|\phi_a(k)\right\|^2 g_{\max}^2}\right)\end{pmatrix}\end{aligned} \tag{C.3}$$

Define the following

$$\gamma_{12} = \gamma_{12}^{'}\gamma_{12}^{''} = \gamma_{12}^{'}\frac{1-\alpha_a\left\|\phi_a(k)\right\|^2 g_{\min}}{g_{\min}-\alpha_a\left\|\phi_a(k)\right\|^2 g_{\max}^2} \le \frac{1}{2}\gamma_{12}^{'} \tag{C.4}$$

Rewrite (C.3) using (C.4)

$$\begin{aligned}\Delta J_{12} \le &-\gamma_{12}\left(g_{\min}-\alpha_a\left\|\phi_a(k)\right\|^2 g_{\max}^2\right)\left(\zeta_a(k)+\frac{I-\alpha_\alpha\left\|\phi_a(k)\right\|^2 g_4(\cdot)}{g_{\min}-\alpha_a\left\|\phi_a(k)\right\|^2 g_{\max}^2}\right)+ \\ &\gamma_{12}^{'}n\left\|\zeta_c(k)\right\|^2 + \gamma_{12}^{'}n\left\|d_a(k)\right\|^2 - \gamma_{12}g_{\min}\left\|\zeta_a(k)\right\|^2\end{aligned} \tag{C.5}$$

Take the third term and replace (105)

$$\begin{aligned}\Delta J_{13} &= \frac{\gamma_{12}}{\alpha_c} tr\left(\tilde{w}_c^T(k+1)\tilde{w}_c(k+1) - \tilde{w}_c^T(k)\tilde{w}_c(k)\right) \\ &\le -\gamma_{13}\left(1-\alpha_c\gamma^2\left\|\phi_a(k)\right\|^2\right)e_c^2(k) - \gamma_{13}\gamma^2\zeta_c^2(k) + \frac{\gamma_{13}}{4}\zeta_c^2(k-1) + \\ &\gamma_{13}Q_{\max}\left\|e(k)\right\|^2 + \frac{\gamma_{13}}{8}R_{\max}\left\|\zeta_a(k)\right\|^2 + \frac{\gamma_{13}}{8}R_{\max}\left\|w_a^T\phi_a(k)\right\|^2 + \gamma_{13}\varepsilon_{cm}^2\end{aligned} \tag{C.6}$$

Take the forth and final term and replace

$$\Delta J_{14} = \gamma_{14}\left(\left\|\zeta_c(k)\right\|^2 - \left\|\zeta_c(k-1)\right\|^2\right) \tag{C.7}$$

Combine (C.2) through (C.7)

$$\Delta J = -\tfrac{\gamma_{11}}{3}\left(1-3l_{\max}^2\right)\left\|e_4(k)\right\|^2 + \gamma_{11}g_{\max}^2\left\|\zeta_a(k)\right\|^2 + \gamma_{11}\left\|d_a(k)\right\|^2 - \gamma_{12}g_{\min}\left\|\zeta_a(k)\right\|^2 +$$
$$\gamma_{12}\frac{1-\alpha_a\left\|\phi_a(k)\right\|^2 g_{\min}}{g_{\min}-\alpha_a\left\|\phi_a(k)\right\|^2 g_{\max}^2}\left\|J(k)+d_a(k)\right\|^2 - \gamma_{13}\left(1-\alpha_c\gamma^2\left\|\phi_a(k)\right\|^2\right)e_c^2(k) -$$
$$\gamma_{12}\left(g_{\min}-\alpha_a\left\|\phi_a(k)\right\|^2 g_{\max}^2\right)\left(\zeta_a(k)+\frac{I-\alpha_\alpha\left\|\phi_a(k)\right\|^2 g_4(\cdot)}{g_{\min}-\alpha_a\left\|\phi_a(k)\right\|^2 g_{\max}^2}\right) - \qquad \text{(C.8)}$$
$$\gamma_{13}\gamma^2\zeta_c^2(k) + \frac{\gamma_{13}}{4}\zeta_c^2(k-1) + \gamma_{13}Q_{\max}\left\|e(k)\right\|^2 + \frac{\gamma_{13}}{8}R_{\max}\left\|\zeta_a(k)\right\|^2 +$$
$$\gamma_{14}\left(\left\|\zeta_c(k)\right\|^2 - \left\|\zeta_c(k-1)\right\|^2\right) + D_M^2$$

where

$$D_M^2 = \frac{\gamma_{12}^{'}n}{2}d_a^2(k) + \left(\frac{\gamma_{13}}{4}+\frac{\gamma_{12}^{'}n}{2}\right)J_m^2 + \frac{\gamma_{13}}{6}R_{\max}\left\|w_a^T\phi_a(k)\right\|^2 + \gamma_{11}\left\|d_a(k)\right\|^2 + \gamma_{13}\varepsilon_{cm}^2 \qquad \text{(C.9)}$$

The overall Lyapunov function is negative as long as (115) and (117) holds *and one* of the following is true.

$$\left\|e(k)\right\| \geq \frac{2\sqrt{3}D_M}{\sqrt{4\gamma_{11}\left(1-3l_{\max}^2\right)-3\gamma_3 Q_{\max}}} \qquad \text{(C.10)}$$

$$\left\|\zeta_a(k)\right\| \leq \frac{2\sqrt{2}D_M}{\sqrt{8\gamma_{12}g_{\min}-8\gamma_{11}g_{\max}^2-\gamma_{13}R_{\max}}} \qquad \text{(C.11)}$$

$$\left\|\zeta_c(k)\right\| \leq \frac{D_M}{\sqrt{\gamma_{13}\gamma^2-\gamma_{12}^{'}n-\gamma_{14}}} \qquad \text{(C.12)}$$

REFERENCES

[1] M. Krstic, I. Kanellakopoulos, and P. Kokotovic, Nonlinear and Adaptive Control Design: John Wiley & Sons, Inc, 1995.

[2] S. S. Ge, T. H. Lee, G. Y. Li, and J. Zhang, "Adaptive NN control for a class of discrete-time nonlinear systems," Int. J. Contr., vol. 76, pp. 334-354, 2003.

[3] F. C. Chen and H. K. Khalil, "Adaptive control of a class of nonlinear discrete-time systems using neural networks," IEEE Trans. Automat. Contr, vol. 40, pp. 791-801, 1995.

[4] J. Si, in NSF Workshop on Learning and Approximate Dynamic Programming, Playacar, Mexico, 2002.

[5] P. J. Werbos, Neurocontrol and supervised learning: An overview and evaluation. New York: Van Nostrand Reinhold, 1992.

[6] J. J. Murray, C. Cox, G. G. Lendaris, and R. Saeks, "Adaptive dynamic programming," IEEE Trans. Syst., Man, Cybern., vol. 32, pp. 140-153, 2002.

[7] D. P. Bertsekas and J. N. Tsitsiklis, Neuro-Dynamic Programming. Balmont, MA: Athena Scientific, 1996.

[8] J. Si and Y. T. Wang, "On-line learning control by association and reinforcement," IEEE Trans. on Neural Networks, vol. 12, pp. 264-276, 2001.

[9] X. Lin and S. N. Balakrishnan, "Convergence analysis of adaptive ciritc based optimal control," Proceedings of the American Control Conference, vol. 12, 264-276 2000.

[10] F. L. Lewis, S. Jagannathan, and A. Yesilderek, Neural Network control of robot manipulators and nonlinear systems. UK: Taylor and Francis, 1999.

[11] J. Vance, P. He, S. Jagannathan, and J. Drallmeier, "Neural Network-based Output Feedback Controller for Lean Operation of Spark Ignition Engine," in American Control Conference, Portland, OR, 2006.

[12] N. Hovakimyan, F. Nardi, A. Calise, and N. Kim, "Adaptive output feedback control of uncertain nonlinear systems using single-hidden-layer neural networks," IEEE Trans. on Neural Networks, vol. 13, pp. 1420-1431, 2002.

[13] A. N. Atassi and H. K. Khalil, "A separation principle for the stabilization of a class of nonlinear systems," IEEE Trans. Automat. Contr, vol. 76, pp. 334-354, 2003.

[14] B. Igelruk and Y. H. Pao, "Stochastic choice of basis functions in adaptive function approximation and the functional-link net," IEEE Trans. Neural Networks, vol. 6, pp. 1320-1329, 1995.

[15] S. Jagannathan, Neural Network Control of Nonlinear Discrete-time Systems. London, UK: Taylor and Francis, 2006.

[16] C. S. Daw, C. E. A. Finney, M. B. Kennel, F. T. Connolly, "Observing and Modeling Nonlinear Dynamics in an Internal Combustion Engine," Phys. Rev. E, vol. 57, pp. 2811-2819, 1998.

[17] R. W. Sutton and J. A. Drallmeier, "Development of nonlinear cyclic dispersion in spark ignition engines under teh influence of high levels of EGR," in Proc. of the Central States Section of the Combustion Institute, Indianapolis, Indiana, 2000, pp. 175-180.

[18] J. B. Vance, A. Singh, B. Kaul, S. Jagannathan, and J. Drallmeier, "Neural Network Controller Development and Implementation for Spark Ignition Engines with High EGR Levels," IEEE Tans. Neural Networks, 2006.

APPENDIX

DETAILED LYAPUNOV PROOF

Consider the nonlinear discrete-time system given by

$$\begin{aligned} x_1(k+1) &= f_1\left(x_1(k), x_2(k)\right) + g_1\left(x_1(k), x_2(k)\right)x_2(k) + d_1(k) \\ &= f_1(\cdot) + g_1(\cdot)x_2(k) + d_1(k) \end{aligned} \tag{1}$$

$$\begin{aligned} x_2(k+1) &= f_2\left(x_1(k), x_2(k)\right) + g_2\left(x_1(k), x_2(k)\right)u(k) + d_2(k) \\ &= f_2(\cdot) + g_2(\cdot)u(k) + d_2(k) \end{aligned} \tag{2}$$

$$y(k+1) = f_3\left(x_1(k), x_2(k)\right) = f_3(\cdot) \tag{3}$$

Define the estimation errors $e_1(k+1)$ and $e_2(k+1)$ as

$$e_1(k+1) = -\zeta_3(k) - w_3^T\tilde{\phi}_3(k) + \varepsilon_3(k) - l_5\hat{e}_1(k) - e_2(k) + d_1(k) \tag{4}$$

$$e_2(k+1) = l_6e_2(k) - g_2(\cdot)\varepsilon_4(k) + g_2(\cdot)\zeta_4(k) + g_2(\cdot)w_4^T\tilde{\phi}_4(k) + d_2(k) \tag{5}$$

where

$$\hat{e}_1(k) = \hat{x}_1(k) - x_{1d}(k) \tag{6}$$

$$\hat{e}_2(k) = \hat{x}_2(k) - \hat{x}_{2d}(k) \tag{7}$$

$$e_1(k) = x_1(k) - x_{1d}(k) \tag{8}$$

$$e_2(k) = x_2(k) - \hat{x}_{2d}(k) \tag{9}$$

$$\varepsilon_i(k) = \varepsilon\left(v_i\hat{z}_i(k)\right), i \in \{1,2,3,4\} \tag{10}$$

$$\hat{\phi}_i(k) = \phi\left(v_i\hat{z}_i(k)\right), i \in \{1,2,3,4\} \tag{11}$$

$$\phi_i(k) = \phi\left(v_iz_i(k)\right), i \in \{1,2,3,4\} \tag{12}$$

$$\tilde{\phi}_i(k) = \phi\left(v_i\hat{z}_i(k)\right) - \phi\left(v_iz_i(k)\right), i \in \{1,2,3,4\} \tag{13}$$

$$\zeta_i(k) = \hat{w}_i^T(k)\phi\left(v_i\hat{z}_i(k)\right) - w_i^T\phi\left(v_i\hat{z}_i(k)\right) = \tilde{w}_i^T(k)\hat{\phi}_i(k), i \in \{1,2,3,4\} \tag{14}$$

$$0 < l_i < 1, i \in \{1,2,3,4,5,6\} \tag{15}$$

$$\left|d_i(k)\right| < d_{i,\max}, i \in \{1,2\} \tag{16}$$

$$\begin{aligned}\hat{e}_1(k) &= \hat{e}_1(k) + e_1(k) - e_1(k) \\ &= \left(\hat{x}_1(k) - x_{1d}(k) - x_1(k) + x_{1d}(k)\right) + e_1(k) \\ &= \tilde{x}_1(k) + e_1(k)\end{aligned} \tag{17}$$

Define the NN weight estimation errors $\tilde{w}_1(k+1)$, $\tilde{w}_2(k+1)$, $\tilde{w}_3(k+1)$, and $\tilde{w}_4(k+1)$ as

$$\hat{w}_1(k+1) = \hat{w}_1(k) - \alpha_1 \hat{\phi}_1(k)\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right) \tag{18}$$

$$\tilde{w}_2(k+1) = \tilde{w}_2(k) - \alpha_2 \hat{\phi}_2(k)\left(\hat{Q}(k) + \beta^{N+1}p(k) - \beta\hat{Q}(k-1)\right)^T \tag{19}$$

$$\tilde{w}_3(k+1) = \tilde{w}_3(k) - \alpha_3 \hat{\phi}_3(k)\left(\hat{Q}(k) + \hat{w}_3^T(k)\hat{\phi}_3(k)\right) \tag{20}$$

$$\hat{w}_4(k+1) = \hat{w}_4(k) - \alpha_4 \hat{\phi}_4(k)\left(\hat{w}_4^T(k)\hat{\phi}_4(k) + \hat{Q}(k)\right) \tag{21}$$

where

$$0 < \alpha_i < 1, i \in \{1,2,3,4\} \tag{22}$$

$$0 < \beta < 1 \tag{23}$$

$$\hat{Q}(k) = \hat{w}_2^T(k)\hat{\phi}_2(k) = \zeta_2(k) + w_2^T\hat{\phi}_2(k) \tag{24}$$

$$\tilde{y}(k) = \hat{y}(k) - y(k) \tag{25}$$

Define the state estimation errors $\tilde{x}_1(k+1)$ and $\tilde{x}_2(k+1)$ as

$$\tilde{x}_1(k+1) = \hat{x}_1(k+1) - x_1(k+1) = f_{10} - \hat{x}_2(k) + l_2\tilde{y}(k) - f_1(\cdot) - g_1(\cdot)x_2(k) - d_1(k) \tag{26}$$

$$\tilde{x}_2(k+1) = \hat{x}_2(k+1) - x_2(k+1) = f_{20} + g_{20}u(k) + l_3\tilde{y}(k) - f_2(\cdot) - g_2(\cdot)u(k) - d_2(k) \tag{27}$$

where

$$f_1(\cdot) = f_{10} + \Delta f_1(\cdot) \tag{28}$$

$$f_2(\cdot) = f_{20} + \Delta f_2(\cdot) \tag{29}$$

$$g_1(\cdot) = g_{10} + \Delta g_1(\cdot) \tag{30}$$

$$g_2(\cdot) = g_{20} + \Delta g_2(\cdot) \tag{31}$$

Define the following equation

$$\begin{aligned}\Phi(\cdot) &= f_1(\cdot) + g_1(\cdot)x_2(k) + x_2(k) = w_3^T\phi_3(k) + \varepsilon_3(k) \\ f_1(\cdot) &= w_3^T\phi_3(k) + \varepsilon_3(k) - g_1(\cdot)x_2(k) - x_2(k)\end{aligned} \tag{32}$$

Define the output error as

$$\tilde{y}(k+1) = \hat{w}_1^T(k)\hat{\phi}_1(k) + l_1\tilde{y}(k) - w_1^T\phi_1(k) - \varepsilon_1(k) \tag{33}$$

Define the Lyapunov function as

$$\begin{aligned} J(k) = \sum_{i=1}^{10} J_i(k) = \frac{\gamma_1}{5}e_1^2(k) + \frac{\gamma_2}{3}e_2^2(k) + \sum_{j=3}^{6}\frac{\gamma_j}{\alpha_{j-2}}\tilde{w}_j^T(k)\tilde{w}_j(k) + \\ \gamma_7\zeta_2^2(k-1) + \frac{\gamma_8}{3}\tilde{x}_1^2(k) + \frac{\gamma_9}{3}\tilde{x}_2^2(k) + \frac{\gamma_{10}}{3}\tilde{y}^2 \end{aligned} \tag{34}$$

where $0 < \gamma_j, j \in \{1,...,10\}$ are auxiliary constants; the system errors $e_1(k+1)$ and $e_2(k+1)$ are defined in (4) and (5), respectively; the NN weights estimation errors $\tilde{w}_1(k+1)$, $\tilde{w}_2(k+1)$, $\tilde{w}_3(k+1)$, and $\tilde{w}_3(k+1)$ are defined in (18), (19), (20), and (21), respectively; the observation errors $\tilde{x}_1(k+1)$, $\tilde{x}_2(k+1)$, are defined in (26), and (27), respectively; the output error $\tilde{y}(k+1)$ is defined in (56), and $\alpha_j, j \in \{1,2,3,4\}$ are NN adaptation gains. The Lyapunov function (34) obviates the need for separation principle. Define the Cauchy-Schwarz inequality.

$$(a_1b_1 + ... + a_nb_n)^2 \le (a_1^2 + ... + a_n^2)(b_1^2 + ... + b_n^2) \tag{35}$$

Take the first Lyapunov term and the first difference then substitute (4) and (17).

$$\begin{aligned} J_1(k) &= \tfrac{\gamma_1}{5}e_1^2(k) \\ \tfrac{5}{\gamma_1}\Delta J_1(k) &= e_1^2(k+1) - e_1^2(k) \\ &= \left(-\zeta_3(k) - w_3^T\tilde{\phi}_3(k) + \varepsilon_3(k) - l_5\hat{e}_1(k) - e_2(k) + d_1(k)\right)^2 - e_1^2(k) \\ &= \left(-\zeta_3(k) - w_3^T\tilde{\phi}_3(k) + \varepsilon_3(k) - l_5\left(\tilde{x}_1(k) + e_1(k)\right) - e_2(k) + d_1(k)\right)^2 - e_1^2(k) \end{aligned} \tag{36}$$

Invoke Cauchy-Schwarz inequality and simplify.

$$\begin{aligned} \tfrac{1}{\gamma_1}\Delta J_1(k) &\le \left(\zeta_3^2(k) + l_5^2\tilde{x}_1^2(k) + l_5^2e_1^2(k) + e_2^2(k) + \left(\varepsilon_3(k) - w_3\tilde{\phi}_3(k) + d_1(k)\right)^2\right) - \tfrac{1}{5}e_1^2(k) \\ \Delta J_1(k) &\le \gamma_1\zeta_3^2(k) + \gamma_1l_5^2\tilde{x}_1^2(k) + \gamma_1l_5^2e_1^2(k) + \gamma_1e_2^2(k) + \gamma_1\begin{pmatrix}\varepsilon_3(k) - w_3\tilde{\phi}_3(k) \\ + d_1(k)\end{pmatrix}^2 - \tfrac{\gamma_1}{5}e_1^2(k) \\ &\le \gamma_1l_5^2\tilde{x}_1^2(k) + \gamma_1l_5^2e_1^2(k) + \gamma_1e_2^2(k) + \gamma_1\zeta_3^2(k) + \gamma_1\begin{pmatrix}\varepsilon_{3\mathrm{m}} + w_{3\mathrm{m}}\tilde{\phi}_{3\mathrm{m}} \\ + d_{1\mathrm{m}}\end{pmatrix}^2 - \tfrac{\gamma_1}{5}e_1^2(k) \\ &\le \gamma_1l_5^2\tilde{x}_1^2(k) + \gamma_1l_5^2e_1^2(k) + \gamma_1e_2^2(k) + \gamma_1\zeta_3^2(k) + \gamma_1D_1^2 - \tfrac{\gamma_1}{5}e_1^2(k) \end{aligned} \tag{37}$$

where

$$D_1 = \varepsilon_{3\mathrm{m}} + w_{3\mathrm{m}}\tilde{\phi}_{3\mathrm{m}} + d_{1\mathrm{m}} \tag{38}$$

The subscript *m* stands for the maximum value. Take the second term and use the same procedure as above then substitute (5) to get

$$\begin{aligned}
J_2(k) &= \tfrac{\gamma_2}{3} e_2^2(k) \\
\tfrac{3}{\gamma_2}\Delta J_2(k) &= e_2^2(k+1) - e_2^2(k) \\
&= \left(l_6 e_2(k) - g_2(\cdot)\varepsilon_4(k) + g_2(\cdot)\zeta_4(k) + g_2(\cdot) w_4^T \tilde{\phi}_4(k) + d_2(k)\right)^2 - e_2^2(k) \\
&\le 3l_6^2 e_2^2(k) + 3g_{2\max}^2 \zeta_4^2(k) + 3\left(d_2(k) - g_{2\max}\varepsilon_4(k) + g_2(\cdot) w_4^T \tilde{\phi}_4(k)\right)^2 - e_2^2(k) \\
&\le 3l_6^2 e_2^2(k) + 3g_{2\max}^2 \zeta_4^2(k) + 3\left(d_{2\mathrm{m}} + g_{2\max}\varepsilon_{4\mathrm{m}} + g_{2\max} w_{4\mathrm{m}}\tilde{\phi}_{4\mathrm{m}}\right)^2 - e_2^2(k) \\
\Delta J_2(k) &\le \gamma_2 l_6^2 e_2^2(k) + \gamma_2 g_{2\max}^2 \zeta_4^2(k) + \gamma_2 D_2^2 - \tfrac{\gamma_2}{3} e_2^2(k)
\end{aligned} \tag{39}$$

where

$$D_2 = d_{2\mathrm{m}} + g_{2\max}\varepsilon_{4\mathrm{m}} + g_{2\max} w_{4\mathrm{m}}\tilde{\phi}_{4\mathrm{m}} \tag{40}$$

Take the third term and substitute (18) to get

$$\begin{aligned}
J_3(k) &= \tfrac{\gamma_3}{\alpha_1}\tilde{w}_1^T(k)\tilde{w}_1(k) \\
\tfrac{\alpha_1}{\gamma_3}\Delta J_3(k) &= \tilde{w}_1^T(k+1)\tilde{w}_1(k+1) - \tilde{w}_1^T(k)\tilde{w}_1(k) \\
&= [\tilde{w}_1(k) - \alpha_1\hat{\phi}_1(k)\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)]^T * \\
&\quad [\tilde{w}_1(k) - \alpha_1\hat{\phi}_1(k)\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)] - \tilde{w}_1^T(k)\tilde{w}_1(k) \\
&= [\tilde{w}_1^T(k) - \alpha_1\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)^T \hat{\phi}_1^T(k)] * \\
&\quad [\tilde{w}_1(k) - \alpha_1\hat{\phi}_1(k)\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)] - \tilde{w}_1^T(k)\tilde{w}_1(k) \\
&= \tilde{w}_1^T(k)\tilde{w}_1(k) + \alpha_1^2 \begin{pmatrix} \hat{w}_1^T(k)\hat{\phi}_1(k) \\ +l_4\tilde{y}(k) \end{pmatrix}^T \hat{\phi}_1^T(k)\hat{\phi}_1(k) \begin{pmatrix} \hat{w}_1^T(k)\hat{\phi}_1(k) \\ +l_4\tilde{y}(k) \end{pmatrix} - \\
&\quad \alpha_1\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)^T \hat{\phi}_1^T(k)\tilde{w}_1^T(k) - \\
&\quad \alpha_1\tilde{w}_1(k)\hat{\phi}_1(k)\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right) - \tilde{w}_1^T(k)\tilde{w}_1(k) \\
&= \alpha_1^2 \left\|\phi_1(k)\right\|^2 \begin{pmatrix} \hat{w}_1^T(k)\hat{\phi}_1(k) \\ +l_4\tilde{y}(k) \end{pmatrix}^2 - 2\alpha_1\tilde{w}_1(k)\hat{\phi}_1(k) \begin{pmatrix} \hat{w}_1^T(k)\hat{\phi}_1(k) \\ +l_4\tilde{y}(k) \end{pmatrix} + \\
&\quad \alpha_1\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)^2 - \alpha_1\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_4\tilde{y}(k)\right)^2
\end{aligned}$$

$$\begin{aligned}
\tfrac{\alpha_1}{\gamma_3}\Delta J_3(k) &= -\alpha_1\left(1-\alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\left(\hat{w}_1^T(k)\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^2 - \\
&\quad 2\alpha_1\zeta_1(k)\left(\zeta_1(k)+w_1^T\hat{\phi}_1+l_4\tilde{y}(k)\right)+\left(\zeta_1(k)+w_1^T\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^2 \\
&= -\alpha_1\left(1-\alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\left(\hat{w}_1^T(k)\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^2 + \\
&\quad \alpha_1\left(\left(\zeta_1(k)+w_1^T\hat{\phi}_1+l_4\tilde{y}(k)\right)-\zeta_1(k)\right)^2-\alpha_1\zeta_1^2(k) \\
&= -\alpha_1\left(1-\alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\begin{pmatrix}\hat{w}_1^T(k)\hat{\phi}_1(k)\\ +l_4\tilde{y}(k)\end{pmatrix}^2+\alpha_1\left(w_1^T\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^2-\alpha_1\zeta_1^2(k) \\
\Delta J_3(k) &\le -\gamma_3\left(1-\alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\left(\hat{w}_1^T(k)\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^2 + \\
&\quad 2\gamma_3\left(w_1\hat{\phi}_1(k)\right)^2+2\gamma_3 l_4^2\tilde{y}^2(k)-\gamma_3\zeta_1^2(k) \\
&\le -\gamma_3\left(1-\alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\begin{pmatrix}\hat{w}_1^T(k)\hat{\phi}_1(k)\\ +l_4\tilde{y}(k)\end{pmatrix}^2+2\gamma_3 D_3^2+2\gamma_3 l_4^2\tilde{y}^2(k)-\gamma_3\zeta_1^2(k)
\end{aligned} \tag{41}$$

where

$$D_3 = w_{1\mathrm{m}}\hat{\phi}_{1\mathrm{m}} \tag{42}$$

Take the fourth term and substitute (19) and (24)

$$\begin{aligned}
J_4(k) &= \tfrac{\gamma_4}{\alpha_2}\tilde{w}_2^T(k)\tilde{w}_2(k) \\
\tfrac{\alpha_2}{\gamma_4}\Delta J_4(k) &= \tilde{w}_2^T(k+1)\tilde{w}_2(k+1)-\tilde{w}_2^T(k)\tilde{w}_2(k) \\
&= [\tilde{w}_2(k)-\alpha_2\hat{\phi}_2(k)\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)^T]^T * \\
&\quad [\tilde{w}_2(k)-\alpha_2\hat{\phi}_2(k)\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)^T]-\tilde{w}_2^T(k)\tilde{w}_2(k) \\
&= \tilde{w}_2^T(k)\tilde{w}_2(k)+\alpha_2^2\left\|\hat{\phi}_2(k)\right\|^2\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)^2 + \\
&\quad \alpha_2\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)^2-\alpha_2\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)^2 \\
&\quad -2\alpha_2\tilde{w}_2^T(k)\hat{\phi}_2(k)\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)-\tilde{w}_2^T(k)\tilde{w}_2(k) \\
&= -\alpha_2\left(1-\alpha_2\left\|\hat{\phi}_2(k)\right\|^2\right)\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)^2 - \\
&\quad 2\alpha_2\zeta_2(k)\left(\zeta_2(k)+w_2^T\hat{\phi}_2(k)+\beta^{N+1}p(k)-\beta\zeta_2(k-1)-\beta w_2^T\hat{\phi}_2(k-1)\right)+ \\
&\quad \alpha_2\left(\zeta_2(k)+w_2^T\hat{\phi}_2(k)+\beta^{N+1}p(k)-\beta\zeta_2(k-1)-\beta w_2^T\hat{\phi}_2(k-1)\right)^2
\end{aligned}$$

$$\frac{1}{\gamma_4}\Delta J_4(k) = -\left(1-\alpha_2\left\|\hat{\phi}_2(k)\right\|^2\right)\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)^2 +$$
$$\left(w_2^T\hat{\phi}_2(k)+\beta^{N+1}p(k)-\beta\zeta_2(k-1)-\beta w_2^T\hat{\phi}_2(k-1)\right)^2-\zeta_2^2(k)$$
$$\Delta J_4(k) \le -\gamma_4\left(1-\alpha_2\left\|\hat{\phi}_2(k)\right\|^2\right)\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)^2 -$$
$$\gamma_4\zeta_2^2(k)+2\gamma_4\beta^2\zeta_2^2(k-1)+2\gamma_4\left(w_2^T\left(\hat{\phi}_2(k)-\beta\hat{\phi}_2(k-1)\right)+\beta^{N+1}p(k)\right)^2$$
$$\le -\gamma_4\left(1-\alpha_2\left\|\hat{\phi}_2(k)\right\|^2\right)\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)^2-\gamma_4\zeta_2^2(k)+$$
$$2\gamma_4\beta^2\zeta_2^2(k-1)+2\gamma_4\left(w_{2\mathrm{m}}\left(\hat{\phi}_{2\mathrm{m}}-\beta\hat{\phi}_{2\mathrm{m}}\right)+\beta^{N+1}\right)^2$$
$$\le -\gamma_4\left(1-\alpha_2\left\|\hat{\phi}_2(k)\right\|^2\right)\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)^2 -$$
$$\gamma_4\zeta_2^2(k)+2\gamma_4\beta^2\zeta_2^2(k-1)+2\gamma_4 D_4^2 \qquad (43)$$

where

$$D_4 = w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}}(1+\beta)+\beta^{N+1} \qquad (44)$$

Take the fifth term and substitute (20) and (24)

$$\Delta J_5(k) = \frac{\gamma_5}{\alpha_3}\tilde{w}_3^T(k)\tilde{w}_3(k)$$
$$\frac{\alpha_3}{\gamma_5}\Delta J_5(k) = \tilde{w}_3^T(k+1)\tilde{w}_3(k+1)-\tilde{w}_3^T(k)\tilde{w}_3(k)$$
$$= \left[\tilde{w}_3(k)-\alpha_3\hat{\phi}_3(k)\left(\hat{Q}(k)+\hat{w}_3^T(k)\hat{\phi}_3(k)\right)\right]^T *$$
$$\left[\tilde{w}_3(k)-\alpha_3\hat{\phi}_3(k)\left(\hat{Q}(k)+\hat{w}_3^T(k)\hat{\phi}_3(k)\right)\right]-\tilde{w}_3^T(k)\tilde{w}_3(k)$$
$$= \alpha_3^2\left\|\hat{\phi}_3(k)\right\|^2\begin{pmatrix}\hat{Q}(k)+\\ \hat{w}_3^T(k)\hat{\phi}_3(k)\end{pmatrix}^2-2\alpha_3\tilde{w}_3(k)\hat{\phi}_3(k)\begin{pmatrix}\hat{Q}(k)+\\ \hat{w}_3^T(k)\hat{\phi}_3(k)\end{pmatrix}$$
$$= \alpha_3^2\left\|\hat{\phi}_3(k)\right\|^2\left(\hat{Q}(k)+\hat{w}_3^T(k)\hat{\phi}_3(k)\right)^2 -$$
$$2\alpha_3\zeta_3(k)\left(\left(\zeta_2(k)+w_2^T\hat{\phi}_2(k)\right)+\left(\zeta_3(k)+w_3^T\hat{\phi}_3(k)\right)\right)+$$
$$\alpha_3\left(\hat{Q}(k)+\hat{w}_3^T(k)\hat{\phi}_3(k)\right)^2-\alpha_3\left(\hat{Q}(k)+\hat{w}_3^T(k)\hat{\phi}_3(k)\right)^2$$
$$\frac{1}{\gamma_5}\Delta J_5(k) = -\left(1-\alpha_3\left\|\hat{\phi}_3(k)\right\|^2\right)\left(\hat{Q}(k)+\hat{w}_3^T(k)\hat{\phi}_3(k)\right)^2 +$$
$$\left(\left(\zeta_2(k)+w_2^T\hat{\phi}_2(k)+\zeta_3(k)+w_3^T\hat{\phi}_3(k)\right)-\zeta_3(k)\right)^2-\zeta_3^2(k)$$

$$\begin{aligned}
\tfrac{1}{\gamma_5}\Delta J_5(k) &= -\left(1-\alpha_3\left\|\hat{\phi}_3(k)\right\|^2\right)\left(\hat{Q}(k)+\hat{w}_3^T(k)\hat{\phi}_3(k)\right)^2 + \\
&\quad \left(\zeta_2(k)+w_2^T\hat{\phi}_2(k)+w_3^T\hat{\phi}_3(k)\right)^2-\zeta_3^2(k) \\
\Delta J_5(k) &\le -\gamma_5\left(1-\alpha_3\left\|\hat{\phi}_3(k)\right\|^2\right)\left(\hat{Q}(k)+\hat{w}_3^T(k)\hat{\phi}_3(k)\right)^2 + \\
&\quad 2\gamma_5\zeta_2^2(k)+2\gamma_5\left(w_2^T\hat{\phi}_2(k)+w_3^T\hat{\phi}_3(k)\right)^2-\gamma_5\zeta_3^2(k) \\
&\le -\gamma_5\left(1-\alpha_3\left\|\hat{\phi}_3(k)\right\|^2\right)\left(\hat{Q}(k)+\hat{w}_3^T(k)\hat{\phi}_3(k)\right)^2 + \\
&\quad 2\gamma_5\zeta_2^2(k)+2\gamma_5\left(w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}}+w_{3\mathrm{m}}\hat{\phi}_{3\mathrm{m}}\right)^2-\gamma_5\zeta_3^2(k) \\
&\le -\gamma_5\left(1-\alpha_3\left\|\hat{\phi}_3(k)\right\|^2\right)\left(\hat{Q}(k)+\hat{w}_3^T(k)\hat{\phi}_3(k)\right)^2 + \\
&\quad 2\gamma_5\zeta_2^2(k)+2\gamma_5 D_5^2-\gamma_5\zeta_3^2(k)
\end{aligned} \tag{45}$$

where

$$D_5 = w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}} + w_{3\mathrm{m}}\hat{\phi}_{3\mathrm{m}} \tag{46}$$

Take the sixth term and substitute (21), (5), then (24)

$$\begin{aligned}
J_6(k) &= \tfrac{\gamma_6}{\alpha_4}\tilde{w}_4^T(k)\tilde{w}_4(k) \\
\tfrac{\alpha_4}{\gamma_6}\Delta J_6(k) &= \tilde{w}_4^T(k+1)\tilde{w}_4(k+1)-\tilde{w}_4^T(k)\tilde{w}_4(k) \\
&= \left[\tilde{w}_4(k)-\alpha_4\hat{\phi}_4(k)\left(\hat{w}_4^T(k)\hat{\phi}_4(k)+\hat{Q}(k)\right)\right]^T * \\
&\quad \left[\tilde{w}_4(k)-\alpha_4\hat{\phi}_4(k)\left(\hat{w}_4^T(k)\hat{\phi}_4(k)+\hat{Q}(k)\right)\right]-\tilde{w}_4^T(k)\tilde{w}_4(k) \\
&= \alpha_4^2\left\|\hat{\phi}_4(k)\right\|^2\left(\hat{w}_4^T(k)\hat{\phi}_4(k)+\hat{Q}(k)\right)^2-2\alpha_4\tilde{w}_4^T\hat{\phi}_4(k)\left(\hat{w}_4^T(k)\hat{\phi}_4(k)+\hat{Q}(k)\right) \\
&= \alpha_4^2\left\|\hat{\phi}_4(k)\right\|^2\left(\hat{w}_4^T(k)\hat{\phi}_4(k)+\hat{Q}(k)\right)^2- \\
&\quad 2\alpha_4\zeta_4(k)\left(\zeta_4(k)+w_4^T\hat{\phi}_4(k)+\zeta_2(k)+w_2^T\hat{\phi}_2(k)\right)+ \\
&\quad \alpha_4\left(\zeta_4(k)+w_4^T\hat{\phi}_4(k)+\zeta_2(k)+w_2^T\hat{\phi}_2(k)\right)^2- \\
&\quad \alpha_4\left(\zeta_4(k)+w_4^T\hat{\phi}_4(k)+\zeta_2(k)+w_2^T\hat{\phi}_2(k)\right)^2 \\
\tfrac{1}{\gamma_6}\Delta J_6(k) &= -\left(1-\alpha_4\left\|\hat{\phi}_4(k)\right\|^2\right)\left(\hat{w}_4^T(k)\hat{\phi}_4(k)+\hat{Q}(k)\right)^2 + \\
&\quad \left(\left(\zeta_4(k)+w_4^T\hat{\phi}_4(k)+\zeta_2(k)+w_2^T\hat{\phi}_2(k)\right)-\zeta_4(k)\right)^2-\zeta_4^2(k)
\end{aligned}$$

$$\begin{aligned}\Delta J_6(k) &= -\gamma_6\left(1-\alpha_4\left\|\hat{\phi}_4(k)\right\|^2\right)\left(\hat{w}_4^T(k)\hat{\phi}_4(k)+\hat{Q}(k)\right)^2 + \\ &\quad 2\gamma_6\left(w_4^T\hat{\phi}_4(k)+w_2^T\hat{\phi}_2(k)\right)^2+2\gamma_6\zeta_2^2(k)-\gamma_6\zeta_4^2(k) \\ &= -\gamma_6\left(1-\alpha_4\left\|\hat{\phi}_4(k)\right\|^2\right)\left(\hat{w}_4^T(k)\hat{\phi}_4(k)+\hat{Q}(k)\right)^2 + \\ &\quad 2\gamma_6 D_6^2+2\gamma_6\zeta_2^2(k)-\gamma_6\zeta_4^2(k)\end{aligned} \tag{47}$$

where

$$D_6 = w_{4\mathrm{m}}\hat{\phi}_{\mathrm{m}} + w_{2\mathrm{m}}\hat{\phi}_{2\mathrm{m}} \tag{48}$$

Take the seventh term

$$\begin{aligned}J_7(k) &= \gamma_7\zeta_2^2(k-1) \\ \Delta J_7(k) &= \gamma_7\zeta_2^2(k)-\gamma_7\zeta_2^2(k-1)\end{aligned} \tag{49}$$

define

$$\gamma_7 = 2\gamma_4\beta^2 \tag{50}$$

Replace (50) into (49)

$$\Delta J_7(k) = 2\gamma_4\beta^2\zeta_2^2(k)-2\gamma_4\beta^2\zeta_2^2(k-1) \tag{51}$$

Take the eighth term and substitute (26) and (32)

$$\begin{aligned}J_8(k) &= \tfrac{\gamma_8}{4}\tilde{x}_1^2(k) \\ \tfrac{3}{\gamma_8}\Delta J_8(k) &= \tilde{x}_1^2(k+1)-\tilde{x}_1^2(k) \\ &= \left(f_{10}-\hat{x}_2(k)+l_2\tilde{y}(k)-f_1(\cdot)-g_1(\cdot)x_2(k)-d_1(k)\right)^2-\tilde{x}_1^2(k) \\ &= \left(f_{10}-\hat{x}_2(k)+l_2\tilde{y}(k)-w_3^T\phi_3(k)-\varepsilon_3(k)+x_2(k)-d_1(k)\right)^2-\tilde{x}_1^2(k) \\ &= \left(f_{10}+l_2\tilde{y}(k)-w_3^T\phi_3(k)-\varepsilon_3(k)-\tilde{x}_2(k)-d_1(k)\right)^2-\tilde{x}_1^2(k) \\ \Delta J_8(k) &\le \gamma_8 l_2^2\tilde{y}^2(k)+\gamma_8\tilde{x}_2^2(k)+\gamma_8\left(-w_3^T\phi_3(k)+f_{10}-\varepsilon_3(k)-d_1(k)\right)^2-\tfrac{\gamma_8}{3}\tilde{x}_1^2(k) \\ &\le \gamma_8 l_2^2\tilde{y}^2(k)+\gamma_8\tilde{x}_2^2(k)+\gamma_8\left(w_{3\mathrm{m}}\phi_{3\mathrm{m}}+f_{10}+\varepsilon_{3\mathrm{m}}+d_{1\mathrm{m}}\right)^2-\tfrac{\gamma_8}{3}\tilde{x}_1^2(k) \\ &\le \gamma_8 l_2^2\tilde{y}^2(k)+\gamma_8\tilde{x}_2^2(k)+\gamma_8 D_7^2-\tfrac{\gamma_8}{3}\tilde{x}_1^2(k)\end{aligned} \tag{52}$$

where

$$D_7 = w_{3\mathrm{m}}\phi_{3\mathrm{m}} + f_{10} + \varepsilon_{3\mathrm{m}} + d_{1\mathrm{m}} \tag{53}$$

Take the ninth term and substitute (27)

$$
\begin{aligned}
J_9(k) &= \tfrac{\gamma_9}{5}\tilde{x}_2^2(k) \\
\tfrac{3}{\gamma_9}\Delta J_9(k) &= \tilde{x}_2^2(k+1) - \tilde{x}_2^2(k) \\
&= \left(f_{20} + g_{20}u(k) + l_3\tilde{y}(k) - f_2(k) - g_2(k)u(k) - d_2(k)\right)^2 - \tilde{x}_2^2(k) \\
&= \left(f_{20} + \left(g_{20} - g_2(k)\right)u(k) + l_3\tilde{y}(k) - f_2(k) - d_2(k)\right)^2 - \tilde{x}_2^2(k) \\
&= \left(f_{20} + \left(g_{20} - g_2(k)\right)\left(\hat{w}_4^T(k)\hat{\phi}_4(k)\right) + l_3\tilde{y}(k) - f_2(k) - d_2(k)\right)^2 - \tilde{x}_2^2(k) \\
&= \begin{pmatrix} f_{20} + \left(g_{20} - g_2(k)\right)\left(\tilde{w}_4^T(k)\hat{\phi}_4(k) + w_4^T\hat{\phi}_4(k)\right) \\ + l_3\tilde{y}(k) - f_2(k) - d_2(k) \end{pmatrix}^2 - \tilde{x}_2^2(k) \\
&\le \begin{pmatrix} f_{20} + \left(g_{20} + g_{2\max}\right)\zeta_4(k) + \left(g_{20} + g_{2\max}\right)w_4^T\hat{\phi}_4(k) \\ + l_3\tilde{y}(k) + f_{2\max} - d_2(k) \end{pmatrix}^2 - \tilde{x}_2^2(k)
\end{aligned}
$$

$$
\begin{aligned}
\Delta J_9(k) &\le \gamma_9\left(f_{20} + \left(g_{20} + g_{2\max}\right)w_4^T\hat{\phi}_4(k) + f_{2\max} - d_2(k)\right)^2 + \\
&\quad \gamma_9\left(g_{20} + g_{2\max}\right)\zeta_4(k) + \gamma_9 l_3^2\tilde{y}^2(k) - \tfrac{\gamma_9}{3}\tilde{x}_2^2(k) \\
&\le \gamma_9\left(f_{20} + \left(g_{20} + g_{2\max}\right)w_{4\mathrm{m}}\hat{\phi}_{4\mathrm{m}} + f_{2\max} + d_{2\mathrm{m}}\right)^2 + \\
&\quad \gamma_9\left(g_{20} + g_{2\max}\right)\zeta_4(k) + \gamma_9 l_3^2\tilde{y}^2(k) - \tfrac{\gamma_9}{3}\tilde{x}_2^2(k) \\
&\le \gamma_9 D_8^2 + \gamma_9\left(g_{20} + g_{2\max}\right)\zeta_4(k) + \gamma_9 l_3^2\tilde{y}^2(k) - \tfrac{\gamma_9}{3}\tilde{x}_2^2(k)
\end{aligned}
\tag{54}
$$

where

$$
D_8 = f_{20} + \left(g_{20} + g_{2\max}\right)w_{4\mathrm{m}}\hat{\phi}_{4\mathrm{m}} + f_{2\max} + d_{2\mathrm{m}} \tag{55}
$$

Take the tenth and final term and substitute (33)

$$
\begin{aligned}
J_{10}(k) &= \tfrac{\gamma_{10}}{3}\tilde{y}^2(k) \\
\Delta J_{10}(k) &= \tfrac{\gamma_{10}}{3}\left(\tilde{y}^2(k+1) - \tilde{y}^2(k)\right) \\
&= \tfrac{\gamma_{10}}{3}\left(\left(\hat{w}_1^T(k)\hat{\phi}_1(k) + l_1\tilde{y}(k) - w_1^T\phi_1(k) - \varepsilon_1(k)\right)^2 - \tilde{y}^2(k)\right) \\
&= \tfrac{\gamma_{10}}{3}\left(\left(\zeta_1(k) + w_1^T\tilde{\phi}_1(k) - \varepsilon_1(k) + l_1\tilde{y}(k)\right)^2 - \tilde{y}^2(k)\right) \\
&= \gamma_{10}\zeta_1^2(k) + \gamma_{10}l_1^2\tilde{y}^2(k) + \gamma_{10}\left(w_1^T\tilde{\phi}_1(k) - \varepsilon_1(k)\right) - \tfrac{\gamma_{10}}{3}\tilde{y}^2(k) \\
&\le \gamma_{10}\zeta_1^2(k) + \gamma_{10}l_1^2\tilde{y}^2(k) + \gamma_{10}\left(w_{1\mathrm{m}}\tilde{\phi}_{1\mathrm{m}} + \varepsilon_{1\mathrm{m}}\right) - \tfrac{\gamma_{10}}{3}\tilde{y}^2(k) \\
&\le \gamma_{10}\zeta_1^2(k) + \gamma_{10}l_1^2\tilde{y}^2(k) + \gamma_{10}D_9^2 - \tfrac{\gamma_{10}}{3}\tilde{y}^2(k)
\end{aligned}
\tag{56}
$$

where

$$
D_9 = w_{1\mathrm{m}}\tilde{\phi}_{1\mathrm{m}} + \varepsilon_{1\mathrm{m}} \tag{57}
$$

Combine (37), (39), (41), (43), (45), (47), (51), (52), (54), and (56) to get the first difference of the Lyapunov function

$$\begin{aligned}
\Delta J \le\ & \gamma_1 l_5^2 \tilde{x}_1^2(k) + \gamma_1 l_5^2 e_1^2(k) + \gamma_1 e_2^2(k) + \gamma_1 \zeta_3^2(k) + \gamma_1 D_1^2 - \tfrac{\gamma_1}{5} e_1^2(k) + \\
& \gamma_2 l_6^2 e_2^2(k) + \gamma_2 g_{2\max}^2 \zeta_4^2(k) + \gamma_2 D_2^2 - \tfrac{\gamma_2}{3} e_2^2(k) + \\
& -\gamma_3 \left(1 - \alpha_1 \left\|\hat{\phi}_1(k)\right\|^2\right) \begin{pmatrix} \hat{w}_1(k)\hat{\phi}_1(k) \\ +l_4 \tilde{y}(k) \end{pmatrix}^2 + 2\gamma_3 D_3^2 + 2\gamma_3 l_4^2 \tilde{y}^2(k) - \gamma_3 \zeta_1^2(k) + \\
& -\gamma_4 \left(1 - \alpha_2 \left\|\hat{\phi}_2(k)\right\|^2\right) \begin{pmatrix} \hat{Q}(k) + \beta^{N+1} p(k) \\ -\beta \hat{Q}(k-1) \end{pmatrix}^2 - \gamma_4 \zeta_2^2(k) + 2\gamma_4 \beta^2 \zeta_2^2(k-1) + 2\gamma_4 D_4^2 + \\
& -\gamma_5 \left(1 - \alpha_3 \left\|\hat{\phi}_3(k)\right\|^2\right) \left(\hat{Q}(k) + \hat{w}_3^T(k)\hat{\phi}_3(k)\right)^2 + 2\gamma_5 \zeta_2^2(k) + 2\gamma_5 D_5^2 - \gamma_5 \zeta_3^2(k) + \\
& -\gamma_6 \left(1 - \alpha_4 \left\|\hat{\phi}_4(k)\right\|^2\right) \left(\hat{w}_4^T(k)\hat{\phi}_4(k) + \hat{Q}(k)\right)^2 + 2\gamma_6 D_6^2 + 2\gamma_6 \zeta_2^2(k) - \gamma_6 \zeta_4^2(k) + \\
& 2\gamma_4 \beta^2 \zeta_2^2(k) - 2\gamma_4 \beta^2 \zeta_2^2(k-1) + \\
& \gamma_8 l_2^2 \tilde{y}^2(k) + \gamma_8 \tilde{x}_2^2(k) + \gamma_8 D_7^2 - \tfrac{\gamma_8}{3} \tilde{x}_1^2(k) + \\
& \gamma_9 D_8^2 + \gamma_9 \left(g_{20} + g_{2\max}\right) \zeta_4(k) + \gamma_9 l_3^2 \tilde{y}^2(k) - \tfrac{\gamma_9}{3} \tilde{x}_2^2(k) \\
& \gamma_{10} \zeta_1^2(k) + \gamma_{10} l_1^2 \tilde{y}^2(k) + \gamma_{10} D_9^2 - \tfrac{\gamma_{10}}{3} \tilde{y}^2(k)
\end{aligned} \tag{58}$$

rearrange

$$\begin{aligned}
\Delta J \le\ & \gamma_1 l_5^2 e_1^2(k) - \tfrac{\gamma_1}{5} e_1^2(k) + \gamma_1 e_2^2(k) + \gamma_2 l_6^2 e_2^2(k) - \tfrac{\gamma_2}{3} e_2^2(k) + \\
& \gamma_{10} \zeta_1^2(k) - \gamma_3 \zeta_1^2(k) + 2\gamma_5 \zeta_2^2(k) + 2\gamma_6 \zeta_2^2(k) + 2\gamma_4 \beta^2 \zeta_2^2(k) - \gamma_4 \zeta_2^2(k) + \\
& \gamma_1 \zeta_3^2(k) - \gamma_5 \zeta_3^2(k) + \gamma_2 g_{2\max}^2 \zeta_4^2(k) + \gamma_9 \left(g_{20} + g_{2\max}\right) \zeta_4(k) - \gamma_6 \zeta_4^2(k) + \\
& \gamma_8 \tilde{x}_2^2(k) - \tfrac{\gamma_9}{3} \tilde{x}_2^2(k) + 2\gamma_4 \beta^2 \zeta_2^2(k-1) - 2\gamma_4 \beta^2 \zeta_2^2(k-1) + \gamma_1 l_5^2 \tilde{x}_1^2(k) - \tfrac{\gamma_8}{3} \tilde{x}_1^2(k) + \\
& 2\gamma_3 l_4^2 \tilde{y}^2(k) + \gamma_8 l_2^2 \tilde{y}^2(k) + \gamma_9 l_3^2 \tilde{y}^2(k) + \gamma_{10} l_1^2 \tilde{y}^2(k) - \tfrac{\gamma_{10}}{3} \tilde{y}^2(k) + \\
& \gamma_1 D_1^2 + \gamma_2 D_2^2 + 2\gamma_3 D_3^2 + 2\gamma_4 D_4^2 + 2\gamma_5 D_5^2 + 2\gamma_6 D_6^2 + \gamma_8 D_7^2 + \gamma_9 D_8^2 + \gamma_{10} D_9^2 \\
& -\gamma_3 \left(1 - \alpha_1 \left\|\hat{\phi}_1(k)\right\|^2\right) \begin{pmatrix} \hat{w}_1(k)\hat{\phi}_1(k) \\ +l_4 \tilde{y}(k) \end{pmatrix}^2 - \gamma_4 \left(1 - \alpha_2 \left\|\hat{\phi}_2(k)\right\|^2\right) \begin{pmatrix} \hat{Q}(k) + \beta^{N+1} p(k) \\ -\beta \hat{Q}(k-1) \end{pmatrix}^2 \\
& -\gamma_5 \left(1 - \alpha_3 \left\|\hat{\phi}_3(k)\right\|^2\right) \begin{pmatrix} \hat{Q}(k) + \\ \hat{w}_3^T(k)\hat{\phi}_3(k) \end{pmatrix}^2 - \gamma_6 \left(1 - \alpha_4 \left\|\hat{\phi}_4(k)\right\|^2\right) \begin{pmatrix} \hat{w}_4^T(k)\hat{\phi}_4(k) \\ +\hat{Q}(k) \end{pmatrix}^2
\end{aligned} \tag{59}$$

gather terms and simplify

$$\begin{aligned}\Delta J \le & -\left(\tfrac{\gamma_1}{5}-\gamma_1 l_5^2\right)e_1^2(k)-\left(\tfrac{\gamma_2}{3}-\gamma_1-\gamma_2 l_6^2\right)e_2^2(k)-\left(\gamma_3-\gamma_{10}\right)\zeta_1^2(k)\\ & -\left(\gamma_4-2\gamma_5-2\gamma_6-2\gamma_4\beta^2\right)\zeta_2^2(k)-\left(\gamma_5-\gamma_1\right)\zeta_3^2(k)\\ & -\left(\gamma_6-\gamma_2 g_{2\max}^2-\gamma_9\left(g_{20}+g_{2\max}\right)\right)\zeta_4^2(k)\\ & -\left(\tfrac{\gamma_8}{3}-\gamma_1 l_5^2\right)\tilde{x}_1^2(k)-\left(\tfrac{\gamma_9}{3}-\gamma_8\right)\tilde{x}_2^2(k)\\ & -\left(\tfrac{\gamma_{10}}{3}-2\gamma_3 l_4^2-\gamma_8 l_2^2-\gamma_9 l_3^2-\gamma_{10} l_1^2\right)\tilde{y}^2(k)+D_M^2\\ & -\gamma_3\left(1-\alpha_1\left\|\hat{\phi}_1(k)\right\|^2\right)\left(\hat{w}_1(k)\hat{\phi}_1(k)+l_4\tilde{y}(k)\right)^2\\ & -\gamma_4\left(1-\alpha_2\left\|\hat{\phi}_2(k)\right\|^2\right)\left(\hat{Q}(k)+\beta^{N+1}p(k)-\beta\hat{Q}(k-1)\right)^2\\ & -\gamma_5\left(1-\alpha_3\left\|\hat{\phi}_3(k)\right\|^2\right)\left(\hat{Q}(k)+\hat{w}_3^T(k)\hat{\phi}_3(k)\right)^2\\ & -\gamma_6\left(1-\alpha_4\left\|\hat{\phi}_4(k)\right\|^2\right)\left(\hat{w}_4^T(k)\hat{\phi}_4(k)+\hat{Q}(k)\right)^2\end{aligned} \tag{60}$$

where

$$D_{\mathrm{m}}^2=\gamma_1 D_1^2+\gamma_2 D_2^2+2\gamma_3 D_3^2+2\gamma_4 D_4^2+2\gamma_5 D_5^2+2\gamma_6 D_6^2+\gamma_8 D_7^2+\gamma_9 D_8^2+\gamma_{10} D_9^2 \tag{61}$$

Solving the following equations

$$\begin{aligned}&\gamma_1>5\gamma_1 l_5^2;\ \gamma_2>3\gamma_1+3\gamma_2 l_6^2;\ \gamma_3>\gamma_{10};\ \gamma_4>2\gamma_5+2\gamma_6+2\gamma_4\beta^2;\ \gamma_5>\gamma_1;\\ &\gamma_6>\gamma_2 g_{2\max}^2+\gamma_9\left(g_{20}+g_{2\max}\right);\ \gamma_7=2\gamma_4\beta^2;\ \gamma_8>3\gamma_1 l_5^2;\ \gamma_9>3\gamma_8;\\ &\gamma_{10}>6\gamma_3 l_4^2+3\gamma_8 l_2^2+3\gamma_9 l_3^2+3\gamma_{10} l_1^2;\end{aligned} \tag{62}$$

will result in unique values for the gains and the auxiliary variables provided (83) through (89) holds. This implies $\Delta J(k)<0$ as long as (62) through (89) hold *and* any *one* of the following hold

$$\begin{aligned}&\left|e_1(k)\right|>\frac{D_M}{\sqrt{\tfrac{\gamma_1}{5}-\gamma_1 l_5^2}};\ \left|e_2(k)\right|>\frac{D_M}{\sqrt{\tfrac{\gamma_2}{3}-\gamma_1-\gamma_2 l_6^2}};\ \left|\zeta_1(k)\right|>\frac{D_M}{\sqrt{\gamma_3-\gamma_{10}}};\\ &\left|\zeta_3(k)\right|>\frac{D_M}{\sqrt{\gamma_5-\gamma_1}};\ \left|\zeta_2(k)\right|>\frac{D_M}{\sqrt{\gamma_4-2\gamma_5-2\gamma_6-2\gamma_4\beta^2}};\\ &\left|\zeta_4(k)\right|>\frac{D_M}{\sqrt{\gamma_6-\gamma_2 g_{2\max}^2-\gamma_9\left(g_{20}+g_{2\max}\right)}};\left|\tilde{x}_1(k)\right|>\frac{D_M}{\sqrt{\tfrac{\gamma_8}{3}-\gamma_1 l_5^2}};\\ &\left|\tilde{x}_2(k)\right|>\frac{D_M}{\sqrt{\tfrac{\gamma_9}{3}-\gamma_8}};\ \left|\tilde{y}(k)\right|>\frac{D_M}{\sqrt{\tfrac{\gamma_{10}}{3}-2\gamma_3 l_4^2-\gamma_8 l_2^2-\gamma_9 l_3^2-\gamma_{10} l_1^2}};\end{aligned} \tag{63}$$

VITA

Peter Shih a.k.a. Lei Shih was born in Taiwan R.O.C on the 5th of August 1980. He earned a bachelor's degree in Biomedical Engineering at Washington University in Saint Louis in 2002. He received a master's degree in Computer Engineering at the University of Missouri – Rolla in May of 2007.