Fall 2013

Finite-horizon optimal control of linear and a class of nonlinear systems

Qiming Zhao

FINITE-HORIZON OPTIMAL CONTROL OF LINEAR AND A CLASS OF NONLINEAR SYSTEMS

by

QIMING ZHAO

A DISSERTATION

Presented to the Faculty of the Graduate School of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

DOCTOR OF PHILOSOPHY

in

ELECTRICAL ENGINEERING

2013

Approved by

Jagannathan Sarangapani, Advisor
Levent Acar
Maciej Zawodniok
Robert G. Landers
Sriram Chellappan

PUBLICATION DISSERTATION OPTION

This dissertation contains the following five articles:

Paper I: Qiming Zhao, Hao Xu and S. Jagannathan, “Finite-Horizon Optimal Adaptive Control of Uncertain Linear Discrete-time Systems”, under review with Optimal Control Applications and Methods and “Fixed Final Time Optimal Adaptive Control of Linear Discrete-time Systems in Input-Output Form”, accepted by Journal of Artificial Intelligence and Soft Computing Research. (Invited paper).

Paper II: Qiming Zhao, Hao Xu and S. Jagannathan, “Finite-Horizon Optimal Adaptive Control of Uncertain Quantized Linear Discrete-time System”, under review with International Journal on Adaptive Control and Signal Processing.

Paper III: Qiming Zhao, Hao Xu and S. Jagannathan, “Neural Network-based Finite-Horizon Optimal Control of Uncertain Affine Nonlinear Discrete-time Systems”, minor revision and resubmitted to IEEE Transactions on Neural Networks and Learning Systems.

Paper IV: Qiming Zhao, Hao Xu and S. Jagannathan, “Fixed Final-Time near Optimal Regulation of Nonlinear Discrete-time Systems in Affine Form using Output Feedback”, under review with Acta Automatica Sinica.

Paper V: Qiming Zhao, Hao Xu and S. Jagannathan, “Finite-Horizon Near Optimal Control of Quantized Nonlinear Discrete-time Systems with Input Constraint using Neural Networks”, under review with IEEE Transactions on Neural Networks and Learning Systems.

Research in part by NSF grant ECCS#1128281 and Intelligent Systems Center

ABSTRACT

Traditionally, optimal control of dynamical systems with known system dynamics is obtained in a backward-in-time and offline manner either by using Riccati or Hamilton-Jacobi-Bellman (HJB) equation. In contrast, in this dissertation, finite-horizon optimal regulation has been investigated for both linear and nonlinear systems in a forward-in-time manner when system dynamics are uncertain. Value and policy iterations are not used while the value function (or Q-function for linear systems) and control input are updated once a sampling interval consistent with standard adaptive control.

First, the optimal adaptive control of linear discrete-time systems with unknown system dynamics is presented in Paper I by using Q-learning and Bellman equation while satisfying the terminal constraint. A novel update law that uses history information of the cost to go is derived. Paper II considers the design of the linear quadratic regulator in the presence of state and input quantization. Quantization errors are eliminated via a dynamic quantizer design and the parameter update law is redesigned from Paper I.

Furthermore, an optimal adaptive state feedback controller is developed in Paper III for the general nonlinear discrete-time systems in affine form without the knowledge of system dynamics. In Paper IV, a NN-based observer is proposed to reconstruct the state vector and identify the dynamics so that the control scheme from Paper III is extended to output feedback. Finally, the optimal regulation of quantized nonlinear systems with input constraint is considered in Paper V by introducing a non-quadratic cost functional. Closed-loop stability is demonstrated for all the controller designs developed in this dissertation by using Lyapunov analysis while all the proposed schemes function in an online and forward-in-time manner so that they are practically viable.

ACKNOWLEDGEMENTS

I would like to express sincere gratitude to my advisor, Professor Jagannathan Saragapani, for his close supervision, deep knowledge, patience, and general support over the last three years. I would also like to thank Dr. Levent Acar, Dr. Maciej Zawodniok, Dr. Robert G. Landers and Dr. Sriram Chellappan for serving on my doctoral committee. Further, I would like to thank Dr. Hao Xu, who gave me many helpful discussions and suggestions concerning my thesis work. In addition, I would like to thank the National Science Foundation (NSF) and Intelligent System Center (ISC) for providing financial support through my Ph.D. study.

I would like to thank my colleagues, Dr. Hassan Zargarzadeh, Avimanyu Sahoo, as well as many other friends at the Embedded Systems and Networking Laboratory, who made my Ph.D. life more fun and interesting.

Finally, I want to thank my wife, Shanshan Wang, my son, Allen Zhao, my parents, Zhenhe Zhao and Yuan Liu, and parents in-law, Zhenqiang Wang and Lifei Zhou, as well as the rest of my familiy for their infinite love, care and encouragement, without which I would have not been at this stage of my life.

TABLE OF CONTENTS

SECTION

LIST OF ILLUSTRATIONS

Figure Page

INTRODUCTION

PAPER I

PAPER II

PAPER III

PAPER IV

PAPER V

1. INTRODUCTION

Optimal control of discrete-time linear and nonlinear systems has been one of the key focus topics of the control area for past several decades [1][2]. In contrast to the infinite-horizon case, which has been intensively studied in the literature [7]-[16], the finite-horizon optimal control that enjoys great practical merits, has been still not well developed due to inherent challenges resulting from time-dependent nature and the terminal constraint, etc. For both infinite and finite horizon optimal control, system dynamics are needed.

Two major differences between finite and infinite-horizon optimal control are briefly given here. First, for infinite-horizon case, the solution to the Hamilton-Jacobi-Bellman (HJB) equation for nonlinear systems or the Riccati equation (RE) for the case of linear systems is time-invariant, whereas in the case of finite-horizon, the solution for either HJB equation or RE becomes essentially time-dependent. Second, a terminal state constraint, which needs to be tackled properly, is imposed for the finite-horizon. By contrast, the terminal constraint is not asserted for the infinite-horizon case. Therefore, solving finite-horizon optimal control presents a great challenge due to the time-dependent nature and with additional requirement on satisfying the terminal constraint.

Traditionally, in the finite-horizon optimal control of linear systems with quadratic performance index (PI), or referred to as linear quadratic regulator (LQR), the optimal control policy is obtained by solving the RE from the terminal value $\boldsymbol{S}_{\mathrm{N}}$, where $\boldsymbol{S}_{\mathrm{N}}$ is the weighting matrix for the terminal states. However, though this method theoretically yields an optimal control policy, it is not suitable for real-time implementation due to its backward-in-time and offline feature. Further, for a general nonlinear affine system, finding optimal control policy is much more difficult even under infinite-horizon case, since the solution to the HJB equation normally does not have a closed-form solution [1][2]. Only approximate and iterative approach is normally utilized.

Given the importance and challenges mentioned above for the finite-horizon optimal control problem, this topic has attracted many control researchers over the past decades who had made great strides to tackle this challenging but promising problem. In next subsection, we present an overview of the current methodologies as well as some discussion on their shortcomings. Subsequently, the organization and contributions of this dissertation are introduced.

1.1 OVERVIEW OF THE OPTIMAL CONTROL METHODOLOGIES

Theoretically, for infinite-horizon optimal control policy for an affine nonlinear system can be obtained by solving the HJB equation, which is essentially an algebraic equation. When considering the case of LQR, the HJB equation further reduces to the algebraic RE (ARE). However, for most cases, it is impossible to solve the HJB equation since it is generally a nonlinear partial differential (or difference) equation [1][2]. Therefore, offline scheme with an approximator, e.g., neural networks (NN), is utilized to find the approximated solution to the HJB equation [9][14], where the NN weights are trained a priori within an operating region before they are implemented in the controller.

The effort in [9][14] provides some insight into solving the nonlinear optimal control problem, whereas offline training is not preferable for realistic implementation since it is not currently clear how much training is needed for a given system. In addition, when the dynamics of the system are not known even partially, which is normally the case in a realistic scenario the optimal control policy cannot be obtained even for the linear systems. Hence, optimal control of dynamic systems by relaxing the requirement on the knowledge of system dynamics poses another challenging problem for the control researchers.

To overcome the difficulties mentioned above, approximate dynamic programming (ADP) has been widely promoted in control community. Policy and/or value iteration serves as a key technique to solve the optimal control. Basically, the iteration-based scheme utilizes an initial stabilizing control input and updates not only the cost/value function, which becomes the solution to the HJB equation, but also the control policy “iteratively” until the estimated control converges to the optimal one all within a sampling interval. This approach enjoys great advantages over the conventional method since the control policy can be obtained in a forward-in-time manner. For LQR problems, Q-learning methodology is rather popular since the complete system dynamics can be relaxed by iteratively approximating an action-dependent Q-function [11][15][19], which in turn provides the Kalman gain.

Even though iteration-based method has been proven to be an effective way of solving optimal control problem with many successful applications, however, either policy or value iteration, requires significant number of iterations within each time step to guarantee convergence. This poses a challenge in the control design since the number of iterations for convergence is not known beforehand. It has been shown in [12] that with an inadequate number of iterations, the system can become unstable. To circumvent this shortcoming, the authors in [7] proposed a novel “time-based” methodology for general nonlinear discrete-time systems in affine form where the optimal control policy can be obtained based on the history information of the system, thus relaxing the need of performing policy/value iterations. The solution of the HJB equation is approximated by utilizing two NNs at each time step and thus the approach yields an online and forward-in-time algorithm. In [8], the authors considered the optimal regulation of a linear system under network imperfections. The idea of Q-learning in the case of linear system is used to relax the system dynamics with an adaptive estimator effectively learning the Q-function and thus relaxing the iterations or offline training phase. However, the algorithms presented in both [7][8] mainly deal with the infinite-horizon case.

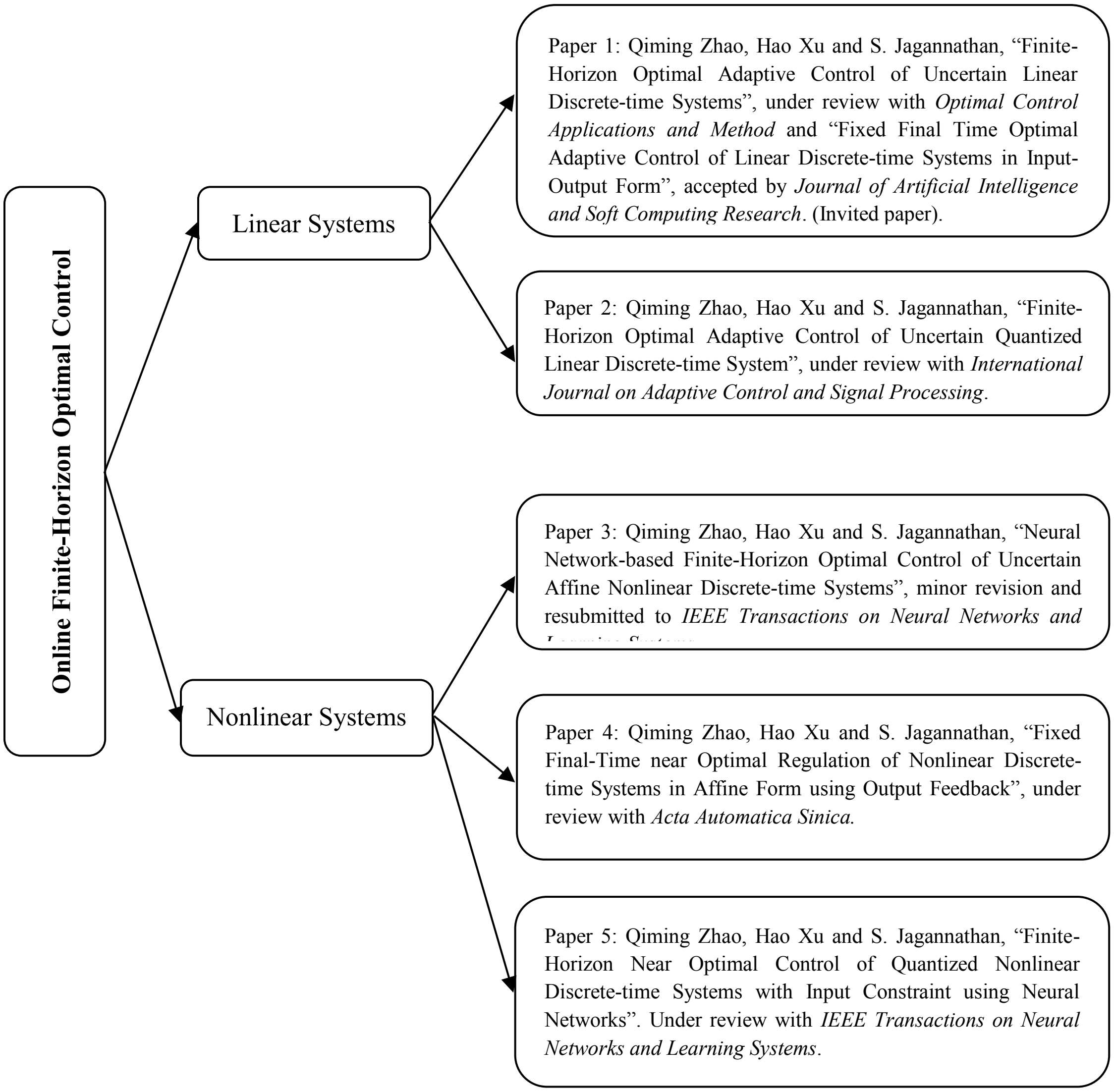
Regarding the finite-horizon optimal control, the terminal constraint as well as the time-varying nature of the solution to either RE or HJB equation needs to be properly taken care of. Other than the theoretical approach [1][2], the author in [3] tackled the problem by solving the generalized HJB (GHJB) equation, which does not depend upon the solution of the system, in a successive way. The terminal constraint is forced to satisfy at an iteration such that the boundary condition can be properly satisfied with the improved control policy. The coefficients of the value function approximator are solved by using Galerkin projections. This however requires extensive computation of a large number of integrals. Later in [4], the author extended the work in [3] by utilizing NN to reduce the computation burden. The NN with the structure of a time-varying weights and state-dependent activation function is used to handle the time-dependent nature of the finite-horizon value function approximation. The optimal control policy is obtained by backward integration of an ordinary differential equation (ODE) from the known terminal NN weights. Therefore, [3] and [4] still yields a backward-in-time solution.

On the other hand, the authors in [5] employed the iterative ADP technique to handle the finite-horizon optimal regulation. A greatest lower bound (ε-bound) of all the performance indices is introduced and it is shown that the ε-optimal control scheme can obtain the suboptimal solutions within a fixed finite number of control steps that make the policy iterations converge to the optimal value with an ε-error. However, the terminal time is not specified in [5] and the terminal state is fixed at the origin. Later in [6], the authors considered the finite-horizon optimal control of nonlinear discrete-time systems with input constraint by using offline training scheme. The time-varying nature of finite-horizon is handled by utilizing a NN which incorporates constant weights and time-varying activation function. The idea proposed in [6] is essentially a standard direct heuristic dynamic programming (DHDP)-based scheme by using policy/value iterations. The terminal constraint is satisfied by introducing an augmented vector incorporating the terminal value of the co-state $\lambda(\mathrm{N})$. Hence, [5] and [6] tackled the finite-horizon optimal control problem is essentially iteration-based.

Although the previous work [3][4][5][6] provided some good insights into solving the finite-horizon optimal control problem, the solutions, however, are either backward-in-time or iterative, and are unsuitable for hardware implementation. Furthermore, all the aforementioned works require the knowledge of the system dynamics which is another bottleneck as mentioned before. Therefore, a control scheme, which can be implemented in an online and forward-in-time manner without needing the system dynamics, is still unresolved and yet to be developed.

1.2 ORGANIZATION OF THE DISSERTATION

In this dissertation, a suite of novel finite-horizon time-based optimal regulation schemes for both linear and nonlinear systems are developed without needing the knowledge of system dynamics. The proposed method yields an online and forward-in-time design scheme which is more preferable under practical situations. This dissertation is presented in the form of five chapters as outlined in Figure 1. The first two papers deal with the linear system, whereas nonlinear systems are considered in the last three papers.



Paper 1: Qiming Zhao, Hao Xu and S. Jagannathan, “Finite-Horizon Optimal Adaptive Control of Uncertain Linear Discrete-time Systems”, under review with *Optimal Control Applications and Method* and “Fixed Final Time Optimal Adaptive Control of Linear Discrete-time Systems in Input-Output Form”, accepted by *Journal of Artificial Intelligence and Soft Computing Research*. (Invited paper).

Paper 2: Qiming Zhao, Hao Xu and S. Jagannathan, “Finite-Horizon Optimal Adaptive Control of Uncertain Quantized Linear Discrete-time System”, under review with *International Journal on Adaptive Control and Signal Processing*.

Paper 3: Qiming Zhao, Hao Xu and S. Jagannathan, “Neural Network-based Finite-Horizon Optimal Control of Uncertain Affine Nonlinear Discrete-time Systems”, minor revision and resubmitted to *IEEE Transactions on Neural Networks and*

Paper 4: Qiming Zhao, Hao Xu and S. Jagannathan, “Fixed Final-Time near Optimal Regulation of Nonlinear Discrete-time Systems in Affine Form using Output Feedback”, under review with *Acta Automatica Sinica.*

Paper 5: Qiming Zhao, Hao Xu and S. Jagannathan, “Finite-Horizon Near Optimal Control of Quantized Nonlinear Discrete-time Systems with Input Constraint using Neural Networks”. Under review with *IEEE Transactions on Neural Networks and Learning Systems*.

Figure 1.1. Outline of the dissertation

In the first paper, the finite-horizon optimal adaptive control of linear discrete-time systems with unknown system dynamics is presented by using ADP technique in a forward-in-time manner. An adaptive estimator (AE) is introduced with the idea of Q-learning to relax the requirement of system dynamics. The time-varying nature of the solution to the Bellman equation is handled by utilizing a time-dependent basis function while the terminal constraint is incorporated as part of the update law of the AE in solving the optimal feedback control. The proposed optimal regulation scheme of the uncertain linear system requires an initial admissible control input and yields a forward-in-time and online solution without using value and/or policy iterations. Furthermore, an adaptive observer is proposed so that the optimal adaptive control design depends only on the reconstructed states so as to realize an optimal output feedback control design. For the time invariant linear discrete-time systems, the closed-loop dynamics becomes non-autonomous and involved, but verified by using standard Lyapunov and Geometric sequence theory.

The second paper investigates the adaptive finite-horizon optimal regulation design for unknown linear discrete-time control systems under the quantization effect for both system states and control inputs. First, dynamic quantizer with time-varying step-size is utilized to mitigate the quantization error wherein it is shown that the quantization error will decrease overtime thus overcoming the drawback of the traditional uniform quantizer. Next, to relax the knowledge of system dynamics and achieve optimality, the Q-learning methodology is adopted under Bellman’s principle. An adaptive online estimator, which learns the time-varying value function, is updated at each time step so that policy and/or value iteration are not performed. Furthermore, an additional error term corresponding to the terminal constraint is defined and minimized along the system trajectory. Consequently, the optimality can be achieved while satisfying the terminal constraint in the presence of quantization errors. The proposed design scheme yields a forward-in-time online scheme, which enjoys great practical merits. Lyapunov analysis is used to show the boundedness of the closed-loop system.

On the other hand, in the third paper, the finite-horizon optimal control design for nonlinear discrete-time systems in affine form is presented. In contrast with the traditional ADP methodology, which requires at least partial knowledge of the system dynamics, the complete system dynamics are relaxed by utilizing a novel NN-based identifier to learn the control coefficient matrix. The identifier is then used together with the actor-critic-based scheme to learn the time-varying solution, referred to as the value function, of the HJB equation in an online and forward-in-time manner. NNs with constant weights and time-varying activation functions are considered to handle the time-varying nature of the value function. To properly satisfy the terminal constraint, an additional error term is incorporated in the novel update law such that the terminal constraint error is also minimized over time. Policy and/or value iterations are not needed and the NN weights are updated once a sampling instant. Stability of the closed-loop system is verified by standard Lyapunov theory under non-autonomous analysis.

In the fourth paper, the idea is extended to the finite-horizon optimal control of affine nonlinear system using output feedback. An extended version of NN-based Luenberger observer is first proposed to reconstruct the system states as well as identify the dynamics of the system. The novel structure of the observer relaxes the need for a separate identifier to construct the control coefficient matrix. Next, reinforcement learning methodology with actor-critic structure is utilized to approximate the time-varying solution of the HJB equation by using a neural network. To properly satisfy the terminal constraint, a new error term is defined and incorporated in the NN update law so that the terminal constraint error is also minimized over time. The NNs with constant weights and time-dependent activation function is employed to approximate the time-varying value function which subsequently is utilized to generate the finite horizon near optimal control policy due to NN reconstruction errors. The proposed scheme functions in a forward-in-time manner without offline training phase. Lyapunov analysis is used to investigate the stability of the overall closed-loop system.

Finally, in the fifth paper, the finite-horizon optimal regulation scheme is further extended to nonlinear discrete-time systems with input constraints and quantization effect. First, by utilizing a non-quadratic cost functional, the effect of actor saturation is taken into consideration while guaranteeing the optimality. Next, the observer design from the fourth paper is used to handle the unavailability of the system states as well as the control coefficient matrix. The actor-critic structure is employed to estimate both the time-dependent value function and the control signals by NNs with constant weights and time-varying activation functions. The terminal constraint, similar as previous papers, is properly satisfied by minimizing a newly defined error term as time evolves. Finally, quantization error is effectively mitigated by using the idea of dynamic quantizer design that is introduced in the second paper. As a result, the input constrained optimal regulation problem is tackled in a forward-in-time and online manner which enjoys great practical merits.

1.3 CONTRIBUTIONS OF THE DISSERTATION

In the past literature, the finite-horizon optimal control is tackled by either backward-in-time solution [3][4] or through offline training [5][6]. The main objective of this dissertation is to develop an online finite-horizon time-based optimal regulation scheme which performs in a forward-in-time fashion. Hence the contributions of this dissertation can be summarized as follows.

In the first paper, the main contributions include the development of finite-horizon optimal adaptive control of uncertain linear discrete-time systems with state and output feedback via an observer provided the observer converges faster than the controller. The terminal constraint is incorporated in the controller design. Boundedness of the regulation and parameter estimation errors are demonstrated by using Lyapunov and geometric sequence analysis. The proposed controller functions forward-in-time with no offline training phase. In addition, the controller does not use value and policy iterations while the cost function and optimal control input are updated once a sampling interval consistent with the standard adaptive control.

The contributions of second paper involve the design of the dynamic quantizer design coupled with the development of finite-horizon optimal adaptive control of uncertain linear discrete-time systems. First a new parameter is introduced in this paper to ensure that the quantizer does not saturate while the quantization error will decrease overtime due to the analysis of the quantization error bound. The terminal state constraint is incorporated and satisfied in the novel controller design scheme. Boundedness of the regulation, parameter estimation and quantization errors are demonstrated by using

Lyapunov stability analysis. If the time interval is stretched, the asymptotic stability of the closed-loop system is demonstrated.

In the third paper, the major contributions include the development of an optimal adaptive NN control scheme in finite-horizon for nonlinear discrete-time systems. Normally under the ADP scheme, at least partial dynamics, i.e., the control coefficient matrix are needed to generate the optimal control policy [7][24]. Therefore, a novel NN-based online identifier is first proposed to learn the control coefficient matrix such that the complete system dynamics are not needed. Actor-critic scheme is utilized to learn the time-varying solution of the HJB equation by two NNs with constant and time-varying activation function. Novel update law incorporating the terminal constraint error is developed based on generalized gradient-descent algorithm. Therefore, the proposed design scheme performs in a forward-in-time manner whereas iteration-based methodology is not needed. Lyapunov analysis verifies the stability of all the parameter estimation errors and the overall closed-loop system.

The contributions of the fourth paper include novel design of finite-horizon optimal regulation of nonlinear discrete-time systems when the system states are not available. An extended Luenberger observer is proposed to estimate both the system states and the control coefficient matrix, which is subsequently used for the optimal controller design. The novel structure of the observer relaxes the need for a separate identifier thus simplifies the overall design.

Finally, the fifth paper further extends our finite-horizon optimal regulatior to the quantized nonlinear systems with input constraint. Though input constrained optimal control is not a new topic [6][14], however, to the best knowledge of the authors, developing an (near) optimal regulator for quantized control systems under finite-horizon scenario in a forward-in-time fashion without using iteration-based approach still remains unresolved. By adopting a newly defined non-quadratic cost functional [25], we are able to successfully utilize our developed ideas from [26], [27] and Paper IV to estimate the value function in a new form so that the optimality can be eventually achieved. Policy/value iteration are not needed due to our parameter tuning laws which are updated once a sampling interval.

1.4 REFERENCES

[1] F. L. Lewis and V. L. Syrmos, *Optimal Control*, 2nd edition, New York: Wiley, 1995.

[2] D. E. Kirk, *Optimal Control Theory: An Introduction*, Prentice-Hall, 1970.

[3] R. Beard, “Improving the closed-loop performance of nonlinear systems,” Ph.D. dissertation, Electr. Eng. Dept., Rensselaer Polytechnic Institute, USA, 1995.

[4] T. Cheng, F. L. Lewis, and M. Abu-Khalaf, “A neural network solution for fixed-final-time optimal control of nonlinear systems,” *Automatica*, vol. 43, pp. 482-490, 2007.

[5] W. Feiyue, J. Ning, L. Derong and W. Qinglai, “Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with ε -error bound,” *IEEE Trans. Neural Networks*, vol. 22, pp. 24-36, 2011.

[6] A. Heydari and S. N. Balakrishnan, “Finite-horizon Control-Constrained Nonlinear Optimal Control Using Single Network Adaptive Critics,” *IEEE Trans. Neural Networks and Learning Systems*, vol.24, pp. 145-157, 2013.

[7] T. Dierks and S. Jagannathan, “Online optimal control of affine nonlinear discrete-time systems with unknown internal dynamics by using time-based policy update,” *IEEE Trans. Neural Networks and Learning Systems*, vol. 23, pp. 1118-1129, 2012.

[8] H. Xu, S. Jagannathan and F. L. Lewis, “Stochastic optimal control of unknown networked control systems in the presence of random delays and packet losses,” *Automatica*, vol. 48, pp. 1017-1030, 2012.

[9] C. Zheng and S. Jagannathan, “Generalized Hamilton-Jacobi-Bellman formulation based neural network control of affine nonlinear discrete-time systems”, *IEEE Trans. Neural Networks*, vol.7, pp. 90-106, 2008.

[10] F. L. Lewis and D. Vrabie, “Reinforcement learning and adaptive dynamic programming for feedback control”, *IEEE Circ. and Syst. Magaz.*, vol. 9, pp. 32-50, 2009.

[11] S. J. Bradtke and B. E. Ydstie, “Adaptive linear quadratic control using policy iteration,” in *Proc. on Amer Control Conf, Baltimore*, pp. 3475-3479, 1994.

[12] H. Xu and S. Jagannathan, "Stochastic optimal controller design for uncertain nonlinear networked control system via neuro dynamic programming*," IEEE Trans. Neural Netw. And Learning Syst*, vol. 24, pp. 471-484, 2013.

[13] T. Dierks, T.B. Thumati and S. Jagannathan, "Adaptive dynamic programming-based optimal control of unknown affine nonlinear discrete-time systems," in *Proc. Intern. Joint Conf. Neur. Netwo*, pp. 711-716, 2009.

[14] M. Abu-Khalaf and F. L. Lewis, "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach," in *Robotics and Automation (ICRA), IEEE International Conference on*, pp. 1551-1558, 2011.

[15] A. Al-Tamimi and F. L. Lewis and M. Abu-Khalaf, "Model-free Q-learning designs for linear discrete-time zero-sum games with application to H-infinity control," *Automatica*, vol. 43, pp. 473-481, 2007.

[16] A. Al-Tamimi, F. L. Lewis, "Discrete-time nonlinear HJB solution using approximate dynamic programming: convergence proof," *IEEE Trans. Systems, Man and Cybernetics, Part B: Cybernetics*, vol. 38, pp. 943-949, 2008.

[17] H. K. Khalil, Nonlinear System, third ed., Prentice-Hall, Upper Saddle River, New Jersey, 2002.

[18] P. J. Werbos, "A menu of designs for reinforcement learing over time," *J. Neural Network Contr.*, vol. 3, pp. 835-846, 1983.

[19] C. Watkins, "Learning from delayed rewards," Ph.D. dissertation, Cambridge University, England, 1989.

[20] K. S. Narendra and A. M. Annaswamy, *Stable Adaptive Systems*, Prentice-Hall, New Jersey. 1989.

[21] J. Si, A. G. Barto, W. B. Powell and D. Wunsch, *Handbook of Learning and Approximate Dynamics Programming*. New York: Wiley, 2004.

[22] F. L. Lewis, S. Jagannathan, and A. Yesilderek, *Neural Network Control of Robot Manipulator and Nonlinear Systems. London*, UK: Taylor & Francis, 1999.

[23] D. P. Bertsekas, *Dynamic programming and optimal control*, 3rd edition. Athena Scientific, 2007.

[24] D. Vrabie, O. Pastravanu, M. Abu-Khalaf and F. L. Lewis, "Adaptive optimal control for continuous-time linear systems based on policy iteration", *Automatica*, vol. 45, pp. 477-484, 2009.

[25] S.E. Lyshevski, "Optimal control of nonlinear continuous-time systems: design of bounded controllers via generazlied nonquadratic functionals," in *Proc. American Control Conf*, USA, pp. 205–209, 1998.

[26] Q. Zhao, H. Xu and S. Jagannathan, "Optimal adaptive controller scheme for uncertain quantized linear discrete-time system," *Proc. 51th IEEE Conf. Dec. Contr.*, Hawaii, 2012, pp. 6132–6137.

[27] Q. Zhao, H. Xu and S. Jagannathan, "Finite-Horizon Optimal Adaptive Neural Network Control of Uncertain Nonlinear Discrete-time Systems", in *Proc. of the IEEE Multi-Conference on Systems and Control*, pp. 41-46, August, 2013.

PAPER

I. FINITE-HORIZON OPTIMAL ADAPTIVE CONTROL OF UNCERTAIN LINEAR DISCRETE-TIME SYSTEMS

Qiming Zhao, Hao Xu and S. Jagannathan

Abstract — In this paper, the finite-horizon optimal adaptive control of linear discrete-time systems with unknown system dynamics is presented in a forward-in-time manner by using adaptive dynamic programming (ADP). An adaptive estimator (AE) is introduced with the idea of Q-learning to relax the requirement of system dynamics. The time-varying nature of the solution to the Bellman equation is handled by utilizing a time-dependent basis function while the terminal constraint is incorporated as part of the update law of the AE in solving the optimal feedback control. The proposed optimal regulation scheme of the uncertain linear system requires an initial admissible control input and yields a forward-in-time and online solution without using value and/or policy iterations. Furthermore, an adaptive observer is proposed so that the optimal adaptive control design depends only on the reconstructed states so as to realize an optimal output feedback control design. For the time invariant linear discrete-time systems, the closed-loop dynamics becomes non-autonomous and involved, but verified by using standard Lyapunov and Geometric sequence theory. Effectiveness of the proposed approach is verified by simulation results.

1. INTRODUCTION

Optimal control of linear systems with quadratic performance index or linear quadratic regulator (LQR) design has been one of the key research problems in control theory. Traditional optimal control [1] addresses finite-horizon problem in an offline and backward-in-time manner by solving Riccati equation (RE) when the system dynamics are known.

For optimal control of a linear system with infinite-horizon, the algebraic Riccati equation (ARE) is utilized to obtain its time invariant solution. In contrast, in finite-horizon scenario, the solution is inherently time-varying [1], and can only be obtained by solving the RE in a backward-in-time manner from the terminal weighting matrix given the full information of the system dynamics. In the absence of system dynamics, RE cannot be solved.

To address optimal regulation problem, model predictive control (MPC) has been widely investigated [2][3]. However, MPC are essentially open-loop control, and the prediction horizon needs to be carefully formulated. In the recent years, adaptive or neural network (NN) based optimal control has been intensely studied for both linear and nonlinear systems with uncertain dynamics in the case of infinite-horizon [4][5][6]. However, the finite-horizon optimal adaptive control of linear and nonlinear systems still remains an open problem for the control researchers when the system dynamics become uncertain. Moreover, solving the optimal control in a forward-in-time manner is quite challenging and involved.

In the past literature, the author in [7] considered the finite-horizon problem for nonlinear systems via iterating backward from terminal time t_f and by solving the socalled generalized Hamilton-Jacobi-Bellman (HJB) equation via Galerkin method. In [8], the authors proposed a fixed final time optimal control laws using NN with time-varying weights and state-dependent activation function to solve backward-in-time the time-varying HJB equation for affine nonlinear continuous-time systems.

In [9], the authors considered the finite-horizon optimal control problem with input-constraints by using standard direct heuristic dynamic programming (DHDP)-based offline NN scheme with constant NN weight matrix and time-varying activation function. The terminal constraint is satisfied by introducing the augmented vector incorporating the terminal value of the co-state $\lambda(\mathrm{N})$. On the other hand, in [10], the authors considered the discrete-time finite-horizon optimal problem under adaptive dynamic programming (ADP) scheme by using value and policy iterations. Here, the terminal time is not specified and final state is fixed at the origin.

The approaches in [7][8][9][10] provided a good insight into the finite-horizon optimal control problem while the solution is iterative, and either backward-in-time and/or offline. It is shown in [11] that iterative schemes require a significant number of iterations within a sampling interval for stability and are unsuitable for real-time control. However, a finite-horizon optimal scheme that can be implemented online and forward-in-time without using policy and value iterations is yet to be developed.

Motivated by the drawbacks aforementioned, in this work, the ADP technique via reinforcement learning (RL) is used to solve the finite-horizon optimal regulation of an uncertain linear discrete-time system in an online and forward-in-time manner without using value and/or policy iteration. The Bellman equation is utilized with an estimated Q-function such that the requirement for the system dynamics is relaxed.

An additional error term corresponding to the terminal constraint is defined and minimized at each time step. Lyapunov theory is utilized to show the stability of our proposed scheme under non-autonomous dynamic system framework. In the proposed scheme, the cost function and control input are updated once a sampling interval consistent with the standard adaptive control notion. In addition, in applications where the system states are unavailable for measurement, an adaptive observer is proposed such that the optimal state feedback controller design scheme can be extended to the output feedback case.

Therefore, the main contributions of this paper include the development of finite-horizon optimal adaptive control of uncertain linear discrete-time systems with state and output feedback via an observer. The terminal constraint is incorporated in the controller design. Boundedness of the regulation and parameter estimation errors are demonstrated by using Lyapunov and geometric sequence analysis. The proposed controller functions forward-in-time with no offline training phase. The controller does not use value and policy iterations while the cost function and optimal control input are updated once a sampling interval consistent with the standard adaptive control.

The remainder of this paper is organized as follows. In Section 2, the finite-horizon optimal control scheme for the uncertain linear discrete-time system along with the stability analysis is presented for the case of full state feedback. Section 2.3 extends the optimal control scheme to the uncertain linear discrete-time system by using output feedback. In Section 3, simulation results are shown to verify the feasibility of proposed method. Conclusions are provided in Section 4.

2. FINITE-HORIZON OPTIMAL CONTROL DESIGN UNDER Q-LEARNING SCHEME WITH STATE FEEDBACK

In this section, finite-horizon optimal control scheme for linear systems with uncertain system dynamics is proposed for the state feedback case. A Q-function [14][15] is first defined and then estimated adaptively by using RL, which is in turn utilized to design the controller by relaxing the system dynamics. An additional error term corresponding to the terminal constraint is also defined and minimized over time. Finally, the stability of the closed-loop system is verified based on the Lyapunov stability theory. The case when the system states are not measurable will be considered in section 2.3.

2.1 PROBLEM FORMULATION

Consider the time-invariant linear discrete-time system described in state-space form given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \tag{1}$$

where $\boldsymbol{x}_k \in \Omega_{\boldsymbol{x}} \subset \Re^n$ is the state vector and $\boldsymbol{u}_k \in \Omega_{\boldsymbol{u}} \subset \Re^m$ is the system state vector and control input vector at time step k, respectively, while the system matrices $\boldsymbol{A} \in \Re^{n\times n}$ and $\boldsymbol{B} \in \Re^{n\times m}$ are assumed to be *unknown*. Moreover, it is assumed that the control input matrix $\boldsymbol{B}$ satisfies $\|\boldsymbol{B}\|_F \le B_{\mathrm{M}}$, where $\|\bullet\|_F$ denotes the Frobenius norm.

In this paper, the free final state optimal regulation problem is addressed [1]. The objective of the controller design is to determine a feedback control policy that minimizes the following time-varying value or cost function

$$J_k = \boldsymbol{x}_{\mathrm{N}}^{\mathrm{T}} \boldsymbol{S}_{\mathrm{N}} \boldsymbol{x}_{\mathrm{N}} + \sum_{i=k}^{\mathrm{N}-1} \left(\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{P}_i \boldsymbol{x}_i + \boldsymbol{u}_i^{\mathrm{T}} \boldsymbol{R}_i \boldsymbol{u}_i \right) \tag{2}$$

where $\boldsymbol{P}_i \in \Re^{n\times n}$ is positive semi-definite matrix, $\boldsymbol{R}_i \in \Re^{m\times m}$ is positive definite matrix and assumed to be symmetric, respectively, whereas $\boldsymbol{S}_{\mathrm{N}} \in \Re^{n\times n}$ is the positive semi-definite symmetric penalty matrix for the terminal state $\boldsymbol{x}_{\mathrm{N}}$, with $[k,\mathrm{N}]$ being the time of interest while N is considered being the final time instant. It should be noted that in finite-horizon scenario, the control inputs becomes essentially time-dependent, i.e. $\boldsymbol{u}_k = \mu(\boldsymbol{x}_k, k)$.

Remark 1: Equation (2) gives the general form of the cost function in quadratic form. In the finite-horizon case, $\boldsymbol{S}_{\mathrm{N}}$ is known as the RE solution at the terminal step and $\boldsymbol{x}_{\mathrm{N}}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{N}}\boldsymbol{x}_{\mathrm{N}}$ is the terminal state constraint for the cost function. As $\mathrm{N}\to\infty$, the problem becomes infinite-horizon optimal control with $\boldsymbol{S}_{\mathrm{N}} = 0$ and the Riccati equation reduces to an algebraic Riccati equation (ARE).

It is well-known that from optimal control theory [1], the finite-horizon optimal control $\boldsymbol{u}_k^*$, can be obtained by solving the RE which is given by

$$\boldsymbol{S}_k = \boldsymbol{A}^{\mathrm{T}}[\boldsymbol{S}_{k+1} - \boldsymbol{S}_{k+1}\boldsymbol{B}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{B} + \boldsymbol{R}_k)^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_{k+1}]\boldsymbol{A} + \boldsymbol{P}_k \tag{3}$$

in a backward-in-time manner provided system matrices are known with time-varying Kalman gain matrix $\boldsymbol{K}_k^*$ given by

$$\boldsymbol{u}_k^* = -\boldsymbol{K}_k^*\boldsymbol{x}_k = -(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{B} + \boldsymbol{R}_k)^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{A}\cdot\boldsymbol{x}_k \tag{4}$$

Solving the RE equation when system matrices being unknown is a major challenge. In the next subsection, it will be shown that the finite-horizon optimal control for such linear discrete-time systems with uncertain dynamics can be tackled in a forward-in-time and online manner. In addition, value and/or policy iterations are not needed and the system dynamics are not required for the controller design since a Q-learning adaptive approach [15] is utilized.

2.2 FINITE-HORIZON OPTIMAL CONTROL DESIGN

In this subsection, a Q-function is estimated and subsequently utilized in obtaining the optimal control.

2.2.1 Q-function Setup. Before proceeding, it should be noted that for the finite-horizon case, the value-function, denoted as $V(\boldsymbol{x}_k, \mathrm{N}-k)$, becomes time-varying [1] and is a function of both system states and time-to-go function. Since the value function $V(\boldsymbol{x}_k, \mathrm{N}-k)$ is equal to the cost function J_k, according to [1], the value function can also be represented in the quadratic form of the system states for (1) as

$$V(\boldsymbol{x}_k, \mathrm{N}-k) = \boldsymbol{x}_k^{\mathrm{T}} \boldsymbol{S}_k \boldsymbol{x}_k \tag{5}$$

where $\boldsymbol{S}_k$ is the solution sequence to the Riccati equation.

According to the optimal control theory [1], define the Hamiltonian as

$$H(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k) = r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) + V(\boldsymbol{x}_{k+1}, \mathrm{N}-k-1) - V(\boldsymbol{x}_k, \mathrm{N}-k) \tag{6}$$

where $r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) = \boldsymbol{x}_k^{\mathrm{T}} \boldsymbol{P}_k \boldsymbol{x}_k + \boldsymbol{u}_k^{\mathrm{T}} \boldsymbol{R}_k \boldsymbol{u}_k$ is the time-varying cost-to-go function due to the time-dependency of the control inputs $\boldsymbol{u}_k$.

The optimal control input, according to [1], is given by using $\partial H(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k)/\partial \boldsymbol{u}_k = 0$, which yields (4). Instead of generating the optimal control input backward-in-time using (4), the value function is estimated and used to derive the control input in forward-in-time and an online manner without using value and policy iterations.

Define the time-varying optimal action dependent value function or Q-function, $Q(\boldsymbol{x}_k,\boldsymbol{u}_k,\mathrm{N}-k)$, as

$$Q(\boldsymbol{x}_k,\boldsymbol{u}_k,\mathrm{N}-k)=r(\boldsymbol{x}_k,\boldsymbol{u}_k,k)+J_{k+1}=\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix}^{\mathrm{T}}\boldsymbol{G}_k\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix} \tag{7}$$

The standard Bellman equation can be written as

$$\begin{aligned}\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix}^{\mathrm{T}}\boldsymbol{G}_k\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix}&=r(\boldsymbol{x}_k,\boldsymbol{u}_k,k)+J_{k+1}\\ &=\boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{P}_k\boldsymbol{x}_k+\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}_k\boldsymbol{u}_k+\boldsymbol{x}_{k+1}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{x}_{k+1}\\ &=\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}\boldsymbol{P}_k & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{R}_k\end{bmatrix}\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix}+(\boldsymbol{A}\boldsymbol{x}_k+\boldsymbol{B}\boldsymbol{u}_k)^{\mathrm{T}}\boldsymbol{S}_{k+1}(\boldsymbol{A}\boldsymbol{x}_k+\boldsymbol{B}\boldsymbol{u}_k)\\ &=\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}\boldsymbol{P}_k+\boldsymbol{A}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{A} & \boldsymbol{A}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{B}\\ \boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{A} & \boldsymbol{R}_k+\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{B}\end{bmatrix}\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix}\end{aligned} \tag{8}$$

Therefore, the *time-varying* matrix $\boldsymbol{G}_k$ can be expressed as

$$\boldsymbol{G}_k=\begin{bmatrix}\boldsymbol{P}_k+\boldsymbol{A}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{A} & \boldsymbol{A}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{B}\\ \boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{A} & \boldsymbol{R}_k+\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{B}\end{bmatrix}=\begin{bmatrix}\boldsymbol{G}_k^{\boldsymbol{xx}} & \boldsymbol{G}_k^{\boldsymbol{xu}}\\ \boldsymbol{G}_k^{\boldsymbol{ux}} & \boldsymbol{G}_k^{\boldsymbol{uu}}\end{bmatrix} \tag{9}$$

By using (9) and (4), the control gain matrix is expressed in terms of $\boldsymbol{G}_k$ as

$$\boldsymbol{K}_k=(\boldsymbol{G}_k^{\boldsymbol{uu}})^{-1}\boldsymbol{G}_k^{\boldsymbol{ux}} \tag{10}$$

From the above analysis, the time-varying Q-function $Q(\boldsymbol{x}_k,\boldsymbol{u}_k,\mathrm{N}-k)$ estimate includes the information of $\boldsymbol{G}_k$ matrix which can be obtained online. Subsequently, the control inputs can be obtained from (10) without using the knowledge of the system dynamics $\boldsymbol{A}$ and $\boldsymbol{B}$.

Remark 2: The above derivations are based on Bellman's principle of optimality under finite-horizon scenario. When the time span of interest goes to infinity, the solution of RE becomes a constant rather than time-varying, i.e., $\boldsymbol{S}_{k+1} \to \boldsymbol{S}$ when $k \to \infty$, where $\boldsymbol{S}$ is the constant solution matrix to the ARE [1].

Next, the process of estimating the time-varying Q function and the Kalman gain is introduced by using an adaptive approach.

2.2.2 Model-free Online Tuning with Q-function Estimator. To overcome the disadvantage of iteration-based scheme, in this subsection, the finite-horizon optimal control design is proposed by using the past information of the system states and control inputs. To properly satisfy the terminal constraint, an additional error term is introduced such that this error is also minimized. Before proceeding, the following assumption and lemma are introduced for the time-varying function approximation.

Assumption 1 [21] (*Linear-in-the-unknown-parameters*): The slowly time-varying Q-function $Q(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k)$ can be expressed as the linear in the unknown parameters (LIP).

By using adaptive control theory [21] and assumption 1, $Q(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k)$ can be expressed in vector form in vector form as

$$Q(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k) = \boldsymbol{z}_k^{\mathrm{T}} \boldsymbol{G}_k \boldsymbol{z}_k = \boldsymbol{g}_k^{\mathrm{T}} \bar{\boldsymbol{z}}_k \tag{11}$$

where $\boldsymbol{z}_k = [\boldsymbol{x}_k^{\mathrm{T}} \quad \boldsymbol{u}_k^{\mathrm{T}}]^{\mathrm{T}} \in \Re^l$, with $l = m + n$, $\bar{\boldsymbol{z}}_k = (z_{k1}^2, \cdots, z_{k1} z_{kl}, z_{k2}^2, \cdots, z_{kl-1} z_{kl}, z_{kl}^2)$ is the Kronecker product quadratic polynomial basis vector, and $\boldsymbol{g}_k = \mathrm{vec}(\boldsymbol{G}_k)$ with $\mathrm{vec}(\bullet)$ a vector function that acts on $l \times l$ matrices and gives a $l \times (l+1)/2 = L$ column vector. The output of $\mathrm{vec}(\boldsymbol{G}_k)$ is constructed by stacking the columns of the squared matrix into a one-column vector with the off-diagonal elements summed as $\boldsymbol{G}_{mn}^k + \boldsymbol{G}_{nm}^k$.

Lemma 1: Let $g(k)$ be a smooth and uniformly piecewise-continuous function in a compact set $\Omega \subset \Re$. Then, for each $\varepsilon > 0$, there exist constant elements $\theta_1,....,\theta_m \in \Re$ with $m \in N$ as well as the elements $\phi_1(k),...,\phi_m(k) \in \Re$ of basis function, such that

$$\left| g(k) - \sum_{i=1}^{m} \theta_i \phi_i(k) \right| < \varepsilon, \qquad k \in [0, \mathrm{N}] \tag{12}$$

Proof: Omitted due to the space limitation.

Based on Assumption 1 and Lemma 1, the smooth and uniformly piecewise-continuous function $\boldsymbol{g}_k$ can be represented as

$$\boldsymbol{g}_k = \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{\varphi}(\mathrm{N}-k) \tag{13}$$

where $\boldsymbol{\theta} \in \Re^L$ is target parameter vector and $\boldsymbol{\varphi}(\mathrm{N}-k) \in \Re^{L \times L}$ is the time-varying basis function matrix, with entries as functions of time-to-go, i.e., $\boldsymbol{\varphi}(\mathrm{N}-k) = \begin{bmatrix} \phi_{11}(\mathrm{N}-k) & \phi_{12}(\mathrm{N}-k) & \cdots & \phi_{1L}(\mathrm{N}-k) \\ \phi_{21}(\mathrm{N}-k) & \phi_{22}(\mathrm{N}-k) & \cdots & \phi_{2L}(\mathrm{N}-k) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{L1}(\mathrm{N}-k) & \phi_{L2}(\mathrm{N}-k) & \cdots & \phi_{LL}(\mathrm{N}-k) \end{bmatrix}$ with $\phi_{ij}(\mathrm{N}-k) = \exp(-\tanh(\mathrm{N}-k)^{L+1-j})$, for $i, j = 1,2,\cdots L$. This time-based function reflects the time-dependency nature of finite-horizon. Further, based on universal approximation theory, $\boldsymbol{\varphi}(\mathrm{N}-k)$ is piecewise-continuous [12][13].

From [1], the standard Bellman equation is given in terms of the Q-function as

$$Q(\boldsymbol{x}_{k+1}, \boldsymbol{u}_{k+1}, \mathrm{N}-k-1) - Q(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k) + r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) = 0 \tag{14}$$

However, (14) does not hold when the estimated value $\hat{\boldsymbol{g}}_k$ is applied. To approximate the time-varying matrix $\boldsymbol{G}_k$, or alternatively $\boldsymbol{g}_k$, define

$$\hat{\boldsymbol{g}}_k = \hat{\boldsymbol{\theta}}_k^{\mathrm{T}} \boldsymbol{\varphi}(\mathrm{N}-k) \tag{15}$$

where $\hat{\boldsymbol{\theta}}_k$ is the estimated value of target parameter vector $\boldsymbol{\theta}$.

Therefore, the approximation of the Q-function can be written as

$$\hat{Q}(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k) = \hat{\boldsymbol{g}}_k^{\mathrm{T}} \bar{\boldsymbol{z}}_k = \hat{\boldsymbol{\theta}}_k^{\mathrm{T}} \boldsymbol{\varphi}(\mathrm{N}-k) \bar{\boldsymbol{z}}_k = \hat{\boldsymbol{\theta}}_k^{\mathrm{T}} \overline{\boldsymbol{X}}_k \tag{16}$$

where $\overline{\boldsymbol{X}}_k = \boldsymbol{\varphi}(\mathrm{N}-k)\bar{\boldsymbol{z}}_k \in \Re^L$ is a time-dependent regression function incorporating the terminal time N while satisfying $\|\overline{\boldsymbol{X}}_k\| = 0$ when $\bar{\boldsymbol{z}}_k = \boldsymbol{0}$.

Remark 3: For the infinite-horizon case, (15) does not have the time-varying term $\boldsymbol{\varphi}(\mathrm{N}-k)$, since the desired value of vector $\boldsymbol{g}$ is constant, or time-invariant [5]. By contrast, for the finite-horizon case, the desired value of $\boldsymbol{g}_k$ is considered to be slowly time-varying. Hence the basis function should be a function of time and can take the form of product of the time-dependent basis function and system state vector [16].

With the approximated value of time-varying Q-function, the estimated Bellman equation can be written as

$$e_{k+1} = \hat{Q}(\boldsymbol{x}_{k+1}, \boldsymbol{u}_{k+1}, \mathrm{N}-k-1) - \hat{Q}(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k) + r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) \tag{17}$$

where e_{k+1} is the Bellman estimation error along the system trajectory.

Using the delayed value for convenience, from (16) and (17), we have

$$\begin{aligned} e_k &= r(\boldsymbol{x}_{k-1,} \boldsymbol{u}_{k-1}, k-1) + \hat{\boldsymbol{\theta}}_k^{\mathrm{T}} \overline{\boldsymbol{X}}_k - \hat{\boldsymbol{\theta}}_k^{\mathrm{T}} \overline{\boldsymbol{X}}_{k-1} \\ &= r(\boldsymbol{x}_{k-1}, \boldsymbol{u}_{k-1}, k-1) + \hat{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta\overline{\boldsymbol{X}}_{k-1} \end{aligned} \tag{18}$$

where $\Delta\overline{\boldsymbol{X}}_{k-1} = \overline{\boldsymbol{X}}_k - \overline{\boldsymbol{X}}_{k-1}$.

Next, define an auxiliary error vector which incorporates the history of cost-to-go errors as

$$\boldsymbol{\Xi}_k = \boldsymbol{\Gamma}_{k-1} + \hat{\boldsymbol{\theta}}_k^{\mathrm{T}} \boldsymbol{\Omega}_{k-1} \tag{19}$$

with $\boldsymbol{\Xi}_k = [e_k, e_{k-1}, \cdots, e_{k-j}] \in \Re^{1\times(j+1)}$, $\boldsymbol{\Gamma}_{k-1} = [r(\boldsymbol{x}_{k-1}, \boldsymbol{u}_{k-1}, k-1), r(\boldsymbol{x}_{k-2}, \boldsymbol{u}_{k-2}, k-2), \cdots,$ $r(\boldsymbol{x}_{k-1-j}, \boldsymbol{u}_{k-1-j}, k-1-j)] \in \Re^{1\times(j+1)}$ and $\boldsymbol{\Omega}_{k-1} = [\Delta\overline{\boldsymbol{X}}_{k-1}, \Delta\overline{\boldsymbol{X}}_{k-2}, \cdots, \Delta\overline{\boldsymbol{X}}_{k-1-j}] \in \Re^{L\times(j+1)}$ for $0 < j < k-1$.

It can be seen from (19) that $\boldsymbol{\Xi}_k$ includes a time history of previous $j+1$ Bellman estimation errors recalculated using the most recent $\hat{\boldsymbol{\theta}}_k$.

The dynamics of the auxiliary vector are generated similar to (19) as

$$\boldsymbol{\Xi}_{k+1} = \boldsymbol{\Gamma}_k + \hat{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}} \boldsymbol{\Omega}_k \tag{20}$$

In the finite-horizon optimal control problem, the terminal constraint of the cost function should also be taken in account. Define the estimated value function for the terminal stage as

$$\hat{Q}_k(\boldsymbol{x}_{\mathrm{N}}) = \hat{\boldsymbol{\theta}}_k^{\mathrm{T}} \boldsymbol{\varphi}(0) \overline{\boldsymbol{z}}_{\mathrm{N}} \tag{21}$$

In (21), it is important to note that the time-dependent basis function $\boldsymbol{\varphi}(\mathrm{N}-k)$ is taken as $\boldsymbol{\varphi}(0)$ since from the definition of $\boldsymbol{\varphi}$, the time index is taken in the reverse order.

Finally, define the error vector for the terminal constraint as

$$\boldsymbol{\Xi}_{k,\mathrm{N}} = \hat{\boldsymbol{g}}_{k,\mathrm{N}} - \boldsymbol{g}_{\mathrm{N}} = \hat{\boldsymbol{\theta}}_k^{\mathrm{T}} \boldsymbol{\varphi}(0) - \boldsymbol{g}_{\mathrm{N}} \tag{22}$$

with $\boldsymbol{g}_{\mathrm{N}}$ being bounded by $\|\boldsymbol{g}_{\mathrm{N}}\| \le g_{\mathrm{M}}$.

Remark 4: For finite-horizon case, the error term $\boldsymbol{\Xi}_{k,\mathrm{N}}$, which indicates the difference between the estimated value and true value for the terminal constraint, or “target”, (in our case, $\boldsymbol{g}_{\mathrm{N}}$), is critical for the controller design. The terminal constraint is satisfied by minimizing $\boldsymbol{\Xi}_{k,\mathrm{N}}$ along the system evolution. Another error term $\boldsymbol{\Xi}_k$, which can be regarded as temporal difference (TD) error, is always needed for tuning the parameter for both finite-horizon and infinite-horizon case. For infinite-horizon case, see [5] and [6].

Now define the total error vector as

$$\boldsymbol{\Xi}_{k,\mathrm{total}} = \boldsymbol{\Xi}_k + \boldsymbol{\Xi}_{k,\mathrm{N}} \tag{23}$$

To incorporate the terminal constraint, we further define

$$\Delta\overline{\boldsymbol{\Pi}}_k = \boldsymbol{\Omega}_k + \boldsymbol{\varphi}(0) \tag{24}$$

The update law for tuning $\hat{\boldsymbol{\theta}}_k$ is selected as

$$\hat{\boldsymbol{\theta}}_{k+1} = \Delta\overline{\boldsymbol{\Pi}}_k \left(\Delta\overline{\boldsymbol{\Pi}}_k^{\mathrm{T}} \Delta\overline{\boldsymbol{\Pi}}_k\right)^{-1} \left(\alpha \boldsymbol{\Xi}_{k,\mathrm{total}}^{\mathrm{T}} - \boldsymbol{\Gamma}_k^{\mathrm{T}} + \boldsymbol{g}_{\mathrm{N}}\right) \tag{25}$$

where $0 < \alpha < 1$ is the tuning rate. It also can be seen from (25) that the update law is essentially the least-squares update.

Expanding (25) with (23), (25) can be written as

$$\hat{\boldsymbol{\theta}}_{k+1} = \Delta\overline{\boldsymbol{\Pi}}_k \left(\Delta\overline{\boldsymbol{\Pi}}_k^{\mathrm{T}} \Delta\overline{\boldsymbol{\Pi}}_k\right)^{-1} \left(\alpha \boldsymbol{\Xi}_k^{\mathrm{T}} - \boldsymbol{\Gamma}_k^{\mathrm{T}} + \boldsymbol{g}_{\mathrm{N}}\right) + \alpha \Delta\overline{\boldsymbol{\Pi}}_k \left(\Delta\overline{\boldsymbol{\Pi}}_k^{\mathrm{T}} \Delta\overline{\boldsymbol{\Pi}}_k\right)^{-1} \boldsymbol{\Xi}_{k,\mathrm{N}}^{\mathrm{T}} \tag{26}$$

Recall from $\Delta\overline{\boldsymbol{\Pi}}_k = \boldsymbol{\Omega}_k + \boldsymbol{\varphi}(0)$, and substituting $\boldsymbol{\Omega}_k = \Delta\overline{\boldsymbol{\Pi}}_k - \boldsymbol{\varphi}(0)$ into (20) yields

$$\begin{aligned}\boldsymbol{\Xi}_{k+1} &= \boldsymbol{\Gamma}_k + \hat{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}} \boldsymbol{\Omega}_k = \boldsymbol{\Gamma}_k + \hat{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}} (\Delta\overline{\boldsymbol{\Pi}}_k - \boldsymbol{\varphi}(0)) \\ &= \boldsymbol{\Gamma}_k + \hat{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}} \Delta\overline{\boldsymbol{\Pi}}_k - \hat{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}} \boldsymbol{\varphi}(0) \\ &= \boldsymbol{\Gamma}_k + \hat{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}} \Delta\overline{\boldsymbol{\Pi}}_k - \hat{\boldsymbol{g}}_{k+1,\mathrm{N}}\end{aligned} \tag{27}$$

To find the error dynamics, substituting (26) into (27), we have

$$\begin{aligned}\boldsymbol{\Xi}_{k+1} &= \boldsymbol{\Gamma}_k + \hat{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}} \Delta\overline{\boldsymbol{\Pi}}_k - \hat{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}} \boldsymbol{\varphi}(0) \\ &= \alpha \boldsymbol{\Xi}_k + \alpha \boldsymbol{\Xi}_{k,\mathrm{N}} - \hat{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}} \boldsymbol{\varphi}(0)\end{aligned} \tag{28}$$

From(28), it can be seen that the Bellman estimation error is coupled with the terminal constraint estimation error. Hence, the dynamics for $\boldsymbol{\mathit{\Xi}}_{k,\mathrm{total}}$ is given by

$$\begin{aligned}\boldsymbol{\mathit{\Xi}}_{k+1,\mathrm{total}} &= \boldsymbol{\mathit{\Xi}}_{k+1} + \boldsymbol{\mathit{\Xi}}_{k+1,\mathrm{N}} \\ &= \alpha\boldsymbol{\mathit{\Xi}}_{k} + \alpha\boldsymbol{\mathit{\Xi}}_{k,\mathrm{N}} - \hat{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}}\boldsymbol{\varphi}(0) + \boldsymbol{\mathit{\Xi}}_{k+1,\mathrm{N}} \\ &= \alpha\boldsymbol{\mathit{\Xi}}_{k} + \alpha\boldsymbol{\mathit{\Xi}}_{k,\mathrm{N}}\end{aligned} \tag{29}$$

Define the approximation error for $\hat{\boldsymbol{\theta}}_k$ as

$$\tilde{\boldsymbol{\theta}}_k = \boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_k \tag{30}$$

Recall from (14) and the Bellman equation, we have the utility vector as

$$\boldsymbol{\mathit{\Gamma}}_k = -\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\mathit{\Omega}}_k \tag{31}$$

Substituting (31) into (20) yields

$$\begin{aligned}\boldsymbol{\mathit{\Xi}}_{k+1} &= \boldsymbol{\mathit{\Gamma}}_k + \hat{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}}\boldsymbol{\mathit{\Omega}}_k = -\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\mathit{\Omega}}_k + \hat{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}}\boldsymbol{\mathit{\Omega}}_k \\ &= -\tilde{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}}\boldsymbol{\mathit{\Omega}}_k\end{aligned} \tag{32}$$

Recalling that $\boldsymbol{\mathit{\Xi}}_{k+1,\mathrm{total}} = \boldsymbol{\mathit{\Xi}}_{k+1} + \boldsymbol{\mathit{\Xi}}_{k+1,\mathrm{N}}$, we further have

$$\tilde{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}}\boldsymbol{\mathit{\Omega}}_k = -\alpha\boldsymbol{\mathit{\Xi}}_k - \alpha\boldsymbol{\mathit{\Xi}}_{k,\mathrm{N}} + \boldsymbol{\mathit{\Xi}}_{k+1,\mathrm{N}} \tag{33}$$

Note that $\boldsymbol{\mathit{\Xi}}_{k,\mathrm{N}} = \hat{\boldsymbol{g}}_{k,\mathrm{N}} - \boldsymbol{g}_{\mathrm{N}} = \hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(0) - \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\varphi}(0) = -\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(0)$, and similarly $\boldsymbol{\mathit{\Xi}}_{k+1,\mathrm{N}} = -\tilde{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}}\boldsymbol{\varphi}(0)$, then (33) becomes

$$\tilde{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}}\boldsymbol{\mathit{\Omega}}_k = \alpha\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\mathit{\Omega}}_{k-1} + \alpha\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(0) - \tilde{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}}\boldsymbol{\varphi}(0) \tag{34}$$

Therefore, we have

$$\tilde{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}}(\boldsymbol{\mathit{\Omega}}_k + \boldsymbol{\varphi}(0)) = \alpha\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}(\boldsymbol{\mathit{\Omega}}_{k-1} + \boldsymbol{\varphi}(0)) \tag{35}$$

Recall from (24) that $\Delta\overline{\boldsymbol{\mathit{\Pi}}}_k = \boldsymbol{\mathit{\Omega}}_k + \boldsymbol{\varphi}(0)$, (35) can be finally expressed as

$$\tilde{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}}\Delta\overline{\boldsymbol{\mathit{\Pi}}}_k = \alpha\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\Delta\overline{\boldsymbol{\mathit{\Pi}}}_{k-1} \tag{36}$$

Remark 5: It is observed, from the definition (16), that when the system states, which are the inputs to the time-varying Q-function estimator, have converged to zero, the Q-function approximation is no longer updated. It can be seen as a persistency of excitation (PE) requirement for the inputs to the Q-function estimator wherein the system states must be persistently exiting long enough for the estimator to learn the Q-function. The PE condition requirement is standard in adaptive control and can be satisfied by adding exploration noise [20] to the augmented system state vector. In this paper, exploration noise is added to satisfy the PE condition [5]. When the estimator effectively learns the Q-function, the PE can be removed thus the terminal state will not affected by the addition of the noise signal.

Next, the estimation of the optimal feedback control input and the entire scheme is introduced.

2.2.3 Estimation of the Optimal Feedback Control and Algorithm. Before proceeding, the flowchart of proposed scheme is shown in Figure 1. We start our proposed algorithm with an initial admissible control which is defined next. After collecting both the Bellman error and terminal constraint error, the parameters for the adaptive estimator are updated once a sampling interval beginning with an initial time and until the terminal time instant in an online and forward-in-time fashion. Next, the following assumption and definition are needed.

Assumption 2: The system $(\boldsymbol{A}, \boldsymbol{B})$ is controllable and system states $\boldsymbol{x}_k \in \Omega_{\boldsymbol{x}}$ are measurable.

Definition 1 [4]: Let $\Omega_{\boldsymbol{u}}$ denote the set of admissible control. A control function $\boldsymbol{u} : \Re^n \rightarrow \Re^m$ is defined to be admissible if the following is true:

$\boldsymbol{u}$ is continuous on $\Omega_{\boldsymbol{u}}$;

$\boldsymbol{u}(\boldsymbol{x})\big|_{\boldsymbol{x}=0}=0$;

$\boldsymbol{u}(\boldsymbol{x})$ stabilize the system (1) on $\Omega_{\boldsymbol{x}}$;

$J(\boldsymbol{x}(0),\boldsymbol{u})<\infty, \forall \boldsymbol{x}(0)\in\Omega_{\boldsymbol{x}}$.

Since the design scheme is similar to policy iteration, we need to solve a fixed-point equation rather than recursive equation. The initial admissible control guarantees the solution of the fixed-potion equation exists, thus the approximation process can be effectively done by our proposed scheme.

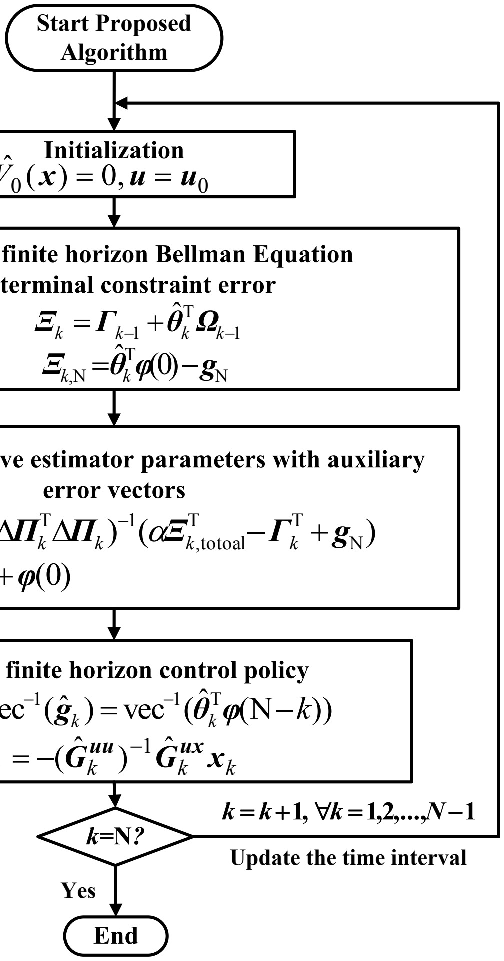

Start Proposed Algorithm
Initialization
$\hat{V}_0(\boldsymbol{x})=0, \boldsymbol{u}=\boldsymbol{u}_0$
Update the finite horizon Bellman Equation and terminal constraint error
$\boldsymbol{\Xi}_k=\boldsymbol{\Gamma}_{k-1}+\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\Omega}_{k-1}$
$\boldsymbol{\Xi}_{k,\mathrm{N}}=\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(0)-\boldsymbol{g}_{\mathrm{N}}$
Update the adaptive estimator parameters with auxiliary error vectors
$\hat{\boldsymbol{\theta}}_{k+1}=\Delta\boldsymbol{\Pi}_k(\Delta\boldsymbol{\Pi}_k^{\mathrm{T}}\Delta\boldsymbol{\Pi}_k)^{-1}(\alpha\boldsymbol{\Xi}_{k,\mathrm{totoal}}^{\mathrm{T}}-\boldsymbol{\Gamma}_k^{\mathrm{T}}+\boldsymbol{g}_{\mathrm{N}})$
with $\Delta\boldsymbol{\Pi}_k=\boldsymbol{\Omega}_k+\boldsymbol{\varphi}(0)$
Update finite horizon control policy
$\hat{\boldsymbol{G}}_k=\mathrm{vec}^{-1}(\hat{\boldsymbol{g}}_k)=\mathrm{vec}^{-1}(\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(\mathrm{N}-k))$
$\hat{\boldsymbol{u}}_k=-(\hat{\boldsymbol{G}}_k^{\boldsymbol{uu}})^{-1}\hat{\boldsymbol{G}}_k^{\boldsymbol{ux}}\boldsymbol{x}_k$
k=N?
$\boldsymbol{k}=\boldsymbol{k}+1, \forall \boldsymbol{k}=1,2,\ldots,\boldsymbol{N}-1$
Update the time interval
Yes
End

Figure 1. Flowchart of the finite-horizon optimal control design

Recall from (10), the estimated optimal control input can be obtained as

$$\hat{\boldsymbol{u}}_k = -(\hat{\boldsymbol{G}}_k^{\boldsymbol{uu}})^{-1} \hat{\boldsymbol{G}}_k^{\boldsymbol{ux}} \cdot \boldsymbol{x}_k \tag{37}$$

From (37), it can be seen that the Kalman gain can be calculated based on the information of $\hat{\boldsymbol{G}}_k$ matrix, which is obtained by estimating the Q-function. This relaxes the requirement of the system dynamics while (25) relaxes the value and policy iterations. Here the Q-function (11) and control policy (37) are updated once a sampling interval.

2.2.4 Stability Analysis. In this subsection, it will be shown that both the estimation error $\tilde{\boldsymbol{\theta}}_k$ and the closed-loop system are uniformly ultimately bounded (UUB). Due to the nature of time-dependency, the system becomes essentially non-autonomous in contrast with [9] and [10]. First, the boundedness of estimation error $\tilde{\boldsymbol{\theta}}_k$ will be shown in Theorem 1. Before proceeding, the following definition is needed.

Definition 2 [17]: An equilibrium point $\boldsymbol{x}_e$ is said to be *uniformly ultimately bounded* (UUB) if there exists a compact set $\Omega_{\boldsymbol{x}} \subset \Re^n$ so that for all $\boldsymbol{x}_0 \in \Omega_{\boldsymbol{x}}$, there exists a bound B and a time $T(B, \boldsymbol{x}_0)$ such that $\|\boldsymbol{x}_k - \boldsymbol{x}_e\| \le B$ for all $k \ge k_0 + T$.

Theorem 1: Let the initial conditions for the Q-function estimator vectors $\hat{\boldsymbol{g}}_0$ be bounded in the set Ω_g which contains the ideal parameter vector $\boldsymbol{g}_k$. Given $\boldsymbol{u}_0(k) \in \Omega_{\boldsymbol{u}}$ an initial admissible control policy for the linear system (1). Let the assumptions stated in the paper hold including the controllability of the system (1) and system states vector $\boldsymbol{x}_k \in \Omega_x$ being measurable. Let the update law for tuning $\hat{\boldsymbol{\theta}}_k$ be given by (25). Then, there exists a positive constant α satisfying $0 < \alpha < 1$ such that the stability of the Q-function estimator is guaranteed at the terminal stage N. Furthermore, when the time of interest goes to infinity, i.e., $k \to \infty$, the parameter estimation error $\tilde{\boldsymbol{\theta}}_k$ will converge to zero asymptotically.

Proof: See Appendix.

Next, we will show the boundedness of the closed-loop system. Before establishing the theorem on system stability, the following lemma is needed.

Lemma 2 [5]: (*Bounds on the closed-loop dynamics with optimal control*): Let the optimal control policy, $\boldsymbol{u}_k^* \in \Omega_{\boldsymbol{u}}$, be applied to the linear discrete-time system (1). Then, the closed-loop system dynamics $\boldsymbol{Ax}_k + \boldsymbol{Bu}_k$ are bounded above with the bounds satisfying

$$\left\| \boldsymbol{Ax}_k + \boldsymbol{Bu}_k^* \right\|^2 \le \rho \left\| \boldsymbol{x}_k \right\|^2 \tag{38}$$

where $0 < \rho < \dfrac{1}{2}$ is a constant.

Proof: See [5].

Theorem 2 (*Boundedness of the Closed-loop System*): Let the linear discrete-time system (1) be controllable and the system states be measurable. Let the initial conditions for the Q-function estimator vectors $\hat{\boldsymbol{g}}_0$ be bounded in the set Ω_g which contains the ideal parameter vector $\boldsymbol{g}_k$. Let $\boldsymbol{u}_0(k) \in \Omega_{\boldsymbol{u}}$ be an initial admissible controlpolicy for the system such that (38) holds with some ρ. Let the parameter vector of Q-function estimator be tuned and the control policy estimation be provided by (25) and (37), respectively. Then, there exists a constant α satisfying $0 < \alpha < 1$ such that the closed-loop system is uniformly bounded at the terminal stage N. Further, when $k \to \infty$, the bounds for both the states and estimation error will converge to zero asymptotically.

Moreover, due to (A.14), the estimated control input will converge to ideal optimal control (i.e. $\hat{\boldsymbol{u}}_k \to \boldsymbol{u}_k^*$) while time goes to infinity (i.e. $k \to \infty$).

Proof: See Appendix.

2.3 FINITE-HORIZON OPTIMAL CONTROL WITH OUTPUT FEEDBACK

In this section, the finite-horizon optimal adaptive control scheme for the linear discrete-time systems with uncertain dynamics is derived with an adaptive observer when the system states are not measurable.

Consider the system

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \\ \boldsymbol{y}_k &= \boldsymbol{C}\boldsymbol{x}_k \end{aligned} \tag{39}$$

where $\boldsymbol{x}_k \in \Omega_{\boldsymbol{x}} \subset \Re^n$, $\boldsymbol{u}_k \in \Omega_{\boldsymbol{u}} \subset \Re^m$ and $\boldsymbol{y}_k \in \Omega_{\boldsymbol{y}} \subset \Re^p$ are the system states, control input and system output, respectively, while the system matrices $\boldsymbol{A} \in \Re^{n\times n}$ and $\boldsymbol{B} \in \Re^{n\times m}$ are assumed to be *unknown*, and output matrix $\boldsymbol{C} \in \Re^{p\times n}$ is assumed to be known.

Then, the cost function is given as

$$\begin{aligned} J_k &= \boldsymbol{y}_{\mathrm{N}}^{\mathrm{T}}\boldsymbol{M}_{\mathrm{N}}\boldsymbol{y}_{\mathrm{N}} + \sum_{i=k}^{\mathrm{N}-1}\left(\boldsymbol{y}_i^{\mathrm{T}}\boldsymbol{H}_i\boldsymbol{y}_i + \boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{R}_i\boldsymbol{u}_i\right) \\ &= \boldsymbol{x}_{\mathrm{N}}^{\mathrm{T}}\boldsymbol{C}^{\mathrm{T}}\boldsymbol{M}_{\mathrm{N}}\boldsymbol{C}\boldsymbol{x}_{\mathrm{N}} + \sum_{i=k}^{\mathrm{N}-1}\left(\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{C}^{\mathrm{T}}\boldsymbol{H}_i\boldsymbol{C}\boldsymbol{x}_i + \boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{R}_i\boldsymbol{u}_i\right) \\ &= \boldsymbol{x}_{\mathrm{N}}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{N}}\boldsymbol{x}_{\mathrm{N}} + \sum_{i=k}^{\mathrm{N}-1}\left(\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{P}_i\boldsymbol{x}_i + \boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{R}_i\boldsymbol{u}_i\right) \end{aligned} \tag{40}$$

where $\boldsymbol{S}_{\mathrm{N}} = \boldsymbol{C}^{\mathrm{T}}\boldsymbol{M}_{\mathrm{N}}\boldsymbol{C}$ and $\boldsymbol{P}_i = \boldsymbol{C}^{\mathrm{T}}\boldsymbol{H}_i\boldsymbol{C}$.

Assumption 3: The system $(\boldsymbol{A},\boldsymbol{B})$ is controllable and $(\boldsymbol{A},\boldsymbol{C})$ is observable. Moreover, the system output vector $\boldsymbol{y}_k \in \Omega_y$ is measurable.

Define an adaptive observer as

$$\hat{\boldsymbol{x}}_{k+1} = \hat{\boldsymbol{A}}\hat{\boldsymbol{x}}_k + \hat{\boldsymbol{B}}\boldsymbol{u}_k - \hat{\boldsymbol{L}}_k(\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k) \tag{41}$$

where $\hat{\boldsymbol{x}}_k \in \Omega_{\hat{x}} \subset \Re^n$ is the reconstructed system states, $\hat{\boldsymbol{A}}$, $\hat{\boldsymbol{B}}$ are estimated system dynamics and $\hat{\boldsymbol{L}}_k$ is the observer gain.

The observer error is given by

$$\begin{aligned} \tilde{\boldsymbol{x}}_{k+1} = \boldsymbol{x}_{k+1} - \hat{\boldsymbol{x}}_{k+1} &= (\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C})\tilde{\boldsymbol{x}}_k + [\tilde{\boldsymbol{A}}_k \quad \tilde{\boldsymbol{B}}_k \quad \tilde{\boldsymbol{L}}_k]\begin{bmatrix} \hat{\boldsymbol{x}}_k \\ \boldsymbol{u}_k \\ \tilde{\boldsymbol{y}}_k \end{bmatrix} \\ &= (\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C})\tilde{\boldsymbol{x}}_k + \tilde{\vartheta}_k^{\mathrm{T}}\psi(\boldsymbol{z}_k) \end{aligned} \tag{42}$$

where $\tilde{\vartheta}_k^{\mathrm{T}} = [\tilde{\boldsymbol{A}}_k \quad \tilde{\boldsymbol{B}}_k \quad \tilde{\boldsymbol{L}}_k] \in \Re^{n \times r}$, $r = m + n + p$, $\tilde{\boldsymbol{A}}_k = \boldsymbol{A} - \hat{\boldsymbol{A}}_k$, $\tilde{\boldsymbol{B}}_k = \boldsymbol{B} - \hat{\boldsymbol{B}}_k$, $\tilde{\boldsymbol{L}}_k = \boldsymbol{L} - \hat{\boldsymbol{L}}_k$, and $\psi(\boldsymbol{z}_k) = [\hat{\boldsymbol{x}}_k^{\mathrm{T}} \quad \boldsymbol{u}_k^{\mathrm{T}} \quad \tilde{\boldsymbol{y}}_k^{\mathrm{T}}]^{\mathrm{T}} \in \Re^r$. Note that in (42), since system (39) is observable, there always exists an observer gain $\boldsymbol{L} \in \Re^{n \times p}$ such that $\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}$ is Hurwitz. Hence, the first term in (42) is always stable. We need to design for $\tilde{\vartheta}_k$ such that the stability of second term in (42) can be ensured.

Next, define an auxiliary observer error as

$$\boldsymbol{C}^{+}\tilde{\boldsymbol{Y}}_{k+1} = (\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C})\boldsymbol{C}^{+}\tilde{\boldsymbol{Y}}_k + \tilde{\vartheta}_k^{\mathrm{T}}\boldsymbol{\zeta}(\boldsymbol{z}_k) \tag{43}$$

where $\boldsymbol{C}^{+} \in \Re^{n \times p}$ is the pseudo inverse of $\boldsymbol{C}$, $\tilde{\boldsymbol{Y}}_k = [\tilde{\boldsymbol{y}}_k \quad \tilde{\boldsymbol{y}}_{k-1} \quad \cdots \quad \tilde{\boldsymbol{y}}_{k-v-1}] \in \Re^{p \times v}$, $\boldsymbol{\zeta}(\boldsymbol{z}_k) = [\psi(\boldsymbol{z}_k) \quad \psi(\boldsymbol{z}_{k-1}) \quad \cdots \quad \psi(\boldsymbol{z}_{k-v-1})] \in \Re^{(n+m+p) \times v}$ and v is the observability index. The update law for the proposed adaptive observer is given as

$$\hat{\vartheta}_{k+1} = \hat{\vartheta}_k + \beta\frac{\boldsymbol{\zeta}(\boldsymbol{z}_k)\tilde{\boldsymbol{Y}}_{k+1}^{\mathrm{T}}}{1 + \left\|\boldsymbol{\zeta}^{\mathrm{T}}(\boldsymbol{z}_k)\boldsymbol{\zeta}(\boldsymbol{z}_k)\right\|^2} \tag{44}$$

where $\beta > 0$ is the tuning rate.

The parameter estimation error can be revealed to be

$$\tilde{\vartheta}_{k+1} = \tilde{\vartheta}_k - \beta \frac{\zeta(z_k)\tilde{\boldsymbol{Y}}_{k+1}^{\mathrm{T}}}{1 + \left\| \zeta^{\mathrm{T}}(z_k)\zeta(z_k) \right\|^2} \tag{45}$$

Next, the boundedness of the parameter estimation error for the adaptive observer is demonstrated, as shown in the following Lemma.

Lemma 3 (*Boundedness of the parameter estimation error for the adaptive observer*): Let the linear discrete-time system (39) be controllable and observable while its output is measurable. Let the initial conditions for the Q-function estimator vectors $\hat{\boldsymbol{g}}_0$ be bounded in the set Ω_g which contains the ideal parameter vector $\boldsymbol{g}_k$. Let the adaptive observer be given by (41) and the update law for the parameter estimation be provided as in (44). Then there exists a positive constant β satisfying $0 < \beta < \dfrac{\left\| \zeta(z_k) \right\|^2}{(1 + \left\| \zeta(z_k) \right\|^2)(1 + \left\| \psi(z_k) \right\|^2) + 1}$ such that given any positive $\varepsilon > 0$, there exists a finite N such that $\left\| \tilde{\vartheta}_k \right\| \le \varepsilon(\tilde{\vartheta}_k, \mathrm{N})$. Furthermore, when $k \to \infty$, the adaptive observer is asymptotically stable.

Proof: See Appendix.

Our objective is still trying to approximate the matrix $\boldsymbol{G}_k$, or equivalently, $\boldsymbol{g}_k$. Based on the proposed adaptive observer design, the total error can be derived from Bellman equation as

$$e_{\mathrm{total},k+1} = \begin{bmatrix} \hat{\boldsymbol{x}}_k \\ \boldsymbol{u}_k \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} \boldsymbol{P}_k + \hat{\boldsymbol{A}}_k^{\mathrm{T}} \hat{\boldsymbol{S}}_{k+1} \hat{\boldsymbol{A}}_k & \hat{\boldsymbol{A}}_k^{\mathrm{T}} \hat{\boldsymbol{S}}_{k+1} \hat{\boldsymbol{B}}_k \\ \hat{\boldsymbol{B}}_k^{\mathrm{T}} \hat{\boldsymbol{S}}_{k+1} \hat{\boldsymbol{A}}_k & \boldsymbol{R}_k + \hat{\boldsymbol{B}}_k^{\mathrm{T}} \hat{\boldsymbol{S}}_{k+1} \hat{\boldsymbol{B}}_k \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{x}}_k \\ \boldsymbol{u}_k \end{bmatrix}$$

$$\begin{aligned} & -\hat{\boldsymbol{x}}_k^{\mathrm{T}}\hat{\boldsymbol{S}}_k\hat{\boldsymbol{x}}_k + 2(\hat{\boldsymbol{A}}_k\hat{\boldsymbol{x}}_k + \hat{\boldsymbol{B}}_k\boldsymbol{u}_k)^{\mathrm{T}}\hat{\boldsymbol{S}}_{k+1}\hat{\boldsymbol{L}}_k\tilde{\boldsymbol{y}}_k \\ & -\tilde{\boldsymbol{y}}_k^{\mathrm{T}}\hat{\boldsymbol{L}}_k^{\mathrm{T}}\hat{\boldsymbol{S}}_{k+1}\hat{\boldsymbol{L}}_k\tilde{\boldsymbol{y}}_k + (\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(0) - \boldsymbol{g}_{\mathrm{N}})\boldsymbol{O} \\ = & -\hat{\boldsymbol{\theta}}_k^{\mathrm{T}}\Delta\hat{\hat{\boldsymbol{X}}}_k + r(\hat{\boldsymbol{x}}_k, \boldsymbol{u}_k, k) - \tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(0)\boldsymbol{O} \end{aligned} \tag{46}$$

where $\Delta\hat{\hat{\boldsymbol{X}}}_k$ is defined similarly as state feedback case but using the reconstructed system states $\hat{\boldsymbol{x}}_k$, $\boldsymbol{O}$ is a row vector consisting of “1”s, i.e., $\boldsymbol{O} = [1, 1, \cdots, 1]$.

Adding and subtracting $r(\hat{\boldsymbol{x}}_k, \boldsymbol{u}_k, k)$, (46) further becomes

$$e_{\mathrm{total},k+1} = \tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\sigma}_k + \boldsymbol{\theta}^{\mathrm{T}}[f(\boldsymbol{x}_k) - f(\hat{\boldsymbol{x}}_k)] + g(\boldsymbol{x}_k) - g(\hat{\boldsymbol{x}}_k) \tag{47}$$

where $\boldsymbol{\sigma}_k = \Delta\hat{\hat{\boldsymbol{X}}}_k - \boldsymbol{\varphi}(0)\boldsymbol{O}$, $f(\boldsymbol{x}_k) = \mathrm{kron}([\boldsymbol{x}_k, \boldsymbol{u}_k])$, $f(\hat{\boldsymbol{x}}_k) = \mathrm{kron}([\hat{\boldsymbol{x}}_k, \boldsymbol{u}_k])$, with kron(•) denoting the Kronecker product quadratic polynomial, and $g(\boldsymbol{x}_k) = \boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{P}_k\boldsymbol{x}_k + \boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}_k\boldsymbol{u}_k$, $g(\hat{\boldsymbol{x}}_k) = \hat{\boldsymbol{x}}_k^{\mathrm{T}}\boldsymbol{P}_k\hat{\boldsymbol{x}}_k + \boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}_k\boldsymbol{u}_k$.

Notice that since $f(\boldsymbol{x}_k)$ and $g(\boldsymbol{x}_k)$ are in quadratic form and hence satisfy Lipschitz condition, i.e., $\|f(\boldsymbol{x}_k) - f(\hat{\boldsymbol{x}}_k)\| \le L_f\|\tilde{\boldsymbol{x}}_k\|$ and $\|g(\boldsymbol{x}_k) - g(\hat{\boldsymbol{x}}_k)\| \le L_g\|\tilde{\boldsymbol{x}}_k\|$, where L_f and L_g are positive Lipchitz constants [18].

The update law for Q-function estimator is finally given as

$$\hat{\boldsymbol{\theta}}_{k+1} = \hat{\boldsymbol{\theta}}_k + \gamma\frac{\boldsymbol{\sigma}_k e_{\mathrm{total},k+1}}{\boldsymbol{\sigma}_k^{\mathrm{T}}\boldsymbol{\sigma}_k + 1} \tag{48}$$

with $\gamma > 0$ the tuning rate. Furthermore, the error dynamics for $\hat{\boldsymbol{\theta}}_k$ can be found to be

$$\tilde{\boldsymbol{\theta}}_{k+1} = \tilde{\boldsymbol{\theta}}_k - \gamma\frac{\boldsymbol{\sigma}_k e_{\mathrm{total},k+1}}{\boldsymbol{\sigma}_k^{\mathrm{T}}\boldsymbol{\sigma}_k + 1} \tag{49}$$

Remark 6: For output feedback, the error term e_{total} is guaranteed to converge due to the convergence of $\tilde{\boldsymbol{\theta}}_k$, which is shown in the closed-loop proof given in the appendix. Note that Lemma 3 guarantees the convergence of the observer error $\tilde{\boldsymbol{x}}_k$ and hence the optimality is ensured by the update law for tuning $\hat{\boldsymbol{\theta}}_k$ together with the proposed adaptive observer design scheme.

Next, the boundedness of the closed-loop system will be shown in next theorem.

Theorem 3: (*Boundedness of the Closed-loop System with adaptive observer*): Let the linear discrete-time system (39) be controllable and observable while its output is measurable. Let the initial conditions for the Q-function estimator vectors $\hat{\boldsymbol{g}}_0$ be bounded in the set Ω_g which contains the ideal parameter vector $\boldsymbol{g}_k$. Let $\boldsymbol{u}_0(k) \in \Omega_{\boldsymbol{u}}$ be an initial admissible control policy for the system (39). Let the parameter vector of the adaptive observer and Q-function estimator be tuned by (44) and (48), respectively. Then, there exists a constant γ satisfying $0 < \gamma < \frac{1}{5}$ such that given any $\varepsilon > 0$, there exists a final time instant N such that $\|\boldsymbol{x}_k\| \le \varepsilon(\boldsymbol{x}_k, \mathrm{N})$, $\|\tilde{\boldsymbol{x}}_k\| \le \varepsilon(\tilde{\boldsymbol{x}}_k, \mathrm{N})$, $\|\tilde{\boldsymbol{\theta}}_k\| \le \varepsilon(\tilde{\boldsymbol{\theta}}_k, \mathrm{N})$ and $\|\tilde{\vartheta}_k\| \le \varepsilon(\tilde{\vartheta}_k, \mathrm{N})$. Furthermore, when $k \to \infty$, the closed-loop system is asymptotically stable. Moreover, due to (A.21), the estimated control input will converge close to optimal control input within the final time instant N and $\hat{\boldsymbol{u}}_k \to \boldsymbol{u}_k^*$ as $k \to \infty$.

Proof: See Appendix.

3. SIMULATION RESULTS

In this section, a practical example for the case of both state feedback output feedback are given to show the feasibility of our proposed finite-horizon optimal control design.

3.1 FINITE-HORIZON OPTIMAL CONTROL WITH STATE FEEDBACK

First, the proposed Q-learning-based finite-horizon optimal control design for state feedback case is evaluated by a numerical example. The example is taken as a continuous-time F-16 aircraft plant with quadratic cost function given by [19]:

$$\dot{\boldsymbol{x}} = \begin{bmatrix} -1.01887 & 0.90506 & -0.00215 \\ 0.82225 & -1.07741 & -0.17555 \\ 0 & 0 & -20.2 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 0 \\ 0 \\ 20.2 \end{bmatrix} \boldsymbol{u} \tag{50}$$

The system state vector is $\boldsymbol{x} = [\alpha \quad q \quad \delta_e]$, where α is the angle of attack, q is the pitch rate and δ_e is the elevator deflection angle. The control input is the elevator actuator voltage.

Discretizing the system with a sampling interval of $T_s = 0.1\sec$, the discrete-time linear system is given by

$$\boldsymbol{x}_{k+1} = \begin{bmatrix} 0.9065 & 0.0816 & -0.0009 \\ 0.0741 & 0.9012 & -0.0159 \\ 0 & 0 & 0.9048 \end{bmatrix} \boldsymbol{x}_k + \begin{bmatrix} 0 \\ -0.0008 \\ 0.0952 \end{bmatrix} \boldsymbol{u}_k \tag{51}$$

The performance index is given as (2) with the weighting matrices $\boldsymbol{P}_k$, $\boldsymbol{R}_k$ and the terminal constraint matrix $\boldsymbol{S}_\mathrm{N}$ are selected as identity matrices with appropriate dimension. The terminal constraint vector is hence given as $\boldsymbol{g}_\mathrm{N} = [1.8272,\ 0.2816,\ -0.002,\ -0.0022,\ 1.8188,\ -0.0128,\ -0.0174,\ 1.0176,\ 0.2303,1.7524]^\mathrm{T}$. The initial system states and initial admissible control gain are chosen as $\boldsymbol{x}_0 = [1,\ -1,\ 0.5]^\mathrm{T}$ and $\boldsymbol{K}_0 = [-0.3,\ 0.3,\ 1.2]$, respectively.

The designing parameter is selected as $\alpha = 0.001$. The time-dependent basis function $\boldsymbol{\varphi}(\mathrm{N}-k)$ is chosen as a function of time-to-go with saturation. Note that for finite time period, $\boldsymbol{\varphi}(\mathrm{N}-k)$ is always bounded. Saturation for $\boldsymbol{\varphi}(\mathrm{N}-k)$ is to ensure the magnitude of $\boldsymbol{\varphi}(\mathrm{N}-k)$ is within a reasonable range such that the parameter estimation is computable. The initial values for $\hat{\boldsymbol{\theta}}_k$ are set to zeros.

First, we examine the response of the system and the control input with our proposed finite-horizon optimal control design scheme. The augmented states are generated as $\boldsymbol{z}_k = [\boldsymbol{x}_k^{\mathrm{T}},\ \boldsymbol{u}_k^{\mathrm{T}}]^{\mathrm{T}} \in \Re^4$ and hence $\bar{\boldsymbol{z}}_k \in \Re^{10}$. From Figure 2, it can be seen that both the system states and the control input finally converge close to zero, which verifies the feasibility of our proposed design scheme.

Next, to verify the optimality and satisfying the terminal constraint, the error histories are plotted in Figure 3. From Figure 3, it clearly shows that the Bellman error eventually converges close to zero, which ensures the optimality of the system. It is more important to note that the history of terminal constraint error e_{N} also converges close to zero, which illustrates the fact that the terminal constraint is also satisfied with our proposed controller design.

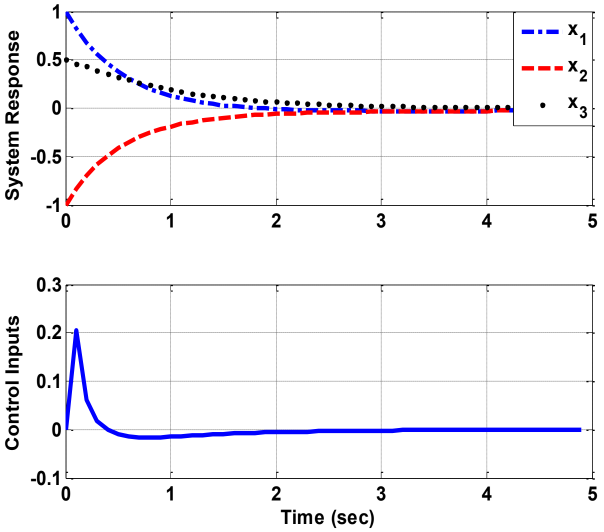


Figure 2. System response and control inputs

Finally, for comparison purpose, the error of cost function between traditional backward-in-time RE-based method and our proposed algorithm are shown in Figure 3. It can be seen clearly from the figure that the difference between two costs converges close to zero. It should note that the error between two costs converges more quickly than the system response, which illustrates that the proposed algorithm indeed yields an (near) optimal control policy.

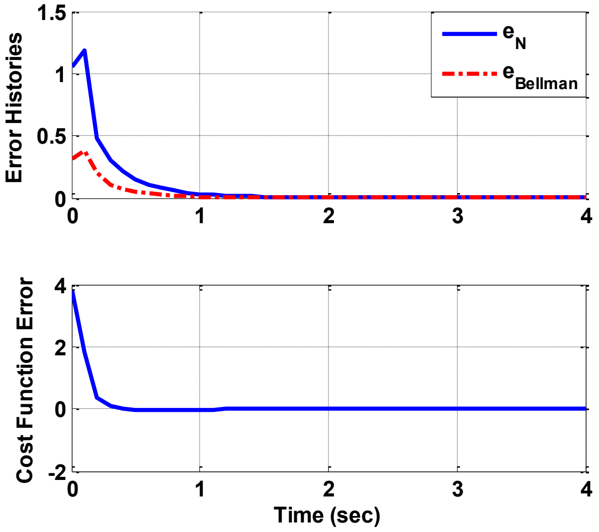


Figure 3. Convergence of error terms and cost function error between traditional and proposed method

3.2 FINITE-HORIZON OPTIMAL CONTROL WITH OUTPUT FEEDBACK

Consider the same F-16 model with output [19]:

$$\dot{\boldsymbol{x}} = \begin{bmatrix} -1.01887 & 0.90506 & -0.00215 \\ 0.82225 & -1.07741 & -0.17555 \\ 0 & 0 & -20.2 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 0 \\ 0 \\ 20.2 \end{bmatrix} \boldsymbol{u} \tag{52}$$
$$\boldsymbol{y} = [0, 57.2958, 0]\boldsymbol{x}$$

The weighting matrices $\boldsymbol{P}_k$, $\boldsymbol{R}_k$ and the terminal constraint matrix $\boldsymbol{S}_{\mathrm{N}}$ are selected to be the same as the state feedback case, and hence the terminal constraint vector $\boldsymbol{g}_{\mathrm{N}}$ is also the same as state feedback case. The initial system states and states estimate are selected to be $[-1,1,1]$ and zeros, respectively, and initial admissible control gain is chosen to be $\boldsymbol{K}_0 = [0.3,-0.3,-1.2]$. The designing parameter is selected as $\beta = 0.1$ and $\gamma = 0.001$. The time-dependent basis function $\boldsymbol{\varphi}(\mathrm{N}-k)$ is chosen similar as the state feedback case as a polynomial of time-to-go with saturation. The initial values for $\hat{\boldsymbol{\theta}}_k$ and $\hat{\vartheta}_k$ are both randomly selected between $[0,1]$. Due to space constraints, simulation results for only observer convergence and total error results are included here.

From Figure 4, it can be seen clearly that the observer error converges as time evolves, which illustrates that the estimated state becomes close to the true value in a short time indicating the feasibility of the proposed adaptive observer design scheme.

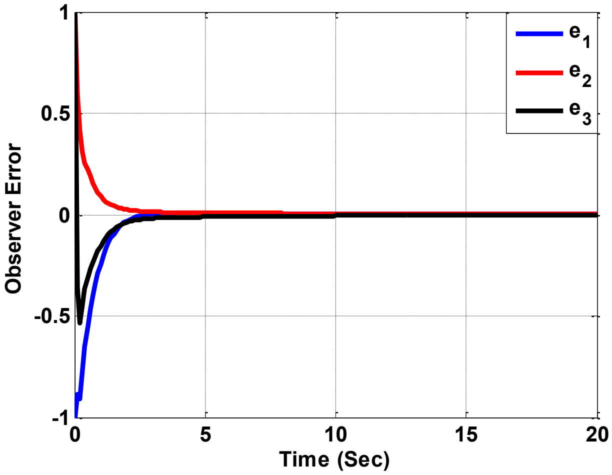


Figure 4. Observer error

It is more important to notice the evolution of the error term, which is shown in Figure 5. The convergence of the total error illustrates the fact that the optimality is guaranteed by the proposed scheme.

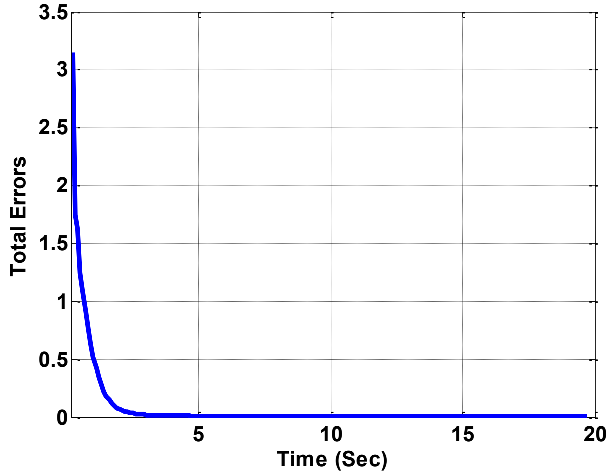


Figure 5. Convergence of the total error

4. CONCLUSIONS

In this paper, the finite-horizon optimal control of linear discrete-time systems with unknown system dynamics is addressed by using the ADP technique. The dynamics of the system are not required with an adaptive estimator generating the Q-function. An additional error is defined and incorporated in the update law so that the terminal constraint for the finite-horizon can be properly satisfied. An initial admissible control ensures the stability of the system while the adaptive estimator learns the value function and the kernel matrix $\boldsymbol{G}_k$. In addition, the proposed control design scheme is extended to output feedback case by novel adaptive observer design. All the parameters are tuned in an online and forward-in-time manner. Stability of the overall closed-loop system is demonstrated by Lyapunov analysis. Policy and value iterations are not needed. The proposed approach yields a forward-in-time and online control design scheme which offers many practical benefits.

5. REFERENCES

[1] F. L. Lewis and V. L. Syrmos, Optimal Control, second ed., Wiley, New York, 1995.

[2] V. Adetola, D. Dehaan and M. Guay, "Adaptive model predictive control for constrained nonlinear systems," System& Control Letters, vol. 58, pp. 320–326, 2008.

[3] J. Shin, H. Kim, S. Park and Y, Kim, "Model predictive flight control suing adaptive support vector regression,", Neurocomputing, vol. 73, pp. 1031–1037, 2010.

[4] Z. Chen and S. Jagannathan, "Generalized Hamilton-Jacobi-Bellman formulation based neural network control of affine nonlinear discrete-time systems", IEEE Trans. Neural Networks, 7 (2008) 90–106.

[5] X. Hao, S. Jagannathan and F. L. Lewis, "Stochastic optimal control of unknown networked control systems in the presence of random delays and packet losses," Automatica, 48 (6) (2012) 1017–1030.

[6] T. Dierks and S. Jagannathan, "Online optimal control of affine nonlinear discrete-time systems with unknown internal dynamics by using time-based policy update," IEEE Trans. Neural Networks and Learning Systems, 23 (2012) 1118–1129.

[7] R. Beard, "Improving the closed-loop performance of nonlinear systems," Ph.D. dissertation, RPI, USA, 1995.

[8] T. Cheng, F. L. Lewis, and M. Abu-Khalaf, "A neural network solution for fixed-final-time optimal control of nonlinear systems," Automatica, 43 (3) (2007) 482–490.

[9] A. Heydari and S. N. Balakrishnan, "Finite-horizon Control-Constrained Nonlinear Optimal Control Using Single Network Adaptive Critics," IEEE Trans. Neural Networks and Learning Systems, 24 (2013) 145–157.

[10] W. Feiyue, J. Ning, L. Derong and W. Qinglai, "Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with ε -error bound," IEEE Trans. Neural Networks, 22 (2011) 24–36.

[11] T. Dierks, T.B. Thumati and S. Jagannathan, "Adaptive dynamic programming-based optimal control of unknown affine nonlinear discrete-time systems," in Proc. Intern. Joint Conf. Neur. Netwo, (2009) 711–716.

[12] Sandberg and W. Erwin, "Notes on uniform approximation of time-varying systems on finite time intervals," IEEE Trans. Circuit Syst. I, Fundam. Theory Appl., vol 45, pp. 863–865, 1998.

[13] G. Cybenko, "Approximation by Superpositions of a Sigmoidal Function," Math. Control Signals Systems, vol 2, pp. 303–314, 1989.

[14] P. J. Werbos, "A menu of designs for reinforcement learing over time," J. Neural Network Contr., 3 (1983) 835–846.

[15] C. Watkins, "Learning from delayed rewards," Ph.D. dissertation, Cambridge University, England, 1989.

[16] F. L. Lewis, S. Jagannathan, and A. Yesildirek, Neural Network Control of Robot Manipulators and Nonlinear Systems, Taylor & Francis, New York, 1999.

[17] S. Jagannathan, Neural Network Control of Nonlinear Discrete-Time Systems, Boca Raton, CRC Press, FL, 2006.

[18] H.K. Khalil, Nonlinear System, third ed., Prentice-Hall, Upper Saddle River, New Jersey, 2002.

[19] B. Stevens and F. L. Lewis, Aircraft Control and Simulation, second ed., John Willey, New Jersey, 2003.

[20] M. Green and J. B. Moore, “Persistency of excitation in linear systems,” Systems & Control Letters 7 (1986) 351–360.

[21] K. S. Narendra and A. M. Annaswamy, Stable Adaptive Systems, Prentice-Hall, New Jersey. 1989.

[22] R. W. Brochett, R. S. Millman, and H. J. Sussmann, Differential geometric control theory, Birkhauser, USA, 1983.

[23] Katsuhiko Ogata, Discrete-Time Control Systems, 2nd edition, Prentice-Hall, Englewood Cliffs, New Jersey, pp. 332, 1995.

APPENDIX

Proof of Theorem 1: Consider the Lyapunov candidate function as

$$L(\tilde{\boldsymbol{\theta}}_k, k) = \mathrm{tr}\{(\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_{k-1})^{\mathrm{T}} \tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_{k-1}\} = \left\| \tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_{k-1} \right\|^2 \tag{A.1}$$

where $\mathrm{tr}\{\bullet\}$ denotes the trace operator.

Note that $L(\tilde{\boldsymbol{\theta}}_k, k)$ is a time-dependent Lyapunov candidate function due to the time-varying nature of $\Delta \overline{\boldsymbol{\Pi}}_{k-1}$. Also note that since $\Delta \overline{\boldsymbol{\Pi}}_{k-1}$ is the state-dependent function and assumed to be piecewise continuous with a finite time span of interest, then $\Delta \overline{\boldsymbol{\Pi}}_{k-1}$ is bounded by $\Delta \overline{\boldsymbol{\Pi}}_{k-1}^{\min} \leq \Delta \overline{\boldsymbol{\Pi}}_{k-1} \leq \Delta \overline{\boldsymbol{\Pi}}_{k-1}^{\max}$ for $0 < k < \mathrm{N} - 1$, where $\Delta \overline{\boldsymbol{\Pi}}_{k-1}^{\min}$ and $\Delta \overline{\boldsymbol{\Pi}}_{k-1}^{\max}$ are the constant lower and upper bound of $\Delta \overline{\boldsymbol{\Pi}}_{k-1}$ for each step k. It should be noted that finding the bounds for $\Delta \overline{\boldsymbol{\Pi}}_{k-1}$ is due to the reason that the proof is essentially under non-autonomous scheme [18]. Hence we have

$$L_1(\tilde{\boldsymbol{\theta}}_k) \leq L(\tilde{\boldsymbol{\theta}}_k, k) \leq L_2(\tilde{\boldsymbol{\theta}}_k) \tag{A.2}$$

where $L_1(\tilde{\boldsymbol{\theta}}_k) = \left\| \tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_{k-1}^{\min} \right\|^2 > 0$ and $L_2(\tilde{\boldsymbol{\theta}}_k) = \left\| \tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_{k-1}^{\max} \right\|^2 > 0$.

Therefore, $L(\tilde{\boldsymbol{\theta}}_k, k)$ is positive definite and decrescent [18].

The first difference of $L(\tilde{\boldsymbol{\theta}}_k, k)$ is given by

$$\begin{aligned} \Delta L(\tilde{\boldsymbol{\theta}}_k, k) &= L(\tilde{\boldsymbol{\theta}}_{k+1}, k+1) - L(\tilde{\boldsymbol{\theta}}_k, k) \\ &= \mathrm{tr}\{(\tilde{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_k)^{\mathrm{T}} \tilde{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_k\} - \mathrm{tr}\{(\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_{k-1})^{\mathrm{T}} \tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_{k-1}\} \\ &= \left\| \tilde{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_k \right\|^2 - \left\| \tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_{k-1} \right\|^2 \end{aligned} \tag{A.3}$$

Recall from the dynamics of the paratermer estimation error (36), we have

$$\begin{aligned}\Delta L(\tilde{\boldsymbol{\theta}}_k,k) &\le \left\|\alpha\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\Delta\overline{\boldsymbol{\Pi}}_{k-1}\right\|^2 - \left\|\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\Delta\overline{\boldsymbol{\Pi}}_{k-1}\right\|^2 \\ &\le \alpha^2\left\|\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\Delta\overline{\boldsymbol{\Pi}}_{k-1}\right\|^2 - \left\|\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\Delta\overline{\boldsymbol{\Pi}}_{k-1}\right\|^2 \\ &\le -(1-\alpha^2)\left\|\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\Delta\overline{\boldsymbol{\Pi}}_{k-1}\right\|^2 \\ &\le -(1-\alpha^2)\left\|\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\Delta\overline{\boldsymbol{\Pi}}_{k-1}^{\min}\right\|^2\end{aligned} \tag{A.4}$$

Therefore, $\Delta L(\tilde{\boldsymbol{\theta}}_k,k)$ is negative definite while $L(\tilde{\boldsymbol{\theta}}_k,k)$ is positive definite. By Lyapunov second method [23], the parameter estimation error $\tilde{\boldsymbol{\theta}}_k$ remains bounded at the terminal stage N. Furthermore, $\tilde{\boldsymbol{\theta}}_k$ will converge to zero as $k\to\infty$.

Proof of Theorem 2: Consider the Lyapunov candidate function as

$$L = L(\tilde{\boldsymbol{\theta}}_k,k) + L(\boldsymbol{x}_k) \tag{A.6}$$

where $L(\tilde{\boldsymbol{\theta}}_k,k)$ is defined in Theorem 1 and $L(\boldsymbol{x}_k)=\boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{\Lambda}\boldsymbol{x}_k$, with $\boldsymbol{\Lambda}=\dfrac{\lambda_{\min}(\boldsymbol{R})}{2B_{\mathrm{M}}^2}\boldsymbol{I}$, where B_{M} is the upper bound for the unknown matrix $\boldsymbol{B}$, $\boldsymbol{I}$ is the identity matrix with appropriate dimension and $\lambda_{\min}(\boldsymbol{R})$ is the smallest eigenvalue of weighting matrix $\boldsymbol{R}$. Next, we consider each term in (A.6) individually.

The first difference of $L(\tilde{\boldsymbol{\theta}}_k,k)$ is given by Theorem 1 as

$$\begin{aligned}\Delta L(\tilde{\boldsymbol{\theta}}_k,k) &= L(\tilde{\boldsymbol{\theta}}_{k+1},k+1) - L(\tilde{\boldsymbol{\theta}}_k,k) \\ &\le -(1-\alpha^2)\left\|\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\Delta\overline{\boldsymbol{\Pi}}_{k-1}^{\min}\right\|^2\end{aligned} \tag{A.7}$$

Next, we consider $L(\boldsymbol{x}_k)$. Define $\Lambda=\|\boldsymbol{\Lambda}\|$, by using Cauchy-Schwartz inequality, the first difference of $L(\boldsymbol{x}_k)$ is given as

$$
\begin{aligned}
\Delta L(\boldsymbol{x}_k) &= L(\boldsymbol{x}_{k+1}) - L(\boldsymbol{x}_k) = \boldsymbol{x}_{k+1}^{\mathrm{T}} \boldsymbol{\Lambda} \boldsymbol{x}_{k+1} - \boldsymbol{x}_k^{\mathrm{T}} \boldsymbol{\Lambda} \boldsymbol{x}_k \\
&= \Lambda \left\| \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k - \boldsymbol{B}\tilde{\boldsymbol{u}}_k \right\|^2 - \Lambda \left\| \boldsymbol{x}_k \right\|^2 \\
&\le 2\Lambda \left\| \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \right\|^2 + 2\Lambda \left\| \boldsymbol{B}\tilde{\boldsymbol{u}}_k \right\|^2 - \Lambda \left\| \boldsymbol{x}_k \right\|^2
\end{aligned} \tag{A.8}
$$

where $\tilde{\boldsymbol{u}}_k$ is the difference between the optimal control input and the approximated control signal. Moreover, according to the control input design, $\tilde{\boldsymbol{u}}_k = \hat{\boldsymbol{u}}_k - \boldsymbol{u}_k^* = \tilde{\boldsymbol{K}}_k \boldsymbol{x}_k$, and then we have $\left\| \tilde{\boldsymbol{K}}_k \right\| = \left\| \boldsymbol{K}_k^* - \hat{\boldsymbol{K}}_k \right\| \le \varsigma_K \left\| \tilde{\boldsymbol{\theta}}_k \right\|$, with ς_K is a positive Lipschitz constant.

Next, applying Lemma 2 yields

$$
\Delta L(\boldsymbol{x}_k) \le -(1-2\rho)\Lambda \left\| \boldsymbol{x}_k \right\|^2 + 2\Lambda \left\| \boldsymbol{B}\tilde{\boldsymbol{u}}_k \right\|^2 \tag{A.9}
$$

Combining (A.7) and (A.9), the first difference ΔL is given by

$$
\begin{aligned}
\Delta L &\le -(1-\alpha^2) \left\| \tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_{k-1}^{\min} \right\|^2 - (1-2\rho)\Lambda \left\| \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \right\|^2 + 2\Lambda \left\| \boldsymbol{B}\tilde{\boldsymbol{u}}_k \right\|^2 \\
&\le -(1-\alpha^2) \left\| \tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_{k-1}^{\min} \right\|^2 - (1-2\rho)\Lambda \left\| \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \right\|^2 + \lambda_{\min}(\boldsymbol{R}) \left\| \tilde{\boldsymbol{\theta}}_k \right\|^2 \left\| \overline{\boldsymbol{X}}_k^{'} \right\|^2 \\
&\le -\left[(1-\alpha^2) \Delta \overline{\boldsymbol{\Pi}}_{k-1}^{\min\,2} - \lambda_{\min}(\boldsymbol{R}) \left\| \overline{\boldsymbol{X}}_k^{'} \right\|^2 \right] \left\| \tilde{\boldsymbol{\theta}}_k \right\|^2 - (1-2\rho)\Lambda \left\| \boldsymbol{x}_k \right\|^2
\end{aligned} \tag{A.10}
$$

where $\overline{\boldsymbol{X}}_k^{'}$ is the gradient of $\overline{\boldsymbol{X}}_k$.

Therefore, ΔL is negative definite while ΔL is positive definite. By Lyapunov second method [17], the system states $\boldsymbol{x}_k$ and parameter estimation error $\tilde{\boldsymbol{\theta}}_k$ remain bounded at the terminal stage N.

Furthermore, assume the system is initialized within a bound $B_{\boldsymbol{x},0}$, i.e. $\left\| \boldsymbol{x}_0 \right\| \le B_{\boldsymbol{x},0}$ and initial estimated Q-function error bounded as $B_{Q,0}$. According to geometric theory [22], $\left\| \boldsymbol{x}_k \right\|^2$ and $\left\| \tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_{k-1} \right\|^2$ can be represented as

$$
\Lambda \left\| \boldsymbol{x}_k \right\|^2 + \left\| \tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta \overline{\boldsymbol{\Pi}}_{k-1} \right\|^2 \le \Lambda B_{\boldsymbol{x},0}^2 + B_{Q,0}^2 + \sum_{i=0}^{k-1} \Delta L_i \tag{A.11}
$$

Using geometric sequence property, equation (A.11) can be further derived as

$$\Lambda\|\boldsymbol{x}_k\|^2 + \left\|\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta\overline{\boldsymbol{\Pi}}_{k-1}\right\|^2 \le (2\rho)^k \Lambda B_{\boldsymbol{x},0}^2 + \left(\frac{1+\alpha^2}{2}\right)^k B_{Q,0}^2 \equiv B_k \tag{A.12}$$

Therefore, the bounds for the system states and Q-function estimation error can be written as

$$\|\boldsymbol{x}_k\| \le \sqrt{\frac{B_k}{\Lambda}} \equiv B_{\boldsymbol{x},k} \tag{A.13a}$$

Or

$$\left\|\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\right\| \le \frac{\sqrt{B_k}}{\Delta\overline{\boldsymbol{\Pi}}_{k-1}^{\min}} \equiv B_{Q,k} \tag{A.13b}$$

Note that since $0 < \rho < \frac{1}{2}$, $0 < \alpha < 1$ and $B_{\boldsymbol{x},0}$, $B_{Q,0}$ are all bounded, then $(2\rho)^k$, $\left(\frac{1+\alpha^2}{2}\right)^k$ and the bound $B_{\boldsymbol{x},k}$ and $B_{Q,k}$ will be decrease as k increases. Also note that as $\mathrm{N} \to \infty$, the system states $\boldsymbol{x}_k$ and estimated Q-function will converge to zeros as $B_{\boldsymbol{x},\infty} = 0$ and $B_{Q,\infty} = 0$.

Next, recall to (A.13) and (A.8), while time goes to fixed final time $\mathrm{NT_s}$, we have the upper bound of $\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k^*$ as

$$\begin{aligned} \left\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k^*\right\| &= \left\|\hat{\boldsymbol{K}}_k \boldsymbol{x}_k - \boldsymbol{K}_k^* \boldsymbol{x}_k\right\| = \left\|\tilde{\boldsymbol{K}}_k \boldsymbol{x}_k\right\| \\ &\le \varsigma_K \left\|\tilde{\boldsymbol{\theta}}_k\right\| \|\boldsymbol{x}_k\| \le \varsigma_K B_{Q,k} B_{\boldsymbol{x},k} \equiv \varepsilon_{\boldsymbol{us}} \end{aligned} \tag{A.14}$$

where $B_{Q,k}$ and $B_{\boldsymbol{x},k}$ are given in (A.13a) and (A.13b).

Since when time goes to infinity (i.e. $k \to \infty$), all the bounds will converge to zeros (i.e. $B_{\boldsymbol{x},\infty} = 0$ and $B_{Q,\infty} = 0$). Moreover, due to (A.14), estimated control input will tend to ideal optimal control (i.e. $\hat{\boldsymbol{u}}_k \to \boldsymbol{u}_k^*$) while time goes to infinity (i.e. $k \to \infty$).

Proof of Lemma 3: Define the Lyapunov candidate function as

$$L_{AO,k} = \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \boldsymbol{\Lambda} \tilde{\boldsymbol{x}}_k + \tilde{\vartheta}_k^{\mathrm{T}} \tilde{\vartheta}_k \tag{A.15}$$

where $\boldsymbol{\Lambda} = \beta^2 \boldsymbol{I}$, with $\boldsymbol{I}$ an identity matrix with appropriate dimension and $0 < \beta < \dfrac{\|\zeta(\boldsymbol{z}_k)\|^2}{(1+\|\zeta(\boldsymbol{z}_k)\|^2)(1+\|\psi(\boldsymbol{z}_k)\|^2)+1}$. The first difference of $L_{AO,k}$ is given as

$$\Delta L_{AO,k} = \tilde{\boldsymbol{x}}_{k+1}^{\mathrm{T}} \boldsymbol{\Lambda} \tilde{\boldsymbol{x}}_{k+1} + \tilde{\vartheta}_{k+1}^{\mathrm{T}} \tilde{\vartheta}_{k+1} - \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \boldsymbol{\Lambda} \tilde{\boldsymbol{x}}_k - \tilde{\vartheta}_k^{\mathrm{T}} \tilde{\vartheta}_k \tag{A.16}$$

Recalling the dynamics for $\tilde{\boldsymbol{x}}_k$, $\tilde{\boldsymbol{Y}}_{k+1}$ and $\tilde{\vartheta}_k$ in (42), (43) and (45), respectively, (A.16) becomes

$$\begin{aligned}
\Delta L_{AO,k} &= ((\boldsymbol{A}-\boldsymbol{LC})\tilde{\boldsymbol{x}}_k + \tilde{\vartheta}_k^{\mathrm{T}}\psi(\boldsymbol{z}_k))^{\mathrm{T}} \boldsymbol{\Lambda}((\boldsymbol{A}-\boldsymbol{LC})\tilde{\boldsymbol{x}}_k + \tilde{\vartheta}_k^{\mathrm{T}}\psi(\boldsymbol{z}_k)) - \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \boldsymbol{\Lambda} \tilde{\boldsymbol{x}}_k + \tilde{\vartheta}_{k+1}^{\mathrm{T}} \tilde{\vartheta}_{k+1} - \tilde{\vartheta}_k^{\mathrm{T}} \tilde{\vartheta}_k \\
&\le 2\tilde{\boldsymbol{x}}_k^{\mathrm{T}}(\boldsymbol{A}-\boldsymbol{LC})^{\mathrm{T}} \boldsymbol{\Lambda}(\boldsymbol{A}-\boldsymbol{LC})\tilde{\boldsymbol{x}}_k - \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \boldsymbol{\Lambda} \tilde{\boldsymbol{x}}_k + 2\psi^{\mathrm{T}}(\boldsymbol{z}_k)\tilde{\vartheta}_k \boldsymbol{\Lambda} \tilde{\vartheta}_k^{\mathrm{T}} \psi(\boldsymbol{z}_k) \\
&\quad + \left(\tilde{\vartheta}_k - \beta \frac{\boldsymbol{\zeta}(\boldsymbol{z}_k)\tilde{\boldsymbol{Y}}_{k+1}^{\mathrm{T}}}{\boldsymbol{\zeta}^{\mathrm{T}}(\boldsymbol{z}_k)\boldsymbol{\zeta}(\boldsymbol{z}_k)+1}\right)^{\mathrm{T}} \left(\tilde{\vartheta}_k - \beta \frac{\boldsymbol{\zeta}(\boldsymbol{z}_k)\tilde{\boldsymbol{Y}}_{k+1}^{\mathrm{T}}}{\boldsymbol{\zeta}^{\mathrm{T}}(\boldsymbol{z}_k)\boldsymbol{\zeta}(\boldsymbol{z}_k)+1}\right) - \tilde{\vartheta}_k^{\mathrm{T}} \tilde{\vartheta}_k \\
&\le -(1-2l_0)\Lambda\|\tilde{\boldsymbol{x}}_k\|^2 + 2\Lambda\|\psi(\boldsymbol{z}_k)\|^2 \|\tilde{\vartheta}_k\|^2 - 2\beta \frac{\tilde{\vartheta}_k^{\mathrm{T}}\boldsymbol{\zeta}(\boldsymbol{z}_k)((\boldsymbol{A}-\boldsymbol{LC})\tilde{\boldsymbol{Y}}_k + \tilde{\vartheta}_k^{\mathrm{T}}\boldsymbol{\zeta}(\boldsymbol{z}_k))^{\mathrm{T}}}{\boldsymbol{\zeta}^{\mathrm{T}}(\boldsymbol{z}_k)\boldsymbol{\zeta}(\boldsymbol{z}_k)+1} \\
&\quad + \beta^2 \frac{((\boldsymbol{A}-\boldsymbol{LC})\tilde{\boldsymbol{Y}}_k + \tilde{\vartheta}_k^{\mathrm{T}}\boldsymbol{\zeta}(\boldsymbol{z}_k))\boldsymbol{\zeta}^{\mathrm{T}}(\boldsymbol{z}_k)\boldsymbol{\zeta}(\boldsymbol{z}_k)((\boldsymbol{A}-\boldsymbol{LC})\tilde{\boldsymbol{Y}}_k + \tilde{\vartheta}_k^{\mathrm{T}}\boldsymbol{\zeta}(\boldsymbol{z}_k))^{\mathrm{T}}}{(\boldsymbol{\zeta}^{\mathrm{T}}(\boldsymbol{z}_k)\boldsymbol{\zeta}(\boldsymbol{z}_k)+1)^2} \\
&\le -(1-2l_0^2-3Cl_0^2)\Lambda\|\tilde{\boldsymbol{x}}_k\|^2 - 2\beta\left(1-\beta-\beta\|\psi(\boldsymbol{z}_k)\|^2 - \frac{1+\beta}{1+\|\zeta(\boldsymbol{z}_k)\|^2}\right)\|\tilde{\vartheta}_k\|^2
\end{aligned} \tag{A.17}$$

where $\Lambda = \|\boldsymbol{\Lambda}\|$ and $C = \|\boldsymbol{C}\|$. Therefore, based on Lyapunov stability theory, the parameter estimation error will converge to zero as $k \to \infty$. Furthermore, the design parameter l_0 satisfies $0 < l_0 < \dfrac{1}{\sqrt{2+3C}}$.

Proof of Theorem 3: Define the Lyapunov candidate function as

$$L = L_{\boldsymbol{x},k} + \Pi_1 L_{AE,k} + \Pi_2 L_{AO,k}$$

where $L_{\boldsymbol{x},k} = \boldsymbol{x}_k^{\mathrm{T}} \boldsymbol{x}_k$, $L_{AE,k} = \mathrm{tr}(\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \tilde{\boldsymbol{\theta}}_k)$, and $L_{AO,k}$ is defined in (A.15).

The first difference of L, by substituting the error dynamics, is given by

$$\begin{aligned}
\Delta L &= \Delta L_{\boldsymbol{x},k} + \Pi_1 \Delta L_{AE,k} + \Pi_2 \Delta L_{AO,k} \\
&= \boldsymbol{x}_{k+1}^{\mathrm{T}} \boldsymbol{x}_{k+1} - \boldsymbol{x}_k^{\mathrm{T}} \boldsymbol{x}_k + \Pi_1 (\mathrm{tr}(\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \tilde{\boldsymbol{\theta}}_k) - \mathrm{tr}(\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \tilde{\boldsymbol{\theta}}_k)) \\
&\quad + \Pi_2 (\tilde{\boldsymbol{x}}_{k+1}^{\mathrm{T}} \boldsymbol{\Lambda} \tilde{\boldsymbol{x}}_{k+1} + \tilde{\vartheta}_{k+1}^{\mathrm{T}} \tilde{\vartheta}_{k+1} - \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \boldsymbol{\Lambda} \tilde{\boldsymbol{x}}_k - \tilde{\vartheta}_k^{\mathrm{T}} \tilde{\vartheta}_k) \\
&= \left\| \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k^* - \boldsymbol{B}\boldsymbol{u}_k^* + \boldsymbol{B}\boldsymbol{u}_k \right\|^2 - \left\| \boldsymbol{x}_k \right\|^2 \qquad\qquad (\mathrm{A.18}) \\
&\quad + \Pi_1 \mathrm{tr}\left(\left(\tilde{\boldsymbol{\theta}}_k - \gamma \frac{\boldsymbol{\sigma}_k e_{\mathrm{total},k+1}}{\boldsymbol{\sigma}_k^{\mathrm{T}} \boldsymbol{\sigma}_k + 1} \right)^{\mathrm{T}} \left(\tilde{\boldsymbol{\theta}}_k - \gamma \frac{\boldsymbol{\sigma}_k e_{\mathrm{total},k+1}}{\boldsymbol{\sigma}_k^{\mathrm{T}} \boldsymbol{\sigma}_k + 1} \right) - \tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \tilde{\boldsymbol{\theta}}_k \right) \\
&\quad + \Pi_2 (\tilde{\boldsymbol{x}}_{k+1}^{\mathrm{T}} \boldsymbol{\Lambda} \tilde{\boldsymbol{x}}_{k+1} + \tilde{\vartheta}_{k+1}^{\mathrm{T}} \tilde{\vartheta}_{k+1} - \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \boldsymbol{\Lambda} \tilde{\boldsymbol{x}}_k - \tilde{\vartheta}_k^{\mathrm{T}} \tilde{\vartheta}_k)
\end{aligned}$$

Note that $\tilde{\boldsymbol{u}}_k = \boldsymbol{u}_k^* - \boldsymbol{u}_k = -\tilde{\boldsymbol{K}}_k \boldsymbol{x}_k + \tilde{\boldsymbol{K}}_k \tilde{\boldsymbol{x}}_k - \boldsymbol{K}_k \tilde{\boldsymbol{x}}_k$, then we have $\left\| \tilde{\boldsymbol{K}}_k \right\| = \left\| \boldsymbol{K}_k - \hat{\boldsymbol{K}}_k \right\| \le \varsigma_K \left\| \tilde{\boldsymbol{\theta}}_k \right\|$. Applying Cauchy-Schwartz inequality and recalling from Lemma 2, (A.18) becomes

$$\begin{aligned}
\Delta L &= 2\left\| \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k^* \right\|^2 + 4B_{\mathrm{M}}^2 K_{\mathrm{M}}^2 \left\| \tilde{\boldsymbol{x}}_k \right\|^2 + 4B_{\mathrm{M}}^2 L_c^2 \left\| \tilde{\boldsymbol{\theta}}_k \right\|^2 - \left\| \boldsymbol{x}_k \right\|^2 + \Pi_1 \mathrm{tr}(\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \tilde{\boldsymbol{\theta}}_k) \\
&\quad - \gamma \Pi_1 \mathrm{tr}\left(\frac{\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \boldsymbol{\sigma}_k e_{\mathrm{total},k+1}}{\boldsymbol{\sigma}_k^{\mathrm{T}} \boldsymbol{\sigma}_k + 1} \right) + \gamma^2 \Pi_1 \mathrm{tr}\left(\frac{e_{\mathrm{total},k+1} \boldsymbol{\sigma}_k^{\mathrm{T}} \boldsymbol{\sigma}_k e_{\mathrm{total},k+1}}{\boldsymbol{\sigma}_k^{\mathrm{T}} \boldsymbol{\sigma}_k + 1} \right) - \Pi_1 \mathrm{tr}(\tilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \tilde{\boldsymbol{\theta}}_k) \\
&\quad + \Pi_2 (\tilde{\boldsymbol{x}}_{k+1}^{\mathrm{T}} \boldsymbol{\Lambda} \tilde{\boldsymbol{x}}_{k+1} + \tilde{\vartheta}_{k+1}^{\mathrm{T}} \tilde{\vartheta}_{k+1} - \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \boldsymbol{\Lambda} \tilde{\boldsymbol{x}}_k - \tilde{\vartheta}_k^{\mathrm{T}} \tilde{\vartheta}_k) \\
&\le -(1 - 2\rho) \left\| \boldsymbol{x}_k \right\|^2 + 4B_{\mathrm{M}}^2 K_{\mathrm{M}}^2 \left\| \tilde{\boldsymbol{x}}_k \right\|^2 + 4B_{\mathrm{M}}^2 L_c^2 \left\| \tilde{\boldsymbol{\theta}}_k \right\|^2 - \gamma \Pi_1 \left(1 - \frac{1}{\boldsymbol{\sigma}_k^{\mathrm{T}} \boldsymbol{\sigma}_k + 1} - 5\gamma \right) \left\| \tilde{\boldsymbol{\theta}}_k \right\|^2 \\
&\quad + \frac{2\gamma^2 \Pi_1 \theta_{\mathrm{M}}^2 (L_f^2 + L_g^2)}{\boldsymbol{\sigma}_k^{\mathrm{T}} \boldsymbol{\sigma}_k + 1} \left\| \tilde{\boldsymbol{x}}_k \right\|^2 - \Pi_2 (1 - 2l_0^2 - 3Cl_0^2) \Lambda \left\| \tilde{\boldsymbol{x}}_k \right\|^2 \\
&\quad - 2\Pi_2 \beta \left(1 - \beta - \beta \left\| \psi(\boldsymbol{z}_k) \right\|^2 - \frac{1 + \beta}{1 + \left\| \boldsymbol{\zeta}(\boldsymbol{z}_k) \right\|^2} \right) \left\| \tilde{\vartheta}_k \right\|^2
\end{aligned}$$

$$\begin{aligned}&\le -(1-2\rho)\|\boldsymbol{x}_k\|^2 - 4B_{\mathrm{M}}^2 L_c^2 \|\tilde{\boldsymbol{\theta}}_k\|^2 - \left(\frac{2\gamma^2 \Pi_1 \theta_{\mathrm{M}}^2 (L_f^2 + L_g^2)}{\boldsymbol{\sigma}_k^{\mathrm{T}} \boldsymbol{\sigma}_k + 1} + 4B_{\mathrm{M}}^2 K_{\mathrm{M}}^2\right)\|\tilde{\boldsymbol{x}}_k\|^2 \\ &\quad - 2\Pi_2 \beta \left(1 - \beta - \beta\|\psi(\boldsymbol{z}_k)\|^2 - \frac{1+\beta}{1+\|\boldsymbol{\zeta}(\boldsymbol{z}_k)\|^2}\right)\|\tilde{\vartheta}_k\|^2 \\ &\le -(1-2\rho)\|\boldsymbol{x}_k\|^2 - \varpi\|\tilde{\boldsymbol{\theta}}_k\|^2 - \varpi\|\tilde{\boldsymbol{x}}_k\|^2 - \varpi\|\tilde{\vartheta}_k\|^2\end{aligned} \tag{A.19}$$

where $\Pi_1 = 8B_{\mathrm{M}}^2 \varsigma_K^2 \Big/ \gamma\left(1 - \frac{1}{\boldsymbol{\sigma}_k^{\mathrm{T}}\boldsymbol{\sigma}_k + 1} - 5\gamma\right)$, $\Pi_2 = \left(\frac{4\gamma^2 \Pi_1 \theta_{\mathrm{M}}^2 (L_f^2 + L_g^2)}{\boldsymbol{\sigma}_k^{\mathrm{T}} \boldsymbol{\sigma}_k + 1} + 8B_{\mathrm{M}}^2 K_{\mathrm{M}}^2\right) \Big/ \left((1 - 2l_0^2 - 3Cl_0^2)\Lambda\right)$

are positive constants and $0 < \varpi < 1$.

Therefore, first difference of Lyapunov function ΔL is negative definite while Lypaunov function L is positive definite. Moreover, using Lyapunov second method and geometric sequence theory, within finite horizon, the system states, parameter estimation error, state quantization error bound and control input quantization error bound will be uniformly ultimately bounded with ultimate bounds depending on initial condition $B_{\boldsymbol{x},0}, B_{\tilde{\boldsymbol{x}},0}, B_{\tilde{\theta},0}, B_{\tilde{\vartheta},0}$ with $\|\boldsymbol{x}(0)\|^2 \le B_{\boldsymbol{x},0}$, $\|\tilde{\boldsymbol{x}}(0)\|^2 \le B_{\tilde{\boldsymbol{x}},0}$, $\|\tilde{\theta}_0\|^2 \le B_{\tilde{\theta},0}$, $\|\tilde{\vartheta}_0\|^2 \le B_{\tilde{\vartheta},0}$, and terminal time $\mathrm{NT_s}$, i.e.,

$$\begin{aligned}&\|\boldsymbol{x}_k\|^2 \le \left(1 - (1-2\rho)\right)^k B_{\boldsymbol{x},0} \equiv B_{\boldsymbol{x},k}, && \forall k = 0,1,\cdots,\mathrm{N} \\ &\|\tilde{\boldsymbol{x}}_k\|^2 \le (1-\varpi)^k B_{\tilde{\boldsymbol{x}},0} \equiv B_{\tilde{\boldsymbol{x}},k}, && \forall k = 0,1,\cdots,\mathrm{N} \\ &\|\tilde{\boldsymbol{\theta}}_k\|^2 \le (1-\varpi)^k B_{\tilde{\theta},0} \equiv B_{\tilde{\boldsymbol{\theta}},k}, && \forall k = 0,1,\cdots,\mathrm{N} \\ &\|\tilde{\vartheta}_k\|^2 \le (1-\varpi)^k B_{\tilde{\vartheta},0} \equiv B_{\tilde{\vartheta},k}, && \forall k = 0,1,\cdots,\mathrm{N}\end{aligned} \tag{A.20}$$

Furthermore, since $0 < \rho < \frac{1}{2}$ and $0 < \varpi < 1$, the bounds in (A.20) are monotonically decreasing as k increases. Furthermore, when time goes infinity, i.e. $\mathrm{N} \to \infty$, all the bounds tend to zero and asymptotically stability of the closed-loop system is achieved.

Eventually, recall to (A.18) and (A.19), while time goes to fixed final time $\mathrm{NT_s}$, we have the upper bound for $\hat{\boldsymbol{u}}_k \to \boldsymbol{u}_k^*$ as

$$\begin{aligned}\left\|\hat{\boldsymbol{u}}_k - \boldsymbol{u}_k^*\right\| &= \left\|\hat{\boldsymbol{K}}_k \hat{\boldsymbol{x}}_k - \boldsymbol{K}_k^* \boldsymbol{x}_k\right\| = \left\|\tilde{\boldsymbol{K}}_k \boldsymbol{x}_k - \tilde{\boldsymbol{K}}_k \tilde{\boldsymbol{x}}_k + \boldsymbol{K}_k \tilde{\boldsymbol{x}}_k\right\| \\ &\le \varsigma_K \left\|\tilde{\boldsymbol{\theta}}_k\right\| \left\|\boldsymbol{x}_k\right\| + \varsigma_K \left\|\tilde{\boldsymbol{\theta}}_k\right\| \left\|\tilde{\boldsymbol{x}}_k\right\| + K_{\mathrm{M}} \left\|\tilde{\boldsymbol{x}}_k\right\| \\ &\le \varsigma_K B_{\tilde{\boldsymbol{\theta}},k} B_{\boldsymbol{x},k} + (\varsigma_K B_{\tilde{\boldsymbol{\theta}},k} + K_{\mathrm{M}}) B_{\tilde{\boldsymbol{x}},k} \equiv \varepsilon_{\boldsymbol{uo}}\end{aligned} \tag{A.21}$$

where $B_{\tilde{\boldsymbol{\theta}},k}$, $B_{\boldsymbol{x},k}$ and $B_{\tilde{\boldsymbol{x}},k}$ are given in (A.20). Since all the bounds will converge to zeros when $\mathrm{N} \to \infty$, the estimated control input will tend to optimal control (i.e. $\hat{\boldsymbol{u}}_k \to \boldsymbol{u}_k^*$) due to (A.21).

II. FINITE-HORIZON ADAPTIVE OPTIMAL CONTROL OF UNCERTAIN QUANTIZED LINEAR DISCRETE-TIME SYSTEM

Qiming Zhao, Hao Xu and S. Jagannathan

Abstract — In this paper, the adaptive finite-horizon optimal regulation design for unknown linear quantized discrete-time control systems is introduced. First, to mitigate the quantization error from input and state quantization, dynamic quantizer with time-varying step-size is utilized wherein it is shown that the quantization error will decrease overtime thus overcoming the drawback of the traditional uniform quantizer. Next, to relax the knowledge of system dynamics and achieve optimality, the Q-learning methodology is adopted under Bellman's principle by using quantized state and input vector. Due to the time-dependency nature of finite-horizon, an adaptive online estimator, which learns the time-varying value function, is updated at each time step so that policy and/or value iteration are not needed. Further, an additional error term corresponding to the terminal constraint is defined and minimized along the system trajectory. The proposed design scheme yields a forward-in-time online scheme, which enjoys great practical merits. Lyapunov analysis is used to show the boundedness of the closed-loop system and when the time horizon is stretched to infinity, asymptotic stability of the closed-loop system is demonstrated. Simulation results are included to verify the theoretical claim. The net result is the design of the optimal adaptive controller for uncertain quantized linear discrete-time system in a forward-in-time manner.

1. INTRODUCTION

In traditional feedback control systems, it is quite common to assume that the measured signals are transmitted to the controller and the control inputs are delivered back to the plant with arbitrarily high precision. However, in practice, the interface between the plant and the controller is often connected via analog to digital (A/D) and digital to analog (D/A) devices which quantize the signals. In addition, in the recent years, networked control system (NCS) is being considered as a next step for control where signals are quantized due to the presence of a communication network within the control loop. As a result, the quantized control system (QCS) has attracted a great deal of attention to the control researchers since quantization process always exists in the computer-based control systems.

In the past literature, the study on the effect of quantization in feedback control systems is normally categorized based on whether or not the quantizer is static or dynamic. The static quantizer, for which the quantization region does not change with time, was first analyzed for unstable linear systems in [1] by means of quantized state feedback. Later in [2], it is pointed out that logarithmic quantizers are preferred.

In the case of dynamic quantizer, for which the quantization region can be adjusted overtime based on the idea of scaling quantization levels, the authors in [3] addressed a hybrid quantized control methodology for feedback stabilization for both continuous and discrete time linear systems while demonstrating globally asymptotic stability. In [4], the author introduced a “zoom” parameter to extend the idea of changing the sensitivity of the quantizer to both linear and nonlinear systems. For these systems, however, stabilization of the closed-loop system in the presence of quantization is the major issue that was addressed when the system dynamics are known whereas the quantization effects in the presence of uncertain system dynamics and optimal control designs for such systems are not yet considered.

On the other hand, traditional optimal control theory [7] addresses both finite and infinite-horizon linear quadratic regulation (LQR) in an offline and backward-in-time manner provided the linear system dynamics are known beforehand. In the past couple of decades, significant effort has been in place to obtain optimal control in the absence of system dynamics in a forward-in-time manner by using adaptive dynamic programming (ADP) schemes [12][13][14]. Normally, to relax the system dynamics and attain optimality, the ADP schemes use policy and/or value iterations to solve either Riccati equation (RE) in the case of linear systems and Hamilton-Jacobi-Bellman (HJB) equation in the case of nonlinear systems to generate infinite-horizon based adaptive optimal control [8][11].

While iterative approach seems interesting, one has to use a significant number of iterations within a sampling interval for obtaining the solution to the RE or HJB. To overcome significant number of iterations within each sampling interval in the iterative-based schemes for convergence, in [10], a time-based ADP is introduced to generate infinite-horizon optimal control for a class of nonlinear affine discrete-time systems. Finite-horizon optimal control, in contrast, is quite difficult to solve since a terminal constraint has to be satisfied while the control is generally time-varying in contrast with the infinite-horizon scenario wherein the terminal constraint is ignored and the control input is time invariant.

For finite-horizon optimal regulation, the authors in [15][16] and [17] provided a good insight using either backward-in-time, or iterative and offline techniques. However, the adaptive optimal control over finite-horizon for uncertain linear systems in a forward-in-time manner without using iterative or offline techniques still remains unresolved. Moreover, to be best knowledge of the authors, no known technique exists for the optimal adaptive control of uncertain quantized linear discrete-time systems.

Motivated by the deficiencies aforementioned, in this paper, the ADP technique via reinforcement learning (RL) is used to solve the finite-horizon optimal regulation of uncertain linear quantized discrete-time control systems in an online and forward-in-time manner without performing value and/or policy iterations.

First, to handle the quantization effect within the control loop, a dynamic quantizer with finite number of bits is proposed. The quantization error will be addressed through the adaptive optimal controller design. Subsequently, the Bellman equation, utilized for optimal adaptive control, is investigated with approximated action-dependent value function [14] by using quantized state and input vector such that the requirement for the system dynamics is not needed. Finally, a terminal constraint error is defined and incorporated in the novel update law such that this term will be minimized at each time step in order to solve the optimal control. Lyapunov approach is utilized to show the stability of our proposed scheme. The addition of state and input quantization makes the optimal control design and its analysis more involved whereas it is addressed in the paper.

The main contributions of this paper include the novel dynamic quantizer design by using a new parameter coupled with the development of finite-horizon optimal adaptive control of uncertain quantized linear discrete-time systems in a forward-in-time manner. The new parameter ensures that the quantizer does not saturate while the quantization error will decrease overtime instead of treating these quantization errors as disturbances. The terminal state constraint is incorporated in the novel controller design scheme. Boundedness of the regulation, parameter estimation and quantization errors are demonstrated by using Lyapunov stability analysis. The proposed controller functions forward-in-time and online. If the time interval is stretched, the asymptotic stability of the closed-loop system including the convergence of the quantization errors along with the state is demonstrated.

The remainder of this paper is organized as follows. In Section 2, background is briefly introduced. In Section 3, the main algorithm developed for the finite-horizon optimal control for quantized control systems is presented. Stability analysis is provided in Section 4. In Section 5, simulation results are given to verify the feasibility of the proposed method. Conclusive remarks are provided in Section 6.

2. BACKGROUND

2.1 SYSTEM DESCRIPTION

Consider the linear system described by

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \tag{1}$$

where $\boldsymbol{x}_k \in \Omega_{\boldsymbol{x}} \subset \Re^n$ is the system state vector and assumed to be mearuable, $\boldsymbol{u}_k \in \Omega_{\boldsymbol{u}} \subset \Re^m$ is the control input received at the actuator at time step k when the quantizers are not present, while the system matrices $\boldsymbol{A} \in \Re^{n\times n}$ and $\boldsymbol{B} \in \Re^{n\times m}$ are unavailable at the controller. Before proceeding further, the following assumptions are needed.

Assumption 1 (*Controllability*): Original linear time-invariant (LTI) system (i.e. $(\boldsymbol{A}, \boldsymbol{B})$) is controllable.

Assumption 2 (*Boundedness of the input matrix*): The control input matrix $\boldsymbol{B}$ satisfies $\|\boldsymbol{B}\|_F \le B_{\mathrm{M}}$, where $\|\bullet\|_F$ denotes the Frobenius norm.

Now, in the presence of state and input quantizers, the general structure of the QCSs considered in this paper is shown in Figure 1. The state measurements are first quantized by a dynamic quantizer before being transmitted to the controller. Similarly, the control inputs are also quantized before the signals are sent to the actuator.

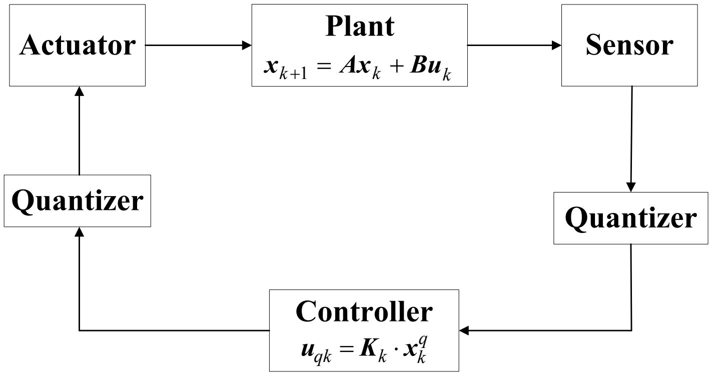

Actuator
Plant
$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k$
Sensor
Quantizer
Quantizer
Controller
$\boldsymbol{u}_{qk} = \boldsymbol{K}_k \cdot \boldsymbol{x}_k^q$

Figure 1. Block diagram of the QCSs

Next a brief background on dynamic quantizer is introduced before introducing the controller design with the quantized state and control input.

2.2 QUANTIZER REPRESENTATION

Consider the uniform quantizer with finite number of bits shown in Figure 2. Let z be the signal to be quantized and M be the quantization range for the quantizer. If z

does not belong to the quantization range, the quantizer saturates. Let e be the quantization error, then it is assumed that the following two conditions hold [4]:

$$\begin{aligned} &1.\text{if} \quad |z| \le \mathrm{M}, \quad \text{then} \;\; e = |q(z) - z| \le \Delta/2 \\ &2.\text{if} \quad |z| > \mathrm{M}, \quad \text{then} \;\; |q(z)| > \mathrm{M} - \Delta/2 \end{aligned} \tag{2}$$

where $q(z) = \Delta \cdot (\lfloor z/\Delta \rfloor + 1/2)$ is a nonlinear mapping that represents a general uniform quantizer representation with the step-size Δ defined as $\Delta = \mathrm{M}/2^{\mathrm{R}}$ with R being the number of bits of the quantizer.

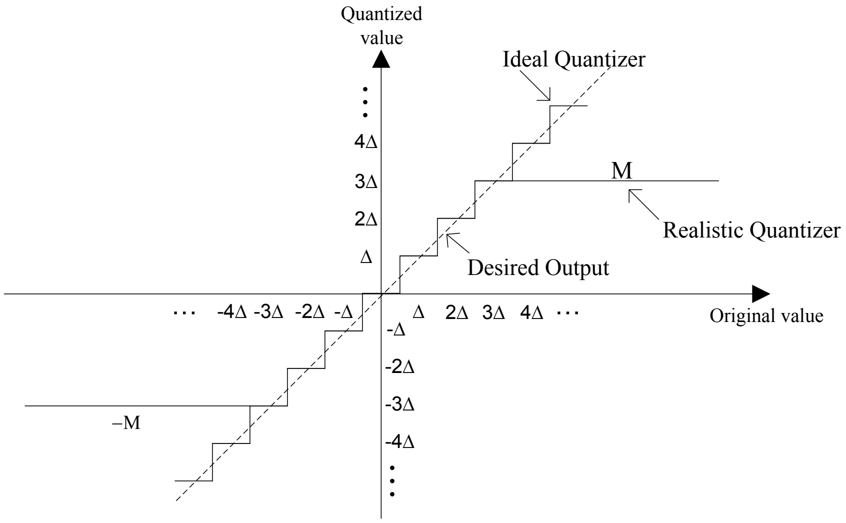


Figure 2. Ideal and realistic uniform quantizer

In addition, theoretically, when the number of bits of the quantizer approaches infinity the quantization error will reduce to zero and hence infinite precision of the quantizer can be achieved. In the realistic scenario, however, both the quantization range and the number of bits cannot be arbitrarily large. To circumvent these drawbacks, a dynamic quantizer scheme is proposed in this paper in the form similar to [4] as

$$z^q = \mu q(z/\mu) \tag{3}$$

where μ is a scaling factor.

The introduction of μ has two purposes. It will be shown in the next section that with the proposed dynamic quantizer design, not only the saturation can be avoided but also the quantization error will be eventually eliminated in contrast with the traditional uniform quantizer wherein the quantization errors never vanish.

Next the optimal control of uncertain linear discrete-time system is introduced in the presence of input and state quantization.

2.3 PROBLEM FORMULATION

Now under this closed-loop configuration, consider the time-invariant linear discrete-time system (1) in the state-space form under the influence of both state and input quantization described by

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k^a \tag{4}$$

where $\boldsymbol{x}_k \in \Omega_{\boldsymbol{x}} \subset \Re^n$ is the system state vector and $\boldsymbol{u}_k^a \in \Omega_{\boldsymbol{u}} \subset \Re^m$ is the control input vector received at the actuator at time step k. Therefore, due to the quantization, we have $\boldsymbol{u}_k^a = \boldsymbol{u}_{qk}^q$, where $\boldsymbol{u}_{qk}^q$ is the quantized control input.

Remark 1: In this paper, the superscripts represent the quantized signals, denoted as $\boldsymbol{x}_k^q$ and $\boldsymbol{u}_{qk}^q$. The subscript for the control inputs $\boldsymbol{u}_k$ represents the unquantized control inputs computed based on the quantized system states, denoted as $\boldsymbol{u}_{qk}$. It should be noted that in the QCS, only the quantized system state vector, $\boldsymbol{x}_k^q$, instead of the true state vector $\boldsymbol{x}_k$, is available to the controller. In contrast, the controller has the information of both $\boldsymbol{u}_{qk}$ and $\boldsymbol{u}_{qk}^{q}$, and hence $\boldsymbol{u}_{qk}$ will be used in the problem formulation. On the other hand, the quantized control inputs, $\boldsymbol{u}_{qk}^{q}$, will be considered in the error analysis in section 2 and the comprehensive closed-loop stability analysis in section 4.

Separating the quantization error from the actual control inputs $\boldsymbol{u}_{k}^{a}$, the system dynamics (4) can be represented as

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_{k} + \boldsymbol{B}(\boldsymbol{u}_{qk} + \boldsymbol{e}_{\boldsymbol{u},k}) = \boldsymbol{A}\boldsymbol{x}_{k} + \boldsymbol{B}\boldsymbol{u}_{qk} + \boldsymbol{B}\boldsymbol{e}_{\boldsymbol{u},k} \tag{5}$$

where $\boldsymbol{e}_{\boldsymbol{u},k}$ is the bounded quantization error for the control input as long as the control signals are within the quantization range.

Remark 2: Note that from (5), the system dynamics can be viewed as the system with only state quantization plus an additional but bounded disturbance term caused by the control input quantization provided the quantizer for the control input does not saturate. The boundedness of quantization error can be ensured by the novel dynamic quantizer design proposed in the next section so that the control input signals do not saturate.

The objective of the controller design is to determine a state feedback control policy that minimizes the following cost function

$$J_{k} = \boldsymbol{x}_{\mathrm{N}}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{N}}\boldsymbol{x}_{\mathrm{N}} + \sum_{i=k}^{\mathrm{N}-1} r(\boldsymbol{x}_{i}, \boldsymbol{u}_{qi}, i) \tag{6}$$

where $[k, \mathrm{N}]$ is time interval of interest, $r(\boldsymbol{x}_{k}, \boldsymbol{u}_{qk}, k)$ is a positive definite utility function which penalizes the system states $\boldsymbol{x}_{k}$ and the control inputs $\boldsymbol{u}_{qk}$ at each intermediate time k in $[k, \mathrm{N}]$. In this paper, the utility function is taken the form $r(\boldsymbol{x}_{k}, \boldsymbol{u}_{qk}, k) = \boldsymbol{x}_{k}^{\mathrm{T}}\boldsymbol{Q}_{k}\boldsymbol{x}_{k} + \boldsymbol{u}_{qk}^{\mathrm{T}}\boldsymbol{R}_{k}\boldsymbol{u}_{qk}$, where the weighting matrices $\boldsymbol{Q}_{k} \in \Re^{n\times n}$ is positive semi-definite, $\boldsymbol{R}_k \in \Re^{m \times m}$ is positive definite and symmetric, respectively while $\boldsymbol{S}_{\mathrm{N}} \in \Re^{n \times n}$ is a positive semi-definite symmetric penalty matrix for the terminal state $\boldsymbol{x}_{\mathrm{N}}$.

3. ADP BASED FINITE-HORIZON OPTIMAL REGULATION DESIGN

In this section, the finite-horizon optimal regulation problem for linear quantized control systems with uncertain system dynamics is addressed. Under ideal case when no saturation occurs, traditional uniform quantizer only yields a bounded response which is not preferable. The process of reducing the quantization error overtime poses a great obstacle for the optimal control design. Therefore, the dynamic quantizer design is first proposed to overcome this difficulty.

Next, to relax the requirement on system dynamics, an action-dependent value-function [13][14], which is defined and estimated adaptively by using the reinforcement learning scheme, will be in turn utilized to design the optimal adaptive controller. The Bellman equation error, which is essential to achieve optimality, is analyzed under quantization effect and parameter estimation. In addition, to satisfy the terminal constraint, an additional error term is defined and minimized as time evolves. Therefore, the objective of the controller design is to minimize both the errors so that the finite-horizon optimal regulation problem is properly investigated.

3.1 DYNAMIC QUANTIZER DESIGN

To handle the saturation caused by limited quantization range for a realistic quantizer, new parameters $\mu_{\boldsymbol{x},k}$ and $\mu_{\boldsymbol{u},k}$ are introduced. The proposed dynamic quantizers for the state and input are defined as

$$\begin{aligned} \boldsymbol{x}_k^q &= \mu_{\boldsymbol{x},k} q(\boldsymbol{x}_k / \mu_{\boldsymbol{x},k}) \\ \boldsymbol{u}_{qk}^q &= \mu_{\boldsymbol{u},k} q(\boldsymbol{u}_{qk} / \mu_{\boldsymbol{u},k}) \end{aligned} \tag{7}$$

where $\mu_{\boldsymbol{x},k}$, $\mu_{\boldsymbol{u},k}$ are the time-varying scaling parameters to be defined later for the system state and control input quantizers, respectively.

Normally, the dynamics of the quantization error cannot be established since it is mainly a round-off error. Instead, we will consider the quantization error bound as presented next, which will aid in the stability analysis. Given the dynamic quantizer in the form (7), the quantization error with respect to the system states and the control inputs are bounded, as long as no saturation occurs and the bound is given by

$$\begin{aligned} \left\|\boldsymbol{e}_{\boldsymbol{x},k}\right\| &= \left\|\boldsymbol{x}_k^q - \boldsymbol{x}_k\right\| \le \frac{1}{2}\mu_{\boldsymbol{x},k}\Delta_{\boldsymbol{x},k} = e_{\mathrm{M}\boldsymbol{x},k} \\ \left\|\boldsymbol{e}_{\boldsymbol{u},k}\right\| &= \left\|\boldsymbol{u}_{qk}^q - \boldsymbol{u}_{qk}\right\| \le \frac{1}{2}\mu_{\boldsymbol{u},k}\Delta_{\boldsymbol{u},k} = e_{\mathrm{M}\boldsymbol{u},k} \end{aligned} \tag{8}$$

where $e_{\mathrm{M}\boldsymbol{x},k}$ and $e_{\mathrm{M}\boldsymbol{u},k}$ are the upper bounds for the state and input quantization error.

Next, define the scaling parameter $\mu_{\boldsymbol{x},k}$ and $\mu_{\boldsymbol{u},k}$ as

$$\mu_{\boldsymbol{x},k} = \left\|\boldsymbol{x}_k\right\| / (\beta^k \mathrm{M}), \quad \mu_{\boldsymbol{u},k} = \left\|\boldsymbol{u}_{qk}\right\| / (\gamma^k \mathrm{M}) \tag{9}$$

where $0 < \beta < 1$ and $0 < \gamma < 1$.

Recall from representation (7) that the signals to be quantized can be “scaled” back into the quantization range with the decaying rate of β^k and γ^k , and thus eliminating the saturation effect.

The convergence of the quantization error for both system states and control inputs will be demonstrated together with the adaptive estimator design in section 3.

Remark 3: The scaling parameter $\mu_{\boldsymbol{x},k}$ and $\mu_{\boldsymbol{u},k}$ have the following properties: First, $\mu_{\boldsymbol{x},k}$ and $\mu_{\boldsymbol{u},k}$ are adjusted to eliminate saturation, which are more applicable in the realistic situations. Second, $\mu_{\boldsymbol{x},k}$ and $\mu_{\boldsymbol{u},k}$ are time-varying parameters and updated at each time interval which in turn results in a monotonic decrease in the quantization error bound. Finally, updating $\mu_{\boldsymbol{x},k}$ and $\mu_{\boldsymbol{u},k}$ only requires the signals to be quantized, which differs from [4] in which μ is a constant and can only obtained by using the system dynamics.

3.2 OPTIMAL REGULATION DESIGN

In this subsection, an action-dependent value-function is first defined and then estimated adaptively. As a result, the estimated action-dependent value-function is utilized to obtain the optimal control and relax the requirement of the system dynamics.

3.2.1 Action-Dependent Value-Function Setup. Before proceeding, it is important to note that in the case of finite-horizon, the value function becomes time-varying [7] and is a function of both system states and time-to-go, and it is denoted as $V(\boldsymbol{x}_k, \mathrm{N}-k)$. Since the value function is equal to the cost function J_k [7], the value function $V(\boldsymbol{x}_k, \mathrm{N}-k)$ for LQR can also be expressed in the quadratic form of the system states as

$$V(\boldsymbol{x}_k, \mathrm{N}-k) = \boldsymbol{x}_k^{\mathrm{T}} \boldsymbol{S}_k \boldsymbol{x}_k \tag{10}$$

with $\boldsymbol{S}_k$ being the solution sequence to the Riccati equation obtained backward-in-time from the terminal value $\boldsymbol{S}_{\mathrm{N}}$ as

$$\boldsymbol{S}_k = \boldsymbol{A}^{\mathrm{T}}[\boldsymbol{S}_{k+1} - \boldsymbol{S}_{k+1}\boldsymbol{B}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{B} + \boldsymbol{R})^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_{k+1}]\boldsymbol{A} + \boldsymbol{Q}_k \tag{11}$$

Next, define the Hamiltonian for the QCS as

$$H(\boldsymbol{x}_k, \boldsymbol{u}_{qk}, \mathrm{N}-k) = r(\boldsymbol{x}_k, \boldsymbol{u}_{qk}, k) + V(\boldsymbol{x}_{k+1}, \mathrm{N}-k-1) - V(\boldsymbol{x}_k, \mathrm{N}-k) \tag{12}$$

By using [7], the optimal control inputs are obtained via stationarity condition, i.e., $H(\boldsymbol{x}_k, \boldsymbol{u}_{qk}, \mathrm{N}-k) / \boldsymbol{u}_{qk} = 0$, which yields

$$\begin{aligned} \boldsymbol{u}_{qk}^{*} = & -(\boldsymbol{R}_k + \boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_k\boldsymbol{B})^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_k\boldsymbol{A}\cdot\boldsymbol{x}_k + (\boldsymbol{R}_k + \boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_k\boldsymbol{B})^{-1} \times \\ & (\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_k\boldsymbol{A}\boldsymbol{e}_{\boldsymbol{x},k} - \boldsymbol{R}\boldsymbol{e}_{\boldsymbol{u},k} - \boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_k\boldsymbol{B}\boldsymbol{e}_{\boldsymbol{u},k} - \boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_k\boldsymbol{e}_{\boldsymbol{x},k+1}) \end{aligned} \tag{13}$$

It can be seen clearly from (13) that the optimal control input calculated based on quantized system states enjoy the same optimal control gain, $\boldsymbol{K}_k = (\boldsymbol{R}_k + \boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_k\boldsymbol{B})^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_k\boldsymbol{A}$, as that of the case when quantization is not taken into account, plus an additional term corresponding to the quantization errors that would vanish with the proposed design as shown later. Since the only available signal to the controller is the quantized measurement $\boldsymbol{x}_k^q$, then using the "certainty equivalence" principle, the control inputs applied to the system is calculated as

$$\boldsymbol{u}_{qk} = -(\boldsymbol{R}_k + \boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_k\boldsymbol{B})^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_k\boldsymbol{A}\cdot\boldsymbol{x}_k^q \tag{14}$$

Remark 4: From (11), it is clear that the conventional approach of finding optimal solution is essentially an offline scheme given the system matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ as needed in (14). To relax the system dynamics, under infinite-horizon case, policy iterations are utilized to estimate the value function and derive the control inputs in a forward-in-time manner [8]. However, inadequate number of iterations will lead to the instability of the system [19]. In this paper, the iterative approach is not utilized and the proposed online estimator parameters are updates once a sampling interval.

Next, we will show that the system dynamics are not required by applying ADP methodology. Since the Kalman gain in (14) is the same as standard Kalman gain without quantization, assume that there is no quantization effect in the system by considering the system (1). Recalling the time-varying nature of finite-horizon control, define a time-varying optimal action dependent value function $V_{\mathrm{AD}}(\boldsymbol{x}_k,\boldsymbol{u}_k,\mathrm{N}-k)$ as

$$V_{\mathrm{AD}}(\boldsymbol{x}_k,\boldsymbol{u}_k,\mathrm{N}-k)=r(\boldsymbol{x}_k,\boldsymbol{u}_k,k)+J_{k+1}=\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix}^{\mathrm{T}}\boldsymbol{G}_k\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix} \tag{15}$$

The standard Bellman equation is given by

$$\begin{aligned}&\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix}^{\mathrm{T}}\boldsymbol{G}_k\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix}=r(\boldsymbol{x}_k,\boldsymbol{u}_k,k)+J_{k+1}\\ &=\boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{Q}_k\boldsymbol{x}_k+\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_k+\boldsymbol{x}_{k+1}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{x}_{k+1}\\ &=\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}\boldsymbol{Q}_k & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{R}_k\end{bmatrix}\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix}+(\boldsymbol{A}\boldsymbol{x}_k+\boldsymbol{B}\boldsymbol{u}_k)^{\mathrm{T}}\boldsymbol{S}_{k+1}(\boldsymbol{A}\boldsymbol{x}_k+\boldsymbol{B}\boldsymbol{u}_k)\\ &=\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}\boldsymbol{Q}_k+\boldsymbol{A}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{A} & \boldsymbol{A}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{B}\\ \boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{A} & \boldsymbol{R}_k+\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{B}\end{bmatrix}\begin{bmatrix}\boldsymbol{x}_k\\ \boldsymbol{u}_k\end{bmatrix}\end{aligned} \tag{16}$$

Therefore, define the *time-varying* matrix $\boldsymbol{G}_k$ as

$$\boldsymbol{G}_k=\begin{bmatrix}\boldsymbol{Q}_k+\boldsymbol{A}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{A} & \boldsymbol{A}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{B}\\ \boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{A} & \boldsymbol{R}_k+\boldsymbol{B}^{\mathrm{T}}\boldsymbol{S}_{k+1}\boldsymbol{B}\end{bmatrix}=\begin{bmatrix}\boldsymbol{G}_k^{\boldsymbol{xx}} & \boldsymbol{G}_k^{\boldsymbol{xu}}\\ \boldsymbol{G}_k^{\boldsymbol{ux}} & \boldsymbol{G}_k^{\boldsymbol{uu}}\end{bmatrix} \tag{17}$$

Compared to (14), the control gain can be expressed in terms of $\boldsymbol{G}_k$ as

$$\boldsymbol{K}_k=(\boldsymbol{G}_k^{\boldsymbol{uu}})^{-1}\boldsymbol{G}_k^{\boldsymbol{ux}} \tag{18}$$

From the above analysis, the time-varying action-dependent value function $V_{\mathrm{AD}}(\boldsymbol{x}_k,\boldsymbol{u}_k,\mathrm{N}-k)$ includes the information of $\boldsymbol{G}_k$ matrix which can be solved online. Therefore, the control inputs can be obtained from (17) instead of using system dynamics $\boldsymbol{A}$ and $\boldsymbol{B}$ as given in (14).

3.2.2 Model-free Online Tuning of Action-Dependent Value-Function with Quantized Signals. In this subsection, finite-horizon optimal control design is proposed without using iteration-based scheme. Recalling the definition of the action-dependent value function $V_{\mathrm{AD}}(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k)$, the following assumption and lemma are introduced before proceeding further.

Assumption 3 (*Linear-in-the-unknown-parameters*): The action-dependent value function $V_{\mathrm{AD}}(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k)$ is slowly varying and can be expressed as the linear in the unknown parameters (LIP).

By adaptive control theory [23] and the definition of the action-dependent value function, using Assumption 1, the action-dependent value function $V_{\mathrm{AD}}(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k)$ can be written in the vector form as

$$V_{\mathrm{AD}}(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k) = \boldsymbol{z}_k^{\mathrm{T}} \boldsymbol{G}_k \boldsymbol{z}_k = \boldsymbol{g}_k^{\mathrm{T}} \bar{\boldsymbol{z}}_k \tag{19}$$

where $\boldsymbol{z}_k = [\boldsymbol{x}_k^{\mathrm{T}} \quad \boldsymbol{u}_k^{\mathrm{T}}]^{\mathrm{T}} \in \Re^{n+m=l}$ is the regression function, $\bar{\boldsymbol{z}}_k = (z_{k1}^2, \cdots, z_{k1} z_{kl}, z_{k2}^2, \cdots, z_{kl-1} z_{kl}, z_{kl}^2)$ is the Kronecker product quadratic polynomial basis vector, and $\boldsymbol{g}_k = \mathrm{vec}(\boldsymbol{G}_k)$, with $\mathrm{vec}(\bullet)$ a vector function that acts on a $l \times l$ matrix and gives a $l \times (l+1)/2 = L$ column vector. The output of $\mathrm{vec}(\boldsymbol{G}_k)$ is constructed by stacking the columns of the square matrix into a one column vector with the off-diagonal elements summed as $\boldsymbol{G}_{mn}^k + \boldsymbol{G}_{nm}^k$.

Lemma 1: Let $g(k)$ be a smooth and uniformly piecewise-continuous function in a compact set $\Omega \subset \Re$. Then, for each $\varepsilon > 0$, there exist constant elements $\theta_1, ..., \theta_m \in \Re$ with $m \in N$ as well as the elements $\phi_1(k), ..., \phi_m(k) \in \Re$ of basis function, such that

$$\left| g(k) - \sum_{i=1}^{m} \theta_i \phi_i(k) \right| < \varepsilon, \qquad k \in [0, \mathrm{N}] \tag{20}$$

Proof: Omitted due to the space limitation.

Based on Assumption 3 and Lemma 1, the smooth and uniformly piecewise-continuous function, the smooth and uniformly piecewise-continuous function $\boldsymbol{g}_k$ can be represented as

$$\boldsymbol{g}_k^{\mathrm{T}} = \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{\varphi}(\mathrm{N}-k) \tag{21}$$

where $\boldsymbol{\theta} \in \Re^{L}$ is target parameter vector and $\boldsymbol{\varphi}(\mathrm{N}-k) \in \Re^{L \times L}$ is the time-varying basis function matrix, with entries as functions of time-to-go, i.e., $\boldsymbol{\varphi}(\mathrm{N}-k) = \begin{bmatrix} \phi_{11}(\mathrm{N}-k) & \phi_{12}(\mathrm{N}-k) & \cdots & \phi_{1L}(\mathrm{N}-k) \\ \phi_{21}(\mathrm{N}-k) & \phi_{22}(\mathrm{N}-k) & \cdots & \phi_{2L}(\mathrm{N}-k) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{L1}(\mathrm{N}-k) & \phi_{L2}(\mathrm{N}-k) & \cdots & \phi_{LL}(\mathrm{N}-k) \end{bmatrix}$ with $\phi_{ij}(\mathrm{N}-k) = \exp(-\tanh(\mathrm{N}-k)^{L+1-j})$, for $i, j = 1, 2, \cdots L$. This time-based function reflects the time-dependency nature of finite-horizon. Furthermore, based on universal approximation theory and given definition, $\boldsymbol{\varphi}(\mathrm{N}-k)$ is piecewise-continuous.

Therefore, the action-dependent value function can be written in terms of $\boldsymbol{\theta}$ as

$$V_{\mathrm{AD}}(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k) = \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{\varphi}(\mathrm{N}-k) \bar{\boldsymbol{z}}_k \tag{22}$$

From [7], the standard Bellman equation can be written in terms of $V_{\mathrm{AD}}(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k)$ as

$$V_{\mathrm{AD}}(\boldsymbol{x}_{k+1}, \boldsymbol{u}_{k+1}, \mathrm{N}-k-1) - V_{\mathrm{AD}}(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k) + r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) = 0 \tag{23}$$

Remark 5: In the infinite-horizon case, (21) does not have the time-varying term $\boldsymbol{\varphi}(\mathrm{N}-k)$, since the desired value of vector $\boldsymbol{g}$ is a constant, or time-invariant [9]. By contrast, in the finite-horizon case, the desired value of $\boldsymbol{g}_k$ is considered to be slowly time-varying. Hence the basis function should be a function of time and can take the form of product of the time-dependent basis function and the system states [20].

To approximate the time-varying matrix $\boldsymbol{G}_k$, or alternatively $\boldsymbol{g}_k$, define

$$\hat{\boldsymbol{g}}_k^{\mathrm{T}} = \hat{\boldsymbol{\theta}}_k^{\mathrm{T}} \boldsymbol{\varphi}(\mathrm{N}-k) \tag{24}$$

where $\hat{\boldsymbol{\theta}}_k$ is the estimation of the time-invariant part of the target parameter vector $\boldsymbol{g}_k$.

Next, when taking both the quantization effect and the estimated value of $\boldsymbol{g}_k$, the Bellman equation (23) becomes

$$e_{BQ,k} = (\boldsymbol{z}_{k+1}^q)^{\mathrm{T}} \hat{\boldsymbol{G}}_{k+1} \boldsymbol{z}_{k+1}^q - (\boldsymbol{z}_k^q)^{\mathrm{T}} \hat{\boldsymbol{G}}_k \boldsymbol{z}_k^q + (\boldsymbol{x}_k^q)^{\mathrm{T}} \boldsymbol{Q}_k \boldsymbol{x}_k^q + (\boldsymbol{u}_{qk}^q)^{\mathrm{T}} \boldsymbol{R}_k \boldsymbol{u}_{qk}^q \tag{25}$$

where $\boldsymbol{z}_k^q = [(\boldsymbol{x}_k^q)^{\mathrm{T}} \quad (\boldsymbol{u}_{qk}^q)^{\mathrm{T}}]^{\mathrm{T}} \in \Re^{n+m=l}$ is the regression function with quantized information, $e_{BQ,k}$ is the error in the Bellman equation, which can be regarded as temporal difference error (TDE).

Furthermore, $e_{BQ,k}$ can be represented as

$$\begin{aligned} e_{BQ,k} &= \psi(\boldsymbol{z}_k, \boldsymbol{z}_k^q) + [(\boldsymbol{z}_k^q)^{\mathrm{T}} \boldsymbol{G}_k \boldsymbol{z}_k^q - (\boldsymbol{z}_{k+1}^q)^{\mathrm{T}} \boldsymbol{G}_{k+1} \boldsymbol{z}_{k+1}^q - (\boldsymbol{x}_k^q)^{\mathrm{T}} \boldsymbol{Q} \boldsymbol{x}_k^q - (\boldsymbol{u}_{qk}^q)^{\mathrm{T}} \boldsymbol{R} \boldsymbol{u}_{qk}^q] \\ &\quad - [(\boldsymbol{z}_k^q)^{\mathrm{T}} \hat{\boldsymbol{G}}_k \boldsymbol{z}_k^q - (\boldsymbol{z}_{k+1}^q)^{\mathrm{T}} \hat{\boldsymbol{G}}_{k+1} \boldsymbol{z}_{k+1}^q - (\boldsymbol{x}_k^q)^{\mathrm{T}} \boldsymbol{Q} \boldsymbol{x}_k^q - (\boldsymbol{u}_{qk}^q)^{\mathrm{T}} \boldsymbol{R} \boldsymbol{u}_{qk}^q] \\ &= \psi(\boldsymbol{z}_k, \boldsymbol{z}_k^q) + (\boldsymbol{z}_k^q)^{\mathrm{T}} \widetilde{\boldsymbol{G}}_k \boldsymbol{z}_k^q - (\boldsymbol{z}_{k+1}^q)^{\mathrm{T}} \widetilde{\boldsymbol{G}}_{k+1} \boldsymbol{z}_{k+1}^q \\ &= \psi(\boldsymbol{z}_k, \boldsymbol{z}_k^q) + \widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}} [\boldsymbol{\varphi}(\mathrm{N}-k) \bar{\boldsymbol{z}}_k^q - \boldsymbol{\varphi}(\mathrm{N}-k-1) \bar{\boldsymbol{z}}_{k+1}^q] \\ &= \psi(\boldsymbol{z}_k, \boldsymbol{z}_k^q) + \widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta\xi(\boldsymbol{z}_k^q, k) \end{aligned} \tag{26}$$

where $\psi(\boldsymbol{z}_k, \boldsymbol{z}_k^q) = \boldsymbol{z}_k^{\mathrm{T}} \boldsymbol{G}_k \boldsymbol{z}_k - \boldsymbol{z}_{k+1}^{\mathrm{T}} \boldsymbol{G}_{k+1} \boldsymbol{z}_{k+1} - \boldsymbol{x}_k^{\mathrm{T}} \boldsymbol{Q}_k \boldsymbol{x}_k - \boldsymbol{u}_k^{\mathrm{T}} \boldsymbol{R}_k \boldsymbol{u}_k - [(\boldsymbol{z}_k^q)^{\mathrm{T}} \boldsymbol{G}_k \boldsymbol{z}_k^q - (\boldsymbol{z}_{k+1}^q)^{\mathrm{T}} \boldsymbol{G}_{k+1} \boldsymbol{z}_{k+1}^q - (\boldsymbol{x}_k^q)^{\mathrm{T}} \boldsymbol{Q}_k \boldsymbol{x}_k^q - (\boldsymbol{u}_{qk}^q)^{\mathrm{T}} \boldsymbol{R}_k \boldsymbol{u}_{qk}^q]$ and $\Delta\xi(\boldsymbol{z}_k^q, k) = \boldsymbol{\varphi}(\mathrm{N}-k) \bar{\boldsymbol{z}}_k^q - \boldsymbol{\varphi}(\mathrm{N}-k-1) \bar{\boldsymbol{z}}_{k+1}^q$.

Since the action-dependent value-function and the utility are in quadratic form, by Lipchitz continuity, we have $\left\|\psi(\boldsymbol{z}_k, \boldsymbol{z}_k^q)\right\| \le L_\psi \left\| \begin{bmatrix} e_{\mathrm{M}\boldsymbol{x},k} \\ e_{\mathrm{M}\boldsymbol{u},k} \end{bmatrix} \right\|^2 \le L_\psi e_{\mathrm{M}\boldsymbol{x},k}^2 + L_\psi e_{\mathrm{M}\boldsymbol{u},k}^2$, where $L_\psi > 0$ is the Lipchitz constant. Therefore, we have

$$e_{BQ,k} \le L_\psi e_{\mathrm{M}\boldsymbol{x},k}^2 + L_\psi e_{\mathrm{M}\boldsymbol{u},k}^2 + \widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta\xi(\boldsymbol{z}_k^q, k) \tag{27}$$

Recall that for the optimal control with finite-horizon, the terminal constraint of cost/value function should be taken into account properly. Therefore, define the estimated value function at the terminal stage as

$$\hat{V}_{\mathrm{AD}}(\boldsymbol{x}_{\mathrm{N}}, 0) = \hat{\boldsymbol{\theta}}_k^{\mathrm{T}} \boldsymbol{\varphi}(0) \bar{\boldsymbol{z}}_{\mathrm{N}} \tag{28}$$

In (28), note that the time-dependent basis function $\boldsymbol{\varphi}(\mathrm{N} - k)$ is taken as $\boldsymbol{\varphi}(0)$ at the terminal stage, since from definition, $\boldsymbol{\varphi}(\bullet)$ is a function of time-to-go and the time index is taken in the reverse order. Next, define the terminal constraint error vector as

$$\boldsymbol{e}_{\mathrm{N},k} = \boldsymbol{g}_{\mathrm{N}} - \hat{\boldsymbol{g}}_{\mathrm{N},k} = \boldsymbol{g}_{\mathrm{N}} - \hat{\boldsymbol{\theta}}_k^{\mathrm{T}} \boldsymbol{\varphi}(0) = \widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \boldsymbol{\varphi}(0) \tag{29}$$

where $\boldsymbol{g}_{\mathrm{N}}$ is upper bounded by $\left\|\boldsymbol{g}_{\mathrm{N}}\right\| \le g_{\mathrm{M}}$.

Remark 6: For both infinite and finite-horizon cases, the TDE $e_{BQ,k}$ is always required for tuning the parameter, see [9] and [10] for the infinite-horizon case without quantization. In finite-horizon case, the terminal error $\boldsymbol{e}_{\mathrm{N},k}$, which indicates the difference between the estimated value and true value of the terminal constraint, or "target" (in our case, $\boldsymbol{g}_{\mathrm{N}}$), is critical for the controller design. The terminal constraint is satisfied by minimizing $\boldsymbol{e}_{\mathrm{N},k}$ along the system evolution.

Remark 7: The Bellman equation with and without quantization effects are not same. The former requires $[\boldsymbol{x}_k^q, \boldsymbol{u}_{qk}^q]$ whereas the latter uses $[\boldsymbol{x}_k, \boldsymbol{u}_k]$. In order to design the optimal adaptive controller, the estimated Bellman equation with quantization effects need to eventually converge to the standard Bellman equation.

Next, define the update law for the adaptive estimator as

$$\begin{aligned}\hat{\boldsymbol{\theta}}_{k+1} = \hat{\boldsymbol{\theta}}_k &+ \alpha_\theta \frac{\Delta\xi(\boldsymbol{z}_k^q, k) e_{BQ,k}^{\mathrm{T}}}{\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q k)\Delta\xi(\boldsymbol{z}_k^q, k)+1} + \alpha_\theta \frac{\begin{bmatrix}\Delta\xi(\boldsymbol{z}_k^q, k)\left(L_\psi e_{\mathrm{M}\boldsymbol{x},k}^2 + L_\psi e_{\mathrm{M}\boldsymbol{u},k}^2\right)\times \\ \mathrm{sgn}(\Delta\xi(\boldsymbol{z}_k^q, k))\end{bmatrix}}{\Delta\xi^T(\boldsymbol{z}_k^q, k)\Delta\xi(\boldsymbol{z}_k^q, k)+1} \\ &+ \alpha_\theta \frac{\boldsymbol{\varphi}(0)\boldsymbol{e}_{\mathrm{N},k}^{\mathrm{T}}}{\|\boldsymbol{\varphi}(0)\|^2+1}\end{aligned} \tag{30}$$

Define the estimation error as $\tilde{\boldsymbol{\theta}}_k = \boldsymbol{\theta} - \hat{\boldsymbol{\theta}}_k$. Then we have

$$\begin{aligned}\tilde{\boldsymbol{\theta}}_{k+1} &= \tilde{\boldsymbol{\theta}}_k - \alpha_\theta \frac{\Delta\xi(\boldsymbol{z}_k^q, k) e_{BQ,k}^{\mathrm{T}}}{\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q k)\Delta\xi(\boldsymbol{z}_k^q, k)+1} - \alpha_\theta \frac{\begin{bmatrix}\Delta\xi(\boldsymbol{z}_k^q, k)\left(L_\psi e_{\mathrm{M}\boldsymbol{x},k}^2 + L_\psi e_{\mathrm{M}\boldsymbol{u},k}^2\right)\times \\ \mathrm{sgn}(\Delta\xi(\boldsymbol{z}_k^q, k))\end{bmatrix}}{\Delta\xi^T(\boldsymbol{z}_k^q, k)\Delta\xi(\boldsymbol{z}_k^q, k)+1} \\ &\quad - \alpha_\theta \frac{\boldsymbol{\varphi}(0)\boldsymbol{e}_{\mathrm{N},k}^{\mathrm{T}}}{\|\boldsymbol{\varphi}(0)\|^2+1} \\ &= \tilde{\boldsymbol{\theta}}_k - \alpha_\theta \frac{\Delta\xi(\boldsymbol{z}_k^q, k)\psi^{\mathrm{T}}(\boldsymbol{z}_k, \boldsymbol{z}_k^q)}{\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q k)\Delta\xi(\boldsymbol{z}_k^q, k)+1} - \alpha_\theta \frac{\Delta\xi(\boldsymbol{z}_k^q, k)\tilde{\boldsymbol{\theta}}_k \Delta^{\mathrm{T}}\xi(\boldsymbol{z}_k^q, k)}{\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q k)\Delta\xi(\boldsymbol{z}_k^q, k)+1} \\ &\quad - \alpha_\theta \frac{\Delta\xi(\boldsymbol{z}_k^q, k)\left(L_\psi e_{\mathrm{M}\boldsymbol{x},k}^2 + L_\psi e_{\mathrm{M}\boldsymbol{u},k}^2\right)\mathrm{sgn}(\Delta\xi(\boldsymbol{z}_k^q, k))}{\Delta\xi^T(\boldsymbol{z}_k^q, k)\Delta\xi(\boldsymbol{z}_k^q, k)+1} - \alpha_\theta \frac{\boldsymbol{\varphi}(0)\boldsymbol{\varphi}^{\mathrm{T}}(0)\tilde{\boldsymbol{\theta}}_k}{\|\boldsymbol{\varphi}(0)\|^2+1} \\ &= \tilde{\boldsymbol{\theta}}_k - \alpha_\theta \frac{\Delta\xi(\boldsymbol{z}_k^q, k)\Delta^{\mathrm{T}}\xi(\boldsymbol{z}_k^q, k)\tilde{\boldsymbol{\theta}}_k}{\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q k)\Delta\xi(\boldsymbol{z}_k^q, k)+1} - \alpha_\theta \frac{\boldsymbol{\varphi}(0)\boldsymbol{\varphi}^{\mathrm{T}}(0)\tilde{\boldsymbol{\theta}}_k}{\|\boldsymbol{\varphi}(0)\|^2+1} \\ &\quad - \alpha_\theta \frac{\begin{bmatrix}\Delta\xi(\boldsymbol{z}_k^q, k)\,\mathrm{sgn}(\Delta\xi(\boldsymbol{z}_k^q, k))\times \\ \left((L_\psi e_{\mathrm{M}\boldsymbol{x},k}^2 + L_\psi e_{\mathrm{M}\boldsymbol{u},k}^2) - \psi^{\mathrm{T}}(\boldsymbol{z}_k, \boldsymbol{z}_k^q)\right)\end{bmatrix}}{\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q k)\Delta\xi(\boldsymbol{z}_k^q, k)+1}\end{aligned}$$

Hence, we have

$$\widetilde{\boldsymbol{\theta}}_{k+1} \leq \widetilde{\boldsymbol{\theta}}_k - \alpha_{\boldsymbol{\theta}} \frac{\Delta\xi(\boldsymbol{z}_k^q, k)\Delta^{\mathrm{T}}\xi(\boldsymbol{z}_k^q, k)\widetilde{\boldsymbol{\theta}}_k}{\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q k)\Delta\xi(\boldsymbol{z}_k^q, k)+1} - \alpha_{\boldsymbol{\theta}} \frac{\boldsymbol{\varphi}(0)\boldsymbol{\varphi}^{\mathrm{T}}(0)\widetilde{\boldsymbol{\theta}}_k}{\|\boldsymbol{\varphi}(0)\|^2+1} \tag{31}$$

Remark 8: It is observed from the definition (19) that the value function becomes zero when $\|\boldsymbol{z}_k^q\| = 0$. Hence, when the quantized system states have converged to zero, the value function approximation is no longer updated. This can be viewed as a persistency of excitation (PE) requirement for the inputs to the value function estimator wherein the system states must be persistently exiting long enough for the estimator to learn the action-dependent value function. The PE condition can be satisfied by adding exploration noise [22] to the augmented state vector. In this paper, exploration noise is added to satisfy the PE condition while it is removed once the parameters converge.

3.3 ESTIMATION OF THE OPTIMAL FEEDBACK CONTROL

The optimal control can be obtained by minimizing the value function [7]. Recall from (18), the approximated optimal control can be obtained as

$$\boldsymbol{u}_{qk} = -\hat{\boldsymbol{K}}_k \cdot \boldsymbol{x}_k^q = -(\hat{\boldsymbol{G}}_k^{\boldsymbol{uu}})^{-1}\hat{\boldsymbol{G}}_k^{\boldsymbol{ux}} \cdot \boldsymbol{x}_k^q \tag{32}$$

From (32), the optimal control gain can be calculated based on the information of $\hat{\boldsymbol{G}}_k$ matrix, which is obtained by estimating the action-dependent value function. This relaxes the requirement of the system dynamics while the parameter estimate is updated by (30) once a sampling interval, which relaxes the value and policy iterations.

The flowchart of proposed scheme is shown in Figure 3. We start our proposed algorithm with an initial admissible control which is defined next. The system states are quantized before transmitting to the controller. After collecting both the Bellman error and terminal constraint error, the parameters for the adaptive estimator are updated once a sampling interval beginning with an initial time and until the terminal time instant in an online and forward-in-time fashion. After the update of the adaptive estimator, the control inputs are quantized by our proposed dynamic quantizer before transmitting back to the plant.

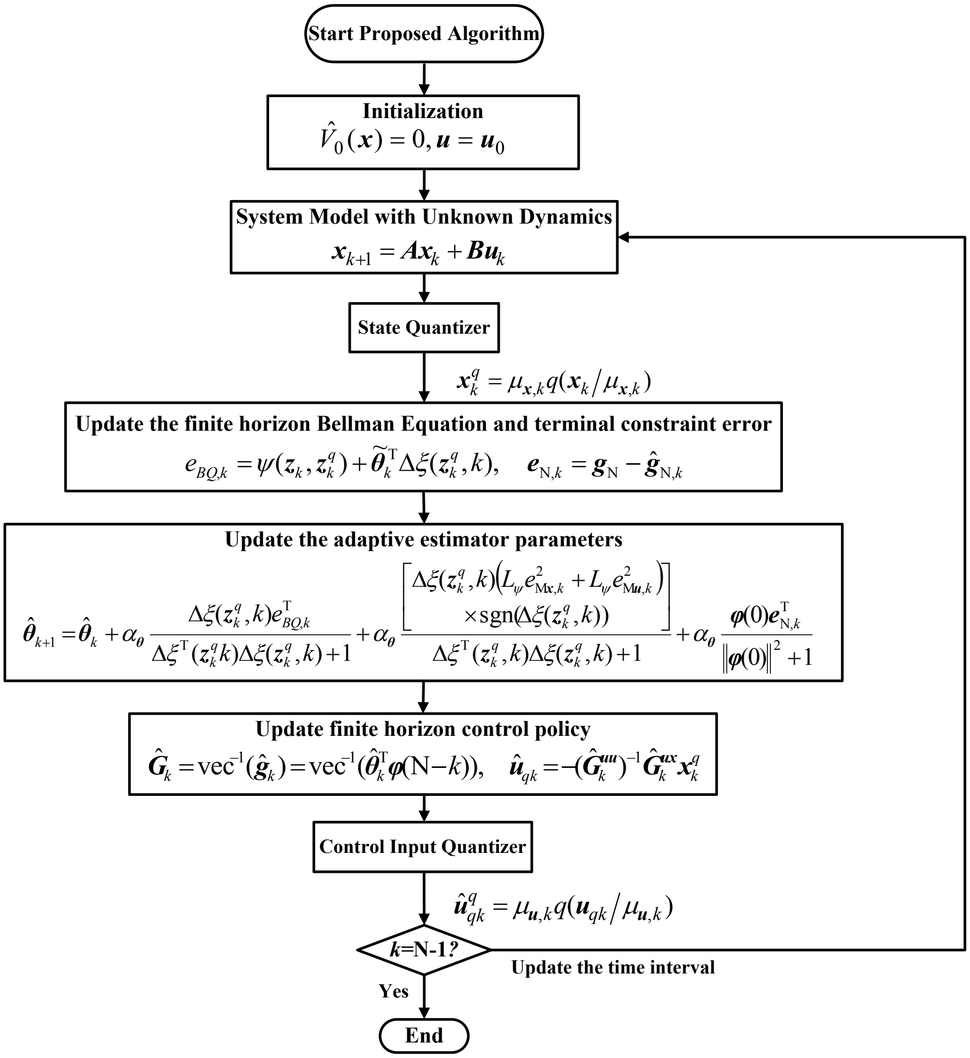

Start Proposed Algorithm
Initialization
$\hat{V}_0(\boldsymbol{x}) = 0, \boldsymbol{u} = \boldsymbol{u}_0$
System Model with Unknown Dynamics
$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k$
State Quantizer
$\boldsymbol{x}_k^q = \mu_{\boldsymbol{x},k} q(\boldsymbol{x}_k / \mu_{\boldsymbol{x},k})$
Update the finite horizon Bellman Equation and terminal constraint error
$e_{BQ,k} = \psi(\boldsymbol{z}_k, \boldsymbol{z}_k^q) + \widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \Delta\xi(\boldsymbol{z}_k^q, k), \quad \boldsymbol{e}_{\mathrm{N},k} = \boldsymbol{g}_{\mathrm{N}} - \hat{\boldsymbol{g}}_{\mathrm{N},k}$
Update the adaptive estimator parameters
$\hat{\boldsymbol{\theta}}_{k+1} = \hat{\boldsymbol{\theta}}_k + \alpha_{\boldsymbol{\theta}} \frac{\Delta\xi(\boldsymbol{z}_k^q, k) e_{BQ,k}^{\mathrm{T}}}{\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q k)\Delta\xi(\boldsymbol{z}_k^q, k) + 1} + \alpha_{\boldsymbol{\theta}} \frac{\begin{bmatrix} \Delta\xi(\boldsymbol{z}_k^q, k)\left(L_\psi e_{\mathrm{M}\boldsymbol{x},k}^2 + L_\psi e_{\mathrm{M}\boldsymbol{u},k}^2\right) \\ \times \mathrm{sgn}(\Delta\xi(\boldsymbol{z}_k^q, k)) \end{bmatrix}}{\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q, k)\Delta\xi(\boldsymbol{z}_k^q, k) + 1} + \alpha_{\boldsymbol{\theta}} \frac{\boldsymbol{\varphi}(0)\boldsymbol{e}_{\mathrm{N},k}^{\mathrm{T}}}{\|\boldsymbol{\varphi}(0)\|^2 + 1}$
Update finite horizon control policy
$\hat{\boldsymbol{G}}_k = \mathrm{vec}^{-1}(\hat{\boldsymbol{g}}_k) = \mathrm{vec}^{-1}(\hat{\boldsymbol{\theta}}_k^{\mathrm{T}} \boldsymbol{\varphi}(\mathrm{N}-k)), \quad \hat{\boldsymbol{u}}_{qk} = -(\hat{\boldsymbol{G}}_k^{\boldsymbol{uu}})^{-1} \hat{\boldsymbol{G}}_k^{\boldsymbol{ux}} \boldsymbol{x}_k^q$
Control Input Quantizer
$\hat{\boldsymbol{u}}_{qk}^q = \mu_{\boldsymbol{u},k} q(\boldsymbol{u}_{qk} / \mu_{\boldsymbol{u},k})$
k=N-1?
Update the time interval
Yes
End

Figure 3. Flowchart of the finite-horizon optimal regulation for QCS

4. STABILITY ANALYSIS

In this section, convergence of quantization error, parameter estimation error and closed-loop stability will be analyzed. It will be shown that all the errors, i.e., $\boldsymbol{e}_{\boldsymbol{x},k}$, $\boldsymbol{e}_{\boldsymbol{u},k}$, $\tilde{\boldsymbol{\theta}}_k$ will converge to zero asymptotically. Before proceeding, the following definitions are needed.

Definition 1 [21]: An equilibrium point $\boldsymbol{x}_e$ is said to be *uniformly ultimately bounded* (UUB) if there exists a compact set $\Omega_{\boldsymbol{x}} \subset \Re^n$ so that for all $\boldsymbol{x}_0 \in \Omega_{\boldsymbol{x}}$, there exists a bound B and a time $T(B, \boldsymbol{x}_0)$ such that $\|\boldsymbol{x}_k - \boldsymbol{x}_e\| \le B$ for all $k \ge k_0 + T$.

Definition 2 [11]: Let $\Omega_{\boldsymbol{u}}$ denote the set of admissible control. A control function $\boldsymbol{u}: \Re^n \to \Re^m$ is defined to be admissible if the following is true:

$\boldsymbol{u}$ is continuous on $\Omega_{\boldsymbol{u}}$;

$\boldsymbol{u}(\boldsymbol{x})\big|_{\boldsymbol{x}=0} = 0$;

$\boldsymbol{u}(\boldsymbol{x})$ stabilize the system (1) on $\Omega_{\boldsymbol{x}}$;

$J(\boldsymbol{x}(0), \boldsymbol{u}) < \infty, \forall \boldsymbol{x}(0) \in \Omega_{\boldsymbol{x}}$.

Since the design scheme is similar to policy iteration, we need to solve a fixed-point equation rather than recursive equation. The initial admissible control guarantees the solution of the fixed-potion equation exists, thus the approximation process can be effectively done by our proposed scheme.

Now, we are ready to show our main mathematical claims.

Theorem 1 (*Convergence of the adaptive estimator error*): Let the initial conditions for $\hat{\boldsymbol{g}}_0$ be bounded in a set Ω_g which contains the ideal parameter vector $\boldsymbol{g}_k$.

Let $\boldsymbol{u}_0(k) \in \Omega_{\boldsymbol{u}}$ be an initial admissible control policy for the linear system (4). Let the assumptions stated in the paper hold including the controllability of the system (1) and system state vector $\boldsymbol{x}_k \in \Omega_{\boldsymbol{x}}$ being measurable. Let the update law for tuning $\hat{\boldsymbol{\theta}}_k$ be given by (30). Then, with a positive constant α_θ satisfying $0 < \alpha_\theta < \frac{1}{4}$, there exists a $\varepsilon > 0$ depending on the initial value $B_{\tilde{\theta},0}$ and terminal stage N, such that for a fixed final time instant N, we have $\left\|\tilde{\boldsymbol{\theta}}_k\right\| \le \varepsilon(\tilde{\boldsymbol{\theta}}_k, \mathrm{N})$. Further the term $\varepsilon(\tilde{\boldsymbol{\theta}}_k, \mathrm{N})$ will converge to zero asymptotically as $\mathrm{N} \to \infty$.

Proof: See Appendix.

After establishing the convergence of the parameter estimation, we are ready to show the convergence of the quantization error for both system states and control inputs. Before proceeding, the following lemma is needed.

Lemma 2 [9]: (*Bounds on the closed-loop dynamics with optimal control*) Consider the linear discrete-time system defined in (4), then with the optimal control policy $\boldsymbol{u}_k^*$ for (4) such that the closed-loop system dynamics $\boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k^*$ can be written as

$$\left\|\boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k^*\right\|^2 \le \rho \left\|\boldsymbol{x}_k\right\|^2 \tag{33}$$

where $0 < \rho < \frac{1}{3}$ is a constant.

Proof: See [9].

Lemma 3 (*Convergence of the state quantization error*): Consider the dynamic quantizer for the system states given in (7). Let the zoom parameter for state quantizer be updated by (9). Let the adaptive estimator be updated according to (30). Then, there exists a $\varepsilon > 0$ depending on the initial value $e_{\mathrm{M}\boldsymbol{x},0}$ and the terminal stage N, such that for a fixed final time instant N, we have $\left\|\boldsymbol{e}_{\boldsymbol{x},k}\right\| \le \varepsilon(\boldsymbol{e}_{\boldsymbol{x},k}, \mathrm{N})$. Furthermore, $\varepsilon(\boldsymbol{e}_{\boldsymbol{x},k}, \mathrm{N})$ will converge to zero asymptotically as $\mathrm{N} \to \infty$.

Proof: See Appendix.

Lemma 4 (*Convergence of the input quantization error*): Consider the dynamic quantizer for the control inputs given in (7). Let the zoom parameter for the input quantizer be updated by (9). Let the adaptive estimator be updated according to (30). Then, there exists a $\varepsilon > 0$ depending on the initial value $e_{\mathrm{M}\boldsymbol{u},0}$ and the terminal stage N, such that for a fixed final time instant N, we have $\left\|\boldsymbol{e}_{\boldsymbol{u},k}\right\| \le \varepsilon(\boldsymbol{e}_{\boldsymbol{u},k}, \mathrm{N})$. Further the term $\varepsilon(\boldsymbol{e}_{\boldsymbol{u},k}, \mathrm{N})$ will converge to zero asymptotically as $\mathrm{N} \to \infty$.

Proof: See Appendix.

Theorem 2 (*Boundedness of the closed-loop system*): Let the linear discrete-time system (1) be controllable and system state be measurable. Let the initial conditions for $\hat{\boldsymbol{g}}_0$ be bounded in a set Ω_g which contains the ideal parameter vector $\boldsymbol{g}_k$. Let $\boldsymbol{u}_0(k) \in \Omega_{\boldsymbol{u}}$ be an initial admissible control policy for the system (1) such that (33) holds for some ρ. Let the scaling parameter $\mu_{\boldsymbol{x},k}$ and $\mu_{\boldsymbol{u},k}$ be updated by (9) with both input and state quantizers present. Further let the parameter vector of the action-dependent value function estimator be tuned based on (30). Then, with the positive constants $\alpha_{\boldsymbol{\theta}}$, β and γ satisfying $0 < \alpha_{\boldsymbol{\theta}} < \frac{1}{4}$, $0 < \beta < 1$ and $0 < \gamma < 1$, there exist some $\varepsilon > 0$ depending on the initial value $B_{\boldsymbol{x},0}, B_{\tilde{\boldsymbol{\theta}},0}, e_{\mathrm{M}\boldsymbol{x},0}, e_{\mathrm{M}\boldsymbol{u},0}$ and the terminal stage N, such that for a fixed final time instant N, we have $\|\boldsymbol{x}_k\| \le \varepsilon(\boldsymbol{x}_k, \mathrm{N})$, $\|\widetilde{\boldsymbol{\theta}}_k\| \le \varepsilon(\widetilde{\boldsymbol{\theta}}_k, \mathrm{N})$, $\|\boldsymbol{e}_{\boldsymbol{x},k}\| \le \varepsilon(\boldsymbol{e}_{\boldsymbol{x},k}, \mathrm{N})$ and $\|\boldsymbol{e}_{\boldsymbol{u},k}\| \le \varepsilon(\boldsymbol{e}_{\boldsymbol{u},k}, \mathrm{N})$. Furthermore, by using geometric theory, when $\mathrm{N} \to \infty$, $\varepsilon(\boldsymbol{x}_k, \mathrm{N})$, $\varepsilon(\widetilde{\boldsymbol{\theta}}_k, \mathrm{N})$, $\varepsilon(\boldsymbol{e}_{\boldsymbol{x},k}, \mathrm{N})$ and $\varepsilon(\boldsymbol{e}_{\boldsymbol{u},k}, \mathrm{N})$ will converge to zero, i.e., the system is asymptotically stable. Moreover, the estimated control input with quantization will converge to ideal optimal control (i.e. $\hat{\boldsymbol{u}}_{qk}^{q} - \boldsymbol{u}_k^*$) while time goes to infinity (i.e. $\mathrm{N} \to \infty$).

Proof: See Appendix.

Remark 9: The idea behind this paper can be extended to the NCS, since the signals need to be quantized before transmitting through the network. The network imperfections such as network-induced delays and packet dropouts can be incorporated into the system by establishing the augmented system [5] and the quantizer design in the NCS can be implemented by the same methodology introduced in this paper due to its advantages mentioned in section 3.1. In the NCS, however, due to the effect of packet dropouts, the scaling parameters $\mu_{\boldsymbol{x},k}$ and $\mu_{\boldsymbol{u},k}$ should be transmitted through a high reliable link so that the quantized signal can be accurately reconstructed on the other side of the network. This issue warrants more discussion and will be done separately.

5. SIMULATION RESULTS

In this section, an example is given to illustrate the feasibility of our proposed dynamic quantizer scheme and the finite-horizon optimal control scheme. Consider the discrete-time system given as

$$\boldsymbol{x}_{k+1} = \begin{bmatrix} 0 & -0.8 \\ 0.8 & 1.8 \end{bmatrix} \boldsymbol{x}_k + \begin{bmatrix} 0 \\ -1 \end{bmatrix} \boldsymbol{u}_k \tag{34}$$

while the performance index is given as in (6) with the weighting matrices $\boldsymbol{Q}$, $\boldsymbol{R}$ and the terminal constraint matrix $\boldsymbol{S}_{\mathrm{N}}$ are selected as the identity matrix with appropriate dimension. The terminal constraint vector is hence given as $\boldsymbol{g}_{\mathrm{N}} = [1.64,\ 2.88,\ -1.6,\ -0.0002,\ 4.88,\ -3.6,\ 2]^{\mathrm{T}}$. The initial system states and initial admissible control gain are chosen to be $\boldsymbol{x}_0 = [0.5,\ 0.5]^{\mathrm{T}}$ and $\boldsymbol{K}_0 = [-0.5,\ -1]$, respectively.

For the dynamic quantizer design, the parameters are selected as $\beta = 0.9$ and $\gamma = 0.9$. For the value function estimator, the designing parameter is chosen as $\alpha_{\boldsymbol{\theta}} = 0.001$. The time-dependent basis function $\boldsymbol{\varphi}(\mathrm{N}-k)$ is selected as a function of time-to-go with saturation. Note that for finite time period, $\boldsymbol{\varphi}(\mathrm{N}-k)$ is always bounded. Saturation for $\boldsymbol{\varphi}(\mathrm{N}-k)$ is to ensure the magnitude of $\boldsymbol{\varphi}(\mathrm{N}-k)$ is within a reasonable range such that the parameter estimation is computable. The initial values for $\hat{\boldsymbol{\theta}}_k$ are randomly selected. The simulation results are given as below.

First, the system response and control input are plotted in Figure 4 and Figure 5, respectively. It is clearly shown from the figures that both system states and control signal converges close to zero within a finite time span, which illustrates the stability of our proposed algorithm.

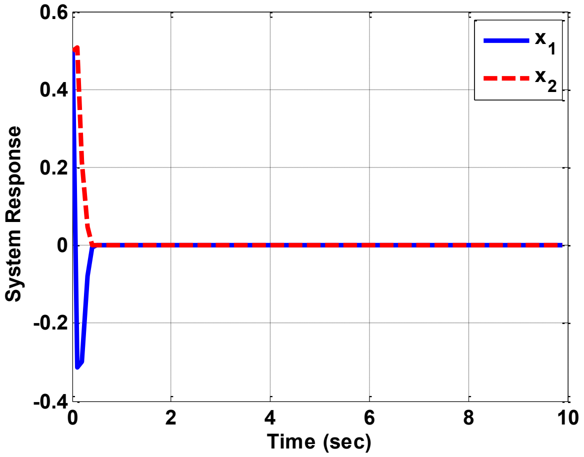


Figure 4. System response

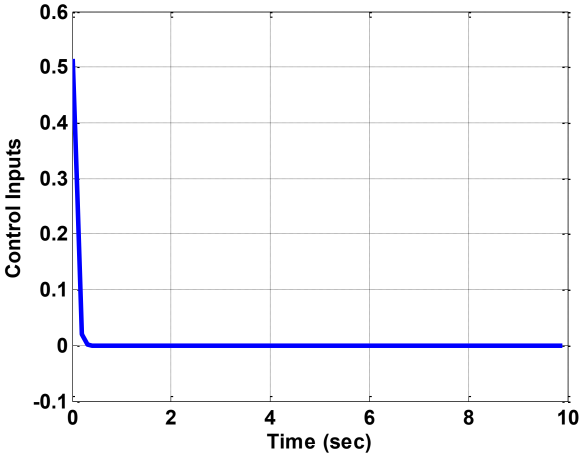


Figure 5. Control inputs

Next, to show the feasibility of the quantizer design, the quantization errors with 4-bit quantizer and 8-bit quantizer are plotted in Figure 6 and Figure 7, respectively, by using our proposed quantizer and the traditional static quantizer. From Figure 6, it can be seen that with a small number of bits, the traditional static quantizer cannot even guarantee the stability of the system due to the relatively large quantization errors, while the proposed dynamic quantizer can keep the system remain stable. This aspect will be advantageous in the NCS since a fewer number of bits for the quantizer indicates lower network traffic preventing congestion.

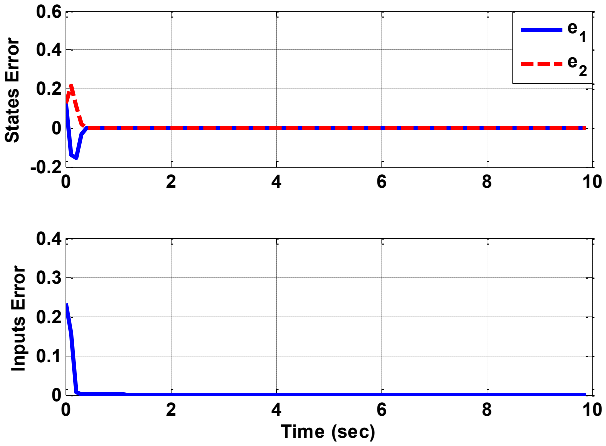


Figure 6. Quantization error with proposed dynamic quantizer with R=4

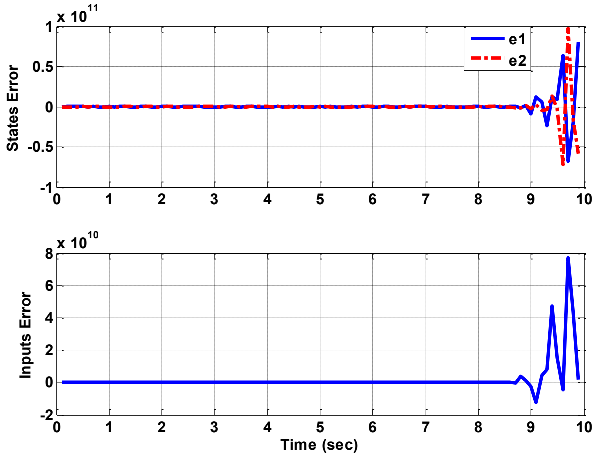


Figure 7. Quantization error with rraditional static quantizer with R=4

On the other hand, when the number of bits for the quantizer is increased to eight, it is clearly shown from Figure 8 that with the proposed dynamic quantizer, the quantization error shrinks over time, whereas in the case of traditional static quantizer as shown in Figure 9, the quantization error remains bounded as time evolves. This illustrates the fact that the effect of the quantization error can be properly handled by our proposed quantizer design.

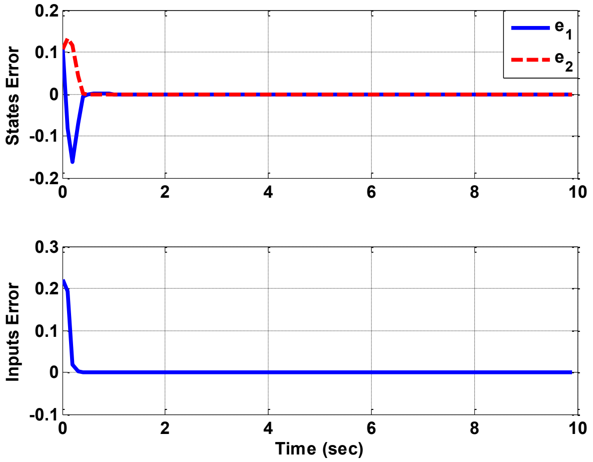


Figure 8. Quantization error with proposed dynamic quantizer with R=8

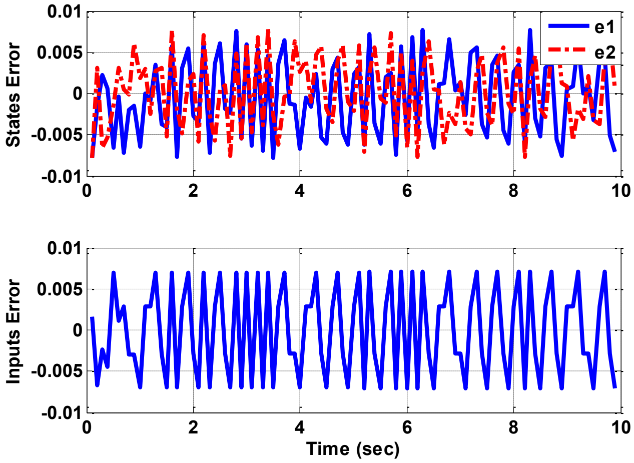


Figure 9. Quantization error with traditional static quantizer with R=8

Next, to show the optimality of our proposed scheme, the error history is given in Figure 10. It can be seen from Figure 10 that the Bellman error converges to zero, which shows that the optimality is indeed achieved. More importantly, the terminal constraint error shown in Figure 10 also converges close to zero as time evolves, which illustrates that the terminal constraint is also properly satisfied with our finite-horizon optimal control design algorithm. It should be noted that the terminal constraint error does not converge exactly to zero due to the choice of the time-dependent regression function. A more appropriate regression function would yield a better convergence of the terminal constraint error which will be considered as our future work.

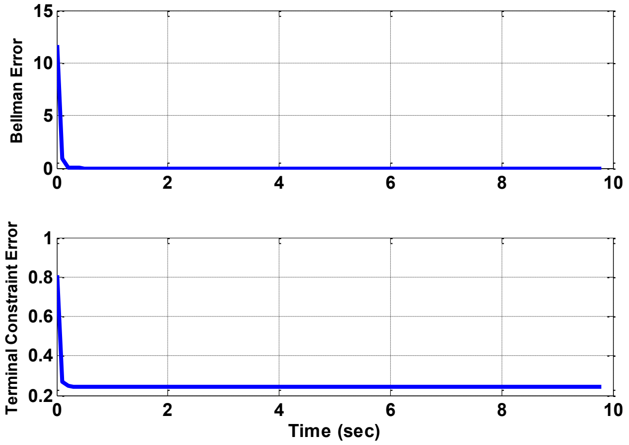


Figure 10. Error history

Finally, for comparison purpose, the difference of the cost function between the backward-in-time RE-based approach and our proposed forward-in-time scheme is shown in Figure 11. The simulation result clearly shows that the difference of the cost also converges to zero much quicker than the system response validating the proposed scheme.

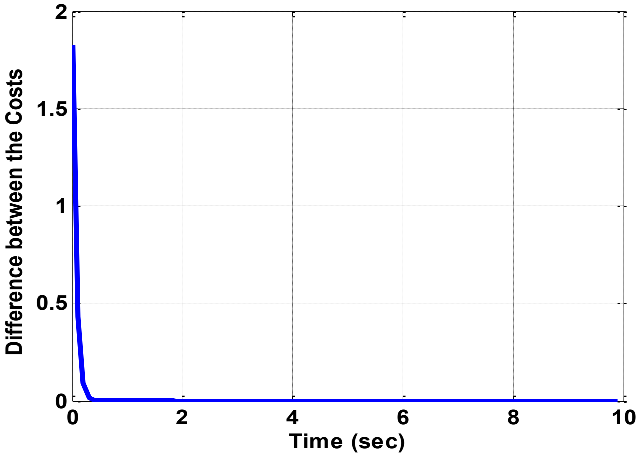


Figure 11. Difference of the cost between proposed and traditional approach

6. CONCLUSIONS

In this paper, the finite-horizon optimal control of linear discrete-time quantized control systems with unknown system dynamics is addressed. A novel dynamic quantizer is proposed to eliminate the saturation effect and quantization error. Dynamics of the system are not needed with an adaptive estimator generating the action-dependent value function $V_{\mathrm{AD}}(\boldsymbol{x}_k, \boldsymbol{u}_k, \mathrm{N}-k)$. An additional error is defined and incorporated in the update law so that the terminal constraint for the finite-horizon can be properly satisfied. An initial admissible control ensures the stability of the system while the adaptive estimator learns the value function and the kernel matrix $\boldsymbol{G}_k$. All the parameters are tuned in an online and forward-in-time manner. Policy and value iterations are not needed. Stability of the overall closed-loop system is demonstrated by Lyapunov analysis.

7. REFERENCES

[1] D. F. Delchamps, *Stabilizing a linear system with quantized state feedback*, IEEE Trans. Automat. Control, 35 (1990), pp. 916–924.

[2] N. Elia and S. K. Mitter, *Stabilization of linear systems with limited information*, IEEE Trans. Automat. Control, 46 (2001), pp. 1384–1400.

[3] R. W. Brockett and D. Liberzon, *Quantized feedback stabilization of linear systems*, IEEE Trans. Automat. Control, 45 (2000), pp. 1279–1289.

[4] D. Liberzon, *Hybrid feedback stabilization of systems with quantized signals*, Automatica J. IFAC, 39 (2003), pp. 1543–1554.

[5] Q. Zhao, H. Xu and S. Jagannathan, *Optimal adaptive controller scheme for uncertain quantized linear discrete-time system*, in Proc. 51th IEEE Conf. Dec. Contr., Hawaii, 2012, pp. 6132–6137.

[6] Q. Zhao, H. Xu and S. Jagannathan, *Adaptive dynamic programming-based-state quantized networked control system without value and/or policy iterations*, in Proc. Int. Joint Conf. Neural Nets, Brisbane, 2012, pp. 1–7.

[7] F. L. Lewis and V. L. Syrmos, *Optimal Control*, 2nd edition. New York: Wiley, 1995.

[8] S. J. BRADTKE AND B. E. YDSTIE, *Adaptive linear quadratic control using policy iteration*, in Proc. Am Contr. Conf., Baltimore, 1994, pp. 3475–3479.

[9] H. XU, S. JAGANNATHAN AND F. L. LEWIS, *Stochastic optimal control of unknown networked control systems in the presence of random delays and packet losses*, Automatica J. IFAC, 48 (2012), pp. 1017–1030.

[10] T. DIERKS AND S. JAGANNATHAN, *Online optimal control of affine nonlinear discrete-time systems with unknown internal dynamics by using time-based policy update*, IEEE Trans. Neural Networks and Learning Systems, 23 (2012), pp. 1118–1129.

[11] Z. CHEN AND S. JAGANNATHAN, *Generalized Hamilton-Jacobi-Bellman formulation based neural network control of affine nonlinear discrete-time systems*, IEEE Trans. Neural Netw, 7 (2008), pp. 90–106.

[12] J. SI, A. G. BARTO, W. B. POWELL AND D. WUNSCH, *Handbook of Learning and Approximate Dynamic Programming*. New York: Wiley, 2004.

[13] P. J. WERBOS, *A menu of designs for reinforcement learing over time*, J. Neural Network Contr., 3 (1983), pp. 835–846.

[14] C. WATKINS, *Learning from delayed rewards*, Ph.D. dissertation, Cambridge University, England, 1989.

[15] R. BEARD, *Improving the closed-loop performance of nonlinear systems*, Ph.D. dissertation, Electr. Eng. Dept., Rensselaer Polytechnic Institute, USA, 1995.

[16] A. HEYDARI AND S. N. BALAKRISHNAN, *Finite-horizon control-constrained nonlinear optimal control using single network adaptive critics*, IEEE Trans. Neural Networks and Learning Systems, 24 (2013), pp. 145–157.

[17] W. FEIYUE, J. NING, L. DERONG AND W. QINGLAI, *Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with ε-error bound,* IEEE Trans. Neural Netw, 22 (2011), pp. 24–36.

[18] Q. ZHAO, H. XU AND S. JAGANNATHAN, *Finite-horizon optimal control design for uncertain linear discrete-time systems*, in Proc. IEEE Symp. Approx. Dyn. Programm. Reinforcement Learn., Singapore, 2013.

[19] H. XU AND S. JAGANNATHAN, *Stochastic optimal controller design for uncertain nonlinear networked control system via neuro dynamic programming*, IEEE Trans. Neural Netw. And Learning Syst, 24 (2013), pp. 471–484.

[20] F. L. LEWIS, S. JAGANNATHAN, AND A. YESILDIREK, *Neural Network Control of Robot Manipulators and Nonlinear Systems*, New York: Taylor & Francis, 1999.

[21] S. JAGANNATHAN, *Neural Network Control of Nonlinear Discrete-Time Systems*, Boca Raton, FL: CRC Press, 2006.

[22] M. GREEN AND J. B. MOORE, *Persistency of excitation in linear systems*, Systems Control Lett., 7 (1986), pp. 351–360.

[23] K. S. NARENDRA AND A. M. ANNASWAMY, *Stable Adaptive Systems*, New Jersey: Prentice-Hall, 1989.

APPENDIX

Proof of Theorem 1:

Consider the Lyapunov candidate function as

$$L_{\boldsymbol{\theta}} = \widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \widetilde{\boldsymbol{\theta}}_k \tag{A.1}$$

The first difference of $L_{\boldsymbol{\theta}}$ is given, according to (31), by

$$\Delta L_{\boldsymbol{\theta}} = \widetilde{\boldsymbol{\theta}}_{k+1}^{\mathrm{T}} \widetilde{\boldsymbol{\theta}}_{k+1} - \widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \widetilde{\boldsymbol{\theta}}_k \le \left(\widetilde{\boldsymbol{\theta}}_k - \alpha_{\boldsymbol{\theta}} \frac{\Delta\xi(\boldsymbol{z}_k^q,k)\Delta^{\mathrm{T}}\xi(\boldsymbol{z}_k^q,k)\widetilde{\boldsymbol{\theta}}_k}{\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q k)\Delta\xi(\boldsymbol{z}_k^q,k)+1} - \alpha_{\boldsymbol{\theta}} \frac{\boldsymbol{\varphi}(0)\boldsymbol{\varphi}^{\mathrm{T}}(0)\widetilde{\boldsymbol{\theta}}_k}{\|\boldsymbol{\varphi}(0)\|^2+1} \right)^T \times$$

$$\left(\widetilde{\boldsymbol{\theta}}_k - \alpha_{\boldsymbol{\theta}} \frac{\Delta\xi(\boldsymbol{z}_k^q,k)\Delta^{\mathrm{T}}\xi(\boldsymbol{z}_k^q,k)\widetilde{\boldsymbol{\theta}}_k}{\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q k)\Delta\xi(\boldsymbol{z}_k^q,k)+1} - \alpha_{\boldsymbol{\theta}} \frac{\boldsymbol{\varphi}(0)\boldsymbol{\varphi}^{\mathrm{T}}(0)\widetilde{\boldsymbol{\theta}}_k}{\|\boldsymbol{\varphi}(0)\|^2+1} \right) - \widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \widetilde{\boldsymbol{\theta}}_k$$

$$\le \widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \widetilde{\boldsymbol{\theta}}_k - 2\alpha_{\boldsymbol{\theta}} \frac{\widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\Delta\xi(\boldsymbol{z}_k^q,k)\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q,k)\widetilde{\boldsymbol{\theta}}_k}{\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q,k)\Delta\xi(\boldsymbol{z}_k^q,k)+1} - 2\alpha_{\boldsymbol{\theta}} \frac{\widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(0)\boldsymbol{\varphi}^{\mathrm{T}}(0)\widetilde{\boldsymbol{\theta}}_k}{\|\sigma_N\|^2+1}$$

$$+ \left(\alpha_{\boldsymbol{\theta}} \frac{\Delta\xi(z_k^q,k)\Delta\xi^{\mathrm{T}}(z_k^q,k)\widetilde{\boldsymbol{\theta}}_k}{\Delta\xi^{T}(\boldsymbol{z}_k^q,k)\Delta\xi(\boldsymbol{z}_k^q,k)+1} + \alpha_{\boldsymbol{\theta}} \frac{\boldsymbol{\varphi}(0)\boldsymbol{\varphi}^{\mathrm{T}}(0)\widetilde{\boldsymbol{\theta}}_k}{\|\sigma_N\|^2+1} \right)^2 - \widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}} \widetilde{\boldsymbol{\theta}}_k$$

$$\le -2\alpha_{\boldsymbol{\theta}} \frac{\widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\Delta\xi(\boldsymbol{z}_k^q,k)\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q,k)\widetilde{\boldsymbol{\theta}}_k}{\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q,k)\Delta\xi(\boldsymbol{z}_k^q,k)+1} - 2\alpha_{\boldsymbol{\theta}} \frac{\widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(0)\boldsymbol{\varphi}^{\mathrm{T}}(0)\widetilde{\boldsymbol{\theta}}_k}{\|\boldsymbol{\varphi}(0)\|^2+1}$$

$$+ 2\alpha_{\boldsymbol{\theta}}^2 \frac{\widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\Delta\xi(\boldsymbol{z}_k^q,k)\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q,k)\widetilde{\boldsymbol{\theta}}_k}{\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q,k)\Delta\xi(\boldsymbol{z}_k^q,k)+1} + 2\alpha_{\boldsymbol{\theta}}^2 \frac{\widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(0)\boldsymbol{\varphi}^{\mathrm{T}}(0)\widetilde{\boldsymbol{\theta}}_k}{\|\boldsymbol{\varphi}(0)\|^2+1}$$

$$\le -2\alpha_{\boldsymbol{\theta}}(1-\alpha_{\boldsymbol{\theta}}) \frac{\widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\Delta\xi(\boldsymbol{z}_k^q,k)\Delta\xi^{\mathrm{T}}(\boldsymbol{z}_k^q,k)\widetilde{\boldsymbol{\theta}}_k}{\Delta\xi^{T}(\boldsymbol{z}_k^q,k)\Delta\xi(\boldsymbol{z}_k^q,k)+1} - 2\alpha_{\boldsymbol{\theta}}(1-\alpha_{\boldsymbol{\theta}}) \frac{\widetilde{\boldsymbol{\theta}}_k^{\mathrm{T}}\boldsymbol{\varphi}(0)\boldsymbol{\varphi}^{\mathrm{T}}(0)\widetilde{\boldsymbol{\theta}}_k}{\|\boldsymbol{\varphi}(0)\|^2+1}$$

$$\le -2\alpha_{\boldsymbol{\theta}}(1-\alpha_{\boldsymbol{\theta}}) \left(\frac{\Delta\xi_{\min}^2}{1+\Delta\xi_{\min}^2} + \frac{\|\boldsymbol{\varphi}(0)\|^2}{\|\boldsymbol{\varphi}(0)\|^2+1} \right) \left\|\widetilde{\boldsymbol{\theta}}_k\right\|^2$$

Define $\zeta = 2\alpha_{\boldsymbol{\theta}}(1-\alpha_{\boldsymbol{\theta}}) \left(\frac{\Delta\xi_{\min}^2}{1+\Delta\xi_{\min}^2} + \frac{\|\boldsymbol{\varphi}(0)\|^2}{\|\boldsymbol{\varphi}(0)\|^2+1} \right)$. We have $0 < \Delta\xi_{\min}^2 \le \left\|\Delta\xi(\boldsymbol{z}_k^q,k)\right\|^2$ due to the PE condition, then,

$$\Delta L_{\boldsymbol{\theta}} \le -2\alpha_{\boldsymbol{\theta}}(1-\alpha_{\boldsymbol{\theta}})\left(\frac{\Delta\xi_{\min}^2}{1+\Delta\xi_{\min}^2}+\frac{\|\boldsymbol{\varphi}(0)\|^2}{\|\boldsymbol{\varphi}(0)\|^2+1}\right)\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 \equiv -\zeta\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 \tag{A.2}$$

Therefore, by Lyapunov stability theory, the estimation error $\tilde{\boldsymbol{\theta}}_k$ will converge to zero as $k \to \infty$.

Proof of Lemma 3: Recall from the quantizer design, for the state quantization, the quantization error is always bounded by $e_{\mathrm{M}\boldsymbol{x},k}$, as shown in (8). Therefore, instead of dealing with the quantization error directly, we focus on the analysis of quantization error bound. Recalling from (8), we have

$$\frac{e_{\mathrm{M}\boldsymbol{x},k+1}^2}{e_{\mathrm{M}\boldsymbol{x},k}^2}=\frac{\mu_{\boldsymbol{x},k+1}^2\Delta_{\boldsymbol{x},k+1}^2}{\mu_{\boldsymbol{x},k}^2\Delta_{\boldsymbol{x},k}^2}=\frac{\|\boldsymbol{x}_{k+1}\|^2}{\|\boldsymbol{x}_k\|^2} \tag{A.3}$$

Substituting the system dynamics (5) into (A.3) yields

$$\begin{aligned}\frac{e_{\mathrm{M}\boldsymbol{x},k+1}^2}{e_{\mathrm{M}\boldsymbol{x},k}^2}&=\frac{\left\|\boldsymbol{Ax}_k+\boldsymbol{Bu}_{qk}+\boldsymbol{Be}_{\boldsymbol{u}k}\right\|^2}{\|\boldsymbol{x}_k\|^2}=\frac{\left\|\boldsymbol{Ax}_k+\boldsymbol{B}\hat{\boldsymbol{K}}_k\boldsymbol{x}_k^q+\boldsymbol{Be}_{\boldsymbol{u},k}\right\|^2}{\|\boldsymbol{x}_k\|^2}\\&=\frac{\left\|\boldsymbol{Ax}_k+\boldsymbol{BK}_k^*\boldsymbol{x}_k+\boldsymbol{B}\hat{\boldsymbol{K}}_k\boldsymbol{x}_k^q-\boldsymbol{BK}_k^*\boldsymbol{x}_k+\boldsymbol{Be}_{\boldsymbol{u},k}\right\|^2}{\|\boldsymbol{x}_k\|^2}\\&=\frac{\left\|\boldsymbol{Ax}_k+\boldsymbol{BK}_k^*\boldsymbol{x}_k+\boldsymbol{B}\tilde{\boldsymbol{K}}_k\boldsymbol{x}_k-\boldsymbol{B}\hat{\boldsymbol{K}}_k\boldsymbol{e}_{\boldsymbol{x},k}+\boldsymbol{Be}_{\boldsymbol{u},k}\right\|^2}{\|\boldsymbol{x}_k\|^2}\end{aligned} \tag{A.4}$$

with $\boldsymbol{K}_k^*$ the true Kalman gain satisfying $\|\boldsymbol{K}_k^*\| \le K_{\mathrm{M}}$, and $\tilde{\boldsymbol{K}}_k=\boldsymbol{K}_k^*-\hat{\boldsymbol{K}}_k$ is the Kalman gain error.

Applying Cauchy-Schwartz inequality and using Lemma 1, (A.4) can be further written as

$$\begin{aligned}\frac{e^2_{\mathrm{M}\boldsymbol{x},k+1}}{e^2_{\mathrm{M}\boldsymbol{x},k}} &\le \frac{3\left\|\boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{K}^*_k\boldsymbol{x}_k\right\|^2}{\left\|\boldsymbol{x}_k\right\|^2} + \frac{3\left\|\boldsymbol{B}\widetilde{\boldsymbol{K}}_k\boldsymbol{x}_k - \boldsymbol{B}\hat{\boldsymbol{K}}_k\boldsymbol{e}_{\boldsymbol{x},k}\right\|^2}{\left\|\boldsymbol{x}_k\right\|^2} + \frac{3\left\|\boldsymbol{B}\boldsymbol{e}_{\boldsymbol{u},k}\right\|^2}{\left\|\boldsymbol{x}_k\right\|^2} \\ &\le 3\rho + \frac{6\left\|\boldsymbol{B}\widetilde{\boldsymbol{K}}_k\boldsymbol{x}_k\right\|^2}{\left\|\boldsymbol{x}_k\right\|^2} + \frac{6\left\|\boldsymbol{B}\hat{\boldsymbol{K}}_k\boldsymbol{e}_{\boldsymbol{x},k}\right\|^2}{\left\|\boldsymbol{x}_k\right\|^2} + \frac{3\left\|\boldsymbol{B}\boldsymbol{e}_{\boldsymbol{u},k}\right\|^2}{\left\|\boldsymbol{x}_k\right\|^2}\end{aligned} \tag{A.5}$$

Recall from (18) and the definition of the adaptive estimator, we have $\boldsymbol{K}^*_k = (\boldsymbol{G}^{\boldsymbol{uu}}_k)^{-1}\boldsymbol{G}^{\boldsymbol{ux}}_k = f(\boldsymbol{g}_k)$ and similarly $\hat{\boldsymbol{K}}_k = (\hat{\boldsymbol{G}}^{\boldsymbol{uu}}_k)^{-1}\hat{\boldsymbol{G}}^{\boldsymbol{ux}}_k = f(\hat{\boldsymbol{g}}_k)$, then the Kalman gain error can be represented as

$$\left\|\widetilde{\boldsymbol{K}}_k\right\| = \left\|f(\boldsymbol{g}_k) - f(\hat{\boldsymbol{g}}_k)\right\| \le L_f\left\|\widetilde{\boldsymbol{g}}_k\right\| = L_f\left\|\widetilde{\boldsymbol{\theta}}_k\phi(\mathrm{N}-k)\right\| \le L_f\phi_{\max}\left\|\widetilde{\boldsymbol{\theta}}_k\right\| \tag{A.6}$$

where L_f is a positive Lipchitz constant, $\phi_{\max}$ always exists since the time of interest is finite and hence $\boldsymbol{\varphi}(\mathrm{N}-k)$ is always bounded. Hence, (A.5) becomes

$$\begin{aligned}\frac{e^2_{\mathrm{M}\boldsymbol{x},k+1}}{e^2_{\mathrm{M}\boldsymbol{x},k}} &\le 3\rho + 6B^2_{\mathrm{M}}\left\|\widetilde{\boldsymbol{K}}_k\right\|^2 + \frac{6B^2_{\mathrm{M}}\left\|\hat{\boldsymbol{K}}_k\right\|^2}{2^{2\mathrm{R}}} + \frac{3B^2_{\mathrm{M}}\left\|\boldsymbol{e}_{\boldsymbol{u},k}\right\|^2}{\left\|\boldsymbol{x}_k\right\|^2} \\ &\le 3\rho + 6B^2_{\mathrm{M}}L^2_f\phi^2_{\max}\left\|\widetilde{\boldsymbol{\theta}}_k\right\|^2 + \frac{6B^2_{\mathrm{M}}\left\|\hat{\boldsymbol{K}}_k\right\|^2}{2^{2\mathrm{R}}} + \frac{3B^2_{\mathrm{M}}\left\|\boldsymbol{e}_{\boldsymbol{u},k}\right\|^2}{\left\|\boldsymbol{x}_k\right\|^2}\end{aligned} \tag{A.7}$$

Furthermore, since $\left\|\boldsymbol{K}^*_k\right\| \le K_{\mathrm{M}}$, we have

$$\begin{aligned}\frac{\left\|\boldsymbol{e}_{\boldsymbol{u}k}\right\|^2}{\left\|\boldsymbol{x}_k\right\|^2} &\le \frac{e^2_{\mathrm{M}\boldsymbol{u},k}}{\left\|\boldsymbol{x}_k\right\|^2} \le \frac{1}{2^{2\mathrm{R}}}\frac{\left\|\boldsymbol{u}_{qk}\right\|^2}{\left\|\boldsymbol{x}_k\right\|^2} \le \frac{1}{2^{2\mathrm{R}}}\frac{\left\|\hat{\boldsymbol{K}}_k\boldsymbol{x}^q_k\right\|^2}{\left\|\boldsymbol{x}_k\right\|^2} \le \frac{1}{2^{2\mathrm{R}}}\frac{\left\|\hat{\boldsymbol{K}}_k\boldsymbol{x}_k + \hat{\boldsymbol{K}}_k\boldsymbol{e}_{\boldsymbol{x},k}\right\|^2}{\left\|\boldsymbol{x}_k\right\|^2} \\ &\le \frac{1}{2^{2\mathrm{R}}}\frac{\left\|\widetilde{\boldsymbol{K}}_k\boldsymbol{x}_k + \hat{\boldsymbol{K}}_k\boldsymbol{e}_{\boldsymbol{x},k} + \boldsymbol{K}^*_k\boldsymbol{x}_k\right\|^2}{\left\|\boldsymbol{x}_k\right\|^2} \le \frac{1}{2^{2\mathrm{R}}}\left(3B^2_{\mathrm{M}}L^2_f\phi^2_{\max}\left\|\widetilde{\boldsymbol{\theta}}_k\right\|^2 + \frac{3\left\|\hat{\boldsymbol{K}}_k\right\|^2}{2^{2\mathrm{R}}} + 3\boldsymbol{K}^2_{\mathrm{M}}\right)\end{aligned}$$

Therefore, (A.7) becomes

$$\begin{aligned}\frac{e^2_{\mathrm{M}\boldsymbol{x},k+1}}{e^2_{\mathrm{M}\boldsymbol{x},k}} &\le 3\rho + 6B_{\mathrm{M}}^2 L_f^2 \phi_{\max}^2 \left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 + \frac{6B_{\mathrm{M}}^2\left\|\hat{\boldsymbol{K}}_k\right\|^2}{2^{2\mathrm{R}}} + \frac{3B_{\mathrm{M}}^2}{2^{2\mathrm{R}}}\left(3B_{\mathrm{M}}^2 L_f^2 \phi_{\max}^2 \left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 + \frac{3\left\|\hat{\boldsymbol{K}}_k\right\|^2}{2^{2\mathrm{R}}} + 3\boldsymbol{K}_{\mathrm{M}}^2\right)\\ &\le 3\rho + B_{\mathrm{M}}^2 L_f^2 \phi_{\max}^2 \left(6 + 9/2^{2\mathrm{R}}\right)\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 + \frac{B_{\mathrm{M}}^2}{2^{2\mathrm{R}}}\left(6 + 9/2^{2\mathrm{R}}\right)\left\|\hat{\boldsymbol{K}}_k\right\|^2 + \frac{9B_{\mathrm{M}}^2 K_{\mathrm{M}}^2}{2^{2\mathrm{R}}}\end{aligned}$$

Hence, for the quantizer, there exists a finite number of bits R_f such that for all $\mathrm{R} \ge \mathrm{R}_f$, we have

$$\frac{B_{\mathrm{M}}^2}{2^{2\mathrm{R}_f}}\left(6 + 9/2^{2\mathrm{R}_f}\right)\left\|\hat{\boldsymbol{K}}_k\right\|^2 + \frac{9B_{\mathrm{M}}^2 K_{\mathrm{M}}^2}{2^{2\mathrm{R}_f}} \le \frac{3 - 3\rho}{2} \tag{A.8}$$

Therefore, (A.7) can be written as

$$\begin{aligned}\frac{e^2_{\mathrm{M}\boldsymbol{x},k+1}}{e^2_{\mathrm{M}\boldsymbol{x},k}} &\le 3\rho + B_{\mathrm{M}}^2 L_f^2 \phi_{\max}^2 \left(6 + 9/2^{2\mathrm{R}_f}\right)\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 + (1 - 3\rho)/2\\ &\le (1 + 3\rho)/2 + B_{\mathrm{M}}^2 L_f^2 \phi_{\max}^2 \left(6 + 9/2^{2\mathrm{R}_f}\right)\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2\end{aligned} \tag{A.9}$$

Recall from Theorem 1, since $0 < \alpha_{\boldsymbol{\theta}} < \dfrac{1}{4}$, thus $0 < \zeta < 1$, which further implies $\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 \le (1 - \zeta)^k \left\|\tilde{\boldsymbol{\theta}}_0\right\|^2$. Hence, (A.9) becomes

$$\frac{e^2_{\mathrm{M}\boldsymbol{x},k+1}}{e^2_{\mathrm{M}\boldsymbol{x},k}} \le (1 + 3\rho)/2 + B_{\mathrm{M}}^2 L_f^2 \phi_{\max}^2 \left(6 + 9/2^{2\mathrm{R}_f}\right)(1 - \zeta)^k \left\|\tilde{\boldsymbol{\theta}}_0\right\|^2 \tag{A.10}$$

Therefore, there exists a finite number k_f such that for all $k > k_f$, $B_{\mathrm{M}}^2 L_f^2 \phi_{\max}^2 \left(6 + 9/2^{2\mathrm{R}_f}\right)(1 - \zeta)^{k_f} \left\|\tilde{\boldsymbol{\theta}}_0\right\|^2 < (1 - 3\rho)/2$. Hence,

$$\frac{e^2_{\mathrm{M}\boldsymbol{x},k+1}}{e^2_{\mathrm{M}\boldsymbol{x},k}} \equiv \eta \le (1 + 3\rho)/2 + (1 - 3\rho)/2 < 1 \tag{A.11}$$

According to (A.11), within finite-horizon, the quantization error bound for system state is UUB with ultimate bound depending on initial quantization error bound $e^2_{\mathrm{M}\boldsymbol{x},0}$ and terminal time, $\mathrm{NT_s}$ i.e.,

$$e_{\mathrm{M}\boldsymbol{x},k}^2 \le \eta^k e_{\mathrm{M}\boldsymbol{x},0}^2, \qquad \forall k = 0,1,\cdots,\mathrm{N} \tag{A.12}$$

Further, the quantization error bound for the system states $e_{\mathrm{M}\boldsymbol{x},k}$ converges to zero asymptotically as $k \to \infty$. Since quantization error never exceeds the bound, then the state quantization error also converge to zero as $k \to \infty$.

Proof of Lemma 4: Recall form (32), the control inputs is given as

$$\boldsymbol{u}_{qk} = \hat{\boldsymbol{K}}_k \boldsymbol{x}_k^q = \boldsymbol{K}_k^* \boldsymbol{x}_k^q - \tilde{\boldsymbol{K}}_k \boldsymbol{x}_k^q \tag{A.13}$$

where $\tilde{\boldsymbol{K}}_k = \boldsymbol{K}_k^* - \hat{\boldsymbol{K}}_k$ is the Kalman gain error.

Similar to the state quantization, we have the quantization error bound for the control input as

$$\begin{aligned} e_{\mathrm{M}\boldsymbol{u},k} &= \frac{\left\|\boldsymbol{u}_{qk}\right\|}{2^{\mathrm{R}}} \le \frac{K_{\mathrm{M}}}{2^{\mathrm{R}}}\left\|\boldsymbol{x}_k\right\| + \frac{K_{\mathrm{M}}}{2^{\mathrm{R}}}\left\|\boldsymbol{e}_{\boldsymbol{x},k}\right\| + \frac{L_f \phi_{\max}}{2^{\mathrm{R}}}\left\|\tilde{\boldsymbol{\theta}}_k\right\|\left\|\boldsymbol{x}_k\right\| + \frac{L_f \phi_{\max}}{2^{\mathrm{R}}}\left\|\tilde{\boldsymbol{\theta}}_k\right\|\left\|\boldsymbol{e}_{\boldsymbol{x},k}\right\| \\ &\le \left(1 + 1/2^{\mathrm{R}}\right) K_{\mathrm{M}} e_{\mathrm{M}\boldsymbol{x},k} + \left(1 + 1/2^{\mathrm{R}}\right) L_f \phi_{\max}\left\|\tilde{\boldsymbol{\theta}}_k\right\| e_{\mathrm{M}\boldsymbol{x},k} \\ &\equiv e_{\mathrm{MM}\boldsymbol{u},k} \end{aligned} \tag{A.14}$$

Define the Lyapunov candidate function as $L(e_{\mathrm{MM}\boldsymbol{u},k}) = e_{\mathrm{MM}\boldsymbol{u},k}^2$.

The first difference of $L(e_{\mathrm{MM}\boldsymbol{u},k})$ is given by

$$\begin{aligned} \Delta L(e_{\mathrm{MM}\boldsymbol{u},k}) &= e_{\mathrm{MM}\boldsymbol{u},k+1}^2 - e_{\mathrm{MM}\boldsymbol{u},k}^2 \\ &= \left(1 + 1/2^{\mathrm{R}}\right)^2 K_{\mathrm{M}}^2 e_{\mathrm{M}\boldsymbol{x},k+1}^2 + \left(1 + 1/2^{\mathrm{R}}\right) L_f^2 \phi_{\max}^2 \left\|\tilde{\boldsymbol{\theta}}_{k+1}\right\|^2 e_{\mathrm{M}\boldsymbol{x},k+1}^2 \\ &\quad - \left(1 + 1/2^{\mathrm{R}}\right)^2 K_{\mathrm{M}}^2 e_{\mathrm{M}\boldsymbol{x},k}^2 - \left(1 + 1/2^{\mathrm{R}}\right) L_f^2 \phi_{\max}^2 \left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 e_{\mathrm{M}\boldsymbol{x},k}^2 \\ &\le -(1-\eta^2)\left(1 + 1/2^{\mathrm{R}}\right)^2 K_{\mathrm{M}}^2 e_{\mathrm{M}\boldsymbol{x},k}^2 - (1-\zeta^2\eta^2)\left(1 + 1/2^{\mathrm{R}}\right)^2 L_f^2 \phi_{\max}^2 \left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 e_{\mathrm{M}\boldsymbol{x},k}^2 \\ &\le \delta e_{\mathrm{MM}\boldsymbol{u},k}^2 \end{aligned} \tag{A.15}$$

where $0 < \delta < \frac{1}{2}\min\left\{1-\eta^2, 1-\zeta^2\eta^2\right\} < 1$.

According to (A.15), within finite horizon, the quantization error bound for control input is UUB with ultimate bound depending on initial quantization error bound $e^2_{\mathrm{MM}\boldsymbol{u},0}$ and terminal time $\mathrm{NT_s}$, i.e.,

$$e^2_{\mathrm{MM}\boldsymbol{u},k} \le (1-\delta)^k e^2_{\mathrm{MM}\boldsymbol{u},0}, \qquad \forall k = 0,1,\cdots,\mathrm{N} \tag{A.16}$$

Moreover, since first difference of Lyapunov function $\Delta L(e_{\mathrm{MM}\boldsymbol{u},k})$ is negative definite while Lypaunov function $L(e_{\mathrm{MM}\boldsymbol{u},k})$ is positive definite, we have $e_{\mathrm{MM}\boldsymbol{u},k} \to 0$ as $k \to \infty$. Since $\left\|e_{\boldsymbol{u},k}\right\| \le e_{\mathrm{M}\boldsymbol{u},k} \le e_{\mathrm{MM}\boldsymbol{u},k}$, $\left\|e_{\boldsymbol{u},k}\right\| \to 0$ as $k \to \infty$.

Proof of Theorem 2: Consider the Lyapunov candidate function as

$$\begin{aligned} L = {} & \Lambda_5 L(\boldsymbol{x}_k) + \Lambda_1\Lambda_5 L(\tilde{\boldsymbol{\theta}}_k) + \Lambda_2\Lambda_5(1-\zeta^2\eta^2)L(\tilde{\boldsymbol{\theta}}_k, e_{\mathrm{M}\boldsymbol{x},k}) \\ & + \Lambda_3\Lambda_5 L(e_{\mathrm{M}\boldsymbol{x},k}) + \Lambda_4\Lambda_5 L(e_{\mathrm{M}\boldsymbol{u},k}) \end{aligned} \tag{A.17}$$

where $L(\boldsymbol{x}_k) = \boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{x}_k$, $L(\tilde{\boldsymbol{\theta}}_k, e_{\mathrm{M}\boldsymbol{x},k}) = \left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 e^2_{\mathrm{M}\boldsymbol{x},k}$, $L(e_{\mathrm{M}\boldsymbol{x},k}) = e^2_{\mathrm{M}\boldsymbol{x},k}$, $L(e_{\mathrm{M}\boldsymbol{u},k}) = e^2_{\mathrm{M}\boldsymbol{u},k}$, $\Lambda_1 = 24B^2_{\mathrm{M}}L^2_f\phi^2_{\max}/(1-\zeta)$, $\Lambda_2 = 24B^2_{\mathrm{M}}L^2_f\phi^2_{\max}/(1-\zeta^2\eta^2)$, $\Lambda_3 = 12B^2_{\mathrm{M}}K^2_{\mathrm{M}}/(1-\eta^2)$ and $\Lambda_4 = 12B^2_{\mathrm{M}}/\left((1-\eta^2)(1+1/2^{\mathrm{R}})^2K^2_{\mathrm{M}}\right)$, $\Lambda_5 = \varpi/\max\{12B^2_{\mathrm{M}}L^2_f\phi^2_{\max}\Theta^2_{\mathrm{M}}\zeta^2_{\mathrm{M}},\ 6B^2_{\mathrm{M}}K^2_{\mathrm{M}},\ 3B^2_{\mathrm{M}}\}$ with $0 < \varpi < \min\left\{1,\ \max\{12B^2_{\mathrm{M}}L^2_f\phi^2_{\max}\Theta^2_{\mathrm{M}}\zeta^2_{\mathrm{M}},\ 6B^2_{\mathrm{M}}K^2_{\mathrm{M}},\ 3B^2_{\mathrm{M}}\}\right\} < 1$.

Next, consider each term in (A.17) individually. Applying Cauchy-Schwartz inequality and recalling Lemma 1, we have

$$\begin{aligned}
\Delta L(\boldsymbol{x}_k) &= \boldsymbol{x}_{k+1}^{\mathrm{T}}\boldsymbol{x}_{k+1} - \boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{x}_k = \left\|\boldsymbol{x}_{k+1}\right\|^2 - \left\|\boldsymbol{x}_k\right\|^2 = \left\|\boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_{qk} + \boldsymbol{B}\boldsymbol{e}_{\boldsymbol{u},k}\right\|^2 - \left\|\boldsymbol{x}_k\right\|^2 \\
&= \left\|\boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{K}_k^*\boldsymbol{x}_k + \boldsymbol{B}\hat{\boldsymbol{K}}_k\boldsymbol{x}_k^q - \boldsymbol{B}\boldsymbol{K}_k^*\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{e}_{\boldsymbol{u},k}\right\|^2 - \left\|\boldsymbol{x}_k\right\|^2 \\
&\le 3\left\|\boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{K}_k^*\boldsymbol{x}_k\right\|^2 + 6B_{\mathrm{M}}^2\left\|\tilde{\boldsymbol{K}}_k\right\|^2\left\|\boldsymbol{x}_k^q\right\|^2 + 6B_{\mathrm{M}}^2K_{\mathrm{M}}^2\left\|\boldsymbol{e}_{\boldsymbol{x},k}\right\|^2 + 3B_{\mathrm{M}}^2\left\|\boldsymbol{e}_{\boldsymbol{u}k}\right\|^2 - \left\|\boldsymbol{x}_k\right\|^2 \\
&\le -(1-3\rho)\left\|\boldsymbol{x}_k\right\|^2 + 12B_{\mathrm{M}}^2L_f^2\phi_{\max}^2\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2\Theta_{\mathrm{M}}^2\left\|\xi_{k-1}\right\|^2 + 12B_{\mathrm{M}}^2L_f^2\phi_{\max}^2\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 e_{\mathrm{M}\boldsymbol{x},k}^2 \\
&\quad + 6B_{\mathrm{M}}^2K_{\mathrm{M}}^2e_{\mathrm{M}\boldsymbol{x},k}^2 + 3B_{\mathrm{M}}^2e_{\mathrm{M}\boldsymbol{u},k}^2 \\
&\le -(1-3\rho)\left\|\boldsymbol{x}_k\right\|^2 + 12B_{\mathrm{M}}^2L_f^2\phi_{\max}^2\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2\Theta_{\mathrm{M}}^2\xi_{\mathrm{M}}^2 + 12B_{\mathrm{M}}^2L_f^2\phi_{\max}^2\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 e_{\mathrm{M}\boldsymbol{x},k}^2 \\
&\quad + 6B_{\mathrm{M}}^2K_{\mathrm{M}}^2e_{\mathrm{M}\boldsymbol{x},k}^2 + 3B_{\mathrm{M}}^2e_{\mathrm{M}\boldsymbol{u},k}^2
\end{aligned}$$

where $\left\|\Theta\right\| = \left\|[\boldsymbol{A}\ \ \boldsymbol{B}]\right\| \le \Theta_{\mathrm{M}}$ and the history information satisfying $\left\|\xi_{k-1}\right\| = \left\|[\boldsymbol{x}_{k-1}^{\mathrm{T}}\ \ \boldsymbol{u}_{qk-1}^{\mathrm{T}}]^{\mathrm{T}}\right\| \le \xi_{\mathrm{M}}$.

Next, recalling from Theorem 1, Lemma 2 and Lemma 3, the total difference of the Lyapunov candidate function is given by

$$\begin{aligned}
\Delta L &= \Lambda_5\Delta L(\boldsymbol{x}_k) + \Lambda_1\Lambda_5\Delta L(\tilde{\boldsymbol{\theta}}_k) + \Lambda_2\Lambda_5(1-\zeta^2\eta^2)\Delta L(\tilde{\boldsymbol{\theta}}_k, e_{\mathrm{M}\boldsymbol{x},k}) \\
&\quad + \Lambda_3\Lambda_5\Delta L(e_{\mathrm{M}\boldsymbol{x},k}) + \Lambda_4\Lambda_5\Delta L(e_{\mathrm{M}\boldsymbol{u},k}) \\
&\le -\Lambda_5(1-3\rho)\left\|\boldsymbol{x}_k\right\|^2 + \Lambda_5 12B_{\mathrm{M}}^2L_f^2\phi_{\max}^2\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2\Theta_{\mathrm{M}}^2\xi_{\mathrm{M}}^2 + \Lambda_5 12B_{\mathrm{M}}^2L_f^2\phi_{\max}^2\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 e_{\mathrm{M}\boldsymbol{x},k}^2 \\
&\quad + \Lambda_5 6B_{\mathrm{M}}^2K_{\mathrm{M}}^2e_{\mathrm{M}\boldsymbol{x},k}^2 + 3\Lambda_5B_{\mathrm{M}}^2e_{\mathrm{M}\boldsymbol{u},k}^2 - \Lambda_1\Lambda_5(1-\zeta)\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 \\
&\quad - \Lambda_4\Lambda_5(1-\eta^2)\left(1+1/2^{\mathrm{R}}\right)^2K_{\mathrm{M}}^2e_{\mathrm{M}\boldsymbol{x},k}^2 \\
&\quad - \Lambda_2\Lambda_5(1-\zeta^2\eta^2)\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 e_{\mathrm{M}\boldsymbol{x},k}^2 - \Lambda_1\Lambda_5(1-\eta^2)\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 \\
&\quad - \Lambda_4\Lambda_5(1-\zeta^2\eta^2)\left(1+1/2^{\mathrm{R}}\right)^2L_f^2\phi_{\max}^2\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 e_{\mathrm{M}\boldsymbol{x},k}^2 \\
&\le -\Lambda_5(1-3\rho)\left\|\boldsymbol{x}_k\right\|^2 - \varpi\left\|\tilde{\boldsymbol{\theta}}_k\right\|^2 - \varpi e_{\mathrm{M}\boldsymbol{x},k}^2 - \varpi e_{\mathrm{M}\boldsymbol{u},k}^2
\end{aligned}$$

where $0 < \rho < \frac{1}{3}$, $0 < \Lambda_5 < 1$ and $0 < \varpi < 1$.

Therefore, first difference of Lyapunov function ΔL is negative definite while Lypaunov function L is positive definite. Moreover, using standard Lyapunov theory and geometric sequence theory, within finite horizon, the system states, parameter estimation error, state quantization error bound and control input quantization error bound will be uniformly ultimately bounded with ultimate bounds depending on initial condition $B_{\boldsymbol{x},0}, B_{\tilde{\boldsymbol{\theta}},0}, e^2_{\mathrm{M}\boldsymbol{x},0}, e^2_{\mathrm{M}\boldsymbol{u},0}$ with $\|\boldsymbol{x}(0)\|^2 \leq B_{\boldsymbol{x},0}$, $\|\tilde{\boldsymbol{\theta}}_0\|^2 \leq B_{\tilde{\boldsymbol{\theta}},0}$, $\|\boldsymbol{e}_{\boldsymbol{x},0}\|^2 \leq e^2_{\mathrm{M}\boldsymbol{x},0}$, $\|\boldsymbol{e}_{\boldsymbol{u},0}\|^2 \leq e^2_{\mathrm{M}\boldsymbol{u},0}$ and terminal time $\mathrm{NT_s}$, i.e.,

$$\begin{aligned}
&\|\boldsymbol{x}_k\|^2 \leq \left(1-(1-3\rho)\Lambda_5\right)^k B_{\boldsymbol{x},0} \equiv B_{\boldsymbol{x},k}, && \forall k = 0,1,\cdots,\mathrm{N} \\
&\|\tilde{\boldsymbol{\theta}}_k\|^2 \leq (1-\varpi)^k B_{\tilde{\boldsymbol{\theta}},0} \equiv B_{\tilde{\boldsymbol{\theta}},k}, && \forall k = 0,1,\cdots,\mathrm{N} \\
&\|\boldsymbol{e}_{\boldsymbol{x},k}\|^2 \leq (1-\varpi)^k e^2_{\mathrm{M}\boldsymbol{x},0} \equiv B_{\boldsymbol{e}_x,k}, && \forall k = 0,1,\cdots,\mathrm{N} \\
&\|\boldsymbol{e}_{\boldsymbol{u},k}\|^2 \leq (1-\varpi)^k e^2_{\mathrm{M}\boldsymbol{u},0} \equiv B_{\boldsymbol{e}_u,k}, && \forall k = 0,1,\cdots,\mathrm{N}
\end{aligned} \tag{A.18}$$

Further, since $0 < \rho < \dfrac{1}{3}$ and $0 < \varpi < 1$, the bounds in (A.18) are monotonically decreasing as k increases. When time goes infinity, i.e. $\mathrm{N} \to \infty$, all the bounds tend to zero and the asymptotic stability of the closed-loop system is achieved.

Eventually, while time goes to fixed final time $\mathrm{NT_s}$, we have the upper bound for $\hat{\boldsymbol{u}}^q_{qk} - \boldsymbol{u}^*_k$ as

$$\begin{aligned}
\|\hat{\boldsymbol{u}}^q_{qk} - \boldsymbol{u}^*_k\| &= \|\hat{\boldsymbol{K}}_k \boldsymbol{x}^q_k + \boldsymbol{e}_{\boldsymbol{u},k} - \boldsymbol{K}^*_k \boldsymbol{x}_k\| = \|\tilde{\boldsymbol{K}}_k \boldsymbol{x}_k + \boldsymbol{K}^*_k \boldsymbol{e}_{\boldsymbol{x},k} - \tilde{\boldsymbol{K}}_k \boldsymbol{e}_{\boldsymbol{x},k} + \boldsymbol{e}_{\boldsymbol{u},k}\| \\
&\leq \varsigma_K \|\tilde{\boldsymbol{\theta}}_k\| \|\boldsymbol{x}_k\| + \varsigma_K \|\tilde{\boldsymbol{\theta}}_k\| \|\boldsymbol{e}_{\boldsymbol{x},k}\| + K_{\mathrm{M}} \|\boldsymbol{e}_{\boldsymbol{x},k}\| + \|\boldsymbol{e}_{\boldsymbol{u},k}\| \\
&\leq \varsigma_K \sqrt{B_{\tilde{\boldsymbol{\theta}},k} B_{\boldsymbol{x},k}} + (\varsigma_K \sqrt{B_{\tilde{\boldsymbol{\theta}},k}} + K_{\mathrm{M}}) \sqrt{B_{\boldsymbol{e}_x,k}} + \sqrt{B_{\boldsymbol{e}_u,k}} \equiv \varepsilon_{\boldsymbol{u}s}
\end{aligned} \tag{A.19}$$

where $B_{\tilde{\boldsymbol{\theta}},k}, B_{\boldsymbol{x},k}, B_{\boldsymbol{e}_x,k}, B_{\boldsymbol{e}_u,k}$ are given in (A.18). Since all the bounds will converge to zeros when $\mathrm{N} \to \infty$, the estimated control input will tend to optimal control (i.e. $\hat{\boldsymbol{u}}^q_{qk} - \boldsymbol{u}^*_k$) due to (A.19).

III. NEURAL NETWORK-BASED FINITE-HORIZON OPTIMAL CONTROL OF UNCERTAIN AFFINE NONLINEAR DISCRETE-TIME SYSTEMS

Qiming Zhao, Hao Xu and S. Jagannathan

Abstract — In this work, the finite-horizon optimal control design for nonlinear discrete-time systems in affine form is presented. In contrast with the traditional approximate dynamic programming (ADP) methodology, which requires at least partial knowledge of the system dynamics, in this paper, the complete system dynamics are relaxed by utilizing a novel neural network (NN)-based identifier to learn the control coefficient matrix. The identifier is then used together with the actor-critic-based scheme to learn the time-varying solution, referred to as the value function, of the Hamilton-Jacobi-Bellman (HJB) equation in an online and forward-in-time manner. Due to the time-dependency of the solution, NNs with constant weights and time-varying activation functions are considered to handle the time-varying nature of the value function. To properly satisfy the terminal constraint, an additional error term is incorporated in the novel update law such that the terminal constraint error is also minimized over time. Policy and/or value iterations are not needed and the NN weights are updated once a sampling instant. The uniform ultimate boundedness (UUB) of the closed-loop system is verified by standard Lyapunov stability theory under non-autonomous analysis. Numerical examples are provided to illustrate the effectiveness of the proposed method.

1. INTRODUCTION

Conventionally, for linear systems with quadratic cost, the optimal regulation problem (LQR) can be tackled by solving the well-known Riccati Equation (RE) [1] with full knowledge of system dynamics $\boldsymbol{A}$ and $\boldsymbol{B}$. In addition, the solution is obtained offline and backward-in-time from the terminal constraint. In the case of infinite-horizon, the solution of the RE becomes a constant and the RE becomes the algebraic Riccati equation (ARE). However, the optimal control of nonlinear systems in affine form is more challenging since it requires the solution to the HJB equation. For infinite-horizon case, the HJB solution reduces to a time-invariant partial differential or difference equation. Therefore, in recent years, adaptive or NN-based optimal control over infinite-horizon has been studied for both linear and nonlinear systems, see [2][3][4]. However, the finite-horizon optimal control problem still remains unresolved for the control researchers.

First, for general affine nonlinear systems, the solution to the HJB equation is inherently time-varying [1] which complicates the analysis. Second, a terminal constraint is imposed on the cost function whereas this constraint is taken as zero in infinite-horizon case. The traditional ADP techniques [4][7][8] address the optimal control problem by solving the HJB equation iteratively. Though iteration-based solutions are mature, they are unsuitable for real-time implementation since inadequate number of iterations in a sampling interval can cause instability [2].

In the past literature, the author in [5] considered the finite-horizon optimal control of continuous-time nonlinear systems by iteratively solving the generalized HJB (GHJB) equation via Galerkin method from the terminal time. The authors in [6] proposed a fixed final-time optimal control for general affine nonlinear continuous-time systems by using a NN with time-dependent weights and state-dependent activation function to solve the HJB equation through backward integration.

On the other hand, in [7], the authors considered the finite-horizon optimal control of nonlinear discrete-time systems with input constraints by using off-line trained direct heuristic dynamic programming (DHDP)-based scheme utilizing a NN which incorporates constant weights and time-varying activation function. Similarly in [8], the authors considered the finite-horizon optimal control of discrete-time systems by using iteration-based ADP technique. However, in [8], the terminal time is not specified.

The past literature [5][6][7][8] for solving the finite-horizon optimal control of nonlinear systems utilize either backward-in-time integration or iteration-based offline training, which requires significant number of iterations within each sampling interval to guarantee the system stability. On the other hand, other ADP schemes [17] normally relax the drift dynamics while the control coefficient matrix is still needed [3]. Therefore, a real-time finite horizon optimal control scheme, which can be implemented in an online and forward-in-time manner with completely unknown system dynamics and without using value and policy iterations, is yet to be developed.

Therefore, in this paper, a novel approach is addressed to solve the finite-horizon optimal control of uncertain affine nonlinear discrete-time systems in an online and forward-in-time manner. First, the control coefficient matrix is generated by using a novel NN-based identifier which functions in an online manner. Next, an error term corresponding to the terminal constraint is defined and minimized overtime such that the terminal constraint can be properly satisfied. To handle the time-varying nature of the solution to the HJB equation or value function, NNs with constant weights and time-varying activation functions are utilized. In addition, in contrast with [7] and [8], the control policy is updated once every sampling instant and hence value/policy iterations are not performed. Finally, due to the time-dependency of the optimal control policy, the closed-loop system becomes essentially non-autonomous, and the stability of our proposed design scheme is demonstrated by Lyapunov stability analysis.

The main contribution of the paper includes the development of an optimal adaptive NN control scheme in finite horizon for nonlinear discrete-time systems without using value and/or policy iterations. An online identifier to generate the system dynamics is introduced and tuning laws for all the NNs are also derived. Lypunov stability is given.

The rest of the paper is organized as follows. In section 2, background and formulation of finite-horizon optimal control for affine nonlinear discrete-time systems are introduced. In section 3 the main control design scheme along with the stability analysis are addressed. In section 4, simulation results are given to verify the feasibility of our approach. Conclusive remarks are provided in Section 5.

2. BACKGROUND AND PROBLEM FORMULATION

In this paper, the finite-horizon optimal regulation for discrete-time affine nonlinear systems is investigated. The system is described as

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_k \tag{1}$$

where $\boldsymbol{x}_k \in \Omega_{\boldsymbol{x}} \subset \Re^n$ are the system states, $f(\boldsymbol{x}_k) \in \Re^n$ and $g(\boldsymbol{x}_k) \in \Re^{n\times m}$ are smooth *unknown* nonlinear dynamics, and $\boldsymbol{u}_k \in \Omega_{\boldsymbol{u}} \subset \Re^m$ is the control input vector. It is also assumed in this paper that $0 < \|g(\boldsymbol{x}_k)\| < g_{\mathrm{M}}$ with g_{M} being a positive constant.

Assumption 1: The nonlinear system given in (1) is controllable. Moreover, the system states $\boldsymbol{x}_k \in \Omega_{\boldsymbol{x}}$ are measurable.

The objective of the optimal control design is to determine a state feedback control policy which minimizes the following time-varying value or cost function given by

$$V(\boldsymbol{x}_k, k) = \psi(\boldsymbol{x}_\mathrm{N}) + \sum_{k=i}^{\mathrm{N}-1} L(\boldsymbol{x}_k, \boldsymbol{u}_k, k) \tag{2}$$

where $[i, \mathrm{N}]$ is the time span of interest, $\psi(\boldsymbol{x}_\mathrm{N})$ is the terminal constraint that penalizes the terminal state $\boldsymbol{x}_\mathrm{N}$, $L(\boldsymbol{x}_k, \boldsymbol{u}_k, k) = \boldsymbol{Q}(\boldsymbol{x}_k, k) + \boldsymbol{u}_k^\mathrm{T} \boldsymbol{R}_k \boldsymbol{u}_k$ is an in-general time-varying function of the state and control input at each intermediate time k in $[i, \mathrm{N}]$, where $\boldsymbol{Q}(\boldsymbol{x}_k, k) \in \Re$, $\boldsymbol{R}_k \in \Re^{m \times m}$ are positive semi-definite function and positive definite symmetric weighting matrix, respectively. It should be noted that in finite-horizon scenario, the control inputs can be time-varying, i.e., $\boldsymbol{u}_k = \mu(\boldsymbol{x}_k, k) \in \Omega_{\boldsymbol{u}}$.

Setting $k = \mathrm{N}$, the terminal constraint for the value function is given as

$$V(\boldsymbol{x}_\mathrm{N}, \mathrm{N}) = \psi(\boldsymbol{x}_\mathrm{N}) \tag{3}$$

Remark 1: In general, the terminal penalty $\psi(\boldsymbol{x}_\mathrm{N})$ is a function of state at terminal stage N and not necessarily to be in quadratic form. In the case of standard LQR, $\psi(\boldsymbol{x}_\mathrm{N})$ takes the quadratic form as $\psi(\boldsymbol{x}_\mathrm{N}) = \boldsymbol{x}_\mathrm{N}^\mathrm{T} \boldsymbol{Q}_\mathrm{N} \boldsymbol{x}_\mathrm{N}$ and the optimal control policy can be obtained by solving the RE in a backward-in-time fashion from the terminal value $\boldsymbol{Q}_\mathrm{N}$. It is also important to note that in the case of finite-horizon, the value function (2) becomes essentially time-dependent, in contrast with the infinite-horizon case where this problem is developed in a forward-in-time manner [2][3].

By Bellman's principle of optimality [1], the optimal cost from k onwards is equal to

$$V^*(\boldsymbol{x}_k,k)=\min_{\boldsymbol{u}_k}\left\{L(\boldsymbol{x}_k,\boldsymbol{u}_k,k)+V^*(\boldsymbol{x}_{k+1},k+1)\right\} \tag{4}$$

The optimal control policy $\boldsymbol{u}_k^*$ that minimizes the value function $V^*(\boldsymbol{x}_k,k)$ is obtained by using the stationarity condition $\partial V^*(\boldsymbol{x}_k,k)/\partial \boldsymbol{u}_k=0$ and revealed to be

$$\boldsymbol{u}_k^*=-\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\frac{\partial V^*(\boldsymbol{x}_{k+1},k+1)}{\partial \boldsymbol{x}_{k+1}} \tag{5}$$

From (5), it is clear that even the full system dynamics are available, the optimal control cannot be obtained for nonlinear discrete-time systems due to the dependency on future state $\boldsymbol{x}_{k+1}$. To avoid this drawback and relax the requirement for system dynamics, iteration-based schemes are normally utilized with NNs by performing offline-training [4]. However, iteration-based schemes are not preferred for hardware implementation since the number of iterations to ensure stability cannot be easily determined. Moreover, iterative approaches cannot be implemented when the system dynamics are completely unknown, since at least the control coefficient matrix $g(\boldsymbol{x}_k)$ is required to generate the control policy [3]. In contrast, in this work, a solution is found with completely unknown dynamics without utilizing iterative approach, as given in next section.

3. NEURAL NETWORK-BASED FINITE-HORIZON OPTIMAL REGULATION WITH COMPLETELY UNKNOWN DYNAMICS

In this section, the finite-horizon optimal regulation scheme for nonlinear discrete-time systems in affine form with completely unknown system dynamics is addressed. First, to relax the requirement of system dynamics, a novel NN-based identifier is designed to learn the true system dynamics in an online manner. Next, the actor-critic methodology is proposed to approximate the time-varying value function with a “critic” network, while the control inputs are generated by the “actor” network, with both NNs having the structure of constant weights and time-varying activation function. In order to satisfy the terminal constraint, an additional error term is defined and incorporated in the novel NN updating law such that this error is also minimized overtime. The stability of the closed-loop system is demonstrated, under non-autonomous analysis, by Lyapunov theory to show that the parameter estimation remains bounded as the system evolves.

3.1 NN-BASED IDENTIFIER DESIGN

Due to the online learning capability, NNs are commonly used for estimation and control. According to the universal approximation property [19], the system dynamics (1) can be rewritten on a compact set Ω by using NN representation as

$$\begin{aligned} \boldsymbol{x}_{k+1} &= f(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_k \\ &= \boldsymbol{W}_f^{\mathrm{T}}\sigma_f(\boldsymbol{x}_k) + \varepsilon_{fk} + \boldsymbol{W}_g^{\mathrm{T}}\sigma_g(\boldsymbol{x}_k)\boldsymbol{u}_k + \varepsilon_{gk}\boldsymbol{u}_k \\ &= \begin{bmatrix} \boldsymbol{W}_f \\ \boldsymbol{W}_g \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} \sigma_f(\boldsymbol{x}_k) & \boldsymbol{0} \\ \boldsymbol{0} & \sigma_g(\boldsymbol{x}_k) \end{bmatrix} \begin{bmatrix} 1 \\ \boldsymbol{u}_k \end{bmatrix} + [\varepsilon_f \quad \varepsilon_g] \begin{bmatrix} 1 \\ \boldsymbol{u}_k \end{bmatrix} \\ &= \boldsymbol{W}^{\mathrm{T}}\sigma(\boldsymbol{x}_k)\bar{\boldsymbol{u}}_k + \bar{\varepsilon}_k \end{aligned} \tag{6}$$

where $\boldsymbol{W} = \begin{bmatrix} \boldsymbol{W}_f \\ \boldsymbol{W}_g \end{bmatrix} \in \Re^{L\times n}$, $\sigma(\boldsymbol{x}_k) = \begin{bmatrix} \sigma_f(\boldsymbol{x}_k) & \boldsymbol{0} \\ \boldsymbol{0} & \sigma_g(\boldsymbol{x}_k) \end{bmatrix} \in \Re^{L\times(1+m)}$, $\bar{\boldsymbol{u}}_{k-1} = \begin{bmatrix} 1 \\ \boldsymbol{u}_{k-1} \end{bmatrix} \in \Re^{(1+m)}$ and $\bar{\boldsymbol{\varepsilon}}_{k-1} = [\varepsilon_f \quad \varepsilon_g]\bar{\boldsymbol{u}}_{k-1} \in \Re^n$, with L being the number of hidden neurons. In addition, the target NN weights are assumed to be upper bounded by $\|\boldsymbol{W}\| \le W_{\mathrm{M}}$, where W_{M} is a positive constant, while the NN activation function and reconstruction error are assumed to be bounded above as $\left\|\sigma(\boldsymbol{x}_k)\right\| \le \sigma_{\mathrm{M}}$ and $\left\|\bar{\boldsymbol{\varepsilon}}_k\right\| \le \bar{\varepsilon}_{\mathrm{M}}$, with σ_{M} and $\bar{\varepsilon}_{\mathrm{M}}$ positive constants. Note that to match the dimension, $\boldsymbol{W}$ can be constructed by stacking zeros in $\boldsymbol{W}_f$ or $\boldsymbol{W}_g$, which does not change the universal approximation property of the NN. Therefore, system dynamics $\boldsymbol{x}_k$ can be identified by updating the target NN weight matrix $\boldsymbol{W}$.

Using NN identifier, the system states at k can be estimated by

$$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1} \tag{7}$$

Define the identification error as

$$\boldsymbol{e}_k = \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k = \boldsymbol{x}_k - \hat{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1} \tag{8}$$

Then the identification error dynamics of (8) can be expressed as

$$\boldsymbol{e}_{k+1} = \boldsymbol{x}_{k+1} - \hat{\boldsymbol{x}}_{k+1} = \boldsymbol{x}_{k+1} - \hat{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \sigma(\boldsymbol{x}_k) \bar{\boldsymbol{u}}_k \tag{9}$$

Next, by incorporating the history information, define an augmented error vector as

$$\boldsymbol{\Xi}_k = \boldsymbol{X}_k - \hat{\boldsymbol{W}}_k^{\mathrm{T}} \boldsymbol{\Theta}_{k-1} \boldsymbol{U}_{k-1} \tag{10}$$

where $\boldsymbol{X}_k = [\boldsymbol{x}_k \quad \boldsymbol{x}_{k-1} \quad \cdots \quad \boldsymbol{x}_{k-l}] \in \Re^{n\times(l+1)}$, $\boldsymbol{\Theta}_{k-1} = [\sigma(\boldsymbol{x}_{k-1}) \quad \sigma(\boldsymbol{x}_{k-2}) \quad \cdots \quad \sigma(\boldsymbol{x}_{k-l-1})]$ $\in \Re^{L\times(l+1)(1+m)}$, and $\boldsymbol{U}_{k-1} = \begin{bmatrix} \bar{\boldsymbol{u}}_{k-1} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{0} & \bar{\boldsymbol{u}}_{k-2} & \boldsymbol{0} & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \cdots & \boldsymbol{0} & \bar{\boldsymbol{u}}_{k-l-1} \end{bmatrix} \in \Re^{(1+m)(l+1)\times(l+1)}$.

It can be seen that (10) includes a time history of previous $l+1$ identification errors recalculated using the most recent weights $\hat{\boldsymbol{W}}_k$.

Similar to (10), the dynamics for the augmented identification error vector becomes

$$\boldsymbol{\Xi}_{k+1} = \boldsymbol{X}_{k+1} - \hat{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \boldsymbol{\Theta}_k \boldsymbol{U}_k \tag{11}$$

Next, the update law for the NN identifier weights $\hat{\boldsymbol{W}}_k$ can be defined as

$$\hat{\boldsymbol{W}}_{k+1} = \boldsymbol{\Theta}_k \boldsymbol{U}_k \cdot (\boldsymbol{U}_k^{\mathrm{T}} \boldsymbol{\Theta}_k^{\mathrm{T}} \boldsymbol{\Theta}_k \boldsymbol{U}_k)^{-1} (\boldsymbol{X}_{k+1}^{\mathrm{T}} - \alpha \boldsymbol{\Xi}_k^{\mathrm{T}}) \tag{12}$$

where $0 < \alpha < 1$ is a design parameter.

Substituting (12) into (11) yields

$$\begin{aligned} \boldsymbol{\Xi}_{k+1} &= \boldsymbol{X}_{k+1} - \hat{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \boldsymbol{\Theta}_k \boldsymbol{U}_k \\ &= \boldsymbol{X}_{k+1} - (\boldsymbol{\Theta}_k \boldsymbol{U}_k \cdot (\boldsymbol{U}_k^{\mathrm{T}} \boldsymbol{\Theta}_k^{\mathrm{T}} \boldsymbol{\Theta}_k \boldsymbol{U}_k)^{-1} (\boldsymbol{X}_{k+1}^{\mathrm{T}} - \alpha \boldsymbol{\Xi}_k^{\mathrm{T}}))^{\mathrm{T}} \boldsymbol{\Theta}_k \boldsymbol{U}_k \\ &= \boldsymbol{X}_{k+1} - (\boldsymbol{X}_{k+1} - \alpha \boldsymbol{\Xi}_k)(\boldsymbol{U}_k^{\mathrm{T}} \boldsymbol{\Theta}_k^{\mathrm{T}} \boldsymbol{\Theta}_k \boldsymbol{U}_k)^{-1} \boldsymbol{U}_k^{\mathrm{T}} \boldsymbol{\Theta}_k^{\mathrm{T}} \boldsymbol{\Theta}_k \boldsymbol{U}_k \\ &= \alpha \boldsymbol{\Xi}_k \end{aligned} \tag{13}$$

Remark 2: For the above identification scheme, $\boldsymbol{\Theta}_k \boldsymbol{U}_k$ needs to be persistently exciting (PE) long enough for the NN identifier to learn the true system dynamics. PE condition is well-known in the adaptive control theory [21] and can be satisfied by adding probing noise [20].

Next, to find the NN weights estimation error dynamics, define $\tilde{\boldsymbol{W}}_k = \boldsymbol{W} - \hat{\boldsymbol{W}}_k$. Recall from (6) and (7), the identification error dynamics can be expressed as

$$\begin{aligned} \boldsymbol{e}_{k+1} &= \boldsymbol{x}_{k+1} - \hat{\boldsymbol{x}}_{k+1} = \boldsymbol{W}^{\mathrm{T}} \sigma(\boldsymbol{x}_k) \bar{\boldsymbol{u}}_k + \bar{\varepsilon}_k - \hat{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \sigma(\boldsymbol{x}_k) \bar{\boldsymbol{u}}_k \\ &= \tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \sigma(\boldsymbol{x}_k) \bar{\boldsymbol{u}}_k + \bar{\varepsilon}_k \end{aligned} \tag{14}$$

Using $\boldsymbol{e}_{k+1} = \alpha \boldsymbol{e}_k$ from (13), we have

$$\boldsymbol{e}_{k+1} = \tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \sigma(\boldsymbol{x}_k) \bar{\boldsymbol{u}}_k + \bar{\varepsilon}_k = \alpha (\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1} + \bar{\varepsilon}_{k-1}) \tag{15}$$

Or equivalently,

$$\tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}}\sigma(\boldsymbol{x}_k)\bar{\boldsymbol{u}}_k = \alpha\tilde{\boldsymbol{W}}_k^{\mathrm{T}}\sigma(\boldsymbol{x}_{k-1})\bar{\boldsymbol{u}}_{k-1} + \alpha\bar{\varepsilon}_{k-1} - \bar{\varepsilon}_k \tag{16}$$

Next, the boundedness of the NN weights estimation error $\tilde{\boldsymbol{W}}_k$ will be demonstrated in Theorem 1. The following definition is needed before proceeding.

Definition [19]: An equilibrium point $\boldsymbol{x}_e$ is said to be *uniformly ultimately bounded* (UUB) if there exists a compact set $\Omega_{\boldsymbol{x}} \subset \Re^n$ so that for all initial state $\boldsymbol{x}_0 \in \Omega_{\boldsymbol{x}}$, there exists a bound B and a time $T(B, \boldsymbol{x}_0)$ such that $\|\boldsymbol{x}_k - \boldsymbol{x}_e\| \le B$ for all $k \ge k_0 + T$.

Theorem 1 (*Boundedness of the NN identifier*): Let the nonlinear system (1) be controllable while the system state $\boldsymbol{x}_k \in \Omega_{\boldsymbol{x}}$ be measurable. Let the initial NN identifier weights $\hat{\boldsymbol{W}}_k$ be selected within a compact set Ω_{ID} which contains the ideal weights $\boldsymbol{W}$. Given the admissible control input $\boldsymbol{u}_0 \in \Omega_{\boldsymbol{u}}$, let the proposed NN identifier be defined as in (7) and the update law for tuning the NN weights be given in (12). Under the assumption that $\boldsymbol{\Theta}_k\boldsymbol{U}_k$ in Remark 2 satisfies persistency of excitation (PE) condition, then there exists a positive constant α satisfying $0 < \alpha < \frac{1}{2}$ such that the identification error $\boldsymbol{e}_k$ as well as the NN weights estimation error $\tilde{\boldsymbol{W}}_k$ are UUB, with the bound given in (A.5) and (A.6).

Proof: See Appendix.

Remark 3: In the proof, the inequality $0 < \Theta_{\mathrm{m}}^2 \le \|\boldsymbol{\Theta}_k\|^2 \le \|\boldsymbol{\Theta}_k\boldsymbol{U}_k\|^2$ holds since $\boldsymbol{\Theta}_k\boldsymbol{U}_k$ satisfies the PE condition [2] such that the NN identifier is able to learn the system dynamics. It should be also noted that the control input is assumed to be bounded, which is consistent with the literature, for the identification scheme since the main purpose of this section is to show the effectiveness of our identifier design. This assumption will be relaxed in our final theorem, where the convergence of the overall closed-loop system is shown with our proposed control design.

3.2 OPTIMAL NN CONTROLLER DESIGN

In this subsection, the finite-horizon optimal regulation design is proposed. To handle the time-dependency of the value function, two NNs with the structure of constant weights and time-varying activation functions are utilized to approximate the time-varying value function and the control input, respectively. An additional error term corresponding to the terminal constraint is also defined and minimized overtime such that the terminal constraint can be properly satisfied. Due to the time-dependency nature for finite-horizon, the closed-loop stability of the system will be shown by Lyapunov theory.

By universal approximation property of NNs [19] and actor-critic methodology, the value function and control inputs can be represented by a "critic" NN and an "actor" NN, respectively, as

$$V(\boldsymbol{x}_k,k)=\boldsymbol{W}_V^{\mathrm{T}}\sigma_V(\boldsymbol{x}_k,\mathrm{N}-k)+\varepsilon_V(\boldsymbol{x}_k,k) \tag{17}$$

and

$$\boldsymbol{u}(\boldsymbol{x}_k,k)=\boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,\mathrm{N}-k)+\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k,k) \tag{18}$$

where $\boldsymbol{W}_V$ and $\boldsymbol{W}_{\boldsymbol{u}}$ are the constant target NN weights, $\sigma_V(\boldsymbol{x}_k,\mathrm{N}-k)$ and $\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,\mathrm{N}-k)$ are the time-varying activation functions incorporating the time-to-go, $\varepsilon_V(\boldsymbol{x}_k,k)$ and $\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k,k)$ are the NN reconstruction errors for the critic and action network, respectively. The target NN weights are assumed to be upper bounded by $\|\boldsymbol{W}_V\| \le W_{V\mathrm{M}}$ and $\|\boldsymbol{W}_{\boldsymbol{u}}\| \le W_{\boldsymbol{u}\mathrm{M}}$, respectively, where both $W_{V\mathrm{M}}$ and $W_{\boldsymbol{u}\mathrm{M}}$ are positive constants [17]. The NN activation functions and the reconstruction errors are also assumed to be upper bounded by $\|\sigma_V(\boldsymbol{x}_k, \mathrm{N}-k)\| \le \sigma_{V\mathrm{M}}$, $\|\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, \mathrm{N}-k)\| \le \sigma_{\boldsymbol{u}\mathrm{M}}$, $|\varepsilon_V(\boldsymbol{x}_k, k)| \le \varepsilon_{V\mathrm{M}}$ and $|\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k, k)| \le \varepsilon_{\boldsymbol{u}\mathrm{M}}$, with $\sigma_{V\mathrm{M}}$, $\sigma_{\boldsymbol{u}\mathrm{M}}$, $\varepsilon_{V\mathrm{M}}$ and $\varepsilon_{\boldsymbol{u}\mathrm{M}}$ all positive constants [19]. In addition, in this work, the gradient of the reconstruction error is also assumed to be upper bounded by $\|\partial\varepsilon_{V,k} / \partial\boldsymbol{x}_{k+1}\| \le \varepsilon'_{V\mathrm{M}}$, with $\varepsilon'_{V\mathrm{M}}$ a positive constant [3][14].

Remark 4: In this paper, we utilize two NNs (critic and actor) to approximate the value function as well as the control inputs. Unlike continuous-time system, where the control inputs can be obtained directly from the information of critic NN, the actor NN is needed in discrete-time system since the future value $\boldsymbol{x}_{k+1}$ is not available. Therefore, the actor NN is utilized to relax the need for $\boldsymbol{x}_{k+1}$.

Similarly as (17), the terminal constraint of the value function can also be written in NN representation as

$$V(\boldsymbol{x}_\mathrm{N}, \mathrm{N}) = \boldsymbol{W}_V^\mathrm{T} \sigma_V(\boldsymbol{x}_\mathrm{N}, 0) + \varepsilon_V(\boldsymbol{x}_\mathrm{N}, \mathrm{N}) \tag{19}$$

with $\sigma_V(\boldsymbol{x}_\mathrm{N}, 0)$ and $\varepsilon_V(\boldsymbol{x}_\mathrm{N}, \mathrm{N})$ having the same meaning as $\sigma_V(\boldsymbol{x}_k, \mathrm{N}-k)$ and $\varepsilon_V(\boldsymbol{x}_k, k)$ but corresponding to the terminal state. Note that the activation function is taking the form $\sigma_V(\boldsymbol{x}_\mathrm{N}, 0)$ at terminal stage since from the definition, the time-varying activation function incorporates the time-to-go.

Remark 5: The fundamental difference between this work and [7] is that our proposed algorithm yields a completely forward-in-time and online solution without using both value and policy iteration and offline training, whereas the algorithm proposed in [7] was essentially an iteration-based DHDP scheme which is performed offline.

3.2.1 Value Function Approximation. The time-varying value function $V(\boldsymbol{x}_k, k)$ can be approximated by the critic NN and written as

$$\hat{V}(\boldsymbol{x}_k, k) = \hat{\boldsymbol{W}}_{V,k}^{\mathrm{T}} \sigma_V(\boldsymbol{x}_k, \mathrm{N}-k) \tag{20}$$

where $\hat{V}(\boldsymbol{x}_k, k)$ represents the estimated value function (2) and $\hat{\boldsymbol{W}}_{V,k}$ is estimation of the target NN weights $\boldsymbol{W}_V$. The basis function should satisfy $\|\sigma_V(0)\| = 0$ for $\|\boldsymbol{x}\| = 0$ to guarantee that $\hat{V}(0) = 0$ can be satisfied [1].

The terminal constraint can be represented by

$$\hat{V}(\boldsymbol{x}_{\mathrm{N}}, \mathrm{N}) = \hat{\boldsymbol{W}}_{V,k}^{\mathrm{T}} \sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}}, 0) \tag{21}$$

where $\hat{\boldsymbol{x}}_{\mathrm{N}}$ is an estimation of the terminal state. It should be noted that since the true value of $\boldsymbol{x}_{\mathrm{N}}$ is not known, $\hat{\boldsymbol{x}}_{\mathrm{N}}$ can be considered to be a “guess” of $\boldsymbol{x}_{\mathrm{N}}$ and can be chosen randomly as long as $\hat{\boldsymbol{x}}_{\mathrm{N}}$ lies within the stability region for a stabilizing control policy [4][7].

To ensure optimality, the Bellman equation should hold along the system trajectory. According to the principle of optimality, the true Bellman equation is given by

$$r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) + V(\boldsymbol{x}_{k+1}, k+1) - V(\boldsymbol{x}_k, k) = 0 \tag{22}$$

However, (22) no longer holds when the NN approximation is considered. Therefore, with estimated values, the Bellman equation (22) becomes

$$\begin{aligned} e_k^{\mathrm{B}} &= r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) + \hat{V}(\boldsymbol{x}_{k+1}, k+1) - \hat{V}(\boldsymbol{x}_k, k) \\ &= r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) + \hat{\boldsymbol{W}}_{V,k}^{\mathrm{T}} \sigma_V(\boldsymbol{x}_{k+1}, \mathrm{N}-k-1) - \hat{\boldsymbol{W}}_{V,k}^{\mathrm{T}} \sigma_V(\boldsymbol{x}_k, \mathrm{N}-k) \\ &= r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) + \hat{\boldsymbol{W}}_{V,k}^{\mathrm{T}} \Delta\sigma_V(\boldsymbol{x}_k, \mathrm{N}-k) \end{aligned} \tag{23}$$

where e_k^{B} is the Bellman error along the system trajectory, and $\Delta\sigma_V(\boldsymbol{x}_k,\mathrm{N}-k)=\sigma_V(\boldsymbol{x}_{k+1},\mathrm{N}-k-1)-\sigma_V(\boldsymbol{x}_k,\mathrm{N}-k)$.

Next, recall from (21), define an additional error term corresponding to the terminal constraint as

$$e_k^{\mathrm{N}}=\psi(\boldsymbol{x}_{\mathrm{N}})-\hat{\boldsymbol{W}}_{V,k}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},0) \tag{24}$$

The objective of the optimal control design is thus to minimize the Bellman error e_k^{B} as well as the terminal constraint error e_k^{N} as the system evolves. Hence, define the total error as

$$e_k^{\mathrm{total}}=e_k^{\mathrm{B}}+e_k^{\mathrm{N}} \tag{25}$$

Based on gradient descent, the update law for critic NN can be defined as

$$\hat{\boldsymbol{W}}_{V,k+1}=\hat{\boldsymbol{W}}_{V,k}-\alpha_V\frac{\sigma_1(\boldsymbol{x}_k,\mathrm{N}-k)e_k^{\mathrm{total}}}{1+\sigma_1^{\mathrm{T}}(\boldsymbol{x}_k,\mathrm{N}-k)\sigma_1(\boldsymbol{x}_k,\mathrm{N}-k)} \tag{26}$$

where $\sigma_1(\boldsymbol{x}_k,\mathrm{N}-k)=\Delta\sigma_V(\boldsymbol{x}_k,\mathrm{N}-k)-\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},0)$, while $\sigma_1(\boldsymbol{x}_k,\mathrm{N}-k)$ is bounded by $\sigma_{1\mathrm{m}}\le\left\|\sigma_1(\boldsymbol{x}_k,\mathrm{N}-k)\right\|\le\sigma_{1\mathrm{M}}$, and α_V is a design parameter with its range given in Theorem 2.

Remark 6: Two points needs to be clarified in the update law (26). First, the total error is minimized such that the optimality can be achieved as well as the terminal constraint can be also properly satisfied. Second, the activation function $\sigma_1(\boldsymbol{x}_k,\mathrm{N}-k)$ is also a combination of the activation function along the system trajectory and the activation function at the terminal stage. For the infinite-horizon case, the update law becomes a standard gradient descent algorithm with time-invariant activation function, and also the terms corresponding to the terminal constraint become all zero, i.e.,

$e_k^{\text{total}} = e_k^{\text{B}}$ and $\sigma_1(\boldsymbol{x}_k) = \Delta\sigma_V(\boldsymbol{x}_k)$.

Next, to find the error dynamics, define $\tilde{\boldsymbol{W}}_{V,k} = \boldsymbol{W}_V - \hat{\boldsymbol{W}}_{V,k}$. Recalling the Bellman equation (22) and the definition of the value function (17), we have

$$\begin{aligned} & r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) + V(\boldsymbol{x}_{k+1}, k+1) - V(\boldsymbol{x}_k, k) \\ &= r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) + \boldsymbol{W}_V^{\text{T}} \sigma_V(\boldsymbol{x}_{k+1}, \text{N} - k - 1) + \varepsilon_V(\boldsymbol{x}_{k+1}, k+1) \\ & \quad - \boldsymbol{W}_V^{\text{T}} \sigma_V(\boldsymbol{x}_k, \text{N} - k) - \varepsilon_V(\boldsymbol{x}_k, k) \\ &= r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) + \boldsymbol{W}_V^{\text{T}} \Delta\sigma_V(\boldsymbol{x}_k, \text{N} - k) + \Delta\varepsilon_V(\boldsymbol{x}_k, k) = 0 \end{aligned} \tag{27}$$

where $e_k^{\text{N}} = \psi(\boldsymbol{x}_{\text{N}}) - \hat{\boldsymbol{W}}_{Vk}^{\text{T}} \sigma_V(\boldsymbol{x}_{\text{N}}, 0)$.

Hence, we have

$$r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) = -\boldsymbol{W}_V^{\text{T}} \Delta\sigma_V(\boldsymbol{x}_k, \text{N} - k) - \Delta\varepsilon_V(\boldsymbol{x}_k, k) \tag{28}$$

Substituting (28) into (23) yields

$$\begin{aligned} e_k^{\text{B}} &= r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) + \hat{\boldsymbol{W}}_{V,k}^{\text{T}} \Delta\sigma_V(\boldsymbol{x}_k, \text{N} - k) \\ &= -\boldsymbol{W}_V^{\text{T}} \Delta\sigma_V(\boldsymbol{x}_k, \text{N} - k) - \Delta\varepsilon_V(\boldsymbol{x}_k, k) + \hat{\boldsymbol{W}}_{V,k}^{\text{T}} \Delta\sigma_V(\boldsymbol{x}_k, \text{N} - k) \\ &= -\tilde{\boldsymbol{W}}_{V,k}^{\text{T}} \Delta\sigma_V(\boldsymbol{x}_k, \text{N} - k) - \Delta\varepsilon_V(\boldsymbol{x}_k, k) \end{aligned} \tag{29}$$

Next Recalling from (19), then the terminal constraint error e_k^{N} can be written as

$$\begin{aligned} e_k^{\text{N}} &= \psi(\boldsymbol{x}_{\text{N}}) - \hat{\boldsymbol{W}}_{V,k}^{\text{T}} \sigma_V(\hat{\boldsymbol{x}}_{\text{N}}, 0) \\ &= \boldsymbol{W}_V^{\text{T}} \sigma_V(\boldsymbol{x}_{\text{N}}, 0) + \varepsilon_V(\boldsymbol{x}_{\text{N}}, 0) - \hat{\boldsymbol{W}}_{V,k}^{\text{T}} \sigma_V(\hat{\boldsymbol{x}}_{\text{N}}, 0) \\ &= \boldsymbol{W}_V^{\text{T}} \sigma_V(\boldsymbol{x}_{\text{N}}, 0) - \boldsymbol{W}_V^{\text{T}} \sigma_V(\hat{\boldsymbol{x}}_{\text{N}}, 0) + \boldsymbol{W}_V^{\text{T}} \sigma_V(\hat{\boldsymbol{x}}_{\text{N}}, 0) \\ & \quad + \varepsilon_V(\boldsymbol{x}_{\text{N}}, 0) - \hat{\boldsymbol{W}}_{V,k}^{\text{T}} \sigma_V(\hat{\boldsymbol{x}}_{\text{N}}, 0) \\ &= \tilde{\boldsymbol{W}}_{V,k}^{\text{T}} \sigma_V(\hat{\boldsymbol{x}}_{\text{N}}, 0) + \boldsymbol{W}_V^{\text{T}} \tilde{\sigma}_V(\boldsymbol{x}_{\text{N}}, 0) + \varepsilon_V(\boldsymbol{x}_{\text{N}}, 0) \end{aligned} \tag{30}$$

where $\tilde{\sigma}_V(\boldsymbol{x}_{\text{N}}, 0) = \sigma_V(\boldsymbol{x}_{\text{N}}, 0) - \sigma_V(\hat{\boldsymbol{x}}_{\text{N}}, 0)$.

Hence, the total error (25) becomes

$$
\begin{aligned}
e_k^{\text{total}} &= e_k^{\text{B}} + e_k^{\text{N}} \\
&= -\tilde{\boldsymbol{W}}_{V,k}^{\text{T}} \Delta\sigma_V(\boldsymbol{x}_k, \text{N}-k) - \Delta\varepsilon_V(\boldsymbol{x}_k, k) + \tilde{\boldsymbol{W}}_{V,k}^{\text{T}} \sigma_V(\hat{\boldsymbol{x}}_{\text{N}}, 0) \\
&\quad + \boldsymbol{W}_V^{\text{T}} \tilde{\sigma}_V(\boldsymbol{x}_{\text{N}}, 0) + \varepsilon_V(\boldsymbol{x}_{\text{N}}, 0) \\
&= -\tilde{\boldsymbol{W}}_{V,k}^{\text{T}} \sigma_1(\boldsymbol{x}_k, \text{N}-k) + \boldsymbol{W}_V^{\text{T}} \tilde{\sigma}_V(\boldsymbol{x}_{\text{N}}, 0) + \varepsilon(\boldsymbol{x}_k, k)
\end{aligned}
\tag{31}
$$

where $\varepsilon(\boldsymbol{x}_k, k) = -\Delta\varepsilon_V(\boldsymbol{x}_k, k) + \varepsilon_V(\boldsymbol{x}_{\text{N}}, 0)$.

Finally, by substituting (31) into the update law (26), the error dynamics for $\tilde{\boldsymbol{W}}_{V,k}$ is revealed to be

$$
\begin{aligned}
\tilde{\boldsymbol{W}}_{V,k+1} &= \tilde{\boldsymbol{W}}_{V,k} - \alpha_V \frac{\sigma_1(\boldsymbol{x}_k, \text{N}-k)\sigma_1^{\text{T}}(\boldsymbol{x}_k, \text{N}-k)\tilde{\boldsymbol{W}}_{V,k}}{1+\sigma_1^{\text{T}}(\boldsymbol{x}_k, \text{N}-k)\sigma_1(\boldsymbol{x}_k, \text{N}-k)} \\
&\quad + \alpha_V \frac{\sigma_1(\boldsymbol{x}_k, \text{N}-k)(\tilde{\sigma}_V^{\text{T}}(\boldsymbol{x}_{\text{N}}, 0)\boldsymbol{W}_V + \varepsilon(\boldsymbol{x}_k, k))}{1+\sigma_1^{\text{T}}(\boldsymbol{x}_k, \text{N}-k)\sigma_1(\boldsymbol{x}_k, \text{N}-k)}
\end{aligned}
\tag{32}
$$

Next, the boundedness of the estimation error for the critic NN weights is presented, as in the following theorem.

Theorem 2 (*Boundedness of the critic NN weights*): Let the nonlinear system (1) be controllable while the system state $\boldsymbol{x}_k \in \Omega_{\boldsymbol{x}}$ be measurable. Let the initial critic NN weights $\hat{\boldsymbol{W}}_{V,k}$ are selected within a compact set Ω_V which contains the ideal weights $\boldsymbol{W}_V$. Let $\boldsymbol{u}_0(\boldsymbol{x}_k) \in \Omega_{\boldsymbol{u}}$ be an admissible control for the system (1). Let the update law for the critic NN be given as (26). Under the assumptions stated in this paper, there exists a positive constant $0 < \alpha_V < \dfrac{2\sigma_{1\text{m}}^2}{3(1+\sigma_{1\text{M}}^2)}$ such that the critic NN weights estimation error $\tilde{\boldsymbol{W}}_{V,k}$ is UUB with a computable bound B_V given in (A.12).

Proof: See Appendix.

3.2.2 Approximation of Optimal Feedback Control Signal. In this subsection, the optimal control policy is obtained such that the estimated value function (20) is minimized. The action NN approximation of (18) is defined as

$$\hat{\boldsymbol{u}}(\boldsymbol{x}_k,k)=\hat{\boldsymbol{W}}_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,\mathrm{N}-k) \tag{33}$$

where $\hat{\boldsymbol{W}}_{\boldsymbol{u},k}$ is the estimation of the target action NN weights.

Next, define the actor error as the difference between the control policy applied to (1) and the control policy which minimizes the estimated value function (20), denoted as

$$\tilde{\boldsymbol{u}}(\boldsymbol{x}_k,k)=\hat{\boldsymbol{u}}_1(\boldsymbol{x}_k,k)-\hat{\boldsymbol{u}}_2(\boldsymbol{x}_k,k) \tag{34}$$

where $\hat{\boldsymbol{u}}_1(\boldsymbol{x}_k,k)=\hat{\boldsymbol{W}}_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,\mathrm{N}-k)$ and

$$\hat{\boldsymbol{u}}_2(\boldsymbol{x}_k,k)=-\frac{1}{2}\boldsymbol{R}^{-1}\hat{g}^{\mathrm{T}}(\boldsymbol{x}_k)\frac{\partial\hat{V}(\boldsymbol{x}_{k+1},k+1)}{\partial\boldsymbol{x}_{k+1}}=-\frac{1}{2}\boldsymbol{R}^{-1}\hat{g}^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},\mathrm{N}-k-1)\hat{\boldsymbol{W}}_{V,k},$$

where ∇ denotes the gradient, $\hat{g}(\boldsymbol{x}_k)$ is the estimated control coefficient matrix from the NN-based identifier and $\hat{V}(\boldsymbol{x}_{k+1},k+1)$ is the approximated value function from the critic network.

Hence, (34) becomes

$$\tilde{\boldsymbol{u}}(\boldsymbol{x}_k,k)=\hat{\boldsymbol{W}}_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,\mathrm{N}-k)+\frac{1}{2}\boldsymbol{R}^{-1}\hat{g}^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},\mathrm{N}-k-1)\hat{\boldsymbol{W}}_{V,k} \tag{35}$$

The update law for tuning the action NN weights can be then defined as

$$\hat{\boldsymbol{W}}_{\boldsymbol{u},k+1}=\hat{\boldsymbol{W}}_{\boldsymbol{u},k}-\alpha_{\boldsymbol{u}}\frac{\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,\mathrm{N}-k)\tilde{\boldsymbol{u}}^{\mathrm{T}}(\boldsymbol{x}_k,k)}{1+\sigma_{\boldsymbol{u}}^{\mathrm{T}}(\boldsymbol{x}_k,\mathrm{N}-k)\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,\mathrm{N}-k)} \tag{36}$$

where $\alpha_{\boldsymbol{u}}>0$ is a design parameter.

Recall that the control policy (18) minimizes the value function (17), then we have

$$\begin{aligned}\boldsymbol{u}(\boldsymbol{x}_k,k)&=\boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,\mathrm{N}-k)+\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k,k)\\&=-\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\left(\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},\mathrm{N}-k-1)\boldsymbol{W}_V+\nabla\varepsilon_V(\boldsymbol{x}_{k+1},k+1)\right)\end{aligned}$$

Or equivalently,

$$\begin{aligned}0=\boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,\mathrm{N}-k)+\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k,k)+\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\times\\\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},\mathrm{N}-k-1)\boldsymbol{W}_V+\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\varepsilon_V(\boldsymbol{x}_{k+1},k+1)\end{aligned}\tag{37}$$

To find the error dynamics for the actor NN weights $\hat{\boldsymbol{W}}_{\boldsymbol{u},k}$, define $\tilde{\boldsymbol{W}}_{\boldsymbol{u},k}=\boldsymbol{W}_{\boldsymbol{u}}-\hat{\boldsymbol{W}}_{\boldsymbol{u},k}$. Subtracting (37) from (35) yields

$$\begin{aligned}-\tilde{\boldsymbol{u}}(\boldsymbol{x}_k,k)=\tilde{\boldsymbol{W}}_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,\mathrm{N}-k)+\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},\mathrm{N}-k-1)\boldsymbol{W}_V\\+\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\varepsilon_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},k+1)-\frac{1}{2}\boldsymbol{R}^{-1}\hat{g}^{\mathrm{T}}(\boldsymbol{x}_k)\times\\\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},\mathrm{N}-k-1)\hat{\boldsymbol{W}}_{V,k}+\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k,k)\end{aligned}\tag{38}$$

Next, for simplicity, rewrite $\sigma_{\boldsymbol{u},k}=\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,\mathrm{N}-k)$, $\nabla\sigma_{V,k+1}=\nabla\sigma_V(\boldsymbol{x}_{k+1},\mathrm{N}-k-1)$, $\nabla\varepsilon_{V,k+1}=\nabla\varepsilon_V(\boldsymbol{x}_{k+1},\mathrm{N}-k-1)$, $\boldsymbol{g}_k=g(\boldsymbol{x}_k)$, $\hat{\boldsymbol{g}}_k=\hat{g}(\boldsymbol{x}_k)$, $\tilde{\boldsymbol{g}}_k=\tilde{g}(\boldsymbol{x}_k)$ and $\varepsilon_{\boldsymbol{u},k}=\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k,k)$, then add and subtract $\frac{1}{2}\boldsymbol{R}^{-1}\hat{g}^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\sigma_{V,k+1}^{\mathrm{T}}\boldsymbol{W}_V$ and arranging terms yields

$$\tilde{\boldsymbol{u}}(\boldsymbol{x}_k,k)=-\tilde{\boldsymbol{W}}_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k}-\frac{1}{2}\boldsymbol{R}^{-1}\tilde{\boldsymbol{g}}_k^{\mathrm{T}}\nabla\sigma_{V,k+1}^{\mathrm{T}}\boldsymbol{W}_V-\frac{1}{2}\boldsymbol{R}^{-1}\hat{\boldsymbol{g}}_k^{\mathrm{T}}\nabla\sigma_{V,k+1}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{V,k}-\tilde{\varepsilon}_{\boldsymbol{u},k}\tag{39}$$

where $\tilde{\boldsymbol{g}}_k=\boldsymbol{g}_k-\hat{\boldsymbol{g}}_k$ and $\tilde{\varepsilon}_{\boldsymbol{u},k}=\frac{1}{2}\boldsymbol{R}^{-1}\boldsymbol{g}_k^{\mathrm{T}}\nabla\varepsilon_{V,k+1}^{\mathrm{T}}+\varepsilon_{\boldsymbol{u},k}$. Furthermore, it can be easily concluded that $\tilde{\varepsilon}_{\boldsymbol{u},k}$ satisfies $\left\|\tilde{\varepsilon}_{\boldsymbol{u},k}\right\|\le\tilde{\varepsilon}_{\boldsymbol{u}\mathrm{M}}$, where $\tilde{\varepsilon}_{\boldsymbol{u}\mathrm{M}}$ is a positive constant.

Finally, the error dynamics for the actor NN weights are revealed to be

$$\begin{aligned}\tilde{\boldsymbol{W}}_{\boldsymbol{u},k+1} &= \tilde{\boldsymbol{W}}_{\boldsymbol{u},k} + \alpha_{\boldsymbol{u}} \frac{\sigma_{\boldsymbol{u},k}\tilde{\boldsymbol{u}}^{\mathrm{T}}(\boldsymbol{x}_k, k)}{1+\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k}} \\ &= \tilde{\boldsymbol{W}}_{\boldsymbol{u},k} - \alpha_{\boldsymbol{u}} \frac{\sigma_{\boldsymbol{u},k}}{1+\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k}} (\tilde{\boldsymbol{W}}_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k} + \frac{1}{2}\boldsymbol{R}^{-1}\tilde{\boldsymbol{g}}_k^{\mathrm{T}}\nabla\sigma_{V,k+1}^{\mathrm{T}}\boldsymbol{W}_V \\ &\quad + \frac{1}{2}\boldsymbol{R}^{-1}\hat{\boldsymbol{g}}_k^{\mathrm{T}}\nabla\sigma_{V,k+1}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{V,k} + \tilde{\varepsilon}_{\boldsymbol{u},k})^{\mathrm{T}}\end{aligned} \tag{40}$$

It should be noted that from the above analysis, the control matrix $g(\boldsymbol{x}_k)$ is not needed for updating the actor NN, in contrast with [3]. Instead, the approximated control matrix $\hat{g}(\boldsymbol{x}_k)$ from the NN identifier is utilized to find the control input, hence the partial knowledge of the system dynamics are relaxed.

To complete this subsection, the flowchart of this scheme is shown in Figure 1. We first collect the information for the steps $k = 1, 2, \cdots, l+1$ with the initial admissible control, which is defined later, for the first time identifier NN weights update. Then the NNs for the, critic, actor and identifier are updated based on our proposed weights tuning laws at each sampling interval in an online and forward-in-time fashion.

3.3 CONVERGENCE ANALYSIS

In this subsection, it will be shown that the closed-loop system will remain bounded. Before proceeding, the following definition and lemma are needed.

Definition [4]: Let $\Omega_{\boldsymbol{u}}$ denote the set of admissible control. A control function $\boldsymbol{u}: \Re^n \to \Re^m$ is defined to be admissible if the following is true:

$\boldsymbol{u}$ is continuous on $\Omega_{\boldsymbol{u}}$;

$\boldsymbol{u}(\boldsymbol{x})\big|_{\boldsymbol{x}=0} = 0$;

$\boldsymbol{u}(\boldsymbol{x})$ stabilize the system (1) on $\Omega_{\boldsymbol{x}}$;

$J(\boldsymbol{x}(0),\boldsymbol{u}) < \infty, \forall \boldsymbol{x}(0) \in \Omega_{\boldsymbol{x}}$.

Since the design scheme is similar to policy iteration, we need to solve a fixed-point equation rather than recursive equation. The initial admissible control guarantees the solution of the fixed-potion equation exists, thus the approximation process can be effectively done by our proposed scheme.

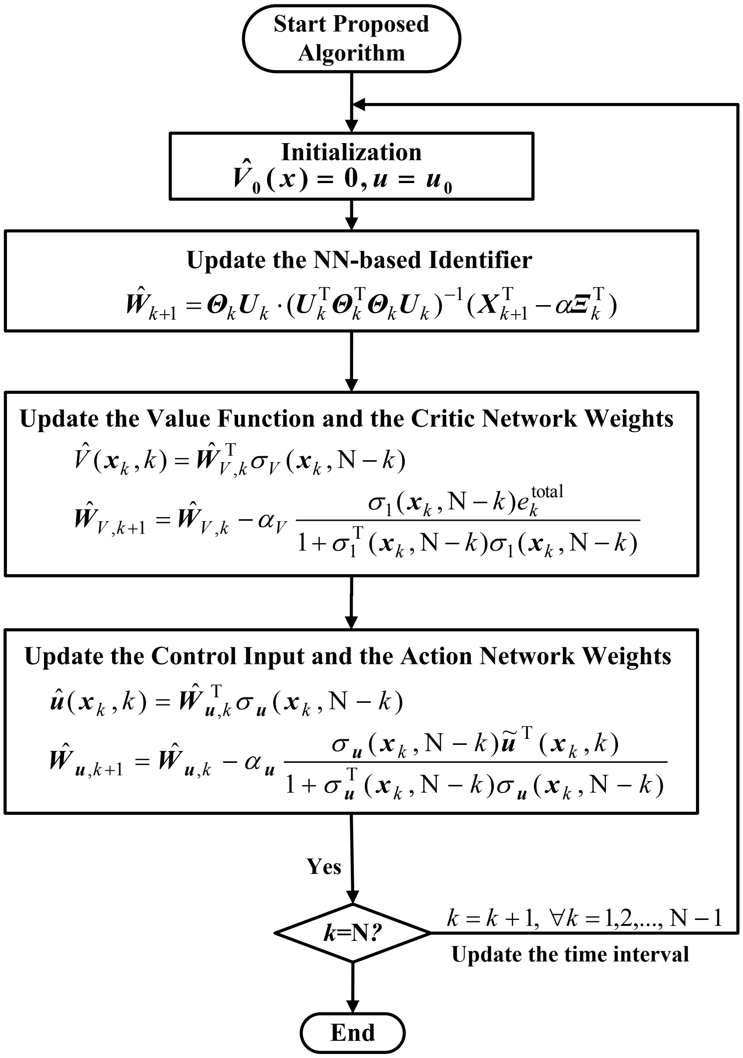

Start Proposed Algorithm
Initialization
$\hat{V}_0(\boldsymbol{x}) = 0, \boldsymbol{u} = \boldsymbol{u}_0$
Update the NN-based Identifier
$\hat{\boldsymbol{W}}_{k+1} = \boldsymbol{\Theta}_k \boldsymbol{U}_k \cdot (\boldsymbol{U}_k^{\mathrm{T}} \boldsymbol{\Theta}_k^{\mathrm{T}} \boldsymbol{\Theta}_k \boldsymbol{U}_k)^{-1} (\boldsymbol{X}_{k+1}^{\mathrm{T}} - \alpha \boldsymbol{\Xi}_k^{\mathrm{T}})$
Update the Value Function and the Critic Network Weights
$\hat{V}(\boldsymbol{x}_k, k) = \hat{\boldsymbol{W}}_{V,k}^{\mathrm{T}} \sigma_V(\boldsymbol{x}_k, \mathrm{N}-k)$
$\hat{\boldsymbol{W}}_{V,k+1} = \hat{\boldsymbol{W}}_{V,k} - \alpha_V \frac{\sigma_1(\boldsymbol{x}_k, \mathrm{N}-k) e_k^{\mathrm{total}}}{1 + \sigma_1^{\mathrm{T}}(\boldsymbol{x}_k, \mathrm{N}-k) \sigma_1(\boldsymbol{x}_k, \mathrm{N}-k)}$
Update the Control Input and the Action Network Weights
$\hat{\boldsymbol{u}}(\boldsymbol{x}_k, k) = \hat{\boldsymbol{W}}_{\boldsymbol{u},k}^{\mathrm{T}} \sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, \mathrm{N}-k)$
$\hat{\boldsymbol{W}}_{\boldsymbol{u},k+1} = \hat{\boldsymbol{W}}_{\boldsymbol{u},k} - \alpha_{\boldsymbol{u}} \frac{\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, \mathrm{N}-k) \tilde{\boldsymbol{u}}^{\mathrm{T}}(\boldsymbol{x}_k, k)}{1 + \sigma_{\boldsymbol{u}}^{\mathrm{T}}(\boldsymbol{x}_k, \mathrm{N}-k) \sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, \mathrm{N}-k)}$
Yes
k=N?
$k = k+1, \forall k = 1,2,..., \mathrm{N}-1$
Update the time interval
End

Figure 1. Flowchart of the finite-horizon optimal control design

Lemma 1 (*Bounds on the optimal closed-loop dynamics*): Consider the discrete-time affine nonlinear system defined in (1), then there exists an optimal control policy $\boldsymbol{u}_k^*$ for (1) such that the closed-loop system dynamics $f(\boldsymbol{x}_k)+g(\boldsymbol{x}_k)\boldsymbol{u}_k^*$ can be written as

$$\left\|f(\boldsymbol{x}_k)+g(\boldsymbol{x}_k)\boldsymbol{u}_k^*\right\|^2 \leq k^*\left\|\boldsymbol{x}_k\right\|^2 \tag{41}$$

where $0<k^*<\dfrac{1}{2}$ is a constant.

Theorem 3 (*Convergence of finite-horizon optimal control signal*) Let the nonlinear system (1) be controllable while the system state $\boldsymbol{x}_k \in \Omega_{\boldsymbol{x}}$ be measurable. Let the initial NN weights for the identifier, critic network and actor network $\hat{\boldsymbol{W}}_k$, $\hat{\boldsymbol{W}}_{V,k}$ and $\hat{\boldsymbol{W}}_{\boldsymbol{u},k}$ be selected within compact set Ω_{ID}, Ω_V and Ω_{AN} which contains the ideal weights $\boldsymbol{W}$, $\boldsymbol{W}_V$ and $\boldsymbol{W}_{\boldsymbol{u}}$. Let $\boldsymbol{u}_0(\boldsymbol{x}_k)\in\Omega_{\boldsymbol{u}}$ be an initial stabilizing control policy for the system (1). Let the NN weights update law for the identifier, critic network and actor network be provided by (12), (26) and (36), respectively. Then, under the assumption stated in this paper, there exists positive constants α, α_V, $\alpha_{\boldsymbol{u}}$ satisfying

$$0<\alpha<\frac{1}{2} \tag{42}$$

$$0<\alpha_{\boldsymbol{u}}<\frac{3}{7} \tag{43}$$

$$0<\alpha_V<\frac{\sigma_{1\mathrm{m}}^2}{4(1+\sigma_{1\mathrm{M}}^2)} \tag{44}$$

such that the system state $\boldsymbol{x}_k$, NN identification error $\boldsymbol{e}_k$, identifier weight estimation errors $\tilde{\boldsymbol{W}}_k$, critic and actor network weights estimation errors $\tilde{\boldsymbol{W}}_{V,k}$ and $\tilde{\boldsymbol{W}}_{\boldsymbol{u},k}$ are all UUB at terminal stage N with the bound $b_{\boldsymbol{x}}$, $b_{\boldsymbol{\Xi}}$, $b_{\boldsymbol{e}}$, $b_{\tilde{\boldsymbol{W}}}$ and $b_{\tilde{\boldsymbol{W}}_V}$ shown in (A.22) ~ (A.26). Moreover, the estimated control input is bounded closed to the optimal value such that $\left\|\boldsymbol{u}^*(\boldsymbol{x}_k,k)-\hat{\boldsymbol{u}}(\boldsymbol{x}_k,k)\right\|\le\varepsilon_{\boldsymbol{u}s}$ for a small positive constant $\varepsilon_{\boldsymbol{u}s}$.

Proof: See Appendix.

4. SIMULATIONS

In this section, the proposed algorithm is evaluated by two numerical examples. A linear system is first utilized followed by a practical two-link robot nonlinear system. For the linear system, one can compare the RE-based solution with the proposed scheme.

4.1 LINEAR CASE

The proposed finite-horizon optimal control design scheme is first evaluated by a linear example. Consider the system

$$\boldsymbol{x}_{k+1}=\begin{bmatrix}0.8 & 1\\ 0 & 0.6\end{bmatrix}\boldsymbol{x}_k+\begin{bmatrix}1\\ 0.5\end{bmatrix}\boldsymbol{u}_k \tag{45}$$

The weighting matrices for the performance index (2) are selected to be $\boldsymbol{Q}(\boldsymbol{x}_k,k)=0.5\boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{x}_k$ and $\boldsymbol{R}_k=1$. For comparison purpose, the terminal constraint is selected to be $\psi(\boldsymbol{x}_{\mathrm{N}})=0$. Non-zero terminal constraint is considered in nonlinear case.

For the NN setup, for linear systems, the input to the identifier NN is chosen to be $\boldsymbol{z}_k=[\boldsymbol{x}_k,\boldsymbol{u}_k]$, the time-varying activation functions for both critic and action network are chosen as $\sigma_V(\boldsymbol{x}_k,\mathrm{N}-k)=\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,\mathrm{N}-k)=[x_1,x_1\exp(-\tau),x_2,x_2\exp(-\tau),x_1^2,x_2^2,x_1x_2\tau]^{\mathrm{T}}$, which results 7 neurons, and $\tau=(\mathrm{N}-k)/\mathrm{N}$ is the normalized time-to-go.

The design parameters are chosen as $\alpha = 0.4$, $\alpha_V = 0.1$ and $\alpha_{\boldsymbol{u}} = 0.01$. The initial admissible control gain is selected as $\boldsymbol{K}(0) = [0.3, \ -0.3]$ and the initial system states are selected as $\boldsymbol{x}_0 = [0.5, -0.5]^{\mathrm{T}}$. The critic and action NN weights are both initialized as zeros. Simulation results are shown as below.

First, the system response is shown in Figure 2. It can be clearly seen from Figure 2 that the system states converge close to the origin within finite time. This confirms that the system remains stable under our proposed design scheme.

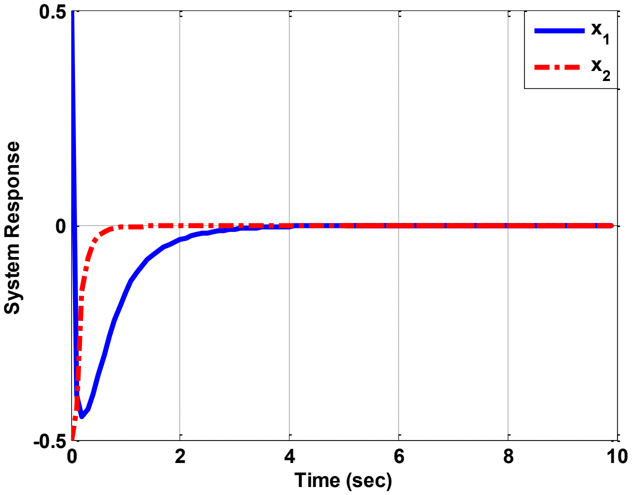


Figure 2. System response

Next, to show the feasibility of the proposed optimal control design scheme, the Bellman error as well as the terminal constraint error is plotted in Figure 3. It is shown from this figure that the Bellman equation error converges close to zero, which illustrates the fact that our proposed controller design indeed achieves optimality. It is more important to note that the convergence of the terminal constraint error indicates that the terminal constraint is also properly satisfied.

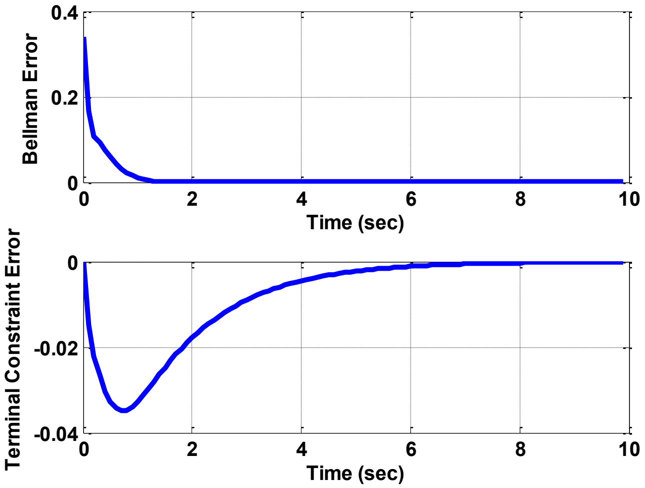


Figure 3. Error history

Next, the convergence of critic and actor NN weights is shown in Figure 4 and Figure 5, respectively. From the results, it can be clearly seen that both weights converge to constants and remain bounded, as desired.

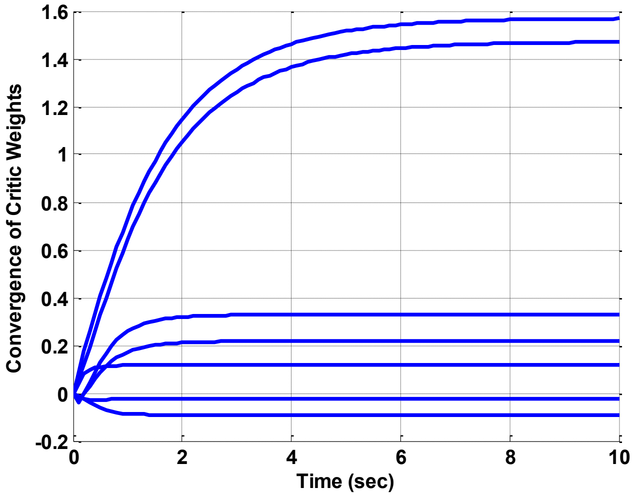


Figure 4. Convergence of critic NN weights

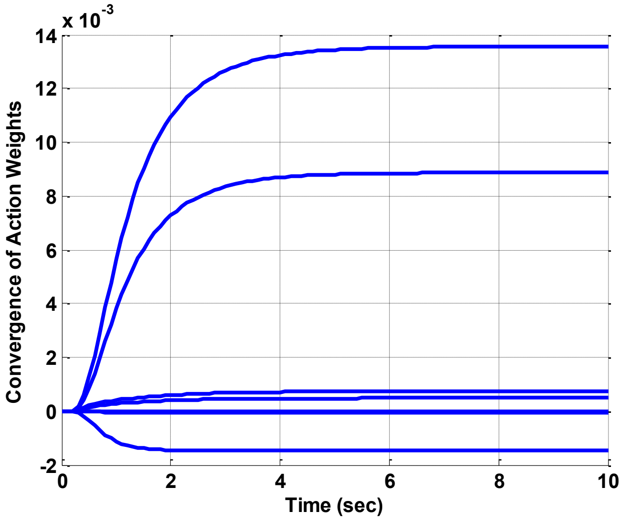


Figure 5. Convergence of actor NN weights

Finally, to compare our proposed design with traditional Riccati equation-based design, the cost is depicted in Figure 6. It can be seen from the figure that the difference between the cost computed from traditional RE-based and our proposed approach converges more quickly than the system states, which illustrates the validity of our proposed method.

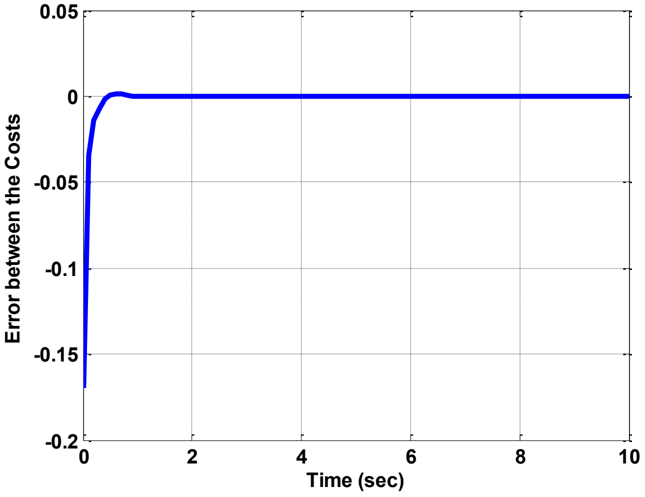


Figure 6. Cost between two methods

4.2 NONLINEAR CASE

Now, consider the two-link planar robot arm depicted in Figure 7.

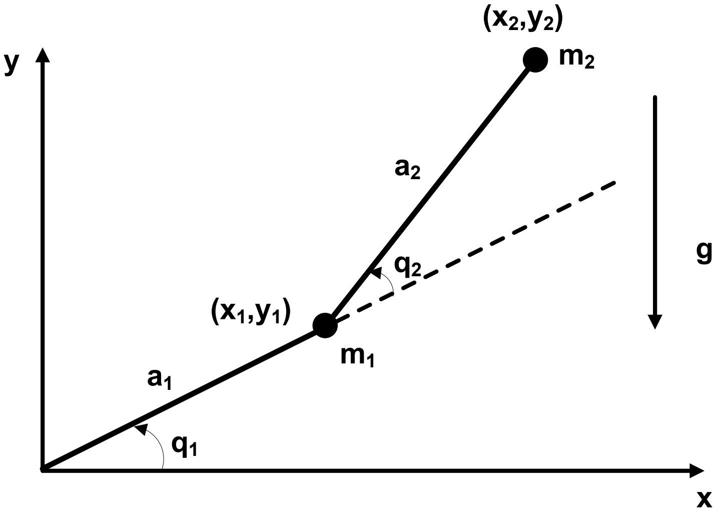


Figure 7. Two-link planar robot arm

The continuous-time dynamics of the two-link robot arm is given by [22]:

$$\begin{bmatrix} \alpha+\beta+2\eta\cos q_2 & \beta+\eta\cos q_2 \\ \beta+\eta\cos q_2 & \beta \end{bmatrix}\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix}+\begin{bmatrix} -\eta(2\dot{q}_1\dot{q}_2)\sin q_2 \\ \eta\dot{q}_1^2\sin q_2 \end{bmatrix}$$
$$+\begin{bmatrix} \alpha e_1\cos q_1+\eta e_1\cos(q_1+q_2) \\ \eta e_1\cos(q_1+q_2) \end{bmatrix}=\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \tag{46}$$

where $\alpha=(m_1+m_2)a_1^2$, $\beta=m_2a_2^2$, $\eta=m_2a_1a_2$, $e=g/a_1$. In the simulation, the parameters are chosen to be $m_1=m_2=1\text{kg}$, $a_1=a_2=1\text{m}$ and $g=10\,\text{m}/s^2$. Hence, $\alpha=2$, $\beta=1$, $\eta=1$ and $e_1=10$.

Define the system states as $\boldsymbol{x}=[x_1,x_2,x_3,x_4]^{\mathrm{T}}=[q_1,q_2,\dot{q}_1,\dot{q}_2]^{\mathrm{T}}$ and the control inputs as $\boldsymbol{u}=[u_1,u_2]^{\mathrm{T}}=[\tau_1,\tau_2]^{\mathrm{T}}$. Then the system dynamics can be written in the affine form as $\dot{\boldsymbol{x}}=f(\boldsymbol{x})+g(\boldsymbol{x})\boldsymbol{u}$, where

$$f(\boldsymbol{x})=\begin{bmatrix} x_3 \\ x_4 \\ \dfrac{\begin{pmatrix} -(2x_3x_4+x_4^2-x_3^2-x_3^2\cos x_2)\sin x_2 \\ +20\cos x_1-10\cos(x_1+x_2)\cos x_2 \end{pmatrix}}{\cos^2 x_2-2} \\ \dfrac{\begin{pmatrix} (2x_3x_4+x_4^2+2x_3x_4\cos x_2+x_4^2\cos x_2+3x_3^2 \\ +2x_3^2\cos x_2+20(\cos(x_1+x_2)-\cos x_1)\times \\ (1+\cos x_2)-10\cos x_2\cos(x_1+x_2) \end{pmatrix}}{\cos^2 x_2-2} \end{bmatrix}, \text{ and } g(\boldsymbol{x})=\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \dfrac{1}{2-\cos^2 x_2} & \dfrac{-1-\cos x_2}{2-\cos^2 x_2} \\ \dfrac{-1-\cos x_2}{2-\cos^2 x_2} & \dfrac{3+2\cos x_2}{2-\cos^2 x_2} \end{bmatrix}.$$

Discretizing the continuous-time system with a sufficient small sampling interval T_s, then the discrete-time version of the system can be written as $\boldsymbol{x}_{k+1}=f(\boldsymbol{x}_k)+g(\boldsymbol{x}_k)\boldsymbol{u}_k$, where

$$f(\boldsymbol{x}_k)=\begin{bmatrix} T_s x_{3k}+x_{1k} \\ T_s x_{4k}+x_{2k} \\ \dfrac{\begin{pmatrix} -(2x_{3k}x_{4k}+x_{4k}^2-x_{3k}^2-x_{3k}^2\cos x_{2k})\sin x_{2k} \\ +20\cos x_{1k}-10\cos(x_{1k}+x_{2k})\cos x_{2k} \end{pmatrix}}{(\cos^2 x_{2k}-2)/T_s}+x_{3k} \\ \dfrac{\begin{pmatrix} (2x_{3k}x_{4k}+x_{4k}^2+2x_{3k}x_{4k}\cos x_{2k}+x_{4k}^2\cos x_{2k} \\ +3x_{3k}^2+2x_{3k}^2\cos x_{2k}+20(\cos(x_{1k}+x_{2k})-\cos x_{1k})\times \\ (1+\cos x_{2k})-10\cos x_{2k}\cos(x_{1k}+x_{2k}) \end{pmatrix}}{(\cos^2 x_{2k}-2)/T_s}+x_{4k} \end{bmatrix},$$

$$g(\boldsymbol{x}_k)=\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \dfrac{1}{(\cos^2 x_{2k}-2)/T_s} & \dfrac{-1-\cos x_{2k}}{(\cos^2 x_{2k}-2)/T_s} \\ \dfrac{-1-\cos x_{2k}}{(\cos^2 x_{2k}-2)/T_s} & \dfrac{3+2\cos x_{2k}}{(\cos^2 x_{2k}-2)/T_s} \end{bmatrix}.$$

In the simulation, we choose $T_s=0.001$, and the value function is given in the form of (2), with $\boldsymbol{Q}(\boldsymbol{x}_k,k)=\boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{x}_k$ an identity matrix with appropriate dimension and

$\boldsymbol{R}=0.005\boldsymbol{I}$. The initial states and admissible control gain are selected to be $\boldsymbol{x}(0)=[\pi/3,\ \pi/6,\ 0,\ 0]^{\mathrm{T}}$ and $\boldsymbol{K}(0)=[-50,0-50,0;20,0,20,-20]$, and the terminal constraint is given as $\psi(\boldsymbol{x}_{\mathrm{N}})=8$.

For the NN setup, the activation function for the identifier is constructed from the expansion of the even polynomial $\sum_{\beta=1}^{M/2}\left(\sum_{i=1}^{n}x_i\right)^{2\beta}$, where M is the order of approximation and n is the dimension of the system. In our case, $n=4$ and we choose $M=4$, which results in 45 neurons. For the critic and action network, the state-dependent part of the time-varying activation functions is also chosen to be the expansion of the even polynomial with $M=4$ and $M=2$, which results in 45 and 10 neurons, respectively, while the time-dependent part are selected as the polynomials of time-to-go with saturation, i.e., $\{0,(\mathrm{N}-k)/L_i\ ,(\mathrm{N}-k)/(L_i-1),\cdots,\mathrm{N}-k\}$, where N is the terminal time and L_i is the number of neurons. In our case, $L_1=45$ and $L_2=10$. Note that saturation for the time-dependent part of the activation function is to ensure its magnitude is within a reasonable range such that the parameter estimation is computable. The tuning parameters are chosen as $\alpha=0.3$, $\alpha_V=0.01$ and $\alpha_{\boldsymbol{u}}=0.1$. All the initial NN weights are randomly selected between $[0,1]$. The simulation results are shown as below.

First, the system response, control inputs and identification errors are given in Figure 8 and Figure 9, respectively. It can be seen clearly from these two figures that the system states, control inputs and identification errors converge close to the origin in finite time, which shows the stability of the system and effectiveness of the NN identifier design.

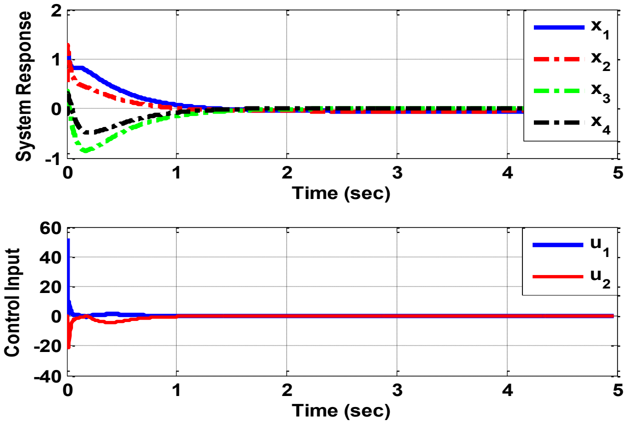


Figure 8. System response and control inputs

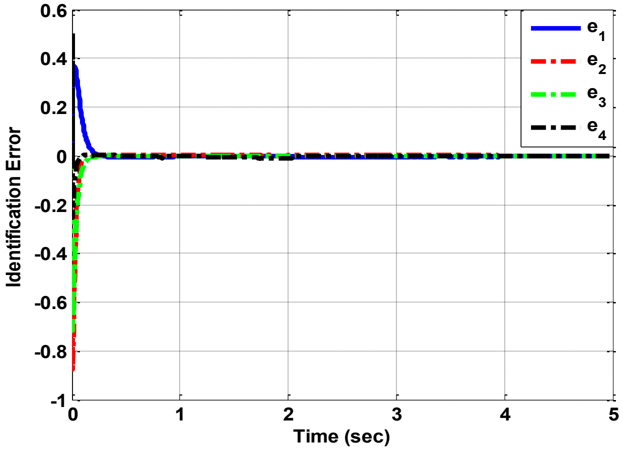


Figure 9. Identification errors

Next, to show the feasibility of our proposed optimal control design scheme, the error histories are plotted in Figure 10. Similar trends as the linear case are shown from Figure 10 that both Bellman equation error and terminal constraint error converge close to zero as system evolves, which illustrates that the proposed algorithm not only achieves optimality but also satisfies the terminal constraint.

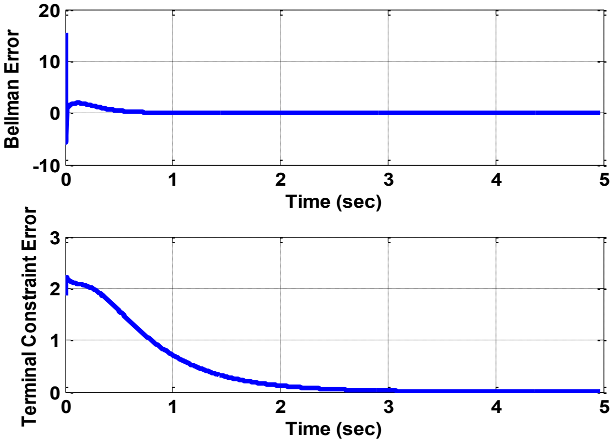


Figure 10. Error histories

Finally, due to the large number of neurons for the critic and actor NN, the norm of the NN weights is shown in Figure 11. It can be clearly seen from the figure that the actual NN weights converge to a constant, as desired.

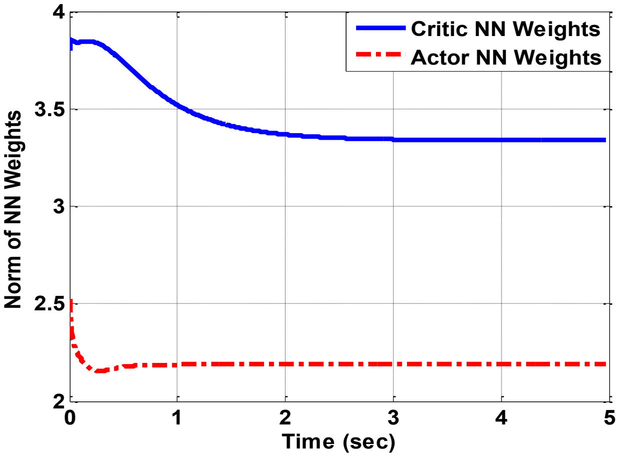


Figure 11. Convergence of critic and actor NN weights

5. CONCLUSIONS

In this paper, the finite-horizon optimal control of affine nonlinear discrete-time systems is addressed with completely unknown system dynamics. First, the NN-based identifier generates suitable control coefficient matrix such that the control input can be computed. Next, the actor-critic structure is utilized to approximately find the optimal control policy. The time-varying nature for finite-horizon optimal control problem is handled by using NNs with constant weights and time-varying activation functions. An additional error term corresponding to the terminal constraint is minimized to guarantee that the terminal constraint can be properly satisfied. In addition, the proposed algorithm is implemented by utilizing a history of cost to go errors instead of traditional iteration-based scheme. The proposed algorithm yields an online and forward-in-time design scheme which enjoys great practical advantages. The convergence of the parameter estimation and closed-loop system are demonstrated by using Lyapunov stability theory under non-autonomous analysis. Simulation results verify the theoretical claim.

6. REFERENCES

[1] F. L. Lewis and V. L. Syrmos, Optimal Control, 2nd edition. New York: Wiley, 1995.

[2] H. Xu and S. Jagannathan, “Stochastic optimal controller design for uncertain nonlinear networked control system via neuro dynamic programming,” IEEE Trans. Neural Netw. and Learning Syst, vol. 24, pp. 471–484, 2013.

[3] T. Dierks and S. Jagannathan, “Online optimal control of affine nonlinear discrete-time systems with unknown internal dynamics by using time-based policy update,” IEEE Trans. Neural Netw. and Learning Syst, vol. 23, pp. 1118–1129, 2012.

[4] Z. Chen and S. Jagannathan, “Generalized Hamilton-Jacobi-Bellman formulation based neural network control of affine nonlinear discrete-time systems”, IEEE Trans. Neural Network, vol. 7, pp. 90–106, 2008.

[5] R. Beard, “Improving the closed-loop performance of nonlinear systems,” Ph.D. dissertation, Rensselaer Polytechnic Institute, USA, 1995.

[6] T. Cheng, F. L. Lewis, and M. Abu-Khalaf, “A neural network solution for fixed-final-time optimal control of nonlinear systems,” Automatica, vol. 43, pp. 482–490, 2007.

[7] A. Heydari and S. N. Balakrishan, “Finite-horizon control-constrained nonlinear optimal control using single network adaptive critics,” IEEE Trans. Neural Netw. and Learning Syst., vol. 24, pp. 145–157, 2013.

[8] F.Y. Wang, N. Jin, D. Liu and Q. Wei, “Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with ε –error bound,” IEEE Trans. Neural Networks, vol. 22, pp. 24–36, 2011.

[9] Q. Zhao, H. Xu and S. Jagannathan, “Finite-Horizon Optimal Adaptive Neural Network Control of Uncertain Nonlinear Discrete-time Systems,” to appear in IEEE Multi-conference on Systems and Control, Hyderabad, India, 2013.

[10] C. Watkins, “Learning from Delayed Rewards,” Ph.D. dissertation, Cambridge University, England, 1989.

[11] H. K. Khalil, Nonlinear Systems, 3rd ed, Prentice-Hall, NJ, 2002.

[12] P. J. Werbos, “A menu of designs for reinforcement learing over time,” J. Neural Network Contr., vol. 3, pp. 835–846, 1983.

[13] T. Dierks and S. Jagannathan, “Optimal control of affine nonlinear discrete-time systems,” in Proc. Medit. Conf. Control Autom., pp. 1390–1395, 2009.

[14] T. Dierks and S. Jagannathan, “Optimal control of affine nonlinear discrete-time systems with unknown internal dynamics,” in Proc. Conf. on Decision and Control, Shanghai, pp. 6750–6755, 2009.

[15] D. P. Bertsekas, Dynamic Programming and Optimal Control, 2nd ed. Belmonth, MA: Athena Scientific, 2000.

[16] T. Dierks, B.T. Thumati and S. Jagannathan, “Optimal control of unknown affine nonlinear discrete-time systems using offline-trained neural networks with prrof of convergence,” Neural Networks, pp. 851–860, 2009.

[17] J. Si, A. G. Barto, W. B. Powell and D. Wunsch, Handbook of Learning and Approximate Dynamics Programming. New York: Wiley, 2004.

[18] D. V. Prokhorov and D. Wunsch, “Adaptive critic designs,” IEEE Trans. Neural Netw., vol 8, pp. 997–1007, 1997.

[19] S. Jagannathan, Neural Network Control of Nonlinear Discrete-Time Systems, Boca Raton, FL: CRC Press, 2006.

[20] M. Green and J. B. Moore, “Persistency of excitation in linear systems,” Syst. and Cont. Letter, vol. 7, pp. 351–360, 1986.

[21] K. S. Narendra and A. M. Annaswamy, Stable Adaptive Systems, New Jersey: Prentice-Hall, 1989.

[22] F. L. Lewis, S. Jagannathan, and A. Yesilderek, Neural Network Control of Robot Manipulator and Nonlinear Systems. London, UK,: Taylor & Francis, 1999.

APPENDIX

Proof of Theorem 1: First observe that $0 < \Theta_{\mathrm{m}}^2 \le \left\|\boldsymbol{\Theta}_k\right\|^2 \le \left\|\boldsymbol{\Theta}_k \boldsymbol{U}_k\right\|^2$, where Θ_{m} is a positive constant. This is ensured by the PE condition. Define the Lyapunov candidate function as

$$L_{\mathrm{ID}}(k) = \boldsymbol{e}_k^{\mathrm{T}} \boldsymbol{e}_k + \Theta_{\mathrm{m}}^2 \mathrm{tr}(\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \tilde{\boldsymbol{W}}_k) + \left\| \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1} \right\|^2 \tag{A.1}$$

where $\mathrm{tr}(\bullet)$ denotes the trace operator. The first difference of $L_{\mathrm{ID}}(k)$ becomes

$$\begin{aligned}
\Delta L_{\mathrm{ID}}(k) &= L_{\mathrm{ID}}(k+1) - L_{\mathrm{ID}}(k) \\
&= \boldsymbol{e}_{k+1}^{\mathrm{T}} \boldsymbol{e}_{k+1} + \Theta_{\mathrm{m}}^2 \mathrm{tr}(\tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \tilde{\boldsymbol{W}}_{k+1}) - \boldsymbol{e}_k^{\mathrm{T}} \boldsymbol{e}_k - \Theta_{\mathrm{m}}^2 \mathrm{tr}(\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \tilde{\boldsymbol{W}}_k) \\
&\quad + \left\| \tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \sigma(\boldsymbol{x}_k) \bar{\boldsymbol{u}}_k \right\|^2 - \left\| \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1} \right\|^2 \\
&\le \boldsymbol{e}_{k+1}^{\mathrm{T}} \boldsymbol{e}_{k+1} - \boldsymbol{e}_k^{\mathrm{T}} \boldsymbol{e}_k + \Theta_{\mathrm{m}}^2 \mathrm{tr}\Big((\tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \sigma(\boldsymbol{x}_k) \bar{\boldsymbol{u}}_k)^{\mathrm{T}} (\tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \sigma(\boldsymbol{x}_k) \bar{\boldsymbol{u}}_k) \Big) - \Theta_{\mathrm{m}}^2 \mathrm{tr}(\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \tilde{\boldsymbol{W}}_k) \\
&\quad + \left\| \tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \sigma(\boldsymbol{x}_k) \bar{\boldsymbol{u}}_k \right\|^2 - \left\| \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1} \right\|^2
\end{aligned} \tag{A.2}$$

Recall $\boldsymbol{e}_{k+1} = \alpha \boldsymbol{e}_k$ and (16), (A.1) can be further written as

$$\begin{aligned}
\Delta L_{\mathrm{ID}}(k) &\le \alpha^2 \left\| \boldsymbol{e}_k \right\|^2 - \left\| \boldsymbol{e}_k \right\|^2 + \left\| \tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \sigma(\boldsymbol{x}_k) \bar{\boldsymbol{u}}_k \right\|^2 - \Theta_{\mathrm{m}}^2 \mathrm{tr}(\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \tilde{\boldsymbol{W}}_k) \\
&\quad + \left\| \alpha \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1} + \alpha \bar{\varepsilon}_{k-1} - \bar{\varepsilon}_k \right\|^2 - \left\| \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1} \right\|^2 \\
&\le -(1-\alpha^2) \left\| \boldsymbol{e}_k \right\|^2 + 2 \left\| \alpha \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1} + \alpha \bar{\varepsilon}_{k-1} - \bar{\varepsilon}_k \right\|^2 \\
&\quad - \Theta_{\mathrm{m}}^2 \mathrm{tr}(\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \tilde{\boldsymbol{W}}_k) - \left\| \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1} \right\|^2
\end{aligned} \tag{A.3}$$

Using Cauchy-Schwartz inequality, (A.2) can be written as

$$\begin{aligned}
\Delta L_{\mathrm{ID}}(k) &\le -(1-\alpha^2) \left\| \boldsymbol{e}_k \right\|^2 + 4\alpha^2 \left\| \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1} \right\|^2 \\
&\quad + 4 \left\| \alpha \bar{\varepsilon}_{k-1} - \bar{\varepsilon}_k \right\|^2 - \Theta_{\mathrm{m}}^2 \mathrm{tr}(\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \tilde{\boldsymbol{W}}_k) - \left\| \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1} \right\|^2 \\
&\le -(1-\alpha^2) \left\| \boldsymbol{e}_k \right\|^2 - \Theta_{\mathrm{m}}^2 \left\| \tilde{\boldsymbol{W}}_k \right\|^2 - (1-4\alpha^2) \left\| \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1} \right\|^2 + 4 \left\| \alpha \bar{\varepsilon}_{k-1} - \bar{\varepsilon}_k \right\|^2 \\
&\le -(1-\alpha^2) \left\| \boldsymbol{e}_k \right\|^2 - \Theta_{\mathrm{m}}^2 \left\| \tilde{\boldsymbol{W}}_k \right\|^2 + \Delta \bar{\varepsilon}_{\mathrm{M}}
\end{aligned} \tag{A.4}$$

where $4 \left\| \alpha \bar{\varepsilon}_{k-1} - \bar{\varepsilon}_k \right\|^2 \le \Delta \bar{\varepsilon}_{\mathrm{M}}$ due to the boundedness of the NN reconstruction error, with

$\Delta\bar{\varepsilon}_{\mathrm{M}}$ a positive constant.

Therefore, the first difference of $L_{\mathrm{ID}}(k)$ is less than zero outside of a compact set as long the following conditions hold

$$\left\|\boldsymbol{e}_k\right\| > \sqrt{\frac{\Delta\bar{\varepsilon}_{\mathrm{M}}}{(1-\alpha^2)}} \equiv b_{\boldsymbol{e}} \tag{A.5}$$

Or

$$\left\|\tilde{\boldsymbol{W}}_k\right\| > \sqrt{\frac{\Delta\bar{\varepsilon}_{\mathrm{M}}}{\Theta_{\mathrm{m}}^2}} \equiv b_{\tilde{\boldsymbol{W}}} \tag{A.6}$$

Proof of Theorem 2: Define the Lyapunov candidate function as

$$L_V(\tilde{\boldsymbol{W}}_{V,k}) = \tilde{\boldsymbol{W}}_{V,k}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{V,k} \tag{A.7}$$

The first difference of $L_V(\tilde{\boldsymbol{W}}_{V,k})$ is given by

$$\Delta L_V(\tilde{\boldsymbol{W}}_{V,k}) = \tilde{\boldsymbol{W}}_{V,k+1}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{V,k+1} - \tilde{\boldsymbol{W}}_{V,k}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{V,k} \tag{A.8}$$

Denote $\sigma_{1k} = \sigma_1(\boldsymbol{x}_k, \mathrm{N}-k)$ for simplicity. Substituting (32) into (A.8) yields

$$\begin{aligned}
\Delta L_V(\tilde{\boldsymbol{W}}_{V,k}) &= \tilde{\boldsymbol{W}}_{V,k+1}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{V,k+1} - \tilde{\boldsymbol{W}}_{V,k}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{V,k} \\
&= \left[\tilde{\boldsymbol{W}}_{Vk} - \alpha_V \frac{\sigma_{1k}^{\mathrm{T}}\sigma_{1k}\tilde{\boldsymbol{W}}_{V,k}}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}} + \alpha_V \frac{\sigma_{1k}\left(\tilde{\sigma}_V^{\mathrm{T}}(\boldsymbol{x}_{\mathrm{N}},0)\boldsymbol{W}_V + \varepsilon(\boldsymbol{x}_k,k)\right)}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}}\right]^{\mathrm{T}} \times \\
&\quad \left[\tilde{\boldsymbol{W}}_{Vk} - \alpha_V \frac{\sigma_{1k}^{\mathrm{T}}\sigma_{1k}\tilde{\boldsymbol{W}}_{V,k}}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}} + \alpha_V \frac{\sigma_{1k}\left(\tilde{\sigma}_V^{\mathrm{T}}(\boldsymbol{x}_{\mathrm{N}},0)\boldsymbol{W}_V + \varepsilon(\boldsymbol{x}_k,k)\right)}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}}\right] - \tilde{\boldsymbol{W}}_{V,k}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{V,k}
\end{aligned}$$

$$\begin{aligned}
&= -2\alpha_V\left[\frac{\sigma_{1k}^{\mathrm{T}}\sigma_{1k}\tilde{\boldsymbol{W}}_{V,k}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{V,k}}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}} - \frac{\sigma_{1k}^{\mathrm{T}}\left(\tilde{\sigma}_V^{\mathrm{T}}(\boldsymbol{x}_{\mathrm{N}},0)\boldsymbol{W}_V + \varepsilon(\boldsymbol{x}_k,k)\right)\tilde{\boldsymbol{W}}_{V,k}}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}}\right] \\
&\quad + a_V^2\left\|\frac{\sigma_{1k}^{\mathrm{T}}\sigma_{1k}\tilde{\boldsymbol{W}}_{V,k}}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}} - \frac{\sigma_{1k}\left(\tilde{\sigma}_V^{\mathrm{T}}(\boldsymbol{x}_{\mathrm{N}},0)\boldsymbol{W}_V + \varepsilon(\boldsymbol{x}_k,k)\right)}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}}\right\|^2
\end{aligned} \tag{A.9}$$

Notice that $\dfrac{\sigma_{1k}^{\mathrm{T}}\sigma_{1k}}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}} < 1$ and applying Cauchy-Schwartz inequality yields

$$
\begin{aligned}
\Delta L_V(\tilde{\boldsymbol{W}}_{V,k}) &\le -2\alpha_V \frac{\sigma_{1k}^{\mathrm{T}}\sigma_{1k}\tilde{\boldsymbol{W}}_{V,k}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{V,k}}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}} + 2\alpha_V \frac{\sigma_{1k}^{\mathrm{T}}\left(\tilde{\sigma}_V^{\mathrm{T}}(\boldsymbol{x}_{\mathrm{N}},0)\boldsymbol{W}_V + \varepsilon(\boldsymbol{x}_k,k)\right)\tilde{\boldsymbol{W}}_{V,k}}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}} \\
&\quad + 2\alpha_V^2\left(\frac{\sigma_{1k}^{\mathrm{T}}\sigma_{1k}}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}}\right)^2 \left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 + \frac{2\alpha_V^2}{(1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k})^2}\left\|\sigma_{1k}\left(\tilde{\sigma}_V^{\mathrm{T}}(\boldsymbol{x}_{\mathrm{N}},0)\boldsymbol{W}_V + \varepsilon(\boldsymbol{x}_k,k)\right)\right\|^2 \\
&\le -2\alpha_V \frac{\sigma_{1k}^{\mathrm{T}}\sigma_{1k}}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}}\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 \\
&\quad + 2\alpha_V \frac{\sigma_{1k}^{\mathrm{T}}\left(\tilde{\sigma}_V^{\mathrm{T}}(\boldsymbol{x}_{\mathrm{N}},0)\boldsymbol{W}_V + \varepsilon(\boldsymbol{x}_k,k)\right)\tilde{\boldsymbol{W}}_{V,k}}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}} + 2\alpha_V^2\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 \\
&\quad + \frac{2\alpha_V^2}{(1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k})^2}\left\|\sigma_{1k}\left(\tilde{\sigma}_V^{\mathrm{T}}(\boldsymbol{x}_{\mathrm{N}},0)\boldsymbol{W}_V + \varepsilon(\boldsymbol{x}_k,k)\right)\right\|^2
\end{aligned}
\tag{A.10}
$$

Note that σ_{1k} is a time-dependent activation function and hence the Lyapunov candidate function becomes non-autonomous. Recall that the time span of interest is finite and σ_{1k} is a smooth function, then σ_{1k} is bounded by $0 < \sigma_{1\mathrm{m}} \le \left\|\sigma_{1k}\right\| \le \sigma_{1\mathrm{M}}$. Then separating the term $\dfrac{\sigma_{1k}^{\mathrm{T}}\left(\tilde{\sigma}_V^{\mathrm{T}}(\boldsymbol{x}_{\mathrm{N}},0)\boldsymbol{W}_V + \varepsilon(\boldsymbol{x}_k,k)\right)\tilde{\boldsymbol{W}}_{V,k}}{1+\sigma_{1k}^{\mathrm{T}}\sigma_{1k}}$ and recalling the bounds $\left\|\tilde{\sigma}_V(\boldsymbol{x}_{\mathrm{N}})\right\| \le 2\sigma_{V\mathrm{M}}$, $\left\|\boldsymbol{W}_V\right\| \le W_{\mathrm{M}}$ and $\left\|\varepsilon(\boldsymbol{x}_k,k)\right\| \le 3\varepsilon_{V\mathrm{M}}$, (A.10) becomes

$$
\begin{aligned}
\Delta L_V(\tilde{\boldsymbol{W}}_{V,k}) &\le -2\alpha_V \frac{\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2}\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 + \alpha_V^2\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 + \\
&\quad 8\alpha_V \frac{\sigma_{V\mathrm{M}}^2 W_{\mathrm{M}}^2}{1+\sigma_{1\mathrm{m}}^2} + 18\varepsilon_{V\mathrm{M}}^2 + 2\alpha_V^2\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 + 2\alpha_V^2(8\sigma_{V\mathrm{M}}^2 W_{\mathrm{M}}^2 + 18\varepsilon_{V\mathrm{M}}^2) \\
&\le -2\alpha_V \frac{\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2}\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 + \alpha_V^2\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 + 2\alpha_V^2\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 \\
&\quad + \alpha_V\left((1+2\alpha_V)8\sigma_{V\mathrm{M}}^2 W_{\mathrm{M}}^2 + 36\alpha_V\varepsilon_{V\mathrm{M}}^2\right) + 18\varepsilon_{V\mathrm{M}}^2 \\
&\le -\alpha_V\left(\frac{2\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2} - 3\alpha_V\right)\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 + \varepsilon_{1V\mathrm{M}}^2
\end{aligned}
\tag{A.11}
$$

where $\varepsilon_{1V\mathrm{M}}^2 = \alpha_V\left((1+2\alpha_V)8\sigma_{V\mathrm{M}}^2 W_\mathrm{M}^2 + 36\alpha_V\varepsilon_{V\mathrm{M}}^2\right) + 18\varepsilon_{V\mathrm{M}}^2$.

From (A.11), it can be seen that the non-autonomous Lyapunov candidate is upper bounded by a time-invariant function. Therefore, $\Delta L_V(\tilde{\boldsymbol{W}}_{V,k})$ is less than zero outside of a compact set as long as the following condition holds:

$$\left\|\tilde{\boldsymbol{W}}_{V,k}\right\| > \sqrt{\frac{\varepsilon_{1V\mathrm{M}}^2}{\alpha_V\left(\dfrac{2\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2} - 3\alpha_V\right)}} \equiv B_{\boldsymbol{W}_V} \tag{A.12}$$

Proof of Theorem 3:

First, denote $\hat{\boldsymbol{u}}_k = \hat{\boldsymbol{u}}(\boldsymbol{x}_k, k)$, $\boldsymbol{f}_k = f(\boldsymbol{x}_k)$, $\boldsymbol{g}_k = g(\boldsymbol{x}_k)$, $\hat{\boldsymbol{g}}_k = \hat{g}(\boldsymbol{x}_k)$, $\tilde{\boldsymbol{g}}_k = \tilde{g}(\boldsymbol{x}_k)$ for simplicity.

Define the Lyapunov candidate function as

$$L = \frac{\alpha_{\boldsymbol{u}}^2\Lambda}{2g_\mathrm{M}^2(1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)}L_{\boldsymbol{x}} + L_{\mathrm{ID}} + L_V + \Lambda L_{\boldsymbol{u}} + L_A + L_B \tag{A.13}$$

where $L_{\boldsymbol{x}} = \boldsymbol{x}_k^\mathrm{T}\boldsymbol{x}_k$, L_{ID}, L_V are defined in (A.1) and (A.7), respectively, $L_{\boldsymbol{u}} = \mathrm{tr}\{\tilde{\boldsymbol{W}}_{\boldsymbol{u},k}^\mathrm{T}\tilde{\boldsymbol{W}}_{\boldsymbol{u},k}\}$, $L_A = (\tilde{\boldsymbol{W}}_{V,k}^\mathrm{T}\tilde{\boldsymbol{W}}_{V,k})^2$ and $L_B = \Theta_\mathrm{m}^2\mathrm{tr}(\tilde{\boldsymbol{W}}_k^\mathrm{T}\tilde{\boldsymbol{W}}_k)^2 + \left\|\tilde{\boldsymbol{W}}_k^\mathrm{T}\sigma(\boldsymbol{x}_{k-1})\bar{\boldsymbol{u}}_{k-1}\right\|^4$.

Define

$$\Lambda = \min\left\{\frac{\alpha_V^2}{2\alpha_{\boldsymbol{u}}\Pi_1}, \frac{\Theta_\mathrm{m}^2}{4\alpha_{\boldsymbol{u}}\Pi_2\sigma_\mathrm{M}^2}, \frac{\alpha_V\left(\dfrac{\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2} - 3\alpha_V\right)}{\alpha_{\boldsymbol{u}}\Pi_3}, \frac{\Theta_\mathrm{m}^4}{2\alpha_{\boldsymbol{u}}\Pi_3\sigma_\mathrm{M}^4}\right\}$$

where

$$\Pi_1 = \max\left\{\frac{(g_{\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R)^2\left(8+9\alpha_{\boldsymbol{u}}\right)}{2}, \frac{3\alpha_{\boldsymbol{u}}^2(g_{\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R)^2}{2\left(3\alpha_{\boldsymbol{u}}+8\right)}\right\},$$

$$\Pi_2 = \max\left\{\frac{(W_{V\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R)^2\left(\alpha_{\boldsymbol{u}}+8\right)}{2}, \frac{3\alpha_{\boldsymbol{u}}^2(W_{V\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R)^2}{2\left(3\alpha_{\boldsymbol{u}}+8\right)}\right\},$$

and $\Pi_3 = \frac{3}{8}(\nabla\sigma_{V\mathrm{M}}R)^2\left(3\alpha_{\boldsymbol{u}}+8\right)$.

Next, the terms in (A.13) will be considered individually. First,

$$\begin{aligned}\Delta L_{\boldsymbol{x}} &= \boldsymbol{x}_{k+1}^{\mathrm{T}}\boldsymbol{x}_{k+1} - \boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{x}_k = \left\|\boldsymbol{f}_k + \boldsymbol{g}_k\hat{\boldsymbol{u}}_k\right\|^2 - \left\|\boldsymbol{x}_k\right\|^2 \\ &= \left\|\boldsymbol{f}_k + \boldsymbol{g}_k\boldsymbol{u}_k^* - \boldsymbol{g}_k\boldsymbol{u}_k^* + \boldsymbol{g}_k\hat{\boldsymbol{u}}_k\right\|^2 - \left\|\boldsymbol{x}_k\right\|^2\end{aligned} \tag{A.14}$$

By using Lemma 1, Cauchy-Schwartz inequality twice and recalling the bounds, (A.14) becomes

$$\begin{aligned}\Delta L_{\boldsymbol{x}} &\le 2\left\|\boldsymbol{f}_k + \boldsymbol{g}_k\boldsymbol{u}_k^*\right\|^2 + 2\left\|\boldsymbol{g}_k\boldsymbol{u}_k^* - \boldsymbol{g}_k\hat{\boldsymbol{u}}_k\right\|^2 - \left\|\boldsymbol{x}_k\right\|^2 \\ &\le -(1-2k^*)\left\|\boldsymbol{x}_k\right\|^2 + 2g_{\mathrm{M}}^2\left\|(\boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u},k} + \varepsilon_{\boldsymbol{u},k} - \hat{\boldsymbol{W}}_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k})\right\|^2 \\ &\le -(1-2k^*)\left\|\boldsymbol{x}_k\right\|^2 + 2g_{\mathrm{M}}^2\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k} + \varepsilon_{\boldsymbol{u},k}\right\|^2 \\ &\le -(1-2k^*)\left\|\boldsymbol{x}_k\right\|^2 + 4g_{\mathrm{M}}^2\left\|\boldsymbol{\varXi}_{\boldsymbol{u},k}\right\|^2 + 4g_{\mathrm{M}}^2\varepsilon_{\boldsymbol{u}\mathrm{M}}^2\end{aligned} \tag{A.15}$$

where $\boldsymbol{\varXi}_{\boldsymbol{u},k} = \tilde{\boldsymbol{W}}_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k}$.

Next, recalling (40) and using the bound, the first difference $\Delta L_{\boldsymbol{u}}$ can be represented as

$$\begin{aligned}\Delta L_{\boldsymbol{u}} &= \mathrm{tr}(\tilde{\boldsymbol{W}}_{\boldsymbol{u},k+1}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{\boldsymbol{u},k+1}) - \mathrm{tr}(\tilde{\boldsymbol{W}}_{\boldsymbol{u},k}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{\boldsymbol{u},k}) \\ &= \frac{2\alpha_{\boldsymbol{u}}}{1+\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k}}\mathrm{tr}\left(\tilde{\boldsymbol{u}}_k\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{\boldsymbol{u},k}\right) + \frac{\alpha_{\boldsymbol{u}}^2\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k}}{(1+\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k})^2}\mathrm{tr}(\tilde{\boldsymbol{u}}_k\tilde{\boldsymbol{u}}_k^{\mathrm{T}})\end{aligned} \tag{A.16}$$

Noticing that $\frac{\sigma_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}k}}{1+\sigma_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}k}} \le 1$, then substituting (39) into (A.16) and using cyclic property of trace operator and applying norm with upper bounds, (A.16) becomes, after collecting the terms, as

$$
\begin{aligned}
\Delta L_{\boldsymbol{u}} \le{} & -\frac{\alpha_{\boldsymbol{u}}(2-\alpha_{\boldsymbol{u}})}{1+\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k}}\left\|\boldsymbol{\Xi}_{\boldsymbol{u},k}\right\|^2 + \frac{\alpha_{\boldsymbol{u}}^2(W_{V\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R)^2}{4(1+\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k})}\left\|\tilde{\boldsymbol{g}}_k\right\|^2 + \frac{\alpha_{\boldsymbol{u}}^2(g_{\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R)^2}{4(1+\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k})}\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 \\
& + \frac{\alpha_{\boldsymbol{u}}^2(\nabla\sigma_{V\mathrm{M}}R)^2}{4(1+\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k})}\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2\left\|\tilde{\boldsymbol{g}}_k\right\|^2 + \frac{\alpha_{\boldsymbol{u}}^2}{(1+\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k})}\left\|\tilde{\varepsilon}_{\boldsymbol{u},k}\right\|^2 + \frac{\alpha_{\boldsymbol{u}}^2(W_{V\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R\tilde{\varepsilon}_{\boldsymbol{u}\mathrm{M}})}{(1+\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k})}\left\|\tilde{\boldsymbol{g}}_k\right\| \\
& + \frac{\alpha_{\boldsymbol{u}}(1+\alpha_{\boldsymbol{u}})}{1+\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k}}\left\|\boldsymbol{\Xi}_{\boldsymbol{u},k}\right\|\left(\begin{gathered}\frac{\nabla\sigma_{V\mathrm{M}}RW_{V\mathrm{M}}\left\|\tilde{\boldsymbol{g}}_k\right\| + \nabla\sigma_{V\mathrm{M}}Rg_{\mathrm{M}}\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|}{1+\alpha_{\boldsymbol{u}}} \\ + \nabla\sigma_{V\mathrm{M}}R\left\|\tilde{\boldsymbol{g}}_k\right\|\left\|\tilde{\boldsymbol{W}}_{V,k}\right\| + 2(1+\alpha_{\boldsymbol{u}})\tilde{\varepsilon}_{\boldsymbol{u}\mathrm{M}}\end{gathered}\right) \\
& + \frac{\alpha_{\boldsymbol{u}}^2(\nabla\sigma_{V\mathrm{M}}R)^2}{2(1+\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k})}\left(g_{\mathrm{M}}\left\|\tilde{\boldsymbol{g}}_k\right\|\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 + W_{V\mathrm{M}}\left\|\tilde{\boldsymbol{g}}_k\right\|^2\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|\right) \\
& + \frac{\alpha_{\boldsymbol{u}}^2\nabla\sigma_{V\mathrm{M}}R}{2(1+\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k})}\left(W_{V\mathrm{M}}g_{\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R + 2\tilde{\varepsilon}_{\boldsymbol{u}\mathrm{M}}\right)\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|\left\|\tilde{\boldsymbol{g}}_k\right\| \\
& + \frac{\alpha_{\boldsymbol{u}}^2}{(1+\sigma_{\boldsymbol{u},k}^{\mathrm{T}}\sigma_{\boldsymbol{u},k})}(g_{\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R\tilde{\varepsilon}_{\boldsymbol{u}\mathrm{M}})\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|
\end{aligned}
\tag{A.17}
$$

where $R = \left\|\boldsymbol{R}^{-1}\right\|$.

Next, observe that

$$
-\alpha_{\boldsymbol{u}}(2-\alpha_{\boldsymbol{u}})\left\|\boldsymbol{\Xi}_{\boldsymbol{u},k}\right\|^2 = \frac{-\alpha_{\boldsymbol{u}}(3-3\alpha_{\boldsymbol{u}})\left\|\boldsymbol{\Xi}_{\boldsymbol{u},k}\right\|^2}{2} - \frac{\alpha_{\boldsymbol{u}}(1+\alpha_{\boldsymbol{u}})\left\|\boldsymbol{\Xi}_{\boldsymbol{u},k}\right\|^2}{2}
$$

Recall that similar as the critic NN, $\sigma_{\boldsymbol{u},k}$ is a time-dependent activation function and bounded, due to the smoothness of $\sigma_{\boldsymbol{u},k}$ and finite time span, by $0 < \sigma_{\boldsymbol{u}\mathrm{m}} \le \left\|\sigma_{\boldsymbol{u},k}\right\| \le \sigma_{\boldsymbol{u}\mathrm{M}}$. Then (A.17) becomes, after completing the squares w.r.t. $\left\|\boldsymbol{\Xi}_{\boldsymbol{u},k}\right\|$, $\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|$, $\left\|\tilde{\boldsymbol{g}}_k\right\|$ and $\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|\left\|\tilde{\boldsymbol{g}}_k\right\|$, as

$$\begin{aligned}\Delta L_{\boldsymbol{u}} \le & -\frac{\alpha_{\boldsymbol{u}}(3-3\alpha_{\boldsymbol{u}})}{2(1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)}\left\|\boldsymbol{\Xi}_{\boldsymbol{u},k}\right\|^2 + \frac{\alpha_{\boldsymbol{u}}(g_{\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R)^2(8+9\alpha_{\boldsymbol{u}})}{2(1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)}\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 \\ & + \frac{\alpha_{\boldsymbol{u}}(W_{V\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R)^2(\alpha_{\boldsymbol{u}}+8)}{2(1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)}\left\|\tilde{\boldsymbol{g}}_k\right\|^2 + \frac{6\alpha_{\boldsymbol{u}}(\nabla\sigma_{V\mathrm{M}}R)^2\alpha_{\boldsymbol{u}}^2 g_{\mathrm{M}}^2}{4(1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)(3\alpha_{\boldsymbol{u}}+8)}\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 \\ & + \frac{3\alpha_{\boldsymbol{u}}(\nabla\sigma_{V\mathrm{M}}R)^2(3\alpha_{\boldsymbol{u}}+8)}{4(1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)}\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2\left\|\tilde{\boldsymbol{g}}_k\right\|^2 + \frac{6\alpha_{\boldsymbol{u}}(\nabla\sigma_{V\mathrm{M}}R)^2\alpha_{\boldsymbol{u}}^2 W_{V\mathrm{M}}^2}{4(1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)(3\alpha_{\boldsymbol{u}}+8)}\left\|\tilde{\boldsymbol{g}}_k\right\|^2 \\ & + \tilde{\varepsilon}_{1\mathrm{M}}^2 + \tilde{\varepsilon}_{2\mathrm{M}}^2 + \tilde{\varepsilon}_{3\mathrm{M}}^2 + \tilde{\varepsilon}_{4\mathrm{M}}^2\end{aligned}$$

where $\tilde{\varepsilon}_{1\mathrm{M}}^2 = \dfrac{8\alpha_{\boldsymbol{u}}(1+\alpha_{\boldsymbol{u}})^3}{(1+\sigma_{\boldsymbol{u}\mathrm{m}}^2)}\tilde{\varepsilon}_{\boldsymbol{u}\mathrm{M}}^2$,

$$\tilde{\varepsilon}_{2\mathrm{M}}^2 = \frac{2\alpha_{\boldsymbol{u}}(g_{\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R)^2(8+9\alpha_{\boldsymbol{u}})}{(1+\sigma_{\boldsymbol{u}\mathrm{m}}^2)}\left(\frac{\alpha_{\boldsymbol{u}}\tilde{\varepsilon}_{\boldsymbol{u}\mathrm{M}}}{g_{\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R(8+9\alpha_{\boldsymbol{u}})}\right)^2,$$

$$\tilde{\varepsilon}_{3\mathrm{M}}^2 = \frac{2\alpha_{\boldsymbol{u}}(W_{V\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R)^2}{(1+\sigma_{\boldsymbol{u}\mathrm{m}}^2)}(\alpha_{\boldsymbol{u}}+8)\left(\frac{\alpha_{\boldsymbol{u}}\tilde{\varepsilon}_{\boldsymbol{u}\mathrm{M}}}{W_{V\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R(\alpha_{\boldsymbol{u}}+8)}\right)^2,$$

$$\tilde{\varepsilon}_{4\mathrm{M}}^2 = \frac{3\alpha_{\boldsymbol{u}}(\nabla\sigma_{V\mathrm{M}}R)^2(3\alpha_{\boldsymbol{u}}+8)}{4(1+\sigma_{\boldsymbol{u}\mathrm{m}}^2)}\left(\frac{\alpha_{\boldsymbol{u}}\left(W_{V\mathrm{M}}g_{\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R+2\tilde{\varepsilon}_{\boldsymbol{u}\mathrm{M}}\right)}{(3\alpha_{\boldsymbol{u}}+8)\nabla\sigma_{V\mathrm{M}}R}\right)^2.$$

Finally, using Young's inequality and recalling from the definition of $\Pi_1 \sim \Pi_3$, we have

$$\begin{aligned}\Delta L_{\boldsymbol{u}} \le & -\frac{\alpha_{\boldsymbol{u}}(3-3\alpha_{\boldsymbol{u}})\left\|\boldsymbol{\Xi}_{\boldsymbol{u},k}\right\|^2}{2(1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)} + \frac{\alpha_{\boldsymbol{u}}}{2}(g_{\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R)^2(8+9\alpha_{\boldsymbol{u}})\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 \\ & + \frac{\alpha_{\boldsymbol{u}}}{2}(W_{V\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R)^2(\alpha_{\boldsymbol{u}}+8)\left\|\tilde{\boldsymbol{g}}_k\right\|^2 + \frac{3\alpha_{\boldsymbol{u}}}{8}(\nabla\sigma_{V\mathrm{M}}R)^2(3\alpha_{\boldsymbol{u}}+8)\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^4 \\ & + \frac{3\alpha_{\boldsymbol{u}}}{8}(\nabla\sigma_{V\mathrm{M}}R)^2(3\alpha_{\boldsymbol{u}}+8)\left\|\tilde{\boldsymbol{g}}_k\right\|^4 + \frac{3\alpha_{\boldsymbol{u}}^2(g_{\mathrm{M}}\nabla\sigma_{V\mathrm{M}}R)^2}{2(3\alpha_{\boldsymbol{u}}+8)}\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 \\ & + \frac{3\alpha_{\boldsymbol{u}}(\nabla\sigma_{V\mathrm{M}}R)^2}{2(3\alpha_{\boldsymbol{u}}+8)}\alpha_{\boldsymbol{u}}^2 W_{V\mathrm{M}}^2\left\|\tilde{\boldsymbol{g}}_k\right\|^2 + \tilde{\varepsilon}_{1\mathrm{M}}^2 + \tilde{\varepsilon}_{2\mathrm{M}}^2 + \tilde{\varepsilon}_{3\mathrm{M}}^2 + \tilde{\varepsilon}_{4\mathrm{M}}^2 \\ \le & -\frac{\alpha_{\boldsymbol{u}}(3-3\alpha_{\boldsymbol{u}})\left\|\boldsymbol{\Xi}_{\boldsymbol{u},k}\right\|^2}{2(1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)} + 2\alpha_{\boldsymbol{u}}\Pi_1\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 + 2\alpha_{\boldsymbol{u}}\Pi_2\left\|\tilde{\boldsymbol{g}}_k\right\|^2 \\ & + \alpha_{\boldsymbol{u}}\Pi_3\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^4 + \alpha_{\boldsymbol{u}}\Pi_3\left\|\tilde{\boldsymbol{g}}_k\right\|^4 + \tilde{\varepsilon}_{1\mathrm{M}}^2 + \tilde{\varepsilon}_{2\mathrm{M}}^2 + \tilde{\varepsilon}_{3\mathrm{M}}^2 + \tilde{\varepsilon}_{4\mathrm{M}}^2\end{aligned} \qquad \text{(A.18)}$$

Next, consider $L_A = (\tilde{\boldsymbol{W}}_{V,k}^{\mathrm{T}} \tilde{\boldsymbol{W}}_{V,k})^2$ using (A.11), we have

$$
\begin{aligned}
\Delta L_A &= (\tilde{\boldsymbol{W}}_{V,k+1}^{\mathrm{T}} \tilde{\boldsymbol{W}}_{V,k+1})^2 - (\tilde{\boldsymbol{W}}_{V,k}^{\mathrm{T}} \tilde{\boldsymbol{W}}_{V,k})^2 \\
&= (\tilde{\boldsymbol{W}}_{V,k+1}^{\mathrm{T}} \tilde{\boldsymbol{W}}_{V,k+1} + \tilde{\boldsymbol{W}}_{V,k}^{\mathrm{T}} \tilde{\boldsymbol{W}}_{V,k})(\tilde{\boldsymbol{W}}_{V,k+1}^{\mathrm{T}} \tilde{\boldsymbol{W}}_{V,k+1} - \tilde{\boldsymbol{W}}_{V,k}^{\mathrm{T}} \tilde{\boldsymbol{W}}_{V,k}) \\
&\le \left[\left[\left(2 - \alpha_V \left(\frac{2\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2} - 3\alpha_V\right)\right)\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 + \varepsilon_{1V\mathrm{M}}^2\right] \times \right. \\
&\quad \left[-\alpha_V \left(\frac{2\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2} - 3\alpha_V\right)\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 + \varepsilon_{1V\mathrm{M}}^2\right] \\
&= -\alpha_V \left(\frac{2\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2} - 3\alpha_V\right)\left(2 - \alpha_V\left(\frac{2\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2} - 3\alpha_V\right)\right)\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^4 \\
&\quad + 2\left(1 - \alpha_V\left(\frac{2\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2} - 3\alpha_V\right)\right)\varepsilon_{1V\mathrm{M}}^2 \left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^2 + \varepsilon_{1V\mathrm{M}}^4 \\
&= -\alpha_V \left(\frac{2\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2} - \alpha_V\right)\left(\frac{3}{2} - \alpha_V\left(\frac{2\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2} - \alpha_V\right)\right)\left\|\tilde{\boldsymbol{W}}_{V,k}\right\|^4 \\
&\quad + \left(1 + 2\Big/ \alpha_V\left(\frac{2\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2} - \alpha_V\right)\right)\varepsilon_{1V\mathrm{M}}^4
\end{aligned}
\tag{A.19}
$$

Next, consider $L_B = \left(\Theta_{\mathrm{m}}^2 \operatorname{tr}(\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \tilde{\boldsymbol{W}}_k)\right)^2 + \left\|\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1}\right\|^4$.

Recalling the NN weights estimation error dynamics (16) and applying Cauchy- Swartz inequality, we have

$$
\begin{aligned}
\Delta L_B &= \left(\Theta_{\mathrm{m}}^2 \operatorname{tr}(\tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \tilde{\boldsymbol{W}}_{k+1})\right)^2 - \left(\Theta_{\mathrm{m}}^2 \operatorname{tr}(\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \tilde{\boldsymbol{W}}_k)\right)^2 \\
&\quad + \left\|\tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \sigma(\boldsymbol{x}_k) \bar{\boldsymbol{u}}_k\right\|^4 - \left\|\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1}\right\|^4 \\
&\le -\Theta_{\mathrm{m}}^4 \left\|\tilde{\boldsymbol{W}}_k\right\|^4 + 2\left\|\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1}\right\|^4 - \left\|\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1}\right\|^4 + \Delta\bar{\varepsilon}_{\mathrm{M}}^2 \\
&\le -\Theta_{\mathrm{m}}^4 \left\|\tilde{\boldsymbol{W}}_k\right\|^4 + 16\alpha^4 \left\|\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1}\right\|^4 - \left\|\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1}\right\|^4 + \Delta\bar{\varepsilon}_{\mathrm{M}}^2 \\
&\le -\Theta_{\mathrm{m}}^4 \left\|\tilde{\boldsymbol{W}}_k\right\|^4 - (1 - 16\alpha^4)\left\|\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\boldsymbol{x}_{k-1}) \bar{\boldsymbol{u}}_{k-1}\right\|^4 + \Delta\bar{\varepsilon}_{\mathrm{M}}^2 \\
&\le -\Theta_{\mathrm{m}}^4 \left\|\tilde{\boldsymbol{W}}_k\right\|^4 + \Delta\bar{\varepsilon}_{\mathrm{M}}^2
\end{aligned}
\tag{A.20}
$$

Finally, combing all the above terms yields the first difference of the Lyapunov candidate function as

$$
\begin{aligned}
\Delta L &= \frac{\alpha_{\boldsymbol{u}}^2 \Lambda}{2 g_{\mathrm{M}}^2 (1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)} \Delta L_{\boldsymbol{x}} + \Delta L_{\mathrm{ID}} + \Delta L_V + \Lambda \Delta L_{\boldsymbol{u}} + \Delta L_A + \Delta L_B \\
&\le -\frac{\alpha_{\boldsymbol{u}}^2 (1-2k^*)\Lambda}{2 g_{\mathrm{M}}^2 (1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)} \|\boldsymbol{x}_k\|^2 - \frac{\alpha_{\boldsymbol{u}}(3-7\alpha_{\boldsymbol{u}})}{2(1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)} \|\boldsymbol{\Xi}_{\boldsymbol{u},k}\|^2 - (1-\alpha^2)\|\boldsymbol{e}_k\|^2 \\
&\quad -\alpha_V \left(\frac{\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2} - 4\alpha_V \right) \|\tilde{\boldsymbol{W}}_{V,k}\|^2 - \frac{1}{2}\Theta_{\mathrm{m}}^2 \|\tilde{\boldsymbol{W}}_k\|^2 \\
&\quad -\alpha_V \left(\frac{\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2} - 3\alpha_V \right) \left(1 - \alpha_V \left(\frac{2\sigma_{1\mathrm{m}}^2}{1+\sigma_{1\mathrm{M}}^2} - 3\alpha_V \right) \right) \|\tilde{\boldsymbol{W}}_{V,k}\|^4 \\
&\quad -\Theta_{\mathrm{m}}^4 \|\tilde{\boldsymbol{W}}_k\|^4 + \varepsilon_{\mathrm{total}}
\end{aligned}
\tag{A.21}
$$

where $\varepsilon_{\mathrm{total}} = \left(1 + 2/\Theta_{\mathrm{m}}^2\right)\Delta\bar{\varepsilon}_{\mathrm{M}}^2 + \left(1 + 2/\Theta_{\mathrm{m}}^2\right)\varepsilon_{1V\mathrm{M}}^4 + \Lambda(\tilde{\varepsilon}_{1\mathrm{M}}^2 + \tilde{\varepsilon}_{2\mathrm{M}}^2 + \tilde{\varepsilon}_{3\mathrm{M}}^2 + \tilde{\varepsilon}_{4\mathrm{M}}^2) + \Delta\bar{\varepsilon}_{\mathrm{M}} + \varepsilon_{1V\mathrm{M}}^2 +$ $2\alpha_{\boldsymbol{u}}^2 \varepsilon_{\boldsymbol{u}\mathrm{M}}^2 / (1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)$. Hence, the non-autonomous Lyapunov candidate is upper bounded by a time-invariant function. Therefore, ΔL is less than zero outside of a compact set as long as the following conditions hold:

$$
\|\boldsymbol{x}_k\| > \sqrt{\frac{2 g_{\mathrm{M}}^2 (1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)\varepsilon_{\mathrm{total}}}{\alpha_{\boldsymbol{u}}^2 (1-2k^*)\Lambda}} \equiv b_{\boldsymbol{x}}
\tag{A.22}
$$

Or

$$
\|\boldsymbol{\Xi}_{\boldsymbol{u},k}\| > \sqrt{\frac{2(1+\sigma_{\boldsymbol{u}\mathrm{M}}^2)\varepsilon_{\mathrm{total}}}{\alpha_{\boldsymbol{u}}(3-7\alpha_{\boldsymbol{u}})\Lambda}} \equiv b_{\boldsymbol{\Xi}}
\tag{A.23}
$$

Or

$$
\|\boldsymbol{e}_k\| > \sqrt{\frac{\varepsilon_{\mathrm{total}}}{1-\alpha^2}} \equiv b_{\boldsymbol{e}}
\tag{A.24}
$$

Or

$$
\|\tilde{\boldsymbol{W}}_k\| > \min\left\{ \sqrt{\frac{2\varepsilon_{\mathrm{total}}}{\Theta_{\mathrm{m}}^2}}, \sqrt[4]{\frac{\varepsilon_{\mathrm{total}}}{4\Theta_{\mathrm{m}}^4}} \right\} \equiv b_{\tilde{\boldsymbol{W}}}
\tag{A.25}
$$

Or

$$\left\|\tilde{\boldsymbol{W}}_{V,k}\right\| > \min\left\{\begin{array}{c}\sqrt{\dfrac{\varepsilon_{\text{total}}}{\alpha_V\left(\dfrac{\sigma_{1\text{m}}^2}{1+\sigma_{1\text{M}}^2}-4\alpha_V\right)}}, \\ \sqrt[4]{\dfrac{\varepsilon_{\text{total}}}{\alpha_V\left(\dfrac{\sigma_{1\text{m}}^2}{1+\sigma_{1\text{M}}^2}-3\alpha_V\right)\left(1-\alpha_V\left(\dfrac{2\sigma_{1\text{m}}^2}{1+\sigma_{1\text{M}}^2}-3\alpha_V\right)\right)}}\end{array}\right\} \equiv b_{\tilde{\boldsymbol{W}}_V} \tag{A.26}$$

Note that the range for $\alpha_{\boldsymbol{u}}$ and k^* will always guarantee $b_{\boldsymbol{x}} > 0$ and $b_{\boldsymbol{\Xi}} > 0$. The range for α will guarantee $b_{\boldsymbol{e}} > 0$ and $b_{\tilde{\boldsymbol{W}}} > 0$. The range for α_V will guarantee $b_{\tilde{\boldsymbol{W}}_V} > 0$ since $0 < \alpha_V < \sigma_{1\text{m}}^2 / 4(1+\sigma_{1\text{M}}^2) < \sigma_{1\text{m}}^2 / 3(1+\sigma_{1\text{M}}^2)$, which will guarantee that the second term shown in (A.26) is positive.

Eventually, the difference between the ideal optimal control and proposed near optimal control inputs is represented as

$$\begin{aligned}&\left\|\boldsymbol{u}^*(\boldsymbol{x}_k,k)-\hat{\boldsymbol{u}}(\boldsymbol{x}_k,k)\right\| \\ &= \left\|\boldsymbol{W}_{\boldsymbol{u}}^{\text{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,k)+\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k,k)-\hat{\boldsymbol{W}}_{\boldsymbol{u}k}^{\text{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,k)\right\| \\ &= \left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}^{\text{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,k)+\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k,k)\right\| \\ &\le b_{\boldsymbol{\Xi}}+\varepsilon_{\boldsymbol{u}\text{M}} \equiv \varepsilon_{us}\end{aligned} \tag{A.27}$$

where $b_{\boldsymbol{\Xi}}$ is given in (A.23).

IV. FIXED FINAL-TIME NEAR OPTIMAL REGULATION OF NONLINEAR DISCRETE-TIME SYSEMS IN AFFINE FORM USING OUTPUT FEEDBACK

Qiming Zhao, Hao Xu and S. Jagannathan

Abstract — In this paper, the output feedback based finite-horizon near optimal regulation of nonlinear affine discrete-time systems with unknown system dynamics is considered. First, a neural network (NN)-based Luenberger observer is proposed to reconstruct both the system states and the control coefficient matrix. In other words, the observer design relaxes the need for a separate identifier to construct the control coefficient matrix. Next, reinforcement learning methodology with actor-critic structure is utilized to approximate the time-varying solution, referred to as the value function, of the Hamilton-Jacobi-Bellman (HJB) equation by using a neural network (NN). To properly satisfy the terminal constraint, a new error term is defined and incorporated in the NN update law so that the terminal constraint error is also minimized over time. The NNs with constant weights and time-dependent activation function is employed to approximate the time-varying value function which subsequently is utilized to generate the finite horizon near optimal control policy due to NN reconstruction errors. The proposed scheme functions in a forward-in-time manner without offline training phase. Lyapunov analysis is used to investigate the stability of the overall closed-loop system. Simulation results are given to show the effectiveness and feasibility of the proposed method.

1. INTRODUCTION

Optimal control has been one of the key topic areas in control for over half a century due to both theoretical merit and a gamut of practical applications. Traditionally, for infinite-horizon optimal regulation of linear systems with quadratic cost function (LQR), a constant solution to the algebraic Riccati equation (ARE) can be found given the system dynamics [1][2] which is subsequently utilized to obtain the optimal policy. For general nonlinear systems, the optimal solution can be obtained by solving the Hamilton-Jacobi-Bellman (HJB) equation, which however, is not an easy task since the HJB equation normally does not have an analytical solution.

In the recent decades, with full state feedback, reinforcement learning methodology is widely used by many researchers to address the optimal control under the infinite-horizon scenario for both linear and nonlinear systems [5][6][7][8]. However, in many practical situations, the system state vector is difficult or expensive to measure. Several traditional nonlinear observers, such as high-gain or sliding mode observers, have been developed during the past few decades [3][4]. However, the above mentioned observer designs are applicable to systems which are expressed in a specific system structure such as Brunovisky-form, and require the system dynamics a priori.

The optimal regulation of nonlinear systems can be addressed either for infinite or finite fixed time scenario. The finite-horizon optimal regulation still remains unresolved due to the following reasons. First, the solution to the optimal control of finite-horizon nonlinear system becomes essentially time-varying thus complicating the analysis, in contrast with the infinite-horizon case, where the solution is time-independent. In addition, the terminal constraint is explicitly imposed in the cost function, whereas in the infinite-horizon case, the terminal constraint is normally ignored. Finally, addition of online approximators such as neural networks (NNs) to overcome the system dynamics and generating an approximate solution to the time dependent HJB equation in a forward-in-time manner while satisfying the terminal constraint as well as proving closed-loop stability with the NNs are quite involved.

The past literature [9][10][11][12] provided some insights into solving finite-horizon optimal regulation of nonlinear system. The developed techniques functioned either backward-in-time [9][10] or require offline training [11][12] with iteration-based scheme. However, backward-in-time solution hinders the real time implementation, while inadequate number of iterations will lead to instability [6]. Further, the state vector is needed in all these techniques [9][10][11][12]. Therefore, a finite-horizon optimal regulation scheme, which can be implemented in an online and forward-in-time manner with completely unknown system dynamics and without using both state measurements and value and policy iterations, is yet to be developed.

Motivated by the aforementioned deficiencies, in this paper, an extended Luenberger observer is first proposed to estimate the system states as well as the control coefficient matrix. The actor-critic architecture is utilized to generate the optimal control policy wherein the value function is approximated by using the critic NN and the optimal policy is generated by using the approximated value function and the control coefficient matrix.

To handle the time-varying nature of the solution to the HJB equation or value function, NNs with constant weights and time-varying activation functions are utilized. In addition, in contrast with [11] and [12], the control policy is updated once every sampling instant and hence value/policy iterations are not performed. An error term corresponding to the terminal constraint is defined and minimized overtime such that the terminal constraint can be properly satisfied. A novel update law for tuning the NN is developed such that the critic NN weights will be tuned not only by using Bellman error but also the terminal constraint errors. Finally, stability of our proposed design scheme is demonstrated by Lyapunov stability analysis.

Therefore, the main contribution of the paper includes the development of a novel approach to solve the finite-horizon output feedback based near optimal control of uncertain nonlinear discrete-time systems in affine form in an online and forward-in-time manner without utilizing value and/or policy iterations. A novel online observer is introduced for generating the state vector and control coefficient matrix while an explicit need for an identifier is relaxed. Tuning laws for all the NNs are also derived. Lyapunov stability is also demonstrated.

The rest of the paper is organized as follows. In Section 2, background and formulation of finite-horizon optimal control for affine nonlinear discrete-time systems are introduced. In Section 3 the main control design scheme along with the stability analysis is addressed. In Section 4, simulation results are given to verify the feasibility of our approach. Conclusive remarks are provided in Section 5.

2. BACKGROUND AND PROBLEM FORMULATION

Consider the following nonlinear system

$$\begin{aligned} \boldsymbol{x}_{k+1} &= f(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_k \\ \boldsymbol{y}_k &= \boldsymbol{C}\boldsymbol{x}_k \end{aligned} \tag{1}$$

where $\boldsymbol{x}_k \in \Omega_{\boldsymbol{x}} \subset \Re^n$, $\boldsymbol{u}_k \in \Omega_{\boldsymbol{u}} \subset \Re^m$ and $\boldsymbol{y}_k \in \Omega_{\boldsymbol{y}} \subset \Re^p$ are the system states, control inputs and system outputs, respectively, $f(\boldsymbol{x}_k) \in \Re^n$, $g(\boldsymbol{x}_k) \in \Re^{n\times m}$ are smooth unknown nonlinear dynamics, and $\boldsymbol{C} \in \Re^{p\times n}$ is the known output matrix. It is assumed that the control coefficient matrix $g(\boldsymbol{x}_k)$ is bounded above such that $0 < \|g(\boldsymbol{x}_k)\| < g_{\mathrm{M}}$, where g_{M} is a positive constant. Before proceeding, the following assumption is needed.

Assumption: The nonlinear system given in (1) is controllable and observable. Moreover, the system output $\boldsymbol{y}_k \in \Omega_y$ is measurable.

The objective of the optimal control design is to determine a feedback control policy that minimizes the following time-varying value or cost function given by

$$V(\boldsymbol{x}_k, k) = \phi(\boldsymbol{x}_{\mathrm{N}}) + \sum_{i=k}^{\mathrm{N}-1} r(\boldsymbol{x}_i, \boldsymbol{u}_i, i) \tag{2}$$

where $[k, \mathrm{N}]$ is the time interval of interest, $\phi(\boldsymbol{x}_{\mathrm{N}})$ is the terminal constraint that penalizes the terminal state $\boldsymbol{x}_{\mathrm{N}}$, $r(\boldsymbol{x}_k, \boldsymbol{u}_k, k)$ is the cost-to-go function at each time step k and takes the quadratic form as $r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) = Q(\boldsymbol{x}_k, k) + \boldsymbol{u}_k^{\mathrm{T}} \boldsymbol{R}_k \boldsymbol{u}_k$, where $Q(\boldsymbol{x}_k, k) \in \Re$ is greater than or equal to zero and $\boldsymbol{R}_k \in \Re^{m\times m}$ is a positive definite symmetric weighting matrix, respectively. By setting $k = \mathrm{N}$, the terminal constraint for the value function is given as

$$V(\boldsymbol{x}_{\mathrm{N}}, \mathrm{N}) = \phi(\boldsymbol{x}_{\mathrm{N}}) \tag{3}$$

Remark 1: Generally, the terminal constraint $\phi(\boldsymbol{x}_{\mathrm{N}})$ is a function of state at terminal stage N and not necessarily to be in quadratic form. In the case of standard LQR, $\phi(\boldsymbol{x}_{\mathrm{N}})$ takes the quadratic form as $\phi(\boldsymbol{x}_{\mathrm{N}}) = \boldsymbol{x}_{\mathrm{N}}^{\mathrm{T}} \boldsymbol{Q}_{\mathrm{N}} \boldsymbol{x}_{\mathrm{N}}$ and the optimal control policy can be obtained by solving the Riccati equation (RE) in a backward-in-time fashion from the terminal value $\boldsymbol{Q}_{\mathrm{N}}$.

It is also important to note that in the case of finite-horizon, the value function (2) becomes essentially time-varying, in contrast with the infinite-horizon case [6][7]. By Bellman's principle of optimality [1][2], the optimal cost from k onwards is equal to

$$V^*(\boldsymbol{x}_k, k) = \min_{\boldsymbol{u}_k} \left\{ r(\boldsymbol{x}_k, \boldsymbol{u}_k, k) + V^*(\boldsymbol{x}_{k+1}, k+1) \right\} \tag{4}$$

The optimal control policy $\boldsymbol{u}_k^*$ that minimizes the value function $V^*(\boldsymbol{x}_k, k)$ is obtained by using the stationarity condition $\partial V^*(\boldsymbol{x}_k, k) / \partial \boldsymbol{u}_k = 0$ and revealed to be

$$\boldsymbol{u}_k^* = -\frac{1}{2} \boldsymbol{R}^{-1} g^{\mathrm{T}}(\boldsymbol{x}_k) \frac{\partial V^*(\boldsymbol{x}_{k+1}, k+1)}{\partial \boldsymbol{x}_{k+1}} \tag{5}$$

From (5), it is clear that even when the full system state vector and dynamics are available, the optimal control cannot be obtained for the nonlinear discrete-time system due to the need for the future state vector $\boldsymbol{x}_{k+1}$. To avoid this drawback and relax the requirement for system dynamics, iteration-based schemes are normally utilized by using NNs with offline-training.

However, iteration-based schemes are not preferred for hardware implementation since the number of iterations to ensure the stability cannot be easily determined [6]. Moreover, the iterative approaches cannot be implemented when the dynamics of the system are completely unknown, since at least the control coefficient matrix $g(\boldsymbol{x}_k)$ is required to generate the control policy [7]. Finally, optimal policy needs to be found even when the states are unavailable. Therefore, in this work, a solution is found with unavailable system states and completely unknown system dynamics without utilizing the iterative approach, as given in the next section.

3. FINITE-HORIZON NEAR OPTIMAL REGULATOR DESIGN WITH OUTPUT FEEDBACK

In this section, the output feedback-based finite-horizon near optimal regulation scheme for nonlinear discrete-time systems in affine form with completely unknown system dynamics is addressed. First, due to the unavailability of the system states and uncertain system dynamics, an extended version of Luenberger observer is proposed to reconstruct both the system states and control coefficient matrix in an online manner. Thus the proposed observer design relaxes the need for an explicit identifier. Next, the reinforcement learning methodology is utilized to approximate the time-varying value function with actor-critic structure, while both NNs are represented by constant weights and time-varying activation functions. In addition, an error term corresponding to the terminal constraint is defined and minimized overtime so that the terminal constraint can be properly satisfied. The stability of the closed-loop system is demonstrated, by Lyapunov theory to show that the parameter estimation remains bounded as the system evolves.

3.1 NN-OBSERVER DESIGN

The system dynamics (1) can be reformulated as

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{A}\boldsymbol{x}_k + F(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_k \\ \boldsymbol{y}_k &= \boldsymbol{C}\boldsymbol{x}_k \end{aligned} \tag{6}$$

where $\boldsymbol{A}$ is a Hurwitz matrix such that $(\boldsymbol{A},\boldsymbol{C})$ is observable and $F(\boldsymbol{x}_k) = f(\boldsymbol{x}_k) - \boldsymbol{A}\boldsymbol{x}_k$.

A NN has been proven to be an effective method in the estimation and control of nonlinear systems due to its online learning capability [16]. According to the universal approximation property [19], the system states can be represented by using NN on a compact set Ω as

$$\begin{aligned}\boldsymbol{x}_{k+1} &= \boldsymbol{A}\boldsymbol{x}_k + F(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_k \\ &= \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{W}_F^{\mathrm{T}}\sigma_F(\boldsymbol{x}_k) + \boldsymbol{W}_g^{\mathrm{T}}\sigma_g(\boldsymbol{x}_k)\boldsymbol{u}_k + \boldsymbol{\varepsilon}_{Fk} + \boldsymbol{\varepsilon}_{gk}\boldsymbol{u}_k \\ &= \boldsymbol{A}\boldsymbol{x}_k + \begin{bmatrix}\boldsymbol{W}_F \\ \boldsymbol{W}_g\end{bmatrix}^{\mathrm{T}} \begin{bmatrix}\sigma_F(\boldsymbol{x}_k) & \boldsymbol{0} \\ \boldsymbol{0} & \sigma_g(\boldsymbol{x}_k)\end{bmatrix} \begin{bmatrix}1 \\ \boldsymbol{u}_k\end{bmatrix} + [\boldsymbol{\varepsilon}_{Fk} \quad \boldsymbol{\varepsilon}_{gk}] \begin{bmatrix}1 \\ \boldsymbol{u}_k\end{bmatrix} \\ &= \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{W}^{\mathrm{T}}\sigma(\boldsymbol{x}_k)\bar{\boldsymbol{u}}_k + \bar{\boldsymbol{\varepsilon}}_k\end{aligned} \tag{7}$$

where $\boldsymbol{W} = \begin{bmatrix}\boldsymbol{W}_F \\ \boldsymbol{W}_g\end{bmatrix} \in \Re^{L\times n}$, $\sigma(\boldsymbol{x}_k) = \begin{bmatrix}\sigma_F(\boldsymbol{x}_k) & \boldsymbol{0} \\ \boldsymbol{0} & \sigma_g(\boldsymbol{x}_k)\end{bmatrix} \in \Re^{L\times(1+m)}$, $\bar{\boldsymbol{u}}_k = \begin{bmatrix}1 \\ \boldsymbol{u}_k\end{bmatrix} \in \Re^{(1+m)}$ and $\bar{\boldsymbol{\varepsilon}}_k = [\boldsymbol{\varepsilon}_{Fk} \quad \boldsymbol{\varepsilon}_{gk}]\bar{\boldsymbol{u}}_k \in \Re^n$, with L being the number of hidden neurons. In addition, the target NN weights, activation function and reconstruction error are assumed to be upper bounded by $\|\boldsymbol{W}\| \le W_{\mathrm{M}}$, $\|\sigma(\boldsymbol{x}_k)\| \le \sigma_{\mathrm{M}}$ and $\|\bar{\boldsymbol{\varepsilon}}_k\| \le \bar{\varepsilon}_{\mathrm{M}}$, where W_{M} , σ_{M} and $\bar{\varepsilon}_{\mathrm{M}}$ are positive constants. Then, the system states $\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + F(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_k$ can be identified by updating the target NN weight matrix $\boldsymbol{W}$.

Since the true system states are unavailable for the controller, we propose the following extended Luenberger observer described by

$$\begin{aligned}\hat{\boldsymbol{x}}_{k+1} &= \boldsymbol{A}\hat{\boldsymbol{x}}_k + \hat{\boldsymbol{W}}_k^{\mathrm{T}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k + \boldsymbol{L}(\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k) \\ \hat{\boldsymbol{y}}_k &= \boldsymbol{C}\hat{\boldsymbol{x}}_k\end{aligned} \tag{8}$$

where $\hat{\boldsymbol{W}}_{k+1}$ is the estimated value of the target NN weights $\boldsymbol{W}$, $\hat{\boldsymbol{x}}_k$ is the reconstructed system state vector, $\hat{\boldsymbol{y}}_k$ is the estimated output vector and $\boldsymbol{L} \in \Re^{n\times p}$ is the observer gain selected by the designer, respectively.

Now express the state estimation error as

$$\begin{aligned}\tilde{\boldsymbol{x}}_{k+1} &= \boldsymbol{x}_{k+1} - \hat{\boldsymbol{x}}_{k+1} \\ &= \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{W}^{\mathrm{T}}\sigma(\boldsymbol{x}_k)\bar{\boldsymbol{u}}_k + \bar{\boldsymbol{\varepsilon}}_k - (\boldsymbol{A}\hat{\boldsymbol{x}}_k + \hat{\boldsymbol{W}}_{k+1}^{\mathrm{T}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k + \boldsymbol{L}(\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k)) \\ &= \boldsymbol{A}_c\tilde{\boldsymbol{x}}_k + \tilde{\boldsymbol{W}}_k^{\mathrm{T}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k + \boldsymbol{W}^{\mathrm{T}}\tilde{\sigma}(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k + \bar{\boldsymbol{\varepsilon}}_k \\ &= \boldsymbol{A}_c\tilde{\boldsymbol{x}}_k + \tilde{\boldsymbol{W}}_k^{\mathrm{T}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k + \bar{\boldsymbol{\varepsilon}}_{Ok}\end{aligned} \tag{9}$$

where $\boldsymbol{A}_c = \boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}$ is the closed-loop matrix, $\tilde{\boldsymbol{W}}_k = \boldsymbol{W} - \hat{\boldsymbol{W}}_k$ is the NN weights estimation error, $\tilde{\sigma}(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k) = \sigma(\boldsymbol{x}_k) - \sigma(\hat{\boldsymbol{x}}_k)$ and $\bar{\boldsymbol{\varepsilon}}_{Ok} = \boldsymbol{W}^{\mathrm{T}}\tilde{\sigma}(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k + \bar{\boldsymbol{\varepsilon}}_k$ are bounded terms due to the bounded values of ideal NN weights, activation functions and reconstruction errors.

Remark 2: It should be noted that the proposed observer (8) has two essential purposes. First, the observer presented in (8) generates the reconstructed system states for the controller design. Second, the structure of the observer is novel in that it also generates the control coefficient matrix $g(\boldsymbol{x}_k)$, which will be viewed as a NN-based identifier. Thus, the NN-based observer (8) can be viewed both as a standard observer and an identifier whose estimate of the control coefficient matrix $g(\boldsymbol{x}_k)$, is utilized in the near optimal control design shown in the next section.

Now select the tuning law for the NN weights as

$$\hat{\boldsymbol{W}}_{k+1} = (1 - \alpha_{\mathrm{I}})\hat{\boldsymbol{W}}_k + \beta_{\mathrm{I}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\tilde{\boldsymbol{y}}_{k+1}^{\mathrm{T}}\boldsymbol{l}^{\mathrm{T}} \tag{10}$$

where α_{I} , β_{I} are the tuning parameters, $\tilde{\boldsymbol{y}}_{k+1} = \boldsymbol{y}_{k+1} - \hat{\boldsymbol{y}}_{k+1}$ is the output error and $\boldsymbol{l} \in \Re^{n \times p}$ is selected to match the dimension.

Hence, the NN weight estimation error dynamics, by recalling from (9), are revealed to be

$$\begin{aligned}\tilde{\boldsymbol{W}}_{k+1} &= \boldsymbol{W} - \hat{\boldsymbol{W}}_{k+1} \\ &= (1-\alpha_{\mathrm{I}})\tilde{\boldsymbol{W}}_k + \alpha_{\mathrm{I}}\boldsymbol{W} - \beta_{\mathrm{I}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\tilde{\boldsymbol{y}}_{k+1}^{\mathrm{T}}\boldsymbol{l}^{\mathrm{T}} \\ &= (1-\alpha_{\mathrm{I}})\tilde{\boldsymbol{W}}_k + \alpha_{\mathrm{I}}\boldsymbol{W} - \beta_{\mathrm{I}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\tilde{\boldsymbol{x}}_k^{\mathrm{T}}\boldsymbol{A}_c^{\mathrm{T}}\boldsymbol{C}^{\mathrm{T}}\boldsymbol{l}^{\mathrm{T}} \\ &\quad - \beta_{\mathrm{I}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\bar{\boldsymbol{u}}_k^{\mathrm{T}}\sigma^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)\tilde{\boldsymbol{W}}_k\boldsymbol{C}^{\mathrm{T}}\boldsymbol{l}^{\mathrm{T}} - \beta_{\mathrm{I}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\bar{\boldsymbol{\varepsilon}}_{Ok}^{\mathrm{T}}\boldsymbol{C}^{\mathrm{T}}\boldsymbol{l}^{\mathrm{T}}\end{aligned} \tag{11}$$

Next, the boundedness of the NN weights estimation error $\tilde{\boldsymbol{W}}_k$ will be demonstrated in Theorem 1. Before proceeding, the following definition is required.

Definition 1 [19]: An equilibrium point $\boldsymbol{x}_e$ is said to be *uniformly ultimately bounded* (UUB) if there exists a compact set $\Omega_{\boldsymbol{x}} \subset \Re^n$ so that for all initial state $\boldsymbol{x}_0 \in \Omega_{\boldsymbol{x}}$, there exists a bound B and a time $T(B, \boldsymbol{x}_0)$ such that $\|\boldsymbol{x}_k - \boldsymbol{x}_e\| \le B$ for all $k \ge k_0 + T$.

Theorem 1 (*Boundedness of the observer error*): Let the nonlinear system (1) be controllable and observable while the system output, $\boldsymbol{y}_k \in \Omega_{\boldsymbol{y}}$, be measurable. Let the initial NN observer weights $\hat{\boldsymbol{W}}_k$ are selected within compact set Ω_{OB} which contains the ideal weights $\boldsymbol{W}$. Given an initial admissible control input $\boldsymbol{u}_0 \in \Omega_{\boldsymbol{u}}$ and let the proposed observer be given as in (8) and the update law for tuning the NN weights be given by (10). Let the control signal be persistently exciting (PE). Then, there exist positive constants α_{I} and β_{I} satisfying $\frac{2-\sqrt{2}}{2} < \alpha_{\mathrm{I}} < 1$ and $0 < \beta_{\mathrm{I}} < \frac{2(1-\alpha_{\mathrm{I}})\lambda_{\min}(\boldsymbol{lC})}{\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2 + 1}$, with $\lambda_{\min}$ denoting the minimum eigenvalue, such that the observer error $\tilde{x}_k$ and the NN weights estimation errors $\tilde{\boldsymbol{W}}_k$ are all UUB, with the bounds given by (A.6) and (A.7).

Proof: See Appendix.

3.2 REINFORCEMENT LEARNING BASED NEAR OPTIMAL CONTROLLER DESIGN

In this subsection, we present the finite-horizon near optimal regulator which requires neither the system states nor the system dynamics. The reason we consider this design being near optimal rather than optimal is due to the observer NN reconstruction errors. Based on the observer design proposed in Section 3.1, the feedback signal for the controller only requires the reconstructed state vector $\hat{\boldsymbol{x}}_k$ generated by the observer and the control coefficient matrix. To overcome the drawback of dependency on the future value of system states (5) as stated in Section 2, reinforcement learning-based methodology with an actor-critic structure is adopted to approximate the value function and control inputs individually.

The value function is obtained approximately by using the temporal difference error while the optimal control policy is generated by minimizing this value function. The time-varying nature of the value function and control inputs are handled by utilizing NNs with constant weights and time-varying activation functions. In addition, the terminal constraint in the cost function can be properly satisfied by defining and minimizing a new error term corresponding to the terminal constraint $\phi(\boldsymbol{x}_{\mathrm{N}})$ overtime. As a result, the proposed algorithm performs in an online and forward-in-time manner which enjoys great practical benefits.

According to the universal approximation property of NNs [19] and actor-critic methodology, the value function and control inputs can be represented by a “critic” NN and an “actor” NN, respectively, as

$$V(\boldsymbol{x}_k,k)=\boldsymbol{W}_V^{\mathrm{T}}\sigma_V(\boldsymbol{x}_k,k)+\varepsilon_V(\boldsymbol{x}_k,k) \qquad (12)$$

and

$$\boldsymbol{u}(\boldsymbol{x}_k, k) = \boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}} \sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, k) + \varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k, k) \tag{13}$$

where $\boldsymbol{W}_V \in \Re^{L_V}$ and $\boldsymbol{W}_{\boldsymbol{u}} \in \Re^{L_{\boldsymbol{u}} \times m}$ are the constant target NN weights, with L_V and $L_{\boldsymbol{u}}$ the number of hidden neurons, $\sigma_V(\boldsymbol{x}_k, k) \in \Re^{L_V}$ and $\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, k) \in \Re^{L_{\boldsymbol{u}}}$ are the *time-varying* activation functions, $\varepsilon_V(\boldsymbol{x}_k, k)$ and $\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k, k)$ are the NN reconstruction errors for the critic and action network, respectively. Under standard assumption, the target NN weights are considered bounded above such that $\|\boldsymbol{W}_V\| \le W_{V\mathrm{M}}$ and $\|\boldsymbol{W}_{\boldsymbol{u}}\| \le W_{\boldsymbol{u}\mathrm{M}}$, respectively, where both $W_{V\mathrm{M}}$ and $W_{\boldsymbol{u}\mathrm{M}}$ are positive constants [19].

The NN activation functions and the reconstruction errors are also assumed to be bounded above such that $\|\sigma_V(\boldsymbol{x}_k, k)\| \le \sigma_{V\mathrm{M}}$, $\|\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, k)\| \le \sigma_{\boldsymbol{u}\mathrm{M}}$, $|\varepsilon_V(\boldsymbol{x}_k, k)| \le \varepsilon_{V\mathrm{M}}$ and $|\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k, k)| \le \varepsilon_{\boldsymbol{u}\mathrm{M}}$, with $\sigma_{V\mathrm{M}}$, $\sigma_{\boldsymbol{u}\mathrm{M}}$, $\varepsilon_{V\mathrm{M}}$ and $\varepsilon_{\boldsymbol{u}\mathrm{M}}$ all positive constants [19]. In addition, in this work, the gradient of the reconstruction error is also assumed to be bounded above such as $\|\partial \varepsilon_{V,k} / \partial \boldsymbol{x}_{k+1}\| \le \varepsilon'_{V\mathrm{M}}$, with $\varepsilon'_{V\mathrm{M}}$ a positive constant [7][15]. The terminal constraint of the value function is defined, similar to (17), as

$$V(\boldsymbol{x}_{\mathrm{N}}, \mathrm{N}) = \boldsymbol{W}_V^{\mathrm{T}} \sigma_V(\boldsymbol{x}_{\mathrm{N}}, \mathrm{N}) + \varepsilon_V(\boldsymbol{x}_{\mathrm{N}}, \mathrm{N}) \tag{14}$$

with $\sigma_V(\boldsymbol{x}_{\mathrm{N}}, \mathrm{N})$ and $\varepsilon_V(\boldsymbol{x}_{\mathrm{N}}, \mathrm{N})$ represent the activation and construction error corresponding to the terminal state $\boldsymbol{x}_{\mathrm{N}}$.

Remark 3: The fundamental difference between this work and [11] is that our proposed scheme yields a completely forward-in-time and online solution without using value/policy iteration and offline training, whereas the scheme proposed in [11] is essentially an iteration-based DHDP scheme and NN weights are trained offline. In addition, state availability is relaxed in this work.

3.2.1 Value Function Approximation. According to (17), the time-varying value function $V(\boldsymbol{x}_k,k)$ can be approximated by using a NN as

$$\hat{V}(\hat{\boldsymbol{x}}_k,k)=\hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_k,k) \tag{15}$$

where $\hat{V}(\hat{\boldsymbol{x}}_k,k)$ represents the approximated value function at time step k. $\hat{\boldsymbol{W}}_{Vk}$ and $\sigma_V(\hat{\boldsymbol{x}}_k,k)$ are the estimated critic NN weights and "reconstructed" activation function with the estimated states vector $\hat{\boldsymbol{x}}_k$ as the inputs.

The value function at the terminal stage can be represented by

$$\hat{V}(\boldsymbol{x}_{\mathrm{N}},\mathrm{N})=\hat{\boldsymbol{W}}_{V,k}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N}) \tag{16}$$

where $\hat{\boldsymbol{x}}_{\mathrm{N}}$ is an estimation of the terminal state. It should be noted that since the true value of $\boldsymbol{x}_{\mathrm{N}}$ is not known, $\hat{\boldsymbol{x}}_{\mathrm{N}}$ can be considered to be an "estimate" of $\boldsymbol{x}_{\mathrm{N}}$ and can be chosen randomly as long as $\hat{\boldsymbol{x}}_{\mathrm{N}}$ lies within a region for a stabilizing control policy [8][11].

To ensure optimality, the Bellman equation should hold along the system trajectory. According to the principle of optimality, the true Bellman equation is given by

$$Q(\boldsymbol{x}_k,k)+(\boldsymbol{u}_k^*)^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_k^*+V^*(\boldsymbol{x}_{k+1},k+1)-V^*(\boldsymbol{x}_k,k)=0 \tag{17}$$

However, (22) no longer holds when the reconstructed system state vector $\hat{x}_k$ and NN approximation are considered. Therefore, with estimated values, the Bellman equation (22) becomes

$$\begin{aligned} e_{\mathrm{BO},k} &= Q(\hat{\boldsymbol{x}}_k,k)+\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_k+\hat{V}(\hat{\boldsymbol{x}}_{k+1},k+1)-\hat{V}(\hat{\boldsymbol{x}}_k,k) \\ &= Q(\hat{\boldsymbol{x}}_k,k)+\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_k+\hat{\boldsymbol{W}}_{V,k}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{k+1},k+1)-\hat{\boldsymbol{W}}_{V,k}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_k,k) \\ &= Q(\hat{\boldsymbol{x}}_k,k)+\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_k-\hat{\boldsymbol{W}}_{V,k}^{\mathrm{T}}\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k) \end{aligned} \tag{18}$$

where $e_{\mathrm{BO},k}$ is the Bellman equation residual error *along the system trajectory*, and $\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k)=\sigma_V(\hat{\boldsymbol{x}}_k,k)-\sigma_V(\hat{\boldsymbol{x}}_{k+1},k+1)$.

Next, using (21), define an additional error term corresponding to the terminal constraint as

$$e_{\mathrm{N},k}=\psi(\boldsymbol{x}_{\mathrm{N}})-\hat{\boldsymbol{W}}_{V,k}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N}) \tag{19}$$

The objective of the optimal control design is thus to minimize the Bellman equation residual error $e_{\mathrm{BO},k}$ as well as the terminal constraint error $e_{\mathrm{N},k}$, so that the optimality can be achieved and the terminal constraint can be properly satisfied. Next, based on gradient descent approach, the update law for critic NN can be defined as

$$\hat{\boldsymbol{W}}_{Vk+1}=\hat{\boldsymbol{W}}_{Vk}+\alpha_V\frac{\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k)e_{\mathrm{BO},k}}{1+\Delta\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_k,k)\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k)}+\alpha_V\frac{\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})e_{\mathrm{N},k}}{1+\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})} \tag{20}$$

where α_V is a design parameter.

Now define $\tilde{\boldsymbol{W}}_{Vk}=\boldsymbol{W}_V-\hat{\boldsymbol{W}}_{Vk}$. The standard Bellman equation (22) can be expressed by NN representation as

$$0=Q(\boldsymbol{x}_k,k)+(\boldsymbol{u}_k^*)^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_k^*-\boldsymbol{W}_V^{\mathrm{T}}\Delta\sigma_V(\boldsymbol{x}_k,k)-\Delta\varepsilon_V(\boldsymbol{x}_k,k) \tag{21}$$

where $\Delta\sigma_V(\boldsymbol{x}_k,k)=\sigma_V(\boldsymbol{x}_k,k)-\sigma_V(\boldsymbol{x}_{k+1},k+1)$ and $\Delta\varepsilon_V(\boldsymbol{x}_k,k)=\varepsilon_V(\boldsymbol{x}_k,k)-\varepsilon_V(\boldsymbol{x}_{k+1},k+1)$.

Subtracting (23) from (21), $e_{\mathrm{BO},k}$ can be further derived as

$$\begin{aligned} e_{\mathrm{BO},k} &= Q(\hat{\boldsymbol{x}}_k,k)+\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_k-\hat{\boldsymbol{W}}_{V,k}^{\mathrm{T}}\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k) \\ &\quad -Q(\boldsymbol{x}_k,k)-(\boldsymbol{u}_k^*)^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_k^*+\boldsymbol{W}_V^{\mathrm{T}}\Delta\sigma_V(\boldsymbol{x}_k,k)+\Delta\varepsilon_V(\boldsymbol{x}_k,k) \end{aligned}$$

$$
\begin{aligned}
&= Q(\hat{\boldsymbol{x}}_k,k)+\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_k - Q(\boldsymbol{x}_k,k)-(\boldsymbol{u}_k^*)^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_k^* - \hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k)+\boldsymbol{W}_V^{\mathrm{T}}\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k)\\
&\quad -\boldsymbol{W}_V^{\mathrm{T}}\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k)+\boldsymbol{W}_V^{\mathrm{T}}\Delta\sigma_V(\boldsymbol{x}_k,k)+\Delta\varepsilon_V(\boldsymbol{x}_k,k)\\
&\le Q(\hat{\boldsymbol{x}}_k,k)+\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_k - Q(\boldsymbol{x}_k,k)-(\boldsymbol{u}_k^*)^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_k^* + \tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k)\\
&\quad +\boldsymbol{W}_V^{\mathrm{T}}\Delta\tilde{\sigma}_V(\boldsymbol{x}_k,\hat{\boldsymbol{x}}_k,k)+\Delta\varepsilon_V(\boldsymbol{x}_k,k)\\
&\le L_m\left\|\tilde{\boldsymbol{x}}_k\right\|^2+\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k)+\Delta\varepsilon_{VB}(\boldsymbol{x}_k,k)
\end{aligned}
\tag{22}
$$

where L_m is a positive Lipschitz constant for $Q(\hat{\boldsymbol{x}}_k,k)+\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_k-Q(\boldsymbol{x}_k,k)-(\boldsymbol{u}_k^*)^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_k^*$ due to the quadratic form in both system states and control inputs. In addition, $\Delta\tilde{\sigma}_V(\boldsymbol{x}_k,\hat{\boldsymbol{x}}_k,k)=\Delta\sigma_V(\boldsymbol{x}_k,k)-\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k)$ and $\Delta\varepsilon_{VB}(\boldsymbol{x}_k,k)=\boldsymbol{W}_V^{\mathrm{T}}\Delta\tilde{\sigma}_V(\boldsymbol{x}_k,\hat{\boldsymbol{x}}_k,k)+\Delta\varepsilon_V(\boldsymbol{x}_k,k)$ are all bounded terms due to the boundedness of ideal NN weights, activation functions and reconstruction errors.

Recalling from (14), the terminal constraint error $e_{\mathrm{N},k}$ can be further expressed as

$$
\begin{aligned}
e_{\mathrm{N},k} &= \psi(\boldsymbol{x}_{\mathrm{N}})-\hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})\\
&= \boldsymbol{W}_V^{\mathrm{T}}\sigma_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N})+\varepsilon_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N})-\hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})\\
&= \boldsymbol{W}_V^{\mathrm{T}}\sigma_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N})-\boldsymbol{W}_V^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})+\varepsilon_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N})\\
&\quad +\boldsymbol{W}_V^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})-\hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})\\
&= \tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})+\boldsymbol{W}_V^{\mathrm{T}}\tilde{\sigma}_V(\boldsymbol{x}_{\mathrm{N}},\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})+\varepsilon_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N})\\
&= \tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})+\varepsilon_{V\mathrm{N}}
\end{aligned}
\tag{23}
$$

where $\tilde{\sigma}_V(\boldsymbol{x}_{\mathrm{N}},\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})=\sigma_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N})-\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})$ and $\varepsilon_{V\mathrm{N}}=\boldsymbol{W}_V^{\mathrm{T}}\tilde{\sigma}_V(\boldsymbol{x}_{\mathrm{N}},\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})+\varepsilon_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N})$ are bounded due to bounded ideal NN weights, activation function and reconstruction errors.

Finally, the error dynamics for critic NN weights are revealed to be

$$
\tilde{\boldsymbol{W}}_{Vk+1}=\tilde{\boldsymbol{W}}_{Vk}-\alpha_V\frac{\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k)e_{\mathrm{BO},k}}{1+\Delta\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_k,k)\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k)}-\alpha_V\frac{\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})e_{\mathrm{N},k}}{1+\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})}
\tag{24}
$$

Next, the boundedness of the critic NN weights will be demonstrated, as shown in the following theorem. Before proceeding, the following definition is needed.

Definition 2 [8]: Let $\Omega_{\boldsymbol{u}}$ denote the set of admissible control. A control function $\boldsymbol{u}:\Re^{n}\to\Re^{m}$ is defined to be admissible if the following is true:

$\boldsymbol{u}$ is continuous on $\Omega_{\boldsymbol{u}}$;

$\boldsymbol{u}(\boldsymbol{x})\big|_{\boldsymbol{x}=0}=0$;

$\boldsymbol{u}(\boldsymbol{x})$ stabilize the system (1) on $\Omega_{\boldsymbol{x}}$;

$J(\boldsymbol{x}(0),\boldsymbol{u})<\infty,\forall\boldsymbol{x}(0)\in\Omega_{\boldsymbol{x}}$.

Since the design scheme is similar to policy iteration, we need to solve a fixed-point equation rather than recursive equation. The initial admissible control guarantees the solution of the fixed-potion equation exists, thus the approximation process can be effectively done by our proposed scheme.

Theorem 2 (*Boundedness of the critic NN weights*): Let the nonlinear system (1) be controllable and observable while the system output, $\boldsymbol{y}_k\in\Omega_{\boldsymbol{y}}$, be measurable. Let the initial critic NN weights $\hat{\boldsymbol{W}}_{Vk}$ are selected within compact set Ω_V which contains the ideal weights $\boldsymbol{W}_V$. Let $\boldsymbol{u}(0)\in\Omega_{\boldsymbol{u}}$ be an initial admissible control input for the system (1). Let the value function be approximated by a critic NN and the tuning law be given by (26). Then, under the assumptions stated in this paper, there exists a positive constant α_V satisfying $0<\alpha_V<\frac{1}{6}$ such that the critic NN weights estimation error $\tilde{\boldsymbol{W}}_{Vk}$ is UUB with a computable bound $b_{\tilde{W}_V}$ given in (A.16).

Proof: See Appendix.

3.2.2 Control Input Approximation. In this subsection, the near optimal control policy is obtained such that the estimated value function (15) is minimized. Recalling (18), the estimation of the control inputs by using NN can be represented as

$$\hat{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k) = \hat{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}} \sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k) \tag{25}$$

where $\hat{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)$ represents the approximated control input vector at time step k, $\hat{\boldsymbol{W}}_{\boldsymbol{u}k}$ and $\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)$ are the estimated values of the actor NN weights and "reconstructed" activation function with the estimated state vector $\hat{\boldsymbol{x}}_k$ as the input.

Define the control input error as

$$\boldsymbol{e}_{\boldsymbol{u}k} = \hat{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k) - \hat{\boldsymbol{u}}_1(\hat{\boldsymbol{x}}_k, k) \tag{26}$$

where $\hat{\boldsymbol{u}}_1(\hat{\boldsymbol{x}}_k, k) = -\frac{1}{2}\boldsymbol{R}^{-1}\hat{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)\nabla\hat{V}(\hat{\boldsymbol{x}}_{k+1}, k+1)$ is the control policy that minimizes the approximated value function $\hat{V}(\hat{\boldsymbol{x}}_k, k)$, ∇ denotes the gradient of the estimated value function with respect to the system states, $\hat{g}(\hat{\boldsymbol{x}}_k)$ is the approximated control coefficient matrix generated by the NN-based observer and $\hat{V}(\hat{\boldsymbol{x}}_{k+1}, k+1)$ is the approximated value function from the critic network.

Therefore, the control error (26) becomes

$$\begin{aligned} \boldsymbol{e}_{\boldsymbol{u}k} &= \hat{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k) - \hat{\boldsymbol{u}}_1(\hat{\boldsymbol{x}}_k, k) \\ &= \hat{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}} \sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k) + \frac{1}{2}\boldsymbol{R}^{-1}\hat{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1}, k+1)\hat{\boldsymbol{W}}_{Vk} \end{aligned} \tag{27}$$

The actor NN weights tuning law is then defined as

$$\hat{\boldsymbol{W}}_{\boldsymbol{u}k+1} = \hat{\boldsymbol{W}}_{\boldsymbol{u}k} - \alpha_{\boldsymbol{u}} \frac{\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)\boldsymbol{e}_{\boldsymbol{u}k}^{\mathrm{T}}}{1 + \sigma_{\boldsymbol{u}}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k, k)\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)} \tag{28}$$

where $\alpha_{\boldsymbol{u}} > 0$ is a design parameter.

To find the error dynamics for the actor NN weights, first observe that

$$\begin{aligned}\boldsymbol{u}(\boldsymbol{x}_k,k) &= \boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,k)+\boldsymbol{\varepsilon}_{\boldsymbol{u}}(\boldsymbol{x}_k,k)\\ &= -\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)(\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},k+1)\boldsymbol{W}_V+\nabla\boldsymbol{\varepsilon}_V(\boldsymbol{x}_{k+1},k+1))\end{aligned} \tag{29}$$

Or equivalently,

$$\begin{aligned}0 &= \boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,k)+\boldsymbol{\varepsilon}_{\boldsymbol{u}}(\boldsymbol{x}_k,k)+\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},k+1)\boldsymbol{W}_V\\ &\quad+\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\varepsilon_V(\boldsymbol{x}_{k+1},k+1)\end{aligned} \tag{30}$$

Subtracting (30) from (27), we have

$$\begin{aligned}\boldsymbol{e}_{\boldsymbol{u}k} &= \hat{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)+\frac{1}{2}\boldsymbol{R}^{-1}\hat{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1},k+1)\hat{\boldsymbol{W}}_{Vk}-\boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,k)-\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}}(\boldsymbol{x}_k,k)\\ &\quad-\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},k+1)\boldsymbol{W}_V-\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\varepsilon_V(\boldsymbol{x}_{k+1},k+1)\\ &= -\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)-\boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\tilde{\sigma}_{\boldsymbol{u}}(\boldsymbol{x}_k,\hat{\boldsymbol{x}}_k,k)-\boldsymbol{\varepsilon}_{\boldsymbol{u}}(\boldsymbol{x}_k,k)\\ &\quad+\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1},k+1)\hat{\boldsymbol{W}}_{Vk}+\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\tilde{\sigma}_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},\hat{\boldsymbol{x}}_{k+1},k+1)\hat{\boldsymbol{W}}_{Vk}\\ &\quad+\frac{1}{2}\boldsymbol{R}^{-1}(\hat{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)-g^{\mathrm{T}}(\boldsymbol{x}_k))\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1},k+1)\hat{\boldsymbol{W}}_{Vk}\\ &\quad-\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},k+1)\boldsymbol{W}_V-\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\varepsilon_V(\boldsymbol{x}_{k+1},k+1)\end{aligned} \tag{31}$$

where $\tilde{\sigma}_{\boldsymbol{u}}(\boldsymbol{x}_k,\hat{\boldsymbol{x}}_k,k)=\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,k)-\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)$ and $\nabla\tilde{\sigma}_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},\hat{\boldsymbol{x}}_{k+1},k+1)=\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},k+1)-\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1},k+1)$.

For simplicity, denote $\tilde{\sigma}_{\boldsymbol{u}k}=\tilde{\sigma}_{\boldsymbol{u}}(\boldsymbol{x}_k,\hat{\boldsymbol{x}}_k,k)$, $\nabla\hat{\sigma}_{Vk+1}^{\mathrm{T}}=\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1},k+1)$, $\nabla\tilde{\sigma}_{Vk+1}^{\mathrm{T}}=\nabla\tilde{\sigma}_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},\hat{\boldsymbol{x}}_{k+1},k+1)$, $\nabla\sigma_{Vk+1}^{\mathrm{T}}=\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1},k+1)$ and $\nabla\varepsilon_{Vk+1}=\nabla\varepsilon_V(\boldsymbol{x}_{k+1},k+1)$, then (31) can be further derived as

$$
\begin{aligned}
\boldsymbol{e}_{\boldsymbol{u}k} &= -\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k) - \boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\tilde{\sigma}_{\boldsymbol{u}k} - \boldsymbol{\varepsilon}_{\boldsymbol{u}}(\boldsymbol{x}_k,k) + \frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\sigma_{Vk+1}^{\mathrm{T}}\boldsymbol{W}_V \\
&\quad -\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\sigma_{Vk+1}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk} + \frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\tilde{\sigma}_{Vk+1}^{\mathrm{T}}\boldsymbol{W}_V \\
&\quad -\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\tilde{\sigma}_{Vk+1}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk} + \frac{1}{2}\boldsymbol{R}^{-1}(\hat{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k) - g^{\mathrm{T}}(\hat{\boldsymbol{x}}_k))\nabla\hat{\sigma}_{Vk+1}^{\mathrm{T}}\boldsymbol{W}_V \\
&\quad +\frac{1}{2}\boldsymbol{R}^{-1}(g^{\mathrm{T}}(\hat{\boldsymbol{x}}_k) - g^{\mathrm{T}}(\boldsymbol{x}_k))\nabla\hat{\sigma}_{Vk+1}^{\mathrm{T}}\boldsymbol{W}_V - \frac{1}{2}\boldsymbol{R}^{-1}(g^{\mathrm{T}}(\hat{\boldsymbol{x}}_k) - g^{\mathrm{T}}(\boldsymbol{x}_k))\nabla\hat{\sigma}_{Vk+1}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk} \\
&\quad -\frac{1}{2}\boldsymbol{R}^{-1}(g^{\mathrm{T}}(\hat{\boldsymbol{x}}_k) - \hat{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k))\nabla\hat{\sigma}_{Vk+1}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk} - \frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\sigma_{Vk+1}^{\mathrm{T}}\boldsymbol{W}_V - \frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\varepsilon_{Vk+1} \\[2ex]
&= -\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k) - \frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\hat{\sigma}_{Vk+1}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk} - \frac{1}{2}\boldsymbol{R}^{-1}\tilde{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)\nabla\hat{\sigma}_{Vk+1}^{\mathrm{T}}\boldsymbol{W}_V \\
&\quad -\frac{1}{2}\boldsymbol{R}^{-1}(g^{\mathrm{T}}(\hat{\boldsymbol{x}}_k) - g^{\mathrm{T}}(\boldsymbol{x}_k))\nabla\hat{\sigma}_{Vk+1}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk} + \frac{1}{2}\boldsymbol{R}^{-1}\tilde{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)\nabla\hat{\sigma}_{Vk+1}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk} + \bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}
\end{aligned}
\tag{32}
$$

where $\tilde{g}(\hat{\boldsymbol{x}}_k) = g(\hat{\boldsymbol{x}}_k) - \hat{g}(\hat{\boldsymbol{x}}_k)$ and $\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k} = -\boldsymbol{\varepsilon}_{\boldsymbol{u}}(\boldsymbol{x}_k,k) + \frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\tilde{\sigma}_{Vk+1}^{\mathrm{T}}\boldsymbol{W}_V + \frac{1}{2}\boldsymbol{R}^{-1} \times$ $(g^{\mathrm{T}}(\hat{\boldsymbol{x}}_k) - g^{\mathrm{T}}(\boldsymbol{x}_k))\nabla\hat{\sigma}_{Vk+1}^{\mathrm{T}}\boldsymbol{W}_V - \frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\varepsilon_{Vk+1} - \boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\tilde{\sigma}_{\boldsymbol{u}k}$ is bounded due to the bounded ideal NN weights, activation function and reconstruction errors. Then the error dynamics for the actor NN weights are revealed to be

$$
\tilde{\boldsymbol{W}}_{\boldsymbol{u}k+1} = \tilde{\boldsymbol{W}}_{\boldsymbol{u}k} + \alpha_{\boldsymbol{u}}\frac{\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)\boldsymbol{e}_{\boldsymbol{u}k}^{\mathrm{T}}}{1+\sigma_{\boldsymbol{u}}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k,k)\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)} \tag{33}
$$

Remark 4: The update law for tuning the actor NN weights is based on gradient descent approach and it is similar to [7] with the difference being the estimated state vector $\hat{\boldsymbol{x}}_k$ is utilized as the input to the actor NN activation function instead of actual system state vector $\boldsymbol{x}_k$. In addition, total error comprising of Bellman error and terminal constraint error are utilized to tune the weights whereas in [7], the terminal constraint is ignored. Further, the optimal control scheme in this work utilizes the identified control coefficient matrix $\hat{g}(\hat{\boldsymbol{x}}_k)$, whereas in [7], the control coefficient matrix $g(\boldsymbol{x}_k)$ is assumed to be known. Due to these differences, the stability analysis differs significantly from [7].

To complete this subsection, the flowchart of our proposed finite-horizon near optimal regulation scheme is shown in Figure 1.

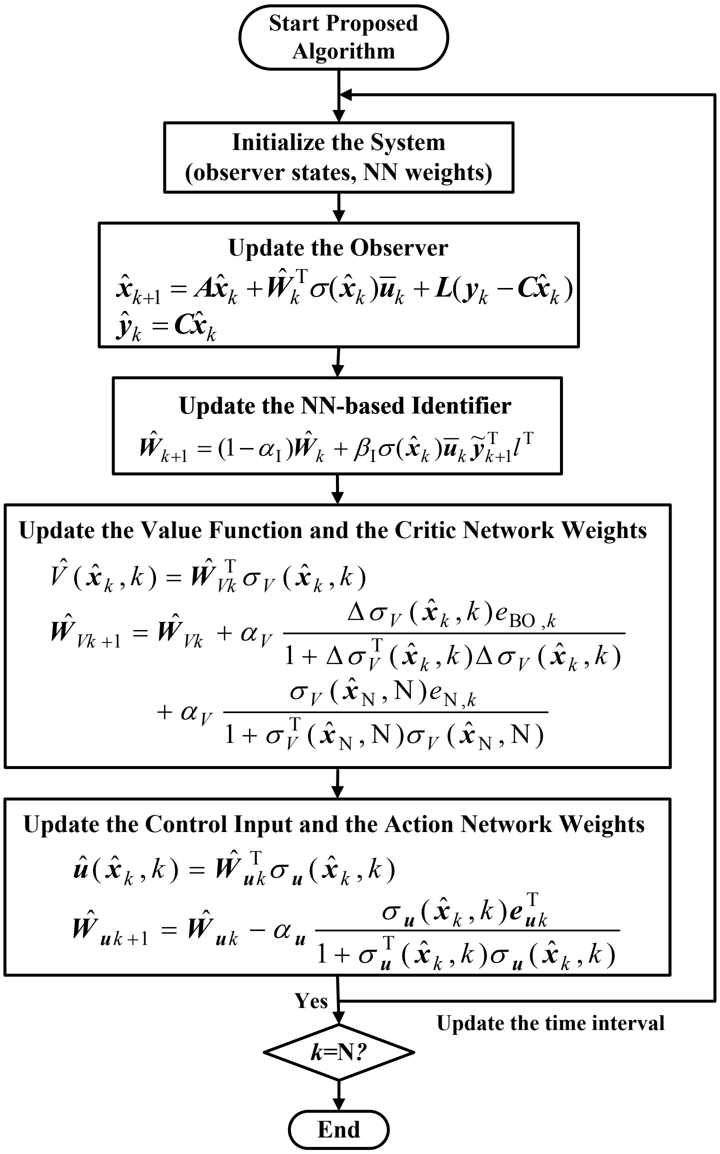

Start Proposed Algorithm
Initialize the System (observer states, NN weights)
Update the Observer
$\hat{x}_{k+1} = A\hat{x}_k + \hat{W}_k^{\mathrm{T}}\sigma(\hat{x}_k)\bar{u}_k + L(y_k - C\hat{x}_k)$
$\hat{y}_k = C\hat{x}_k$
Update the NN-based Identifier
$\hat{W}_{k+1} = (1-\alpha_{\mathrm{I}})\hat{W}_k + \beta_{\mathrm{I}}\sigma(\hat{x}_k)\bar{u}_k\tilde{y}_{k+1}^{\mathrm{T}}l^{\mathrm{T}}$
Update the Value Function and the Critic Network Weights
$\hat{V}(\hat{x}_k,k) = \hat{W}_{Vk}^{\mathrm{T}}\sigma_V(\hat{x}_k,k)$
$\hat{W}_{Vk+1} = \hat{W}_{Vk} + \alpha_V \frac{\Delta\sigma_V(\hat{x}_k,k)e_{\mathrm{BO},k}}{1+\Delta\sigma_V^{\mathrm{T}}(\hat{x}_k,k)\Delta\sigma_V(\hat{x}_k,k)} + \alpha_V \frac{\sigma_V(\hat{x}_{\mathrm{N}},\mathrm{N})e_{\mathrm{N},k}}{1+\sigma_V^{\mathrm{T}}(\hat{x}_{\mathrm{N}},\mathrm{N})\sigma_V(\hat{x}_{\mathrm{N}},\mathrm{N})}$
Update the Control Input and the Action Network Weights
$\hat{u}(\hat{x}_k,k) = \hat{W}_{uk}^{\mathrm{T}}\sigma_u(\hat{x}_k,k)$
$\hat{W}_{uk+1} = \hat{W}_{uk} - \alpha_u \frac{\sigma_u(\hat{x}_k,k)e_{uk}^{\mathrm{T}}}{1+\sigma_u^{\mathrm{T}}(\hat{x}_k,k)\sigma_u(\hat{x}_k,k)}$
Yes
Update the time interval
k=N?
End

Figure 1. Flowchart of the proposed finite-horizon near optimal regulator

We initialize the system with an admissible control as well as proper parameter selection and NN weights initialization. Then, the NNs for observer, critic and actor are updated based on our proposed weights tuning laws at each sampling interval beginning with an initial time and until the final fixed time instant in an online and forward-in-time fashion.

3.3 STABILITY ANALYSIS

In this subsection, the system stability will be investigated. It will be shown that the overall closed-loop system remain bounded under the proposed near optimal regulator design.

Theorem 3 (*Boundedness of the closed-loop system*) Let the nonlinear system (1) be controllable and observable while the system output, $\boldsymbol{y}_k \in \Omega_{\boldsymbol{y}}$, be measurable. Let the initial NN weights for the observer, critic network and actor network $\hat{\boldsymbol{W}}_k$, $\hat{\boldsymbol{W}}_{V,k}$ and $\hat{\boldsymbol{W}}_{\boldsymbol{u},k}$ be selected within compact set Ω_{OB}, Ω_V and Ω_{AN} which contains the ideal weights $\boldsymbol{W}$, $\boldsymbol{W}_V$ and $\boldsymbol{W}_{\boldsymbol{u}}$. Let $\boldsymbol{u}(0) \in \Omega_{\boldsymbol{u}}$ be an initial admissible control input for the system (1). Let the observer be given by (8) and the NN weights update law for the observer, critic network and action network be provided by (10), (26) and (28), respectively. Then, under the assumptions stated in this paper, there exists positive constant $\alpha_{\text{I}}, \alpha_V, \alpha_{\boldsymbol{u}}$, such that the observer error $\tilde{\boldsymbol{x}}_k$, NN observer weight estimation errors $\tilde{\boldsymbol{W}}_k$, critic and action network weights estimation errors $\tilde{\boldsymbol{W}}_{Vk}$ and $\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}$ are all UUB, with the ultimate bounds given by (A.20) ~ (A.23). Moreover, the estimated control input is bounded closed to the optimal value such that $\left\|\boldsymbol{u}^*(\boldsymbol{x}_k, k) - \hat{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)\right\| \le \varepsilon_{\boldsymbol{u}o}$ for a small positive constant $\varepsilon_{\boldsymbol{u}o}$.

Proof: See appendix.

4. SIMULATION RESULTS

In this section, a practical example is considered to illustrate our proposed near optimal regulation design scheme. Consider the Van der Pol oscillator with the dynamics given as

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= (1-x_1^2)x_2 - x_1 + u \\ y &= x_1 \end{aligned} \tag{34}$$

The Euler method is utilized to discretize the system with a step size of $h = 5\text{ms}$.

The weighting matrices in (2) are selected as $Q(\boldsymbol{x}_k, k) = 0.1\boldsymbol{x}_k^{\mathrm{T}}\boldsymbol{x}_k$ and $\boldsymbol{R}_k = 1$, while $\boldsymbol{A} = \begin{bmatrix} 0.5 & 0.1 \\ 0 & 0.025 \end{bmatrix}$. The terminal constraint is chosen as $\phi(\boldsymbol{x}_{\mathrm{N}}) = 1$. For the NN setup, the inputs for the NN observer is selected as $\boldsymbol{z}_k = [\hat{\boldsymbol{x}}_k, \boldsymbol{u}_k]$. The time-varying activation functions for both the critic and actor network are chosen as $\sigma_V(\hat{\boldsymbol{x}}_k, \mathrm{N}-k) = \sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, \mathrm{N}-k) = [\hat{x}_1, \hat{x}_1\exp(-\tau), \hat{x}_2, \hat{x}_2\exp(-\tau), \hat{x}_1^2, \hat{x}_2^2, \hat{x}_1\hat{x}_2\tau, \hat{x}_1^2\tau, \hat{x}_2^2\tau$ $, \hat{x}_1\hat{x}_2, \hat{x}_1\hat{x}_2\tau]^{\mathrm{T}}$, which results in 10 neurons, and $\tau = (\mathrm{N}-k)/\mathrm{N}$ is the normalized time-to-go.

The design parameters are chosen as $\alpha_{\mathrm{I}} = 0.7$, $\beta_{\mathrm{I}} = 0.01$, $\alpha_V = 0.1$, and $\alpha_{\boldsymbol{u}} = 0.03$. The initial system states and the observer states are selected as $\boldsymbol{x}_0 = [0.1, 0.1]^{\mathrm{T}}$ and $\hat{\boldsymbol{x}}_0 = [0,0]^{\mathrm{T}}$, respectively. The observer gain is chosen as $\boldsymbol{L} = [-0.3,\ 0.1]^{\mathrm{T}}$ and the matching matrix is selected as $\boldsymbol{l} = [1,\ 1]^{\mathrm{T}}$. The observer, critic and action NN weights are all initialized at random. Simulation results are shown as below.

First, the system response is shown in Figure 2 and Figure 3. From the figures, it is clear that both system states and control inputs clearly converge close enough to the origin within finite time period, which illustrates the stability of the proposed design scheme.

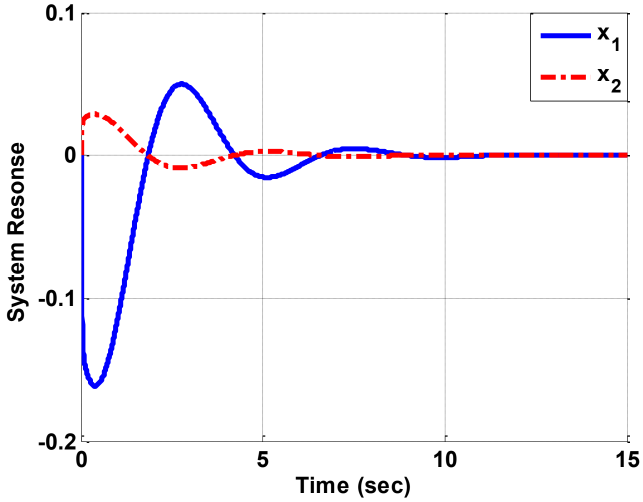


Figure 2. System response

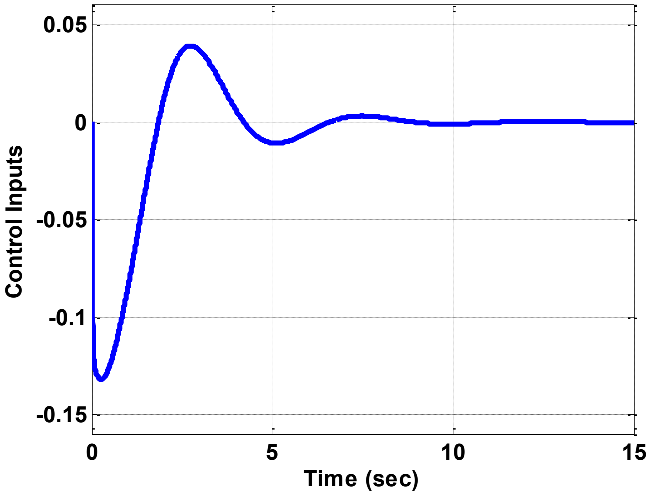


Figure 3. Control signal

Next, the history of observer error is plotted in Figure 4. From the figure, the convergence of the observer error clearly shows the feasibility of the proposed observer design.

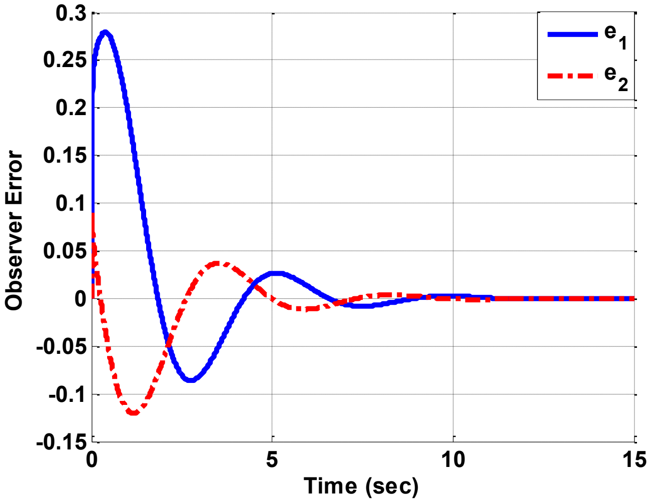


Figure 4. Observer error

Next, the error history in the design procedure is given in Figure 5 and Figure 6. From Figure 5, the Bellman equation error converges close to zero within approximately 5 seconds, which illustrates the fact that the optimality is indeed achieved. More importantly, the evolution of the terminal constraint error is shown in Figure 6. Convergence of the terminal constraint error demonstrates that the terminal constraint is also satisfied by our proposed design scheme.

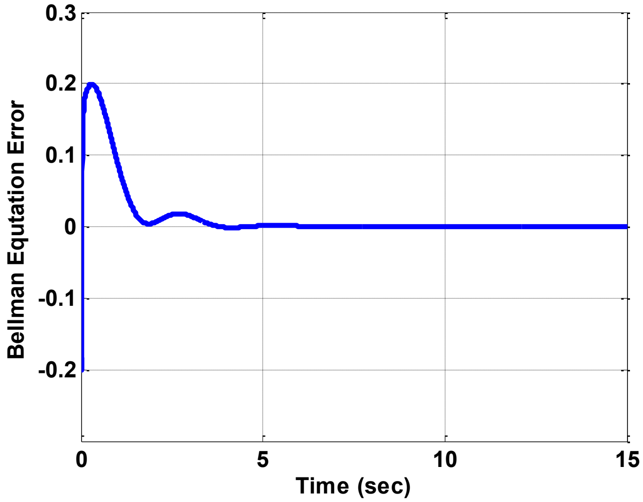


Figure 5. History of bellman equation error

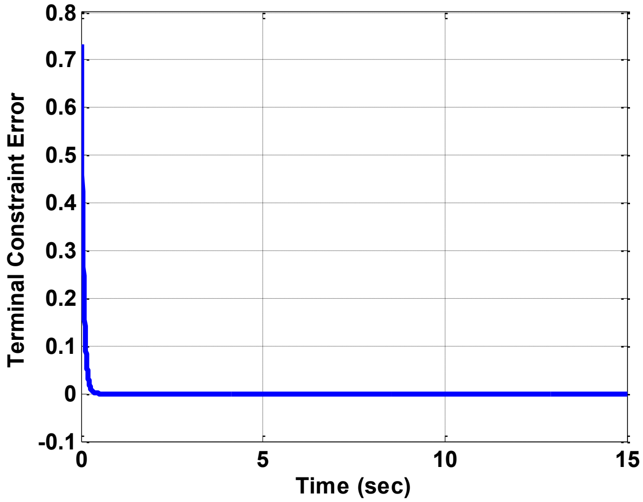


Figure 6. History of terminal constraint error

Next, the convergence of critic and actor NN weights is shown in Figure 7 and Figure 8, respectively. It can be observed from the results that both the weights converge and remain bounded, as desired.

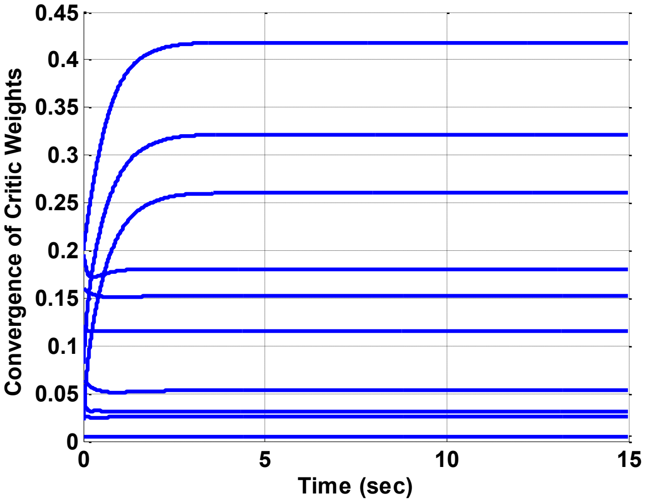


Figure 7. Convergence of critic NN weights

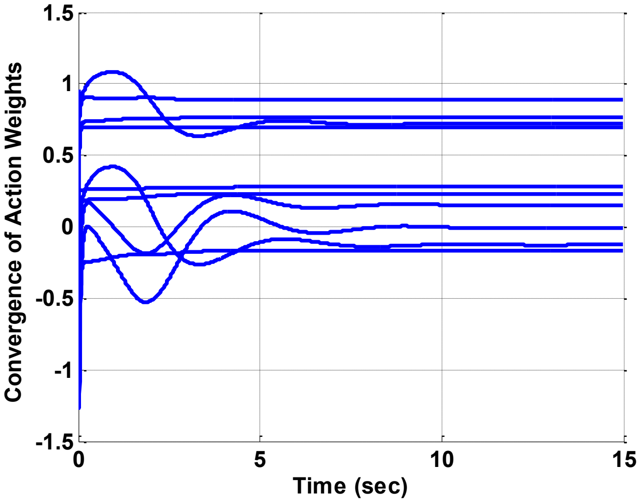


Figure 8. Convergence of actor NN weights

Finally, the comparison of the cost with a stabilizing control and our proposed near optimal control scheme is given in Figure 9. It can be seen clearly from the figure that both the cost converge to the terminal constraint $\phi(\boldsymbol{x}_{\mathrm{N}})=1$, while our design renders a lower cost when compared with the non-optimal controller design.

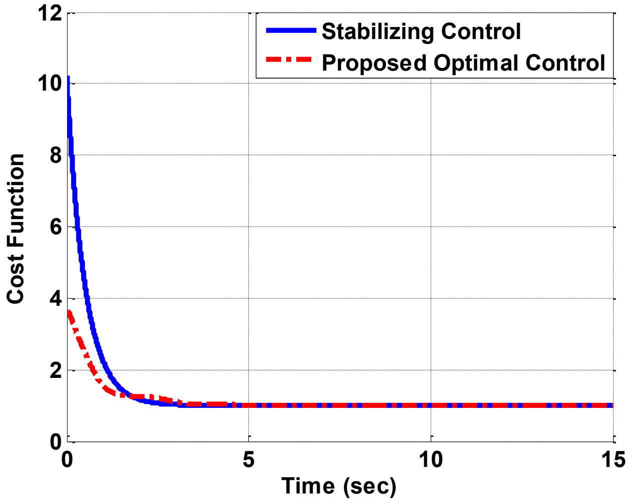


Figure 9. Comparison of the cost

5. CONCLUSIONS

In this paper, the reinforcement learning-based fixed final time near optimal regulator design by using output feedback for nonlinear discrete-time system in affine form with completely unknown system dynamics is addressed. Compared to the traditional finite-horizon optimal regulation design, the proposed scheme not only relaxes the requirement on availability of the system states and control coefficient matrix, but also functions in an online and forward-in-time manner instead of performing offline training and value/policy iteration.

The NN-based Luenberger observer relaxes the need for an additional identifier, while time-dependency nature of the finite-horizon is handled by a NN structure with constant weights and time-varying activation function. The terminal constraint is properly satisfied by minimizing an additional error term along the system trajectory. All NN weights are tuned online by using proposed update laws and Lyapunov stability theory demonstrated that the approximated control inputs converges close to its optimal value as time evolves. The performance of the proposed finite time near optimal regulator is demonstrated via simulation.

6. REFERENCES

[1] D. Kirk, *Optimal Control Theory: An Introduction*, New Jersey, Prentice-Hall, 1970.

[2] F. L. Lewis and V. L. Syrmos, *Optimal Control*, 2nd edition. New York: Wiley, 1995.

[3] J. E. Slotine and W. Li, *Applied Nonlinear Control*, Englewood Cliffs, NJ: Prentice-Hall, 1991.

[4] H. K. Khalil and Laurent Praly, "High-gain observers in nonlinear feedback control", International Journal of Robust and Nonlinear Control, 21 Jul 2013.

[5] S.J. Bradtke and B.E Ydstie, "Adaptive linear quadratic control using policy iteration", Proceedings of American Control Conference, Baltimore, MD, 1994, pp. 3475-3479.

[6] H. Xu and S. Jagannathan, "Stochastic optimal controller design for uncertain nonlinear networked control system via neuro dynamic programming," IEEE Trans. Neural Netw. and Learning Syst, vol. 24, pp. 471–484, 2013.

[7] T. Dierks and S. Jagannathan, "Online optimal control of affine nonlinear discrete-time systems with unknown internal dynamics by using time-based policy update," IEEE Trans. Neural Netw. and Learning Syst, vol. 23, pp. 1118–1129, 2012.

[8] Z. Chen and S. Jagannathan, "Generalized Hamilton-Jacobi-Bellman formulation based neural network control of affine nonlinear discrete-time systems", IEEE Trans. Neural Network, vol. 7, pp. 90–106, 2008.

[9] R. Beard, "Improving the closed-loop performance of nonlinear systems," Ph.D. dissertation, Rensselaer Polytechnic Institute, USA, 1995.

[10] T. Cheng, F. L. Lewis, and M. Abu-Khalaf, "A neural network solution for fixed-final-time optimal control of nonlinear systems," Automatica, vol. 43, pp. 482–490, 2007.

[11] A. Heydari and S. N. Balakrishan, "Finite-horizon control-constrained nonlinear optimal control using single network adaptive critics," IEEE Trans. Neural Netw. and Learning Syst., vol. 24, pp. 145–157, 2013.

[12] F.Y. Wang, N. Jin, D. Liu and Q. Wei, "Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with ε–error bound," IEEE Trans. Neural Networks, vol. 22, pp. 24–36, 2011.

[13] Q. Zhao, H. Xu and S. Jagannathan, "Finite-Horizon Optimal Adaptive Neural Network Control of Uncertain Nonlinear Discrete-time Systems," to appear in IEEE Multi-conference on Systems and Control, Hyderabad, India, 2013.

[14] P. J. Werbos, "A menu of designs for reinforcement learing over time," J. Neural Network Contr., vol. 3, pp. 835–846, 1983.

[15] T. Dierks and S. Jagannathan, “Optimal control of affine nonlinear discrete-time systems with unknown internal dynamics,” in Proc. Conf. on Decision and Control, Shanghai, pp. 6750–6755, 2009.

[16] K. S. Narendra and K. Parthasarathy, “Identification and control of dynamical systems using neural networks,” IEEE Trans. Neural Networks, vol. 1, pp. 4–27, 1990.

[17] J. Si, A. G. Barto, W. B. Powell and D. Wunsch, *Handbook of Learning and Approximate Dynamics Programming*. New York: Wiley, 2004.

[18] D. V. Prokhorov and D. Wunsch, “Adaptive critic designs,” IEEE Trans. Neural Netw., vol 8, pp. 997–1007, 1997.

[19] S. Jagannathan, *Neural Network Control of Nonlinear Discrete-Time Systems*, Boca Raton, FL: CRC Press, 2006.

[20] M. Green and J. B. Moore, “Persistency of excitation in linear systems,” Syst. and Cont. Letter, vol. 7, pp. 351–360, 1986.

APPENDIX

Proof of Theorem 1: Consider the Lyapunov function candidate as

$$L_{\mathrm{IO}}(k) = L_{\tilde{\boldsymbol{x}},k} + L_{\tilde{\boldsymbol{W}},k} \tag{A.1}$$

where $L_{\tilde{\boldsymbol{x}},k} = \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \tilde{\boldsymbol{x}}_k$, $L_{\tilde{\boldsymbol{W}},k} = \mathrm{tr}\{\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \boldsymbol{\varLambda} \tilde{\boldsymbol{W}}_k\}$ and $\boldsymbol{\varLambda} = \dfrac{2(1+\chi_{\min}^2)}{\beta_{\mathrm{I}}} \boldsymbol{I}$, with $\boldsymbol{I} \in \Re^{L \times L}$ the identity matrix and $0 < \chi_{\min}^2 < \left\| \sigma(\hat{\boldsymbol{x}}_k) \bar{\boldsymbol{u}}_k \right\|^2$ is ensured to exist by the PE conditions, and $\mathrm{tr}\{\bullet\}$ denotes the trace operator.

The first difference of $L_{\mathrm{IO}}(k)$ is given by

$$\Delta L_{\mathrm{IO}}(k) = \Delta L_{\tilde{\boldsymbol{x}},k} + \Delta L_{\tilde{\boldsymbol{W}},k} \tag{A.2}$$

Next, we consider each term in (A.2) individually. First, recall from the observer error dynamics (9), we have

$$\begin{aligned}
\Delta L_{\tilde{\boldsymbol{x}},k} &= \tilde{\boldsymbol{x}}_{k+1}^{\mathrm{T}} \tilde{\boldsymbol{x}}_{k+1} - \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \tilde{\boldsymbol{x}}_k \\
&= (\boldsymbol{A}_c \tilde{\boldsymbol{x}}_k + \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\hat{\boldsymbol{x}}_k) \bar{\boldsymbol{u}}_k + \bar{\boldsymbol{\varepsilon}}_{Ok})^{\mathrm{T}} (\boldsymbol{A}_c \tilde{\boldsymbol{x}}_k + \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\hat{\boldsymbol{x}}_k) \bar{\boldsymbol{u}}_k + \bar{\boldsymbol{\varepsilon}}_{Ok}) - \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \tilde{\boldsymbol{x}}_k \\
&= \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \boldsymbol{A}_c^{\mathrm{T}} \boldsymbol{A}_c \tilde{\boldsymbol{x}}_k + [\sigma(\hat{\boldsymbol{x}}_k) \bar{\boldsymbol{u}}_k]^{\mathrm{T}} \tilde{\boldsymbol{W}}_k \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\hat{\boldsymbol{x}}_k) \bar{\boldsymbol{u}} + \bar{\boldsymbol{\varepsilon}}_{Ok}^{\mathrm{T}} \bar{\boldsymbol{\varepsilon}}_{Ok} + 2 \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \boldsymbol{A}_c^{\mathrm{T}} \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\hat{\boldsymbol{x}}_k) \bar{\boldsymbol{u}}_k \\
&\quad + 2 \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \boldsymbol{A}_c^{\mathrm{T}} \bar{\boldsymbol{\varepsilon}}_{Ok} + 2 \bar{\boldsymbol{\varepsilon}}_{Ok}^{\mathrm{T}} \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\hat{\boldsymbol{x}}_k) \bar{\boldsymbol{u}} - \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \tilde{\boldsymbol{x}}_k \\
&\le -(1-\gamma) \left\| \tilde{\boldsymbol{x}}_k \right\|^2 + 3 \left\| \tilde{\boldsymbol{W}}_k \right\|^2 \left\| \sigma(\hat{\boldsymbol{x}}_k) \bar{\boldsymbol{u}}_k \right\|^2 + 3 \left\| \bar{\boldsymbol{\varepsilon}}_{Ok} \right\|^2
\end{aligned} \tag{A.3}$$

where $\gamma = 3 \left\| \boldsymbol{A}_c \right\|^2$.

Next, recall the NN weight estimation error dynamics (11), we have

$$\begin{aligned}
\Delta L_{\tilde{\boldsymbol{W}},k} &= \mathrm{tr}\{\tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}} \boldsymbol{\varLambda} \tilde{\boldsymbol{W}}_{k+1}\} - \mathrm{tr}\{\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \boldsymbol{\varLambda} \tilde{\boldsymbol{W}}_k\} \\
&\le 2(1-\alpha_{\mathrm{I}})^2 \mathrm{tr}\{\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \boldsymbol{\varLambda} \tilde{\boldsymbol{W}}_k\} + 6 \alpha_{\mathrm{I}}^2 \Lambda W_{\mathrm{M}}^2 + 6 \beta_{\mathrm{I}}^2 \Lambda \left\| \sigma(\hat{\boldsymbol{x}}_k) \bar{\boldsymbol{u}}_k \right\|^2 \left\| \boldsymbol{A}_c \right\|^2 \left\| \boldsymbol{lC} \right\|^2 \left\| \tilde{\boldsymbol{x}}_k \right\|^2 \\
&\quad - 4(1-\alpha_{\mathrm{I}}) \beta_{\mathrm{I}} \lambda_{\min}(\boldsymbol{lC}) \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \left\| \sigma(\hat{\boldsymbol{x}}_k) \bar{\boldsymbol{u}}_k \right\|^2 \tilde{\boldsymbol{W}}_k + 2 \beta_{\mathrm{I}}^2 \Lambda \left\| \sigma(\hat{\boldsymbol{x}}_k) \bar{\boldsymbol{u}}_k \right\|^2 \left\| \boldsymbol{lC} \right\|^2 \left\| \tilde{\boldsymbol{W}}_k \right\|^2 \\
&\quad + 6 \beta_{\mathrm{I}}^2 \Lambda \left\| \sigma(\hat{\boldsymbol{x}}_k) \bar{\boldsymbol{u}}_k \right\|^2 \left\| \boldsymbol{lC} \right\|^2 \left\| \bar{\boldsymbol{\varepsilon}}_{Ok} \right\|^2 - \mathrm{tr}\{\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \boldsymbol{\varLambda} \tilde{\boldsymbol{W}}_k\}
\end{aligned}$$

$$
\begin{aligned}
&\le -(1-2(1-\alpha_{\mathrm{I}})^2)\mathrm{tr}\{\tilde{\boldsymbol{W}}_k^{\mathrm{T}}\boldsymbol{A}\tilde{\boldsymbol{W}}_k\} + 6\alpha_{\mathrm{I}}^2\Lambda W_{\mathrm{M}}^2 + 6\beta_{\mathrm{I}}^2\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2\|\boldsymbol{A}_c\|^2\|\boldsymbol{lC}\|^2\|\tilde{\boldsymbol{x}}_k\|^2 \\
&\quad - 2\beta_{\mathrm{I}}((1-\alpha_{\mathrm{I}})\lambda_{\min}(\boldsymbol{lC}) - \beta_{\mathrm{I}}\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2)\Lambda \times \|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2\|\tilde{\boldsymbol{W}}_k\|^2 \\
&\quad + 6\beta_{\mathrm{I}}^2\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2\|\boldsymbol{lC}\|^2\|\bar{\boldsymbol{\varepsilon}}_{Ok}\|^2 \qquad\qquad \text{(A.4)} \\
&\le -(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\|\tilde{\boldsymbol{W}}_k\|^2 + 6\beta_{\mathrm{I}}^2\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2\|\boldsymbol{A}_c\|^2\|\boldsymbol{lC}\|^2\|\tilde{\boldsymbol{x}}_k\|^2 \\
&\quad - 2\beta_{\mathrm{I}}((1-\alpha_{\mathrm{I}})\lambda_{\min}(\boldsymbol{lC}) - \beta_{\mathrm{I}}\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2)\Lambda \times \|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2\|\tilde{\boldsymbol{W}}_k\|^2 + \varepsilon_{\mathrm{WM}}
\end{aligned}
$$

where $\Lambda = \|\boldsymbol{A}\|$ and $\varepsilon_{\mathrm{WM}} = 6\alpha_{\mathrm{I}}^2\Lambda W_{\mathrm{M}}^2 + 6\beta_{\mathrm{I}}^2\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2\|\boldsymbol{lC}\|^2\|\bar{\boldsymbol{\varepsilon}}_{Ok}\|^2$.

Therefore, the first difference of the total Lyapunov candidate, by combining (A.3) and (A.4), is given as

$$
\begin{aligned}
&\Delta L_{\mathrm{IO}}(k) = \Delta L_{\tilde{\boldsymbol{x}},k} + \Delta L_{\tilde{\boldsymbol{W}},k} \\
&\le -(1-\gamma)\|\tilde{\boldsymbol{x}}_k\|^2 + 3\|\tilde{\boldsymbol{W}}_k\|^2\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2 + 3\|\bar{\boldsymbol{\varepsilon}}_{Ok}\|^2 - (1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\|\tilde{\boldsymbol{W}}_k\|^2 \\
&\quad - \frac{2\beta_{\mathrm{I}}\Lambda}{1+\chi_{\min}^2}\|\tilde{\boldsymbol{W}}_k\|^2\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2 + 2\beta_{\mathrm{I}}\gamma\|\boldsymbol{lC}\|^2\Lambda\|\tilde{\boldsymbol{x}}_k\|^2 + \varepsilon_{\mathrm{WM}} \qquad\qquad \text{(A.5)} \\
&\le -(1-(1+4\|\boldsymbol{lC}\|^2(1+\chi_{\min}^2))\gamma)\|\tilde{\boldsymbol{x}}_k\|^2 - \|\tilde{\boldsymbol{W}}_k\|^2\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2 \\
&\quad - (1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\|\tilde{\boldsymbol{W}}_k\|^2 + \varepsilon_{\mathrm{OM}}
\end{aligned}
$$

where $\varepsilon_{\mathrm{OM}} = 3\|\bar{\boldsymbol{\varepsilon}}_{Ok}\|^2 + \varepsilon_{\mathrm{WM}}$. By standard Lyapunov stability theory, $\Delta L_{\mathrm{IO},k}$ is less than zero outside a compact set as long as the following conditions hold:

$$
\|\tilde{\boldsymbol{x}}_k\| > \sqrt{\frac{\varepsilon_{\mathrm{OM}}}{1-(1+4\|\boldsymbol{lC}\|^2(1+\chi_{\min}^2))\gamma}} \equiv b_{\tilde{\boldsymbol{x}}} \qquad \text{(A.6)}
$$

Or

$$
\|\tilde{\boldsymbol{W}}_k\| > \sqrt{\frac{\varepsilon_{\mathrm{OM}}}{\chi_{\min}^2 + (1-2(1-\alpha_{\mathrm{I}})^2)\Lambda}} \equiv b_{\tilde{\boldsymbol{W}}} \qquad \text{(A.7)}
$$

Note that in (A.6) the denominator is guaranteed to be positive, i.e., $0 < \gamma < \dfrac{1}{1+4\|\boldsymbol{lC}\|^2(1+\chi_{\min}^2)}$, by properly selecting the designed parameters $\boldsymbol{A}$, $\boldsymbol{L}$ and $\boldsymbol{l}$.

Proof of Theorem 2: First, for simplicity, denote $\Delta\hat{\sigma}_{Vk} = \Delta\sigma_V(\hat{\boldsymbol{x}}_k, k)$, $\Delta\varepsilon_{VBk} = \Delta\varepsilon_{VB}(\boldsymbol{x}_k, k)$ and $\hat{\sigma}_{V\mathrm{N}} = \sigma_V(\hat{\boldsymbol{x}}_\mathrm{N}, \mathrm{N})$. Consider the following Lyapunov candidate

$$L_{\tilde{\boldsymbol{W}}_V}(k) = L(\tilde{\boldsymbol{W}}_{Vk}) + \Pi L(\tilde{\boldsymbol{x}}_k) + \Pi L(\tilde{\boldsymbol{W}}_k) \tag{A.8}$$

where $L(\tilde{\boldsymbol{W}}_{Vk}) = \tilde{\boldsymbol{W}}_{Vk}^\mathrm{T}\tilde{\boldsymbol{W}}_{Vk}$, $L(\tilde{\boldsymbol{x}}_k) = (\tilde{\boldsymbol{x}}_k^\mathrm{T}\tilde{\boldsymbol{x}}_k)^2$ $L(\tilde{\boldsymbol{W}}_k) = (\mathrm{tr}\{\tilde{\boldsymbol{W}}_k^\mathrm{T}\boldsymbol{\Lambda}\tilde{\boldsymbol{W}}_k\})^2$ and $\Pi = \dfrac{\alpha_V(1+3\alpha_V)L_m^2}{(1+\Delta\hat{\sigma}_{\min}^2)(1-3\gamma^2)}$. Next, take each term in (A.8) individually. The first difference of $L(\tilde{\boldsymbol{W}}_{Vk})$, by recalling (24), is given by

$$\begin{aligned}
\Delta L(\tilde{\boldsymbol{W}}_{Vk}) &= \tilde{\boldsymbol{W}}_{Vk+1}^\mathrm{T}\tilde{\boldsymbol{W}}_{Vk+1} - \tilde{\boldsymbol{W}}_{Vk}^\mathrm{T}\tilde{\boldsymbol{W}}_{Vk} \\
&= \left(\tilde{\boldsymbol{W}}_{Vk} - \alpha_V\frac{\Delta\hat{\sigma}_{Vk}e_{\mathrm{BO},k}}{1+\Delta\hat{\sigma}_{Vk}^\mathrm{T}\Delta\hat{\sigma}_{Vk}} - \alpha_V\frac{\hat{\sigma}_{V\mathrm{N}}e_{\mathrm{N},k}}{1+\hat{\sigma}_{V\mathrm{N}}^\mathrm{T}\hat{\sigma}_{V\mathrm{N}}}\right)^\mathrm{T} \times \\
&\quad \left(\tilde{\boldsymbol{W}}_{Vk} - \alpha_V\frac{\Delta\hat{\sigma}_{Vk}e_{\mathrm{BO},k}}{1+\Delta\hat{\sigma}_{Vk}^\mathrm{T}\Delta\hat{\sigma}_{Vk}} - \alpha_V\frac{\hat{\sigma}_{V\mathrm{N}}e_{\mathrm{N},k}}{1+\hat{\sigma}_{V\mathrm{N}}^\mathrm{T}\hat{\sigma}_{V\mathrm{N}}}\right) - \tilde{\boldsymbol{W}}_{Vk}^\mathrm{T}\tilde{\boldsymbol{W}}_{Vk} \\
&= -2\alpha_V\frac{\tilde{\boldsymbol{W}}_{Vk}^\mathrm{T}\Delta\hat{\sigma}_{Vk}e_{\mathrm{BO},k}}{1+\Delta\hat{\sigma}_{Vk}^\mathrm{T}\Delta\hat{\sigma}_{Vk}} - 2\alpha_V\frac{\tilde{\boldsymbol{W}}_{Vk}^\mathrm{T}\hat{\sigma}_{V\mathrm{N}}e_{\mathrm{N},k}}{1+\hat{\sigma}_{V\mathrm{N}}^\mathrm{T}\hat{\sigma}_{V\mathrm{N}}} + \alpha_V^2\frac{e_{\mathrm{BO},k}^2\Delta\hat{\sigma}_{Vk}^\mathrm{T}\Delta\hat{\sigma}_{Vk}}{(1+\Delta\hat{\sigma}_{Vk}^\mathrm{T}\Delta\hat{\sigma}_{Vk})^2} + \alpha_V^2\frac{e_{\mathrm{N},k}^2\hat{\sigma}_{V\mathrm{N}}^\mathrm{T}\hat{\sigma}_{V\mathrm{N}}}{(1+\hat{\sigma}_{V\mathrm{N}}^\mathrm{T}\hat{\sigma}_{V\mathrm{N}})^2}
\end{aligned} \tag{A.9}$$

Recall from (22) and (23), the first difference of $L(\tilde{\boldsymbol{W}}_{Vk})$ can be further derived as

$$
\begin{aligned}
\Delta L(\tilde{\boldsymbol{W}}_{Vk}) \le & -2\alpha_V \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}\left(L_m\|\tilde{\boldsymbol{x}}_k\|^2+\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}+\Delta\varepsilon_{VBk}\right)}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}} - 2\alpha_V \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}\left(\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}+\varepsilon_{V\mathrm{N}}\right)}{1+\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}} \\
& +\alpha_V^2 \frac{\left(L_m\|\tilde{\boldsymbol{x}}_k\|^2+\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}+\Delta\varepsilon_{VBk}\right)^2\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}}{(1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk})^2} + \alpha_V^2 \frac{\left(\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}+\varepsilon_{V\mathrm{N}}\right)^2\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}}{(1+\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}})^2} \\
\le & -2\alpha_V \frac{L_m\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}\|\tilde{\boldsymbol{x}}_k\|^2}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}} - 2\alpha_V \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk}}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}} - 2\alpha_V \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}\Delta\varepsilon_{VBk}}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}} \\
& -2\alpha_V \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk}}{1+\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}} - 2\alpha_V \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}\varepsilon_{V\mathrm{N}}}{1+\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}} + 3\alpha_V^2 \frac{L_m^2\|\tilde{\boldsymbol{x}}_k\|^4}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}} \\
& +3\alpha_V^2 \frac{\Delta\varepsilon_{VBk}^2}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}} + 3\alpha_V^2 \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk}}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}} \\
& +2\alpha_V^2 \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{V\mathrm{N}}\Delta\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk}}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}} + 2\alpha_V^2 \frac{\varepsilon_{V\mathrm{N}}^2}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}} \\
\le & -\frac{\alpha_V(1-6\alpha_V)}{2}\frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk}}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}} - \alpha_V(1-2\alpha_V)\frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk}}{1+\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}} \\
& +\alpha_V(1+3\alpha_V)\frac{L_m^2\|\tilde{\boldsymbol{x}}_k\|^4}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}} + \varepsilon_{V\mathrm{TM}} \\
\le & -\frac{\alpha_V(1-6\alpha_V)}{2}\frac{\Delta\hat{\sigma}_{\min}^2}{1+\Delta\hat{\sigma}_{\min}^2}\left\|\tilde{\boldsymbol{W}}_{Vk}\right\|^2 - \alpha_V(1-2\alpha_V)\frac{\hat{\sigma}_{\min}^2}{1+\hat{\sigma}_{\min}^2}\left\|\tilde{\boldsymbol{W}}_{Vk}\right\|^2 \\
& +\frac{\alpha_V(1+3\alpha_V)}{1+\Delta\hat{\sigma}_{\min}^2}L_m^2\|\tilde{\boldsymbol{x}}_k\|^4 + \varepsilon_{V\mathrm{TM}}
\end{aligned}
\tag{A.10}
$$

where $\varepsilon_{V\mathrm{TM}} = \alpha_V(1+3\alpha_V)\dfrac{\Delta\varepsilon_{VBk}^2}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}} + \alpha_V(1+2\alpha_V)\dfrac{\varepsilon_{V\mathrm{N}}^2}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}}$.

Next, consider $L(\tilde{\boldsymbol{x}}_k)$. Recall (A.3) and apply Cauchy-Schwartz inequality, the first difference of $L(\tilde{\boldsymbol{x}}_k)$ is given by

$$\begin{aligned}
\Delta L(\tilde{\boldsymbol{x}}_k) &= (\tilde{\boldsymbol{x}}_{k+1}^{\mathrm{T}}\tilde{\boldsymbol{x}}_{k+1})^2 - (\tilde{\boldsymbol{x}}_k^{\mathrm{T}}\tilde{\boldsymbol{x}}_k)^2 \\
&\le \left[-(1-\gamma)\|\tilde{\boldsymbol{x}}_k\|^2 + 3\left\|\tilde{\boldsymbol{W}}_k\right\|^2 \|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2 + 3\|\bar{\boldsymbol{\varepsilon}}_{Ok}\|^2 \right] \times \\
&\quad \left[(1+\gamma)\|\tilde{\boldsymbol{x}}_k\|^2 + 3\left\|\tilde{\boldsymbol{W}}_k\right\|^2 \|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2 + 3\|\bar{\boldsymbol{\varepsilon}}_{Ok}\|^2 \right] \\
&\le -(1-\gamma^2)\|\tilde{\boldsymbol{x}}_k\|^4 + \left(3\left\|\tilde{\boldsymbol{W}}_k\right\|^2 \|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2 + 3\|\bar{\boldsymbol{\varepsilon}}_{Ok}\|^2 \right)^2 \\
&\quad + 2\gamma\|\tilde{\boldsymbol{x}}_k\|^2 \left(3\left\|\tilde{\boldsymbol{W}}_k\right\|^2 \|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2 + 3\|\bar{\boldsymbol{\varepsilon}}_{Ok}\|^2 \right) \\
&\le -(1-2\gamma^2)\|\tilde{\boldsymbol{x}}_k\|^4 + 36\left\|\tilde{\boldsymbol{W}}_k\right\|^4 \|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^4 + 36\|\bar{\boldsymbol{\varepsilon}}_{Ok}\|^4
\end{aligned} \tag{A.11}$$

Next, take $L(\tilde{\boldsymbol{W}}_k)$. Recall (A.4) and write the difference $L(\tilde{\boldsymbol{W}}_k)$ as

$$\begin{aligned}
\Delta L(\tilde{\boldsymbol{W}}_k) &= (\mathrm{tr}\{\tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}}\boldsymbol{\Lambda}\tilde{\boldsymbol{W}}_{k+1}\})^2 - (\mathrm{tr}\{\tilde{\boldsymbol{W}}_k^{\mathrm{T}}\boldsymbol{\Lambda}\tilde{\boldsymbol{W}}_k\})^2 \\
&\le \{-(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\left\|\tilde{\boldsymbol{W}}_k\right\|^2 - 2\beta_{\mathrm{I}}\eta\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2\left\|\tilde{\boldsymbol{W}}_k\right\|^2 \\
&\quad + 6\beta_{\mathrm{I}}^2\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2\|\boldsymbol{A}_c\|^2\|\boldsymbol{lC}\|^2\|\tilde{\boldsymbol{x}}_k\|^2 + \varepsilon_{\mathrm{WM}}\} \times \{(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\left\|\tilde{\boldsymbol{W}}_k\right\|^2 \\
&\quad - 2\beta_{\mathrm{I}}\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2\left\|\tilde{\boldsymbol{W}}_k\right\|^2 + 6\beta_{\mathrm{I}}^2\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2\|\boldsymbol{A}_c\|^2\|\boldsymbol{lC}\|^2\|\tilde{\boldsymbol{x}}_k\|^2 + \varepsilon_{\mathrm{WM}}\} \\
&\le \{-(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\left\|\tilde{\boldsymbol{W}}_k\right\|^2 - 2\beta_{\mathrm{I}}\eta\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^4\left\|\tilde{\boldsymbol{W}}_k\right\|^2 \\
&\quad + 6\beta_{\mathrm{I}}^2\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2\|\boldsymbol{A}_c\|^2\|\boldsymbol{lC}\|^2\|\tilde{\boldsymbol{x}}_k\|^2 + \varepsilon_{\mathrm{WM}}\} \times \\
&\quad \{(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\left\|\tilde{\boldsymbol{W}}_k\right\|^2 - 2\beta_{\mathrm{I}}\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^4\left\|\tilde{\boldsymbol{W}}_k\right\|^2 \\
&\quad + 6\beta_{\mathrm{I}}^2\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2\|\boldsymbol{A}_c\|^2\|\boldsymbol{lC}\|^2\|\tilde{\boldsymbol{x}}_k\|^2 + \varepsilon_{\mathrm{WM}}\} \\
&\le -(1-8(1-\alpha_{\mathrm{I}})^4)\Lambda^2\left\|\tilde{\boldsymbol{W}}_k\right\|^4 + 5\varepsilon_{\mathrm{WM}}^2 - 8(1-\alpha_{\mathrm{I}})^2\beta_{\mathrm{I}}\eta\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^4\left\|\tilde{\boldsymbol{W}}_k\right\|^2 \\
&\quad + 210\beta_{\mathrm{I}}^2\Lambda^2\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^4\|\boldsymbol{A}_c\|^4\|\boldsymbol{lC}\|^4\|\tilde{\boldsymbol{x}}_k\|^4 + 12\beta_{\mathrm{I}}^2\eta^2\Lambda^2\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^8\left\|\tilde{\boldsymbol{W}}_k\right\|^4 \\
&\le -(1-8(1-\alpha_{\mathrm{I}})^4)\Lambda^2\left\|\tilde{\boldsymbol{W}}_k\right\|^4 - 4(2(1-\alpha_{\mathrm{I}})^2 - 3\eta)\beta_{\mathrm{I}}\eta\Lambda^2\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^4\left\|\tilde{\boldsymbol{W}}_k\right\|^4 \\
&\quad + 210\Lambda^2\|\boldsymbol{A}_c\|^4\|\boldsymbol{lC}\|^4\|\tilde{\boldsymbol{x}}_k\|^4 + 5\varepsilon_{\mathrm{WM}}^2
\end{aligned} \tag{A.12}$$

where $\eta = 2(1-\alpha_{\mathrm{I}})\lambda_{\min}(\boldsymbol{lC}) - \beta_{\mathrm{I}}\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2$.

Therefore, combining (A.11) and (A.12) yields

$$\begin{aligned}
&\Delta L(\tilde{\boldsymbol{x}}_k) + \Delta L(\tilde{\boldsymbol{W}}_k) \\
&\leq -(1-2\gamma^2)\|\tilde{\boldsymbol{x}}_k\|^4 + 36\|\tilde{\boldsymbol{W}}_k\|^4\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^4 + 36\|\bar{\boldsymbol{\varepsilon}}_{Ok}\|^4 - (1-8(1-\alpha_{\mathrm{I}})^4)\Lambda^2\|\tilde{\boldsymbol{W}}_k\|^4 \\
&\quad - 4(2(1-\alpha_{\mathrm{I}})^2 - 3\eta)\beta_{\mathrm{I}}\eta\Lambda^2\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^4\|\tilde{\boldsymbol{W}}_k\|^4 + 52.5\Lambda^2\gamma^2\|\boldsymbol{A}_c\|^4\|\boldsymbol{lC}\|^4\|\tilde{\boldsymbol{x}}_k\|^4 + 5\varepsilon_{\mathrm{WM}}^2 \\
&\leq -(1-3\gamma^2)\|\tilde{\boldsymbol{x}}_k\|^4 - 4\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^4\|\tilde{\boldsymbol{W}}_k\|^4 - (1-8(1-\alpha_{\mathrm{I}})^4)\Lambda^2\|\tilde{\boldsymbol{W}}_k\|^4 + \varepsilon_4
\end{aligned} \tag{A.13}$$

where $\varepsilon_4 = 36\|\bar{\boldsymbol{\varepsilon}}_{Ok}\|^4 + 5\varepsilon_{\mathrm{WM}}^2$.

Finally, combining (A.10) and (A.13) yields the first difference of total Lyapunov candidate as

$$\begin{aligned}
&\Delta L_{\tilde{\boldsymbol{W}}_V}(k) = \Delta L(\tilde{\boldsymbol{W}}_{Vk}) + \Pi\Delta L(\tilde{\boldsymbol{x}}_k) + \Pi\Delta L(\tilde{\boldsymbol{W}}_k) \\
&\leq -\frac{\alpha_V(1-6\alpha_V)}{2}\frac{\Delta\hat{\sigma}_{\min}^2}{1+\Delta\hat{\sigma}_{\min}^2}\|\tilde{\boldsymbol{W}}_{Vk}\|^2 - \alpha_V(1-2\alpha_V)\frac{\hat{\sigma}_{\min}^2}{1+\hat{\sigma}_{\min}^2}\|\tilde{\boldsymbol{W}}_{Vk}\|^2 \\
&\quad -\frac{\alpha_V(1+3\alpha_V)}{1+\Delta\hat{\sigma}_{\min}^2}L_m^2\|\tilde{\boldsymbol{x}}_k\|^4 - 4\Pi\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^4\|\tilde{\boldsymbol{W}}_k\|^4 - (1-8(1-\alpha_{\mathrm{I}})^4)\Pi\Lambda^2\|\tilde{\boldsymbol{W}}_k\|^4 + \varepsilon_1
\end{aligned} \tag{A.14}$$

where $\varepsilon_1 = \varepsilon_{V\mathrm{TM}} + \Pi\varepsilon_4$. By using standard Lyapunov stability analysis, ΔL is less than zero outside a compact set as long as the following conditions hold:

$$\|\tilde{\boldsymbol{x}}_k\| > \sqrt[4]{\frac{\varepsilon_1}{\frac{\alpha_V(1+3\alpha_V)}{1+\Delta\hat{\sigma}_{\min}^2}L_m^2}} \equiv b_{\tilde{\boldsymbol{x}}} \tag{A.15}$$

Or

$$\|\tilde{\boldsymbol{W}}_{Vk}\| > \sqrt{\frac{\varepsilon_1}{\frac{\alpha_V(1-6\alpha_V)}{2}\frac{\Delta\hat{\sigma}_{\min}^2}{1+\Delta\hat{\sigma}_{\min}^2} + \alpha_V(1-2\alpha_V)\frac{\hat{\sigma}_{\min}^2}{1+\hat{\sigma}_{\min}^2}}} \equiv b_{\tilde{\boldsymbol{W}}_V} \tag{A.16}$$

Or

$$\|\tilde{\boldsymbol{W}}_k\| > \sqrt[4]{\frac{\varepsilon_1}{4\Pi\chi_{\min}^4 + (1-8(1-\alpha_{\mathrm{I}})^4)\Pi\Lambda^2}} \equiv b_{\tilde{\boldsymbol{W}}} \tag{A.17}$$

Proof of Theorem 3: Consider the following Lyapunov candidate as

$$L(k) = L_{\mathrm{IO}}(k) + L_{\widetilde{\boldsymbol{W}}_V}(k) + \Sigma L_{\widetilde{\boldsymbol{W}}_{\boldsymbol{u}}}(k) \tag{A.18}$$

where $L_{\mathrm{IO}}(k)$ and $L_{\widetilde{\boldsymbol{W}}_V}(k)$ are defined in (A.1) and (A.8), respectively, and

$$L_{\widetilde{\boldsymbol{W}}_{\boldsymbol{u}}}(k) = \left\|\widetilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\| \text{ with } \Sigma = \min\left\{ \begin{array}{c} \dfrac{\alpha_V(1-2\alpha_V)\hat{\sigma}^2_{\min}\bar{\varepsilon}_{\boldsymbol{u}\mathrm{M}}}{\alpha_{\boldsymbol{u}}(5+4\bar{\varepsilon}_{\boldsymbol{u}\mathrm{M}})\nabla\sigma^2_{V\max}\hat{\sigma}^2_{\min}+2\bar{\varepsilon}_{\boldsymbol{u}\mathrm{M}}}, \\ \dfrac{(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda(1+\hat{\sigma}^2_{\min})\bar{\varepsilon}_{\boldsymbol{u}\mathrm{M}}}{2\alpha_{\boldsymbol{u}}\sigma^2_g(1+\hat{\sigma}^2_{\min}+\bar{\varepsilon}_{\boldsymbol{u}\mathrm{M}}\lambda_{\max}(\boldsymbol{R}^{-1}))} \end{array} \right\}.$$

Denote $\hat{\sigma}_{\boldsymbol{u}k} = \sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)$, $\boldsymbol{g}_k^{\mathrm{T}} = g^{\mathrm{T}}(\boldsymbol{x}_k)$, $\hat{\boldsymbol{g}}_k^{\mathrm{T}} = g^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)$ and $\widetilde{\boldsymbol{g}}_k^{\mathrm{T}} = \widetilde{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)$ for simplicity, then the first difference of $L_{\widetilde{\boldsymbol{W}}_{\boldsymbol{u}}}(k)$ recalling from (33) and (32), is given by

$$\begin{aligned}
\Delta L_{\widetilde{\boldsymbol{W}}_{\boldsymbol{u}}}(k) &= \left\|\widetilde{\boldsymbol{W}}_{\boldsymbol{u}k+1}\right\| - \left\|\widetilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\| = \left\|\widetilde{\boldsymbol{W}}_{\boldsymbol{u}k} + \alpha_{\boldsymbol{u}}\frac{\hat{\sigma}_{\boldsymbol{u}k}\boldsymbol{e}^{\mathrm{T}}_{\boldsymbol{u}k}}{1+\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k}}\right\| - \left\|\widetilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\| \\
&\le \left\| \begin{array}{c} \widetilde{\boldsymbol{W}}_{\boldsymbol{u}k} - \alpha_{\boldsymbol{u}}\dfrac{\hat{\sigma}_{\boldsymbol{u}k}\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\widetilde{\boldsymbol{W}}_{\boldsymbol{u}k}}{1+\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k}} - \alpha_{\boldsymbol{u}}\dfrac{\hat{\sigma}_{\boldsymbol{u}k}\boldsymbol{R}^{-1}\boldsymbol{g}^{\mathrm{T}}_k\nabla\hat{\sigma}^{\mathrm{T}}_{Vk+1}\widetilde{\boldsymbol{W}}_{Vk}}{2(1+\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k})} \\ -\alpha_{\boldsymbol{u}}\dfrac{\hat{\sigma}_{\boldsymbol{u}k}\boldsymbol{R}^{-1}\widetilde{\boldsymbol{g}}^{\mathrm{T}}_k\nabla\hat{\sigma}^{\mathrm{T}}_{Vk+1}\boldsymbol{W}_V}{2(1+\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k})} \\ -\alpha_{\boldsymbol{u}}\dfrac{\hat{\sigma}_{\boldsymbol{u}k}\boldsymbol{R}^{-1}\left(\hat{\boldsymbol{g}}^{\mathrm{T}}_k - \boldsymbol{g}^{\mathrm{T}}_k\right)\nabla\hat{\sigma}^{\mathrm{T}}_{Vk+1}\widetilde{\boldsymbol{W}}_{Vk}}{2(1+\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k})} \\ +\alpha_{\boldsymbol{u}}\dfrac{\hat{\sigma}_{\boldsymbol{u}k}\boldsymbol{R}^{-1}\widetilde{\boldsymbol{g}}^{\mathrm{T}}_k\nabla\hat{\sigma}^{\mathrm{T}}_{Vk+1}\widetilde{\boldsymbol{W}}_{Vk}}{2(1+\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k})} + \alpha_{\boldsymbol{u}}\dfrac{\hat{\sigma}_{\boldsymbol{u}k}\bar{\boldsymbol{\varepsilon}}^{\mathrm{T}}_{\boldsymbol{u}k}}{1+\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k}} \end{array} \right\| - \left\|\widetilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\| \\
&\le \left(1-\alpha_{\boldsymbol{u}}\frac{\hat{\sigma}^2_{\boldsymbol{u}\min}}{1+\hat{\sigma}^2_{\boldsymbol{u}\min}}\right)\left\|\widetilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\| + \alpha_{\boldsymbol{u}}\left\|\frac{\boldsymbol{R}^{-2}g^2_{\mathrm{M}}\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k}\bar{\boldsymbol{\varepsilon}}^{\mathrm{T}}_{\boldsymbol{u}k}}{4(1+\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k})^2}\right\| + \alpha_{\boldsymbol{u}}\left(1+\frac{5}{4\left\|\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}\right\|}\right)\left\|\nabla\hat{\sigma}^{\mathrm{T}}_{Vk+1}\widetilde{\boldsymbol{W}}_{Vk}\right\|^2 \\
&\quad + \frac{\alpha_{\boldsymbol{u}}\sigma^2_g}{4\left\|\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}\right\|}\left\|\widetilde{\boldsymbol{W}}_k\right\|^2 + \alpha_{\boldsymbol{u}}\left\|\frac{\boldsymbol{R}^{-2}\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k}\nabla\hat{\sigma}^{\mathrm{T}}_{Vk+1}\nabla\hat{\sigma}_{Vk+1}W^2_{\mathrm{VM}}\bar{\varepsilon}_{\boldsymbol{u}\mathrm{M}}}{4(1+\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k})^2}\right\| + \alpha_{\boldsymbol{u}}\left\|\frac{\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k}\boldsymbol{R}^{-2}g^2_{\mathrm{M}}\bar{\varepsilon}_{\boldsymbol{u}\mathrm{M}}}{(1+\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k})^2}\right\| \\
&\quad + \alpha_{\boldsymbol{u}}\left\|\frac{\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k}\boldsymbol{R}^{-2}}{4(1+\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k})^2}\right\|\sigma^2_g\left\|\widetilde{\boldsymbol{W}}_k\right\|^2 + \alpha_{\boldsymbol{u}}\left\|\frac{\hat{\sigma}_{\boldsymbol{u}k}\bar{\boldsymbol{\varepsilon}}^{\mathrm{T}}_{\boldsymbol{u}k}}{1+\hat{\sigma}^{\mathrm{T}}_{\boldsymbol{u}k}\hat{\sigma}_{\boldsymbol{u}k}}\right\| - \left\|\widetilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\| \\
&\le -\alpha_{\boldsymbol{u}}\frac{\hat{\sigma}^2_{\boldsymbol{u}\min}}{1+\hat{\sigma}^2_{\boldsymbol{u}\min}}\left\|\widetilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\| + \alpha_{\boldsymbol{u}}\left(1+\frac{5}{4\left\|\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}\right\|}\right)\nabla\sigma^2_{V\max}\left\|\widetilde{\boldsymbol{W}}_{Vk}\right\|^2 \\
&\quad + \alpha_{\boldsymbol{u}}\sigma^2_g\left(\frac{1}{4\left\|\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}\right\|} + \frac{\lambda_{\max}(\boldsymbol{R}^{-1})}{4(1+\sigma^2_{\boldsymbol{u}\min})}\right)\left\|\widetilde{\boldsymbol{W}}_k\right\|^2 + \bar{\varepsilon}_{T\mathrm{M}}
\end{aligned} \tag{A.19}$$

where $\bar{\varepsilon}_{T\mathrm{M}} = \alpha_{\boldsymbol{u}} \left(\begin{array}{c} \dfrac{5\lambda_{\max}(\boldsymbol{R}^{-2}) g_{\mathrm{M}}^2}{4(1+\sigma_{\boldsymbol{u}\min}^2)} \\ + \dfrac{\lambda_{\max}(\boldsymbol{R}^{-2}) \nabla\sigma_{V\max}^2 W_{\mathrm{VM}}^2}{4(1+\sigma_{\boldsymbol{u}\min}^2)} + \dfrac{\sigma_{\boldsymbol{u}\max}}{1+\sigma_{\boldsymbol{u}\min}^2} \end{array} \right) \bar{\varepsilon}_{\boldsymbol{u}\mathrm{M}}$, $0 < \sigma_{\boldsymbol{u}\min} < \left\| \hat{\sigma}_{\boldsymbol{u}k} \right\| < \sigma_{\boldsymbol{u}\max}$ and $\left\| \nabla \hat{\sigma}_{Vk+1}^{\mathrm{T}} \right\| \le \nabla\sigma_{V\max}$.

Combine (A.5), (A.14) and (A.19) to obtain the first difference of the total Lyapunov candidate as

$$
\begin{aligned}
&\Delta L(k) = \Delta L_{\mathrm{IO}}(k) + \Delta L_{\widetilde{\boldsymbol{W}}_V}(k) + \Delta \Sigma L_{\widetilde{\boldsymbol{W}}_{\boldsymbol{u}}}(k) \\
&\le -(1-(1+4\gamma\|\boldsymbol{lC}\|^2)(1+\chi_{\min}^2)\gamma)\|\widetilde{\boldsymbol{x}}_k\|^2 + \varepsilon_{\mathrm{OM}} - \left\|\widetilde{\boldsymbol{W}}_k\right\|^2 \|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2 \\
&\quad - (1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\left\|\widetilde{\boldsymbol{W}}_k\right\|^2 + \frac{\alpha_V(1-6\alpha_V)}{2}\frac{\Delta\hat{\sigma}_{\min}^2}{1+\Delta\hat{\sigma}_{\min}^2}\left\|\widetilde{\boldsymbol{W}}_{Vk}\right\|^2 \\
&\quad - \alpha_V(1-2\alpha_V)\frac{\hat{\sigma}_{\min}^2}{1+\hat{\sigma}_{\min}^2}\left\|\widetilde{\boldsymbol{W}}_{Vk}\right\|^2 - \frac{\alpha_V(1+3\alpha_V)}{1+\Delta\hat{\sigma}_{\min}^2} L_m^2 \|\widetilde{\boldsymbol{x}}_k\|^4 - 4\Pi\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^4 \left\|\widetilde{\boldsymbol{W}}_k\right\|^4 \\
&\quad - (1-8(1-\alpha_{\mathrm{I}})^4)\Pi\Lambda^2\left\|\widetilde{\boldsymbol{W}}_k\right\|^4 + \varepsilon_1 - \alpha_{\boldsymbol{u}}\Sigma\frac{\hat{\sigma}_{\boldsymbol{u}\min}^2}{1+\hat{\sigma}_{\boldsymbol{u}\min}^2}\left\|\widetilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\| \\
&\quad + \alpha_{\boldsymbol{u}}\Sigma\left(1+\frac{5}{4\|\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}\|}\right)\nabla\sigma_{V\max}^2\left\|\widetilde{\boldsymbol{W}}_{Vk}\right\|^2 + \alpha_{\boldsymbol{u}}\Sigma\sigma_g^2\left(\frac{1}{4\|\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}\|} + \frac{\lambda_{\max}(\boldsymbol{R}^{-1})}{4(1+\sigma_{\boldsymbol{u}\min}^2)}\right)\left\|\widetilde{\boldsymbol{W}}_k\right\|^2 + \Sigma\bar{\varepsilon}_{T\mathrm{M}} \\
&\le -\left(1-(1+4\gamma\|\boldsymbol{lC}\|^2)(1+\chi_{\min}^2)\gamma\right)\|\widetilde{\boldsymbol{x}}_k\|^2 + \varepsilon_{\mathrm{CLM}} - \left\|\widetilde{\boldsymbol{W}}_k\right\|^2 \|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2 \\
&\quad - \frac{1}{2}(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\left\|\widetilde{\boldsymbol{W}}_k\right\|^2 - \frac{\alpha_V(1-6\alpha_V)}{2}\frac{\Delta\hat{\sigma}_{\min}^2}{1+\Delta\hat{\sigma}_{\min}^2}\left\|\widetilde{\boldsymbol{W}}_{Vk}\right\|^2 \\
&\quad - \frac{\alpha_V(1-2\alpha_V)}{2}\frac{\Delta\hat{\sigma}_{\min}^2}{1+\Delta\hat{\sigma}_{\min}^2}\left\|\widetilde{\boldsymbol{W}}_{Vk}\right\|^2 - 4\Pi\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^4\left\|\widetilde{\boldsymbol{W}}_k\right\|^4 - \frac{\alpha_V(1+3\alpha_V)}{1+\Delta\hat{\sigma}_{\min}^2} L_m^2\|\widetilde{\boldsymbol{x}}_k\|^4 \\
&\quad - (1-8(1-\alpha_{\mathrm{I}})^4)\Pi\Lambda^2\left\|\widetilde{\boldsymbol{W}}_k\right\|^4 - \alpha_{\boldsymbol{u}}\Sigma\frac{\hat{\sigma}_{\boldsymbol{u}\min}^2}{1+\hat{\sigma}_{\boldsymbol{u}\min}^2}\left\|\widetilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\|
\end{aligned}
$$

where $\varepsilon_{\mathrm{CLM}} = \varepsilon_{\mathrm{OM}} + \varepsilon_1 + \Sigma\bar{\varepsilon}_{T\mathrm{M}}$.

By using standard Lyapunov stability analysis, ΔL is less than zero outside a compact set as long as the following conditions hold:

$$\left\|\tilde{\boldsymbol{x}}_k\right\| > \min\left\{\begin{array}{l}\sqrt{\dfrac{\varepsilon_{\mathrm{CLM}}}{1-(1+4\gamma\left\|\boldsymbol{lC}\right\|^2)(1+\chi_{\min}^2)\gamma}},\\ \sqrt[4]{\dfrac{\varepsilon_{\mathrm{CLM}}}{\dfrac{\alpha_V(1+3\alpha_V)}{1+\Delta\hat{\sigma}_{\min}^2}L_m^2}}\end{array}\right\} \equiv b_{\tilde{\boldsymbol{x}}} \tag{A.20}$$

Or

$$\left\|\tilde{\boldsymbol{W}}_{Vk}\right\| > \sqrt{\frac{\varepsilon_{\mathrm{CLM}}}{\dfrac{\alpha_V(1-6\alpha_V)}{2}\dfrac{\Delta\hat{\sigma}_{\min}^2}{1+\Delta\hat{\sigma}_{\min}^2}+\alpha_V(1-2\alpha_V)\dfrac{\hat{\sigma}_{\min}^2}{1+\hat{\sigma}_{\min}^2}}} \equiv b_{\tilde{\boldsymbol{W}}_V} \tag{A.21}$$

Or

$$\left\|\tilde{\boldsymbol{W}}_k\right\| > \min\left\{\begin{array}{l}\sqrt{\dfrac{\varepsilon_{\mathrm{CLM}}}{\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\right\|^2+\dfrac{1}{2}(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda}},\\ \sqrt[4]{\dfrac{\varepsilon_{\mathrm{CLM}}}{4\Pi\chi_{\min}^4+(1-8(1-\alpha_{\mathrm{I}})^4)\Pi\Lambda^2}}\end{array}\right\} \equiv b_{\tilde{\boldsymbol{W}}} \tag{A.22}$$

Or

$$\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\| > \frac{\varepsilon_{\mathrm{CLM}}}{\alpha_{\boldsymbol{u}}\Sigma\dfrac{\hat{\sigma}_{\boldsymbol{u}\min}^2}{1+\hat{\sigma}_{\boldsymbol{u}\min}^2}} \equiv b_{\tilde{\boldsymbol{W}}_{\boldsymbol{u}}} \tag{A.23}$$

Eventually, the difference between the ideal optimal control and proposed near optimal control inputs is represented as

$$\begin{aligned}&\left\|\boldsymbol{u}^*(\boldsymbol{x}_k,k)-\hat{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)\right\|\\ &=\left\|\boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,k)+\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k,k)-\hat{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)\right\|\\ &=\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)+\boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\tilde{\sigma}_{\boldsymbol{u}}(\boldsymbol{x}_k,\hat{\boldsymbol{x}}_k,k)+\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k,k)\right\|\\ &\le b_{\tilde{\boldsymbol{W}}_{\boldsymbol{u}}}\sigma_{\boldsymbol{u}\mathrm{M}}+W_{\boldsymbol{u}\mathrm{M}}\left\|\tilde{\sigma}_{\boldsymbol{u}}(\boldsymbol{x}_k,\hat{\boldsymbol{x}}_k,k)\right\|+\varepsilon_{\boldsymbol{u}\mathrm{M}}\\ &\le b_{\tilde{\boldsymbol{W}}_{\boldsymbol{u}}}\sigma_{\boldsymbol{u}\mathrm{M}}+l_\sigma W_{\boldsymbol{u}\mathrm{M}}b_{\tilde{\boldsymbol{x}}}+\varepsilon_{\boldsymbol{u}\mathrm{M}}\equiv\varepsilon_{\boldsymbol{u}o}\end{aligned} \tag{A.24}$$

where l_σ is the Lipschitz constant of $\sigma_{\boldsymbol{u}}(\bullet)$, and $b_{\tilde{\boldsymbol{W}}_{\boldsymbol{u}}}$, $b_{\tilde{\boldsymbol{x}}}$ are given in (A.23) and (A.20).

V. FINITE-HORIZON NEAR OPTIMAL CONTROL OF QUANTIZED NONLINEAR DISCRETE-TIME SYSTEMS WITH INPUT CONSTRAINT USING NEURAL NETWORKS

Qiming Zhao, Hao Xu and S. Jagannathan

Abstract — In this work, the output feedback based finite-horizon near optimal regulation of uncertain quantized affine nonlinear discrete-time systems with control constraint is considered. First, the effect of control constraint is handled by a nonquadratic cost functional. Next, a neural network (NN)-based Luenberger observer is proposed to reconstruct both the system states and the control coefficient matrix so that a separate identifier is not needed. Then, approximate dynamic programming methodology with actor-critic structure is utilized to approximate the time-varying solution of the Hamilton-Jacobi-Bellman (HJB) by using NNs with constant weights and time-dependent activation functions. A new error term is defined and incorporated in the NN update law so that the terminal constraint error is also minimized over time. Finally, a novel dynamic quantizer for the control inputs with adaptive step-size is designed to eliminate the quantization error overtime thus overcoming the drawback of the traditional uniform quantizer. The proposed scheme functions in a forward-in-time manner without offline training phase. Lyapunov analysis is used to investigate the stability of the overall closed-loop system. Simulation results are given to show the effectiveness and feasibility of the proposed method.

1. INTRODUCTION

Actuator saturation is very common in practical control system applications due to physical limitations imposed on the controller and the plant. Control of systems with saturating actuators has been one of the focuses of many researchers for many years [1][2]. However, most of these approaches considered only stabilization whereas optimality is not considered. To address optimal control problem with input constraint, the author in [6] presented a general framework for the design of optimal control laws based on dynamic programing. It has been shown in [6] that the use of a non-quadratic functional can effectively tackle the input constraint while achieving optimality.

On the other hand, under practical applications, the interface between the plant and the controller is often connected via analog to digital (A/D) and digital to analog (D/A) devices which quantize the signals. As a result, the design of control systems with quantization effect has attracted a great deal of attention to the control researchers since quantization process is unavoidable in the computer-based control systems. However, quantization error never vanishes when the signals are processed by a traditional uniform quantizer [7]. In addition, in many practical situations, the system state vector is difficult or expensive to measure. Several traditional nonlinear observers, such as high-gain or sliding mode observers, have been developed during the past few decades [12][11]. However, the above mentioned observer designs [12][11] are applicable to systems which are expressed in a specific system structure such as Brunovisky-form, and require the system dynamics a priori.

On the other hand, the optimal regulation of nonlinear systems can be addressed either for infinite or finite fixed time scenario. The finite-horizon optimal regulation still remains unresolved due to the following reasons. First, the solution to the optimal control of finite-horizon nonlinear system becomes essentially time-varying thus complicating the analysis, in contrast with the infinite-horizon case, where the solution is time-independent. In addition, the terminal constraint is explicitly imposed in the cost function, whereas in the infinite-horizon case, the terminal constraint is normally ignored.

The past literature [16][17][18][19] provided some insights into solving finite-horizon optimal regulation of nonlinear system. However, the developed techniques functioned either backward-in-time [16][17] or require offline training [18][19] with iteration-based scheme which are not suitable for real-time implementation. Further, all the existing literature [16][17][18][19] considered only state feedback case without quantization effect. Therefore, the input-constraint finite-horizon optimal regulation scheme for nonlinear quantized systems, which can be implemented in an online and forward-in-time manner with completely unknown system dynamics and without using both state measurements and value and policy iterations, is yet to be developed.

Motivated by the aforementioned deficiencies, in this paper, an extended Luenberger observer is first proposed to estimate the system states as well as the control coefficient matrix. The actor-critic architecture is then utilized to generate the near optimal control policy wherein the value function is approximated by using the critic NN and the optimal policy is generated by using the approximated value function and the control coefficient matrix provided an initial admissible control is chosen. Finally, a novel dynamic quantizer is proposed to mitigate the effect of quantization error for the control inputs. Due to the presence of observer errors, the control policy will be near optimal.

To handle the time-varying nature of the solution to the HJB equation or value function, NNs with constant weights and time-varying activation functions are utilized. In addition, in contrast with [18] and [19], the control policy is updated once every sampling instant and hence value/policy iterations are not performed. An error term corresponding to the terminal constraint is defined and minimized overtime such that the terminal constraint can be properly satisfied. A novel update law for tuning the NN is developed such that the critic NN weights will be tuned not only by using Bellman error but also the terminal constraint errors. Finally, stability of our proposed design scheme is demonstrated by using Lyapunov stability analysis.

Therefore, the main contribution of the paper includes the development of a novel approach to solve the finite-horizon output feedback based near optimal control of uncertain quantized nonlinear discrete-time systems in affine form in an online and forward-in-time manner without utilizing value and/or policy iterations. A novel dynamic quantizer as well as an online observer is introduced for eliminating the quantization error and generating the state vector and control coefficient matrix so that an explicit need for an identifier is relaxed, Tuning laws for all the NNs are also derived. Lyapunov stability is also demonstrated.

The remainder of this paper is organized as follows. In section 2, background and formulation of finite-horizon optimal control problem for nonlinear quantized systems are given. Section 3 presents the main algorithm developed for the finite-horizon problem. In Section 4, simulation results are shown to verify the feasibility of proposed method. Conclusions are provided in section 5.

2. PROBLEM FORMULATION

In this paper, the finite-horizon optimal control of general affine quantized nonlinear discrete-time system is studied. Consider the system of the form

$$\begin{aligned} \boldsymbol{x}_{k+1} &= f(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_{qk} \\ \boldsymbol{y}_k &= \boldsymbol{C}\boldsymbol{x}_k \end{aligned} \tag{1}$$

where $\boldsymbol{x}_k \in \Omega_{\boldsymbol{x}} \subset \Re^n$ and $\boldsymbol{y}_k \in \Omega_y \subset \Re^p$ are the system states and outputs, respectively. $\boldsymbol{u}_{qk} = q_d(\boldsymbol{u}_k) \in \Omega_{\boldsymbol{u}} \subset \Re^m$ is the quantized control input vector, where $q_d(\bullet)$ is the dynamic quantizer defined later, $\boldsymbol{u}_k \in \boldsymbol{U} \subset \Re^m$, where $\boldsymbol{U} = \{\boldsymbol{u} = (u_1, u_2, \cdots, u_m) \in \Re^m :$ $a_i \le u_i \le b_i, i = 1,2,\cdots,m\}$ is the saturated control with a_i and b_i being the constant bounds [5], $f(\boldsymbol{x}_k) : \Re^n \to \Re^n$, $g(\boldsymbol{x}_k) : \Re^n \to \Re^{n\times m}$ are unknown nonlinear dynamics and $\boldsymbol{C} \in \Re^{p\times n}$ is the known output matrix. In addition, the input matrix $g(\boldsymbol{x}_k)$ is considered to be bounded such that $0 < \|g(\boldsymbol{x}_k)\| < g_{\mathrm{M}}$, where g_{M} is a positive constant. The general structure of the quantized nonlinear discrete-time system considered in this paper is illustrated in Figure 1.

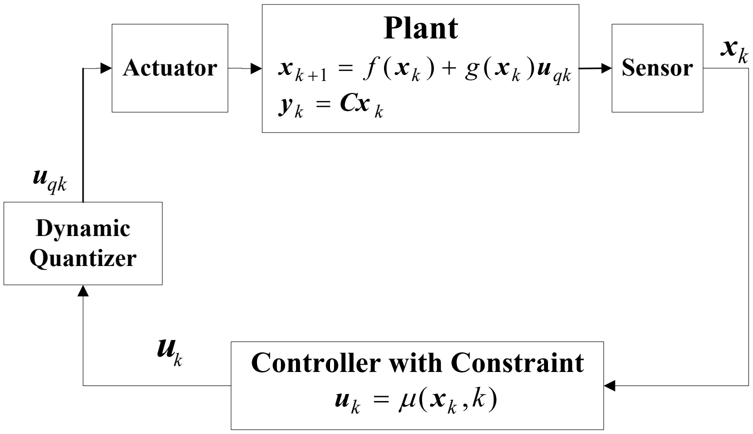

Actuator
Plant
$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_{qk}$
$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k$
Sensor
$\boldsymbol{x}_k$
$\boldsymbol{u}_{qk}$
Dynamic Quantizer
$\boldsymbol{u}_k$
Controller with Constraint
$\boldsymbol{u}_k = \mu(\boldsymbol{x}_k, k)$

Figure 1. Block diagram of the quantized system with input saturation

It is important to note that digital communication network is usually used to connect sensor, controller and actuator in practical scenario [13]. Due to limited communication bandwidth, system states and control inputs should be quantized before transmission [23]. In our previous work [24], state quantization has been considered. Therefore, control input quantization is considered here.

Assumption 1: The nonlinear system given in (1) is controllable and observable. Moreover, the system output, $\boldsymbol{y}_k \in \Omega_{\boldsymbol{y}}$, is measurable.

The objective of the control design is to determine a feedback control policy that minimizes the following time-varying cost function

$$V(\boldsymbol{x}_k, k) = \psi(\boldsymbol{x}_{\mathrm{N}}) + \sum_{i=k}^{\mathrm{N}-1} (\boldsymbol{Q}(\boldsymbol{x}_i, i) + W(\boldsymbol{u}_i)) \tag{2}$$

which is subjected to the system dynamics (1), $[k, \mathrm{N}]$ is the time interval of interest, $\psi(\boldsymbol{x}_{\mathrm{N}})$ is the terminal constraint that penalizes the terminal state $\boldsymbol{x}_{\mathrm{N}} \in \Omega_{\boldsymbol{x}}$, $\boldsymbol{Q}(\boldsymbol{x}_k, k) \in \Re$ is positive semi-definite function and $W(\boldsymbol{u}_k) \in \Re$ is positive definite. It should be noted that in the finite-horizon scenario, the control inputs can be time-varying, i.e., $\boldsymbol{u}_k = \mu(\boldsymbol{x}_k, k) \in \Omega_{\boldsymbol{u}}$.

Setting $k = \mathrm{N}$, the terminal constraint for the value function is given as

$$V(\boldsymbol{x}_{\mathrm{N}}, \mathrm{N}) = \psi(\boldsymbol{x}_{\mathrm{N}}) \tag{3}$$

For unconstrained control inputs, $W(\boldsymbol{u}_k)$ is generally taking the form $W(\boldsymbol{u}_k) = \boldsymbol{u}_k^{\mathrm{T}} \boldsymbol{R} \boldsymbol{u}_k$, with $\boldsymbol{R} \in \Re^{m \times m}$ a positive definite and symmetric weighting matrix. However, in this paper, to confront the actuator saturation, we employ a non-quadratic functional [6] as:

$$W(\boldsymbol{u}_k) = 2\int_0^{\boldsymbol{u}_k} \left(\boldsymbol{\varphi}^{-1}(\boldsymbol{v})\right)^{\mathrm{T}} \boldsymbol{R} d\boldsymbol{v} \tag{4}$$

with

$$\begin{aligned} \boldsymbol{\varphi}(\boldsymbol{v}) &= [\phi(v_1) \quad \cdots \quad \phi(v_m)]^{\mathrm{T}} \\ \boldsymbol{\varphi}^{-1}(\boldsymbol{u}_k) &= [\phi^{-1}(u_1(k)) \quad \cdots \quad \phi^{-1}(u_m(k))] \end{aligned} \tag{5}$$

where $\boldsymbol{v} \in \Omega_{\boldsymbol{v}} \subset \Re^m$, $\boldsymbol{\varphi} \in \Omega_{\boldsymbol{\varphi}} \subset \Re^m$, and $\boldsymbol{\varphi}(\bullet)$ is a bounded one-to-one function that belongs to C^p ($p \geq 1$). Define the notation $w(\boldsymbol{v}) = \boldsymbol{\varphi}^{-1}(\boldsymbol{v})\boldsymbol{R}$, and

$$\int_0^{\boldsymbol{u}_k} \boldsymbol{w}^{\mathrm{T}}(\boldsymbol{v}) d\boldsymbol{v} \equiv \int_0^{u_1(k)} w_1(v_1) dv_1 + \cdots + \int_0^{u_m(k)} w_m(v_m) dv_m \tag{6}$$

is a scalar, for $\boldsymbol{u}_k \in \Omega_{\boldsymbol{u}} \subset \Re^m$, $\boldsymbol{v} \in \Omega_{\boldsymbol{v}} \subset \Re^m$ and $w(\boldsymbol{v}) = [w_1 \quad \cdots \quad w_m] \in \Omega_w \subset \Re^m$.

Moreover, it is a monotonic odd function with its first derivative bounded by a constant U. An example is the hyperbolic tangent function $\boldsymbol{\varphi}(\bullet) = \tanh(\bullet)$. Note that $W(\boldsymbol{u}_k)$ is positive definite since $\boldsymbol{\varphi}^{-1}(\boldsymbol{u})$ is monotonic odd and $\boldsymbol{R}$ is positive definite.

By Bellman's principle of optimality [3][4], the optimal value function should satisfy the following HJB equation

$$\begin{aligned} V^*(\boldsymbol{x}_k, k) &= \min_{\boldsymbol{u}_k} \left\{ \boldsymbol{Q}(\boldsymbol{x}_k, k) + W(\boldsymbol{u}_k) + V^*(\boldsymbol{x}_{k+1}, k+1) \right\} \\ &= \min_{\boldsymbol{u}_k} \left\{ \boldsymbol{Q}(\boldsymbol{x}_k, k) + 2\int_0^{\boldsymbol{u}_k} \left(\boldsymbol{\varphi}^{-1}(\boldsymbol{v})\right)^{\mathrm{T}} \boldsymbol{R} d\boldsymbol{v} + V^*(\boldsymbol{x}_{k+1}, k+1) \right\} \end{aligned} \tag{7}$$

The optimal control policy $\boldsymbol{u}_k^* \in \Omega_{\boldsymbol{u}}$ that minimizes the value function $V^*(\boldsymbol{x}_k, k)$ is revealed to be

$$\begin{aligned} \boldsymbol{u}_k^* &= \arg\min_{\boldsymbol{u}_k} \left\{ \boldsymbol{Q}(\boldsymbol{x}_k, k) + 2\int_0^{\boldsymbol{u}_k} \left(\boldsymbol{\varphi}^{-1}(\boldsymbol{v})\right)^{\mathrm{T}} \boldsymbol{R} d\boldsymbol{v} + V^*(\boldsymbol{x}_{k+1}, k+1) \right\} \\ &= -\boldsymbol{\varphi}\left(\frac{1}{2} \boldsymbol{R}^{-1} g^{\mathrm{T}}(\boldsymbol{x}_k) \frac{\partial V^*(\boldsymbol{x}_{k+1}, k+1)}{\partial \boldsymbol{x}_{k+1}} \right) \end{aligned} \tag{8}$$

It is clear from (8) that the optimal control policy cannot be obtained for the nonlinear discrete-time system even with available system state vector due to the dependency on the future state vector $\boldsymbol{x}_{k+1} \in \Omega_{\boldsymbol{x}}$. To avoid this drawback and relax the requirement for system dynamics, iteration-based schemes are normally utilized by using NNs with offline-training [15]. However, iteration-based schemes are not preferable for hardware implementation since the number of iterations to ensure the stability cannot be easily determined [13]. Moreover, the iterative approaches cannot be implemented when the dynamics of the system are completely unknown, since at least the control coefficient matrix $g(\boldsymbol{x}_k)$ is required to generate the control policy [14]. Therefore, in this work, a solution is found with unavailable system states and completely unknown system dynamics without utilizing the iterative approach and in the presence of quantization effect, as will be given in the next section.

Finally, to take into account the quantization effect on the control inputs, consider the uniform quantizer with finite number of bits shown in Figure 2. Let z be the signal to be quantized and M be the quantization range for the quantizer. If z does not belong to the quantization range, the quantizer saturates. Let e be the quantization error, it is assumed that the following two conditions hold [10]:

$$\begin{aligned} &1.\,\text{if} \quad |z| \le \mathrm{M}, \quad \text{then} \;\; e = |q(z) - z| \le \Delta/2 \\ &2.\,\text{if} \quad |z| > \mathrm{M}, \quad \text{then} \;\; |q(z)| > \mathrm{M} - \Delta/2 \end{aligned} \tag{9}$$

where $q(z) = \Delta \cdot \left(\lfloor z/\Delta \rfloor + 1/2 \right)$ is a nonlinear mapping that represents a general uniform quantizer representation with the step-size Δ defined as $\Delta = \mathrm{M}/2^{\mathrm{R}}$ with R being the number of bits of the quantizer.

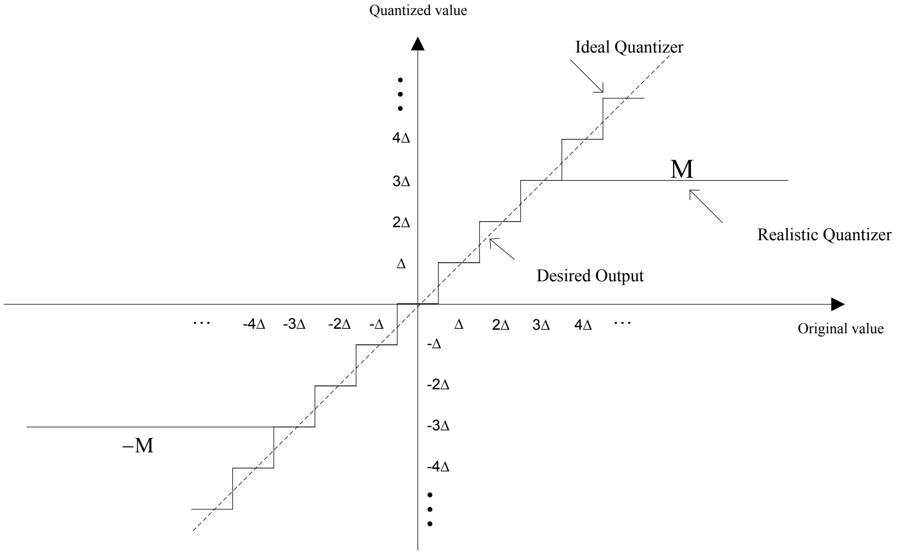


Figure 2. Ideal and realistic quantizer

In addition, theoretically, when the number of bits of the quantizer approaches infinity the quantization error will reduce to zero and hence infinite precision of the quantizer can be achieved. In the realistic scenario, however, both the quantization range and the number of bits cannot be arbitrarily large. To circumvent these drawbacks, a dynamic quantizer scheme is proposed in this paper in the form similar to [10] as

$$z_q = q_d(z) = \mu q(z/\mu) \tag{10}$$

where μ is a scaling factor.

3. FINITE-HORIZON NEAR OPTIMAL REGULATOR DESIGN USING OUTPUT FEEDBACK WITH CONTROL CONSTRAINT

In this section, the output feedback-based finite-horizon near optimal regulation scheme for uncertain quantized nonlinear discrete-time systems with input constraint is addressed. First, due to the unavailability of the system states and uncertain system dynamics, an extended version of Luenberger observer is proposed to reconstruct both the system states and control coefficient matrix in an online manner.

Thus the proposed observer design relaxes the need for an explicit identifier. Next, the approximate dynamic programming methodology is utilized to approximate the time-varying value function with actor-critic structure, while both NNs are represented by constant weights and time-varying activation functions. Furthermore, an error term corresponding to the terminal constraint is defined and minimized overtime so that the terminal constraint can be properly satisfied. Finally, a novel dynamic quantizer is proposed to reduce the quantization error overtime. The stability of the closed-loop system is demonstrated by Lyapunov theory to show that the parameter estimation remains bounded as the system evolves provided an initial admissible control input is chosen.

3.1 OBSERVER DESIGN

The system dynamics (1) can be reformulated as

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{A}\boldsymbol{x}_k + F(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_{qk} \\ \boldsymbol{y}_k &= \boldsymbol{C}\boldsymbol{x}_k \end{aligned} \tag{11}$$

where $\boldsymbol{A}$ is a Hurwitz matrix such that $(\boldsymbol{A},\boldsymbol{C})$ is observable and $F(\boldsymbol{x}_k) = f(\boldsymbol{x}_k) - \boldsymbol{A}\boldsymbol{x}_k$.

A NN has been proven to be an effective method in the estimation and control of nonlinear systems due to its online learning capability [21]. According to the universal approximation property [22], the system states can be represented by using NN on a compact set Ω as

$$\begin{aligned}\boldsymbol{x}_{k+1} &= \boldsymbol{A}\boldsymbol{x}_k + F(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_{qk} \\ &= \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{W}_F^{\mathrm{T}}\sigma_F(\boldsymbol{x}_k) + \boldsymbol{W}_g^{\mathrm{T}}\sigma_g(\boldsymbol{x}_k)\boldsymbol{u}_{qk} + \boldsymbol{\varepsilon}_{Fk} + \boldsymbol{\varepsilon}_{gk}\boldsymbol{u}_k \\ &= \boldsymbol{A}\boldsymbol{x}_k + \begin{bmatrix}\boldsymbol{W}_F \\ \boldsymbol{W}_g\end{bmatrix}^{\mathrm{T}} \begin{bmatrix}\sigma_F(\boldsymbol{x}_k) & \boldsymbol{0} \\ \boldsymbol{0} & \sigma_g(\boldsymbol{x}_k)\end{bmatrix} \begin{bmatrix}1 \\ \boldsymbol{u}_{qk}\end{bmatrix} + [\boldsymbol{\varepsilon}_{Fk} \quad \boldsymbol{\varepsilon}_{gk}] \begin{bmatrix}1 \\ \boldsymbol{u}_{qk}\end{bmatrix} \\ &= \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{W}^{\mathrm{T}}\sigma(\boldsymbol{x}_k)\bar{\boldsymbol{u}}_{qk} + \bar{\boldsymbol{\varepsilon}}_k\end{aligned} \tag{12}$$

where $\boldsymbol{W} = \begin{bmatrix}\boldsymbol{W}_F \\ \boldsymbol{W}_g\end{bmatrix} \in \Re^{L\times n}$, $\sigma(\boldsymbol{x}_k) = \begin{bmatrix}\sigma_F(\boldsymbol{x}_k) & \boldsymbol{0} \\ \boldsymbol{0} & \sigma_g(\boldsymbol{x}_k)\end{bmatrix} \in \Re^{L\times(1+m)}$, $\bar{\boldsymbol{u}}_{qk} = \begin{bmatrix}1 \\ \boldsymbol{u}_{qk}\end{bmatrix} \in \Re^{(1+m)}$ and $\bar{\boldsymbol{\varepsilon}}_k = [\boldsymbol{\varepsilon}_{Fk} \quad \boldsymbol{\varepsilon}_{gk}]\bar{\boldsymbol{u}}_{qk} \in \Re^n$, with L being the number of hidden neurons. In addition, the target NN weights, activation function and reconstruction error are assumed to be upper bounded by $\|\boldsymbol{W}\| \le W_{\mathrm{M}}$, $\|\sigma(\boldsymbol{x}_k)\| \le \sigma_{\mathrm{M}}$ and $\|\bar{\boldsymbol{\varepsilon}}_k\| \le \bar{\varepsilon}_{\mathrm{M}}$, where W_{M}, σ_{M} and $\bar{\varepsilon}_{\mathrm{M}}$ are positive constants. Then, the system states $\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + F(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_{qk}$ can be identified by updating the target NN weight matrix $\boldsymbol{W}$.

Since the true system states are unavailable for the controller, we propose the following extended Luenberger observer described by

$$\begin{aligned}\hat{\boldsymbol{x}}_{k+1} &= \boldsymbol{A}\hat{\boldsymbol{x}}_k + \hat{\boldsymbol{W}}_k^{\mathrm{T}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk} + \boldsymbol{L}(\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k) \\ \hat{\boldsymbol{y}}_k &= \boldsymbol{C}\hat{\boldsymbol{x}}_k\end{aligned} \tag{13}$$

where $\hat{\boldsymbol{W}}_k$ is the estimated value of the target NN weights $\boldsymbol{W}$, $\hat{\boldsymbol{x}}_k$ is the reconstructed system state vector, $\hat{\boldsymbol{y}}_k$ is the estimated output vector and $\boldsymbol{L} \in \Re^{n\times p}$ is the observer gain selected by the designer, respectively. Then the state estimation error can be express as

$$\begin{aligned}\tilde{\boldsymbol{x}}_{k+1} &= \boldsymbol{x}_{k+1} - \hat{\boldsymbol{x}}_{k+1} \\ &= \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{W}^{\mathrm{T}}\sigma(\boldsymbol{x}_k)\bar{\boldsymbol{u}}_{qk} + \bar{\boldsymbol{\varepsilon}}_k - (\boldsymbol{A}\hat{\boldsymbol{x}}_k + \hat{\boldsymbol{W}}_{k+1}^{\mathrm{T}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk} + \boldsymbol{L}(\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k)) \\ &= \boldsymbol{A}_c\tilde{\boldsymbol{x}}_k + \tilde{\boldsymbol{W}}_k^{\mathrm{T}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk} + \boldsymbol{W}^{\mathrm{T}}\tilde{\sigma}(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk} + \bar{\boldsymbol{\varepsilon}}_k \\ &= \boldsymbol{A}_c\tilde{\boldsymbol{x}}_k + \tilde{\boldsymbol{W}}_k^{\mathrm{T}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk} + \boldsymbol{\varepsilon}_{Ok}\end{aligned} \tag{14}$$

where $\boldsymbol{A}_c = \boldsymbol{A} - \boldsymbol{LC}$ is the closed-loop matrix, $\tilde{\boldsymbol{W}}_k = \boldsymbol{W} - \hat{\boldsymbol{W}}_k$ is the NN weights estimation error, $\tilde{\sigma}(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k) = \sigma(\boldsymbol{x}_k) - \sigma(\hat{\boldsymbol{x}}_k)$ and $\bar{\boldsymbol{\varepsilon}}_{Ok} = \boldsymbol{W}^{\mathrm{T}}\tilde{\sigma}(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk} + \bar{\boldsymbol{\varepsilon}}_k$ are bounded terms due to the bounded values of ideal NN weights, activation functions and reconstruction errors.

Remark 1: It should be noted that the proposed observer (13) has two essential purposes. First, the observer presented in (13) generates the reconstructed system states for the controller design. Second, the structure of the observer is novel in that it also generates the control coefficient matrix $g(\boldsymbol{x}_k)$, which will be viewed as a NN-based identifier. Thus, the NN-based observer (13) can be viewed both as a standard observer and an identifier whose estimate of the control coefficient matrix $g(\boldsymbol{x}_k)$, is utilized in the near optimal control design shown in the next section.

Now select the tuning law for the NN weights as

$$\hat{\boldsymbol{W}}_{k+1} = (1-\alpha_{\mathrm{I}})\hat{\boldsymbol{W}}_k + \beta_{\mathrm{I}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\tilde{\boldsymbol{y}}_{k+1}^{\mathrm{T}}\boldsymbol{l}^{\mathrm{T}} \tag{15}$$

where α_{I}, β_{I} are the tuning parameters, $\tilde{\boldsymbol{y}}_{k+1} = \boldsymbol{y}_{k+1} - \hat{\boldsymbol{y}}_{k+1}$ is the output error and $\boldsymbol{l} \in \Re^{n\times p}$ is selected to match the dimension.

Hence, the NN weight estimation error dynamics, by recalling from (14), are revealed to be

$$\begin{aligned}
\tilde{\boldsymbol{W}}_{k+1} &= \boldsymbol{W} - \hat{\boldsymbol{W}}_{k+1} \\
&= (1-\alpha_{\mathrm{I}})\tilde{\boldsymbol{W}}_k + \alpha_{\mathrm{I}}\boldsymbol{W} - \beta_{\mathrm{I}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\tilde{\boldsymbol{y}}_{k+1}^{\mathrm{T}}\boldsymbol{l}^{\mathrm{T}} \\
&= (1-\alpha_{\mathrm{I}})\tilde{\boldsymbol{W}}_k + \alpha_{\mathrm{I}}\boldsymbol{W} - \beta_{\mathrm{I}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\tilde{\boldsymbol{x}}_k^{\mathrm{T}}\boldsymbol{A}_c^{\mathrm{T}}\boldsymbol{C}^{\mathrm{T}}\boldsymbol{l}^{\mathrm{T}} \\
&\quad - \beta_{\mathrm{I}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\bar{\boldsymbol{u}}_{qk}^{\mathrm{T}}\sigma^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)\tilde{\boldsymbol{W}}_k\boldsymbol{C}^{\mathrm{T}}\boldsymbol{l}^{\mathrm{T}} - \beta_{\mathrm{I}}\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\bar{\boldsymbol{\varepsilon}}_{Ok}^{\mathrm{T}}\boldsymbol{C}^{\mathrm{T}}\boldsymbol{l}^{\mathrm{T}}
\end{aligned} \tag{16}$$

Next, the boundedness of the NN weights estimation error $\tilde{\boldsymbol{W}}_k$ will be demonstrated in Theorem 1. Before proceeding, the following definitions are required.

Definition 1 [22]: An equilibrium point $\boldsymbol{x}_e$ is said to be uniformly ultimately bounded (UUB) if there exists a compact set $\Omega_{\boldsymbol{x}} \subset \Re^n$ so that for all initial state $\boldsymbol{x}_0 \in \Omega_{\boldsymbol{x}}$, there exists a bound B and a time $T(B, \boldsymbol{x}_0)$ such that $\|\boldsymbol{x}_k - \boldsymbol{x}_e\| \le B$ for all $k \ge k_0 + T$.

Definition 2 [15]: Let $\Omega_{\boldsymbol{u}}$ denote the set of admissible control. A control function $\boldsymbol{u} : \Re^n \to \Re^m$ is defined to be admissible if the following is true:

$\boldsymbol{u}$ is continuous on $\Omega_{\boldsymbol{u}}$;

$\boldsymbol{u}(\boldsymbol{x})\big|_{\boldsymbol{x}=0} = 0$;

$\boldsymbol{u}(\boldsymbol{x})$ stabilize the system (1) on $\Omega_{\boldsymbol{x}}$;

$J(\boldsymbol{x}(0), \boldsymbol{u}) < \infty, \forall \boldsymbol{x}(0) \in \Omega_{\boldsymbol{x}}$.

Since the design scheme is similar to policy iteration, we need to solve a fixed-point equation rather than recursive equation. The initial admissible control guarantees the solution of the fixed-potion equation exists, thus the approximation process can be effectively done by our proposed scheme.

Theorem 1 (*Boundedness of the observer error*): Let the nonlinear system (1) be controllable and observable while the system output, $\boldsymbol{y}_k \in \Omega_{\boldsymbol{y}}$, be measurable. Let the initial NN observer weights $\hat{\boldsymbol{W}}_k$ be selected within compact set Ω_{OB} which contains the ideal weights $\boldsymbol{W}$. Given the admissible control input, $\boldsymbol{u}(0) \in \Omega_{\boldsymbol{u}}$ and let the proposed observer be given as in (13) and the update law for tuning the NN weights be given by (15). Let the control signal be persistently exciting (PE). Then, there exist positive constants α_{I} and β_{I} satisfying $\frac{2-\sqrt{2}}{2} < \alpha_{\mathrm{I}} < 1$ and $0 < \beta_{\mathrm{I}} < \frac{2(1-\alpha_{\mathrm{I}})\lambda_{\min}(\boldsymbol{l}\boldsymbol{C})}{\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2 + 1}$, with $\lambda_{\min}$ denoting the minimum eigenvalue, such that the observer error $\tilde{\boldsymbol{x}}_k$ and the NN weights estimation errors $\tilde{\boldsymbol{W}}_k$ are all UUB, with the bounds given by (A.6) and (A.7).

Proof: See Appendix.

3.2 ADP BASED NEAR OPTIAML REGULATOR DESIGN

According to the universal approximation property of NNs [22] and actor-critic methodology, the value function and control inputs can be represented by a "critic" NN and an "actor" NN, respectively, as

$$V(\boldsymbol{x}_k, k) = \boldsymbol{W}_V^{\mathrm{T}} \sigma_V(\boldsymbol{x}_k, k) + \varepsilon_V(\boldsymbol{x}_k, k) \tag{17}$$

and

$$\boldsymbol{u}(\boldsymbol{x}_k, k) = \boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}} \sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, k) + \varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k, k) \tag{18}$$

where $\boldsymbol{W}_V \in \Re^{L_V}$ and $\boldsymbol{W}_{\boldsymbol{u}} \in \Re^{L_{\boldsymbol{u}} \times m}$ are the constant target NN weights, with L_V and $L_{\boldsymbol{u}}$ the number of hidden neurons, $\sigma_V(\boldsymbol{x}_k, k) \in \Re^{L_V}$ and $\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, k) \in \Re^{L_{\boldsymbol{u}}}$ are the time-varying activation functions, $\varepsilon_V(\boldsymbol{x}_k, k)$ and $\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k, k)$ are the NN reconstruction errors for the critic and action network, respectively. Under standard assumption, the target NN weights are considered bounded above such that $\left\|\boldsymbol{W}_V\right\| \le W_{V\mathrm{M}}$ and $\left\|\boldsymbol{W}_{\boldsymbol{u}}\right\| \le W_{\boldsymbol{u}\mathrm{M}}$, respectively, where both $W_{V\mathrm{M}}$ and $W_{\boldsymbol{u}\mathrm{M}}$ are positive constants [22].

The NN activation functions and the reconstruction errors are also assumed to be bounded above such that $\left\|\sigma_V(\boldsymbol{x}_k, k)\right\| \le \sigma_{V\mathrm{M}}$, $\left\|\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, k)\right\| \le \sigma_{\boldsymbol{u}\mathrm{M}}$, $\left|\varepsilon_V(\boldsymbol{x}_k, k)\right| \le \varepsilon_{V\mathrm{M}}$ and

$\left|\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k,k)\right| \le \varepsilon_{\boldsymbol{u}\mathrm{M}}$, with $\sigma_{V\mathrm{M}}$, $\sigma_{\boldsymbol{u}\mathrm{M}}$, $\varepsilon_{V\mathrm{M}}$ and $\varepsilon_{\boldsymbol{u}\mathrm{M}}$ all positive constants [22]. In addition, in this work, the gradient of the reconstruction error is also assumed to be bounded above such as $\left\|\partial\varepsilon_{V,k}/\partial\boldsymbol{x}_{k+1}\right\| \le \varepsilon_{V\mathrm{M}}^{'}$, with $\varepsilon_{V\mathrm{M}}^{'}$ a positive constant [14]. The terminal constraint of the value function is defined, similar to (18), as

$$V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N}) = \boldsymbol{W}_V^{\mathrm{T}}\sigma_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N}) + \varepsilon_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N}) \tag{19}$$

with $\sigma_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N})$ and $\varepsilon_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N})$ represent the activation and construction error corresponding to the terminal state $\boldsymbol{x}_{\mathrm{N}}$.

3.2.1 Value Function Approximation. According to (17), the time-varying value function $V(\boldsymbol{x}_k,k)$ can be approximated by using a NN as

$$\hat{V}(\hat{\boldsymbol{x}}_k,k) = \hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_k,k) \tag{20}$$

where $\hat{V}(\hat{\boldsymbol{x}}_k,k)$ represents the approximated value function at time step k. $\hat{\boldsymbol{W}}_{Vk}$ and $\sigma_V(\hat{\boldsymbol{x}}_k,k)$ are the estimated critic NN weights and "reconstructed" activation function with the estimated states vector $\hat{\boldsymbol{x}}_k$ as the inputs.

The value function at the terminal stage can be represented by

$$\hat{V}(\boldsymbol{x}_{\mathrm{N}},\mathrm{N}) = \hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N}) \tag{21}$$

where $\hat{\boldsymbol{x}}_{\mathrm{N}}$ is an estimation of the terminal state. It should be noted that since the true value of $\boldsymbol{x}_{\mathrm{N}}$ is not known, $\hat{\boldsymbol{x}}_{\mathrm{N}}$ can be considered to be an "estimate" of $\boldsymbol{x}_{\mathrm{N}}$ and can be chosen randomly as long as $\hat{\boldsymbol{x}}_{\mathrm{N}}$ lies within a region for a stabilizing control policy [15][18].

To ensure optimality, the Bellman equation should hold along the system trajectory. According to the principle of optimality, the true Bellman equation is given by

$$\boldsymbol{Q}(\boldsymbol{x}_k,k)+W(\boldsymbol{u}_k^*)+V^*(\boldsymbol{x}_{k+1},k+1)-V^*(\boldsymbol{x}_k,k)=0 \tag{22}$$

However, (22) no longer holds when the reconstructed system state vector $\hat{\boldsymbol{x}}_k$ and NN approximation are considered. Therefore, with estimated values, the Bellman equation (22) becomes

$$\begin{aligned} e_{\mathrm{B},k} &= \boldsymbol{Q}(\hat{\boldsymbol{x}}_k,k)+W(\boldsymbol{u}_k)+\hat{V}(\hat{\boldsymbol{x}}_{k+1},k+1)-\hat{V}(\hat{\boldsymbol{x}}_k,k) \\ &= \boldsymbol{Q}(\hat{\boldsymbol{x}}_k,k)+W(\boldsymbol{u}_k)+\hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{k+1},k+1)-\hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_k,k) \\ &= \boldsymbol{Q}(\hat{\boldsymbol{x}}_k,k)+W(\boldsymbol{u}_k)-\hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k) \end{aligned} \tag{23}$$

where $e_{\mathrm{B},k}$ is the Bellman equation residual error *along the system trajectory*, and $\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k)=\sigma_V(\hat{\boldsymbol{x}}_k,k)-\sigma_V(\hat{\boldsymbol{x}}_{k+1},k+1)$.

Next, using (21), define an additional error term corresponding to the terminal constraint as

$$e_{\mathrm{N},k}=\psi(\boldsymbol{x}_{\mathrm{N}})-\hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N}) \tag{24}$$

The objective of the optimal control design is thus to minimize the Bellman equation residual error $e_{\mathrm{B},k}$ as well as the terminal constraint error $e_{\mathrm{N},k}$, so that the optimality can be achieved and the terminal constraint can be properly satisfied. Next, based on gradient descent approach, the update law for critic NN can be defined as

$$\begin{aligned} \hat{\boldsymbol{W}}_{Vk+1} = \hat{\boldsymbol{W}}_{Vk} &+ \alpha_V \frac{(\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k)+\sigma_V{}'(\hat{\boldsymbol{x}}_{k+1},k+1)+2\sigma_{V\mathrm{M}}\boldsymbol{B}_l)e_{\mathrm{B},k}}{1+\left\|\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k)+\sigma_V{}'(\hat{\boldsymbol{x}}_{k+1},k+1)+2\sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} \\ &-\alpha_V\frac{\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})e_{\mathrm{N},k}}{1+\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})} \end{aligned} \tag{25}$$

where α_V is a design parameter, $\sigma_V{}'(\hat{\boldsymbol{x}}_{k+1},k+1)$ is the gradient of $\sigma_V(\hat{\boldsymbol{x}}_{k+1},k+1)$ and $\boldsymbol{B}_l\in\Re^{L_V}$ is a constant vector.

Now define $\tilde{\boldsymbol{W}}_{Vk} = \boldsymbol{W}_V - \hat{\boldsymbol{W}}_{Vk}$. The standard Bellman equation (22) can be expressed by NN representation as

$$0 = \boldsymbol{Q}(\boldsymbol{x}_k, k) + W(\boldsymbol{u}_k^*) - \boldsymbol{W}_V^{\mathrm{T}} \Delta\sigma_V(\boldsymbol{x}_k, k) - \Delta\varepsilon_V(\boldsymbol{x}_k, k) \tag{26}$$

where $\Delta\sigma_V(\boldsymbol{x}_k, k) = \sigma_V(\boldsymbol{x}_k, k) - \sigma_V(\boldsymbol{x}_{k+1}, k+1)$ and $\Delta\varepsilon_V(\boldsymbol{x}_k, k) = \varepsilon_V(\boldsymbol{x}_k, k) - \varepsilon_V(\boldsymbol{x}_{k+1}, k+1)$.

Subtracting (23) from (26), $e_{\mathrm{B},k}$ can be further derived as

$$\begin{aligned}
e_{\mathrm{B},k} &= \boldsymbol{Q}(\hat{\boldsymbol{x}}_k, k) + W(\boldsymbol{u}_k) - \hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}} \Delta\sigma_V(\hat{\boldsymbol{x}}_k, k) + \hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}} \Delta\sigma_V(f(\hat{\boldsymbol{x}}_k) + g(\hat{\boldsymbol{x}}_k)\boldsymbol{u}_k, k+1) \\
&\quad - \hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}} \Delta\sigma_V(f(\hat{\boldsymbol{x}}_k) + g(\hat{\boldsymbol{x}}_k)\boldsymbol{u}_k, k+1) \\
&\quad - \boldsymbol{Q}(\boldsymbol{x}_k, k) - W(\boldsymbol{u}_k^*) + \boldsymbol{W}_V^{\mathrm{T}} \Delta\sigma_V(\boldsymbol{x}_k, k) + \Delta\varepsilon_V(\boldsymbol{x}_k, k) \\
&= \boldsymbol{Q}(\hat{\boldsymbol{x}}_k, k) + W(\boldsymbol{u}_k) - \boldsymbol{Q}(\boldsymbol{x}_k, k) - W(\boldsymbol{u}_k^*) - \hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}} \Delta\sigma_V(\hat{\boldsymbol{x}}_k, k) + \boldsymbol{W}_V^{\mathrm{T}} \Delta\sigma_V(\hat{\boldsymbol{x}}_k, k) \\
&\quad - \boldsymbol{W}_V^{\mathrm{T}} \Delta\sigma_V(\hat{\boldsymbol{x}}_k, k) + \boldsymbol{W}_V^{\mathrm{T}} \Delta\sigma_V(\boldsymbol{x}_k, k) + \Delta\varepsilon_V(\boldsymbol{x}_k, k) \\
&\quad + \hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}} \Delta\sigma_V(f(\hat{\boldsymbol{x}}_k) + g(\hat{\boldsymbol{x}}_k)\boldsymbol{u}_k, k+1) - \hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}} \Delta\sigma_V(f(\hat{\boldsymbol{x}}_k) + g(\hat{\boldsymbol{x}}_k)\boldsymbol{u}_k, k+1)
\end{aligned}$$

$$\begin{aligned}
&\le L_Q \left\|\tilde{\boldsymbol{x}}_k\right\|^2 + 2\int_{\boldsymbol{u}_k^*}^{\boldsymbol{u}_k} \left(\boldsymbol{\varphi}^{-1}(\boldsymbol{v})\right)^{\mathrm{T}} \boldsymbol{R} d\boldsymbol{v} + \tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}} \Delta\sigma_V(\hat{\boldsymbol{x}}_k, k) + W_{V\mathrm{M}} L_{\sigma_V} \left\|e_{\boldsymbol{u}k}\right\| + \tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}} \sigma_{V\mathrm{M}} \boldsymbol{B}_l \operatorname{sgn}(\tilde{\boldsymbol{W}}_{Vk}) \\
&\quad + \boldsymbol{W}_V^{\mathrm{T}} \Delta\tilde{\sigma}_V(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k, k) + \Delta\varepsilon_V(\boldsymbol{x}_k, k) \\
&\le (L_Q + W_{V\mathrm{M}} L_{\sigma_V}) \left\|\tilde{\boldsymbol{x}}_k\right\|^2 + 2R \max(\boldsymbol{u}_k, \boldsymbol{u}_k^*) \tilde{\boldsymbol{u}}_k + \tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}} \Delta\sigma_V(\hat{\boldsymbol{x}}_k, k) \\
&\quad + \boldsymbol{W}_V^{\mathrm{T}} \Delta\tilde{\sigma}_V(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k, k) + \Delta\varepsilon_V(\boldsymbol{x}_k, k) + 2W_{V\mathrm{M}} \sigma_{V\mathrm{M}} + \tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}} \sigma_{V\mathrm{M}} \boldsymbol{B}_l \operatorname{sgn}(\tilde{\boldsymbol{W}}_{Vk}) \\
&\le (L_Q + W_{V\mathrm{M}} L_{\sigma_V}) \left\|\tilde{\boldsymbol{x}}_k\right\|^2 + 2R u_{\mathrm{M}} \left\|\tilde{\boldsymbol{u}}_k\right\| + \tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}} \Delta\sigma_V(\hat{\boldsymbol{x}}_k, k) \\
&\quad + 2W_{V\mathrm{M}} \sigma_{V\mathrm{M}} + \tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}} \sigma_{V\mathrm{M}} \boldsymbol{B}_l \operatorname{sgn}(\tilde{\boldsymbol{W}}_{Vk}) + \Delta\varepsilon_{VB}(\boldsymbol{x}_k, k)
\end{aligned} \tag{27}$$

where L_Q is a positive Lipschitz constant for $\boldsymbol{Q}(\bullet, k)$ due to the selected quadratic form in system states and L_{σ_V} is a positive Lipschitz constant for $\sigma_V(\bullet, k)$. In addition, $\Delta\tilde{\sigma}_V(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k, k) = \Delta\sigma_V(\boldsymbol{x}_k, k) - \Delta\sigma_V(\hat{\boldsymbol{x}}_k, k)$ and $\Delta\varepsilon_{VB}(\boldsymbol{x}_k, k) = \boldsymbol{W}_V^{\mathrm{T}} \Delta\tilde{\sigma}_V(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k, k) + \Delta\varepsilon_V(\boldsymbol{x}_k, k)$ are all bounded terms due to the boundedness of ideal NN weights, activation functions and reconstruction errors.

Recalling from (19), the terminal constraint error $e_{\mathrm{N},k}$ can be further expressed as

$$\begin{aligned}
e_{\mathrm{N},k} &= \psi(\boldsymbol{x}_{\mathrm{N}}) - \hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N}) \\
&= \boldsymbol{W}_V^{\mathrm{T}}\sigma_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N}) + \varepsilon_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N}) - \hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N}) \\
&= \boldsymbol{W}_V^{\mathrm{T}}\sigma_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N}) - \boldsymbol{W}_V^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N}) + \varepsilon_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N}) \\
&\quad + \boldsymbol{W}_V^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N}) - \hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N}) \\
&= \tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N}) + \boldsymbol{W}_V^{\mathrm{T}}\tilde{\sigma}_V(\boldsymbol{x}_{\mathrm{N}},\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N}) + \varepsilon_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N}) \\
&= \tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N}) + \varepsilon_{V\mathrm{N}}
\end{aligned} \tag{28}$$

where $\tilde{\sigma}_V(\boldsymbol{x}_{\mathrm{N}},\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N}) = \sigma_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N}) - \sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})$ and $\varepsilon_{V\mathrm{N}} = \boldsymbol{W}_V^{\mathrm{T}}\tilde{\sigma}_V(\boldsymbol{x}_{\mathrm{N}},\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N}) + \varepsilon_V(\boldsymbol{x}_{\mathrm{N}},\mathrm{N})$ are bounded due to the bounded ideal NN weights, activation function and reconstruction errors.

Finally, the error dynamics for critic NN weights are revealed to be

$$\begin{aligned}
\tilde{\boldsymbol{W}}_{Vk+1} = \tilde{\boldsymbol{W}}_{Vk} &- \alpha_V \frac{(\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k) + \sigma_V{}'(\hat{\boldsymbol{x}}_{k+1},k+1) + 2\sigma_{V\mathrm{M}}\boldsymbol{B}_l)e_{\mathrm{B},k}}{1 + \left\|\Delta\sigma_V(\hat{\boldsymbol{x}}_k,k) + \sigma_V{}'(\hat{\boldsymbol{x}}_{k+1},k+1) + 2\sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} \\
&- \alpha_V \frac{\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})e_{\mathrm{N},k}}{1 + \sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}},\mathrm{N})}
\end{aligned} \tag{29}$$

Next, the boundedness of the critic NN weights will be demonstrated, as shown in the following theorem.

Theorem 2 (*Boundedness of the critic NN weights*): Let the nonlinear system (1) be controllable and observable while the system output, $\boldsymbol{y}_k \in \Omega_{\boldsymbol{y}}$, be measurable. Let the initial critic NN weights $\hat{\boldsymbol{W}}_{Vk}$ be selected within compact set Ω_V which contains the ideal weights $\boldsymbol{W}_V$. Let $\boldsymbol{u}(0) \in \Omega_{\boldsymbol{u}}$ be an initial admissible control input for the system (1). Let the value function be approximated by a critic NN and the tuning law be given by (25). Then, under the assumption stated in this paper, there exists a positive constant α_V

satisfying $0 < \alpha_V < 1/6$ such that the critic NN weights estimation error $\tilde{\boldsymbol{W}}_{Vk}$ is UUB with a computable bound $b_{\tilde{W}_V}$ given in (A.16).

Proof: See Appendix.

3.2.2 Control Input Approximation. In this subsection, the near optimal control policy is obtained such that the estimated value function (20) is minimized. Recalling (18), the estimation of the control inputs by using NN can be represented as

$$\boldsymbol{u}(\hat{\boldsymbol{x}}_k, k) = \hat{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}} \sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k) \tag{30}$$

where $\boldsymbol{u}(\hat{\boldsymbol{x}}_k, k)$ represents the approximated control input vector at time step k, $\hat{\boldsymbol{W}}_{\boldsymbol{u}k}$ and $\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)$ are the estimated values of the actor NN weights and "reconstructed" activation function with the estimated state vector $\hat{\boldsymbol{x}}_k$ as the input.

Define the control input error as

$$\boldsymbol{e}_{\boldsymbol{u}k} = \boldsymbol{u}(\hat{\boldsymbol{x}}_k, k) - \boldsymbol{u}_1(\hat{\boldsymbol{x}}_k, k) \tag{31}$$

where $\boldsymbol{u}_1(\hat{\boldsymbol{x}}_k, k) = -\boldsymbol{\varphi}\left(\frac{1}{2}\boldsymbol{R}^{-1}\hat{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1}, k+1)\hat{\boldsymbol{W}}_{Vk}\right)$ is the control policy that minimizes the approximated value function $\hat{V}(\hat{\boldsymbol{x}}_k, k)$, ∇ denotes the gradient of the estimated value function with respect to the system states, $\hat{g}(\hat{\boldsymbol{x}}_k)$ is the approximated control coefficient matrix generated by the NN-based observer and $\hat{V}(\hat{\boldsymbol{x}}_{k+1}, k+1)$ is the approximated value function from the critic network.

Therefore, the control error (31) becomes

$$\begin{aligned} \boldsymbol{e}_{\boldsymbol{u}k} &= \boldsymbol{u}(\hat{\boldsymbol{x}}_k, k) - \boldsymbol{u}_1(\hat{\boldsymbol{x}}_k, k) \\ &= \hat{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}} \sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k) + \boldsymbol{\varphi}\left(\frac{1}{2}\boldsymbol{R}^{-1}\hat{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1}, k+1)\hat{\boldsymbol{W}}_{Vk}\right) \end{aligned} \tag{32}$$

The actor NN weights tuning law is then defined as

$$\hat{\boldsymbol{W}}_{\boldsymbol{u}k+1} = \hat{\boldsymbol{W}}_{\boldsymbol{u}k} - \alpha_{\boldsymbol{u}} \frac{\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)\boldsymbol{e}_{\boldsymbol{u}k}^{\mathrm{T}}}{1 + \sigma_{\boldsymbol{u}}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k, k)\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)} \tag{33}$$

where $\alpha_{\boldsymbol{u}} > 0$ is a design parameter.

To find the error dynamics for the actor NN weights, first observe that

$$\begin{aligned} \boldsymbol{u}(\boldsymbol{x}_k, k) &= \boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, k) + \boldsymbol{\varepsilon}_{\boldsymbol{u}}(\boldsymbol{x}_k, k) \\ &= -\boldsymbol{\varphi}\left(\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)(\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1}, k+1)\boldsymbol{W}_V + \nabla\boldsymbol{\varepsilon}_V(\boldsymbol{x}_{k+1}, k+1))\right) \end{aligned} \tag{34}$$

Or equivalently,

$$\begin{aligned} 0 &= \boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, k) + \boldsymbol{\varepsilon}_{\boldsymbol{u}}(\boldsymbol{x}_k, k) \\ &\quad + \boldsymbol{\varphi}\left(\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)(\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1}, k+1)\boldsymbol{W}_V + \nabla\boldsymbol{\varepsilon}_V(\boldsymbol{x}_{k+1}, k+1))\right) \end{aligned} \tag{35}$$

Subtracting (35) from (32), we have

$$\begin{aligned} \boldsymbol{e}_{\boldsymbol{u}k} &= \hat{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k) + \boldsymbol{\varphi}\left(\frac{1}{2}\boldsymbol{R}^{-1}\hat{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1}, k+1)\hat{\boldsymbol{W}}_{Vk}\right) - \boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, k) + \boldsymbol{\varepsilon}_{\boldsymbol{u}}(\boldsymbol{x}_k, k) \\ &\quad - \boldsymbol{\varphi}\left(\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)(\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1}, k+1)\boldsymbol{W}_V + \nabla\boldsymbol{\varepsilon}_V(\boldsymbol{x}_{k+1}, k+1))\right) \\ &= \hat{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k) - \boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k) + \boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k) + \tilde{\boldsymbol{\varphi}}_k - \boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, k) + \boldsymbol{\varepsilon}_{\boldsymbol{u}}(\boldsymbol{x}_k, k) \\ &= -\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k) - \boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\tilde{\sigma}_{\boldsymbol{u}}(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k, k) + \tilde{\boldsymbol{\varphi}}_k + \boldsymbol{\varepsilon}_{\boldsymbol{u}}(\boldsymbol{x}_k, k) \\ &= -\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k) - \frac{1}{2}L_\phi\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1}, k+1)\tilde{\boldsymbol{W}}_{Vk} \\ &\quad - \frac{1}{2}L_\phi\boldsymbol{R}^{-1}\tilde{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1}, k+1)\boldsymbol{W}_V \\ &\quad - \frac{1}{2}L_\phi\boldsymbol{R}^{-1}(g^{\mathrm{T}}(\hat{\boldsymbol{x}}_k) - g^{\mathrm{T}}(\boldsymbol{x}_k))\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1}, k+1)\tilde{\boldsymbol{W}}_{Vk} \\ &\quad + \frac{1}{2}L_\phi\boldsymbol{R}^{-1}\tilde{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1}, k+1)\tilde{\boldsymbol{W}}_{Vk} + \bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}}(\boldsymbol{x}_k, k) \end{aligned} \tag{36}$$

where $\tilde{\boldsymbol{W}}_{\boldsymbol{u}k} = \boldsymbol{W}_{\boldsymbol{u}} - \hat{\boldsymbol{W}}_{\boldsymbol{u}k}$, L_{ϕ} is the positive Lipschitz constant for the saturation function $\phi(\bullet)$, $\tilde{\sigma}_{\boldsymbol{u},k}(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k, k) = \sigma_{\boldsymbol{u}}(\boldsymbol{x}_k, k) - \sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)$, $\tilde{\boldsymbol{\varphi}}_k = \phi\left(\frac{1}{2}\boldsymbol{R}^{-1}\hat{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1}, \mathrm{N}-k-1)\hat{\boldsymbol{W}}_{V,k}\right) - \varphi\left(\frac{1}{2}\boldsymbol{R}^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)(\nabla\sigma_V^{\mathrm{T}}(\boldsymbol{x}_{k+1}, k+1)\boldsymbol{W}_V + \nabla\varepsilon_V(\boldsymbol{x}_{k+1}, k+1))\right)$ and $\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}}(\boldsymbol{x}_k, k) = -\boldsymbol{\varepsilon}_{\boldsymbol{u}}(\boldsymbol{x}_k, k) + \frac{1}{2}L_{\phi}\boldsymbol{R}^{-1} \times g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\tilde{\sigma}_V^{\mathrm{T}}(\boldsymbol{x}_{k+1}, \hat{\boldsymbol{x}}_{k+1}, k+1)\boldsymbol{W}_V + \frac{1}{2}L_{\phi}\boldsymbol{R}^{-1}(g^{\mathrm{T}}(\hat{\boldsymbol{x}}_k) - g^{\mathrm{T}}(\boldsymbol{x}_k))\nabla\sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1}, k+1)\boldsymbol{W}_V - \frac{1}{2}L_{\phi}\boldsymbol{R}^{-1} \times g^{\mathrm{T}}(\boldsymbol{x}_k)\nabla\varepsilon_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{k+1}, k+1) - \boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\tilde{\sigma}_{\boldsymbol{u}}(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k, k)$. Note that $\tilde{\sigma}_{\boldsymbol{u},k}(\boldsymbol{x}_k, \hat{\boldsymbol{x}}_k, k)$ and $\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}}(\boldsymbol{x}_k, k)$ are all bounded due to the boundedness of NN activation function and reconstruction error.

Then the error dynamics for the actor NN weights are revealed to be

$$\tilde{\boldsymbol{W}}_{\boldsymbol{u}k+1} = \tilde{\boldsymbol{W}}_{\boldsymbol{u}k} + \alpha_{\boldsymbol{u}}\frac{\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)\boldsymbol{e}_{\boldsymbol{u}k}^{\mathrm{T}}}{1 + \sigma_{\boldsymbol{u}}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k, k)\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)} \tag{37}$$

Remark 2: The update law for tuning the actor NN weights is based on gradient descent approach and it is similar to [14] with the difference being the estimated state vector $\hat{\boldsymbol{x}}_k$ is utilized as the input to the actor NN activation function instead of actual system state vector $\boldsymbol{x}_k$. In addition, total error comprising of Bellman error and terminal constraint error are utilized to tune the weights whereas in [14], the terminal constraint is ignored. Further, the optimal control scheme in this work utilizes the identified control coefficient matrix $\hat{g}(\hat{\boldsymbol{x}}_k)$, whereas in [14], the control coefficient matrix $g(\boldsymbol{x}_k)$ is assumed to be known. Due to these differences, the stability analysis differs significantly from [14].

3.3 DYNAMIC QUANTIZER DESIGN

To handle the saturation caused by limited quantization range for a realistic quantizer, a new parameter μ_k is introduced. The proposed dynamic quantizers for the control input is defined as

$$\boldsymbol{u}_{qk} = q_d(\boldsymbol{u}_k) = \mu_k q(\boldsymbol{u}_k / \mu_k) \tag{38}$$

where μ_k is a time-varying scaling parameter to be defined later for the control input quantizers, respectively. Normally, the dynamics of the quantization error cannot be established since it is mainly a round-off error. Instead, we will consider the quantization error bound as presented next, which will aid in the stability analysis. Given the dynamic quantizer in the form (38), the quantization error for the control inputs is bounded, as long as no saturation occurs and the bound is given by

$$\left\|\boldsymbol{e}_{\boldsymbol{u}k}\right\| = \left\|\boldsymbol{u}_{qk} - \boldsymbol{u}_k\right\| \le \frac{1}{2}\mu_k \Delta_k = e_{\mathrm{M},k} \tag{39}$$

where $e_{\mathrm{M},k}$ is the upper bound for the control input quantization error.

Next, define the scaling parameter μ_k as

$$\mu_k = \left\|\boldsymbol{u}_k\right\| / (\lambda^k \mathrm{M}) \tag{40}$$

where $0 < \lambda < 1$. Recall from representation (38) that the signals to be quantized can be “scaled” back into the quantization range with the decaying rate of λ^k , and thus eliminating the saturation effect.

Remark 3: The scaling parameter μ_k have the following properties: First, μ_k are adjusted to eliminate saturation, which are more applicable in the realistic situations. Second, μ_k are time-varying parameters and updated at each time interval. Finally, updating μ_k only requires the signals to be quantized, which differs from [10] in which μ is a constant and can only obtained by using the system dynamics.

To complete this subsection, the flowchart of our proposed finite-horizon near optimal regulation scheme is shown in Figure 3.

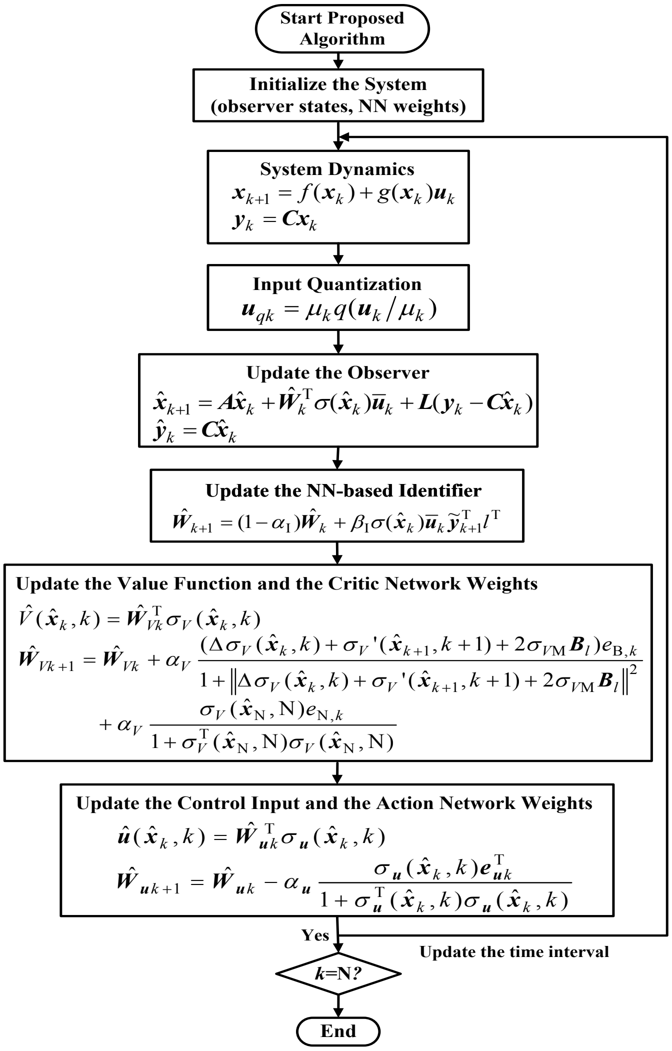

Start Proposed Algorithm
Initialize the System (observer states, NN weights)
System Dynamics
$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_k$
$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k$
Input Quantization
$\boldsymbol{u}_{qk} = \mu_k q(\boldsymbol{u}_k / \mu_k)$
Update the Observer
$\hat{\boldsymbol{x}}_{k+1} = \boldsymbol{A}\hat{\boldsymbol{x}}_k + \hat{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\hat{\boldsymbol{x}}_k)\overline{\boldsymbol{u}}_k + \boldsymbol{L}(\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k)$
$\hat{\boldsymbol{y}}_k = \boldsymbol{C}\hat{\boldsymbol{x}}_k$
Update the NN-based Identifier
$\hat{\boldsymbol{W}}_{k+1} = (1-\alpha_{\mathrm{I}})\hat{\boldsymbol{W}}_k + \beta_{\mathrm{I}}\sigma(\hat{\boldsymbol{x}}_k)\overline{\boldsymbol{u}}_k \tilde{\boldsymbol{y}}_{k+1}^{\mathrm{T}} l^{\mathrm{T}}$
Update the Value Function and the Critic Network Weights
$\hat{V}(\hat{\boldsymbol{x}}_k, k) = \hat{\boldsymbol{W}}_{Vk}^{\mathrm{T}} \sigma_V(\hat{\boldsymbol{x}}_k, k)$
$\hat{\boldsymbol{W}}_{Vk+1} = \hat{\boldsymbol{W}}_{Vk} + \alpha_V \frac{(\Delta\sigma_V(\hat{\boldsymbol{x}}_k, k) + \sigma_V'(\hat{\boldsymbol{x}}_{k+1}, k+1) + 2\sigma_{V\mathrm{M}}\boldsymbol{B}_l)e_{\mathrm{B},k}}{1 + \left\| \Delta\sigma_V(\hat{\boldsymbol{x}}_k, k) + \sigma_V'(\hat{\boldsymbol{x}}_{k+1}, k+1) + 2\sigma_{V\mathrm{M}}\boldsymbol{B}_l \right\|^2}$
$+ \alpha_V \frac{\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}}, \mathrm{N})e_{\mathrm{N},k}}{1 + \sigma_V^{\mathrm{T}}(\hat{\boldsymbol{x}}_{\mathrm{N}}, \mathrm{N})\sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}}, \mathrm{N})}$
Update the Control Input and the Action Network Weights
$\hat{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k) = \hat{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}} \sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)$
$\hat{\boldsymbol{W}}_{\boldsymbol{u}k+1} = \hat{\boldsymbol{W}}_{\boldsymbol{u}k} - \alpha_{\boldsymbol{u}} \frac{\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)\boldsymbol{e}_{\boldsymbol{u}k}^{\mathrm{T}}}{1 + \sigma_{\boldsymbol{u}}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k, k)\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k, k)}$
Yes
Update the time interval
k=N?
End

Figure 3. Flowchart of the proposed finite-horizon near optimal regulator

We initialize the system with an admissible control as well as proper parameter selection and NN weights initialization. The control input is then quantized using proposed dynamic quantizer. The NNs for observer, critic and actor are updated based on our proposed weights tuning laws at each sampling interval beginning with an initial time and until the final fixed time instant in an online and forward-in-time fashion.

3.4 STABILITY ANALYSIS

In this subsection, the system stability will be investigated. It will be shown that the overall closed-loop system remain bounded under the proposed near optimal regulator design. Before proceeding, the following lemma is needed.

Lemma: (*Bounds on the optimal closed-loop dynamics*) Consider the discrete-time nonlinear system (1), then there exists an optimal control policy $\boldsymbol{u}_k^*$ such that closed-loop system dynamics $f(\boldsymbol{x}_k)+g(\boldsymbol{x}_k)\boldsymbol{u}_k^*$ can be written as

$$\left\|f(\boldsymbol{x}_k)+g(\boldsymbol{x}_k)\boldsymbol{u}_k^*\right\|^2 \le \rho\left\|\boldsymbol{x}_k\right\|^2 \qquad (41)$$

where $0<\rho<1$ is a constant.

Theorem 3 (*Boundedness of the closed-loop system*) Let the nonlinear system (1) be controllable and observable while the system output, $\boldsymbol{y}_k \in \Omega_{\boldsymbol{y}}$, be measurable. Let the initial NN weights for the observer, critic network and actor network $\hat{\boldsymbol{W}}_k$, $\hat{\boldsymbol{W}}_{Vk}$ and $\hat{\boldsymbol{W}}_{\boldsymbol{u}k}$ be selected within compact set Ω_{OB}, Ω_V and Ω_{AN} which contains the ideal weights $\boldsymbol{W}$, $\boldsymbol{W}_V$ and $\boldsymbol{W}_{\boldsymbol{u}}$. Let $\boldsymbol{u}(0)\in\Omega_{\boldsymbol{u}}$ be an initial admissible control input for the system (1). Let the observer be given by (13) and the NN weights update law for the observer, critic network and action network be provided by (15), (25) and (33), respectively. Then, there exists positive constant $\frac{2-\sqrt{2}}{2}<\alpha_{\mathrm{I}}<1$, $0<\alpha_V<1/6$ and $0<\alpha_{\boldsymbol{u}}<1$, such that the system state $\boldsymbol{x}_k$, observer error $\tilde{\boldsymbol{x}}_k$, NN observer weight estimation errors $\tilde{\boldsymbol{W}}_k$, critic and action network weights estimation errors $\tilde{\boldsymbol{W}}_{Vk}$ and $\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}$ are all UUB, with the ultimate bounds given by (A.20) ~ (A.24). Moreover, the estimated control input is bounded closed to the optimal value such that $\left\|\boldsymbol{u}^*(x_k,k)-\hat{\boldsymbol{u}}(\hat{x}_k,k)\right\|\le\varepsilon_{\boldsymbol{u}o}$ for a small positive constant $\varepsilon_{\boldsymbol{u}o}$.

Proof: See appendix.

4. SIMULATION RESULTS

In this section, a practical example is considered to illustrate our proposed near optimal regulation design scheme. Consider the two-link planar robot arm [15]:

$$\begin{aligned}\dot{\boldsymbol{x}}&=f(\boldsymbol{x})+g(\boldsymbol{x})\boldsymbol{u}\\ \boldsymbol{y}&=\boldsymbol{C}\boldsymbol{x}\end{aligned}\tag{42}$$

where $f(\boldsymbol{x})=\begin{bmatrix} x_3\\ x_4\\ \dfrac{\begin{pmatrix}-(2x_3x_4+x_4^2-x_3^2-x_3^2\cos x_2)\sin x_2\\ +20\cos x_1-10\cos(x_1+x_2)\cos x_2\end{pmatrix}}{\cos^2 x_2-2}\\ \dfrac{\begin{pmatrix}(2x_3x_4+x_4^2+2x_3x_4\cos x_2+x_4^2\cos x_2+3x_3^2\\ +2x_3^2\cos x_2+20(\cos(x_1+x_2)-\cos x_1)\times\\ (1+\cos x_2)-10\cos x_2\cos(x_1+x_2)\end{pmatrix}}{\cos^2 x_2-2}\end{bmatrix}$ and $g(\boldsymbol{x})=\begin{bmatrix}0&0\\ 0&0\\ \dfrac{1}{2-\cos^2 x_2}&\dfrac{-1-\cos x_2}{2-\cos^2 x_2}\\ \dfrac{-1-\cos x_2}{2-\cos^2 x_2}&\dfrac{3+2\cos x_2}{2-\cos^2 x_2}\end{bmatrix}$.

The system is discretized with sampling time of $h=5\text{ms}$ and the control constraint is set to be $U=1.5$, i.e., $-1.5\le u_1\le 1.5$ and $-1.5\le u_2\le 1.5$. Define the performance index

$$V(\boldsymbol{x}_k,k)=\psi(\boldsymbol{x}_{\mathrm{N}})+\sum_{i=k}^{\mathrm{N}-1}(\boldsymbol{Q}(\boldsymbol{x}_i,i)+2\int_0^{\boldsymbol{u}_i}U\tanh^{-\mathrm{T}}\left(\frac{\boldsymbol{v}}{U}\right)\boldsymbol{R}d\boldsymbol{v})\tag{43}$$

where $\boldsymbol{Q}(\boldsymbol{x}_k, k)$, for simplicity, is selected as standard quadratic form of the system states as $\boldsymbol{Q}(\boldsymbol{x}_k, k) = \boldsymbol{x}_k^{\mathrm{T}} \overline{\boldsymbol{Q}} \boldsymbol{x}_k$ with $\overline{\boldsymbol{Q}} = 0.1\boldsymbol{I}_4$ and weighting matrix $\boldsymbol{R}$ is selected as $\boldsymbol{R} = 0.001\boldsymbol{I}_2$, where $\boldsymbol{I}$ denotes the identity matrix with appropriate dimension. The Hurwitz matrix $\boldsymbol{A}$ is selected as a 4×4 block diagonal matrix whose blocks $\boldsymbol{A}_{ii}$ are chosen to be $\boldsymbol{A}_{ii} = \begin{bmatrix} 0.9 & 0.1 \\ 0 & 0.9 \end{bmatrix}$. The terminal constraint is chosen as $\phi(\boldsymbol{x}_{\mathrm{N}}) = 3$. For the NN setup, the inputs for the NN observer are selected as $\boldsymbol{z}_k = [\hat{\boldsymbol{x}}_k, \boldsymbol{u}_k]$. The time-varying activation functions for the critic and actor network are chosen as sigmoid function with input to be $[\hat{x}_1, \cdots \hat{x}_4, \tau, \hat{x}_1\hat{x}_2, \cdots, \hat{x}_3\hat{x}_4, \tau^2, \hat{x}_1\tau, \cdots \hat{x}_4\tau, \hat{x}_1^2, \cdots, \hat{x}_4^2]$ and $[\hat{x}_1, \cdots \hat{x}_4, \hat{x}_1\tau, \cdots, \hat{x}_4\tau]$, which result in 24 and 8 neurons, respectively, and $\tau = (\mathrm{N} - k)/\mathrm{N}$ is the normalized time-to-go.

The design parameters are chosen as $\alpha_{\mathrm{I}} = 0.7$, $\beta_{\mathrm{I}} = 0.01$, $\alpha_V = 0.1$, $\alpha_{\boldsymbol{u}} = 0.03$ and $\lambda = 0.9$. The initial system states and the observer states are selected as $\boldsymbol{x}_0 = [\pi/3, \pi/6, 0, 0]^{\mathrm{T}}$ and $\hat{\boldsymbol{x}}_0 = [0,0,0,0]^{\mathrm{T}}$, respectively. The initial admissible control input is chosen as $\boldsymbol{u}(0) = [0.2;\ -1]$. The observer gain is chosen as $\boldsymbol{L} = [-0.3,\ 0.1, 0.7, 1]^{\mathrm{T}}$ and the matching matrices $\boldsymbol{B}_l$ and $\boldsymbol{l}$ are selected as column vectors with all ones. All the NN weights are initialized at random.

First, the system response and control input are shown in Figure 4 and Figure 5, respectively. Both system states and control clearly converge close enough to the origin within finite time period, which illustrates the stability of the proposed design scheme.

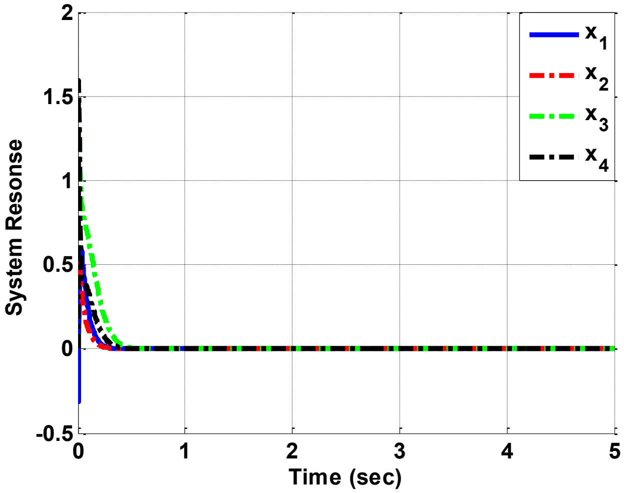


Figure 4. System response

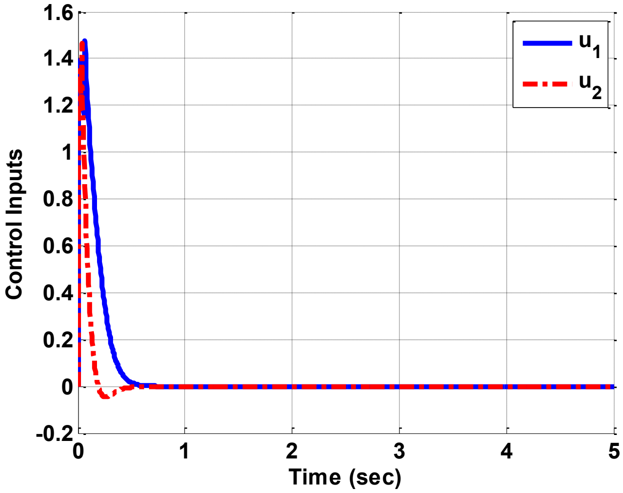


Figure 5. Control inputs

Next, the quantization errors for the control inputs with proposed dynamic quantizer and traditional uniform quantizer are shown in Figure 6 and Figrue 7, respectively. Comparing with Figure 6 and 7, it is clear that the quantization errors are decreasing overtime instead of keep bounded as for traditional uniform quantizer, which illustrates the effectiveness of the proposed dynamic quantizer design.

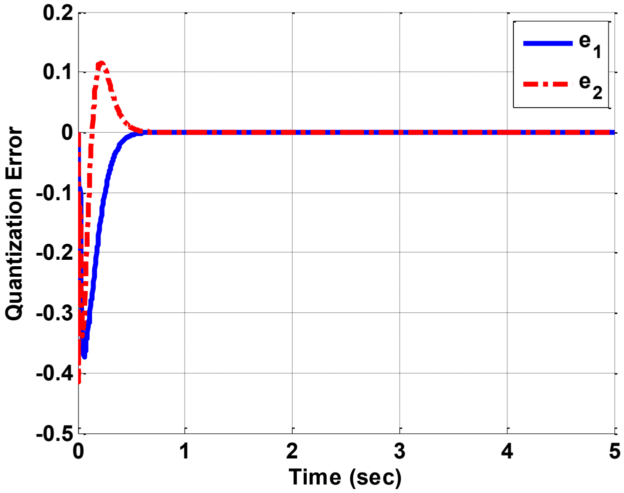


Figure 6. Quantization error with dynamic quantizer

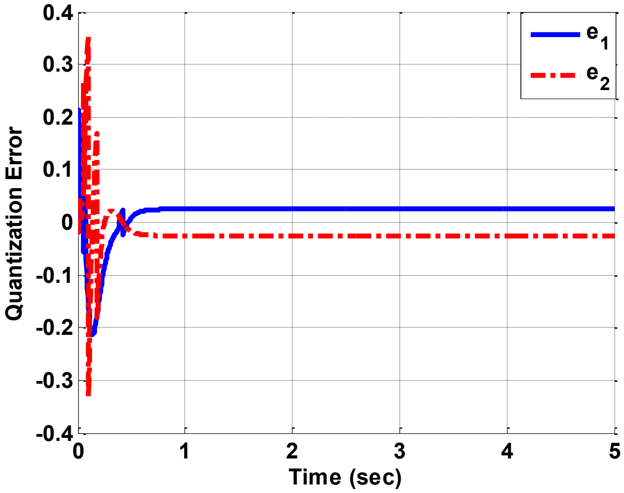


Figure 7. Quantization error with static quantizer

Next, the error history in the design procedure is given in Figure 8 and Figure 9, respectively. From the figure, it can be seen that Bellman equation error eventually converges close to zero, which illustrates the fact that the optimality is indeed achieved. More importantly, the convergence of the terminal constraint error demonstrates that the terminal constraint is also satisfied by our proposed design scheme.

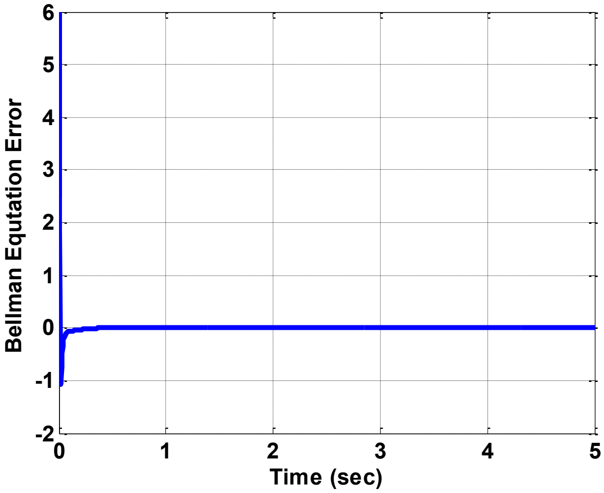


Figure 8. History of bellman equation error

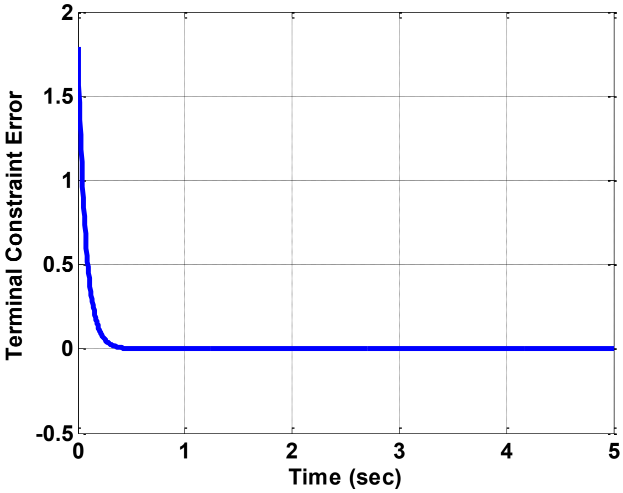


Figure 9. History of terminal constraint error

Finally, the convergence of critic and actor NN weights is shown in Figure 10. It can be observed from the results that the novel NN structure with our proposed tuning law guarantees that the NN weights converge to constants and remain bounded, as desired. This illustrates the feasibility of NN approximation for time-varying functions.

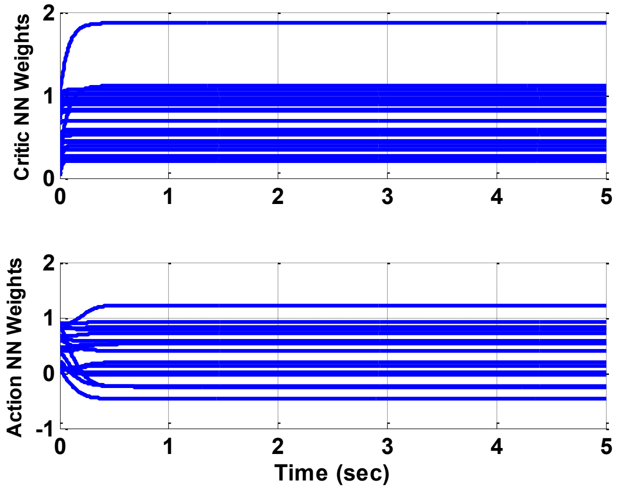


Figure 10. Convergence of critic/actor NN weights

5. CONCLUSIONS

In this paper, the NN-based fixed final time near optimal regulator design by using output feedback for quantized nonlinear discrete-time system in affine form with completely unknown system dynamics is addressed. Compared to the traditional finite-horizon optimal regulation design, the proposed scheme not only relaxes the requirement on availability of the system states and control coefficient matrix, but also takes input-constraint and quantization effect into account as well as functions in an online and forward-in-time manner instead of offline training and using value/policy iterations. An initial admissible control input is needed.

The input-constraint is handled by using a non-quadratic cost functional so that the optimality can be achieved. The dynamic quantizer effectively mitigates the quantization error for the control inputs while the NN-based Luenberger observer relaxes the need for an additional identifier. Time-dependency nature of the finite-horizon is handled by a NN structure with constant weights and time-varying activation function.

The terminal constraint is properly satisfied by minimizing an additional error term along the system trajectory. All NN weights are tuned online by using proposed update laws and Lyapunov stability theory demonstrated that the approximated control inputs converges close to its optimal value as time evolves. The performance of the proposed finite time near optimal regulator is demonstrated via simulation.

6. REFERENCES

[1] H. Sussmann, E. D. Sontag and Y. Yang, "A general result on the stabilization of linear systems using bounded controls," IEEE Trans. Autom. Control, vol. 39, pp. 2411–2425, 1994.

[2] Saberi, A., Z. Lin, A. Teel, "Control of linear systems with saturating actuators," Autom. Control, vol. 41, pp. 368–378, 1996.

[3] D. Kirk, Optimal Control Theory: An Introduction, New Jersey, Prentice-Hall, 1970.

[4] F. L. Lewis and V. L. Syrmos, Optimal Control, 2nd edition. New York: Wiley, 1995.

[5] M. Abu-Khalaf and F. L. Lewis, "Near optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach," Automatica, vol. 41, pp. 779–791, 2005.

[6] S.E. Lyshevski, "Optimal control of nonlinear continuous-time systems: design of bounded controllers via generazlied nonquadratic functionals," Proc. American Control Conf, USA, pp. 205–209, 1998.

[7] Li Tan, Digital Signal Processing, Fundamentals and Applications, Chapter 2, Academic Press, 2007.

[8] Delchamps DF, "Stabilizing a linear system with quantized state feedback," IEEE Trans. Autom. Control, vol 35: pp. 916–924, 1990.

[9] Brockett RW and Liberzon D, "Quantized feedback stabilization of linear systems," IEEE Trans. Autom. Control , vol 45, pp. 1279–1289, 2000.

[10] Liberzon D, "Hybrid feedback stabilization of systems with quantized signals," Automatica, vol 39, pp. 1543–1554, 2003.

[11] J. E. Slotine and W. Li, Applied Nonlinear Control, Englewood Cliffs, NJ: Prentice-Hall, 1991.

[12] H. K. Khalil and Laurent Praly, "High-gain observers in nonlinear feedback control", International Journal of Robust and Nonlinear Control, 21 Jul 2013.

[13] H. Xu and S. Jagannathan, "Stochastic optimal controller design for uncertain nonlinear networked control system via neuro dynamic programming," IEEE Trans. Neural Netw. and Learning Syst, vol. 24, pp. 471–484, 2013.

[14] T. Dierks and S. Jagannathan, “Online optimal control of affine nonlinear discrete-time systems with unknown internal dynamics by using time-based policy update,” IEEE Trans. Neural Netw. and Learning Syst, vol. 23, pp. 1118–1129, 2012.

[15] Z. Chen and S. Jagannathan, “Generalized Hamilton-Jacobi-Bellman formulation based neural network control of affine nonlinear discrete-time systems”, IEEE Trans. Neural Network, vol. 7, pp. 90–106, 2008.

[16] R. Beard, “Improving the closed-loop performance of nonlinear systems,” Ph.D. dissertation, Rensselaer Polytechnic Institute, USA, 1995.

[17] T. Cheng, F. L. Lewis, and M. Abu-Khalaf, “A neural network solution for fixed-final-time optimal control of nonlinear systems,” Automatica, vol. 43, pp. 482–490, 2007.

[18] A. Heydari and S. N. Balakrishan, “Finite-horizon control-constrained nonlinear optimal control using single network adaptive critics,” IEEE Trans. Neural Netw. and Learning Syst., vol. 24, pp. 145–157, 2013.

[19] F.Y. Wang, N. Jin, D. Liu and Q. Wei, “Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with ε –error bound,” IEEE Trans. Neural Networks, vol. 22, pp. 24–36, 2011.

[20] T. Dierks and S. Jagannathan, “Optimal control of affine nonlinear discrete-time systems with unknown internal dynamics,” in Proc. Conf. on Decision and Control, Shanghai, pp. 6750–6755, 2009.

[21] K. S. Narendra and K. Parthasarathy, “Identification and control of dynamical systems using neural networks,” IEEE Trans. Neural Networks, vol. 1, pp. 4–27, 1990.

[22] S. Jagannathan, Neural Network Control of Nonlinear Discrete-Time Systems, Boca Raton, FL: CRC Press, 2006.

[23] D. Tse, and P. Viswanath, Fundamentals of wireless communication, Cambridge University Press, 2005.

[24] Q. Zhao, H. Xu, and S. Jagannathan, “Adaptive dynamic programming-based state quantized networked control system without value and/or policy iterations,” in Proc. Int. Joint Conf. on Neur. Net., pp. 1-7, 2012.

APPENDIX

Proof of Theorem 1: Consider the following Lyapunov candidate

$$L_{\mathrm{IO}}(k) = L_{\tilde{\boldsymbol{x}},k} + L_{\tilde{\boldsymbol{W}},k} \tag{A.1}$$

where $L_{\tilde{\boldsymbol{x}},k} = \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \tilde{\boldsymbol{x}}_k$, $L_{\tilde{\boldsymbol{W}},k} = \mathrm{tr}\{\tilde{\boldsymbol{W}}_k^{\mathrm{T}} \boldsymbol{\Lambda} \tilde{\boldsymbol{W}}_k\}$ and $\boldsymbol{\Lambda} = \dfrac{2(1+\chi_{\min}^2)}{\beta_{\mathrm{I}}} \boldsymbol{I}$, with $\boldsymbol{I} \in \Re^{L \times L}$ the identity matrix and $0 < \chi_{\min}^2 < \sigma_{\min}^2 < \left\|\sigma(\hat{\boldsymbol{x}}_k)\right\|^2 < \left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2$ is ensured to exist by the PE conditions, and $\mathrm{tr}\{\bullet\}$ denotes the trace operator.

The first difference of $L_{\mathrm{IO}}(k)$ is given by

$$\Delta L_{\mathrm{IO}}(k) = \Delta L_{\tilde{\boldsymbol{x}},k} + \Delta L_{\tilde{\boldsymbol{W}},k} \tag{A.2}$$

Next, we consider each term in (A.2) individually. First, recall from the observer error dynamics (14), we have

$$\begin{aligned}
\Delta L_{\tilde{\boldsymbol{x}},k} &= \tilde{\boldsymbol{x}}_{k+1}^{\mathrm{T}} \tilde{\boldsymbol{x}}_{k+1} - \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \tilde{\boldsymbol{x}}_k \\
&= (\boldsymbol{A}_c \tilde{\boldsymbol{x}}_k + \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk} + \bar{\boldsymbol{\varepsilon}}_{Ok})^{\mathrm{T}} \times (\boldsymbol{A}_c \tilde{\boldsymbol{x}}_k + \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk} + \bar{\boldsymbol{\varepsilon}}_{Ok}) - \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \tilde{\boldsymbol{x}}_k \\
&= \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \boldsymbol{A}_c^{\mathrm{T}} \boldsymbol{A}_c \tilde{\boldsymbol{x}}_k + [\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}]^{\mathrm{T}} \tilde{\boldsymbol{W}}_k \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk} + \bar{\boldsymbol{\varepsilon}}_{Ok}^{\mathrm{T}} \bar{\boldsymbol{\varepsilon}}_{Ok} \\
&\quad + 2\tilde{\boldsymbol{x}}_k^{\mathrm{T}} \boldsymbol{A}_c^{\mathrm{T}} \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk} + 2\tilde{\boldsymbol{x}}_k^{\mathrm{T}} \boldsymbol{A}_c^{\mathrm{T}} \bar{\boldsymbol{\varepsilon}}_{Ok} + 2\bar{\boldsymbol{\varepsilon}}_{Ok}^{\mathrm{T}} \tilde{\boldsymbol{W}}_k^{\mathrm{T}} \sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk} - \tilde{\boldsymbol{x}}_k^{\mathrm{T}} \tilde{\boldsymbol{x}}_k \\
&\le -(1-\gamma)\left\|\tilde{\boldsymbol{x}}_k\right\|^2 + 3\left\|\tilde{\boldsymbol{W}}_k\right\|^2 \left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2 + 3\left\|\bar{\boldsymbol{\varepsilon}}_{Ok}\right\|^2
\end{aligned} \tag{A.3}$$

where $\gamma = 3\left\|\boldsymbol{A}_c\right\|^2$.

Next, recall the NN weight estimation error dynamics (16), we have

$$\begin{aligned}
\Delta L_{\tilde{W},k} &= \mathrm{tr}\{\tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}}\boldsymbol{\varLambda}\tilde{\boldsymbol{W}}_{k+1}\} - \mathrm{tr}\{\tilde{\boldsymbol{W}}_{k}^{\mathrm{T}}\boldsymbol{\varLambda}\tilde{\boldsymbol{W}}_{k}\} \\
&\le 2(1-\alpha_{\mathrm{I}})^2\,\mathrm{tr}\{\tilde{\boldsymbol{W}}_{k}^{\mathrm{T}}\boldsymbol{\varLambda}\tilde{\boldsymbol{W}}_{k}\} + 6\alpha_{\mathrm{I}}^2\Lambda W_{\mathrm{M}}^2 + 6\beta_{\mathrm{I}}^2\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2\left\|\boldsymbol{A}_c\right\|^2\left\|\boldsymbol{lC}\right\|^2\left\|\tilde{\boldsymbol{x}}_k\right\|^2 \\
&\quad - 4(1-\alpha_{\mathrm{I}})\beta_{\mathrm{I}}\lambda_{\min}(\boldsymbol{lC})\tilde{\boldsymbol{W}}_{k}^{\mathrm{T}}\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2\tilde{\boldsymbol{W}}_{k} + 2\beta_{\mathrm{I}}^2\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2\left\|\boldsymbol{lC}\right\|^2\left\|\tilde{\boldsymbol{W}}_{k}\right\|^2 \\
&\quad + 6\beta_{\mathrm{I}}^2\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2\left\|\boldsymbol{lC}\right\|^2\left\|\bar{\boldsymbol{\varepsilon}}_{Ok}\right\|^2 - \mathrm{tr}\{\tilde{\boldsymbol{W}}_{k}^{\mathrm{T}}\boldsymbol{\varLambda}\tilde{\boldsymbol{W}}_{k}\} \\
&\le -(1-2(1-\alpha_{\mathrm{I}})^2)\mathrm{tr}\{\tilde{\boldsymbol{W}}_{k}^{\mathrm{T}}\boldsymbol{\varLambda}\tilde{\boldsymbol{W}}_{k}\} + 6\alpha_{\mathrm{I}}^2\Lambda W_{\mathrm{M}}^2 + 6\beta_{\mathrm{I}}^2\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2\left\|\boldsymbol{A}_c\right\|^2\left\|\boldsymbol{lC}\right\|^2\left\|\tilde{\boldsymbol{x}}_k\right\|^2 \qquad \text{(A.4)} \\
&\quad - 2\beta_{\mathrm{I}}((1-\alpha_{\mathrm{I}})\lambda_{\min}(\boldsymbol{lC}) - \beta_{\mathrm{I}}\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{k}\right\|^2)\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2\left\|\tilde{\boldsymbol{W}}_{k}\right\|^2 \\
&\quad + 6\beta_{\mathrm{I}}^2\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2\left\|\boldsymbol{lC}\right\|^2\left\|\bar{\boldsymbol{\varepsilon}}_{Ok}\right\|^2 \\
&\le -(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\left\|\tilde{\boldsymbol{W}}_{k}\right\|^2 + 6\beta_{\mathrm{I}}^2\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2\left\|\boldsymbol{A}_c\right\|^2\left\|\boldsymbol{lC}\right\|^2\left\|\tilde{\boldsymbol{x}}_k\right\|^2 \\
&\quad - 2\beta_{\mathrm{I}}((1-\alpha_{\mathrm{I}})\lambda_{\min}(\boldsymbol{lC}) - \beta_{\mathrm{I}}\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2)\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2\left\|\tilde{\boldsymbol{W}}_{k}\right\|^2 + \varepsilon_{\mathrm{WM}}
\end{aligned}$$

where $\Lambda = \left\|\boldsymbol{\varLambda}\right\|$ and $\varepsilon_{\mathrm{WM}} = 6\alpha_{\mathrm{I}}^2\Lambda W_{\mathrm{M}}^2 + 6\beta_{\mathrm{I}}^2\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2\left\|\boldsymbol{lC}\right\|^2\left\|\bar{\boldsymbol{\varepsilon}}_{Ok}\right\|^2$.

Therefore, the first difference of the total Lyapunov candidate, by combining (A.3) and (A.4), is given as

$$\begin{aligned}
\Delta L_{\mathrm{IO}}(k) &= \Delta L_{\tilde{x},k} + \Delta L_{\tilde{W},k} \\
&\le -(1-\gamma)\left\|\tilde{\boldsymbol{x}}_k\right\|^2 + 3\left\|\tilde{\boldsymbol{W}}_{k}\right\|^2\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2 + 3\left\|\bar{\boldsymbol{\varepsilon}}_{Ok}\right\|^2 - (1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\left\|\tilde{\boldsymbol{W}}_{k}\right\|^2 \\
&\quad - \frac{2\beta_{\mathrm{I}}\Lambda}{1+\chi_{\min}^2}\left\|\tilde{\boldsymbol{W}}_{k}\right\|^2\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2 + 2\beta_{\mathrm{I}}\gamma\left\|\boldsymbol{lC}\right\|^2\Lambda\left\|\tilde{\boldsymbol{x}}_k\right\|^2 + \varepsilon_{\mathrm{WM}} \qquad \text{(A.5)} \\
&\le -(1-(1+4\left\|\boldsymbol{lC}\right\|^2(1+\chi_{\min}^2))\gamma)\left\|\tilde{\boldsymbol{x}}_k\right\|^2 - \left\|\tilde{\boldsymbol{W}}_{k}\right\|^2\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2 \\
&\quad - (1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\left\|\tilde{\boldsymbol{W}}_{k}\right\|^2 + \varepsilon_{\mathrm{OM}}
\end{aligned}$$

where $\varepsilon_{\mathrm{OM}} = 3\left\|\bar{\boldsymbol{\varepsilon}}_{Ok}\right\|^2 + \varepsilon_{\mathrm{WM}}$. By standard Lyapunov stability theory [22], $\Delta L_{\mathrm{IO},k}$ is less than zero outside a compact set as long as the following conditions hold:

$$\left\|\tilde{\boldsymbol{x}}_k\right\| > \sqrt{\frac{\varepsilon_{\mathrm{OM}}}{1-(1+4\left\|\boldsymbol{lC}\right\|^2(1+\chi_{\min}^2))\gamma}} \equiv b_{\tilde{x}} \qquad \text{(A.6)}$$

Or

$$\left\|\widetilde{\boldsymbol{W}}_k\right\| > \sqrt{\frac{\varepsilon_{\mathrm{OM}}}{\chi_{\min}^2 + (1-2(1-\alpha_{\mathrm{I}})^2)\Lambda}} \equiv b_{\widetilde{W}} \tag{A.7}$$

Note that in (A.6) the denominator is guaranteed to be positive, i.e., $0 < \gamma < \dfrac{1}{1+4\left\|\boldsymbol{lC}\right\|^2(1+\chi_{\min}^2)}$, by properly selecting the designed parameters $\boldsymbol{A}$, $\boldsymbol{L}$ and $\boldsymbol{l}$.

Proof of Theorem 2: First, for simplicity, denote $\Delta\hat{\sigma}_{Vk} = \Delta\sigma_V(\hat{\boldsymbol{x}}_k, k)$, $\hat{\sigma}_{Vk+1}' = \sigma_V'(\hat{\boldsymbol{x}}_{k+1}, k+1)$, $\Delta\varepsilon_{VBk} = \Delta\varepsilon_{VB}(\boldsymbol{x}_k, k)$ and $\hat{\sigma}_{V\mathrm{N}} = \sigma_V(\hat{\boldsymbol{x}}_{\mathrm{N}}, \mathrm{N})$. Consider the following Lyapunov candidate

$$L_{\widetilde{W}_V}(k) = L(\widetilde{\boldsymbol{W}}_{Vk}) + \Pi L(\widetilde{\boldsymbol{x}}_k) + \Pi L(\widetilde{\boldsymbol{W}}_k) \tag{A.8}$$

where $L(\widetilde{\boldsymbol{W}}_{Vk}) = \widetilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\widetilde{\boldsymbol{W}}_{Vk}$, $L(\widetilde{\boldsymbol{x}}_k) = (\widetilde{\boldsymbol{x}}_k^{\mathrm{T}}\widetilde{\boldsymbol{x}}_k)^2$, $L(\widetilde{\boldsymbol{W}}_k) = (\mathrm{tr}\{\widetilde{\boldsymbol{W}}_k^{\mathrm{T}}\boldsymbol{\Lambda}\widetilde{\boldsymbol{W}}_k\})^2$ and $\Pi = \dfrac{\alpha_V(1+3\alpha_V)(L_Q + W_{V\mathrm{M}}L_{\sigma_V})^2}{(1+\Delta\sigma_{\min}^2)(1-3\gamma^2)}$. Next, take each term in (A.8) individually. The first difference of $L(\widetilde{\boldsymbol{W}}_{Vk})$, by recalling (29), is given by

$$\begin{aligned}
\Delta L(\widetilde{\boldsymbol{W}}_{Vk}) &= \widetilde{\boldsymbol{W}}_{Vk+1}^{\mathrm{T}}\widetilde{\boldsymbol{W}}_{Vk+1} - \widetilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\widetilde{\boldsymbol{W}}_{Vk} \\
&= \left(\widetilde{\boldsymbol{W}}_{Vk} - \alpha_V\frac{(\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l)e_{\mathrm{B},k}}{1+\left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} - \alpha_V\frac{\hat{\sigma}_{V\mathrm{N}}e_{\mathrm{N},k}}{1+\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}}\right)^{\mathrm{T}} \times \\
&\quad \left(\widetilde{\boldsymbol{W}}_{Vk} - \alpha_V\frac{(\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l)e_{\mathrm{B},k}}{1+\left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} - \alpha_V\frac{\hat{\sigma}_{V\mathrm{N}}e_{\mathrm{N},k}}{1+\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}}\right) - \widetilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\widetilde{\boldsymbol{W}}_{Vk} \\
&= -2\alpha_V\frac{\widetilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}(\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l)e_{\mathrm{B},k}}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}} - 2\alpha_V\frac{\widetilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}e_{\mathrm{N},k}}{1+\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}} \\
&\quad + \alpha_V^2\frac{e_{\mathrm{B},k}^2(\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l)^{\mathrm{T}}(\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l)}{(1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk})^2} + \alpha_V^2\frac{e_{\mathrm{N},k}^2\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}}{(1+\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}})^2}
\end{aligned} \tag{A.9}$$

Recall from (27) and (28), the first difference of $L(\widetilde{\boldsymbol{W}}_{Vk})$ can be further derived as

$$
\begin{aligned}
\Delta L(\tilde{\boldsymbol{W}}_{Vk}) \le & -2\alpha_V \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}\begin{pmatrix} (L_Q + W_{V\mathrm{M}}L_{\sigma_V})\|\tilde{\boldsymbol{x}}_k\|^2 + \\ \tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}(\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l) + \Delta\varepsilon_{VBk} \end{pmatrix}}{1 + \left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} \\
& - 2\alpha_V \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}\left(\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}} + \varepsilon_{V\mathrm{N}}\right)}{1 + \hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}} + \alpha_V^2 \frac{\left(\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}} + \varepsilon_{V\mathrm{N}}\right)^2 \hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}}{(1 + \hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}})^2} \\
& + \alpha_V^2 \frac{\begin{pmatrix} (L_Q + W_{V\mathrm{M}}L_{\sigma_V})\|\tilde{\boldsymbol{x}}_k\|^2 + \\ \tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}(\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l) + \Delta\varepsilon_{VBk} \end{pmatrix}^2}{1 + \left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} \left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2 \\
\le & -2\alpha_V \frac{(L_Q + W_{V\mathrm{M}}L_{\sigma_V})\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}\|\tilde{\boldsymbol{x}}_k\|^2}{\left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} - 2\alpha_V \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2\tilde{\boldsymbol{W}}_{Vk}}{1 + \left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} \\
& - 2\alpha_V \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2\Delta\varepsilon_{VBk}}{1 + \left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} - 2\alpha_V \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk}}{1 + \left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} \\
& - 2\alpha_V \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}\varepsilon_{V\mathrm{N}}}{1 + \hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}} + 3\alpha_V^2 \frac{(L_Q + W_{V\mathrm{M}}L_{\sigma_V})^2\|\tilde{\boldsymbol{x}}_k\|^4}{1 + \left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} \\
& + 3\alpha_V^2 \frac{\Delta\varepsilon_{VBk}^2}{1 + \left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} + 2\alpha_V^2 \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{V\mathrm{N}}\Delta\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk}}{1 + \Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}} \\
& + 3\alpha_V^2 \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2\tilde{\boldsymbol{W}}_{Vk}}{1 + \left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} + 2\alpha_V^2 \frac{\varepsilon_{V\mathrm{N}}^2}{1 + \left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} \\
\le & -\frac{\alpha_V(1 - 6\alpha_V)}{2} \frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2\tilde{\boldsymbol{W}}_{Vk}}{1 + \left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} - \alpha_V(1 - 2\alpha_V)\frac{\tilde{\boldsymbol{W}}_{Vk}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}\hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk}}{1 + \hat{\sigma}_{V\mathrm{N}}^{\mathrm{T}}\hat{\sigma}_{V\mathrm{N}}} \\
& + \alpha_V(1 + 3\alpha_V)\frac{(L_Q + W_{V\mathrm{M}}L_{\sigma_V})^2\|\tilde{\boldsymbol{x}}_k\|^4}{1 + \left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2} + \varepsilon_{V\mathrm{TM}} \\
\le & -\frac{\alpha_V(1 - 6\alpha_V)}{2}\frac{\omega_{\min}^2}{1 + \omega_{\min}^2}\left\|\tilde{\boldsymbol{W}}_{Vk}\right\|^2 - \alpha_V(1 - 2\alpha_V)\frac{\sigma_{\min}^2}{1 + \sigma_{\min}^2}\left\|\tilde{\boldsymbol{W}}_{Vk}\right\|^2 \\
& + \frac{\alpha_V(1 + 3\alpha_V)}{1 + \omega_{\min}^2}(L_Q + W_{V\mathrm{M}}L_{\sigma_V})^2\|\tilde{\boldsymbol{x}}_k\|^4 + \varepsilon_{V\mathrm{TM}}
\end{aligned}
\tag{A.10}
$$

where $0 < \omega_{\min}^2 < \left\|\Delta\hat{\sigma}_{Vk} + \sigma_{Vk+1}{}' + \sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2$, and

$$\varepsilon_{V\mathrm{TM}} = \alpha_V(1+3\alpha_V)\frac{\Delta\varepsilon_{VBk}^2}{1+\left\|\Delta\hat{\sigma}_{Vk}+\sigma_{Vk+1}{}'+\sigma_{V\mathrm{M}}\boldsymbol{B}_l\right\|^2}+2W_{V\mathrm{M}}\sigma_{V\mathrm{M}}+\alpha_V(1+2\alpha_V)\frac{\varepsilon_{V\mathrm{N}}^2}{1+\Delta\hat{\sigma}_{Vk}^{\mathrm{T}}\Delta\hat{\sigma}_{Vk}}\,.$$

Next, consider $L(\tilde{\boldsymbol{x}}_k)$. Recall (A.3) and apply Cauchy-Schwartz inequality, the first difference of $L(\tilde{\boldsymbol{x}}_k)$ is given by

$$\begin{aligned}
\Delta L(\tilde{\boldsymbol{x}}_k) &= (\tilde{\boldsymbol{x}}_{k+1}^{\mathrm{T}}\tilde{\boldsymbol{x}}_{k+1})^2-(\tilde{\boldsymbol{x}}_k^{\mathrm{T}}\tilde{\boldsymbol{x}}_k)^2\\
&\le\left[-(1-\gamma)\left\|\tilde{\boldsymbol{x}}_k\right\|^2+3\left\|\tilde{\boldsymbol{W}}_k\right\|^2\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2+3\left\|\bar{\boldsymbol{\varepsilon}}_{Ok}\right\|^2\right]\times\\
&\quad\left[(1+\gamma)\left\|\tilde{\boldsymbol{x}}_k\right\|^2+3\left\|\tilde{\boldsymbol{W}}_k\right\|^2\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2+3\left\|\bar{\boldsymbol{\varepsilon}}_{Ok}\right\|^2\right]\\
&\le-(1-\gamma^2)\left\|\tilde{\boldsymbol{x}}_k\right\|^4+\left(3\left\|\tilde{\boldsymbol{W}}_k\right\|^2\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2+3\left\|\bar{\boldsymbol{\varepsilon}}_{Ok}\right\|^2\right)^2\\
&\quad+2\gamma\left\|\tilde{\boldsymbol{x}}_k\right\|^2\left(3\left\|\tilde{\boldsymbol{W}}_k\right\|^2\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2+3\left\|\bar{\boldsymbol{\varepsilon}}_{Ok}\right\|^2\right)\\
&\le-(1-2\gamma^2)\left\|\tilde{\boldsymbol{x}}_k\right\|^4+36\left\|\tilde{\boldsymbol{W}}_k\right\|^4\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^4+36\left\|\bar{\boldsymbol{\varepsilon}}_{Ok}\right\|^4
\end{aligned}\tag{A.11}$$

Next, take $L(\tilde{\boldsymbol{W}}_k)$. Recall (A.4) and write the difference $L(\tilde{\boldsymbol{W}}_k)$ as

$$\begin{aligned}
\Delta L(\tilde{\boldsymbol{W}}_k) &= (\mathrm{tr}\{\tilde{\boldsymbol{W}}_{k+1}^{\mathrm{T}}\boldsymbol{\Lambda}\tilde{\boldsymbol{W}}_{k+1}\})^2-(\mathrm{tr}\{\tilde{\boldsymbol{W}}_k^{\mathrm{T}}\boldsymbol{\Lambda}\tilde{\boldsymbol{W}}_k\})^2\\
&\le\{-(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\left\|\tilde{\boldsymbol{W}}_k\right\|^2-2\beta_{\mathrm{I}}\eta\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\right\|^2\left\|\tilde{\boldsymbol{W}}_k\right\|^2\\
&\quad+6\beta_{\mathrm{I}}^2\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2\left\|\boldsymbol{A}_c\right\|^2\left\|\boldsymbol{lC}\right\|^2\left\|\tilde{\boldsymbol{x}}_k\right\|^2+\varepsilon_{\mathrm{WM}}\}\times\\
&\quad\{(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\left\|\tilde{\boldsymbol{W}}_k\right\|^2-2\beta_{\mathrm{I}}\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2\left\|\tilde{\boldsymbol{W}}_k\right\|^2\\
&\quad+6\beta_{\mathrm{I}}^2\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2\left\|\boldsymbol{A}_c\right\|^2\left\|\boldsymbol{lC}\right\|^2\left\|\tilde{\boldsymbol{x}}_k\right\|^2+\varepsilon_{\mathrm{WM}}\}\\
&\le\{-(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\left\|\tilde{\boldsymbol{W}}_k\right\|^2-2\beta_{\mathrm{I}}\eta\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^4\left\|\tilde{\boldsymbol{W}}_k\right\|^2\\
&\quad+6\beta_{\mathrm{I}}^2\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2\left\|\boldsymbol{A}_c\right\|^2\left\|\boldsymbol{lC}\right\|^2\left\|\tilde{\boldsymbol{x}}_k\right\|^2+\varepsilon_{\mathrm{WM}}\}\times\\
&\quad\{(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\left\|\tilde{\boldsymbol{W}}_k\right\|^2-2\beta_{\mathrm{I}}\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\right\|^4\left\|\tilde{\boldsymbol{W}}_k\right\|^2\\
&\quad+6\beta_{\mathrm{I}}^2\Lambda\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\right\|^2\left\|\boldsymbol{A}_c\right\|^2\left\|\boldsymbol{lC}\right\|^2\left\|\tilde{\boldsymbol{x}}_k\right\|^2+\varepsilon_{\mathrm{WM}}\}
\end{aligned}$$

$$\begin{aligned}
&\le \{-(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\|\tilde{\boldsymbol{W}}_k\|^2 - 2\beta_{\mathrm{I}}\eta\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\|^4\|\tilde{\boldsymbol{W}}_k\|^2 \\
&\quad + 6\beta_{\mathrm{I}}^2\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\|^2\|\boldsymbol{A}_c\|^2\|\boldsymbol{lC}\|^2\|\tilde{\boldsymbol{x}}_k\|^2 + \varepsilon_{\mathrm{WM}}\}\times \\
&\quad \{(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\|\tilde{\boldsymbol{W}}_k\|^2 - 2\beta_{\mathrm{I}}\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^4\|\tilde{\boldsymbol{W}}_k\|^2 \\
&\quad + 6\beta_{\mathrm{I}}^2\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_k\|^2\|\boldsymbol{A}_c\|^2\|\boldsymbol{lC}\|^2\|\tilde{\boldsymbol{x}}_k\|^2 + \varepsilon_{\mathrm{WM}}\} \\
&\le -(1-8(1-\alpha_{\mathrm{I}})^4)\Lambda^2\|\tilde{\boldsymbol{W}}_k\|^4 - 8(1-\alpha_{\mathrm{I}})^2\beta_{\mathrm{I}}\eta\Lambda\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\|^4\|\tilde{\boldsymbol{W}}_k\|^2 \\
&\quad + 210\beta_{\mathrm{I}}^2\Lambda^2\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\|^4\|\boldsymbol{A}_c\|^4\|\boldsymbol{lC}\|^4\|\tilde{\boldsymbol{x}}_k\|^4 \\
&\quad + 12\beta_{\mathrm{I}}^2\eta^2\Lambda^2\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\|^8\|\tilde{\boldsymbol{W}}_k\|^4 + 5\varepsilon_{\mathrm{WM}}^2
\end{aligned}$$

$$\begin{aligned}
&\le -(1-8(1-\alpha_{\mathrm{I}})^4)\Lambda^2\|\tilde{\boldsymbol{W}}_k\|^4 - 4(2(1-\alpha_{\mathrm{I}})^2 - 3\eta)\beta_{\mathrm{I}}\eta\Lambda^2\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\|^4\|\tilde{\boldsymbol{W}}_k\|^4 \\
&\quad + 210\Lambda^2\|\boldsymbol{A}_c\|^4\|\boldsymbol{lC}\|^4\|\tilde{\boldsymbol{x}}_k\|^4 + 5\varepsilon_{\mathrm{WM}}^2
\end{aligned} \tag{A.12}$$

where $\eta = 2(1-\alpha_{\mathrm{I}})\lambda_{\min}(\boldsymbol{lC}) - \beta_{\mathrm{I}}\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\|^2$.

Therefore, combining (A.11) and (A.12) yields

$$\begin{aligned}
&\Delta L(\tilde{\boldsymbol{x}}_k) + \Delta L(\tilde{\boldsymbol{W}}_k) \\
&\le -(1-2\gamma^2)\|\tilde{\boldsymbol{x}}_k\|^4 + 36\|\tilde{\boldsymbol{W}}_k\|^4\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\|^4 + 36\|\bar{\boldsymbol{\varepsilon}}_{Ok}\|^4 \\
&\quad - (1-8(1-\alpha_{\mathrm{I}})^4)\Lambda^2\|\tilde{\boldsymbol{W}}_k\|^4 - 4(2(1-\alpha_{\mathrm{I}})^2 \\
&\quad - 3\eta)\beta_{\mathrm{I}}\eta\Lambda^2\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\|^4\|\tilde{\boldsymbol{W}}_k\|^4 + 52.5\Lambda^2\gamma^2\|\boldsymbol{A}_c\|^4\|\boldsymbol{lC}\|^4\|\tilde{\boldsymbol{x}}_k\|^4 + 5\varepsilon_{\mathrm{WM}}^2 \\
&\le -(1-3\gamma^2)\|\tilde{\boldsymbol{x}}_k\|^4 - 4\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\|^4\|\tilde{\boldsymbol{W}}_k\|^4 - (1-8(1-\alpha_{\mathrm{I}})^4)\Lambda^2\|\tilde{\boldsymbol{W}}_k\|^4 + \varepsilon_4
\end{aligned} \tag{A.13}$$

where $\varepsilon_4 = 36\|\bar{\boldsymbol{\varepsilon}}_{Ok}\|^4 + 5\varepsilon_{\mathrm{WM}}^2$.

Finally, combining (A.10) and (A.13) yields the first difference of total Lyapunov candidate as

$$\begin{aligned}\Delta L_{\tilde{\boldsymbol{W}}_V}(k) &= \Delta L(\tilde{\boldsymbol{W}}_{Vk}) + \Pi\Delta L(\tilde{\boldsymbol{x}}_k) + \Pi\Delta L(\tilde{\boldsymbol{W}}_k) \\ &\le -\frac{\alpha_V(1-6\alpha_V)}{2}\frac{\omega_{\min}^2}{1+\omega_{\min}^2}\left\|\tilde{\boldsymbol{W}}_{Vk}\right\|^2 - \alpha_V(1-2\alpha_V)\frac{\sigma_{\min}^2}{1+\sigma_{\min}^2}\left\|\tilde{\boldsymbol{W}}_{Vk}\right\|^2 \\ &\quad -\frac{\alpha_V(1+3\alpha_V)}{1+\omega_{\min}^2}(L_Q + W_{V\mathrm{M}}L_{\sigma_V})^2\left\|\tilde{\boldsymbol{x}}_k\right\|^4 - 4\Pi\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^4\left\|\tilde{\boldsymbol{W}}_k\right\|^4 \\ &\quad -(1-8(1-\alpha_{\mathrm{I}})^4)\Pi\Lambda^2\left\|\tilde{\boldsymbol{W}}_k\right\|^4 + \varepsilon_1\end{aligned} \tag{A.14}$$

where $\varepsilon_1 = \varepsilon_{V\mathrm{TM}} + \Pi\varepsilon_4$. By using standard Lyapunov stability analysis [22], ΔL is less than zero outside a compact set as long as the following conditions hold:

$$\left\|\tilde{\boldsymbol{x}}_k\right\| > \sqrt[4]{\frac{\varepsilon_1}{\frac{\alpha_V(1+3\alpha_V)}{1+\omega_{\min}^2}(L_Q + W_{V\mathrm{M}}L_{\sigma_V})^2}} \equiv b_{\tilde{\boldsymbol{x}}} \tag{A.15}$$

Or

$$\left\|\tilde{\boldsymbol{W}}_{Vk}\right\| > \sqrt{\frac{\varepsilon_1}{\frac{\alpha_V(1-6\alpha_V)}{2}\frac{\omega_{\min}^2}{1+\omega_{\min}^2} + \alpha_V(1-2\alpha_V)\frac{\sigma_{\min}^2}{1+\sigma_{\min}^2}}} \equiv b_{\tilde{\boldsymbol{W}}_V} \tag{A.16}$$

Or

$$\left\|\tilde{\boldsymbol{W}}_k\right\| > \sqrt[4]{\frac{\varepsilon_1}{4\Pi\chi_{\min}^4 + (1-8(1-\alpha_{\mathrm{I}})^4)\Pi\Lambda^2}} \equiv b_{\tilde{\boldsymbol{W}}} \tag{A.17}$$

Proof of Theorem 3: Consider the following Lyapunov candidate as

$$L(k) = L_{\boldsymbol{x}}(k) + L_{\mathrm{IO}}(k) + L_{\tilde{\boldsymbol{W}}_V}(k) + \frac{\Sigma}{2}L_{\tilde{\boldsymbol{W}}_{\boldsymbol{u}}}(k) + \frac{2^b\Sigma}{\sigma_{\boldsymbol{u}\mathrm{M}}}L_{e_u}(k) \tag{A.18}$$

where $L_{\mathrm{IO}}(k)$ and $L_{\tilde{\boldsymbol{W}}_V}(k)$ are defined in (A.1) and (A.8), respectively, $L_{\tilde{\boldsymbol{W}}_{\boldsymbol{u}}}(k) = \left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\|$ and $L_{e_u}(k) = e_{\mathrm{M}\boldsymbol{u},k}^2$ is the upper bound for the quantization error defined later. Moreover, the first term is defined as $L_{\boldsymbol{x}}(k)=\Xi\|\boldsymbol{x}_k\|$ with $\Xi=\frac{\alpha_{\boldsymbol{u}}\Sigma}{2\sigma_{\boldsymbol{u}\mathrm{M}}}\frac{\sigma^2_{\boldsymbol{u}\min}}{1+\sigma^2_{\boldsymbol{u}\min}}$. Denote $\sigma_{\boldsymbol{u}k}=\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,k)$, $\hat{\sigma}_{\boldsymbol{u}k}=\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)$, $\nabla\hat{\sigma}_{Vk+1}=\nabla_V(\hat{\boldsymbol{x}}_{k+1},k+1)$, $\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}=\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}}(\boldsymbol{x}_k,k)$, $\boldsymbol{g}_k^{\mathrm{T}}=g^{\mathrm{T}}(\boldsymbol{x}_k)$, $\hat{\boldsymbol{g}}_k^{\mathrm{T}}=g^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)$ and $\tilde{\boldsymbol{g}}_k^{\mathrm{T}}=\tilde{g}^{\mathrm{T}}(\hat{\boldsymbol{x}}_k)$ for simplicity, then the first difference of $L_{\tilde{\boldsymbol{W}}_{\boldsymbol{u}}}(k)$ recalling from (37) and (36), is given by

$$\begin{aligned}
\Delta L_{\tilde{\boldsymbol{W}}_{\boldsymbol{u}}}(k) &= \left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k+1}\right\|-\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\| \\
&\le \left(1-\alpha_{\boldsymbol{u}}\frac{\sigma^2_{\boldsymbol{u}\min}}{1+\sigma^2_{\boldsymbol{u}\min}}\right)\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\|+\alpha_{\boldsymbol{u}}\left\|\frac{L_\phi^2\boldsymbol{R}^{-2}g_{\mathrm{M}}^2\sigma_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}k}\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}^{\mathrm{T}}}{4(1+\sigma_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}k})^2}\right\| \\
&\quad +\alpha_{\boldsymbol{u}}\left(1+\frac{5}{4\left\|\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}\right\|}\right)\left\|\nabla\sigma_{Vk+1}^{\mathrm{T}}\tilde{\boldsymbol{W}}_{Vk}\right\|^2+\frac{\alpha_{\boldsymbol{u}}L_\phi^2\sigma_g^2}{4\left\|\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}\right\|}\left\|\tilde{\boldsymbol{W}}_k\right\|^2 \\
&\quad +\alpha_{\boldsymbol{u}}\left\|\frac{L_\phi^2\boldsymbol{R}^{-2}\sigma_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}k}\nabla\sigma_{Vk+1}^{\mathrm{T}}\nabla\sigma_{Vk+1}W_{V\mathrm{M}}^2\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}^{\mathrm{T}}}{4(1+\sigma_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}k})^2}\right\|+\alpha_{\boldsymbol{u}}\left\|\frac{L_\phi^2\boldsymbol{R}^{-2}g_{\mathrm{M}}^2\sigma_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}k}\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}^{\mathrm{T}}}{(1+\sigma_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}k})^2}\right\| \\
&\quad +\alpha_{\boldsymbol{u}}\left\|\frac{L_\phi^2\boldsymbol{R}^{-2}\sigma_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}k}}{4(1+\sigma_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}k})^2}\right\|\sigma_g^2\left\|\tilde{\boldsymbol{W}}_k\right\|^2+\alpha_{\boldsymbol{u}}\left\|\frac{\sigma_{\boldsymbol{u}k}^{\mathrm{T}}\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}^{\mathrm{T}}}{1+\sigma_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}k}}\right\|-\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\|^2 \\
&\le -\alpha_{\boldsymbol{u}}\frac{\sigma^2_{\boldsymbol{u}\min}}{1+\sigma^2_{\boldsymbol{u}\min}}\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\|+\alpha_{\boldsymbol{u}}\left(1+\frac{5}{4\left\|\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}\right\|}\right)\nabla\sigma_{V\mathrm{M}}^2\left\|\tilde{\boldsymbol{W}}_{Vk}\right\|^2 \\
&\quad +\alpha_{\boldsymbol{u}}L_\phi^2\sigma_g^2\left(\frac{1}{4\left\|\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}\right\|}+\frac{\lambda_{\max}(\boldsymbol{R}^{-1})}{4(1+\sigma^2_{\boldsymbol{u}\min})}\right)\left\|\tilde{\boldsymbol{W}}_k\right\|^2+\bar{\varepsilon}_{T\mathrm{M}}
\end{aligned} \tag{A.19}$$

where $\bar{\varepsilon}_{T\mathrm{M}}=\alpha_{\boldsymbol{u}}\left(\frac{5L_\phi^2\lambda_{\max}(\boldsymbol{R}^{-2})g_{\mathrm{M}}^2}{4(1+\sigma^2_{\boldsymbol{u}\min})}+\frac{L_\phi^2\lambda_{\max}(\boldsymbol{R}^{-2})\nabla\sigma_{V\mathrm{M}}^2W_{V\mathrm{M}}^2}{4(1+\sigma^2_{\boldsymbol{u}\min})}+\frac{\sigma_{\boldsymbol{u}\mathrm{M}}}{1+\sigma^2_{\boldsymbol{u}\min}}\right)\bar{\varepsilon}_{\boldsymbol{u}\mathrm{M}}$ and $0<\sigma_{\boldsymbol{u}\min}<\left\|\hat{\sigma}_{\boldsymbol{u}k}\right\|<\sigma_{\boldsymbol{u}\mathrm{M}}$.

Next, consider $L_{e_u}(k)$.

The control input is given as

$$\hat{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)=\hat{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)=\boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)-\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)$$

Then the quantization error bound is given by

$$\begin{aligned} e_{\mathrm{M}\boldsymbol{u},k} &= \frac{\left\|\boldsymbol{u}(\hat{\boldsymbol{x}}_k,k)\right\|}{2^b} = \frac{\left\|\boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\hat{\sigma}_{\boldsymbol{u}k} - \tilde{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\hat{\sigma}_{\boldsymbol{u}k}\right\|}{2^b} \\ &\le \frac{W_{\boldsymbol{u}\mathrm{M}}\sigma_{\boldsymbol{u}\mathrm{M}}}{2^b} + \frac{\sigma_{\boldsymbol{u}\mathrm{M}}}{2^b}\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\| \equiv e^2_{\mathrm{M}\boldsymbol{u},k} \end{aligned}$$

The first difference of $L_{e_u}(k)$ is given as

$$\begin{aligned} \Delta L_{e_u}(k) &= e^2_{\mathrm{M}\boldsymbol{u},k+1} - e^2_{\mathrm{M}\boldsymbol{u},k} \\ &= \frac{W_{\boldsymbol{u}\mathrm{M}}\sigma_{\boldsymbol{u}\mathrm{M}}}{2^b} + \frac{\sigma_{\boldsymbol{u}\mathrm{M}}}{2^b}\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k+1}\right\| - \left(\frac{W_{\boldsymbol{u}\mathrm{M}}\sigma_{\boldsymbol{u}\mathrm{M}}}{2^b} + \frac{\sigma_{\boldsymbol{u}\mathrm{M}}}{2^b}\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\|\right) \\ &= \frac{\sigma_{\boldsymbol{u}\mathrm{M}}}{2^b}\left(\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k+1}\right\| - \left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\|\right) \end{aligned} \tag{A.20}$$

Recalling from (A.19), we further have

$$\begin{aligned} \Delta L_{e_u}(k) &= \frac{\sigma_{\boldsymbol{u}\mathrm{M}}}{2^b}\left(\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k+1}\right\| - \left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\|\right) \\ &\le -\alpha_{\boldsymbol{u}}\frac{\sigma_{\boldsymbol{u}\mathrm{M}}}{2^b}\frac{\sigma^2_{\boldsymbol{u}\min}}{1+\sigma^2_{\boldsymbol{u}\min}}\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\| + \frac{\sigma_{\boldsymbol{u}\mathrm{M}}}{2^b}\bar{\boldsymbol{\varepsilon}}_{TM} + \alpha_{\boldsymbol{u}}\frac{\sigma_{\boldsymbol{u}\mathrm{M}}}{2^b}\left(1+\frac{5}{4\left\|\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}\right\|}\right)\nabla\sigma^2_{V\mathrm{M}}\left\|\tilde{\boldsymbol{W}}_{Vk}\right\|^2 \\ &\quad + \alpha_{\boldsymbol{u}}L_\phi^2\sigma_g^2\frac{\sigma_{\boldsymbol{u}\mathrm{M}}}{2^b}\left(\frac{1}{4\left\|\bar{\boldsymbol{\varepsilon}}_{\boldsymbol{u}k}\right\|} + \frac{\lambda_{\max}(\boldsymbol{R}^{-1})}{4(1+\sigma^2_{\boldsymbol{u}\min})}\right)\left\|\tilde{\boldsymbol{W}}_k\right\|^2 \end{aligned} \tag{A.21}$$

Combine (A.5), (A.14), (A.19) and (A.21) to obtain the first difference of the total Lyapunov candidate as

$$\begin{aligned} \Delta L(k) &= \Delta L_{\boldsymbol{x}}(k) + \Delta L_{\mathrm{IO}}(k) + \Delta L_{\tilde{\boldsymbol{W}}_V}(k) + \frac{\Sigma}{2}\Delta L_{\tilde{\boldsymbol{W}}_{\boldsymbol{u}}}(k) + \frac{2^b\Sigma}{\sigma_{\boldsymbol{u}\mathrm{M}}}\Delta L_{e_u}(k) \\ &\le \Xi\left\|f(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_k^* - g(\boldsymbol{x}_k)\boldsymbol{u}_k^* + g(\boldsymbol{x}_k)\hat{\boldsymbol{u}}_k\right\| - \Xi\left\|\boldsymbol{x}_k\right\| \\ &\quad - (1-(1+4\gamma\left\|\boldsymbol{lC}\right\|^2)(1+\chi^2_{\min})\gamma)\left\|\tilde{\boldsymbol{x}}_k\right\|^2 \\ &\quad - \left\|\tilde{\boldsymbol{W}}_k\right\|^2\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2 - \frac{1}{2}(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda\left\|\tilde{\boldsymbol{W}}_k\right\|^2 \\ &\quad - \frac{\alpha_V(1-6\alpha_V)}{2}\frac{\omega^2_{\min}}{1+\omega^2_{\min}}\left\|\tilde{\boldsymbol{W}}_{Vk}\right\|^2 - \frac{\alpha_V(1-2\alpha_V)}{2}\frac{\omega^2_{\min}}{1+\omega^2_{\min}}\left\|\tilde{\boldsymbol{W}}_{Vk}\right\|^2 \end{aligned}$$

$$
\begin{aligned}
&-\frac{\alpha_V(1+3\alpha_V)}{1+\omega_{\min}^2}(L_Q+W_{V\mathrm{M}}L_{\sigma_V})^2\left\|\tilde{\boldsymbol{x}}_k\right\|^4-4\Pi\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^4\left\|\tilde{\boldsymbol{W}}_k\right\|^4\\
&-(1-8(1-\alpha_\mathrm{I})^4)\Pi\Lambda^2\left\|\tilde{\boldsymbol{W}}_k\right\|^4-\alpha_{\boldsymbol{u}}\Sigma\frac{\sigma_{\boldsymbol{u}\min}^2}{1+\sigma_{\boldsymbol{u}\min}^2}\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\|+\varepsilon_{\mathrm{CLM}}\\
\le{}&\Xi\left\|f(\boldsymbol{x}_k)+g(\boldsymbol{x}_k)\boldsymbol{u}_k^*\right\|-\Xi\left\|\boldsymbol{x}_k\right\|+g_\mathrm{M}\Xi\left\|\boldsymbol{W}_{\boldsymbol{u}}^\mathrm{T}(\sigma_{\boldsymbol{u}k}-\hat{\sigma}_{\boldsymbol{u}k})+\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}^\mathrm{T}\right\|\\
&-(1-(1+4\gamma\left\|\boldsymbol{lC}\right\|^2)(1+\chi_{\min}^2)\gamma)\left\|\tilde{\boldsymbol{x}}_k\right\|^2\\
&-\left\|\tilde{\boldsymbol{W}}_k\right\|^2\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2-\frac{1}{2}(1-2(1-\alpha_\mathrm{I})^2)\Lambda\left\|\tilde{\boldsymbol{W}}_k\right\|^2\\
&-\frac{\alpha_V(1-6\alpha_V)}{2}\frac{\omega_{\min}^2}{1+\omega_{\min}^2}\left\|\tilde{\boldsymbol{W}}_{Vk}\right\|^2-\frac{\alpha_V(1-2\alpha_V)}{2}\frac{\omega_{\min}^2}{1+\omega_{\min}^2}\left\|\tilde{\boldsymbol{W}}_{Vk}\right\|^2\\
&-\frac{\alpha_V(1+3\alpha_V)}{1+\omega_{\min}^2}(L_Q+W_{V\mathrm{M}}L_{\sigma_V})^2\left\|\tilde{\boldsymbol{x}}_k\right\|^4-4\Pi\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^4\left\|\tilde{\boldsymbol{W}}_k\right\|^4\\
&-(1-8(1-\alpha_\mathrm{I})^4)\Pi\Lambda^2\left\|\tilde{\boldsymbol{W}}_k\right\|^4-\alpha_{\boldsymbol{u}}\Sigma\frac{\sigma_{\boldsymbol{u}\min}^2}{1+\sigma_{\boldsymbol{u}\min}^2}\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\|+\varepsilon_{\mathrm{CLM}}
\end{aligned}
$$

$$
\begin{aligned}
\le{}&-(1-\rho)\Xi\left\|\boldsymbol{x}_k\right\|-(1-(1+4\gamma\left\|\boldsymbol{lC}\right\|^2)(1+\chi_{\min}^2)\gamma)\left\|\tilde{\boldsymbol{x}}_k\right\|^2\\
&-\left\|\tilde{\boldsymbol{W}}_k\right\|^2\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^2-\frac{1}{2}(1-2(1-\alpha_\mathrm{I})^2)\Lambda\left\|\tilde{\boldsymbol{W}}_k\right\|^2\\
&-\frac{\alpha_V(1-6\alpha_V)}{2}\frac{\omega_{\min}^2}{1+\omega_{\min}^2}\left\|\tilde{\boldsymbol{W}}_{Vk}\right\|^2-\frac{\alpha_V(1-2\alpha_V)}{2}\frac{\omega_{\min}^2}{1+\omega_{\min}^2}\left\|\tilde{\boldsymbol{W}}_{Vk}\right\|^2\\
&-\frac{\alpha_V(1+3\alpha_V)}{1+\omega_{\min}^2}(L_Q+W_{V\mathrm{M}}L_{\sigma_V})^2\left\|\tilde{\boldsymbol{x}}_k\right\|^4-4\Pi\left\|\sigma(\hat{\boldsymbol{x}}_k)\bar{\boldsymbol{u}}_{qk}\right\|^4\left\|\tilde{\boldsymbol{W}}_k\right\|^4\\
&-(1-8(1-\alpha_\mathrm{I})^4)\Pi\Lambda^2\left\|\tilde{\boldsymbol{W}}_k\right\|^4-\alpha_{\boldsymbol{u}}\Sigma\frac{\sigma_{\boldsymbol{u}\min}^2}{1+\sigma_{\boldsymbol{u}\min}^2}\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\|+\varepsilon_{\mathrm{CLM}}
\end{aligned}
$$

where $\varepsilon_{\mathrm{CLM}}=\varepsilon_{\mathrm{OM}}+\varepsilon_1+\bar{\varepsilon}_{T\mathrm{M}}+g_\mathrm{M}W_{\boldsymbol{u}\mathrm{M}}\sigma_{\boldsymbol{u}\mathrm{M}}\Xi$.

By using standard Lyapunov stability analysis [22], ΔL is less than zero outside a compact set as long as the following conditions hold:

$$
\left\|\boldsymbol{x}_k\right\|>\frac{2\varepsilon_{\mathrm{CLM}}}{(1-\rho)\Xi}\equiv b_{\boldsymbol{x}} \qquad \text{(A.22)}
$$

Or

$$\left\|\tilde{\boldsymbol{x}}_k\right\| > \min\left\{\begin{array}{l}\sqrt{\dfrac{\varepsilon_{\mathrm{CLM}}}{1-(1+4\gamma\left\|\boldsymbol{lC}\right\|^2)(1+\chi_{\min}^2)\gamma}}, \\ \sqrt[4]{\dfrac{\varepsilon_{\mathrm{CLM}}}{\dfrac{\alpha_V(1+3\alpha_V)}{1+\omega_{\min}^2}(L_Q+W_{V\mathrm{M}}L_{\sigma_V})^2}}\end{array}\right\} \equiv b_{\tilde{\boldsymbol{x}}} \tag{A.23}$$

Or

$$\left\|\tilde{\boldsymbol{W}}_{Vk}\right\| > \sqrt{\dfrac{\varepsilon_{\mathrm{CLM}}}{\dfrac{\alpha_V(1-6\alpha_V)}{2}\dfrac{\omega_{\min}^2}{1+\omega_{\min}^2}+\alpha_V(1-2\alpha_V)\dfrac{\omega_{\min}^2}{1+\omega_{\min}^2}}} \equiv b_{\tilde{\boldsymbol{W}}_V} \tag{A.24}$$

Or

$$\left\|\tilde{\boldsymbol{W}}_k\right\| > \min\left\{\begin{array}{l}\sqrt{\dfrac{\varepsilon_{\mathrm{CLM}}}{\chi_{\min}^2+\dfrac{1}{2}(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda}}, \\ \sqrt[4]{\dfrac{\varepsilon_{\mathrm{CLM}}}{4\Pi\chi_{\min}^4+(1-8(1-\alpha_{\mathrm{I}})^4)\Pi\Lambda^2}}\end{array}\right\} \equiv b_{\tilde{\boldsymbol{W}}} \tag{A.25}$$

Or

$$\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}\right\| > \dfrac{\varepsilon_{\mathrm{CLM}}}{\alpha_{\boldsymbol{u}}\Sigma\dfrac{\sigma_{\boldsymbol{u}\min}^2}{1+\sigma_{\boldsymbol{u}\min}^2}} \equiv b_{\tilde{\boldsymbol{W}}_{\boldsymbol{u}}} \tag{A.26}$$

where $\Sigma = \min\left\{\begin{array}{l}\dfrac{\alpha_V(1-2\alpha_V)\sigma_{\min}^2\bar{\varepsilon}_{\boldsymbol{u}\mathrm{M}}}{\alpha_{\boldsymbol{u}}(5+4\bar{\varepsilon}_{\boldsymbol{u}\mathrm{M}})\nabla\sigma_{V\mathrm{M}}^2\sigma_{\min}^2+2\bar{\varepsilon}_{\boldsymbol{u}\mathrm{M}}}, \\ \dfrac{(1-2(1-\alpha_{\mathrm{I}})^2)\Lambda(1+\sigma_{\min}^2)\bar{\varepsilon}_{\boldsymbol{u}\mathrm{M}}}{2\alpha_{\boldsymbol{u}}\sigma_g^2L_\phi^2(1+\sigma_{\min}^2+\bar{\varepsilon}_{\boldsymbol{u}\mathrm{M}}\lambda_{\max}(\boldsymbol{R}^{-1}))}\end{array}\right\}.$

Eventually, recall to (A.21), the difference between the ideal optimal control and proposed near optimal control inputs is represented as

$$\begin{aligned}
&\left\|\boldsymbol{u}^*(\boldsymbol{x}_k,k)-\hat{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)\right\| \\
&=\left\|\boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\boldsymbol{x}_k,k)+\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k,k)-\hat{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)\right\| \\
&=\left\|\tilde{\boldsymbol{W}}_{\boldsymbol{u}k}^{\mathrm{T}}\sigma_{\boldsymbol{u}}(\hat{\boldsymbol{x}}_k,k)+\boldsymbol{W}_{\boldsymbol{u}}^{\mathrm{T}}\tilde{\sigma}_{\boldsymbol{u}}(\boldsymbol{x}_k,\hat{\boldsymbol{x}}_k,k)+\varepsilon_{\boldsymbol{u}}(\boldsymbol{x}_k,k)\right\| \\
&\le b_{\tilde{\boldsymbol{W}}_{\boldsymbol{u}}}\sigma_{\boldsymbol{u}\mathrm{M}}+W_{\boldsymbol{u}\mathrm{M}}\left\|\tilde{\sigma}_{\boldsymbol{u}}(\boldsymbol{x}_k,\hat{\boldsymbol{x}}_k,k)\right\|+\varepsilon_{\boldsymbol{u}\mathrm{M}} \\
&\le b_{\tilde{\boldsymbol{W}}_{\boldsymbol{u}}}\sigma_{\boldsymbol{u}\mathrm{M}}+l_\sigma W_{\boldsymbol{u}\mathrm{M}}\left\|\tilde{\boldsymbol{x}}_k\right\|+\varepsilon_{\boldsymbol{u}\mathrm{M}} \\
&\le b_{\tilde{\boldsymbol{W}}_{\boldsymbol{u}}}\sigma_{\boldsymbol{u}\mathrm{M}}+l_\sigma W_{\boldsymbol{u}\mathrm{M}}b_{\tilde{\boldsymbol{x}}}+\varepsilon_{\boldsymbol{u}\mathrm{M}}\equiv\varepsilon_{\boldsymbol{u}o}
\end{aligned} \tag{A.27}$$

where l_σ is the Lipschitz constant of $\sigma_{\boldsymbol{u}}(\bullet)$, and $b_{\tilde{\boldsymbol{W}}_{\boldsymbol{u}}}$, $b_{\tilde{\boldsymbol{x}}}$ are given in (A.26) and (A.23), respectively.

SECTION

2. CONLUSIONS AND FUTURE WORK

In this dissertation, finite-horizon optimal regulation problem is considered for linear and a class of nonlinear discrete-time systems. Time-dependency aspect of the optimal solution and terminal constraint are two major concerns in finite-horizon optimal control which are handled by using novel parameter update laws with time dependent basis functions. Linear in the unknown parameter adaptive control and neural network (NN) based schemes are considered to deal with linear and nonlinear systems, respectively. A Q-learning methodology is utilized for the case of linear systems and NN identifier/observer is proposed for nonlinear systems so that the requirements on dynamics of the system and the system states are relaxed. The five papers included in this dissertation address (near) optimal regulation of both linear and nonlinear systems.

2.1 CONCLUSIONS

The first paper addresses the finite-horizon optimal control of linear discrete-time systems with completely unknown system dynamics by using approximate dynamic programming technique. The requirement on the dynamics of the system is relaxed with an adaptive estimator generating the Q-function. An online adaptive estimator learns time-varying optimal control gain provided by the tuned Q-function by using history information thus relaxing the need for policy and/or value iterations. An additional error is defined and incorporated in the update law so that the terminal constraint for the finite-horizon can be properly satisfied. An initial admissible control ensures the stability of the system. In addition, the proposed control design scheme is extended to output feedback case by novel adaptive observer design. All the parameters are tuned in an online and forward-in-time manner. Stability of the overall closed-loop system is demonstrated by Lyapunov analysis. The proposed approach yields a forward-in-time and online control design scheme which offers many practical benefits.

The second paper investigated the adaptive finite-horizon optimal regulation design for unknown linear discrete-time control systems under the quantization effects for both system states and control inputs. By introducing a new scaling parameter and analyzing the quantization error bound, the proposed dynamic quantizer design effectively eliminated the saturation effect as well as the quantization error. The system dynamics are not needed with an adaptive estimator generating the action-dependent value function, and a novel update law different from the first paper was considered to tune the value function estimator which then was used to calculate the Kalman gain needed for the optimal control policy. By minimizing the Bellman and terminal constraint errors simultaneously once a sampling interval, the update law functions in a forward-in-time fashion without performing iterations while satisfying the terminal constraint. Lyapunov theory demonstrated the effectiveness of the proposed scheme.

In the third paper, we considered the finite-horizon optimal control problem of affine nonlinear discrete-time systems. With a novel NN-based identifier, the complete system dynamics were relaxed in contrast with the literature where the control coefficient matrix was needed. An initial admissible control policy guarantees that the system is stable, while actor-critic structure is utilized to approximately find the optimal control input. The time-dependency nature for finite-horizon optimal control problem is handled by using novel NN structure with constant weights and time-varying activation functions, while an additional error term corresponding to the terminal constraint is minimized to guarantee that the terminal constraint can be properly satisfied. In addition, the proposed algorithm is implemented by utilizing a history of cost to go errors instead of traditional iteration-based scheme. As a consequence, the proposed design scheme performs in an online and forward-in-time fashion which is highly suitable for real-time implementation. The convergence of the parameter estimation and closed-loop system are demonstrated by using Lyapunov stability theory under non-autonomous analysis.

In the fourth paper, the idea from Paper III is extended to the output feedback case. The novel structure of the proposed observer relaxes the need for a separate identifier thus simplifies the overall design. Time-dependency nature of the finite-horizon is handled by a NN structure with constant weights and time-varying activation function. The terminal constraint is properly satisfied by minimizing an additional error term along the system trajectory. All NN weights are tuned online by using proposed update laws and Lyapunov stability theory demonstrated that the approximated control inputs converges close to its optimal value as time evolves. Compared to the traditional finite-horizon optimal regulation design, the proposed scheme not only relaxes the requirement on availability of the system states and control coefficient matrix, but also functions in an online and forward-in-time manner instead of performing offline training and value/policy iteration.

Finally, the fifth paper presents the finite-horizon near optimal regulation of general discrete-time nonlinear systems in affine form in the presence of quantization and input constraints. A non-quadratic cost functional incorporates the effect of actuator saturation while still guaranteeing the optimality of the system. The quantization error for the control inputs is effectively mitigated by the design of a dynamic quantizer from the Paper II, while an extended NN-based Luenberger observer from Paper IV relaxes the need for an additional identifier thus simplifying the overall design. The actor-critic structure ensures that the newly defined time-varying value function and control inputs by using NN with constant weights and time-dependent activation functions indeed generate optimal control. Terminal constraint is properly satisfied by minimizing an error term corresponding to the terminal constraint along the system trajectory. Lyapunov stability theory demonstrated that the approximated control input converges close to its optimal value as time evolves.

2.2 FUTURE WORK

As part of the future work, our proposed finite-horizon optimal control scheme can be possibly improved by more carefully considering about the fundamental concepts of finite-horizon optimal control and approximation theory. The work presented in this dissertation is still a starting point for finite-horizon optimal control problem. Further research such as how the convergence rate is affected by the terminal time can be a promising direction. In addition, there's no general rule for picking the most suitable activation function for NN approximation, especially when the function to be approximated becomes time-varying. More detailed investigation in properly selecting the activation function can be more difficult however worth of our effort in the future. Finally, a more general nonlinear system description, i.e., nonlinear systems in non-affine form, can also be another potential topic to further extend our work.

VITA

Qiming Zhao was born on January 28, 1985 in Xi’an, China. He earned the Bachelor of Science degree in Electrical Engineering from Central South University and Master of Science degree in Electrical Engineering from Northwestern Polytechnical University, China, in 2007 and 2010, respectively. He joined Missouri University of Science and Technology (formerly the University of Missouri-Rolla) in January 2011 and received the degree of Doctor of Philosophy in Electrical Engineering in December 2013.