IET Control Theory & Applications

Research Article

IET Journals
The Institution of Engineering and Technology

H_∞ tracking control via variable gain gradient descent-based integral reinforcement learning for unknown continuous time non-linear system

ISSN 1751-8644
Received on 27th January 2020
Revised 3rd November 2020
Accepted on 25th November 2020
E-First on 22nd February 2021
doi: 10.1049/iet-cta.2020.0098
www.ietdl.org

Amardeep Mishra[1] ✉, *Satadal Ghosh*[1]

[1]*Department of Aerospace Engineering, IIT Madras, Chennai 600036, India*

✉ *E-mail: ae15d405@smail.iitm.ac.in*

Abstract: Optimal tracking of continuous-time non-linear systems has been extensively studied in literature. However, in several applications, absence of knowledge about system dynamics poses a severe challenge in solving the optimal tracking problem. This has found growing attention among researchers recently, and integral reinforcement learning based method augmented with actor neural network (NN) have been deployed to this end. However, very few studies have been directed to model-free H_∞ optimal tracking control that helps in attenuating the effect of disturbances on the system performance without any prior knowledge about system dynamics. To this end, a recursive least square-based parameter update was recently proposed. However, gradient descent-based parameter update scheme is more sensitive to real-time variation in plant dynamics. Experience replay (ER) technique has been shown to improve the convergence of NN weights by utilising past observations iteratively. Motivated by these, this study presents a novel parameter update law based on variable gain gradient descent and ER technique for tuning the weights of critic, actor and disturbance NNs. The presented update law leads to improved model-free tracking performance under $\mathscr{L}_2$-bounded disturbance. Simulation results are presented to validate the presented update law.

1 Introduction

Optimal control is one of the prominent control techniques that aims to find control policies that minimises a cost function subjected to plant dynamics as constraints. Traditional optimal control techniques require full knowledge of plant dynamics and corresponding parameters for their implementation. However, in practice knowledge of the same might be partially available or unavailable. In order to implement optimal control methods online under such limitations, reinforcement learning (RL) [1, 2] and adaptive dynamic programming (ADP) [3, 4] approaches were proposed that solve optimal control problem forward in time.

Regulation problems and trajectory tracking problems are the two broad classifications of the optimal control problem. The prime objective of regulation problems [5–11] is to find a control policy that brings the desired states to origin in finite amount of time while minimising a cost function. On the other hand, optimal tracking control problem (OTCP) [12–15] entails finding control policies that will make the desired states (output of the system) track a time varying reference trajectory. Traditionally, the OTCP requires development of two different controllers: (i) transient controller and (ii) steady-state control [4, 15]. Limitation of traditional OTCP solving schemes lies in the requirement of (i) knowledge of reference dynamics and (ii) invertibility condition on control gain matrix. Modares *et al.* [16, 17] proposed augmented system comprising of error and desired dynamics to by-pass this limitation. Finding the control policy that stabilises the augmented system while minimising the performance index was the prime objective of their novel control algorithm. The control policy generated by their algorithm also consisted of both transient and steady-state controllers.

In the ADP schemes mentioned above, identifiers were used to obviate the exact knowledge of nominal plant dynamics. However, identifiers add to the computational complexity and also reduce the accuracy of the computations [18]. In most cases, identifiers also require the knowledge of structure of the plant dynamics. Hence, efforts have been devoted to make RL schemes either partially model-free or completely model-free. In order to develop continuous-time optimal control policies under partial or no knowledge of plant dynamics integral reinforcement learning (IRL) algorithm was leveraged. While first few results in this direction for regulation problem were presented in [19–22], Modares *et al.* [16] developed algorithms for OTCP for partially-unknown system. Thereafter, Zhu *et al.* [18] developed off-policy model-free tracking control of continuous-time non-linear systems using IRL. They leveraged experience replay (ER) technique to effectively utilise past observations in order to update the NN weights. Further, the uniform ultimate boundedness (UUB) stability of the update law was also proved.

It may also be noted that most of the aforementioned RL schemes, for both regulation and tracking problems, do not deal with attenuation of the effects of disturbance. To this end, H_∞ regulation problem has been studied using RL both offline [23, 24] and online [25, 26]. Online IRL was also utilised in [27, 28] for H_∞ regulation problem for partially-unknown system. Note that under partial or no knowledge of the plant dynamics structure, IRL has been leveraged in several literature for regulation problem, while very few studies have dealt with IRL for OTCP problem with disturbance rejection. To the best of the authors' knowledge, [29, 30] are the only few papers that have recently presented control policies for model-free OTCP of continuous-time non-linear system in H_∞ framework. Modares *et al.* [29] updated the parameters of critic, actor and disturbance NNs using least square method, which could only be initiated after certain number of data had been collected. This makes their algorithm less sensitive to real-time variations in plant dynamics [18]. On the other hand, Zhang *et al.* [30] utilised gradient descent driven parameter update law for H_∞ tracking control problem, and UUB stability of the parameter update law was proven. However, their gradient descent followed a constant learning rate.

While continuous-time update law driven by gradient descent is more sensitive to real-time variations in plant dynamics, ER technique has been shown to improve the learning speed significantly by utilising past observations iteratively [18]. Also, addition of 'robust terms' in update law was shown to shrink the

residual set in [31]. Inspired by [18, 29], this paper presents a novel off-policy IRL-based H_∞ tracking control scheme for continuous-time non-linear system, in which the parameter update laws for tuning the weights of critic, actor and disturbance NNs are driven by variable gain gradient descent and ER technique in addition to robust terms. Instead of a constant learning rate in traditional gradient descent-based schemes, the learning rate of gradient descent developed in this paper is variable and a function of Hamilton–Jacobi–Isaac (HJI) error. This results in an increased learning rate when HJI error is large and the learning rate is reduced as the HJI error becomes smaller. The variable gain gradient descent technique is also shown to have the added advantage of shrinking the size of the residual sets, which the NN weights finally converge to. Term corresponding to ER technique and robust term in the update law also contribute in further shrinking the size of the residual set. Unlike [31], the update law presented in this paper, not only leverages robust terms for present instance but also past instances as well. After the completion of learning phase, the final learnt policies leveraging variable gain gradient descent are executed and they are shown to reduce the oscillations in transient phase and steady-state errors, thus resulting in improved tracking performance. Thus, the key innovation of this paper can be summarised as the proposition and analysis of continuous-time parameter update law, which is driven by variable gain gradient descent and augmented with ER and robust term, to tune NN weights to solve H_∞ tracking control problem with disturbance rejection for completely unknown continuous-time non-linear system.

The rest of the paper is structured as follows. Preliminaries and background of H_∞ tracking controller and the tracking HJI equation for augmented system have been presented in Section 2. Next, in order to obviate the requirement of system dynamics in policy evaluation step, model-free version of HJI equation to formulate IRL, and neural networks (NNs) to approximate value function, control and disturbance policies are presented in Section 3. Section 4 highlights the main contribution of this paper, i.e. the continuous-time weight update law that is driven by variable gain gradient descent and ER technique. The update law also incorporates the robust terms and their past observations. UUB stability analysis for the proposed mechanism is shown. Numerical studies are presented in Section 5 to justify the effectiveness of the presented algorithm. Finally, Section 6 provides concluding remarks.

2 H_∞ tracking problem and HJI equation

2.1 Problem formulation

It is desired to drive certain states of interest of the dynamical system to follow predefined reference trajectories under $\mathscr{L}_2$-bounded disturbance in an optimal way. Let the dynamical system be described by an affine-in-control differential equation

$$\dot{x} = f(x) + g(x)u + k(x)d \tag{1}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $d \in \mathbb{R}^{q_1}$, $f(x): \mathbb{R}^n \to \mathbb{R}^n$ is the drift dynamics, $g(x): \mathbb{R}^n \to \mathbb{R}^{n\times m}$ represents the control coupling dynamics and $k(x): \mathbb{R}^n \to \mathbb{R}^{n\times q_1}$ is disturbance dynamics. In the subsequent analysis in this paper, it is assumed that none of the system dynamics, that is $f(x), g(x)$ and $k(x)$, are known. However, Lipschitz continuity for the system dynamics as well as controllability of the system over a compact set $\Omega \in \mathbb{R}^n$ are assumed. A bounded reference trajectory is generated by a command generator or a reference system whose dynamics is described by

$$\dot{x}_d = \eta(x_d) \tag{2}$$

Thus, the error is given by

$$e = x - x_d \tag{3}$$

Therefore, the error dynamics is given as

$$\dot{e} = (f(x) + g(x)u + k(x)d - \eta(x_d)) \tag{4}$$

In order to formulate corresponding HJI equation and assess the effect of disturbance on the closed-loop system, a virtual performance index ($|X|^2$) is defined as [29]

$$|X|^2 = e^{\mathrm{T}}Qe + u^{\mathrm{T}}Ru \tag{5}$$

where Q and R are positive definite matrices with only diagonal entries. It is to be noted that all the vector or matrix norms used in this paper are 2-norm or the Eucledian norm. In [29], the disturbance attenuation condition was characterised as the $\mathscr{L}_2$-gain is smaller than or equal to α for all $d \in L_2[0, \infty)$, that is

$$\frac{\int_t^\infty e^{-\gamma(\tau-t)}|X|^2 d\tau}{\int_t^\infty e^{-\gamma(\tau-t)}|d(\tau)|^2 d\tau} \le \alpha^2 \tag{6}$$

where $0 \le \gamma$ is the discount factor and α determines the degree of attenuation from disturbance input to the virtual performance measure. The value of α is selected based on trial and error. The minimum value of α, for which (6) is satisfied provides optimal-robust control solution [29]. Now, using (5) and (6)

$$\int_t^\infty e^{-\gamma(\tau-t)}(e^{\mathrm{T}}Qe + u^{\mathrm{T}}Ru)\mathrm{d}\tau \le \alpha^2 \int_t^\infty e^{-\gamma(\tau-t)} \parallel d(\tau) \parallel^2 \mathrm{d}\tau \tag{7}$$

Finding a control policy u dependent on tracking error and reference trajectory such that the system dynamics (1) satisfies the disturbance attenuation condition (7) and that the error dynamics (4) is locally asymptotically stable for $d = 0$ forms H_∞ tracking control problem [29].

2.2 HJI equation: preliminaries

The first part of this section deals with the development of HJI equation for solving the H_∞ tracking problem stated above, while the second part discusses about policy iteration steps. As discussed in [29], the H_∞ tracking problem can also be posed as a min–max optimisation problem subjected to augmented system dynamics comprising of error dynamics and desired states dynamics. Subsequently, the solution to min–max optimisation problem is obtained by imposing the stationarity condition on the Hamiltonian. In order to formulate tracking HJI equation, an augmented state vector is defined as

$$z = [e^{\mathrm{T}}, x_d^{\mathrm{T}}]^{\mathrm{T}} \tag{8}$$

The augmented system dynamics is then given as

$$\dot{z} = F(z) + G(z)u + K(z)d \tag{9}$$

Where

$$F(z) = \begin{pmatrix} (f(x) - \eta(x_d)) \\ \eta(x_d) \end{pmatrix},\ G(z) = \begin{pmatrix} g(x) \\ 0 \end{pmatrix},\ K(z) = \begin{pmatrix} k(x) \\ 0 \end{pmatrix} \tag{10}$$

denote augmented drift, control coupling and disturbance dynamics, respectively. In the subsequent analysis, $F \triangleq F(z)$, $G \triangleq G(z)$ and $K \triangleq K(z)$. From the Lipschitz continuity of the original system and the boundedness of the reference signal it can be concluded that the augmented system is also Lipschitz continuous in $z \in \Omega_1 \subset \mathbb{R}^{2n}$, where Ω_1 is a compact set. Using augmented states, the attenuation condition (7) can be described as

$$\int_t^\infty e^{-\gamma(\tau-t)}(z^{\mathrm{T}}Q_1 z + u^{\mathrm{T}}Ru)\mathrm{d}\tau \le \alpha^2 \int_t^\infty e^{-\gamma(\tau-t)}(\parallel d(\tau) \parallel^2)\mathrm{d}\tau \tag{11}$$

where, Q_1 is given by

$$Q_1 = \begin{pmatrix} Q_{n\times n} & 0_{n\times n} \\ 0_{n\times n} & 0_{n\times n} \end{pmatrix}_{2n\times 2n} \tag{12}$$

and R is penalty matrix on control that has positive diagonal entries. Thus, a final performance index consisting of disturbance input is defined as

$$J(u,d) = \int_t^\infty e^{-\gamma(\tau-t)}\big(z(\tau)^{\mathrm{T}}Q_1 z(\tau) + u(\tau)^{\mathrm{T}}Ru(\tau) - \alpha^2 \parallel d(\tau) \parallel^2\big)\mathrm{d}\tau \tag{13}$$

Let the value function V be defined as $V \triangleq J(u,d)$. Based on the definition of J in (13), the value function V also depends on augmented states trajectory z for given control policy u and disturbance d. The gradient of value function ($\nabla_z V$) along the augmented system trajectory z plays a key role in subsequent discussion. Note, ∇V, V_z and $\nabla_z V$ are used equivalently in the paper. The problem of finding control input u that satisfies (6) is same as minimising (13) subjected to augmented dynamics. In [32], a direct relationship between H_∞ control problem and two-player zero-sum differential game was established. It was shown that solution of the H_∞ control problem is equivalent to solution of the following zero-sum game:

$$V^*(z) = J(u^*, d^*) = \min_u \max_d J(u,d) \tag{14}$$

The term J is as defined in (13) and V^* is optimal value function. Existence of game theoretic saddle point was also shown to guarantee the existence of solution of the two-player zero-sum game control problem. This is encapsulated in following Nash condition:

$$V^*(z) = \min_u \max_d J(u,d) = \max_d \min_u J(u,d) \tag{15}$$

Differentiating (13) along augmented system trajectories, the following Bellman equation is obtained:

$$z^{\mathrm{T}}Q_1 z + u^{\mathrm{T}}Ru - \alpha^2 d^{\mathrm{T}}d - \gamma V + \nabla V^{\mathrm{T}}(F + Gu + Kd) = 0 \tag{16}$$

where ∇V is the gradient of cost with respect to augmented states (z). Let the Hamiltonian be defined as

$$\mathscr{H}(z,V,u,d) = z^{\mathrm{T}}Q_1 z + u^{\mathrm{T}}Ru - \alpha^2 d^{\mathrm{T}}d - \gamma V + \nabla V^{\mathrm{T}}(F + Gu + Kd) \tag{17}$$

V^* being the optimal cost, satisfies the Bellman equation. Applying stationarity condition on the Hamiltonian, both optimal control input and disturbance input are obtained as follows:

$$\frac{\partial \mathscr{H}(z,V^*,u,d)}{\partial u} = 0 \implies u^* = -\frac{1}{2}R^{-1}G^{\mathrm{T}}\nabla V^*$$
$$\frac{\partial \mathscr{H}(z,V^*,u,d)}{\partial d} = 0 \implies d^* = \frac{1}{2\alpha^2}K^{\mathrm{T}}\nabla V^* \tag{18}$$

where ∇V^* is the gradient of the optimal cost with respect to augmented states (z). The optimal control input and disturbance input given above provide saddle point solution to the game [23]. Using (18) in (17) tracking HJI equation is

$$z^{\mathrm{T}}Q_1 z + V_z^{*T}F - \gamma V^* - \frac{1}{4}\nabla V^{*T}G^{\mathrm{T}}R^{-1}G\nabla V^* + \frac{1}{4\alpha^2}\nabla V^{*T}KK^{\mathrm{T}}\nabla V^* = \mathscr{H}(z,V^*,u^*,d^*) = 0 \tag{19}$$

Policy iteration framework is a computation approach to iteratively solve the Bellman equation and improve the control policies. It is generally started off with some known initial stabilising policy u and then following two steps are iteratively repeated till convergence is achieved.

(i) Policy evaluation: Given initial admissible control and disturbance policies, this step entails solving the Bellman equation as (where V_i, u_i, d_i denote improved value function and policies at i th iteration)

$$\underbrace{\nabla V_i^{\mathrm{T}}(F + Gu_i + Kd_i)}_{\dot{V}_i|_{u_i,d_i}} = \gamma V_i - z^{\mathrm{T}}Q_1 z - u_i^{\mathrm{T}}Ru_i + \alpha^2 d_i^{\mathrm{T}}d_i \tag{20}$$

(ii) Policy improvement: This step produces improved control and disturbance policies

$$u_{i+1} = -\frac{1}{2}R^{-1}G^{\mathrm{T}}\nabla V_i;\ d_{i+1} = \frac{1}{2\alpha^2}K^{\mathrm{T}}\nabla V_i \tag{21}$$

Theorem 1: For an admissible policy (u_i, d_i), if V_i is the solution of (20), satisfying the boundary condition $V_i(0) = 0$, then the improved policies, i.e. u_{i+1}, d_{i+1} are also admissible. Additionally, if V_{i+1} is unique positive definite function satisfying Bellman equation (20) with $V_{i+1}(0) = 0$, then $V^* \le V_{i+1} \le V_i$.

Proof: The proof of this theorem is provided in the Appendix (Section 8.2).□

Theorem 2: Given an initial admissible policy u_i, d_i, then according to Theorem 1, the improved policies (u_{i+1}, d_{i+1}) are also admissible, additionally, $V_{i+1} \to V^*$, $u_{i+1} \to u^*$ and $d_{i+1} \to d^*$ on a compact set Ω_1 (where Ω_1 is as defined after (10)).

Proof: The proof of the convergence of actor and disturbance policies and the value function to their respective optima follows using similar methodology as in the proofs of Theorem 1 of [6] or Theorem 2 of [33]. □

3 Integral reinforcement learning and value function approximation

3.1 Derivation of model-free HJI equation

Note that implementation of traditional policy iteration algorithms [i.e. Equations (20) and (21)] requires complete knowledge of system dynamics. Hence, in order to completely remove the requirement of system dynamics from policy evaluation step (i.e. (20)), IRL [34] will be leveraged in the following way. In the subsequent analysis, u refers to the executed control policy and d refers to the disturbance present in the system. It is assumed that an initial admissible policy is known. Improved policies on the other hand are denoted by u_i, d_i. Then, adding and subtracting Gu_i and Kd_i to (9)

$$\dot{z} = F + Gu_i + Kd_i + G(u - u_i) + K(d - d_i) \tag{22}$$

This is done to derive an alternate form of Bellman equation containing both the executed and improved policies for off-policy version of IRL similar to the one presented in [18]. Taking derivative of $V_i(z)$ along (22) a revised form of Bellman equation [see (16)] is given by

$$\underbrace{\nabla V_i^{\mathrm{T}}(F + Gu_i + Kd_i) + \nabla V_i^{\mathrm{T}}(G(u - u_i)) + \nabla V_i^{\mathrm{T}}(K(d - d_i))}_{\dot{V}_i} - \gamma V_i = -z^{\mathrm{T}}Q_1 z - u_i^{\mathrm{T}}Ru_i + \alpha^2 d_i^{\mathrm{T}}d_i \tag{23}$$

Multiplying both sides of (23) by $e^{-\gamma t}$, left-hand side (LHS) of (23) can be expressed as

$$\frac{d(e^{-\gamma t}V_i(z))}{dt} = e^{-\gamma t}(\nabla V_i^{\mathrm{T}}(F + Gu_i + Kd_i) + \nabla V_i^{\mathrm{T}}(G(u - u_i)) + \nabla V_i^{\mathrm{T}}(K(d - d_i)) - \gamma V_i) \tag{24}$$

Using (20) and (21) in (24), $d(e^{-\gamma t}V_i)/dt$ can be rewritten as

$$\begin{aligned}\frac{d(e^{-\gamma t}V_i(z))}{dt} &= e^{-\gamma t}(-z^{\mathrm{T}}Q_1 z - u_i R u_i + \alpha^2 d_i^{\mathrm{T}} d_i \\ &\quad + \nabla V_i^{\mathrm{T}} G(u - u_i) + \nabla V_i^{\mathrm{T}} K(d - d_i)) \\ &= e^{-\gamma t}(-z^{\mathrm{T}}Q_1 z - u_i R u_i + \alpha^2 d_i^{\mathrm{T}} d_i \\ &\quad - 2u_{i+1}^{\mathrm{T}} R(u - u_i) + 2\alpha^2 d_{i+1}^{\mathrm{T}}(d - d_i))\end{aligned} \tag{25}$$

Integrating the above equation over $[t - T, t]$ on both sides of (25) and rearranging

$$\begin{aligned}&V_i(t) - e^{\gamma T}V_i(t - T) + \int_{t-T}^{t} e^{-\gamma(\tau - t)}(z^{\mathrm{T}}Q_1 z + u_i^{\mathrm{T}} R u_i \\ &-\alpha^2 d_i^{\mathrm{T}} d_i + 2u_{i+1}^{\mathrm{T}} R(u - u_i) - 2\alpha^2 d_{i+1}^{\mathrm{T}}(d - d_i))d\tau = 0\end{aligned} \tag{26}$$

where $V_i(t) \triangleq V_i(z(t))$ and $V_i(t - T) \triangleq V_i(z(t - T))$. Note that (26) resembles (23) or (20) in the limiting sense when $T \to 0$. To maintain this equivalence, the reinforcement interval T should be selected as small as possible [34]. The primary advantage of (26) compared to (23) or (20) is that it does not require the prior information of drift dynamics. Since, (20) or (23) are equivalent to the modified Bellman equation presented in (26), all the convergence properties proved for (20) and (21) (see Theorems 1 and 2) hold true for (26) and (21) as shown in [29, 35].

Note that when improved policies obtained in last iteration are executed to the system during learning, then (25) leads to on-policy IRL form of Bellman equation [16, 22] instead of (26) above.

3.2 Approximation of value function, control policy and disturbance policy

Similar to [18, 29], value function and improved policies are represented by

$$V_i = W_c^{\mathrm{T}}\sigma_c + \varepsilon_c;\ u_{i+1} = W_a^{\mathrm{T}}\sigma_a + \varepsilon_a;\ d_{i+1} = W_d^{\mathrm{T}}\sigma_d + \varepsilon_d \tag{27}$$

where $W_c \in \mathbb{R}^{a_1}$ is the weight for critic NN, $\sigma_c \in \mathbb{R}^{a_1}$ is the regressor vector for critic NN, $W_a \in \mathbb{R}^{a_2 \times m}$ is the weight matrix for actor NN, $\sigma_a \in \mathbb{R}^{a_2}$ is the regressor vector for actor NN, $W_d \in \mathbb{R}^{a_3 \times l}$ is the weight matrix for disturbance NN and $\sigma_d \in \mathbb{R}^{a_3}$ is the regressor vector for disturbance NN, $\varepsilon_c \in \mathbb{R}, \varepsilon_a \in \mathbb{R}^m$ and $\varepsilon_d \in \mathbb{R}^{q_1}$ are approximation errors for critic, actor and disturbance NN, respectively. Using (27) in (26), the HJI error becomes

$$\begin{aligned}\varepsilon_{HJI} &= W_c^{\mathrm{T}}[\sigma_c(t) - e^{\gamma T}\sigma_c(t - T)] + I_1 \\ &+ \int_{t-T}^{t} e^{-\gamma(\tau - t)}(2\sigma_a^{\mathrm{T}} W_a R(u - u_i) - 2\alpha^2 \sigma_d^{\mathrm{T}} W_d(d - d_i))\mathrm{d}\tau\end{aligned} \tag{28}$$

where

$$I_1 \triangleq \int_{t-T}^{t} e^{-\gamma(\tau - t)}(z^{\mathrm{T}}Q_1 z + u_i^{\mathrm{T}} R u_i - \alpha^2 d_i^{\mathrm{T}} d_i)\mathrm{d}\tau \tag{29}$$

Now, since ideal weights are not known, their estimates will be utilised instead

$$\hat{V}_i = \hat{W}_c^{\mathrm{T}}\sigma_c;\ \hat{u}_{i+1} = \hat{W}_a^{\mathrm{T}}\sigma_a;\ \hat{d}_{i+1} = \hat{W}_d^{\mathrm{T}}\sigma_d \tag{30}$$

Then, the instantaneous HJI error in terms of estimated weights can be written as

$$\begin{aligned}\hat{e}_1(t) &= \hat{W}_c^{\mathrm{T}}[\sigma_c(t) - e^{\gamma T}\sigma_c(t - T)] + I_1 \\ &+ \mathrm{v}(\hat{W}_a)^{\mathrm{T}} \int_{t-T}^{t} 2e^{-\gamma(\tau - t)}(R(u - u_i) \otimes \sigma_a)\mathrm{d}\tau \\ &- \mathrm{v}(\hat{W}_d)^{\mathrm{T}} \int_{t-T}^{t} 2e^{-\gamma(\tau - t)}(\alpha^2(d - d_i) \otimes \sigma_d)\mathrm{d}\tau\end{aligned} \tag{31}$$

Equation (31) can be written in compact form as

$$\hat{e}_1(t) = \hat{W}^{\mathrm{T}}\rho_1 + I_1(t) \tag{32}$$

where I_1 is the reinforcement integral given in (29).

$$\hat{W} = \begin{pmatrix} \hat{W}_c \\ \mathrm{v}(\hat{W}_a) \\ \mathrm{v}(\hat{W}_d) \end{pmatrix};\ \rho_1 = \begin{pmatrix} \Delta\sigma_c \\ \int_{t-T}^{t} 2e^{-\gamma(\tau - t)}(R(u - u_i) \otimes \sigma_a(t))d\tau \\ -\int_{t-T}^{t} 2e^{-\gamma(\tau - t)}(\alpha^2(d - d_i) \otimes \sigma_d(t))d\tau \end{pmatrix} \tag{33}$$

where $\mathrm{v}(.)$ represents vectorisation of matrix and $\otimes$ denotes the kronecker product. Here, $\hat{W} \in \mathbb{R}^q$ is the composite NN weight vector, and $\rho_1 \in \mathbb{R}^q$ is the composite regressor vector, where $q = a_1 + ma_2 + la_3$, in which m is the dimension of the control vector and l is the dimension of the disturbance vector, and a_1, a_2, a_3 are number of neurons in the hidden layer or the size of regressor vector for critic, actor and disturbance, respectively. $\Delta\sigma_c = [\sigma_c(t) - e^{\gamma T}\sigma_c(t - T)]$. In subsequent discussion, $\hat{e}_1(t)$, $\hat{e}_1(t_j)$ are denoted by $\hat{e}_1$ and $\hat{e}_{1j}$, respectively, and $\rho_1 \triangleq \rho_1(t)$, $\rho_{1j} \triangleq \rho_1(t_j)$. Similarly, $I_1 \triangleq I_1(t)$ and $I_{1j} \triangleq I_1(t_j)$.

3.3 Exisiting update laws in literature for off-policy IRL

In [29], a recursive least square (RLS)-based update law was proposed to minimise the HJI approximation error to solve H_∞-tracking problem. Their update law was given by

$$\hat{W} = (\aleph\aleph^{\mathrm{T}})^{-1}\aleph Y \tag{34}$$

where $\aleph = [\rho_1(t_1), \rho_1(t_2), \rho_1(t_3), \ldots, \rho_1(t_N)]$ and $Y = [-I_1(t_1), -I_1(t_2), -I_1(t_3), \ldots, -I_1(t_N)]^{\mathrm{T}}$. Equation (34) yields V_i, u_{i+1} and d_{i+1}. However, this discontinuous-time update law requires that N samples be collected, using a fixed control policy in each phase of N time-steps before (34) could produce a new update of $\hat{W}$ at the end of that respective phase. The number of samples N that need to be collected is equal to or greater than the size of the regressor vector, i.e. $N \geq \mathrm{size}(\rho_1)$. This procedure makes it less sensitive to real-time variation in plant parameters as was indicated in [18]

In order to remedy these issues, a continuous-time update law was presented in [18] utilising constant learning gradient descent and ER technique to train the actor and critic NN in off-policy IRL to solve optimal tracking problem.

However, this control formulation did not incorporate any disturbance rejection. Their update law was given as

$$\dot{\hat{W}} = -\frac{\eta}{N+1}\left(\frac{\rho_2}{m_s^2}\hat{e}_2 + \sum_{j=1}^{N}\left(\frac{\rho_{2j}}{m_{sj}^2}\hat{e}_{2j}\right)\right) \tag{35}$$

where ρ_2 in (35) is made up of first two components of ρ_1 in (33), that is ρ_2 does not contain the last component of ρ_1 that corresponds to disturbance term (d).

Recently, a continuous-time update law was presented in [30] for H_∞-tracking control incorporating terminal constraints, in which the update law relied only on constant learning-based gradient descent apart from a term dedicated for incorporating terminal constraints.

4 Variable gain gradient descent and ER technique-based parameter update law

4.1 Novel update law

All the update laws mentioned in Section 3.3 either utilise RLS method [29] or gradient descent with constant learning rate [30]. While the RLS-based update laws are usually found to be less sensitive to real-time parameter variations in plant dynamics [18], constant learning rate-based update law cannot scale the learning rate based on the instantaneous value of the HJI error [36]. Also, ER technique and inclusion of robust term in update laws were found to beneficial [18, 31, 37]. Considering these, a novel continuous-time update law is presented in this paper to tune the critic, actor and disturbance NN weights online in order to solve H_∞-tracking problem. The novel update law utilising variable gain gradient descent and ER technique is now presented as

$$\begin{aligned}\dot{\hat{W}} = &-\frac{\eta}{N+1}\left(\frac{\rho_1}{m_s^2}g_2\hat{e}_1+\sum_{j=1}^{N}\left(\frac{\rho_{1j}}{m_{sj}^2}g_{2j}\hat{e}_{1j}\right)\right.\\ &\left.-K_1g_2\frac{\rho_1^{\mathrm{T}}}{m_s}\hat{W}+g_2K_2\hat{W}-K_1\sum_{j=1}^{N}\frac{g_{2j}\rho_{1j}^{\mathrm{T}}}{m_{sj}}\hat{W}\right)\end{aligned} \tag{36}$$

where $g_2 = \left|\hat{e}_1\right|^{k_1} + l$ and $g_{2j} = \left|\hat{e}_1(t_j)\right|^{k_1} + l$ (l is a small positive constant) and $m_s = \sqrt{1+\rho_1^{\mathrm{T}}\rho_1}$ and $m_{sj} = \sqrt{1+\rho_{1j}^{\mathrm{T}}\rho_{1j}}$ and $K_1 \in \mathbb{R}^q$ (where $q = a_1 + ma_2 + la_3$ and m is the dimension of the control vector and l is the dimension of the disturbance vector, and a_1, a_2, a_3 are as defined after (25)) , $K_2 \in \mathbb{R}^{q\times q}$. Now, in order to implement ER technique, past N time-step data $\left(\{\hat{e}_1(t_j), \rho_1(t_j), I_1(t_j)\}_{j=1}^{N}\right)$ is stored in memory stack of size N where $N \geq q$. In order to store $\rho_1(t_j)$, a matrix ϱ of size $q \times N$ is created and initialised with 0. Similarly, in order to store $\hat{e}_1(t_j)$ and $I_1(t_j)$, two row vectors of size N are initialised with 0. As the new data arrive at every iteration, all the columns of ϱ are shifted one place to the right and the first column is filled up by the new value of composite regressor vector for that particular iteration. Similar procedure is applied to update $\hat{e}_1(t_j)$ and $I_1(t_j)$. Unlike discontinuous-time update laws such as [29], the update law presented in this paper does not need to wait for prolonged period of time to collect N samples altogether in an episode to generate the NN weight update. Further, it also does not require fixed control policy to generate samples during the learning phase.

It can be seen that the update law (36) utilises variable learning rate (via the term $\left|\hat{e}_1\right|^{k_1}$) that is a function of instantaneous HJI error. This has the advantage of scaling the learning rate and reducing the size of the residual set for error in NN weights as will become clear in the stability proof of Theorems 3 and 4. Additionally, the second and fifth terms under summation correspond to the ER terms, these terms can use past observations much more effectively. The memory stack in ER can be updated with recent data as and when they arrive. This leads to an efficient learning from past data. Finally, inclusion of robust terms in the update law provides robustness against variations in approximation errors and also reduce the size of the residual set for error in NN weights.

Proposition 1: Let $x \in \mathbb{R}^n$ and $M \in \mathbb{R}^{n\times n}$ be any square matrix, then, $\lambda_{\min}(\frac{M+M^{\mathrm{T}}}{2}) \parallel x \parallel^2 \leq x^{\mathrm{T}}Mx \leq \lambda_{\max}(\frac{M+M^{\mathrm{T}}}{2}) \parallel x \parallel^2$. Where, $\lambda_{\min}(.)$ and $\lambda_{\max}(.)$ denote the minimum and maximum eigenvalues of corresponding matrices, respectively.

Proof: The proof of this proposition is provided in the Appendix (Section 8.1). □

Assumption 1: It is assumed that the control policy u is admissible policy for the augmented system. This makes the augmented system remain in the compact set $\Omega_1 \subset \mathbb{R}^{2n}$. Such an admissible policy is chosen for online training of critic, actor and disturbance NNs.

Assumption 2: There exist bounds such that, $\parallel W \parallel \leq W_M$, $\parallel \bar{\rho} \parallel \leq \bar{\rho}_M$, $|m_s| \leq m_{sM}$, $\parallel W_c \parallel \leq W_{cm}$, $\parallel W_a \parallel \leq W_{am}$, $\parallel W_d \parallel \leq W_{dm}$, $\parallel \sigma_c \parallel \leq b_c$, . $\parallel \sigma_a \parallel \leq b_a$, $\parallel \sigma_d \parallel \leq b_d$, $\parallel \varepsilon_c \parallel \leq b_{ec}$, $\parallel \varepsilon_a \parallel \leq b_{ea}$, $\parallel \varepsilon_d \parallel \leq b_{ed}$

This is in line with Assumption 2 of [18]

Theorem 3: Let $\hat{W}$ be the estimated parameters for critic, actor and disturbance. Under the Assumptions 1, 2, and that the normalised regressor $\bar{\rho}_1 \triangleq \rho_1/\sqrt{1+\rho_1^{\mathrm{T}}\rho_1}$ is persistently excited, the update law mentioned in (36) ensures the error in NN weights $\tilde{W}$ to be UUB stable.

Proof: From (28) and (31), $\hat{e}_1$ can be written in terms of $\tilde{W} = W - \hat{W}$ as,

$$\hat{e}_1 = \varepsilon_{HJI} - \tilde{W}^{\mathrm{T}}\rho_1 \tag{37}$$

From Assumption 2, it is clear that ε_{HJI} and ρ_1 are bounded. Let the Laypunov function candidate be, $L = \frac{1}{2}\tilde{W}^{\mathrm{T}}\eta^{-1}\tilde{W}$. Utilising (37) and (36), derivative of Lyapunov function can be written as

$$\begin{aligned}\dot{L} = &\tilde{W}^{\mathrm{T}}\varepsilon\left(\frac{\rho_1 g_2}{m_s^2(N+1)}+\sum_{j=1}^{N}\frac{\rho_1(t_j)g_{2j}}{m_s^2(t_j)(N+1)}\right)-\tilde{W}^{\mathrm{T}}\left(\frac{g_2\bar{\rho}_1\bar{\rho}_1^{\mathrm{T}}}{(N+1)}\right.\\ &\left.+\sum_{j=1}^{N}\frac{g_{2j}\bar{\rho}_{1j}\bar{\rho}_{1j}^{\mathrm{T}}}{(N+1)}\right)\tilde{W}-\frac{1}{N+1}g_2\tilde{W}^{\mathrm{T}}K_1^{\mathrm{T}}\frac{\rho_1^{\mathrm{T}}}{m_s}W+\frac{g_2}{N+1}\tilde{W}^{\mathrm{T}}K_1^{\mathrm{T}}\frac{\rho_1^{\mathrm{T}}}{m_s}\tilde{W}\\ &+\frac{g_2}{N+1}\tilde{W}^{\mathrm{T}}K_2W-\frac{g_2}{N+1}\tilde{W}^{\mathrm{T}}K_2\tilde{W}-\tilde{W}^{\mathrm{T}}K_1\sum_{j=1}^{N}\frac{g_{2j}\rho_1(t_j)^{\mathrm{T}}}{m_s(N+1)}W\\ &+\tilde{W}^{\mathrm{T}}K_1\sum_{j=1}^{N}\frac{g_{2j}\rho_1(t_j)^{\mathrm{T}}}{m_s(N+1)}\tilde{W}\end{aligned} \tag{38}$$

where $\dot{\tilde{W}} = -\dot{\hat{W}}$. Note that when $K_1 = 0, K_2 = 0$ in (38) then, in order to ensure negative definiteness of $\dot{L}$, the sum of outer product matrices formed by present and past composite regressor vectors ρ_1 and ρ_{1j}, respectively, should be positive definite, which happens when at least q independent past composite regressor vectors (ρ_{1j}) are present in the memory stack. This necessitate $N \geq q$. This condition is similar to persistence of excitation condition. Now, let

$$\begin{aligned}\mathscr{P}(\tilde{W},\rho_1) &\triangleq \left(\tilde{W}^{\mathrm{T}}\bar{\rho}_1,\ \tilde{W},\ \tilde{W}^{\mathrm{T}}\sum_{j=1}^{N}\bar{\rho}_1(t_j)\right)^{\mathrm{T}}\\ M(g_2) &\triangleq \begin{pmatrix} g_2 & -\frac{g_2K_1^{\mathrm{T}}}{2} & \sum g_{2j}\\ -\frac{g_2K_1}{2} & g_2K_2 & -K_1\frac{\sum g_{2j}}{2}\\ \sum g_{2j} & -K_1^{\mathrm{T}}\frac{\sum g_{2j}}{2} & \sum g_{2j}\end{pmatrix}\end{aligned} \tag{39}$$

$$\mathscr{N}(g_2) \triangleq \begin{pmatrix} g_2\varepsilon - g_2K_1^{\mathrm{T}}W\\ g_2K_2W\\ \varepsilon\sum_{j=1}^{N}g_{2j} - K_1\sum_{j=1}^{N}g_{2j}W\end{pmatrix} \tag{40}$$

In the subsequent analysis, $\mathscr{P} \triangleq \mathscr{P}(\tilde{W},\rho_1)$, $M \triangleq M(g_2)$ and $N \triangleq N(g_2)$. Now, $\parallel \tilde{W} \parallel$ is assumed to be bounded as $\parallel \tilde{W} \parallel \leq B$, which follows from a similar assumption on the boundedness of $\parallel \tilde{W} \parallel$ made in the proof of Theorem 3 of [18]. Using this assumption along with Assumption 2 in the definition of $\hat{e}_1$ in (37) implies that $g_2 = \left|\hat{e}_1\right|^{k_1} + l$ [defined after (36)] is also bounded.

Now, using (39) and (40) and Proposition 1, (38) can be written in compact form as

$$\dot{L} = \frac{1}{N+1}\left(-\mathscr{P}^{\mathrm{T}} M \mathscr{P} + \mathscr{P}\mathscr{N}\right) \leq \frac{-\lambda_{\min}(M') \parallel \mathscr{P} \parallel^2 + b_N \parallel \mathscr{P} \parallel}{N+1} \tag{41}$$

where $M' \triangleq \frac{M+M^{\mathrm{T}}}{2}$, b_N is the maximum value of norm of $\mathscr{N}$, i.e. $\parallel \mathscr{N} \parallel \leq b_N = \max(\parallel N \parallel)$. For Lyapunov derivative to be negative definite, the following inequality should hold:

$$\parallel \mathscr{P} \parallel > \frac{b_N}{\lambda_{\min}(M')} \tag{42}$$

From the definition of $\mathscr{P}$ as mentioned in (39)

$$\parallel \mathscr{P} \parallel \leq \parallel \tilde{W} \parallel \sqrt{1 + \parallel \bar{\rho}_{1M} \parallel^2 + \left\| \sum_{j=1}^{N} \bar{\rho}_{1M}(t_j) \right\|^2} \tag{43}$$

where ρ_{1M} is the maximum value of ρ_1. The right hand side of (43) will be represented as

$$\mathscr{S}(\rho_{1M}) \triangleq \sqrt{1 + \parallel \bar{\rho}_{1M} \parallel^2 + \left\| \sum_{j=1}^{N} \bar{\rho}_{1M}(t_j) \right\|^2} \tag{44}$$

Therefore, from (42) and (43)

$$\parallel \tilde{W} \parallel > \frac{b_N}{\mathscr{S}(\rho_{1M})\lambda_{\min}(M')} \tag{45}$$

This concludes the UUB stability proof for (36). □

Remark 1: From Theorem 3, it is clear that the residual set of $\parallel \tilde{W} \parallel$ is controlled by the term $\lambda_{\min}(M')$. The gains K_1 and K_2 in (36) can be appropriately selected such that $\lambda_{\min}(M')$ becomes a large value hence shrinking the UUB bound of $\parallel \tilde{W} \parallel$.

Now, assuming that u_i and d_i are in sufficiently small neighbourhood of the optimal policies, the HJI error can be modified in the same way as mentioned in [18] (refer to Section 4.2 in [18]). It is assumed that there exist NNs that can approximate the optimal value, action and disturbance policies as

$$V^* = W_c^{\mathrm{T}}\sigma_c + \varepsilon_c;\ u^* = W_a^{\mathrm{T}}\sigma_a + \varepsilon_a;\ d^* = W_d^{\mathrm{T}}\sigma_d + \varepsilon_d \tag{46}$$

Under these assumptions, using (46) in (26), the HJI approximation error is obtained as

$$\begin{aligned}\varepsilon_{HJI} = {} & W_c^{\mathrm{T}}[\sigma_c(t) - e^{\gamma T}\sigma(t-T)] + \int_{t-T}^{t} e^{-\gamma(\tau-t)}(z^{\mathrm{T}}Q_1 z \\ & + 2\sigma_a^{\mathrm{T}}W_a R u - \sigma_a^{\mathrm{T}}W_a R W_a^{\mathrm{T}}\sigma_a + \alpha^2\sigma_d^{\mathrm{T}}W_d R W_d^{\mathrm{T}}\sigma_d \\ & - 2\alpha^2\sigma_d^{\mathrm{T}}W_d d)\mathrm{d}\tau\end{aligned} \tag{47}$$

In terms of approximation errors, the HJI error $\varepsilon_{\mathrm{HJI}}$ can equivalently be given as

$$\begin{aligned}\varepsilon_{\mathrm{HJI}} = {} & \varepsilon_c(t) - e^{\gamma T}\varepsilon_c(t-T) - \int_{t-T}^{t} e^{-\gamma(\tau-t)}\big[2\varepsilon_a^{\mathrm{T}}R u \\ & - 2\varepsilon_a^{\mathrm{T}}R\sigma_a - \varepsilon_a^{\mathrm{T}}R\varepsilon_a + \alpha^2\varepsilon_d^{\mathrm{T}}R\varepsilon_d - 2\alpha^2\varepsilon_d^{\mathrm{T}}d\big]\mathrm{d}\tau\end{aligned} \tag{48}$$

Now, since, ideal NN weights are not known, their estimates will be utilised to express optimal value function and optimal policies

$$\hat{V}^* = \hat{W}_c\sigma_c,\ \hat{u}^* = \hat{W}_a\sigma_a,\ \hat{d}^* = \hat{W}_d\sigma_d \tag{49}$$

Based on these estimated weights, following (47) the approximate HJI error can be re-stated as

$$\begin{aligned}\hat{e} = {} & \hat{W}_c^{\mathrm{T}}[\sigma_c(t) - e^{\gamma T}\sigma(t-T)] + \int_{t-T}^{t} e^{-\gamma(\tau-t)}(z^{\mathrm{T}}Q_1 z \\ & + 2\sigma_a^{\mathrm{T}}\hat{W}_a R u - \sigma_a^{\mathrm{T}}\hat{W}_a R\hat{W}_a^{\mathrm{T}}\sigma_a + \alpha^2\sigma_d^{\mathrm{T}}\hat{W}_d R\hat{W}_d^{\mathrm{T}}\sigma_d \\ & - 2\alpha^2\sigma_d^{\mathrm{T}}\hat{W}_d d)\mathrm{d}\tau\end{aligned} \tag{50}$$

The approximate HJI error, thus, can be expressed in a compact form as

$$\hat{e}(t) = \hat{W}^{\mathrm{T}}\rho + \mathrm{v}(\hat{W}_a)^{\mathrm{T}}\mathscr{A}_2\mathrm{v}(\hat{W}_a) - \mathrm{v}(\hat{W}_d)^{\mathrm{T}}\mathscr{B}_2\mathrm{v}(\hat{W}_d) + I_2 \tag{51}$$

where

$$\begin{aligned}&\Delta\sigma_c \triangleq \sigma_c(t) - e^{\gamma T}\sigma(t-T);\ \mathscr{A}_1 \triangleq \int_{t-T}^{t} e^{-\gamma(\tau-t)}(Ru \otimes \sigma_a)\mathrm{d}\tau \\ &\mathscr{A}_2 \triangleq \int_{t-T}^{t} e^{-\gamma(\tau-t)}(R \otimes \sigma_a\sigma_a^{\mathrm{T}})\mathrm{d}\tau;\ \mathscr{B}_1 \triangleq \int_{t-T}^{t} e^{-\gamma(\tau-t)}(\alpha^2 d \otimes \sigma_d)\mathrm{d}\tau \\ &\mathscr{B}_2 \triangleq \int_{t-T}^{t} e^{-\gamma(\tau-t)}(\alpha^2 \otimes \sigma_d\sigma_d^{\mathrm{T}})\mathrm{d}\tau;\ I_2 \triangleq \int_{t-T}^{t} e^{-\gamma(\tau-t)}(z^{\mathrm{T}}Q_1 z)\mathrm{d}\tau\end{aligned} \tag{52}$$

$$\hat{W} \triangleq \begin{pmatrix} \hat{W}_c \\ \mathrm{v}(\hat{W}_a) \\ \mathrm{v}(\hat{W}_d) \end{pmatrix},\ \rho \triangleq \begin{pmatrix} \Delta\sigma_c \\ 2\mathscr{A}_1 - 2\mathscr{A}_2\mathrm{v}(\hat{W}_a) \\ -2\mathscr{B}_1 + 2\mathscr{B}_2\mathrm{v}(\hat{W}_d) \end{pmatrix} \tag{53}$$

Similar to (36), continuous-time update law for this case can be written as

$$\begin{aligned}\dot{\hat{W}} = {} & -\frac{\eta}{N+1}\left(\frac{\rho}{m_s^2}g_1\hat{e} + \sum_{j=1}^{N}\left(\frac{\rho_j}{m_{sj}^2}g_{1j}\hat{e}_j\right) - K_1 g_1\frac{\rho^{\mathrm{T}}}{m_s}\hat{W}\right. \\ & \left. + g_1 K_2\hat{W} - K_1\sum_{j=1}^{N}\frac{g_{1j}\rho^{\mathrm{T}}(t_j)}{m_{sj}}\hat{W}\right)\end{aligned} \tag{54}$$

where

$$g_1 = \left|\hat{e}\right|^{k_1} + l,\ g_{1j} = \left|\hat{e}_j\right|^{k_1} + l,\ (j = 1, 2, \ldots, N) \tag{55}$$

and l is a small positive constant. It could be observed that certain terms appearing in (54) are defined differently from (36). For instance, ($\hat{e}$ and ρ) appearing in (54) are given by (51) and (53), respectively.

Note that the update law presented above (54) is different from least square-based update law mentioned in [29] and continuous-time gradient descent-based update laws mentioned in [18, 30]. The update law presented in this paper consists of five terms. The first term is directly responsible for reducing the HJI error, while the second term is a representation of its past observations over the memory stack. Also, unlike the constant learning rate in [18, 30], in this paper the learning rate in (54) is time varying and considered as a function of the HJI error such that it can accelerate the learning when the HJI error is large and reduce the learning speed when the HJI error becomes small. The next three terms are responsible for providing robustness in achieving small residual set. Moreover, the second and fifth terms correspond to the ER of the first and third terms, respectively. Significance of each term in the update law (54) in improving the performance of the tracking controller would be evident in the proof of Theorem 4 and its subsequent discussion.

4.2 Stability proof of the update law

Theorem 4: Let $\hat{W}$ be the estimated parameters for critic, actor and disturbance. Under the Assumptions 1, 2, and that the normalised regressor $\bar{\rho} \triangleq \rho/\sqrt{1+\rho^{\mathrm{T}}\rho}$ is persistently excited, the update law mentioned in (54) ensures the error in NN weights $\tilde{W}$ to be UUB stable.

Proof: Let the Lyapunov function candidate be: $L = (1/2)\tilde{W}^{\mathrm{T}}\eta^{-1}\tilde{W}$. In order to prove stability of the update law, the HJI error needs to be expressed as a function of $\tilde{W}$. In order to accomplish this, using (47) and (51)

$$\hat{e} = \varepsilon_{\mathrm{HJI}} - \tilde{W}^{\mathrm{T}}\rho - \mathrm{v}(\tilde{W}_a)^{\mathrm{T}}\mathscr{A}_2\mathrm{v}(\tilde{W}_a) + \mathrm{v}(\tilde{W}_d)^{\mathrm{T}}\mathscr{B}_2\mathrm{v}(\tilde{W}_d) \quad (56)$$

In subsequent discussion, ε would be equivalently used in place of $\varepsilon_{\mathrm{HJI}}$. Differentiating the Lyapunov function

$$\begin{aligned}\dot{L} = \tilde{W}^{\mathrm{T}}\eta^{-1}\dot{\tilde{W}} = {} & \frac{g_1}{(N+1)m_s^2}(\tilde{W}^{\mathrm{T}}\rho\varepsilon - \tilde{W}^{\mathrm{T}}\rho\tilde{W}^{\mathrm{T}}\rho \\ & - \tilde{W}^{\mathrm{T}}\rho\mathrm{v}(\tilde{W}_a)^{\mathrm{T}}\mathscr{A}_2\mathrm{v}(\tilde{W}_a) + \tilde{W}^{\mathrm{T}}\rho\mathrm{v}(\tilde{W}_d)^{\mathrm{T}}\mathscr{B}_2\mathrm{v}(\tilde{W}_d)) \\ & + \frac{1}{N+1}\Big(\tilde{W}\varepsilon\sum_{j=1}^{N}\frac{\rho(t_j)g_{1j}}{m_s^2(t_j)} - \tilde{W}^{\mathrm{T}}\sum_{j=1}^{N}\frac{\rho(t_j)\rho(t_j)^{\mathrm{T}}g_{1j}}{m_s^2(t_j)}\tilde{W} \\ & - \tilde{W}^{\mathrm{T}}\sum_{j=1}^{N}\frac{g_{1j}\rho(t_j)^{\mathrm{T}}}{m_s^2(t_j)}\mathrm{v}(\tilde{W}_a)^{\mathrm{T}}\mathscr{A}_2(t_j)\mathrm{v}(\tilde{W}_a) \\ & + \tilde{W}^{\mathrm{T}}\sum_{j=1}^{N}\frac{\rho(t_j)g_{1j}}{m_s^2(t_j)}(\mathrm{v}(\tilde{W}_d)^{\mathrm{T}}\mathscr{B}_2(t_j)\mathrm{v}(\tilde{W}_d))\Big) \\ & - \frac{1}{N+1}g_1\tilde{W}^{\mathrm{T}}K_1^{\mathrm{T}}\frac{\rho^{\mathrm{T}}}{m_s}W + \frac{g_1}{N+1}\tilde{W}^{\mathrm{T}}K_1^{\mathrm{T}}\frac{\rho^{\mathrm{T}}}{m_s}\tilde{W} \\ & + \frac{g_1}{N+1}\tilde{W}^{\mathrm{T}}K_2W - \frac{g_1}{N+1}\tilde{W}^{\mathrm{T}}K_2\tilde{W} \\ & - \tilde{W}^{\mathrm{T}}K_1\sum_{j=1}^{N}\frac{g_{1j}\rho(t_j)^{\mathrm{T}}}{m_s(N+1)}W + \tilde{W}^{\mathrm{T}}K_1\sum_{j=1}^{N}\frac{g_{1j}\rho(t_j)^{\mathrm{T}}}{m_s(N+1)}\tilde{W}\end{aligned} \quad (57)$$

Now, in order to find a bound over $\dot{L}$, it is required to find bound over terms containing $\mathscr{A}_2$ and $\mathscr{B}_2$ [refer to (52)]. Recall from Assumption 2 that $\| \sigma_a \| \leq b_a$ and $\| \sigma_d \| \leq b_d$. Utilising properties of matrices (maximum and minimum eigenvalue) and persistent excitation (PE) condition ($\lambda_1 I \leq \int_{t-T}^{t}\rho\rho^{\mathrm{T}}\mathrm{d}\tau \leq \lambda_2 I$, where λ_1, λ_2 are positive constants) on regressor, the bounds over terms containing $\mathscr{A}_2$ and $\mathscr{B}_2$ can be derived as,

$$\begin{aligned}\mathrm{v}(\tilde{W}_a)^{\mathrm{T}}(R\otimes\sigma_a\sigma_a^{\mathrm{T}})\mathrm{v}(\tilde{W}_a) &\leq q_1 \| \mathrm{v}(\tilde{W}_a) \|^2 \leq q_1 \| \tilde{W} \|^2 \\ \mathrm{v}(\tilde{W}_d)^{\mathrm{T}}(\alpha^2\otimes\sigma_d\sigma_d^{\mathrm{T}})\mathrm{v}(\tilde{W}_d) &\leq q_2 \| \mathrm{v}(\tilde{W}_d) \|^2 \leq q_2 \| \tilde{W} \|^2\end{aligned} \quad (58)$$

where q_1 and q_2 are the maximum eigenvalues of the matrices given by $(R\otimes\sigma_a\sigma_a^{\mathrm{T}})$ and $(\alpha^2\otimes\sigma_d\sigma_d^{\mathrm{T}})$, respectively. Further, the bound over terms, $\mathrm{v}(\tilde{W}_a)^{\mathrm{T}}\mathscr{A}_2\mathrm{v}(\tilde{W}_a)$ and $\mathrm{v}(\tilde{W}_d)^{\mathrm{T}}\mathscr{B}_2\mathrm{v}(\tilde{W}_d)$ can be derived from (52), (53) and (58) as

$$\begin{aligned}\mathrm{v}(\tilde{W}_a)^{\mathrm{T}}\mathscr{A}_2\mathrm{v}(\tilde{W}_a) &\leq q_1\int_{t-T}^{t}e^{-\gamma(\tau-t)} \| \tilde{W} \|^2\,\mathrm{d}\tau \\ &\leq \frac{q_1}{\gamma\beta_1}(e^{\gamma T}-1) \| \tilde{W}^{\mathrm{T}}\bar{\rho} \|^2 = \frac{q_1}{\gamma\beta_1}(e^{\gamma T}-1)\tilde{W}^{\mathrm{T}}\bar{\rho}\bar{\rho}^{\mathrm{T}}\tilde{W}\end{aligned} \quad (59)$$

$$\begin{aligned}\mathrm{v}(\tilde{W}_d)^{\mathrm{T}}\mathscr{B}_2\mathrm{v}(\tilde{W}_d) &\leq q_2\int_{t-T}^{t}e^{-\gamma(\tau-t)} \| \tilde{W} \|^2\,\mathrm{d}\tau \\ &\leq \frac{q_2}{\gamma\beta_1}(e^{\gamma T}-1) \| \tilde{W}^{\mathrm{T}}\bar{\rho} \|^2 = \frac{q_2}{\gamma\beta_1}(e^{\gamma T}-1)\tilde{W}^{\mathrm{T}}\bar{\rho}\bar{\rho}^{\mathrm{T}}\tilde{W}\end{aligned} \quad (60)$$

where $\beta_1 = \| \rho \|^2$. Using the same PE condition on $\mathscr{A}_2, \mathscr{B}_2$ in ρ, from (53) there exists a constant L_1 such that

$$\left|\tilde{W}^{\mathrm{T}}\frac{\rho}{m_s^2}\right| \leq L_1 \| \tilde{W} \| \quad (61)$$

Combining (59), (60) and (61)

$$\begin{aligned}\left|\tilde{W}^{\mathrm{T}}\frac{\rho}{m_s^2}\mathrm{v}(\tilde{W}_a)^{\mathrm{T}}\mathscr{A}_2\mathrm{v}(\tilde{W}_a)\right| &\leq L_1\frac{q_1}{\gamma\beta_1}(e^{\gamma T}-1) \| \tilde{W} \| \tilde{W}^{\mathrm{T}}\bar{\rho}\bar{\rho}^{\mathrm{T}}\tilde{W} \\ \left|\tilde{W}^{\mathrm{T}}\frac{\rho}{m_s^2}\mathrm{v}(\tilde{W}_d)^{\mathrm{T}}\mathscr{B}_2\mathrm{v}(\tilde{W}_d)\right| &\leq L_1\frac{q_2}{\gamma\beta_1}(e^{\gamma T}-1) \| \tilde{W} \| \tilde{W}^{\mathrm{T}}\bar{\rho}\bar{\rho}^{\mathrm{T}}\tilde{W}\end{aligned} \quad (62)$$

where $\bar{\rho} = \rho/m_s$ and the reinforcement interval T can be selected such that

$$L_1\frac{q_1}{\gamma\beta_1}(e^{\gamma T}-1) \| \tilde{W} \| \leq \varepsilon_{Ta};\ L_1\frac{q_2}{\gamma\beta_2}(e^{\gamma T}-1) \| \tilde{W} \| \leq \varepsilon_{Td} \quad (63)$$

where, ε_{Ta} and ε_{Td} are two small positive scalar constants. Therefore, using (63) in (62)

$$\begin{aligned}\left|\tilde{W}^{\mathrm{T}}\frac{\rho}{m_s^2}\mathrm{v}(\tilde{W}_a)^{\mathrm{T}}\mathscr{A}_2\mathrm{v}(\tilde{W}_a)\right| &\leq \varepsilon_{Ta}\tilde{W}^{\mathrm{T}}\bar{\rho}\bar{\rho}^{\mathrm{T}}\tilde{W} \\ \left|\tilde{W}^{\mathrm{T}}\frac{\rho}{m_s^2}\mathrm{v}(\tilde{W}_d)^{\mathrm{T}}\mathscr{B}_2\mathrm{v}(\tilde{W}_d)\right| &\leq \varepsilon_{Td}\tilde{W}^{\mathrm{T}}\bar{\rho}\bar{\rho}^{\mathrm{T}}\tilde{W}\end{aligned} \quad (64)$$

Similarly, their ER versions can be represented as

$$\begin{aligned}\left|\tilde{W}^{\mathrm{T}}\sum_{j=1}^{N}\frac{g_{1j}\rho(t_j)}{m_s^2(t_j)}\mathrm{v}(\tilde{W}_a)^{\mathrm{T}}\mathscr{A}_2(t_j)\mathrm{v}(\tilde{W}_a)\right| &\leq \varepsilon_{Tsa}\tilde{W}^{\mathrm{T}}\sum_{j=1}^{N}\bar{\rho}_j\bar{\rho}_j^{\mathrm{T}}\tilde{W} \\ \left|\tilde{W}^{\mathrm{T}}\sum_{j=1}^{N}\frac{g_{1j}\rho(t_j)}{m_s^2(t_j)}\mathrm{v}(\tilde{W}_a)^{\mathrm{T}}\mathscr{B}_2(t_j)\mathrm{v}(\tilde{W}_a)\right| &\leq \varepsilon_{Tsd}\tilde{W}^{\mathrm{T}}\sum_{j=1}^{N}\bar{\rho}_j\bar{\rho}_j^{\mathrm{T}}\tilde{W}\end{aligned} \quad (65)$$

Now, using (64) and (65), (57) can be rewritten as

$$\begin{aligned}\dot{L} \leq {} & \tilde{W}^{\mathrm{T}}\varepsilon\Big(\frac{\rho g_1}{m_s^2(N+1)} + \sum_{j=1}^{N}\frac{\rho(t_j)g_1(t_j)}{m_s^2(t_j)(N+1)}\Big) \\ & - \tilde{W}^{\mathrm{T}}\Big(\frac{g_1\bar{\rho}\bar{\rho}^{\mathrm{T}}}{(N+1)} + \sum_{j=1}^{N}\frac{g_{1j}\bar{\rho}_j\bar{\rho}_j^{\mathrm{T}}}{(N+1)}\Big)\tilde{W} \\ & + \tilde{W}^{\mathrm{T}}\Big(\frac{1}{N+1}\varepsilon_{Ta}\bar{\rho}\bar{\rho}^{\mathrm{T}} + \frac{1}{N+1}\varepsilon_{Tsa}\sum_{j=1}^{N}\bar{\rho}_j\bar{\rho}_j^{\mathrm{T}}\Big)\tilde{W} \\ & + \tilde{W}^{\mathrm{T}}\Big(\frac{1}{N+1}\varepsilon_{Td}\bar{\rho}\bar{\rho}^{\mathrm{T}} + \frac{1}{N+1}\varepsilon_{Tsd}\sum_{j=1}^{N}\bar{\rho}_j\bar{\rho}_j^{\mathrm{T}}\Big)\tilde{W} \\ & - \frac{1}{N+1}g_1\tilde{W}^{\mathrm{T}}K_1^{\mathrm{T}}\frac{\rho^{\mathrm{T}}}{m_s}W + \frac{g_1}{N+1}\tilde{W}^{\mathrm{T}}K_1^{\mathrm{T}}\frac{\rho^{\mathrm{T}}}{m_s}\tilde{W} \\ & + \frac{g_1}{N+1}\tilde{W}^{\mathrm{T}}K_2W - \frac{g_1}{N+1}\tilde{W}^{\mathrm{T}}K_2\tilde{W} \\ & - \tilde{W}^{\mathrm{T}}K_1\sum_{j=1}^{N}\frac{g_{1j}\rho(t_j)^{\mathrm{T}}}{m_s(N+1)}W + \tilde{W}^{\mathrm{T}}K_1\sum_{j=1}^{N}\frac{g_{1j}\rho(t_j)^{\mathrm{T}}}{m_s(N+1)}\tilde{W}\end{aligned} \quad (66)$$

After further simplification, (66) can be rendered into the following inequality:

$$\dot{L} \leq \frac{-\mathscr{P}^{\mathrm{T}}M\mathscr{P} + \mathscr{P}^{\mathrm{T}}\mathscr{N} - \mathscr{P}^{\mathrm{T}}M_{\varepsilon_{Ta}}\mathscr{P} + \mathscr{P}^{\mathrm{T}}M_{\varepsilon_{Td}}\mathscr{P}}{N+1} \quad (67)$$

where $\mathscr{P}$ in (67) is $\mathscr{P}(\tilde{W}, \rho)$ instead of $\mathscr{P}(\tilde{W}, \rho_1)$, M is $M(g_1)$ and N is $N(g_1)$ defined in Theorem 3 and

$$M_{\varepsilon_{Ta}} \triangleq \begin{pmatrix}\varepsilon_{Ta} & G_1^{\mathrm{T}} & -c_1 \\ -G_1 & 0_{q\times q} & L_1 \\ c_1 & -L_1^{\mathrm{T}} & \varepsilon_{Tsa}\end{pmatrix};\quad M_{\varepsilon_{Td}} \triangleq \begin{pmatrix}\varepsilon_{Td} & -G_2^{\mathrm{T}} & c_2 \\ G_2 & 0_{q\times q} & -L_2 \\ -c_2 & L_2^{\mathrm{T}} & \varepsilon_{Tsd}\end{pmatrix} \quad (68)$$

where $K_1 \in \mathbb{R}^q$ and $K_2 \in \mathbb{R}^{q\times q}$ and q being the dimension of the composite regressor vector ρ (see (53)). Also, $G_1, G_2 \in \mathbb{R}^q$, $c_1, c_2 \in \mathbb{R}$, $L_1, L_2 \in \mathbb{R}^q$ are constants used in (68).

Using Proposition 1, (67) can be simplified into

$$\dot{L} \leq (-\lambda_{\min}(M') \parallel \mathscr{P} \parallel^2 + b_N \parallel \mathscr{P} \parallel - \lambda_{\min}(M'_{e_{Ta}}) \parallel \mathscr{P} \parallel^2 + \lambda_{\max}(M'_{e_{Td}}) \parallel \mathscr{P} \parallel^2)/(N+1) \tag{69}$$

Note that in (69), following substitutions were made:

$$M'_{e_{Ta}} \triangleq \frac{M_{e_{Ta}} + M_{e_{Ta}}^{\mathrm{T}}}{2}, \quad M'_{e_{Td}} \triangleq \frac{M_{e_{Td}} + M_{e_{Td}}^{\mathrm{T}}}{2} \tag{70}$$

From (69), in order to ensure negative definiteness of $\dot{L}$, the following inequality should hold:

$$\parallel \mathscr{P} \parallel > \frac{b_N}{\lambda_{\min}(M') + \lambda_{\min}(M'_{e_{Ta}}) - \lambda_{\max}(M'_{e_{Td}})} \implies \dot{L} < 0 \tag{71}$$

From (71), (43) and (44) the UUB set for error in NN weights is obtained as,

$$\parallel \tilde{W} \parallel > \frac{b_N}{\mathscr{S}(\rho_M)(\lambda_{\min}(M') + \lambda_{\min}(M'_{e_{Ta}}) - \lambda_{\max}(M'_{e_{Td}}))} \tag{72}$$

where ρ_M being the maximum value of ρ and $\mathscr{S} \triangleq \mathscr{S}(\rho_M)$. Thus, from (71), (43) and (72), under the NN parameter update law (54), the error in NN weights are guaranteed to decrease outside the residual ball given as

$$\Omega_{\tilde{W}} = \left\{ \tilde{W} : \parallel \tilde{W} \parallel \leq \frac{b_N}{\mathscr{S}(\lambda_{\min}(M') + \lambda_{\min}(M'_{e_{Ta}}) - \lambda_{\max}(M'_{e_{Td}}))} \right\} \tag{73}$$

This concludes the stability proof of the continuous-time update mechanism.□

4.3 Discussion on the presented update law

Remark 2: Note that the update law presented in (54) is different from the gradient descent-based update laws of [18, 30] and least square-based one presented in [29] in several ways. First of all, being a continuous-time update law based on gradient descent, it is more sensitive to variations in plant dynamics than least square-based update mechanism in [29]. Secondly, unlike [18], it utilises H_∞ framework for disturbance rejection as well. While [30] utilised H_∞ framework for their tracking controller, their gradient descent had only constant learning rate and lacked ER and robust terms to further shrink the size of the residual set. The prime novelties of the update law (54) are the use of variable gain gradient descent and incorporation of robust terms, i.e. the last three terms in (54). These help in improving the performance of the final learnt control policies to track a given reference trajectory.

Remark 3: From Theorem 4 it is evident that $\parallel \tilde{W} \parallel$ decreases in the stable region, i.e. where $\dot{L}$ is negative definite. This results in estimated NN weights, i.e. $\hat{W}$ getting closer to ideal NN weights W, which in turn implies that the HJI error (55) is decreasing in the stable region. Now, note that the numerator in the right-hand side (RHS) of (73)), i.e. b_N is a function of $g_1 = |\hat{e}|^{k_1}$ and $g_{1j} = |\hat{e}(t_j)|^{k_1}$ [see definition of b_N after (41)], which implies that the size of the ball (73) shrinks due to decreasing g_1 and g_{1j}. Thus, b_N encapsulates the effect of variable gain gradient descent in off-policy parameter update law. Further, the variable gain in gradient descent, i.e. $|\hat{e}|^{k_1}$ and $|\hat{e}_j|^{k_1}, j = 1, 2, \ldots, N$ scale the learning rate based on instantaneous and past values of HJI error, respectively, where the constant $k_1 \geq 0$ governs the amount of scaling in the learning of gradient descent. These terms increase the learning rate when the HJI error is large and slow it down as the HJI error becomes smaller in magnitude. So, the actual learning rate becomes, $l = \eta |\hat{e}|^{k_1}$. Note that if the $|\hat{e}| \leq 1$, then $l \leq \eta$ for all $k_1 \geq 0$.

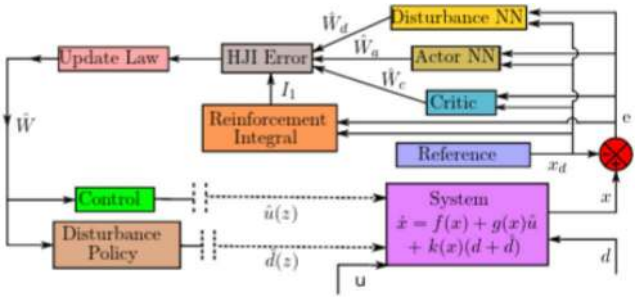

Fig. 1 *Block diagram of the control system*

However, if $|\hat{e}| \geq 1$, then $l \geq \eta$ for all $k_1 \geq 0$. Furthermore, the gains K_1 and K_2 in the robust term of the adaptation law (54) can be selected, so as to have a large $\lambda_{\min}((M + M^{\mathrm{T}})/2)$ [refer to (39)], which in turn leads to a smaller ball [refer to (73)] and hence a tighter residual set for $\tilde{W}$. With these novel modifications, the variable gain gradient descent-based off-policy update law presented in this paper yields a much tighter residual set for $\tilde{W}$ and hence improved tracking performance.

5 Simulation results

The entire control scheme can be represented as shown in Fig. 1. There are two distinct phases in off-policy IRL control scheme, i.e. (i) exploration phase and (ii) execution phase. All the signals involved in exploration phase are marked with bold lines in Fig. 1 (except the reference signal, which is required in both exploration and execution phase), whereas the dotted lines indicate the execution of the learnt policies. At first, an exploratory control policy is fired into the system, and the system is allowed to explore the state space. During the exploration process, the improved policies are not executed to the system, and hence, there is a decoupling between control and disturbance policy block and system block in Fig. 1. The update law tries to minimise the instantaneous HJI error which is dependent on actor, critic and disturbance NNs and reinforcement integral (I_1 and I_2 from (29) and (52), respectively). The regressor vector for actor, critic and disturbance NNs needs the information of augmented state vector z as can be seen in Fig. 1. When critic, actor and disturbance NN weights converge, the exploration is stopped, and learnt policies ($\hat{u}(z) = W_a^* \sigma_a$ and $\hat{d}(z) = W_d^* \sigma_d$) are executed to the system as shown in Fig. 1 (where W_a^* and W_d^* denote the final converged weights for actor and disturbance NNs, while σ_a and σ_d represent regressor vectors for actor and disturbance NNs, respectively). Therefore, the dynamics of the closed loop system when final learnt policies are executed is given by, $\dot{x} = f(x) + g(x)\hat{u} + k(x)(d + \hat{d})$.

In order to evaluate the performance of update law proposed in this paper, two applications are considered for simulation studies in this section.

- Non-linear system [18] in Section 5.1
- Linearised F16 Model [29] in Section 5.2

5.1 Non-linear system

Dynamics of a non-linear system is considered from [18] and is described as

$$\begin{aligned} \dot{x}_1 &= -\sin x_1 + x_2 \\ \dot{x}_2 &= -x_1^3 + u + d \\ y &= x_1 \end{aligned} \tag{74}$$

where, disturbance affecting the system is given as $d = .1e^{-.1t}\sin.1t$. The reference system is considered as [18]

$$\dot{x}_d = \begin{pmatrix} 0 & 0.3 \\ -0.3 & 0 \end{pmatrix} \begin{pmatrix} 0.1\sin(0.3t) \\ 0.1\sin(0.3t) \end{pmatrix} \tag{75}$$

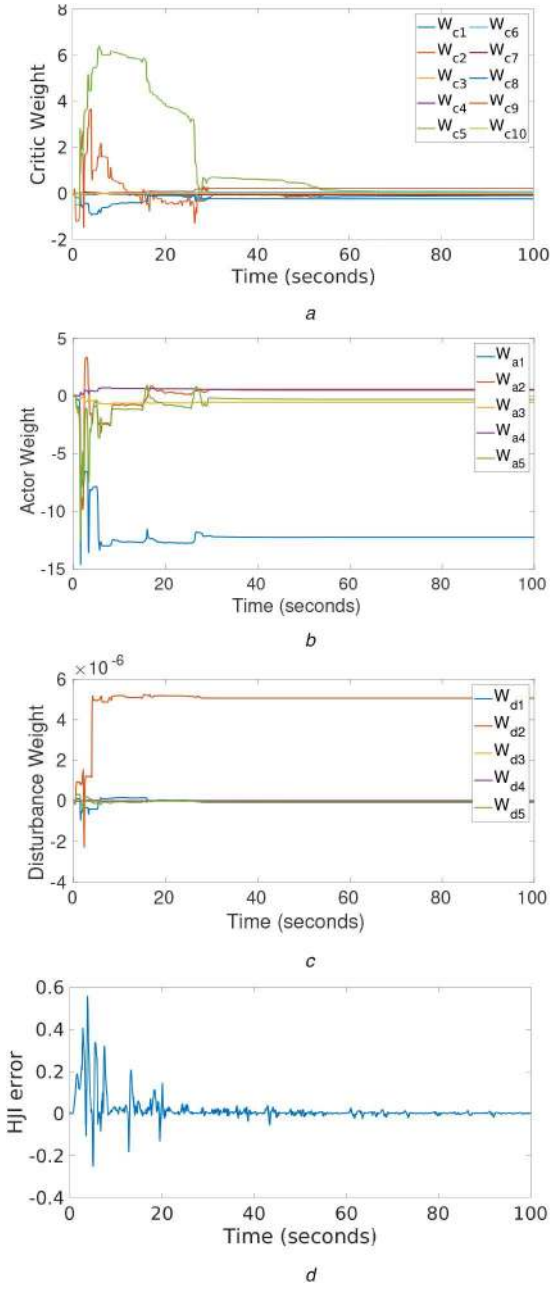

Fig. 2 *Online training of NN weights and HJI error*
(a) Critic NN weights, ***(b)*** Actor NN weight, ***(c)*** Disturbance NN weights, ***(d)*** HJI error during the learning phase

The penalty on states as appearing in (13), i.e. Q_1 is chosen to be

$$Q_1 = \begin{pmatrix} \text{diag}(217, 0) & 0 \\ 0 & 0 \end{pmatrix} \tag{76}$$

Here, x_{d1} is the desired trajectory for the output $y = x_1$. The inital state of the system is (.5; .5). The regressor vectors for critic, actor and disturbance NNs are chosen as

$$\begin{aligned} \sigma_c &= \left(z_1^2,\ z_2^2,\ z_3^2,\ z_4^2,\ z_1z_2,\ z_1z_3,\ z_1z_4,\ z_2z_3,\ z_2z_4,\ z_3z_4\right) \\ \sigma_a &= \left(z_1,\ z_2,\ z_3,\ z_4,\ z_1z_2^2\right) \\ \sigma_d &= \left(z_1^2,\ z_2^2,\ z_1z_3,\ z_1z_4,\ z_1z_2\right) \end{aligned} \tag{77}$$

where $z = (e^{\mathrm{T}}, x_d^{\mathrm{T}})^{\mathrm{T}} \in \mathbb{R}^4$ and z_i is individual component of z. The exploratory control signal considered has the form, $u(t) = 2e^{-0.009t}(\sin(11.9t)^2\cos(19.5t) + \sin(2.2t)^2\cos(5.8t) + \sin(1.2t)^2\cos(9.5t) + \sin(2.4t)^5)$
similar to the one mentioned in [22]. The constant part of the

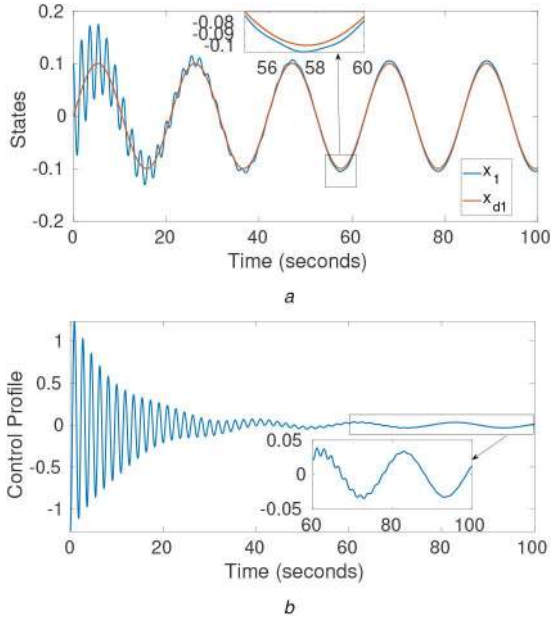

Fig. 3 *State and control profile with constant learning-based gradient descent for non-linear system*
(a) State profile, ***(b)*** Control profile

learning rate for both the cases is selected to be, $\eta = 2998$, the size of memory stack, i.e. N for ER technique is chosen to be 20. The level of attenuation α is choosen to be 0.01. The value of reinforcement interval should be selected as small as possible in order to preserve the relationship between Bellman equation and IRL equation (refer to Section 3.1). Here, the reinforcement interval T is selected as 0.001 s. All the NN weights are initialised to 0. Also, note that in order to yield tighter residual set, $\lambda_{\min}(M')$ in the denominator of RHS of (73) needs to be large, which could be made possible by selecting the gains in robust terms K_1 and K_2 with high norms. However, it should also be noted that since norms of both K_1 and K_2 appear in numerator of RHS of (73) too. Hence, gains K_1 and K_2 cannot be selected with very high norms. For ease in the simulation study, K_1 and K_2 are both selected as 0_q and $(0_{q\times q})$ (for both the cases), where q is the dimension of composite regressor vector ρ [refer to (53)], i.e. $q = 19$.

5.1.1 Validation of continuous-time update law presented in [18] with disturbance terms: It could be noted that the update law presented in this paper [see (36) or (54)] without the variable gain terms (i.e. $k_1 = 0$) and robust terms (i.e. $K_1 = 0, K_2 = 0$) is similar to the one presented in [18] (except the presence of disturbance terms via the composite regressor vector ρ or ρ_1). Simulation results for the update law (54) without variable gain gradient descent and robust terms on the non-linear system considered above (74) are shown in Figs. 2 and 3. The NN weights corresponding to critic, actor and disturbance are shown to converge in finite amount of time in Figs. 2*a*–c, respectively. Fig. 2*d* shows the HJI error during the learning phase when constant learning-based update law was used. The final learnt control policy due to the converged weights of critic, actor and disturbance NNs is depicted in Fig. 3*b*. The stopping condition for the NN weights for all the cases was set as $\| W_{k+1} - W_k \| \leq 10^{-5}$. The final learnt policy is able to make the output of the system track the desired reference trajectory as can be seen in Fig. 3*a*. However, It can be observed that there still exists a lot of transient oscillations and small steady-state error in tracking performance (see Fig. 3*a*).

5.1.2 Validation of off-policy IRL algorithm presented in this paper: Variable gain gradient descent-based update law (54) is validated on the non-linear system (74) in Figs. 4 and 5. Here, the exponent in variable gain term, i.e. (k_1) is chosen to be 0.145. All other parameters are kept same. NN weights of critic, actor and disturbance NNs converge very close to their ideal values in a

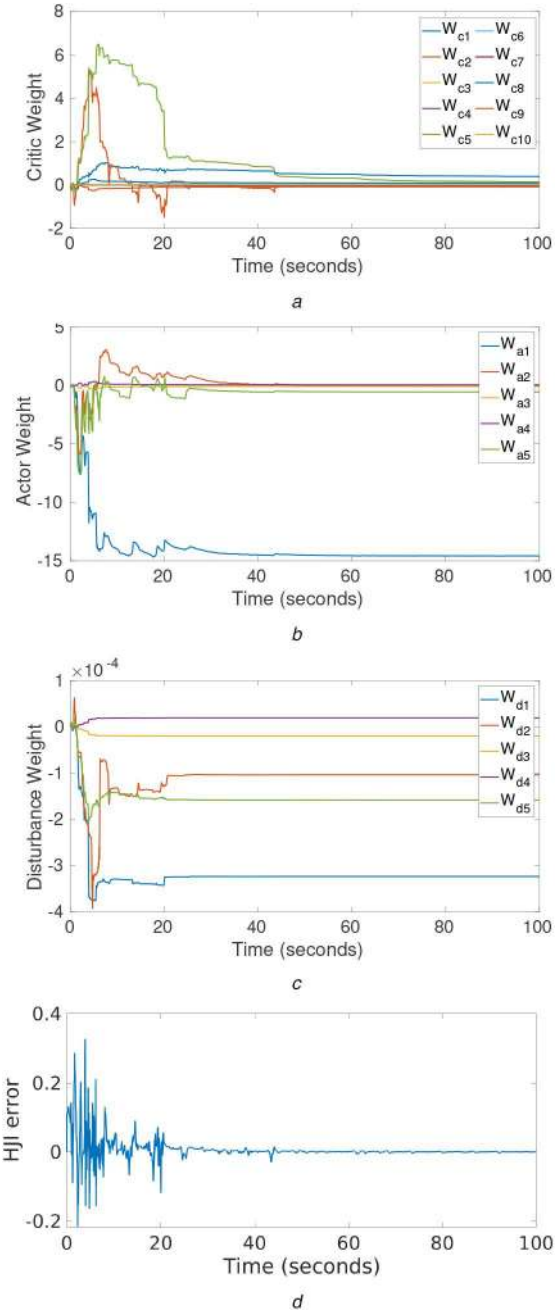

Fig. 4 *Online training of NN weights and HJI error under variable gain-based gradient descent for non-linear system*
(a) Critic NN weights, ***(b)*** Actor NN weight, ***(c)*** Disturbance NN weights, ***(d)*** HJI error during the learning phase

finite amount of time as can be seen in Figs. 4*a*–*c*, respectively. The HJI error profile during the learning phase is depicted in Fig. 4*d* and it can be seen that $|\hat{e}| \leq 1$ during the learning phase. The learnt control policy arising out of the converged NN weight is depicted in Fig. 5*b*. It is able to track the reference trajectory with high accuracy in finite amount of time as evident from Fig. 5*a*.

Note that the oscillations in learnt control policies (see Fig. 3*b*) are more and persist for longer duration in the case when constant learning rate was used as compared to the case when variable gain gradient descent (Fig. 5*b*) is utilised. This in turn leads to an oscillatory tracking performance (Fig. 3*a*) in the transient phase with steady-state error for the case with constant learning speed. On the other hand, the final learnt policies arising out of variable gain gradient descent-based update law leads to very less oscillations and almost no steady-state error (Fig. 5*a*). All this is possible because, the variable gain gradient descent-based update law leads to a much tighter residual set for $\tilde{W}$. This implies that the control policies resulting out of variable gain gradient descent-based update law are closer to the ideal optimal controller than the policies due to just the constant learning rate gradient descent-based update laws. It could also be noted from Figs. 2*d* and 4*d* that the HJI error is within the $[-1, 1]$, and since variable gain gradient descent uses a learning rate that is function of instantaneous HJI error, for our problem set, the presence of term $g_1 = |\hat{e}|^{k_1}$ actually reduces the learning rate (refer to the discussion in Section 4.3). This is also the reason why in this case, the convergence time of Fig. 4*a* is slightly longer than Fig. 2*a*. However, when HJB or HJI error is large ($|\hat{e}| > 1$), the variable gain gradient descent-based update law leads to faster convergence of NN weights as can be observed in [36].

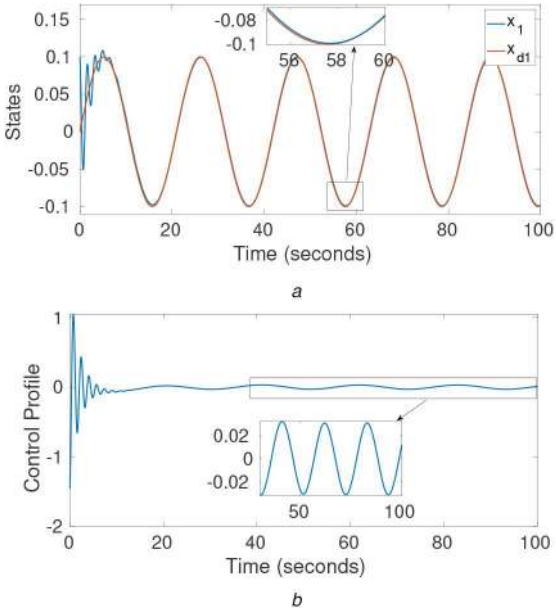

Fig. 5 *State and control profile with variable gain-based gradient descent for non-linear system*
(a) State profile, ***(b)*** Control profile

5.2 Linearised F16 model

The linearised F16 model is considered from [29] with dynamics as follows:

$$\dot{x} = Ax + Bu + Dd \tag{78}$$

where

$$A = \begin{pmatrix} -1.01887 & .90506 & -.00215 \\ .82225 & -1.07741 & -.17555 \\ 0 & 0 & -1 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 0 \\ 5 \end{pmatrix}, \quad D = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \tag{79}$$

The state vector $x = [\alpha, q, \delta_e]^{\mathrm{T}}$, where α is angle of attack (AoA), q is the pitch rate and δ_e is the elevator deflection. The control input is the voltage signal to the elevators and disturbance is caused by the wind gust to the AoA. It is required to track constant reference AoA which is given by

$$\alpha_d = \begin{cases} 2 & \forall t < 30 \\ 3 & \forall t \geq 30 \end{cases} \tag{80}$$

For this, the augmented dynamics of $z = [e^{\mathrm{T}}, x_d^{\mathrm{T}}]^{\mathrm{T}}$ is given as

$$\dot{z} = A_1 z + B_1 u + D_1 d \tag{81}$$

where

$$A_1 = \begin{pmatrix} A & A \\ 0_{3\times 3} & 0_{3\times 3} \end{pmatrix}, B_1 = \begin{pmatrix} B \\ 0_{3\times 1} \end{pmatrix}, D_1 = \begin{pmatrix} D \\ 0_{3\times 1} \end{pmatrix} \tag{82}$$

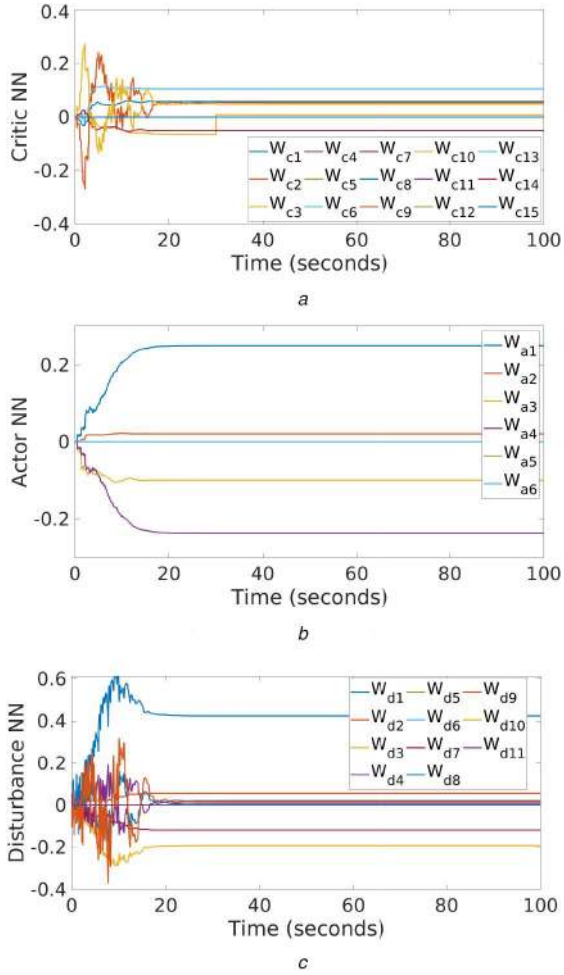

Fig. 6 *Online training of NN weights for F16 model*
(a) Critic NN weights, ***(b)*** Actor NN weight, ***(c)*** Disturbance NN weights

Note that since in this example the control system is required to track constant AoA (set-point tracking), we have $\dot{x}_d = 0$. The problem set-up in this section is considered in line with that in [29], where the same problem was considered using least-square-based update law. The disturbance is assumed to have the form, $d = .1e^{-.1t}\sin(.1t)$. Now, a comparison in performance between H_∞-tracking controller developed using off-policy IRL algorithm presented in this paper and that of [29] will be made in Section 5.2.1. Subsequently, the off-policy IRL algorithm presented in this paper will be compared against the traditional GARE approach in Section 5.2.2.

5.2.1 Comparison with RLS-based update law of [29]: In this section, model-free off-policy IRL-based controller will be validated on F-16 model. The parameters required for IRL are, reinforcement interval $T = .001$ s and $R = 1, Q_1 = \text{diag}([10, 0, 0, 0, 0, 0])$. Discount factor $\gamma = 0.33$ was chosen for simulation. In the simulation, the desired value of output was $\alpha_d = 2$ for first 30 s and then was subsequently changed to $\alpha_d = 3$ thereafter. The constant part of the learning rate is selected as $\eta = 209.1$ with variable gain exponent k_1 in $|\hat{e}|^{k_1}$ as $k_1 = 0.2$. The regressor vectors for critic, actor and disturbance NNs were chosen to be

$$\begin{aligned}\sigma_c &= \begin{pmatrix} z_1z_2, z_1z_3, z_1z_4, z_1z_5, z_1z_6, z_2z_3, z_2z_4, z_2z_5, z_2z_6, \ldots \\ z_3z_4, z_3z_5, z_3z_6, z_4z_5, z_4z_6, z_5z_6 \end{pmatrix}^{\mathrm{T}} \\ \sigma_a &= (z_1, z_2, z_3, z_4, z_5, z_6)^{\mathrm{T}} \\ \sigma_d &= (z_1, z_2, z_3, z_4, z_5, z_6, z_1z_2, z_1z_3, z_1z_4, z_1z_5, z_1z_6)^{\mathrm{T}}\end{aligned} \tag{83}$$

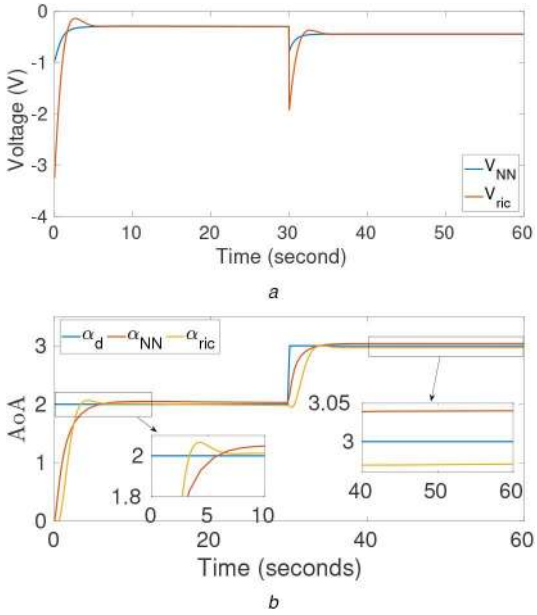

Fig. 7 *Optimal control profile and AoA profile of linearised F-16 aircraft pitch dynamics model under presented off-policy IRL algorithm and GARE*
(a) Optimal control policy for F16 model, ***(b)*** AoA profile for F16 model

The exploratory control policy used during the learning phase is given by
$u(t) = 2e^{(-.009t)}(\sin(t)^2\cos(t) + \sin(3t)^4\cos(1.5t) + \sin(9t)^2\cos(8.4t).$
$+ \sin(3.9t)\cos(2.9t)\sin(19t) + \sin(11.9t)\cos(5.3t)^2 + \sin(12t)\cos$
$(2.5t)^4 + \sin(15t)\cos(1.62t)^2)$
Note from Figs. 6*a–c* that the NN weights converge close to their ideal values in finite amount of time. The disturbance attenuation factor was chosen to be $\alpha = 10$, i.e.. same as in [29]. Unlike [29], the continuous-time update law presented in this paper can produce updates for NN weights instantly in each sampling interval without having to wait for collecting N different samples in each phase. The final learnt control policy developed using off-policy IRL algorithm presented in this paper is marked V_{NN} in Fig. 7*a* and the AoA profile resulting out of this learnt policy is marked α_{NN} in Fig. 7*b*. Observing α_{NN} in Fig. 7*b* and Fig. 4 in [29], one can conclude that there are no peak overshoots in α_{NN}. Finally, all the states of linearised F-16 aircraft pitch dynamics model under presented control scheme are plotted in Fig. 8*a* and the approximated cost function is shown in Fig. 8*b*.

5.2.2 Comparison with game algebraic riccati equation (GARE): It could be noted that HJI equation for linear time invariant systems can be represented as GARE with optimal policies for control and disturbance as

$$u^* = -R^{-1}B_1^{\mathrm{T}}Pz;\ d^* = \frac{1}{\alpha^2}D_1^{\mathrm{T}}Pz \tag{84}$$

where P is a symmetric matrix that satisfies the GARE, i.e.

$$Q_2^{\mathrm{T}} + A_2^{\mathrm{T}}P + PA_2 - \gamma P - PB_1R^{-1}B_1^{\mathrm{T}}P + \frac{1}{\alpha^2}PD_1D_1^{\mathrm{T}}P = 0 \tag{85}$$

where $C = [1, 0, 0, 0, 0, 0]^{\mathrm{T}}$ and matrix $A_2 = [A, A; 0_{3\times 6}] \in \mathbb{R}^{6\times 6}$, $Q_2 = 20CC^{\mathrm{T}}$. Discount and attenuation are chosen as $\gamma = 0.33$ and $\alpha = 10$, respectively. It can be converted into algebraic Riccati equation (ARE) as [29]

$$\begin{aligned}&Q_2^{\mathrm{T}} + (A_2 - \frac{1}{2}\gamma I)^{\mathrm{T}}P + P(A_2 - \frac{1}{2}\gamma I) \\ &-P(B_1R^{-1}B_1^{\mathrm{T}} - \frac{1}{\alpha^2}D_1D_1^{\mathrm{T}})P = 0\end{aligned} \tag{86}$$

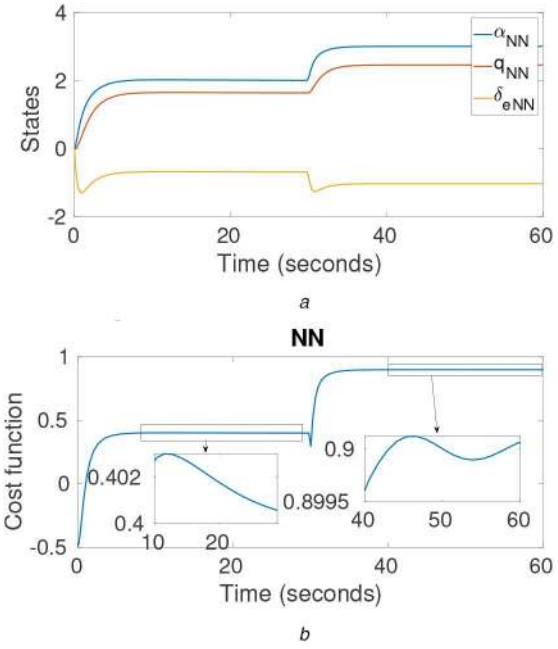

Fig. 8 *State profile and approximated cost under NN-based control law for linearised F-16 aircraft pitch dynamics model*

(a) State profile of F-16 under NN-based control law, ***(b)*** Approximated cost under NN-based control law for F16 model

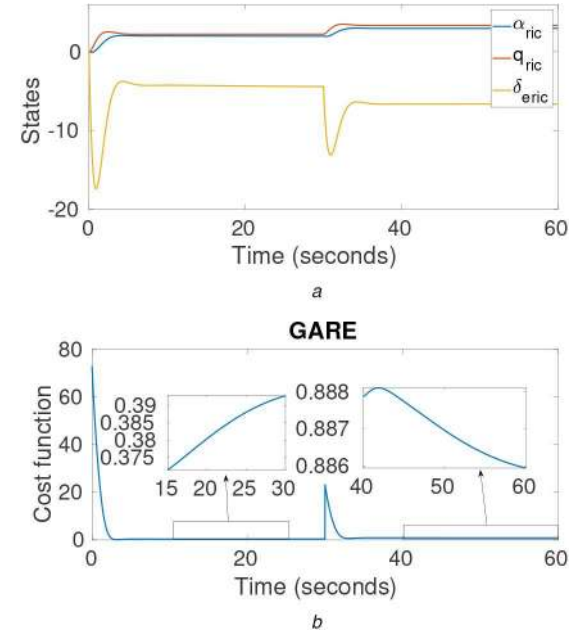

Fig. 9 *State profile and cost under GARE for linearised F-16 aircraft pitch dynamics model*

(a) State profile of F-16 under GARE, ***(b)*** Cost under GARE

The ARE in (86) is solved using Matlab's 'care' routine and the final gain P is given by

$$P = \begin{pmatrix} 8.998 & 3.594 & -0.283 & -5.030 & 3.594 & -0.283 \\ 3.594 & 2.197 & -0.215 & -3.289 & 2.197 & -0.215 \\ -0.283 & -0.215 & 0.026 & 0.359 & -0.215 & 0.026 \\ -5.030 & -3.289 & 0.359 & 5.529 & -3.289 & 0.359 \\ 3.594 & 2.197 & -0.215 & -3.289 & 2.197 & -0.215 \\ -0.283 & -0.215 & 0.026 & 0.359 & -0.215 & 0.026 \end{pmatrix} \quad (87)$$

Unlike the algorithm presented in this paper, solving GARE requires knowledge of system dynamics. The plots marked V_{NN} and V_{ric} in Fig. 7*a* show the voltage profile under model-free off-policy IRL developed in this paper and GARE solution, respectively. The tracking performance of the control effort from both these algorithms is depicted in Fig. 7*b*, in which α_{des}, α_{NN} and α_{ric} are desired AoA, AoA under presented off-policy IRL and AoA under GARE solution, respectively. All the states of linearised F-16 aircraft pitch dynamics model under GARE are shown in Fig. 9*a* and the corresponding cost function is plotted in Fig. 9*b*. A small overshoot can be seen in AoA tracking under GARE solution, when compared against algorithm presented in this paper. A similar overshoot could also be observed in tracking performance presented in [29] (refer to Fig. 4 of [29]) for the same problem set-up. Compared to these results, the AoA tracking under off-policy IRL presented in this paper is devoid of any significant peak overshoot. In addition, the approximated cost in the case of presented update law (as shown in Fig. 8*b*) converges to a value of 0.9 which is very close to the value (= 0.886) the GARE solution cost converges to in Fig. 9*b*.

It could be noted that the variable gain gradient descent-based continuous-time update law presented in this paper yields better tracking performance on non-linear system when compared against [18] in Section 5.1. As compared to the RLS-based update law of [29], the update law presented in this paper is more sensitive to parametric variations during learning phase and does not produce any peak overshoots when final learnt policies are executed as observed in Section 5.2.1. Finally, it is also observed that in the case of linear system, the tracking performance of presented control scheme is very close to that of the GARE control solution in steady state.

6 Conclusion

A continuous-time NN parameter update law driven by variable gain gradient descent, ER technique and robust terms for model-free H_∞ OTCP of continuous-time non-linear system has been presented in this paper. IRL has been leveraged in policy iteration framework in this paper. Incorporation of IRL obviates the requirement of drift dynamics in policy evaluation stage, while usage of actor and disturbance NNs to approximate control and disturbance policies obviates the requirement of control coupling dynamics and disturbance dynamics in policy improvement stage. Variable gain gradient descent increases the learning rate when HJI error is large and it dampens the learning rate when HJI error becomes smaller. It also results in smaller residual set over which the errors in NN weights converge to. Besides this, the ER term and robust terms in the update law help in further shrinking the size of the residual set on which the error in NN weights finally converge to. This results in an improved learnt control policy, sufficiently close to the ideal optimal controller, leading to highly accurate tracking performance.

7 References

[1] Sutton, R.S., Barto, A.G.: '*Introduction to reinforcement learning*', vol. 2 (MIT press Cambridge, Cambridge, UK, 1998)

[2] Lewis, F.L., Liu, D.: ''*Reinforcement learning and approximate dynamic programming for feedback control*', vol. 17 (John Wiley & Sons, New Jersey, USA, 2013)

[3] Powell, W.B.: ''*Approximate dynamic programming: solving the curses of dimensionality*', vol. 703 (John Wiley & Sons, New Jersey, USA, 2007)

[4] Zhang, H., Liu, D., Luo, Y., *et al.*: '*Adaptive dynamic programming for control: algorithms and stability*' (Springer Science & Business Media, London, UK, 2012)

[5] Murray, J.J., Cox, C.J., Lendaris, G.G., *et al.*: 'Adaptive dynamic programming', *IEEE Trans. Syst. Man Cybern., Part C*, 2002, **32**, (2), pp. 140–153

[6] Abu-Khalaf, M., Lewis, F.L.: 'Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network hjb approach', *Automatica*, 2005, **41**, (5), pp. 779–791

[7] Li, H., Liu, D.: 'Optimal control for discrete-time affine non-linear systems using general value iteration', *IET Control Theory Applic.*, 2012, **6**, (18), pp. 2725–2736

[8] Yang, X., Liu, D., Wei, Q.: 'Online approximate optimal control for affine non-linear systems with unknown internal dynamics using adaptive dynamic programming', *IET Control Theory Applic.*, 2014, **8**, (16), pp. 1676–1688

[9] Zhao, D., Zhu, Y.: 'Mec–a near-optimal online reinforcement learning algorithm for continuous deterministic systems', *IEEE Trans. Neural Netw. Learning Syst.*, 2014, **26**, (2), pp. 346–356

[10] Zhu, Y., Zhao, D., Liu, D.: 'Convergence analysis and application of fuzzy-hdp for nonlinear discrete-time hjb systems', *Neurocomputing*, 2015, **149**, pp. 124–131

[11] Bhasin, S., Kamalapurkar, R., Johnson, M., *et al.*: 'A novel actor–critic–identifier architecture for approximate optimal control of uncertain nonlinear systems', *Automatica*, 2013, **49**, (1), pp. 82–92

[12] Park, Y.M., Choi, M.S., Lee, K.Y.: 'An optimal tracking neuro-controller for nonlinear dynamic systems', *IEEE Trans. Neural Netw.*, 1996, **7**, (5), pp. 1099–1110

[13] Toussaint, G.J., Basar, T., Bullo, F.: 'H_∞-optimal tracking control techniques for nonlinear underactuated systems'. Proc. of the 39th IEEE Conf. on Decision and Control (Cat. No. 00CH37187) (IEEE), Sydney, Australia, 2000, vol. 3, pp. 2078–2083

[14] Alameda-Hernandez, E., Blanco, D., Ruiz, D., *et al.*: 'Optimal tracking of time-varying systems with the overdetermined recursive instrumental variable algorithm', *IET Control Theory Applic.*, 2007, **1**, (1), pp. 291–297

[15] Zhang, H., Cui, L., Zhang, X., *et al.*: 'Data-driven robust approximate optimal tracking control for unknown general nonlinear systems using adaptive dynamic programming method', *IEEE Trans. Neural Netw.*, 2011, **22**, (12), pp. 2226–2236

[16] Modares, H., Lewis, F.L.: 'Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning', *Automatica*, 2014, **50**, (7), pp. 1780–1792

[17] Kiumarsi, B., Lewis, F.L., Modares, H., *et al.*: 'Reinforcement q-learning for optimal tracking control of linear discrete-time systems with unknown dynamics', *Automatica*, 2014, **50**, (4), pp. 1167–1175

[18] Zhu, Y., Zhao, D., Li, X.: 'Using reinforcement learning techniques to solve continuous-time non-linear optimal tracking problem without system dynamics', *IET Control Theory Applic.*, 2016, **10**, (12), pp. 1339–1347

[19] Vrabie, D., Pastravanu, O., Abu-Khalaf, M.: 'Adaptive optimal control for continuous-time linear systems based on policy iteration', *Automatica*, 2009, **45**, (2), pp. 477–484

[20] Jiang, Y., Jiang, Z.P.: 'Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics', *Automatica*, 2012, **48**, (10), pp. 2699–2704

[21] Luo, B., Wu, H.N., Huang, T., *et al.*: 'Data-based approximate policy iteration for nonlinear continuous-time optimal control design', arXiv preprint arXiv:13110396, 2013

[22] Vamvoudakis, K.G., Vrabie, D., Lewis, F.L.: 'Online adaptive algorithm for optimal control with integral reinforcement learning', *Int. J. Robust Nonlinear Control*, 2014, **24**, (17), pp. 2686–2710

[23] Abu-Khalaf, M., Lewis, F.L., Huang, J.: 'Neurodynamic programming and zero-sum games for constrained control systems', *IEEE Trans. Neural Netw.*, 2008, **19**, (7), pp. 1243–1252

[24] Zhang, H., Wei, Q., Liu, D.: 'An iterative adaptive dynamic programming method for solving a class of nonlinear zero-sum differential games', *Automatica*, 2011, **47**, (1), pp. 207–214

[25] Vamvoudakis, K.G., Lewis, F.L.: 'Online solution of nonlinear two-player zero-sum games using synchronous policy iteration', *Int. J. Robust Nonlinear Control*, 2012, **22**, (13), pp. 1460–1483

[26] Modares, H., Lewis, F.L., Sistani, M.B.N.: 'Online solution of nonquadratic two-player zero-sum games arising in the h control of constrained input systems', *Int. J. Adapt. Control Signal Process.*, 2014, **28**, (3-5), pp. 232–254

[27] Vrabie, D., Lewis, F.: 'Adaptive dynamic programming for online solution of a zero-sum differential game', *J. Control Theory Appl.*, 2011, **9**, (3), pp. 353–360

[28] Luo, B., Wu, H.N., Huang, T.: 'Off-policy reinforcement learning for H_∞ control design', *IEEE Trans. Cybern.*, 2014, **45**, (1), pp. 65–76

[29] Modares, H., Lewis, F.L., Jiang, Z.P.: 'H_∞ tracking control of completely unknown continuous-time systems via off-policy reinforcement learning', *IEEE Trans. Neural Netw. Learning Syst.*, 2015, **26**, (10), pp. 2550–2562

[30] Zhang, H., Cui, X., Luo, Y., *et al.*: 'Finite-horizon H_∞ tracking control for unknown nonlinear systems with saturating actuators', *IEEE Trans. Neural Netw. Learning Syst.*, 2017, **29**, (4), pp. 1200–1212

[31] Liu, D., Yang, X., Wang, D., *et al.*: 'Reinforcement-learning-based robust controller design for continuous-time uncertain nonlinear systems subject to input constraints', *IEEE Trans. Cybern.*, 2015, **45**, (7), pp. 1372–1385

[32] Başar, T., Bernhard, P.: '*H∞ optimal control and related minimax design problems: a dynamic game approach*' (Springer Science & Business Media, New Jersey, USA, 2008)

[33] Liu, D., Yang, X., Li, H.: 'Adaptive optimal control for a class of continuous-time affine nonlinear systems with unknown internal dynamics', *Neural Comput. Appl.*, 2013, **23**, (7-8), pp. 1843–1850

[34] Vrabie, D., Lewis, F.: 'Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems', *Neural Netw.*, 2009, **22**, (3), pp. 237–246

[35] Wu, H.N., Luo, B.: 'Neural network based online simultaneous policy update algorithm for solving the hji equation in nonlinear H_∞ control', *IEEE Trans. Neural Netw. Learning Syst.*, 2012, **23**, (12), pp. 1884–1895

[36] Mishra, A., Ghosh, S.: 'Variable gain gradient descent-based reinforcement learning for robust optimal tracking control of uncertain nonlinear system with input-constraints', arXiv preprint arXiv:191104157, 2019

[37] Modares, H., Lewis, F.L., Naghibi-Sistani, M.B.: 'Adaptive optimal control of unknown constrained-input systems using policy iteration and neural networks', *IEEE Transactions on Neural Networks and Learning Systems*, 2013, **24**, (10), pp. 1513–1525

8 Appendix

8.1 Proof of Proposition 1

Proof:

$$x^{\mathrm{T}}Mx = x^{\mathrm{T}}\left(\frac{M+M^{\mathrm{T}}}{2} + \frac{M-M^{\mathrm{T}}}{2}\right)x \tag{88}$$

RHS of above equation can be rewritten as

$$\begin{aligned} x^{\mathrm{T}}Mx &= x^{\mathrm{T}}\left(\frac{M+M^{\mathrm{T}}}{2}\right)x + .5x^{\mathrm{T}}Mx - .5x^{\mathrm{T}}M^{\mathrm{T}}x \\ &= x^{\mathrm{T}}\left(\frac{M+M^{\mathrm{T}}}{2}\right)x + .5x^{\mathrm{T}}Mx - .5(x^{\mathrm{T}}Mx)^{\mathrm{T}} \end{aligned} \tag{89}$$

Therefore

$$x^{\mathrm{T}}Mx = x^{\mathrm{T}}\left(\frac{M+M^{\mathrm{T}}}{2}\right)x \tag{90}$$

Using (90)

$$\lambda_{\min}\left(\frac{M+M^{\mathrm{T}}}{2}\right)\| x \|^2 \leq x^{\mathrm{T}}Mx \leq \lambda_{\max}\left(\frac{M+M^{\mathrm{T}}}{2}\right)\| x \|^2 \tag{91}$$

□

8.2 Proof of Theorem 1

Proof: In first part of the proof, the admissibility of the improved policies will be proved, thereafter it will be shown that $V_i \geq V_{i+1} \geq V^*$. Observe that $V_i \geq 0$ (due to the definition of V) and $V_i(z(t)) = 0$ iff, $z(t) = 0$. Further, $\nabla V(z(t))|_{z=0} = 0$, this leads to $u_{i+1} = 0$ and $d_{i+1} = 0$ when $z = 0$. Now, rate of variation of V_i along the trajectory generated by improved policies (u_{i+1}, d_{i+1}) is

$$\dot{V}_i(z, u_{i+1}, d_{i+1}) = \nabla V_i^{\mathrm{T}}(F + Gu_{i+1} + Kd_{i+1}) \tag{92}$$

Since, V_i, u_i and d_i satisfy (20), $\nabla V_i^{\mathrm{T}}F$ can be written as

$$\nabla V_i^{\mathrm{T}}F = -\nabla V_i^{\mathrm{T}}(Gu_i + Kd_i) + \gamma V_i - z^{\mathrm{T}}Q_1 z - u_i^{\mathrm{T}}Ru_i + \alpha^2 d_i^{\mathrm{T}}d_i \tag{93}$$

Using (93) in (92), with $\bar{Q}(z) \triangleq z^{\mathrm{T}}Q_1 z = e^{\mathrm{T}}Qe$.

$$\begin{aligned} \dot{V}_i(z, u_{i+1}, d_{i+1}) &= -\nabla V_i^{\mathrm{T}}(Gu_i + Kd_i) + \gamma V_i - z^{\mathrm{T}}Q_1 z - u_i^{\mathrm{T}}Ru_i \\ &\quad + \alpha^2 d_i^{\mathrm{T}}d_i + \nabla V_i^{\mathrm{T}}(Gu_{i+1} + Kd_{i+1}) \\ &= \gamma V_i - \bar{Q}(z) - \underbrace{[u_i^{\mathrm{T}}Ru_i + 2u_{i+1}^{\mathrm{T}}R(u_{i+1} - u_i)]}_{\triangleq a_1} \\ &\quad + \underbrace{[\alpha^2 d_i^{\mathrm{T}}d_i + 2\alpha^2 d_{i+1}^{\mathrm{T}}(d_{i+1} - d_i)]}_{\triangleq a_2} \end{aligned} \tag{94}$$

Shifting γV_i to the LHS of (94) and multiplying both sides by $e^{-\gamma t}$ and using (21)

$$\begin{aligned} \frac{de^{-\gamma t}V_i}{dt} = e^{-\gamma t}\Bigg[&-\bar{Q}(z) - \underbrace{\left(\sum_{k=1}^{m} R_k(u_{i+1,k} - u_{i,k})^2 + \sum_{k=1}^{m} R_k u_{i+1,k}^2\right)}_{a_1} \\ &+ e^{-\gamma t}\underbrace{\alpha^2\left[\sum_{k=1}^{m}(d_{i+1,k} - d_{i,k})^2 + \sum_{k=1}^{m} d_{i+1,k}^2\right]}_{a_2} \end{aligned} \tag{95}$$

where (i,k) and $(i+1,k)$ represent kth component of ith and $i+1$th policies. Now, if $\gamma = 0$ and $\alpha = 0$, then $\dot{V}_i(z, u_{i+1}, d_{i+1}) < 0$ along the augmented system trajectories generated by improved policies

(u_{i+1} and d_{i+1}). This proves that, improved policies lead to asymptotic stability of e when $\gamma = 0, \alpha = 0$ and hence are admissible. However, when $\gamma \neq 0$ and $\alpha \neq 0$, then from (94), $\dot{V}_i(z, u_{i+1}, d_{i+1}) < 0$ can be analysed in two cases, Case (i): When $\gamma V_i \leq a_1 - a_2$. In this case, $\dot{V}_i(z, u_{i+1}, d_{i+1}) < 0$ for all the values of e and hence e is said to be asymptotically stable. Case (ii): When $\gamma V_i \geq a_1 - a_2$. In this case, $\dot{V}_i(z, u_{i+1}, d_{i+1}) < 0$ if $\| e \| > \sqrt{(\gamma V_i - a_1 + a_2)/\lambda_{\min}(Q)}$. This inequality is the UUB set for state error. For the second part of the proof, derivative of V_i and V_{i+1} needs to be taken along the augmented system trajectory produced by u_{i+1}, d_{i+1}, i.e.

$$V_{i+1} - V_i = -\int_t^\infty \left(\frac{d(V_{i+1} - V_i)}{dx}\right)^{\mathrm{T}} (F + Gu_{i+1} + Kd_{i+1}) \mathrm{d}\tau \quad (96)$$

Using (20) for $\nabla V_{i+1}{}^{\mathrm{T}} F$ and $\nabla V_i{}^{\mathrm{T}} F$ along with (21) in (96),

$$\begin{aligned} &V_{i+1} - V_i = \\ &-\int_t^\infty \Big[[2u_{i+1}^{\mathrm{T}} R(u_{i+1} - u_i) + u_i^{\mathrm{T}} R u_i - u_{i+1}^{\mathrm{T}} R u_{i+1}] + \\ &[2\alpha^2 d_{i+1}^{\mathrm{T}}(d_i - d_{i+1}) + \alpha^2 d_{i+1}^{\mathrm{T}} d_{i+1} - \alpha^2 d_i^{\mathrm{T}} d_i] - \gamma(V_i - V_{i+1}) \Big] \mathrm{d}\tau \end{aligned} \quad (97)$$

It can be further simplified as

$$\begin{aligned} V_{i+1} - V_i = -\int_t^\infty \Big[\underbrace{\sum_{k=1}^{m} \left(\underbrace{R_k (u_{i+1,k} - u_{i,k})^2}_{\triangleq a_3} - \underbrace{\alpha^2 (d_{i+1,k} - d_{i,k})^2}_{\triangleq a_4} \right)}_{\triangleq b_1} \\ -\gamma(V_i - V_{i+1}) \Big] \mathrm{d}\tau \end{aligned} \quad (98)$$

Now, let $\mathscr{F}(t) \triangleq V_{i+1}(z(t)) - V_i(z(t))$, therefore above equation can be written as

$$\mathscr{F}(t) = -\gamma \int_t^\infty \mathscr{F}(\tau) \mathrm{d}\tau - \int_t^\infty b_1 \mathrm{d}\tau \quad (99)$$

Differentiating (99) using Leibniz rule, one obtains $\dot{\mathscr{F}}(t) - \gamma \mathscr{F}(t) = b_1$, now supposing $b_1 > 0$ and multiplying both sides of this equation with $e^{-\gamma t}$

$$\frac{de^{-\gamma t}\mathscr{F}(t)}{dt} = e^{-\gamma t} b_1 \Rightarrow \frac{de^{-\gamma t}\mathscr{F}(t)}{dt} > 0 \quad (100)$$

Integrating both sides of the final inequality of (100) over $[t, \infty]$, we obtain, $\mathscr{F}(t)e^{-\gamma t} < 0$ or $\mathscr{F}(t) < 0$, or, $V_{i+1} < V_i$. Therefore, $V_{i+1} < V_i$ only if $b_1 > 0$, which happens if $a_3 > a_4$ in (98). This can be ensured by suitable selection of diagonal matrix R and disturbance attenuation factor α. Also, as u_i approaches u^* and d_i approaches d^* in (98), b_1 approaches zero, which implies from (100) that V_i approaches V^∞. Thus, by monotone convergence theorem, V_i converges point-wise to $V^\infty = V^*$, the unique minima of V over the compact set Ω_1. Therefore, $V_1 > V_2 > V_3 > \cdots > V_{i+1} > \cdots \geq V^\infty = V^*$. □