Applications of integral reinforcement learning control in electrical machines and power converter systems

Yu, Yang

2021

Yu, Y. (2021). Applications of integral reinforcement learning control in electrical machines and power converter systems. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/145783>

<https://doi.org/10.32657/10356/145783>

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Downloaded on 26 Apr 2023 07:43:40 SGT

Applications of Integral Reinforcement Learning Control in Electrical Machines and Power Converter Systems

---

Yu Yang

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

2021

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

23-Nov-2020

.....

Date

.....

Yu Yang

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

23-Nov-2020

.....

Date

.....

Prof. Su Rong

Authorship Attribution Statement

This thesis contains material from 3 paper(s) published in the following peer-reviewed journal(s) / from papers accepted at conferences in which I am listed as an author.

Chapter 3 is published as Yang Yu, Shuai Wang, Rong Su V. Viswanathan, S. Ramakrishna, C. J. Gajanayake, and Amit K. Gupta. "Application of Off-policy Integral Reinforcement Learning for H_∞ , Input Constrained Control of Permanent Magnet Synchronous Machine." in 2019 IEEE Applied Power Electronics Conference and Exposition (APEC), pp. 2570-2576. IEEE, 2019.

Chapter 3 also published as Yang Yu, Xiaohe Ma, Rong Su, Tseng King Jet, V. Viswanathan, Chandana Jayampathi Gajanayake, S. RamaKrishna, and A. K. Gupta. "Application of integral reinforcement learning for optimal control of a high speed flux-switching permanent magnet machine." in IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society, pp. 2702-2707. IEEE, 2016.

The contributions of the co-authors are as follows:

- Prof Su Rong provided the initial project direction.
- I designed the controller, performed all the laboratory work and prepared the manuscript drafts. The manuscript was revised by Prof Su Rong.
- Mr Wang Shuai, Dr Ma Xiaohe, Prof Tseng King Jet, Dr V. Viswanathan, Mr S. Ramakrishna, Dr C.J Gajanyaka, Dr Amit K. Gupta provided their valuable comments on this draft.

Part of the control technique in chapter 3 is also published in Yicheng Zhang, Yang Yu, Rong Su, and Jian Chen. "Power Scheduling in More Electric Aircraft based on An Optimal Adaptive Control Strategy." accepted by IEEE Transactions on Industrial Electronics (2019).

The contributions of the co-authors are as follows:

- Dr Zhang Yicheng provided the initial project direction and edited the manuscript drafts.
- Dr Zhang Yicheng designed the majority of an optimization algorithm and verified it with simulation.
- I designed part of the optimal controller using integral reinforcement learning.

- Prof Su Rong provided their valuable comments on this draft.

23-Nov-2020

.....

Date

.....

Yu Yang

Acknowledgements

I would like to express my sincere gratitude to my supervisor Professor Su Rong, for his patience, support, and encouragement during my entire journey as a PhD student. He not only taught me the subject matter related to my research but also the opportunities and challenges for my future research direction and career. The research is conducted within the Rolls-Royce @ NTU Corp Lab with financial support from the National Research Foundation (NRF), Singapore, under the Corporate Laboratory @ University Scheme. Hence, I would like to thank Rolls-Royce, NRF and NTU for the opportunity in working in the Corporate Laboratory with continuous guidance from Prof Amit Gupta, Prof Tseng King Jet and Prof Josep Pou.

I am also thankful to my research group members, Dr Ma Xiaohe, Mr Wang Shuai, Mr Loh Jenn Yen, Mr Anand Prabhu Mathivanan, Dr Vaiyapuri Viswanathan, Mr Shanmukha Ramakrishna and Dr Chadnana Gajanayake for their support in Embedded Machine in Hostile Environments project and Modular Power Converter and Distribution System project.

Special appreciation goes to Professor Frank Lewis, Dr Shan Zuo for their guidance and valuable advice in the field of control theory related to reinforcement learning. I also would like to thank my colleagues and friends in the Electrical Power System Integration Lab@NTU (EPSIL@N) for their help and friendship. They are Dr Zhang Yicheng, Dr Zhai Jianyang, Dr Tu Pengfei, Mr FanFei, Dr Shi Donghan, Mr Wang Shuai, Dr Ji Dongxu, Dr Xu Qianwen.

“If I had one hour to save the world, I would spend 55 minutes defining the problem and only five minutes finding the solution.”

—Einstein, Albert

To my dear family

Abstract

With the rapid development of power electronic devices, the ideas of more-electric aircraft (MEA) have become practical with many of the traditional hydraulic and pneumatic systems being replaced with equivalent electrical systems. The MEA concepts have shown promising results in reductions of fuel consumption as in both Airbus A380 and Boeing 787. MEA also motivates the development of segmented machines to better utilize the available space inside the gas turbine. The stator of a segmented machine is non-asymmetrically distributed so that existing pipes and cables of a gas turbine engine can be accommodated in the confined space of this gas turbine engine.

The model of segmented machine is normally derived numerically based on finite element analysis (FEA). Considering the data-driven nature of segmented machine, a promising control method called integral reinforcement learning (IRL) has been gaining attention to deal with the control problems associated with the segmented machine. Living organisms learn by acting on their environment, observing the resulting reward stimulus. IRL controller uses an actor/critic structure and adapts its control actions based on interactions with the data driven model. The idea of IRL based control combines the appealing features of optimal control theory and adaptive control theory.

The first part of this thesis describes how IRL has been successful experimentally implemented into H_∞ control of 2-kW electrical machine, which shows the superiority of an IRL controller over other classical controllers with the distinct advantage of learning an optimal control law without prior knowledge of the system model. However, relevant technical improvements are needed to make IRL more computationally efficient. Meanwhile, convergence of IRL should not rely on the assumption of initial stable controller.

In the second part of this thesis, two theoretical improvements are made. Firstly, existing IRL based H_∞ control algorithms require the update of a disturbance

policy at each iteration due the fact that the standard H_∞ control covers all disturbance (i.e. the worst-case disturbance). However, the pattern or characteristic of the disturbance may be available and the main goal is to develop an effective and computationally inexpensive method to address the given disturbance. Secondly, most IRL works are developed based on policy iteration where an initial stable controller is needed to start the whole iteration process. To this end, value iteration based IRL is developed without the prerequisite requirement of having an initial stable control law which results in a model-free initial condition.

Contents

Acknowledgements	v
Abstract	vii
List of Figures	xii
List of Tables	xiv
Symbols and Acronyms	xv
1 Introduction	1
1.1 Background	1
1.2 Research Motivations	3
1.3 Contributions of This Work	5
1.4 Outline of This Thesis	5
2 Theoretical Background of Integral Reinforcement Learning	7
2.1 Chapter Background	7
2.2 Theoretical background	8
2.3 Policy Iteration based Reinforcement Learning	10
2.4 Tracking a Constant Reference	13
2.4.1 Existing Formulation	14
2.4.2 Direct Formulation of Optimal Tracking Problem	14
2.4.3 Output Regulation Theory	15
2.4.4 Robust Output Regulation Theory	16
2.5 Conclusion	17
3 Integral Reinforcement Learning with an Application to H_∞, Input Constrained Control of a Permanent Magnet Synchronous Machine	18
3.1 Chapter Background	19
3.2 Electrical Machine System Model Description	21
3.3 H_∞ Tracking Controller Design	23

3.4	An Optimal Adaptive Controller Synthesis Based on Off-policy Integral Reinforcement Learning	26
3.5	Implementation of Off-policy Integral Reinforcement Learning for Machine Control	27
3.5.1	Determining the initial control using the Routh-Hurwitz stability criterion	27
3.5.2	Design of the optimal control problem : Q and R selection	28
3.5.3	Procedure of Algorithm 3.2 off-policy integral reinforcement learning	29
3.6	Experimental Verification of Integral Reinforcement Learning controller	30
3.6.1	Drive Setup	30
3.6.2	Learning Process of the Integral Reinforcement Learning Controller	31
3.6.3	Comparison of the proposed controller and PID control method	33
3.6.4	Experimental Discussion	38
3.7	Conclusion	39
4	Data-driven Intermediate H_∞ Control Using Off-policy Integral Reinforcement Learning	40
4.1	Chapter Background	40
4.2	System Model and Problem Formulation	43
4.2.1	Mathematical Model of DC-DC converter	43
4.2.2	Voltage-tracking Problem	44
4.3	Standard H_∞ Control Problem	45
4.4	Intermediate H_∞ Control Problem	48
4.5	Simulation Verification of the Proposed Method	52
4.6	Implementation of The Off-policy IRL for a DC-DC Boost Converter	54
4.6.1	Design of The Cost Function: Q and R Selection	54
4.6.2	Initial Stabilizing Gain	55
4.6.3	Procedure of Algorithm 4.4 Off-policy IRL	56
4.7	Experimental Verification of IRL controller on DC-DC Boost Converter	57
4.7.1	Prototype Setup	57
4.7.2	Comparison of Algorithm 4.4 Based Controller with Other Advanced Control Methods	58
4.8	Conclusions	62
5	Value Iteration based Off-policy Integral Reinforcement Learning and Convergence Analysis	64
5.1	Chapter Background	65
5.2	System Modeling and Problem Formulation	66
5.3	VI Based On-Policy IRL for a Nonlinear Optimal Control Problem	67
5.4	Implementation of Algorithm 5.1 Using Neural Networks	73

5.5	Verification of Convergence Property of Algorithm 5.1	75
5.5.1	Linear System Verification of Convergence Property of Algorithm 5.1	75
5.5.2	Nonlinear System Verification of Convergence Property of Algorithm 5.1	77
5.6	VI based Off-policy IRL for a Completely Unknown Nonlinear System	78
5.6.1	Convergence Property of Algorithm 5.2	79
5.6.2	Implementation of Algorithm 5.2	80
5.7	VI based Off-policy IRL version 2 for Completely Unknown Nonlinear System	81
5.7.1	Simulation Verification of Algorithm 5.4 using the same linear system in Section 5.5	84
5.7.2	Simulation Verification of Algorithm 5.4 using the same nonlinear system in Section 5.5	84
5.8	Conclusion	86
6	Some On-going and Future work	88
6.1	Future Work Topic : Off-policy Value Iteration for H_∞ Regulation of Nonlinear System	88
6.1.1	Background	88
6.1.2	Proposed VI based IRL for H_∞ Control of a Completely Unknown System	90
6.1.3	Preliminary Results	90
6.2	Chapter Conclusion	92
7	Conclusion	93
	List of Author's Awards, Patents, and Publications	95
	Bibliography	97

List of Figures

1.1	More Electric Aircraft	1
1.2	Longitudinal Cross-section More Electric Engine with embedded segmented machine	2
1.3	Transverse cross-section through a segment electrical machine	3
1.4	JMAG-RT numerically based model	4
1.5	IRL control for a numerical plant model	4
2.1	Actor/critic structure for solving the optimal control problem in algorithm 2.1	11
3.1	Flowchart for the implementation of Algorithm 3.2	30
3.2	Block diagram of proposed control system for PMSM drive	31
3.3	Experiment Setup for the PMSM drive	32
3.4	Convergence of the control matrix W_2	33
3.5	Experimental results of speed trajectory under the initial control policy and the last (converged) control policy	34
3.6	Experiment results of input voltages under the initial control law and last (converged) control law	35
3.7	Block diagram of classical PID controller	35
3.8	Experimental results of the conventional PID controller and IRL controller when the desired speed suddenly changes	37
3.9	Experimental results of the conventional PID controller and IRL controller when the load torque suddenly changed	38
4.1	Circuit Diagram of DC-DC boost converter	43
4.2	Convergence of optimal control policy	53
4.3	State trajectory under converged policy u^∞	54
4.4	DC-DC boost converter prototype	57
4.5	Step change of output reference under initial H_∞ controller	59
4.6	Output voltage reference change with final control law	59
4.7	Step change of input voltage under initial H_∞ controller	60
4.8	Step change of input voltage under final control law	60
4.9	Load change with initial H_∞ controller	61
4.10	Load change with final control law	61

5.1	Evaluation of W_1 with different initial value and sampling time for a linear system	76
5.2	Evaluation of W_1 with different initial value and sampling time for a linear system	78
5.3	Evaluation of W_1 and W_2 in a linear system with $T_s=0.1$ under Algorithm 5.4	84
5.4	Evaluation of W_1 and W_2 in a linear system with $T_s=0.2$ under Algorithm 5.4	85
5.5	Evaluation of W_2 with sampling time $T_s = 0.1$ in Algorithm 5.4 . .	86
5.6	State trajectory under u^* in Algorithm 5.1 and u^* in Algorithm 5.4	86
6.1	Evaluation of W_1 under VI IRL Algorithm 6.2 and PI IRL Algorithm 4.2	91

List of Tables

3.1	PMSM TEST RIG PARAMETERS	22
3.2	PARAMETER SETTING FOR THE OPTIMAL CONTROL PROBLEM	31
3.3	COMPARISON PERFORMANCE OF TWO CONTROL METHODS IN THE EXPERIMENT	37
4.1	DC-DC CONVERTER PROTOTYPE PARAMETERS	44
4.2	PARAMETER SETTING FOR THE OPTIMAL CONTROL PROBLEM	55
4.3	COMPONENT LIST of DC CONVERTER PROTOTYPE	58
4.4	PERFORMANCE COMPARISON of THOSE TWO CONTROL METHODS	62

Symbols and Acronyms

Symbols

$\mathcal{R}^n$	the n -dimensional Euclidean space
∇f	the gradient vector of a differential function
$\mathcal{C}^k$	any function
$\mathcal{C}^1$	the function with continuous partial derivatives up to first order

Acronyms

MEA	More Electrical Aircraft
FEM	Finite-Element Model
HILS	Hardware in the Loop Simulations
ECU	Electronic Control Unit
RL	Reinforcement Learning
IRL	Integral Reinforcement Learning
PI	Policy Iteration
VI	Value Iteration
LQR	Linear Quadratic Regulator
HJB	Hamilton-Jacobi-Bellman
HJI	Hamilton-Jacobi-Issac
PMSM	Permanent Magnet Synchronous Machine
PID	Proportional Integral Derivative
MPC	Model Predictive Control
MDP	Markov Decision Process
CT	Continuous-time
DT	Discrete-time

Chapter 1

Introduction

1.1 Background

In order to overcome environmental concerns and achieve aircraft operating cost reduction, aircraft energy efficiency improvement has been the main objective for civil aviation industry [1]. One strategy to increase aircraft energy efficiency and reduce maintenance requirements is to increase the use of electrical power systems as electrical energy is proved to be more efficient than other three kinds of energy (pneumatic, hydraulic and mechanical) that are commonly used in the aircraft [2] [3] [4]. With the advancement of power electronic and more drive system technologies, civil aviation industry is pushing ahead with More Electric Aircraft (MEA) architecture as in Figure 1.1.

FIGURE 1.1: More Electric Aircraft [5]

Many MEA projects are currently being intensified around the world. One of them is the E-Fan X project (2017-2020) that developed the first prototypes for the world's largest hybrid-electric aircraft [6]. A 70-seater demonstrator was developed by Airbus and Rolls Royce.

The concept of MEA emphasizes the utilization of the electrical power to power the non-propulsive aircraft systems. One of the major challenges in this framework is that the electric power is generated by means of electrical generators connected to the jet turbine shaft [7]. An oversized and bulky generator is needed to satisfy the peak power demand. Possible solutions to this problem is to directly design an embedded generator inside the more electric engine (MEE) in Fig 1.2 based on the segmented machine .

FIGURE 1.2: Longitudinal Cross-section More Electric Engine with embedded segmented machine [8]

By breaking down a whole annular stator into circumferentially distributed segments, with each segment individually sized according to the available space and the required electrical load, the available space in the gas turbine engine can be utilized more efficiently. Moreover, each stator can be individually removed or replaced, facilitating maintenance. An example of a segment machine is given in Figure 1.3.

FIGURE 1.3: Transverse cross-section through a segment electrical machine [9]

1.2 Research Motivations

Several main technical obstacles appear when designing a controller for this type of segmented machine. Initially, there is no detailed parametric analytical model for this type of segmented machine. Instead, it is typically derived numerically based on a commercial software JMAG-RT in Figure 1.4 [10]. Since it is based on Finite Element Model results, it will not sacrifice accuracy to achieve this calculation speed. The JMAG-RT plant model can be used in system level simulations to provide a more accurate representation of the electromagnetic device. The JMAG-RT block can be used in Hardware in the Loop Simulations (HILS) which connect a physical Electronic Control Unit (ECU) to a virtual motor.

To overcome these engineering difficulties, a promising advanced control technique called reinforcement learning (RL) [11] [12], has been investigated to find the optimal control input or action by observing the reward or penalty from the operational environment. It can handle the optimal control of nonlinear discrete time systems in [13] [14] and [15] without knowledge of system dynamics. Its integral form, called integral reinforcement learning (IRL) has been proposed in [16] to deal with optimal control of continuous time systems. The block diagram of applying IRL in control of power converter system is depicted in Figure 1.5.

The integral reinforcement learning is generally configured into an actor-critic

FIGURE 1.4: JMAG-RT numerically based model [10]

FIGURE 1.5: IRL control for a numerical plant model

structure [17]. It updates the system behavior through a recursive two-step iteration process namely, policy evaluation and policy improvement. At the policy evaluation step, the IRL controller parametrically approximates the value function according to the current control law. During the policy evaluation step, it generates the new control policy using the gradient of the value function estimated in the previous policy evaluation step.

1.3 Contributions of This Work

The main objective this work is to experimentally apply this IRL strategy in a real-time power converter system and make relevant technical improvement.

Some detailed contributions are listed below.

1. An off-policy integral reinforcement learning control has been applied and experimentally validated for H_∞ , input constrained optimal control problem of an electrical machine. The major advantage of this data-driven control strategy is that the proposed controller can learn the optimal control law online without prior knowledge of the actual machine system coefficients.
2. Existing IRL based H_∞ control algorithm requires the update of disturbance policy at each iteration due to the fact that the standard H_∞ control covers all disturbance (i.e., the worst-case disturbance). However, the pattern or characteristic of the disturbance may be available and the main goal is to develop an effective and computationally inexpensive method to address given disturbance.
3. Most IRL works are developed based on policy iteration where an initial stable controller is needed to start the whole iteration process. To this end, value iteration based IRL is investigated without the prerequisite requirement of having an initial stable control law which results in a model-free initial condition. The convergence modes of value iteration based IRL is fully analyzed. A proposed value iteration based off-policy IRL is applied to a nonlinear quadratic regulation problem.

1.4 Outline of This Thesis

The structure of this report is organized as follows. Chapter 1 starts with a general description of an engineering problem. Some technical obstacles are pointed out. Motivations and major contributions for this thesis are given.

Chapter 2 gives the theoretical background of using IRL control to solve a nonlinear optimal regulation problem. Moreover, the IRL for regulation problems is further extended to solve a nonlinear optimal tracking problem.

Chapter 3 applies policy iteration based IRL to H_∞ control of an electrical machine speed control problem. The 2kW electrical machine serves an application representative of an segmented machine. The major improvement of this data-driven control strategy is that it can greatly enhance parameter robustness of PMSM drives since it requires no prior knowledge of actual machine parameters to design the optimal control policy. Moreover, the proposed method learns the optimal control gains on-line based on measurements from sensors. Some technical limitations are also observed.

Chapter 4 proposes a modified IRL method for intermediate H_∞ control problem with no disturbance update. The main goal is to develop a computationally efficient algorithm to achieve disturbance attenuation for the given time dependent disturbance instead of all disturbance. Compared with the conventional off-policy IRL for standard H_∞ control, the proposed method does not need disturbance update while solving the intermediate H_∞ control problem. The number of unknown variables and the number of neural network during learning process are further reduced and hence is more computationally efficient.

Chapter 5 proposes a value iteration based IRL control to overcome the limitations in policy iteration based IRL where the knowledge of initial stable control law is required as a prerequisite. It has been proved that the selection of the initial value function V^0 as well as the sample time T_s will result in different modes of convergence. Moreover, a VI based off-policy IRL algorithm is proposed to solve the optimal control problems with completely unknown system dynamics, and its convergence has been proved to be equivalent to the modes observed in VI based on-policy IRL algorithm with small T_s . Extensive numerical simulations are provided to verify the feasibility and effectiveness of the proposed VI based off-policy IRL algorithm.

Chapter 6 presents some on-going and further works and Chapter 7 draws the conclusions.

Chapter 2

Theoretical Background of Integral Reinforcement Learning

In this chapter, the theoretical background of IRL is briefly introduced. It is essentially a reinforcement learning scheme that provides online synthesis of optimal control for a class of nonlinear systems with unknown dynamics. IRL can be implemented by making use of two function approximation structures, in an Actor-Critic interconnection. In this continuous-time formulation, the result is a hybrid control structure, which involves a continuous-time controller and a supervisory adaptation structure that operates based on data sampled from the plant and from the continuous-time performance dynamics.

2.1 Chapter Background

The research of integral reinforcement learning begins with the development of an adaptive controller which solves the LQR problem [16]. The online procedure provides the solution of the algebraic Riccati equation (ARE) underlying the LQR problem while renouncing the requirement of exact knowledge on the drift term of the controlled system, while only using discrete measurements of the system's states and performance. It becomes then interesting to develop a more powerful IRL technique, namely off-policy IRL [18] for a completely unknown linear system.

Theoretical development for nonlinear optimal control becomes a bit more complicated since the equation underlying the optimal control problem is the Hamilton-Jacobi-Bellman (HJB) equation, a nonlinear partial differential equation which is in general impossible to be solved analytically and most often does not have a smooth solution. The new online data-based approach to adaptive optimal control is extended to provide a local approximate optimal control solution for the case of nonlinear systems [19].

The convergence guarantee of the online algorithm is given under the realistic assumption that the two function approximators involved in the online policy iteration procedure, namely actor and critic, do not provide perfect representations for the nonlinear control and cost functions. However, the approximation error will approach zero as long as the size of the neural network is large enough based on Weierstrass higher order approximation theorem [20] [21].

2.2 Theoretical background

Consider the nonlinear quadratic regulation problem for a general input affine time-invariant nonlinear system in equation (2.1). The type of system is general enough to cover the dynamic of most power converter systems including electrical machine drives and DC-DC converters.

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t) \quad (2.1)$$

with the system state $x(t) \in \mathbf{R}^n$, the control variable $u(t) \in \mathbf{R}^m$, the system dynamics $f(x(t)) : \mathbf{R}^n \rightarrow \mathbf{R}^n$ and the input dynamics $g(x(t)) : \mathbf{R}^n \rightarrow \mathbf{R}^{n \times m}$, which satisfies $f(0) = 0$. In addition, $f(x(t))$, $g(x(t))$ are Lipschitz continuous functions on a compact set $\Omega \in \mathbf{R}^n$, which contains the origin such that $x = 0$ is an equilibrium point of the system [22]. Assume that system (2.1) is controllable with at least one continuous control law u which can asymptotically stabilize the system at the origin.

The infinite horizon integral cost associated with the control input is defined as

$$V = \int_t^{\infty} r(x(\tau), u(\tau)) dt = \int_t^{\infty} (x^T Q x + u^T R u) dt \quad (2.2)$$

The infinitesimal version of (2.2) is

$$0 = x^T Q x + u^T R u + (V_x)^T (f(x) + g(x)u(x)) \quad (2.3)$$

Defining the Hamiltonian of the problem

$$H(x, u, V_x) = r + (V_x)^T (f(x) + g(x)u(x)) \quad (2.4)$$

The optimal cost function V_x^* satisfies the HJB equation (2.4).

$$0 = \min_u [H(x, u, V_x^*)] \quad (2.5)$$

This Hamiltonian corresponds to the infinite horizon optimal control problem, must be identically zero. The optimal control law for the given problem is in equation (2.6)

$$u^*(x) = -\frac{1}{2} R^{-1} g^T(x) V_x^* \quad (2.6)$$

Substituting the optimal control policy into the Hamiltonian of the problem in equation, we end up with Hamilton-Jacobi-Bellman (HJB) in equation (2.7)

$$0 = x^T Q x + (V_x^*)^T f(x) - \frac{1}{4} (V_x^*)^T g(x) R^{-1} g^T(x) V_x^* \quad (2.7)$$

The original HJB is hard to solve since the HJB equation is nonlinear in V^* . However by using the Bellman dynamic programming concept with the value function $V(t)$ defined in (2.8), it can be rewritten as a functional $V(x(t))$. Based on the linear partial derivative equation (2.4) and updated control law in (2.5), the HJB equation can be solved iteratively based on reinforcement learning schemes.

The online algorithms presented herein are reinforcement learning schemes which provide online synthesis of optimal control for a class of nonlinear systems with unknown drift term. The results are direct adaptive control algorithms which converge to the optimal control solution without using an explicit, a priori obtained, model

of the drift dynamics of the system. The online algorithms can be implemented while making use of two function approximation structures, in an Actor-Critic interconnection. In this continuous-time formulation the result is a hybrid control structure, which involves a continuous-time controller and a supervisory adaptation structure which operates based on data sampled from the plant and from the continuous-time performance dynamics.

2.3 Policy Iteration based Reinforcement Learning

It has been proven in [22] that iterating on equations (2.4) and (2.6), conditioned at an initial stable control law, the subsequent control law will be stable and will finally converge to the solution of HJB equation. Considering **Algorithm 2.1** reinforcement learning based on (2.8) and (2.9), it provides a solution for HJB equation. However the information regarding $f(x)$ and $g(x)$ is still needed.

Algorithm 2.1: Policy Iteration based on Reinforcement Learning [22]

Step 1. Under an initial stable control law u^i , solve the equation for V^i .

$$0 = x^T Q x + (u^i)^T R u^i + (V_x^i)^T (f(x) + g(x)u^i) \quad (2.8)$$

Step 2. Update the control policy using equation (2.9) and apply u^{i+1} to the system.

$$u^{i+1} = -\frac{1}{2} R^{-1} g^T(x) V_x^i \quad (2.9)$$

Step 3. Go back to Step 1 until convergence.

A flowchart of **Algorithm 2.1** is depicted here in Figure 2.1 based on a two-step iteration, namely policy evaluation and policy improvement.

The equation (2.8) is considered as a bad Bellman equation since the information of $f(x)$ is required. By integrating the equation (2.8) from t to $t + T$, a good Bellman equation is formulated in equation (2.10). It is equivalent to equation (2.8) however it does not require the system information of $g(x)$. The Integral Reinforcement Learning is shown in **Algorithm 2.2**, which only requires partial information of the system dynamics, namely $g(x)$.

FIGURE 2.1: Actor/critic structure for solving the optimal control problem in algorithm 2.1

$$V^i(x(t)) = V^i(x(t+T)) + \int_t^{t+T} (x^T Q x + (u^i)^T R u^i(\tau)) d\tau \quad (2.10)$$

Algorithm 2.2: Policy Iteration based Integral Reinforcement Learning for partial unknown system [19]

Step 1. Let $i=0$, under a stable initial control policy $u^i(x(t))$, solve the equation (2.11) for the value of $V^i(x(t))$ where $u^i(x(t))$, $V^i(x(t))$ denotes the control policy and value function to be updated at i -th iteration.

$$V^i(x(t)) = V^i(x(t+T)) + \int_t^{t+T} x^T Q x + (u^i)^T R u^i(\tau) d\tau \quad (2.11)$$

Step 2. Updating the control policy using

$$u^{i+1}(x(t)) = \arg \min_u H(x, u, V_x) = -\frac{1}{2} R^{-1} g^T(x) V_x^i \quad (2.12)$$

Step 3. Let $i = i + 1$ and Go back to 1 until convergence

The Integral Reinforcement Learning in **Algorithm 2.2** only allows for partially unknown dynamics. Here the off-policy Integral Reinforcement Learning with completely unknown dynamics is introduced in [23]-[24].

Let the control policy u^i , then the original system dynamic (2.1) becomes (2.13) where u_0 is a fixed initial control law and applied to the system until convergence.

$$\dot{x} = f(x) + g(x)u^i + g(x)(u_0 - u^i) \quad (2.13)$$

Then differentiating V along the system dynamic in equation (2.13).

$$\dot{V} = V_x^T (f(x) + g(x)u^i + g(x)(u_0 - u^i)) \quad (2.14)$$

Substituting the policy evaluation and policy improvement equations (2.8) and (2.9) into equation (2.14), obtains (2.15)

$$\dot{V} = -(x^T Q x + (u^i)^T R u^i - 2u^{i+1} R (u_0 - u^i)) \quad (2.15)$$

Integrating $\dot{V}$ in equation (2.15) from t to $t+T$, the integral version of off-policy bellman equation is obtained.

$$V^i(x(t+T)) - V^i(x(t)) = - \int_t^{t+T} (x^T Q x + (u^i)^T R u^i) d\tau + \int_t^{t+T} -2u^{i+1} R (u_0 - u^i) d\tau \quad (2.16)$$

Algorithm 2.3: Policy Iteration based Off-policy Integral Reinforcement Learning algorithm [25],[26],[27]

Step 1: Apply a pre-chosen control policy u_0 to the system continuously and collect required system information about the state x , control input u from t to $t+T$.

Step 2: With the collected data of $x(t)$, $u(t)$ as well as the optimal control policy, disturbance policy u^i at the i -th iteration, solve the following Bellman equation (2.17) for the cost function V^i , optimal control policy u^{i+1} and disturbance policy d^{i+1} at the $(i+1)$ -th iteration.

$$V^i(x(t+T)) - V^i(x(t)) = - \int_t^{t+T} (x^T Q x + (u^i)^T R u^i) d\tau - \int_t^{t+T} 2(u^{i+1})^T R (u_0 - u^i) d\tau \quad (2.17)$$

Step 3: Let $i = i + 1$ and go back to Step 2 until convergence

---

So far in **Algorithm 2.1 to 2.3**, the reinforcement learning algorithm is introduced to solve a nonlinear quadratic regulation problem. The extension from nonlinear quadratic regulation problem to nonlinear quadratic tracking problem are

also made in [28]-[29], which is more attractive in industry applications.

2.4 Tracking a Constant Reference

Initially, optimal control deals with a regulation problem where the system is supposed to be asymptotically stable at origin. However, in practically applications, we always desire to control some system state to track a piece-wise constant state reference.

Consider nonlinear quadratic tracking problem for a time-invariant the same nonlinear system described equation (2.18).

$$\dot{x}(t) = f(x) + g(x)u(t) \quad (2.18)$$

with the system state $x(t) \in \mathbf{R}^n$, the control variable $u(t) \in \mathbf{R}^m$, the system dynamics $f(x(t)) : \mathbf{R}^n \rightarrow \mathbf{R}^n$ and the input dynamics $g(x(t)) : \mathbf{R}^n \rightarrow \mathbf{R}^{n \times m}$, which satisfies $f(0) = 0$. In addition, $f(x(t))$, $g(x(t))$ are Lipschitz continuous on a compact set $\Omega \in \mathbf{R}^n$, which contains the origin such that $x = 0$ is an equilibrium point of the system.

The desired trajectory $x_d \in \mathbf{R}^1$ is a constant reference which satisfies

$$\dot{x}_d(t) = 0 \quad (2.19)$$

The tracking error $e_d(t)$ is defined as

$$e_d(t) = x - x_d \quad (2.20)$$

Our control objective for this tracking problem is to design u such that the the system

2.4.1 Existing Formulation

In the existing standard formulation of the optimal tracking control problem, if the dynamics of the system is known and the inverse of the input dynamics $g^{-1}(x_d(t))$ exists [28], the steady-state part of the control input is

$$u_d(t) = g^{-1}(x_d(t))(-f(x_d(t))) \quad (2.21)$$

On the other hand, the feedback part of the control is designed to stabilize the tracking error dynamics in an optimal manner by minimizing the following performance function.

$$V(e_d(t)) = \int_t^\infty (e_d(\tau)^T Q e_d(\tau) + u_e^T R u_e) d\tau \quad (2.22)$$

Then, the final control law $u = u_e(t) + u_d(t)$ which consists of two parts, the steady state part and the feedback part.

However, the exact knowledge of system dynamics is required to find the steady-state part of the control input given by (2.21), which cancels the usefulness of IRL technique.

2.4.2 Direct Formulation of Optimal Tracking Problem

Define the augment system state as $X = [x(t) \ x_d(t)]$. Putting (2.18) and (2.19) yields the augmented system

$$\dot{X}(t) = F(X) + G(X)u(t) \quad (2.23)$$

where $u(t) = u(X(t))$ and

$$F(X) = \begin{bmatrix} f(x(t)) \\ 0 \end{bmatrix}, G(X) = \begin{bmatrix} g(x(t)) \\ 0 \end{bmatrix}. \quad (2.24)$$

The infinite horizon integral cost associated with the control input u is defined as

$$V(t) = \int_t^{\infty} r(x(\tau), u(\tau)) dt = \int_t^{\infty} ((x - x_d)^T Q(x - x_d) + u^T R u) d\tau \quad (2.25)$$

The original value function $V(t)$ needs to be modified by introducing the discount factor α . The use of α is to make the performance function bounded since the feed forward part of the control input does not converge to zero in general.

$$V(t) = \int_t^{\infty} r(x(\tau), u(\tau)) dt = \int_t^{\infty} e^{-\alpha(\tau-t)} ((x - x_d)^T Q(x - x_d) + u^T R u) d\tau \quad (2.26)$$

The discount factor α and the weight matrix Q in equation (2.26) should be appropriately chosen to make the error state go to a very small region near zero. However, this formulation of nonlinear optimal tracking cannot guarantee a zero steady-state tracking error. It can be briefly explained as follows.

Assume the nonlinear optimal control law u^* achieve zero steady-state tracking error, therefore the first part of integrand $(x - x_d)^T Q(x - x_d) = 0$. However the control input u is normally not zero since the control input u contains a feed forward term that depends on the reference trajectory [30]. Therefore, as $t \rightarrow \infty$, $V(t)$ is not 0, the first part $(x - x_d)^T Q(x - x_d) = 0$ is not 0.

2.4.3 Output Regulation Theory

Consider an uncertain nonlinear plant as follows

$$\dot{x}(t) = f(x) + g(x)u(t) \quad (2.27)$$

with the same desired state reference

$$\dot{x}_d(t) = 0 \quad (2.28)$$

Assume that a linear state feedback controller of this form

$$u = K_1 x + K_2 x_d \quad (2.29)$$

where K_1, K_2 are the control gains of proper dimensions.

First, the nonlinear regulation equations are solved for $\mathbf{x}(x_d)$ and $\mathbf{u}(x_d)$ such that $\mathbf{x}(0) = 0$ and $\mathbf{u}(0) = 0$.

$$\begin{aligned} 0 &= f(\mathbf{x}(x_d)) + g(\mathbf{x}(x_d))\mathbf{u}(x_d) \\ 0 &= \mathbf{x}(x_d) - x_d \end{aligned} \quad (2.30)$$

Then the control law $u = \mathbf{u}(x_d) + K_1(x - \mathbf{x}(x_d))$ solves the output regulation problem if $\frac{\partial(f(x)+g(x)K_1x)}{\partial x}(0,0)$ has negative real parts.

A more systematic solution to the tracking results reviewed has been obtained by integration of model-based output regulation theory and data-based integral reinforcement learning in [31] and [32].

2.4.4 Robust Output Regulation Theory

Consider an uncertain nonlinear plant as follows

$$\dot{x}(t) = f(x) + g(x)u(t) \quad (2.31)$$

with the same desired state reference

$$\dot{x}_d(t) = 0 \quad (2.32)$$

Assume that a linear state feedback controller of this form

$$\begin{aligned} u &= K_1 x + K_2 z \\ \dot{z} &= e = x - x_d \end{aligned} \quad (2.33)$$

where K_1, K_2 are the control gains of proper dimensions. z is the compensation state vector based on internal model principle. Letting $x_c = \text{col}(x, z)$, the resulting closed-loop can be written as

$$\dot{x}_c(t) = f_c(x_c(t), x_d) \quad (2.34)$$

$$\text{where } f_c(x_c, x_d) = \begin{bmatrix} f(x) + g(x)(K_1x + K_2z) \\ x - x_d \end{bmatrix}$$

If $\frac{\partial f_c}{\partial x_c}(0, 0)$ has negative real part, there will always exist a sufficiently smooth function $\mathbf{x}_c(x_d)$ with $\mathbf{x}_c(0) = 0$ satisfying the nonlinear regulator equations in (2.35) based on robust nonlinear output regulation theory [33].

$$\begin{aligned} 0 &= f_c(\mathbf{x}_c(x_d), x_d) \\ 0 &= \mathbf{x}(x_d) - x_d \end{aligned} \quad (2.35)$$

Since the controller (2.33) incorporates the p-copy internal model of the reference, this controller also solves the robust output regulation problem for the nonlinear system.

Therefore, we can design an optimal $u^* = K_1x + K_2z$ to minimize $V(t)$ in (2.36). Then $\frac{\partial f_c}{\partial x_c}(0, 0)$ will have negative real part.

$$V(t) = \int_t^\infty r(x(\tau), u(\tau)) d\tau = \int_t^\infty ((x_c^T Q(x_c) + u^T R u) d\tau \quad (2.36)$$

2.5 Conclusion

In this chapter, theoretical background of IRL is briefly presented for an optimal regulation problem of an input affine nonlinear system. On-policy IRL is introduced to solve the optimal control without knowledge of system internal dynamic $f(x)$ while off-policy IRL is proposed to solve the same problem without knowledge of system internal dynamic $f(x)$ as well as system input matrix $g(x)$. It is also shown that a tracking problem can be transformed into regulation problem based on nonlinear robust output regulation theory.

Chapter 3

Integral Reinforcement Learning with an Application to H_∞ , Input Constrained Control of a Permanent Magnet Synchronous Machine

In this chapter, an IRL control scheme which combines the attractive features of adaptive control and optimal control, is applied to an input constrained H_∞ tracking problem for permanent magnet synchronous motor (PMSM) drives. Its data-driven characteristic makes it less sensitive to unknown motor parameter uncertainties and external disturbances.

The IRL provides a data-based solution to the Hamilton-Jacobi-Isaac (HJI) equation without using actual information of the system model parameters. This algorithm is based on policy iteration and implemented using an actor/critic structure, which makes use of neural networks to parametrically represent the value function and control law, tunes the controller parameters based on observed system outputs. The proposed control solution has been experimentally validated on a 2-KW PMSM test rig.

3.1 Chapter Background

In recent years, permanent magnet synchronous machines (PMSM) are extensively used in home appliances as well as electric vehicles, wind generation system and industrial rotors, owing to their high efficiency, low maintenance cost and high power density.

However, high performance robust control of PMSM drives is still a challenging task because it is a nonlinear multi-variable system with on-line variations of system parameters depending on temperature variations, inductance saturation and etc. Therefore, a considerable amount of research work has been carried out on the design of high performance controller that provides fast response, high accuracy and robustness against the motor parameter uncertainties and load torque variations.

Traditionally, the proportional-integral-derivative (PID) controller is widely adopted to control the PMSM system in industrial applications owing to its simplicity and clear design guideline. However, a big problem of the traditional PID controller is its sensitivity to system uncertainty. The control performance of the conventional PID method can be seriously degraded under parameter variations. The conventional PID controller also lacks the auto-tuning ability considering some long-term degradation coefficients of the machine. Various multi-variable strategies such as back-stepping control [34], model predictive control (MPC) [35], adaptive control [36] [37] [38], have been proposed for high performance control of PMSM. The back-stepping control and MPC require the true values of the inductance, flux linkage of permanent magnet as well as inertia in order to ensure global closed-loop stability. Hence, those approaches will suffer from system parameter uncertainties. The adaptive control technique can still ensure closed-loop stability under parameter uncertainty [39]. Unfortunately, adaptive control is not optimal in general cases. While high-performance operation of machines and power converters requires fast dynamic response and accurate regulation of controlled variables, the only concern of adaptive control is the stability considering parameter uncertainty. The transient closed-loop performance such as less overshoot, less fluctuation cannot be guaranteed. On the other hand, an H_∞ tracking strategy has been widely used for its robustness in counteracting parameter uncertainty and external disturbances. The main point of the H_∞ control is to synthesize a feedback law that renders the closed-loop system to satisfy a prescribed H_∞ -norm constraint, representing desired

stability or tracking requirements. Existing solutions to the H_∞ tracking problem require solving Hamilton-Jacobi-Issac equations with complete knowledge of the system dynamics. A promising alternative to deal with parameter uncertainties is to tune the H_∞ controller based on interactions between the plant and environment. Motivated by this idea, an integral reinforcement learning (IRL) method is adopted to learn the Hamilton-Jacobi-Issac (HJI) solution online without using actual knowledge of the system dynamics.

The integral reinforcement learning is generally configured into an actor-critic structure. It updates the system behavior through a recursive two-step iteration [40] [18], namely policy evaluation and policy improvement. At the policy evaluation step, the IRL controller parametrically approximates the value function according to the current control law. During the policy improvement step, it gives the new control policy using the gradient of value function estimated in the previous policy evaluation step. Practical implementation of integral reinforcement learning controller can be found in the optimal-load-frequency control in power systems [41], a hybrid system of a jumping robot [42] and controlling arterial oxygen saturation [43]. Therefore, it is motivated to apply this modern control strategy for high performance drive of PMSM.

Compared with existing literature, the main contributions of this chapter are listed below.

1. Neural network techniques are applied to approximate the perturbation terms such as stator inductance and resistance. Data-driven Off-policy integral reinforcement learning is applied to solve HJI without information of system coefficients.
2. Design initial stabilizing control law using the Routh-Hurwitz stability criterion.
3. The IRL based controller is experimentally validated.

The rest of this chapter is organized as follows. Section 3.2 provides the system model description for PMSM. Section 3.3 formulates the tracking problem for PMSM so that the controller can be designed based on H_∞ technique. Section 3.4 gives introduction of the off-policy reinforcement learning algorithm. The proposed controller is implemented in Section 3.5 and experimentally validated and

compared with a classical PID controller in a PMSM test rig in Section 3.6 . Section 3.7 draws the conclusion.

3.2 Electrical Machine System Model Description

The mathematical model of PMSM in rotating d-q reference frame is generally described as [44].

$$\begin{aligned} \dot{i}_d &= -\frac{R_s}{L_d}i_d + \frac{1}{L_d}u_d + P\omega\frac{L_q}{L_d}i_q \\ \dot{i}_q &= -\frac{R_s}{L_q}i_q - \frac{\lambda_m}{L_q}P\omega + \frac{1}{L_q}u_q - P\omega\frac{L_d}{L_q}i_d \\ \dot{\omega} &= \frac{3}{2J}P(\lambda_m i_q + (L_d - L_q)i_d i_q) - \frac{1}{J}T_L \end{aligned} \quad (3.1)$$

where ω is the mechanical rotor speed, i_d and i_q are the d-axis and q-axis stator currents, u_d and u_q are the d-axis and q-axis voltage inputs, L_d, L_q are the stator d-axis and q-axis inductance, R_s is the stator resistance, λ_m is the permanent magnet flux linkage, J is the moment of inertia, P is the number of pole pairs and T_L is the load torque. The nominal values of those coefficients are given in TABLE 3.1.

The following assumptions are made:

1. All the electrical and mechanical coefficients are bounded where the upper and lower bounds of these parameters are known such that there exists known positive constants $R_{\min}, R_{\max}, L_{\min}, L_{\max}, J_{\min}, J_{\max}$ that make the inequalities in (3.2) hold.

$$\begin{aligned} 0 &< R_{\min} \leq R_s \leq R_{\max}; \\ 0 &< L_{\min} \leq L_d, L_q \leq L_{\max}; \\ 0 &< J_{\min} \leq J \leq J_{\max} \end{aligned} \quad (3.2)$$

Where $R_{\max}, L_{\max}, J_{\max}$ are considered as 150% of their nominal values respectively and $R_{\min}, L_{\min}, J_{\min}$ are considered as 50% of their nominal values.

2. The speed and position of PMSM are accessible for measurement. Therefore, the machine in the a-b-c (stationary) reference frame can be transformed into d-q-0 synchronously rotating reference frame as expressed in (3.1).

3. For a given PMSM, which is constrained by the voltage source, protect facilities, operating environments and so on, current and speed must be bounded in a certain range. The following assumptions are made, which are also adopted in [45] [46] for robust control of PMSM.

$$\begin{aligned} i_{dmin} &\leq i_d \leq i_{dmax}; \\ i_{qmin} &\leq i_q \leq i_{qmax}; \\ \omega_{min} &\leq \omega \leq \omega_{max}; \end{aligned} \tag{3.3}$$

where $i_{dmin}, i_{dmax}, i_{qmin}, i_{qmax}, \omega_{min}, \omega_{max}$ are the lower and upper bounds of i_d, i_q and ω respectively. presented in TABLE 3.1.

TABLE 3.1: PMSM TEST RIG PARAMETERS

Symbol	Quantity	Value
P_{rated}	Rated power	2 kW
ω	Rated mechanical speed	1000 rpm
P	Number of pole pairs	3
L_d	Direct-axis inductance	2.00 mH
L_q	Quadrature-axis inductance	2.70 mH
λ_m	Flux linkage of PM	0.18 Wb
R_s	Stator resistance	0.2 Ω
J	Inertial of the shaft	0.006 kgm^2
i_{dmax}, i_{dmin}	max and min value of i_d	5 A, -5 A
i_{dmax}, i_{qmin}	max and min value of i_q	20 A, -20 A
ω_{max}	maximum rotating speed	157 rad/s
ω_{min}	minimum rotating speed	-157 rad/s

The control objective of PMSM drive is to make the actual rotation speed track desired reference regardless of system coefficients drifting and load disturbance changes. The following typical closed-loop performance indices are defined:

1. Settling time (T_s): time elapsed from speed reference step time (e.g., ω_{ref}) until output (e.g., ω) tracks the reference with less than 3 % of error;
2. Overshoot due to step load change (M_p): maximum peak value of the speed over/under the desired speed when a step load change T_L is applied in machine shaft.

3.3 H_∞ Tracking Controller Design

In this section, the tracking problem is formulated so that the controller can be designed using H_∞ technique. Since an H_∞ solution to the above system in (3.1) does not provide a tracking performance, an additional state variable q defined in (3.4) is introduced whose derivative is the difference between the speed reference ω_{ref} and actual speed ω , shown below.

$$\dot{q} = \omega_{ref} - \omega \quad (3.4)$$

The introduction of the integral state q is necessary to achieve a zero steady-state tracking error based internal model principle.

Write (3.1),(3.4) as follows.

$$\dot{x} = f(x) + g(x)u + k(x)d \quad (3.5)$$

where $x = [i_d \ i_q \ \omega \ q]^T$ denotes the state of the system, $u = [u_d \ u_q]^T$ represents the input vector, $d = [T_L \ \omega_{ref}]^T$ is the load torque and speed reference. $f(x)$, $g(x)$ and $k(x)$ are the machine internal dynamic, input dynamic and disturbance dynamic, respectively, where

$$f(x) = \begin{bmatrix} \frac{L_q}{L_d} P \omega i_q - \frac{R_s}{L_d} i_d \\ \frac{-\lambda_m}{L_q} P \omega - \frac{L_d}{L_q} P \omega i_d - \frac{R_s}{L_q} i_q \\ \frac{3P}{2J} [\lambda_m i_q + (L_d - L_q) i_d i_q] \\ -\omega \end{bmatrix} \quad (3.6)$$

$$g(x) = \begin{bmatrix} \frac{1}{L_d} & 0 \\ 0 & \frac{1}{L_q} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, k(x) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -\frac{1}{J} & 0 \\ 0 & 1 \end{bmatrix} \quad (3.7)$$

If the system dynamic in (3.5)-(3.7) with $d = [0, 0]^T$ can be stabilized with the control law u , the speed tracking performance can be guaranteed with the same state-feedback control law. To design a stabilizing controller, define the H_∞ performance index.

$$V(t) = \int_t^\infty (x^T Q x + W(u) - \gamma^2 d^T d) d\tau. \quad (3.8)$$

The infinite horizon cost function $V(t)$ is finite as long as the control input u is stable and the disturbance d will approach zero when time goes to ∞ . Q is a positive semidefinite matrix and $W(u)$ is a positive definite function, which determine the quadratic cost on state vector x and input vector u , respectively.

The i th diagonal element of Q , which is $q_{i,i}$, determines the direct weighting on the i th state. For unconstrained control inputs, a common choice for $W(u)$ is $W(u) = u^T R u$, where R is a real symmetric and positive matrix. The above formulation considers the reference speed and load torque as bounded disturbances of the system. However there exists no way to find the smallest value of γ for the nonlinear system. Therefore, a predetermined γ is chosen large enough to satisfy the bounded L_2 gain [29].

For practical applications, the control input u to the machine is fed by a voltage source inverter. Therefore, the control input u should be bounded by the input voltage limit, namely $|u_d| \leq \lambda, |u_q| \leq \lambda$, where λ is the input voltage limit. To this end, non-quadratic functional $W(u)$ is introduced in (3.9) [47] to capture constraints on inputs.

$$W(u) = 2 \int_0^u \lambda \tanh^{-1}(v/\lambda)^T R dv \quad (3.9)$$

where $\tanh^{-1}$ is the inverse hyperbolic tangent function, R is a positive matrix.

Definition 3.1. The H_∞ control problem is to find a control policy u , such that the following two conditions hold.

- 1) The closed-loop system (3.5) satisfies the following attenuation condition

$$\int_t^\infty (x^T Q x + W(u)) d\tau \leq \gamma^2 \int_t^\infty (d^T d) d\tau.$$

- 2) The closed-loop system dynamics with $d = [0, 0]^T$ is locally asymptotically stable at the origin.

Our **nonlinear optimal control problem** is to find the optimal control policy u^* to minimize the cost $J = V(0)$ subject to the system dynamics (3.5)-(3.7).

To solve the optimal control problem, the following Hamiltonian is defined, where V_x is the derivative of V over $x(t)$,

$$H(u, d) = x^T Q x + W(u) - \gamma^2 d^T d + V_x^T (f + gu + kd). \quad (3.10)$$

Clearly $H(u, d) = 0$ when $u = u^*$ and $d = d^*$. The optimal control and disturbance satisfy the stationary conditions of $\frac{\partial H}{\partial u^*} = \frac{\partial H}{\partial d^*} = 0$, which leads to

$$\begin{aligned} u^* &= -\lambda \tanh\left(\frac{1}{2\lambda} R^{-1} g^T V_x^*\right) \\ d^* &= \frac{1}{2\gamma^2} k^T V_x^* \end{aligned} \quad (3.11)$$

Substitutes equation (3.11) into (3.10) will give the modified HJT equation, which is generally hard to solve. A classical approach to solve the modified HJI equation of optimal control problem is using the policy iteration in **Algorithm 3.1**.

Algorithm 3.1: H_∞ , input constrained control with full information of the system dynamics [48]

Step 1: Apply control and disturbance law u^i, d^i at i -th iteration to the system, find V^i using (3.12).

$$\begin{aligned} &x^T Q x + (V_x^i)^T (f + gu^i + kd^i) \\ &+ 2 \int_0^{u^i} (\lambda \tanh^{-1}(v/\lambda)) - \gamma^2 (d^i)^T d^i = 0 \end{aligned} \quad (3.12)$$

Step 2 Update the control law and disturbance law in (3.13)

$$\begin{aligned} u^{i+1} &= -\lambda \tanh\left(\frac{1}{2\lambda} R^{-1} g^T V_x^i\right) \\ d^{i+1} &= \frac{1}{2\gamma^2} k^T V_x^i. \end{aligned} \quad (3.13)$$

Step 3 Let $i = i + 1$ and Go back to **Step 1** until convergence.

3.4 An Optimal Adaptive Controller Synthesis Based on Off-policy Integral Reinforcement Learning

While **Algorithm 3.1** needs the actual information of the system dynamic such as $f(x), g(x), k(x)$, a more model-free method called off-policy integral reinforcement learning in Algorithm 3.2 is given in [29] to implement **Algorithm 3.1** online which uses the state measurements and obtain enhanced system performance.

Algorithm 3.2: Off-policy integral reinforcement learning algorithm for H_∞ , input constrained control [49] [50]

Step 1: Apply a pre-chosen control policy u_0 to the system continuously and collect required system information about the state x , control input u from t to $t+T$.

Step 2: With the collected data of $x(t), u(t)$ as well as the optimal control policy u^i , disturbance policy d^i at the i -th iteration, solve the following Bellman equation (3.14) for the cost function V^i , optimal control policy u^{i+1} and disturbance policy d^{i+1} at the $(i+1)$ -th iteration.

$$\begin{aligned}
 & V^i(x(t+T)) - V^i(x(t)) \\
 &= - \int_t^{t+T} (x^T Q x + W(u^i) - \gamma^2 (d^i)^T d^i) d\tau \\
 &\quad - \int_t^{t+T} 2\lambda \tanh^{-1}\left(-\frac{u^{i+1}}{\lambda}\right)^T R(u - u^i) d\tau \\
 &\quad + \int_t^{t+T} 2(d^{i+1})^T \gamma^2 (d - d^i) d\tau
 \end{aligned} \tag{3.14}$$

Step 3: Let $i = i + 1$ and go back to **Step 2** until convergence

---

Based on the well-known high-order Weierstrass approximation theorem, any smooth continuous function can be represented by a linearly independent basis function set. Therefore, three neural networks (NNs) in (3.15), i.e. the critic NN, the controller NN and the disturbance NN, are used to approximate the solution $V^i(x), u^{i+1}, d^{i+1}$ of the Bellman equation (3.14).

$$\begin{aligned}
 V^i(x) &= W_1^T \varphi_1(x) + \varepsilon_1 \\
 u^{i+1} &= -\lambda \tanh\left(\frac{W_2^T \varphi_2(x)}{\lambda}\right) + \varepsilon_2 \\
 d^{i+1} &= W_3^T \varphi_3(x) + \varepsilon_3
 \end{aligned} \tag{3.15}$$

where $\varphi_1 \in R^{N_1 \times 1}, \varphi_2 \in R^{N_2 \times 1}, \varphi_3 \in R^{N_3 \times 1}$ provide the base function vectors. $W_1 \in R^{N_1 \times 1}, W_2 \in R^{N_2 \times 2}, W_3 \in R^{N_3 \times 2}$ N_1, N_2, N_3 are the sizes for each NN, $\varepsilon_1, \varepsilon_2, \varepsilon_3$ are the approximation errors using NN. The base function vectors in this chapter as follows.

$$\begin{aligned}\varphi_1(x) &= [x_1^2 \quad x_1x_2 \quad x_1x_3 \quad \dots \quad x_4^2]^T \\ \varphi_2(x) &= [x_1 \quad x_2 \quad x_3 \quad x_4]^T \\ \varphi_3(x) &= [x_1 \quad x_2 \quad x_3 \quad x_4]^T\end{aligned}\tag{3.16}$$

Substituting value function and control law approximation equations (3.15) (3.16) into the off-policy bellman equation (3.14). The following equation (3.17) is derived,

$$\begin{aligned}& (W_1^i)^T \varphi_1(x(t+T)) - (W_1^i)^T \varphi_1(x(t)) = \\ & - \int_t^{t+T} (W(u^i) + x^T Q x - \gamma^2 d^T d) d\tau \\ & \int_t^{t+T} 2\varphi_2^T W_2^{i+1} R(u - u_i) d\tau + \int_t^{t+T} 2\varphi_3^T W_3^{i+1} \gamma^2 (d - d_i) d\tau\end{aligned}\tag{3.17}$$

3.5 Implementation of Off-policy Integral Reinforcement Learning for Machine Control

3.5.1 Determining the initial control using the Routh-Hurwitz stability criterion

For all integral reinforcement learning approach with policy iteration, there exists a strong assumption that the initial control law must be stabilizing. Not much work has been done to show how to choose the initial stabilizing control law for a practical application. However, considering the electrical machine with assumptions stated in Table 3.1, we can determine the following initial stabilizing control law using the Routh-Hurwitz stability criterion.

Reconsider the system dynamics in (3.5-3.7) with $d = [0, 0]^T$ and $u = -\lambda \tanh(\frac{Kx}{\lambda})$ and the parameter bounds in (3) and linearize the system with equilibrium operation point $i_d^e = 0, i_q^e = 10A$ and $\omega^e = 52rad/s$. Then the linearized closed-loop system dynamic with $d=0$ can be written in the form as

$$\dot{x} = A_c x\tag{3.18}$$

where $A_c = A + BK$ with A, B in

$$A = \begin{bmatrix} -\frac{R}{L_d} & \frac{L_q}{L_d}P\omega^e & \frac{L_q}{L_d}Pi_q^e & 0 \\ -\frac{L_d}{L_q}P\omega^e & -\frac{R}{L_q} & \frac{-\lambda_m}{L_q}P & 0 \\ \frac{3P}{2J}(L_d - L_q)i_q^e & \frac{3P}{2J}\lambda_m & 0 & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix} \quad (3.19)$$

$$B = \begin{bmatrix} \frac{1}{L_d} & 0 \\ 0 & \frac{1}{L_q} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (3.20)$$

Based on the system operation region and coefficients bounds in TABLE 3.1, The characteristic equation $|sI - A_c|$ has negative roots based on Routh-Hurwitz stability criterion with the control gain W_2^0

$$W_2^0 = \begin{bmatrix} -5 & 0 & 0 & 0 \\ 0 & -5 & 0 & 12 \end{bmatrix} \quad (3.21)$$

Then, the integral reinforcement learning can be started, observe the system states and enhance system performances via state measurements.

3.5.2 Design of the optimal control problem : Q and R selection

As discussed in previous section, the Q and R matrices in (3.8) and (3.9) are the key design parameters to formulate the value function for the optimal control problem. The Q matrix weights the different elements inside the state vector. As a counterpart, the R matrix weights the control effort. The practical design of the H_∞ controller or LQR controller typically requires some iteration in the Q and R values until the desired performance is achieved [51]. Therefore, the following heuristic rules are proposed.

1. First of all, it is obvious that the optimum control law u^* remains invariant when Q and R are multiplied by a common scalar factor.

2. For design simplicity, the Q and R values out of the main diagonal are set to zero.
3. $q_{1,1}, q_{2,2}$ is the weighting cost on the current i_d, i_q , which have effects on the transient overshoot of machine currents.
4. $q_{3,3}$ is the weighting cost on the machine speed ω . The overshoot in speed tracking trajectory can be tuned by modify $q_{3,3}$. The overshoot of speed trajectory decreases as $q_{3,3}$ increases.
5. $q_{4,4}$ is the key design parameter to achieve speed tracking and disturbance attenuation. Normally a large weight is assigned to $q_{4,4}$.

The Q, R in this chapter are selected based on the heuristic rules above and given in Table 3.2 in Section 3.6.

3.5.3 Procedure of Algorithm 3.2 off-policy integral reinforcement learning

With the stable control law u^0 in previous subsection, the **Algorithm 3.2** can be initialized. The flowchart for implementing **Algorithm 3.2** is illustrated provided as follows in Figure 3.1.

1. Assign an initial stable controller u^0 based on linearized model and Routh-Hurwitz stability criterion .
2. The equation (3.17) can be regarded as a linear-regression problem. The weight $W_1^i, W_2^{i+1}, W_3^{i+1}$, can be solved after sufficient data are collected from the system.
3. Convergence test. Evaluate the norm between the updated estimated W^{i+1} and the previous estimated W^i . If the norm value is smaller than a predefined threshold ϵ , **Algorithm 3.2** will be terminated. Otherwise, go to Step 2 and continue the iteration.

FIGURE 3.1: Flowchart for the implementation of Algorithm 3.2

3.6 Experimental Verification of Integral Reinforcement Learning controller

3.6.1 Drive Setup

In this section, the effectiveness of the proposed IRL control scheme is evaluated through experiments on a prototype PMSM using TMS320F28335 Digital Signal Processor (DSP) with switching frequency 10kHz.

The integral reinforcement learning control scheme is implemented in the DSP with a sampling period of $T = 0.1\text{ms}$ as control diagram in Fig. 3.2 The experimental setup in Fig. 3.3 incorporates the following hardware components : a DC voltage source of 100V, a three-phase inverter and its driving circuits and control board, a brake, a Hengstler incremental encoder and hall effect current sensors and voltage sensors. The rotor position is sensed by the incremental encoder and then it is used to calculate the motor speed by differentiating the position obtained from the encoder. The three-phase currents are then fed to the DSP board through analogue to digital converters and then converted to the actual i_d, i_q for the control purpose.

Therefore, all states namely i_d, i_q, ω, q are accessible and the integral reinforcement learning can be implemented.

FIGURE 3.2: Block diagram of proposed control system for PMSM drive

TABLE 3.2: PARAMETER SETTING FOR THE OPTIMAL CONTROL PROBLEM

Symbol	Quantity	Value
Q	Weight matrix	$\text{diag}([1 \ 1 \ 1 \ 10])$
R	Weight matrix	$\text{diag}([0.1 \ 0.1])$
γ	Attenuation level	20
λ	Input voltage constraint	50

3.6.2 Learning Process of the Integral Reinforcement Learning Controller

In order to start Algorithm 3.2, an initial control policy W_2^0 is given to the system which converges to W_2^* as shown in (3.22) and the parameter evolution of each entry in the weight matrix W_2 is shown in Fig. 3.4.

FIGURE 3.3: Experiment Setup for the PMSM drive

$$\begin{aligned}
 W_2^0 &= \begin{bmatrix} -5 & 0 & 0 & 0 \\ 0 & -5 & 0 & 12 \end{bmatrix} \\
 W_2^* &= \begin{bmatrix} -2.9674 & -0.0327 & 0.0258 & -0.4273 \\ -0.0172 & -3.03 & -0.789 & 9.995 \end{bmatrix}
 \end{aligned} \tag{3.22}$$

During the learning process, a step change of the desired speed is applied to the machine. Initially, the machine runs at 500 rpm. The desired speed is set to be 300 rpm and then back to 500 rpm at $t=1.5$ s and 3 s respectively.

In order to show the performance improvement during the learning process, the machine speed trajectories and the input voltages under the first policy and the last policy are depicted in Fig. 3.5 and Fig. 3.6, respectively. Both control laws can achieve tracking of the machine speed, however the transient performance is different in terms of overshoot and setting time.

FIGURE 3.4: Convergence of the control matrix W_2

Clearly, the last policy gives a better machine machine speed trajectory with overshoot less than 1% and shorter settling time around 0.5 second.

3.6.3 Comparison of the proposed controller and PID control method

In this section, the machine performances under two control algorithms, namely H_∞ controller with IRL mechanism and classical PID controller, will be evaluated under two test scenarios including a step change of the speed reference and a step change of the load torque.

The control input based on the classical PID controllers with the feed-forward compensation terms can be described in (3.23) and (3.24) as follows.

$$u_d = K_{P,i_d}(i_d^* - i_d) + K_{I,i_d} \int_0^t (i_d^* - i_d) d\tau - P\omega L_q i_q \quad (3.23)$$

FIGURE 3.5: Experimental results of speed trajectory under the initial control policy and the last (converged) control policy

$$u_q = K_{P,i_q}(i_q^* - i_q) + K_{I,i_q} \int_0^t (i_d^* - i_d) d\tau + P\omega L_d \dot{i}_d + \omega \lambda_m \quad (3.24)$$

where K_{P,i_d} , K_{I,i_q} , K_{P,i_q} , K_{P,i_d} are the control gains in the PID controllers, which need to be tuned. The q-axis current reference signal i_q^* is given in (3.25) the d-axis current reference signal i_d^* is regarded as 0.

$$i_q^* = K_{p,\omega}(\omega^* - \omega) + K_{I,\omega} \int_0^t (\omega^* - \omega) dt \quad (3.25)$$

Based on the designed control law in (3.23), (3.24) and (3.25), the block diagram with the classical PID controller is shown in Fig. 3.7. The output loop is the speed control loop, which gives the reference current i_q^* , i_d^* to the current loop controller. The inner loop is the current loop, which tracks the reference current with control inputs u_d , u_q . The design procedure of a classical PID controller can be illustrated as follows [39].

FIGURE 3.6: Experiment results of input voltages under the initial control law and last (converged) control law

FIGURE 3.7: Block diagram of classical PID controller

Assume that the actual machine parameter is the same as the nominal values in TABLE 3.1, the current-loop PID gains can be designed as $K_{P,i_d} = L_d\omega_{ci}$, $K_{P,i_q} = L_q\omega_{ci}$ and $K_{I,i_d} = K_{I,i_q} = R\omega_{ci}$, where ω_{ci} is a design parameter which stands for the cut-off frequency in the current loop. The classical current loop PID controller gives the closed-loop transfer function $G_{i_d}(s), G_{i_q}(s)$ in (3.26), where s is the Laplace variable.

$$G_{i_d}(s) = \frac{i_d}{i_d^*} = \frac{\omega_{ci}}{s + \omega_{ci}}, G_{i_q}(s) = \frac{i_q}{i_q^*} = \frac{\omega_{ci}}{s + \omega_{ci}} \quad (3.26)$$

Based on the the closed-loop current transfer function, $G_{i_d}(s) \approx 1, G_{i_q}(s) \approx 1$ when ω_{ci} is sufficient high, which is $\omega_c = 3300\text{rad/s}$ in our case study and $K_{P,i_d} = 9, K_{P,i_q} = 660$. This closed-loop transfer function from the input ω_{ref} to ω given by $G_\omega(s)$ holds as long as the actual machine parameter exactly equal to the nominal parameter values.

The outer-loop speed PID controller can be chosen as follows.

$$k_{P,\omega} = \frac{4J}{3P\lambda_m}\omega_{cs}, k_{I,\omega} = \frac{4J}{3P\lambda_m}\omega_{cs}^2$$

Therefore, the closed-loop transfer function from ω_{ref} to ω can be given in (3.27). The designed parameter ω_{cs} for the speed loop was tuned as $\omega_{cs} = 10\text{rad/s}$, which gives the $k_{P,\omega} = 0.15, k_{I,\omega} = 1.5$.

$$G_\omega(s) = \frac{2\omega_{cs}s + 2\omega_{cs}^2}{s^2 + 2\omega_{cs}s + 2\omega_{cs}^2} \quad (3.27)$$

Firstly, the speed tracking performances due to a step change of the desired speed under the classical PID controller and the proposed controller are depicted in Fig. 3.8. It is clear that the PID controller will bring a slightly larger overshoot compared with proposed controller. Fig. 3.9 shows that the speed tracking performance due to a step change of the load torque. Initially the machine rotates at 500 rpm with the load torque $T_L = 8N$. Then the load torque is removed from the machine at $t=1$ second. Therefore, a speed rise is expected owing to the removal of the load torque. With the PID controller, the speed rises up to 750 rpm and returns back to the 500 rpm in 1.1 seconds. On the other hand, when the IRL controller

FIGURE 3.8: Experimental results of the conventional PID controller and IRL controller when the desired speed suddenly changes

is applied, the speed overshoot is reduced to 650 rpm and the speed tracking error decays to 0 within 0.7 second.

The performance comparison between the proposed controller and the classical PID controller under two test scenarios is shown in TABLE 3.3.

TABLE 3.3: COMPARISON PERFORMANCE OF TWO CONTROL METHODS IN THE EXPERIMENT

Test scenarios	IRL controller	PID controller
Step change of the desired speed	Settling time 0.5s overshoot 0%	Settling time 1s overshoot 10%
Step change of the load torque	Settling time 0.7s, overshoot 30%	Settling time 1.1s overshoot 50%

FIGURE 3.9: Experimental results of the conventional PID controller and IRL controller when the load torque suddenly changed

3.6.4 Experimental Discussion

We can observe the following features of the proposed IRL controller.

1. The IRL controller is designed to be centralized, which overcomes the structure and order constraints in most PID controllers. A clear cost function is defined and minimized via solving the HJI equation with integral reinforcement learning.
2. The IRL block can be considered as a plug-in control, which can be deliberately activated and deactivated depending on system current performance. When the integral reinforcement learning module is deactivated, the controller degrades into a static H_∞ controller with fixed gains. Moreover, it can be activated when abnormal system behaviors (such as large oscillations or long settling times) are observed. Then the auto-tuning learning process starts and will come up with a new control law.

3. The proposed IRL controller is a data-driven learning-based controller. For sudden parameter changes, the IRL controller needs extra learning time (normally within several seconds) so that the control law converges within few iterations. The proposed controller performs better than the conventional fixed gain PID controller due to its adaptive and learning mechanism.

3.7 Conclusion

An IRL controller has been proposed for H_∞ , input constrained optimal control of PMSM. Policy improvement and policy evaluation are alternatively implemented until the optimal solution of HJI equation is achieved. The major contribution of this data-driven control strategy is that it can greatly enhance parameter robustness of PMSM drives since it requires no prior knowledge of actual machine parameters to design the optimal control policy.

However, relevant technical improvements are needed due to some disadvantages observed during implementation of IRL controller.

1. Three neural networks are needed in off-policy IRL Algorithm for H_∞ control problem, which is not computational efficient enough.
2. Most IRL methods are based on policy iteration where the initial control law needs to be chosen in a way such that it stabilize the system. Otherwise, convergence is not guaranteed in policy iteration based IRL.

Chapter 4

Data-driven Intermediate H_∞ Control Using Off-policy Integral Reinforcement Learning

This chapter addresses an intermediate H_∞ control problem for general unknown linear systems under any given disturbance signals. A novel data-driven IRL algorithm with no disturbance update is proposed. In contrast to the existing model-based control approaches, the proposed IRL algorithm does not require any prior knowledge of the system model and is trained online using the input/output data samples. Moreover, it improves the conventional IRL based H_∞ controller by eliminating the disturbance update step and thus, significantly reduces the computational burden. This makes the proposed algorithm more efficient and computationally inexpensive in industrial applications. The proposed control approach is experimentally validated in a DC-DC boost converter prototype.

4.1 Chapter Background

With the penetration of the renewable energy sources and electronic loads, DC-DC power converters become prevalent in modern power systems [52]. In view of the fast dynamics and high performance requirement of DC-DC power converters,

the development of real-time controllers, which provide fast response, high accuracy, excellent robustness and comparatively low computational burden, attracts a substantial amount of research attentions.

Traditionally, the PID controller is widely adopted to control DC-DC boost converters in industrial applications owing to its simple structure and design procedure. However, a big problem of the traditional PID controller is its sensitivity to the system uncertainty. Its control performance can be seriously degraded under parameter variations. Various control strategies such as back-stepping control, adaptive control [53], model predictive control [54] [55] have been proposed for high performance control of DC to DC converter system. The back-stepping control requires the actual values of the inductance as well as capacitance in order to ensure global closed-loop stability. However this approach suffers from system parameter uncertainties. The adaptive control technique can still ensure closed-loop stability under parameter uncertainty. Nonetheless, adaptive control is not optimal in general and consequently, the transient closed-loop performance such as less overshoot and shorter settling time, cannot be guaranteed for most applications as the objective of adaptive control is to achieve infinite time horizon system stability.

Existing solutions to the optimal state-feedback tracking problem [51] require solving the Hamilton-Jacobi-Bellman equations with complete knowledge of the system dynamics. A promising alternative to deal with parameter uncertainties is to tune the optimal tracking controller based on the interactions between the plant and environment. Motivated by this idea, an integral reinforcement learning (IRL) method was proposed in [16, 19, 56], which enables the controller to learn the Hamilton-Jacobi-Issac (HJI) solutions online without using actual knowledge of the system dynamics. The IRL is generally configured into an actor-critic structure. It updates the system behavior through a recursive two-step iteration [40], namely policy evaluation and policy improvement. Given an initial stable control law, during the policy evaluation step, the IRL controller parametrically approximates the value function according to the current control law. During the policy evaluation step, it gives the new control policy using the gradient of value function estimated in the policy evaluation step. Yang et al. [57], Zhang et al. [58] and Zhang et al. [59] extend the IRL methodology for solving H_∞ regulation problems in some more specific control applications. Practical implementation of IRL can be found in the optimal-load-frequency control in power system [41], a hybrid system

of a jumping robot[42] and controlling arterial oxygen saturation [43]. However, the IRL is seldom applied in power electronic converters. Hence, it is interesting and meaningful to investigate such data-driven control strategy for a DC-DC boost converter. Furthermore, the real-time implementation of IRL at high bandwidth might be considered unrealizable in most existing literature. Noting that existing IRL based H_∞ control algorithm requires the update of the disturbance at each iteration due to the fact that the standard H_∞ control covers for all disturbances (i.e., worst case control). In some practical applications, however, the pattern or characteristics of the disturbance may be available, and the main goal is to develop an effective and computationally inexpensive method to address given disturbances.

Therefore, it is motivated to introduce the intermediate H_∞ control problem for any given disturbances. A novel off-policy IRL algorithm is then proposed to reduce the real-time computational efforts without disturbance update at each iteration. Compared with the existing literature, the main contributions of this chapter are enumerated here.

- A new data-driven off-policy IRL algorithm is proposed to solve the optimal intermediate H_∞ control problem without information of system coefficients. Moreover, the proposed algorithm does not require disturbance update. Different from the conventional H_∞ IRL algorithm, the proposed algorithm has much lower computational burden due to the reduced number of unknowns and therefore is suitable for real-time applications.
- The proposed IRL algorithm is experimentally validated on a DC-DC boost converter.

The rest of this chapter is organized as follows. Section 4.2 describes the modeling of DC-DC boost converter and Section 4.3 formulates an optimal tracking problem using standard H_∞ techniques. Section 4.4 proposes a new intermediate H_∞ control problem and a novel off-policy IRL algorithm is given without disturbance update. Section 4.5 gives the simulation verification of the proposed method and Section 4.6 provides the implementation of the proposed IRL controller. The proposed IRL controller is experimentally validated in real-time on laboratory DC-DC boost converter prototypes in Section 4.7. Section 4.8 draws the conclusions.

4.2 System Model and Problem Formulation

In this section, the state-space model of a DC-DC boost converter is presented and an optimal control problem is formulated.

4.2.1 Mathematical Model of DC-DC converter

The circuit diagram of DC-DC boost converter is depicted in Figure 4.1.

FIGURE 4.1: Circuit Diagram of DC-DC boost converter

The average mathematical model of DC-DC boost converter can be described by the following equations

$$\begin{aligned} \dot{i}_L &= \frac{1}{L}V_{DC} - \frac{1}{L}(1-D)V_{out} \\ \dot{V}_{out} &= \frac{1}{C}(1-D)i_L - \frac{V_{out}}{R_L C} - \frac{1}{C}i_{load} \end{aligned} \quad (4.1)$$

where i_L and v_{out} are the inductor current and output voltage across the output capacitor, respectively. L is the inductance of the inductor. C is the capacitance of the output capacitor. R is the nominal load resistance. i_{load} is the additional fluctuation current due to load change. V_{DC} is the constant source voltage. D is the duty ratio of the power transistor. Note that L , C , and R are unknown constants and D is time-varying.

Define a new control input variable $u := -V_{DC} + (1-D)v_{out}$. Then the new system dynamic becomes

$$\begin{aligned} \dot{i}_L &= -\frac{1}{L}u \\ \dot{V}_{out} &= \frac{u+V_{DC}}{Cv_{out}}i_L - \frac{V_{out}}{R_L C} - \frac{1}{C}i_{load} \end{aligned} \quad (4.2)$$

We make the following assumptions:

Assumption 1: All the electrical coefficients L, C, R_L, D are bounded where the upper and lower bounds of these parameters are known such that there exist known positive constants $L_{\min}, L_{\max}, C_{\min}, C_{\max}, R_{\min}, R_{\max}, D_{\min}$, and $D_{\max}$ satisfy the following conditions

$$\begin{aligned} 0 < L_{\min} &\leq L \leq L_{\max} \\ 0 < C_{\min} &\leq C \leq C_{\max} \\ 0 < R_{\min} &\leq R_L \leq R_{\max} \\ 0 < D_{\min} &\leq D \leq D_{\max} \end{aligned} \tag{4.3}$$

Assumption 2: The inductor current and input/output voltage of DC-DC converter are accessible for measurement.

TABLE 4.1: DC-DC CONVERTER PROTOTYPE PARAMETERS

Parameter	Nominal Value	Actual Value
C	1200 μF	[800,1400] μF
L	390 μH	[100 600] μH
R_L	10 Ω	[5 20] Ω
D	0.5	[0.2 0.8]

The control objective of the DC-DC boost converter can be explained as follows:

1. Assure system stability of the converter within the operation region.
2. Achieve zero steady-state voltage tracking error and disturbance rejection, namely, V_{out} converges to the reference value V_{ref} in the presence of disturbances.
3. Satisfy the transient specifications of the system, which usually concern the settling time and overshoot of the regulated output voltage.

4.2.2 Voltage-tracking Problem

In this section, the voltage-tracking problem is formulated for DC boost converter so that the controller can be designed using H_∞ techniques. Since a nonlinear

quadratic controller to the above system in (4.1) does not provide a tracking performance, an additional state variable x_I is introduced in the following, based on the internal model principle.

$$\dot{x}_I = v_{ref} - v_{out} \quad (4.4)$$

Putting (4.2) and (4.4) together yields the following augmented system, which is used to design an optimal state-feedback control tracking control law.

$$\dot{x} = f(x) + g(x)u + k(x)d + h(x)E \quad (4.5)$$

where $x = [i_L \ v_{out} \ x_I]^T$ denotes the states of the system in a vector form. u represents the control input. $d = i_{load}$ is considered as disturbance. $f(x)$, $g(x)$ and $k(x)$ are the drifting internal dynamics, input dynamics, and disturbance dynamics, respectively. $E = v_{ref}$ and $h(x)$ are considered as a known reference dynamics. More explicitly

$$f(x) = \begin{bmatrix} 0 & 0 & 0 \\ \frac{V_{DC}}{CV_{out}} & -\frac{1}{R_L C} & 0 \\ 0 & -1 & 0 \end{bmatrix} x, g(x) = \begin{bmatrix} -1/L \\ \frac{i_L}{CV_{out}} \\ 0 \end{bmatrix} \quad (4.6)$$

$$k(x) = \begin{bmatrix} 0 \\ -1/C \\ 0 \end{bmatrix}, h(x) = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (4.7)$$

If the system dynamic in (4.5) with $d = 0$ and $E = 0$ can be stabilized with the control law $u = Kx$, the output voltage tracking performance can be guaranteed with the same state-feedback control law. This is because (4.5) contains the internal model of the constant reference input, namely, the integrator (4.4).

4.3 Standard H_∞ Control Problem

In this section, we design a robust controller to reject the disturbance $d(t)$ in (4.5) using the standard H_∞ control techniques. Due to the internal model (4.4), we

consider the regulator problem for (4.5) with $E = 0$. The same controller will then guarantee the tracking performance if E is a nonzero constant.

Definition 4.1. The H_∞ control problem is to find a control policy u , such that the following two conditions hold.

1) The closed-loop system (4.5) satisfies the following attenuation condition

$$\int_t^\infty (x^T Q x + u^T R u) d\tau \leq \gamma^2 \int_t^\infty (d^T d) d\tau. \quad (4.8)$$

where Q is positive semidefinite Hermitian matrix. The i th diagonal element of Q , which is $q_{i,i}$, determines the direct weighting on the i th state. R is a positive scalar which determines the weighting on the control input.

2) The closed-loop system dynamics with $d = 0$ is locally asymptotically stable at the origin.

Based on (4.8), define the performance function

$$J(u, d) = \int_t^\infty (x^T Q x + u^T R u - \gamma^2 d^T d) d\tau. \quad (4.9)$$

Differentiating (4.9) and noting that $V(x(t)) = J(u(t), d(t))$ give the following Bellman equation

$$H(V, u, d) = x^T Q x + u^T R u - \gamma^2 d^T d + V_x^T (f + gu + kd) = 0, \quad (4.10)$$

where V_x is the derivative of V over $x(t)$. Clearly $H(u^*) = 0$. The optimal control and disturbance satisfy the stationary conditions of $\frac{\partial H}{\partial u^*} = \frac{\partial H}{\partial d^*} = 0$, which leads to

$$u^* = -\frac{1}{2} R^{-1} g^T V_x^* \quad (4.11)$$

$$d^* = \frac{1}{2\gamma^2} k^T V_x^* \quad (4.12)$$

where V^* is the optimal value function.

Putting (4.11) and (4.12) into (4.10) yields the Hamilton-Jacobi-Issacs (HJI) equation

$$\begin{aligned} H(V^*, u^*, d^*) &= x^T Q x + (V_x^*)^T f \\ &\quad - \frac{1}{4} (V_x^*)^T g R^{-1} g^T V_x^* + \frac{1}{4\gamma^2} V_x^{*T} k k^T V_x^* = 0. \end{aligned} \quad (4.13)$$

Algorithm 4.1: Standard H_∞ control with full information of the system dynamics [29]

Step 1: Let $i=0$ and select stabilizing initial policy u^0 and set $d^0 = 0, E = 0$. Apply control and disturbance law u^i, d^i at i -th iteration to the system, find V^i using the Bellman equation in (4.14).

$$\begin{aligned} &x^T Q x + (V_x^i)^T (f + g u_i + k d_i) \\ &\quad + (u^i)^T R u^i - \gamma^2 (d^i)^T d^i = 0. \end{aligned} \quad (4.14)$$

Step 2: Update the control law and disturbance law using

$$u^{i+1} = -\frac{1}{2} R^{-1} g^T V_x^i \quad (4.15)$$

$$d^{i+1} = \frac{1}{2\gamma^2} k^T V_x^i \quad (4.16)$$

Step 3: Let $i = i + 1$ and Go back to **Step 1** until convergence.

Algorithm 4.2: Off-policy IRL algorithm for standard H_∞ control [29]

Step 1: Let $i=0$ and select stabilizing initial policy u^0 and set $d^0 = 0, E = 0$. Apply a fixed control policy u to the system continuously and collect required system information about the state x , control input u and disturbance d at N different sampling interval T .

Step 2: With the collected data of $x(t), u(t)$ as well as the optimal control policy u_i , disturbance policy d_i at the i -th iteration, solve the following off-policy Bellman equation for the cost function V_i , optimal control policy u_{i+1} and disturbance policy d_{i+1} at the $(i+1)$ -th iteration simultaneously

$$\begin{aligned} &V^i(x(t+T)) - V^i(x(t)) = \\ &\quad \int_t^{t+T} (-x^T Q x - (u^i)^T R u^i + \gamma^2 (d^i)^T d^i) d\tau + \\ &\quad \int_t^{t+T} (-2u^{i+1T} R(u - u^i) + 2\gamma^2 (d^{i+1})^T (d - d_i)) d\tau. \end{aligned} \quad (4.17)$$

Step 3: Let $i = i + 1$ and go back to **Step 2** until convergence

---

A standard policy iteration technique to find the optimal H_∞ control is given in **Algorithm 4.1**. It is shown in [29] that **Algorithm 4.1** converges to the solution of the HJI.

Theorem 4.1. [56]: Select $\gamma > \gamma^* > 0$, suppose $V^*(x) > 0 \in C^1$ is a smooth positive definite solution to (13). Then the closed-loop system

$$\dot{x} = f(x) + g(x)u^* = f(x) - \frac{1}{2}gR^{-1}g^TV_x^*$$

is locally asymptotically stable with the optimal control law u^* . Moreover, for this choice of control input, (13) holds for any $d(t) \in \mathcal{L}_2[0, \infty)$.

The off-policy IRL method in **Algorithm 4.2** is given in [29] to implement **Algorithm 4.1** online using measured data. **Algorithm 4.1 and 4.2** both require the update of disturbance d^i , which is not efficient for industry application since the worst-case disturbance will hardly occur in real-time application.

4.4 Intermediate H_∞ Control Problem

In this section, a new problem called intermediate H_∞ control problem is introduced, which deals with a given disturbance time-signal. Then, a new on-policy IRL algorithm is proposed without updating disturbance (4.16). Finally, an off-policy IRL algorithm is developed, without requiring disturbance updates, and hence is more efficient than conventional Algorithm 4.2.

Definition 4.2. For any given disturbance time-signal $d(t)$, the intermediate H_∞ control problem is to find a control policy u , such that the two conditions in Definition 1 hold.

Substituting (4.11) into (4.10) but leaving $d(t)$ arbitrary yields the following intermediate HJI equation

$$\begin{aligned} 0 = & x^T Q x + V_x^T f(x) - \frac{1}{4} V_x^T g R^{-1} g^T V_x \\ & - \gamma^2 d^T d + V_x^T k d, \end{aligned} \tag{4.18}$$

which is a partial differential equation with a time-varying signal $d(t)$.

An equivalence integral version of Bellman equation (4.10) is (4.19) with arbitrary T

$$V(x(t)) - V(x(t+T)) = \int_t^{t+T} (x^T Q x + u^T R u - \gamma^2 d^T d) d\tau \quad (4.19)$$

Substitute (4.11) into (4.19) to obtain the modified HJI equation

$$\begin{aligned} V^*(x(t)) - V^*(x(t+T)) = \\ \int_t^{t+T} (x^T Q x + \frac{1}{4}(V_x^*)^T g R^{-1} g^T V_x^* - \gamma^2 d^T d) d\tau \end{aligned} \quad (4.20)$$

We call (4.18) the model-based intermediate HJI equation and note that only parameter L is needed in $g(x)$ to implement (4.20).

Theorem 4.2. Let u^{i+1} be given by $u^{i+1} = -\frac{1}{2}R^{-1}g^T V_x^i$ where $V^i(t)$ is the solution to (4.19) under control law u^i and given disturbance $d(t)$. Then u^{i+1} stabilizes the system (4.5) with $d = 0$. Moreover equation (4.8) holds for that specific $d(t)$ applied.

Proof. Firstly, we prove the stability of the new control policy u^{i+1} . Consider the function V^i , which is a Lyapunov function that makes equation (4.18) holds for the control law u^i and disturbance d . Taking the derivative of V^i during the system trajectory $\dot{x} = f(x) + g(x)u^{i+1} + k(x)d$ and using the facts that $x^T Q x + (u^i)^T R u^i - \gamma^2 d^T d + V_x^T(f + gu^i + kd) = 0$ and $u^{i+1} = -\frac{1}{2}R^{-1}g^T V_x^i$. Then, we obtain

$$\begin{aligned} \dot{V}^i &= (V_x^i)^T (f + gu^i + kd) + (V_x^i)^T g((u^{i+1}) - u^i) \\ &= -x^T Q x - (u^i)^T R u^i + \gamma^2 d^T d - 2u^{i+1} R g(u^{i+1} - u^i) \\ &= -x^T Q x + \gamma^2 d^T d - (u^{i+1})^T R u^{i+1} \\ &\quad - (u^{i+1} - u^i)^T R (u^{i+1} - u^i) \end{aligned} \quad (4.21)$$

It is clear that the $\dot{V}$ is negative when $d = 0$. Next, It is proved that the bounded $\mathcal{L}_2$ gain is satisfied by the new control law. As shown in (4.22), the derivative of V^i over new system trajectory becomes

$$\begin{aligned} \dot{V}^i &= -x^T Q x + \gamma^2 d^T d - (u^{i+1})^T R u^{i+1} \\ &\quad - (u^{i+1} - u^i)^T R (u^{i+1} - u^i) \\ &\leq -x^T Q x + \gamma^2 d^T d - (u^{i+1})^T R u^{i+1} \end{aligned} \quad (4.22)$$

$$\begin{aligned} & V(x(T)) - V(x(0)) \\ & \leq \int_0^T (-x^T Q x + \gamma^2 d^T d - (u^{i+1})^T R u^{i+1}) d\tau \end{aligned} \quad (4.23)$$

Select $x(0) = 0$ with $V(0) = 0, V(x(T)) > 0$, one has

$$\int_0^T (x^T Q x + (u^{i+1})^T R u^{i+1}) d\tau \leq \int_0^T (\gamma^2 d^T d) d\tau \quad (4.24)$$

This completes the proof. $\square$

We are now in a position to develop a more efficient off-policy IRL algorithm that does not need to update disturbance. Algorithm 4.3 is based on Theorem 4.2. It is the same as Algorithm 4.1 without the disturbance update. Algorithm 4.4 is based on Theorem 4.2. It is the same as Algorithm 4.2 without the disturbance update

---

Algorithm 4.3: H_∞ on-policy Reinforcement Learning (IRL) algorithm without updating disturbance

---

Select stabilizing initial policy u^0

Step 1: Apply a given time signal $d(t)$ and control law u^i at i -th iteration to the system, find V^i using the following on-policy Bellman equation.

$$x^T Q x + (V_x^i)^T (f + g u^i + k d) + u^{iT} R u^i - \gamma^2 d^T d = 0 \quad (4.25)$$

Step 2: Update the control law using

$$u^{i+1} = \arg \min_u [H(V^i, u, d)] = -\frac{1}{2} R^{-1} g^T V_x^i \quad (4.26)$$

Step 3: Let $i = i + 1$ and Go back to **Step 1** until convergence.

Theorem 4.3. **Algorithm 4.3 and Algorithm 4.4** are equivalent. Moreover, if they converge, they converge to the solution to the intermediate HJI equation for any given disturbance time signal $d(t)$. That is, the intermediate H_∞ control problem is solved.

Proof. We prove that both **Algorithms 4.3 and 4.4** have the same convergence properties. Dividing both the sides of the off-policy IRL Bellman equation in Algorithm 4.4 by T , and taking limit results in

---

Algorithm 4.4: H_∞ off-policy Integral Reinforcement Learning (RL) algorithm without updating disturbance

---

Selecting stabilizing initial policy u^0

Step 1: Let $i=0$ and select stabilizing initial policy u^0 and set $d^0 = 0, E = 0$. Apply disturbance $d(t)$ to the system continuously and collect required system information about the state x , control input u^0 from t to $t + T$.

Step 2: With the collected data of $x(t), u(t)$ as well as the optimal control policy u^i at the i -th iteration, solve the following Bellman equation (4.27) for the cost function V^i , optimal control policy u^{i+1} at the $(i + 1)$ -th iteration.

$$\begin{aligned} & V^i(x(t + T)) - V^i(x(t)) \\ &= - \int_t^{t+T} (x^T Q x + (u^i)^T R u^i - \gamma^2 d^T d) d\tau \\ &+ \int_t^{t+T} 2(-u^{i+1})^T R (u - u^i) d\tau + \int_t^{t+T} V_x^T h(x) E d\tau \end{aligned} \quad (4.27)$$

Step 3: Let $i = i + 1$ and go back to **Step 2** until convergence

---

$$\begin{aligned} & \lim_{T \rightarrow 0} \frac{V^i(x(t + T)) - V^i(x(t))}{T} = \\ & \lim_{T \rightarrow 0} \frac{\int_t^{t+T} (-x^T Q x - (u^i)^T R u^i + \gamma^2 d^T d) d\tau}{T} \\ & + \lim_{T \rightarrow 0} \frac{\int_t^{t+T} (-2u^{i+1T} R (u - u^i)) d\tau}{T} \end{aligned} \quad (4.28)$$

where u is the pre-chosen control law, d is the actual disturbance to the system and u^i is the control law and disturbance law to be updated at each iteration.

By L'Hospital's rule, the left side becomes

$$\begin{aligned} & \lim_{T \rightarrow 0} \frac{V^i(x(t + T)) - V^i(x(t))}{T} \\ &= (V_x^i)^T (f + g u^i + k d + g (u - u^i) + h(x) E) \end{aligned} \quad (4.29)$$

and the right side becomes

$$\begin{aligned}
& \lim_{T \rightarrow 0} \frac{\int_t^{t+T} (-x^T Q x - (u^i)^T R u^i + \gamma^2 d^T d) d\tau}{T} \\
& + \lim_{T \rightarrow 0} \frac{\int_t^{t+T} (-2(u^{i+1})^T R (u - u^i)) d\tau}{T} \\
& = -x^T Q x - (u^i)^T R u^i + \gamma^2 d^T d \\
& - 2(u^{i+1})^T R (u - u^i) + V_x^i h(x) E
\end{aligned} \tag{4.30}$$

Hence, we obtain

$$\begin{aligned}
& (V_x^i)^T (f + g u^i + k d + g(u - u^i) + h(x) E) \\
& = -x^T Q x - (u^i)^T R u^i + \gamma^2 d^T d \\
& - 2(u^{i+1})^T R (u - u^i) + (V_x^i)^T h(x) E
\end{aligned} \tag{4.31}$$

Substituting the updated policy u^{i+1} from **Algorithm 4.3** into the above equation, we obtain

$$\begin{aligned}
& x^T Q x + (V_x^i)^T (f + g u^i + k d) \\
& + (u^i)^T R u^i - \gamma^2 d^T d = 0
\end{aligned} \tag{4.32}$$

This completes the proof. $\square$

4.5 Simulation Verification of the Proposed Method

In order to show the effectiveness of the proposed **Algorithm 4.4**, a simulation case study on a linearized DC-DC boost converter is presented here.

Consider the linear version of (4.5) with parameters stated in Table 4.1 given by

$$\dot{x} = Ax + Bu + Kd + HE \tag{4.33}$$

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 416 & -83 & 0 \\ 0 & -1 & 0 \end{bmatrix}, B = \begin{bmatrix} -2564 \\ 0 \\ 0 \end{bmatrix} \tag{4.34}$$

$$K = \begin{bmatrix} 0 \\ -833 \\ 0 \end{bmatrix}, H = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (4.35)$$

Select performance function (4.9) with $Q = \text{diag}([0.1, 0.1, 500])$, $R = 0.1$, $\gamma = 10$, $\alpha = 0.3$, $E = 25$. During the learning process with **Algorithm 4.4**, the knowledge of A, B, K are not used.

Start with an initial stable control policy $u^0 = \begin{bmatrix} 5 & 5 & -30 \end{bmatrix} x$ and perform **Algorithm 4.4** with $d(t) = 0.1 \sin(100\pi t)$. The algorithm converges to the control policy $u^\infty = \begin{bmatrix} 1.4 & 0.9 & -70 \end{bmatrix} x = W^T x$. This converging policy is stabilizing, as predicted by **Theorem 4.2**.

The convergence plots for the feedback gains $W^T = \begin{bmatrix} W(1) & W(2) & W(3) \end{bmatrix}$ are given in Figure 4.2. The state convergence is shown in Figure 4.3. It is seen that $x(2) = V_{out}$ converges to reference $E = 25$

FIGURE 4.2: Convergence of optimal control policy

FIGURE 4.3: State trajectory under converged policy u^∞

4.6 Implementation of The Off-policy IRL for a DC-DC Boost Converter

In order to implement the intermediate H_∞ off-policy IRL **Algorithm 4.4** on the DC-DC boost converter, some preliminary analysis is needed. In this section, we discuss selection of the matrices, Q and R in (4.9) and the choice of an initial stable gain in **Algorithm 4.4**. Then, **Algorithm 4.4** is applied to the DC-DC boost converter hardware in the next section.

4.6.1 Design of The Cost Function: Q and R Selection

The Q and R matrices in (4.9) are the key design parameters of the optimal control problem. The Q matrix weights each state while R weights the control effort. The practical design of LQR controller typically requires some iteration in Q and

R values until desired performance is achieved. The following thumb rules are proposed to give some practical design criteria [51] and the selection of Q and R is given in TABLE 4.2.

1. First of all, it is obvious that the optimum control remains invariant when Q and R are multiplied by a common scalar factor.
2. For design simplicity, the Q and R values out of the main diagonal are set to zero.
3. The overshoot and settling time in inductor current and output voltage trajectory can be adjusted by tuning $q_{1,1}$ and $q_{2,2}$, respectively.
4. In order to achieve voltage reference tracking and disturbance attenuation, a large weight $q_{3,3}$ is assigned to the integral state x_I .

TABLE 4.2: PARAMETER SETTING FOR THE OPTIMAL CONTROL PROBLEM

Symbol	Quantity	Value
Q	Weight matrix	$\text{diag}([0.1 \ 0.1 \ 500])$
R	Weight matrix	0.1
γ	Attenuation level	10

4.6.2 Initial Stabilizing Gain

For IRL algorithm, there normally exists a strong assumption that the initial law must be stabilizing. Not much work has been done to show how to choose the initial stabilizing control law in the case study of the off-policy IRL algorithm. However, an initial robust controller can be determined based on parameter bounds stated in Section 4.2 and Linear Matrix Inequality [60].

Reconsider the system dynamics in (4.5) to (4.7) without disturbance and the parameter bounds in (4.3), then

$$\dot{x} = f(x) + g(x)u \quad (4.36)$$

$f(x)$, $g(x)$ can be rewritten as $A(p)x$ and $B(p)$, where p denotes the system states and coefficients. Based on the system operation region and coefficients, the system matrix with uncertainty $A(p)$ and $B(p)$ can be included in a convex polytope, which is defined by its vertices $G_1, G_2, \dots, G_{16}$

$$\begin{aligned} \begin{bmatrix} A(p) & B(p) \end{bmatrix} &\in Co\{G_1 \dots G_{16}\} \\ &:= \left\{ \sum_{i=1}^{16} \lambda_i G_i, \lambda_i \geq 0, \sum_{i=1}^{16} \lambda_i = 1 \right\} \end{aligned} \quad (4.37)$$

where G_i is one pair of system uncertain matrix $A(p), B(p)$ such as

$$\begin{aligned} G_1 &= [A_1(L_{\min}, C_{\max}, R_{\min}, D_{\min}), B_1(L_{\min})]; \\ G_2 &= [A_2(L_{\min}, C_{\min}, R_{\max}, D_{\min}), B_2(L_{\min})]; \\ &\dots \\ G_{16} &= [A_{16}(L_{\max}, C_{\max}, R_{\max}, D_{\max}), B_{16}(L_{\max})]; \end{aligned} \quad (4.38)$$

The system can be stabilized by state feedback $u = K_0 x$ with a guaranteed region if and only if there exists a positive definite matrix P such that the following inequality holds for all $\forall i = 1, 2, \dots, 16$

$$(A_i + B_i K)^T P + P(A_i + B_i K) < 0 \quad (4.39)$$

4.6.3 Procedure of Algorithm 4.4 Off-policy IRL

Based on the initial stabilizing control law, the procedure for implementing Algorithm 4 is provided as follows.

1. Assign an initial stable controller u^0 that can stabilize the system.
2. Equation (4.27) can be regarded as a linear-regression problem and solved by recursive least squares algorithm. The weight vector $W_1^i, W_2^{i+1}, W_3^{i+1}$, can be solved after sufficient information is collected from the system.
3. Convergence test. Evaluate the norm between the updated estimated W^{i+1} and the previous estimated W^i . If the norm value is smaller than a predefined threshold ϵ , Algorithm 4.4 will be terminated. Otherwise, Algorithm 4.4 will return back to step 2 to further update the control policy.

4.7 Experimental Verification of IRL controller on DC-DC Boost Converter

Here in this section, we carry out hardware experiments to implement Algorithm 4.4 on the DC-DC boost converter.

4.7.1 Prototype Setup

To verify the dynamic performance of the proposed **Algorithm 4.4**, a DC-DC boost converter prototype has been built as shown in Figure 4.4 with the parameters listed in Table 4.3. A dSPACE MicroLab Box with DS1202 base board is adopted for real-time implementation of the proposed controller with a switching frequency of 20 kHz.

FIGURE 4.4: DC-DC boost converter prototype

TABLE 4.3: COMPONENT LIST of DC CONVERTER PROTOTYPE

Name	Manufacture Part Number	Value
DC converter	CRD-5FF0912P	NA
Inductor	1140-391k-RC	390 μ H
Capacitor	500C122T250BA2B	1200 μ F
Resistor	NA	20 Ω

4.7.2 Comparison of Algorithm 4.4 Based Controller with Other Advanced Control Methods

Note that it is not easy to directly change the DC-DC boost converter in hardware experiments. As an alternative to implement parameter change in DC-DC converter, it can be done easily by changing the parameters in control scheme. In this chapter, changes of system parameters in the controller has been made to verify the learning performance of the proposed method.

Based on the discussion in Section 4.6, the initial H_∞ control law $u = W_2^0 x$ using initial value of parameter coefficients is chosen and finally converges the $u = W_2^* x$ as in (4.40) via IRL.

$$\begin{aligned} W_2^0 &= \begin{bmatrix} 5 & 5 & -30 \end{bmatrix} \\ W_2^* &= \begin{bmatrix} 1.4 & 0.0327 & -70 \end{bmatrix} \end{aligned} \quad (4.40)$$

The performance comparison between the proposed controller and initial robust controller under three Test Scenarios are depicted in Figure 4.5 to Figure 4.10 under three test scenarios, where channel C1, C2, C3, and C4 represent DC source input voltage, DC source input current, output voltage, load current respectively.

1. Test Scenario 1: Step change of the output reference voltage from 25V to 15V.
2. Test Scenario 2: Step change of the input voltage from 10V to 5V.
3. Test Scenario 3: Step change of the load resistance from 20 Ω to 6.7 Ω .

Scenario 1: Under this scenario, the desired output voltage (v_{out}) is set from 25V to 15V and the load resistance remains at 20 Ω . Figure 4.5 and Figure 4.6 present the

FIGURE 4.5: Step change of output reference under initial H_∞ controller

FIGURE 4.6: Output voltage reference change with final control law

comparative experimental waveforms of the initial H_∞ controller and the proposed IRL controller, respectively. It can be inferred from Figure 4.5 and Figure 4.6 that the regulation performance of the conventional is significantly improved after applying the IRL learning laws. That is, the proposed control scheme precisely tracks the desired speed with fast dynamic response with settling time 50ms and nearly zero overshoot. On the contrary, it is obvious from Figure 4.5 that the conventional initial H_∞ controller shows its poor capacity with settle time 70ms and overshoot 10%.

Scenario 2: In this experimental scenario, the desired speed input voltage V_{DC} is suddenly changed from 10V to 5V and the load current is set at 0.75A. Figure 4.7 and Figure 4.8 show the experimental results of and the initial H_∞ controller

FIGURE 4.7: Step change of input voltage under initial H_∞ controller

FIGURE 4.8: Step change of input voltage under final control law

FIGURE 4.9: Load change with initial H_∞ controller

FIGURE 4.10: Load change with final control law

and the proposed IRL controller. In Figure 4.8, the proposed IRL control scheme (settling time: 55 ms) exhibits the faster dynamic behavior with smaller voltage drop 2.5V.

Scenario 3: Under this scenario, the desired output voltage (v_{out}) is set to 15V and the load current suddenly changes from 0.75 A to 2.2A. Figure 4.9 and Figure 4.10 present the comparative experimental waveforms of the initial H_∞ controller and the proposed IRL controller, respectively. It can be inferred from Figure 4.9 and Figure 4.10 that the regulation performance of the proposed controller is better with the settling time 50ms and output voltage maximum sag 2V.

The performance comparison between the proposed controller and other controllers is also summarized in TABLE 4.4.

TABLE 4.4: PERFORMANCE COMPARISON of THOSE TWO CONTROL METHODS

Test scenarios	Initial H_∞ control law	Final Control law with IRL
1	Settling time 70ms 10 % overshoot	Settling time 50 ms no overshoot
2	Settling time 70 ms voltage drop 3.5 V	Settling time 55 ms voltage drop 2.5 V
3	Settling time 60 ms voltage drop 5 V	Settling time 50 ms voltage drop 2 V

4.8 Conclusions

An off-policy IRL algorithm for the intermediate H_∞ control problem with no disturbance update has been implemented on a DC boost converter in real-time. Compared with existing control techniques for a DC boost converter, the proposed controller does not need to update the disturbance, which is computationally inexpensive since the unknown variables have been reduced. As a result, the main accomplishments of this chapter are summarized as follows:

1. A novel off-policy IRL algorithm has been proposed to solve the intermediate H_∞ control problem. Compared with the conventional off-policy IRL for standard H_∞ control, the proposed method does not need disturbance

update while solving the intermediate H_∞ control problem. The number of unknown variables during the learning process are further reduced, making the proposed method computationally efficient.

2. The rigorous mathematical proof of the stability and convergence result of the proposed IRL algorithm has been presented to solve the intermediate H_∞ control problem.
3. The proposed IRL based controller has been experimentally tested on the DC-DC boost converter to precisely track the reference voltage under cases of parameter variations, external load current disturbances and input voltage disturbance.
4. The comparative studies on the experimental results of the conventional fixed gain H_∞ state-feedback control and the proposed control protocols have been presented.

Chapter 5

Value Iteration based Off-policy Integral Reinforcement Learning and Convergence Analysis

This chapter investigates the convergence properties of a value iteration (VI) based on-policy IRL technique for solving optimal control problems of the continuous-time (CT) input-affine nonlinear systems without using either system drift dynamics or knowledge of initial stable control law. As a result, three modes of global monotonic convergence of VI based IRL are presented. The selection of an initial value function and sampling time will determine the convergence mode of VI based on policy IRL. Numerical simulations are carried out to verify and further investigate these properties. Moreover, to further overcome the technical limitation of on-policy IRL where system input dynamic is needed, VI based off-policy IRL is proposed for solving the same control problem with zero information of the input affine nonlinear system. The performance of the proposed VI based off-policy IRL is evaluated using the same nonlinear system example which show the feasibility and effectiveness of the proposed algorithms.

5.1 Chapter Background

In the past several decades, considerable research has been devoted into developing RL methods to solve optimal control problems in the discrete Markov decision process (MDP) [14] which is then extended to discrete-time (DT) [61] and continuous-time (CT) dynamical systems [20]. The iteration schemes of RL can be divided into two classes: policy iteration (PI) algorithms and value iteration algorithms (VI) while PI exactly evaluates the value function at each iteration. VI approximates the value function by one-step back-up operator.

For continuous time (CT) cases, the PI technique was first proposed by Murray et al. [62], and then, a class of algorithms known as IRL was proposed by Vrabie et al in [16] to solve a CT linear optimal control problem. The convergence and stability of PI based on-policy IRL have been completely proven under an initial stabilizing policy for a CT linear optimal control problem [16] and a CT nonlinear optimal control problem [19] without knowing the internal dynamics model. Furthermore, to make PI based IRL work for completely unknown system, off-policy PI based IRL methods have been proposed under an initial stabilizing policy for a CT linear optimal control problem [18] and a CT nonlinear optimal control problem [63]. However, all PI based IRL methods have the technical limitations that the initial stable control law is always needed as a prerequisite.

Unlike the PI based IRL methods, VI based IRL methods start with a given cost function, which is simple in calculation and easy to be implemented. Moreover, during the iteration process, there is no need to guarantee the initial control policy to be admissible. There are also some efforts for showing the properties of VI based on-policy IRL for a linear optimal control problem, including its stability and global/local convergence [64] [65] [66] [67]. However, the convergence mode is not properly addressed in early researches. The mixed convergence mode is not highlighted in existing literature [58] [66] [68] [69]. The VI based IRL for a linear system is also introduced in [70]. This paper solves a linear optimal regulation using stochastic programming and online data. It solves the LQR problem using stochastic programming and online data only. However, the proposed method [70] may introduce additional design challenges when determining the stochastic sequence in the stochastic programming approach.

Compared with the existing literature on VI based IRL [64] [65] [66], the main contributions of this chapter are enumerated here.

- Firstly, the convergence analyses are investigated for a VI-based on-policy IRL on a nonlinear input-affine system. The impacts of the sampling time T_s and the initial value function V^0 on convergence modes of the VI-based on-policy IRL is discussed in detail. The convergence could be either in a PI mode (a monotonically decreasing mode), or a VI mode (a monotonically increasing mode) or a mixed mode.
- Secondly, a VI-based off-policy IRL algorithm is proposed for completely unknown systems. The proposed algorithm does not require any knowledge on the initial stable control law, system drifting dynamic or the input dynamics matrix. It is an online learning method, which is more practical in industry applications considering that actual control tasks are always with tight time constraints.

The rest of this chapter is organized as follows. Section 5.2 describes the system modeling and formulation of a nonlinear optimal control problem. Section 5.3 presents the analyses of three different convergence modes of VI based on-policy IRL. Section 5.4 shows the implementation of Algorithm 5.1 using neural networks. The mode of convergence is verified in a linear system and an input-affine nonlinear system in Section 5.5. Section 5.6 proposes a VI based off-policy in Algorithm 5.2. Section 5.7 proposes a more efficient VI based off-policy in Algorithm 5.4. Finally, Section 5.8 draws the conclusion.

5.2 System Modeling and Problem Formulation

In this section, the model of a time-invariant input-affine nonlinear system is presented and the problem formulation is proposed for one kind of nonlinear optimal control problem based on this. The system model could be denoted as follows,

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t), \quad (5.1a)$$

$$x(0) = x_0, \quad (5.1b)$$

with the system state $x(t) \in \mathbf{R}^n$, the control variable $u(t) \in \mathbf{R}^m$, the system dynamics $f(x(t)) : \mathbf{R}^n \rightarrow \mathbf{R}^n$ and the input dynamics $g(x(t)) : \mathbf{R}^n \rightarrow \mathbf{R}^{n \times m}$, which satisfies $f(0) = 0$. In addition, $f(x(t)), g(x(t))$ are Lipschitz continuous on a compact set $\Omega \in \mathbf{R}^n$, which contains the origin such that $x = 0$ is an equilibrium point of the system. Assume that system (5.1) is controllable with at least one continuous control law u which can asymptotically stabilize the system at the origin.

The optimal performance function is defined in (5.2),

$$V(x(t)) = \min_u \int_t^\infty (Q(x(\tau)) + u(\tau)^T R u(\tau)) d\tau \quad (5.2)$$

where $Q(x)$ is a positive definite function of x , i.e., $\forall x \neq 0, Q(x) > 0$. $Q(x)$ equals 0 if and only if $x = 0$. R is a positive definite matrix $\in \mathbf{R}^{n \times n}$

The optimal V^* can be obtained by solving the Hamilton-Jacobi-Bellman (HJB) equation,

$$H(V^*, u^*) = Q(x) + (V_x^*)^T f - \frac{1}{4} (V_x^*)^T g R^{-1} g^T V_x^* = 0. \quad (5.3)$$

5.3 VI Based On-Policy IRL for a Nonlinear Optimal Control Problem

A conventional VI based on-policy IRL technique to solve the optimal nonlinear control problem is given in **Algorithm 5.1** [67]. However, its convergence property is not properly addressed in existing literature. In this section, we propose few lemmas and theorems to complete the convergence mode analysis for a VI based IRL algorithm.

The following **Definition 5.1** and four lemmas are firstly introduced, which will be used in the convergence analysis of the VI based on-policy IRL.

Lemma 5.1. Given $V^i(x) \geq 0$ holds for $\forall x \in \Omega$, the value function $V^{i+1}(x)$ computed using (5.5) satisfies $V^{i+1}(x) \geq 0$

Algorithm 5.1: A VI based on-policy IRL for nonlinear optimal control with partial information of system dynamics [67]

---

1. **Step 1: Initialization.** Let $i = 0$ and select an initial value function $V^0(x)$ with $V^0(0) = 0$, and the control law is shown as follows,

$$u^0 = -\frac{1}{2}R^{-1}g^TV_x^0 \quad (5.4)$$

2. **Step2:** For each iteration i , use the knowledge V^i and the control law u^i to determine the value function V^{i+1} using (5.5).

$$V^{i+1}(x(t)) = \int_t^{t+T_s} (Q(x) + (u^i)^TRu^i)d\tau + V^i(x(t+T_s)) \quad (5.5)$$

where T_s is the sampling time.

3. **Step 3:** Update the control law to be implemented at the $i + 1$ iteration u^{i+1} using,

$$u^{i+1} = -\frac{1}{2}R^{-1}g^TV_x^{i+1} \quad (5.6)$$

4. **Step 4:** Let $i \leftarrow i + 1$, and go to **Step 2** until convergence.
-

Proof. It is straightforward using (5.5) only.

Given $V^i(x) \geq 0$ holds for all $x \in \Omega$, then $V^i(x(t+T_s)) \geq 0$. The sum of left-hand side of (5.5) is always positive definite.

Therefore $V^{i+1} \geq 0$, which completes the proof. □

Definition 5.1. Let $HJB(V(x))$ be the nonlinear HJB operator defined as

$$HJB(V(x)) = V_x^T(f(x) + g(x)u) + Q(x) + u^TRu \quad (5.7)$$

with $u = -\frac{1}{2}R^{-1}g^TV_x$

Lemma 5.2. If $HJB(V^i(x)) \geq 0$, then $V^i(x) \leq V^*(x)$.

Proof. Define the difference between $V^*(x)$ and $V^i(x)$ as

$$\Delta V^i := V^*(x) - V^i(x), \quad (5.8)$$

and calculate the derivative of ΔV^i along the system trajectory $\dot{x} = f(x) + g(x)u^*$

$$\begin{aligned}\Delta \dot{V}^i &= (V_x^*)^T(f(x) + g(x)u^*) - (V_x^i)^T(f(x) + g(x)u^i) \\ &= -HJB(V^i(x)) - (u^i - u^*)^T R(u^i - u^*)\end{aligned}\tag{5.9}$$

Integrate (5.9) from t to ∞ along the system trajectory $\dot{x} = f(x) + g(x)u^*$.

$$\Delta V^i(x(t)) - \Delta V^i(x(\infty)) = - \int_t^\infty \Delta \dot{V}^i d\tau\tag{5.10}$$

Using the fact $x(\infty) = 0$ with system trajectory under u^* and $V^i(0) = 0$, $\Delta V^i(x(t)) \geq 0$ can be derived.

Therefore, we could conclude $V^*(x) \geq V^i(x)$ if $HJB(V^i(x)) \geq 0$ holds.

□

Lemma 5.3. if $HJB(V^i(x)) \leq 0$, then $V^i(x) \geq V^*(x)$

Proof. Consider the $V^i(x)$ along the system trajectory $\dot{x} = f(x) + g(x)u^i$ and use the condition $HJB(V^i(x)) \leq 0$

$$HJB(V^i(x)) \leq 0\tag{5.11a}$$

$$(V_x^i)^T(f(x) + g(x)u^i) \leq -(Q(x) + u^i R u^i)\tag{5.11b}$$

$$V^i(\infty) - V^i(x(t)) \leq \int_t^\infty -(Q(x(\tau)) + (u^i(\tau))^T R u^i(\tau)) d\tau\tag{5.11c}$$

Rearrange equation (5.11),

$$V^i(x(t)) \geq \int_t^\infty (Q(x(\tau)) + u^i(\tau)^T R u^i(\tau)) d\tau \geq V^*(x(t))\tag{5.12}$$

This completes the proof.

□

Lemma 5.4. The sequence $V^i(x(t))$ generated by Algorithm 5.1 satisfies the following iteration equation.

$$\begin{aligned}HJB(V^{i+1}(x(t))) &= HJB(V^i(x(t + T_s))) \\ &\quad - (u^i - u^{i+1})^T R(u^i - u^{i+1})\end{aligned}\tag{5.13}$$

Proof. Differentiate equation (5.5) with respect to time t along the system trajectory $\dot{x} = f(x) + g(x)u^i$

$$\begin{aligned} & [(V_x^{i+1})^T (f(x) + g(x)u^i)]|_{x(t)} \\ &= [(Q(x) + (u^i)^T R u^i)]|_{x(t+T_s)} - [(Q(x) + (u^i)^T R u^i)]|_{x(t)} \\ & \quad + [(V_x^i)^T (f(x) + g(x)u^i)]|_{x(t+T_s)} \end{aligned} \quad (5.14)$$

Rearrange (5.14) and substitute (5.6) into it,

$$\begin{aligned} & [(V_x^{i+1})^T (f(x) + g(x)u^{i+1}) + Q(x) + (u^{i+1})^T R u^{i+1}]|_{x(t)} \\ &= [(V_x^i)^T (f(x) + g(x)u^i) + Q(x) + (u^i)^T R u^i]|_{x(t+T_s)} \\ & \quad - [(u^{i+1} - u^i)^T R (u^{i+1} - u^i)]|_{x(t)} \end{aligned} \quad (5.15)$$

(5.15) is essentially (5.13) based on the definition of HJB operator in **Definition 5.1**.

This completes the proof. □

Now, three convergence modes of the VI based on-policy IRL are stated in **Theorem 5.1** to **Theorem 5.3**.

Theorem 5.1. The Policy iteration [PI] mode is decreasingly convergent.

With the initial value function V^0 satisfying $V^0 \geq V^*$, then the sequence $\{V^i\}_{i=0}^\infty$ is monotonically decreasing, i.e.,

$$V^0 \geq \dots \geq V^i \geq V^\infty = V^*. \quad (5.16)$$

Proof. Given $V^0 \geq V^*$, then $HJB(V^0) \leq 0$ is guaranteed. Otherwise **Lemma 5.2** will be violated.

Using **Lemma 3**, $HJB(V^i) \leq 0$ for all $i \geq 0$ as long as $HJB(V^0) \leq 0$ holds. Now we are trying to prove the monotonically decreasing property of $\{V^i\}_{i=0}^\infty$.

Use the definition of $HJB(V^i)$ and the condition $HJB(V^i) \leq 0$,

$$V_x^T (f + g u^i) \leq -(Q(x) + (u^i)^T R u^i) \quad (5.17a)$$

$$\dot{V}^i \leq -(Q(x) + (u^i)^T R u^i) \quad (5.17b)$$

(5.17) holds for all $x(t)$ with $t \leq t \leq t + T_s$. Integrating from $t = t$ to $t = t + T_s$,

$$V^i(x(t)) \geq \int_t^{t+T_s} (Q(x) + (u^i)^T R u^i) d\tau + V^i(x(t + T_s)) \quad (5.18)$$

Substitutes (5.5) into (5.18),

$$V^i(x(t)) \geq V^{i+1}(x(t)) \quad (5.19)$$

The monotonically decreasing property of $\{V^i\}_{i=0}^\infty$ is proved. Moreover, since V_i is lower bounded by 0 as in **Lemma 5.1**. This proves V^i with $i = \infty$ convergence to V^∞ .

$HJB(V^\infty)$ has to equal 0 otherwise $\{V^i\}_{i=0}^\infty$ will keep decreasing. Therefore $V^\infty = V^*$, which completes proof for Theorem 5.1. $\square$

Theorem 5.2. The Value iteration [VI] mode is increasingly convergent.

With the initial value function V^0 satisfying the condition $0 \leq V^0 \leq V^*$, if V^i generated by Algorithm 5.1 satisfies $HJB(V^i) \geq 0$, then the sequence $\{V^i\}_0^\infty$ is monotonically increasing, i.e.,

$$0 \leq V^0 \leq \dots \leq V^i \leq V^\infty. \quad (5.20)$$

Proof. Since the condition $HJB(V^i) \geq 0$ holds for all i , then,

$$V_x^T(f + g u^i) \geq -(Q(x) + (u^i)^T R u^i) \quad (5.21a)$$

$$\dot{V}^i \geq -(Q(x) + (u^i)^T R u^i) \quad (5.21b)$$

(5.21) holds for all $x(t)$ with $t \leq t \leq t + T_s$. Integrating (5.21) from t to $t + T_s$

$$V^i(x(t)) \leq \int_t^{t+T_s} (Q(x) + (u^i)^T R u^i) d\tau + V^i(x(t + T_s)) \quad (5.22)$$

Substituting (5.5) into (5.22) gives

$$V^i(x(t)) \leq V^{i+1}(x(t)) \quad (5.23)$$

The monotonically increasing property of $\{V^i\}$ is proved. Moreover, since V^i is bounded by 0 and V^* . This proves the monotonically convergence to V^* . This completes the proof. $\square$

Theorem 5.3. Mixed mode of convergence.

With the initial value function V^0 satisfying $0 \leq V^0$, if V^i generated by Algorithm 5.1 satisfies $HJB(V^i) \geq 0$ for all $i \leq k$, then $\{V_i\}_{i=0}^k$ has the monotonic increase property up to $i = k$. If $HJB(V^i) \leq 0$ for $i \geq k + 1$, then $\{V^i\}_{i \geq k+1}^\infty$ has the monotonic decrease property. That is,

$$V^0 \leq \dots \leq V^k \leq V^*, \quad (5.24a)$$

$$V^{k+1} \geq V^{k+2} \geq \dots \geq V^\infty = V^*. \quad (5.24b)$$

Proof. On the one hand, if $HJB(V^i) \geq 0$ for all $i \leq k$, then $\{V^i\}_{i=0}^k$ has the monotonic increase property up to $i = k$ based on **Theorem 5.2**.

On the other hand, if $HJB(V^i) \leq 0$ for $i \geq k+1$, then $\{V^i\}_{i=k+1}^\infty$ has the monotonic decrease property up to $i = \infty$ based on **Theorem 5.1**. $\square$

Lemma 5.5. With the initial value function V^0 satisfying $0 \leq V^0 \leq V^*$, the selection of T_s will determine whether $\{V^i\}_{i=0}^\infty$ converges in VI mode or mixed mode.

Proof. Given $0 \leq V^0 \leq V^*$, then $HJB(V^0) \geq 0$ is guaranteed. Otherwise **Lemma 5.3** will be violated. Using **Lemma 5.4**, (5.25) is obtained.

$$HJB(V^1) = HJB(V^0(x(t + T_s))) - (u^0 - u^1)^T R(u^0 - u^1) \quad (5.25)$$

If T_s is selected sufficiently small ($T_s \rightarrow 0$),

$$\|V^1 - V^0\| \approx 0, \quad (5.26)$$

which means $HJB(V^1) \geq 0$ since the second term in left hand side of (5.25) $(u^1 - u^0)^T R(u^1 - u^0)$ is neglectable due to small sampling time T_s and $HJB(V^1)$ will remain positive semi-definiteness. This result has been discussed in [67] for a LQR problem. That is, Algorithm 5.1 will converge in VI mode with $0 \leq V^0 \leq V^*$ under small T_s .

On the other hand, if T_s is too large, then $HJB(V^1)$ might become negative semi-definite.

Consider the special case with initial value function $V^0(x) = 0$ and assume that $u^0 = 0$ is a stabilizing control law. Apply Algorithm 5.1 with T_s approach infinite. $x(t + T_s)$ approaches zero as u^0 stabilizes the system. Then $HJB(V^1) \leq 0$ based on (5.25) and $V^1 \geq V^*$ accordingly. That is, in this special case, Algorithm 5.1 will converge in mixed mode with $0 \leq V^0 \leq V^*$ under large T_s .

In conclusion, $\{V^i\}_{i=0}^\infty$ will always converge in either VI mode or mixed mode with $0 \leq V^0 \leq V^*$ for all T_s . □

5.4 Implementation of Algorithm 5.1 Using Neural Networks

In this section, an approximation of the value function based on NNs is presented. To approximate V^i at each iteration of Algorithm 5.1. In the following analysis, the completeness property of the series is applied, and the definition is given as follows.

Definition 5.2. (See the work of Vrabie and Lewis [19])

A given linearly independent series $\{\varphi_j(x)\}_1^\infty$ has the completeness property if, for any function $F(x) \in C^1(\Omega)$, $F(0) = 0$, $F(x)$ can be represented as a linear combination of a subset of $\{\varphi_j(x)\}_{j=1}^\infty$.

We follow the actor-critic structure in this implementation. Select a sequence of activation functions $\varphi(x) = [\varphi_1(x), \dots, \varphi_{N_1}(x)]^T$, which satisfies the completeness property shown in **Definition 5.2**. N_1 is the cardinality of this selected subset. The function $V^{i+1}(x)$ at the $(i+1)$ -th iteration of the updating process is approximated by a critic NN as shown below,

$$\hat{V}^{i+1} = (W_1^{i+1})^T \varphi(x) \quad (5.27)$$

where $\hat{V}^{i+1}$ is the approximation of V^{i+1} , $W_1^{i+1} \in R^{N_1}$ is a weight vector corresponding with the proposed NN.

Given $\hat{V}^i$ estimated by (5.27), the actor NN is applied to estimate the iterative control policy, which is shown as,

$$u^{i+1}(x(t)) = -\frac{1}{2}R^{-1}g^TV_x^{i+1} = -\frac{1}{2}R^{-1}g^T\nabla_x\varphi(x)^T(W_1^{i+1}), \quad (5.28)$$

where $\nabla_x\varphi(x)$ is the derivative of $\varphi(x)$ over state x .

Theorem 5.4. The Completeness property of activation function $\varphi_1(x)$ with sufficient large length of neural networks guarantees that the convergence property in Section III will not be affected by any neural network approximation error [19] [57].

Substituting value function approximation (5.27) into equation (5.5), we can achieve,

$$\begin{aligned} & (W_1^{i+1})^T\varphi(x(t)) \\ &= \int_t^{t+T_s} (Q(x) + (u^i)^TRu^i)d\tau + (W_1^i)^T\varphi(x(t+T_s)). \end{aligned} \quad (5.29)$$

A standard least mean square regression problem is defined as follows with the real-time collected data samples $h(t_j), y(t_j)$ and unknown weight vector W_1^{i+1}

$$(W_1^{i+1})^Th(t_j) = y(t_j) \quad (5.30a)$$

$$h(t_j) = \begin{bmatrix} \varphi(x(t_j)) \end{bmatrix} \in R^{N_1} \quad (5.30b)$$

$$y(t_j) = \int_{t_j}^{t_j+T_s} (Q(x) + (u^i)^TRu^i)d\tau + (W_1^i)^T\varphi(x(t+T_s)) \quad (5.30c)$$

Note that there are totally N_1 independent unknown entries in the weight vector W_1^{i+1} . Therefore in order to estimate W_1^{i+1} with online collected data samples, at least N data samples should be collected where $N \geq N_1$ to make sure the least mean square estimation problem well-posed.

By collected $H = [h(t_1) \dots h(t_N)]$ and $Y = [y(t_1) \dots y(t_N)]^T$, the weight vector W_1^{i+1} based on least method square method is estimated as,

$$W_1^{i+1} = (HH^T)^{-1}HY. \quad (5.31)$$

5.5 Verification of Convergence Property of Algorithm 5.1

5.5.1 Linear System Verification of Convergence Property of Algorithm 5.1

In this section, we firstly apply **Algorithm 5.1** to a linear system [64] to verify and investigate the convergence properties of **Algorithm 5.1**. The linear dynamic system is given in (5.32b) with a cost function where $Q(x) = x_1^2 + 10x_2^2$ and $R = 1$

$$\dot{x} = Ax + Bu \quad (5.32a)$$

$$A = \begin{bmatrix} -1 & 0.1 \\ -0.002 & -2 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 2 \end{bmatrix} \quad (5.32b)$$

The optimal value function V^* can easily be calculated by solving an Algebraic Riccati Equation (ARE) using the information of A and B , where the optimal value function is,

$$V^* = 0.5x_1^2 + 0.0013x_1x_2 + 1.158x_2^2. \quad (5.33)$$

The value function V^{i+1} is approximated via $V^{i+1} = W_1^{i+1} [x_1^2 \ x_1x_2 \ x_2^2]^T$. **Algorithm 5.1** is applied to this linear system with various initial value function V^0 and sampling time T_s . Results are given in Figure 5.1.

Figure 5.1(a) and Figure 5.1(b) show the convergence results of V^i from the initial value function,

$$V^0 = 2x_1^2 + 2x_1x_2 + 2x_2^2$$

FIGURE 5.1: Evaluation of W_1 with different initial value and sampling time for a linear system

with sampling time 0.1s and 0.2s, respectively. These two sets of results illustrate **Theorem 5.1**, i.e., given an initial condition $V^0 \geq V^*$, PI mode convergence is guaranteed for an arbitrary T_s .

Figure 5.1(c) shows the simulation results with $W_1^0 = 0$ and $T_s = 0.1$, which illustrates the VI mode monotonically increasing property in **Theorem 5.2**. Figure 5.1(d) shows the simulation results with $W_2^0 = 0$ and $T_s = 0.2$. It shows that there is a slight overshoot $W_1(3) = 1.3$ in the first iteration. This is due to the mixed mode convergence as discussed in **Theorem 5.3**, where the sequence $\{V^i\}_{i=0}^\infty$ follows a trend of monotonically increasing property for $i \leq k$ and follows monotonically decreasing property for $i \geq k + 1$.

5.5.2 Nonlinear System Verification of Convergence Property of Algorithm 5.1

In this section, we apply **Algorithm 5.1** to a nonlinear system shown in (5.34), which has strong nonlinearities [19].

$$\dot{x}_1 = -x_1 + x_2 + 2x_2^3 \quad (5.34a)$$

$$\dot{x}_2 = f_2(x) + g_2(x_1)u \quad (5.34b)$$

where $f_2(x) = -\frac{1}{2}(x_1 + x_2) + \frac{1}{2}x_2(1 + 2x_2^2)\sin^2(x_1)$, $g_2(x_1) = \sin(x_1)$. The cost function is defined with $Q(x) = x_1^2 + x_2^2 + 2x_2^4$ and $R = 1$.

The optimal cost function and the optimal controller are shown as follows,

$$V^*(x) = \frac{1}{2}x_1^2 + x_2^2 + x_2^4, \quad (5.35a)$$

$$u^*(x) = -\sin(x_1)(x_2 + 2x_2^3). \quad (5.35b)$$

The value function V^{i+1} is approximated via

$$\hat{V}^{i+1} = (W_1^{i+1})^T [x_1^2 \quad x_1x_2 \quad x_2^2 \quad x_1^4 \quad \dots \quad x_2^4]^T$$

The simulations are conducted for different initial values of V^0 with various W^0 and sample time T_s .

Figure 5.2(a) and Figure 5.2(b) shows the convergence waveforms with V^0 at $T_s = 0.1s$ and $T_s = 0.2s$ respectively, where monotonically decreasing property of W_1 to $W_1^* = [0.5, 0, 1, 0, 0, 0, 0, 1]^T$ is observed.

$$\hat{V}^0 = (W_1^0)^T \varphi_1(x) = x_1^2 + x_2^2 + 2x_2^4 \geq V^* \quad (5.36)$$

Fig. 5.2(c) and Fig. 5.2(d) show the convergence waveform with $V^0 = 0$ at $T_s = 0.1s$ and $T_s = 0.2s$. W_1^i converges the same W_1^* in VI mode as in Fig. 5.2(c)

(a) Evaluation of W_1 with $T_s=0.1$ and with W_1^0 in (38) (b) Evaluation of W_1 with $T_s=0.2$ and with W_1^0 in (38) for a nonlinear input affine system

(c) Evaluation of W_1 with $T_s=0.1$, $W_1^0 = [0 \ 0 \ 0]^T$ (d) Evaluation of W_1 with $T_s=0.2$, $W_1^0 = [0 \ 0 \ 0]^T$

FIGURE 5.2: Evaluation of W_1 with different initial value and sampling time for a linear system

with $T_s = 0.1s$ and in mixed mode as in Fig. 5.2(d) with a larger sampling time $T_s = 0.2s$.

5.6 VI based Off-policy IRL for a Completely Unknown Nonlinear System

To further overcome the technical limitation of on-policy IRL where system input dynamic is needed, in this section, we propose an VI based off-policy IRL Algorithm (**Algorithm 5.2**) for a completely unknown input-affine nonlinear system

to overcome the technical limitation in VI based on-policy (**Algorithm 5.1**).

5.6.1 Convergence Property of Algorithm 5.2

The Algorithm 5.2 is shown as follows and we state the convergence property of this VI based off-policy IRL algorithm in **Theorem 5.5**.

Algorithm 5.2: A VI based off-policy IRL algorithm for an input-affine nonlinear system with completely unknown dynamics

---

1. **Step 1: Initialization.** Select an initial value function $V^0(x) = 0$ and a control law $u^0 = 0$.
2. **Step 2.** Use knowledge of V^i and apply u^i at the i -th iteration to the system, find V^{i+1} using the following on-policy Bellman equation.

$$\begin{aligned} V^{i+1}(x(t)) &= \int_t^{t+T_s} (Q(x(\tau)) + (u^i(\tau))^T R u^i(\tau)) d\tau + V^i(x(t + T_s)) \end{aligned} \quad (5.37)$$

where T_s is the sampling time.

3. **Step 3:** Apply a control law u to the system with. knowledge of V^{i+1} , find u^{i+1} .

$$\begin{aligned} V^{i+1}(x(t + T_r)) - V^{i+1}(x(t)) &= \int_t^{t+T_r} ((V_x^{i+1})^T f(x)) d\tau + \int_t^{t+T_r} 2(-u^{i+1})^T R(u) d\tau \end{aligned} \quad (5.38)$$

where T_r is another arbitrary sampling time.

4. **Step 4:** Let $i = i + 1$ and Go back to **Step 2** until convergence.
-

Theorem 5.5. Convergence Property of **Algorithm 5.2**.

Algorithm 5.2 is equivalent to Algorithm 5.1, therefore they share the same convergence property.

Proof. Obviously, **Step 2** of Algorithm 5.2 is the same as (5.5). Thus, the equivalence between Algorithm 5.2 and Algorithm 5.1 can be proved by showing the equivalence between (5.38) and (5.6).

Differentiating the value function V^{i+1} along the system trajectory given by $\dot{x} = f(x) + g(x)u$, where the input u is an arbitrary function.

$$\dot{V}^{i+1} = (V_x^{i+1})^T (f(x) + g(x)u) \quad (5.39)$$

Integrating equation (5.39) from t to $t + T_r$ gives (5.40), where T_r is an arbitrary sampling time in **Step 3** of Algorithm 5.2

$$\begin{aligned} & V^{i+1}(x(t + T_r)) - V^{i+1}(x(t)) \\ &= \int_t^{t+T_r} ((V_x^{i+1})^T f(x)) d\tau + \int_t^{t+T_r} 2(-u^{i+1})^T R(u) d\tau \end{aligned} \quad (5.40)$$

This proves the equivalence between Algorithm 5.1 and Algorithm 5.2. $\square$

5.6.2 Implementation of Algorithm 5.2

In this section, **Algorithm 5.2** is applied to the same input-affine nonlinear system example in Section 5.5. Due to the lack of information for input dynamic matrix $g(x)$, we can not directly derive u^{i+1} using (5.6). Instead, we solve (5.38) directly for u^{i+1} using online collected data samples. Totally three NNs are required to be introduced to estimate V_x^{i+1} , $V_x^{i+1}f(x)$ and u^{i+1} respectively, as shown below.

$$\hat{V}^{i+1} = (W_1^{i+1})^T \varphi(x) \quad (5.41a)$$

$$\hat{u}^{i+1} = (W_2^{i+1})^T \phi(x) \quad (5.41b)$$

$$\hat{V}^{i+1}f(x) = (W_3^{i+1})^T \sigma(x) \quad (5.41c)$$

where $W_1^{i+1}, W_2^{i+1}, W_3^{i+1}$ are the weight vectors and $\varphi(x), \phi(x), \sigma(x)$ are the activation functions.

5.7 VI based Off-policy IRL version 2 for Completely Unknown Nonlinear System

It is obvious that **Algorithm 5.2** solves V^{i+1} and u^{i+1} in two separate steps which may require more real-time data to be collected. Three NNs are needed in (5.41). To this end, a more efficient VI based off-policy IRL controller is proposed to further reduce the computation complexity. Firstly, the convergence of **Algorithm 5.3** is proved and the final VI based off-policy IRL is proposed in **Algorithm 5.4** based on **Algorithm 5.3**.

Firstly, it is shown that **Algorithm 5.1** is equivalent to solving a backward differential equation as in **Algorithm 5.3**.

Algorithm 5.3: VI based IRL Algorithm for an input-affine nonlinear systems based on solving differential equations backward in time

---

1. **Step 1: Initialization.** Select an initial value function $V^0(x) = 0$ and a control law $u^0 = 0$.
2. **Step 2.** Solve $V(x(t), t)$ according to the differential equation in (5.42) with boundary conditions $V(x(t + T_s), t + T_s) = V^i(x(t + T_s))$ with $u^i = -\frac{1}{2}R^{-1}g^T V_x^i$.

$$-\frac{\partial V}{\partial t} = \frac{\partial V^T}{\partial x} (f(x) + g(x)u^i) + Q(x) + (u^i)^T R u^i \quad (5.42)$$

3. **Step 3:** Update V^{i+1} using $V^{i+1}(x(t)) = V(x(t), t)$ and update u^{i+1} using

$$u^{i+1}(x(t)) = -\frac{1}{2}R^{-1}g^T V_x^{i+1} \quad (5.43)$$

4. **Step 4:** Let $i = i + 1$ and go back to **Step 2** until convergence.
-

Theorem 5.6. **Algorithm 5.3** is equivalent to **Algorithm 5.1** and therefore they share the same convergence property.

Proof. Consider the value function $V(x, t)$ and calculate its derivative with respect to time t

$$\begin{aligned} \frac{dV}{dt} &= \frac{\partial V}{\partial t} + \frac{\partial V^T}{\partial x} (f(x) + g(x)u^i) \\ \frac{dV}{dt} &= -(Q(x) + (u^i)^T R u^i) \end{aligned} \quad (5.44)$$

□

Integrating (5.44) from t to $t + T_s$

$$\begin{aligned} & V(x(t + T_s), t + T_s) - V(x, t) \\ &= - \int_t^{t+T_s} (Q(x(\tau)) + u^i(\tau) R u^i(\tau)) d\tau \end{aligned} \quad (5.45)$$

Using the definition of $V(x(t + T_s), t + T_s)$ and $V(x, t)$ obtains

$$\begin{aligned} & V^i(x(t + T_s)) - V^{i+1}(x(t)) \\ &= - \int_t^{t+T_s} (Q(x(\tau)) + u^i(\tau) R u^i(\tau)) d\tau \end{aligned} \quad (5.46)$$

Now, **Algorithm 5.4** is proposed without using knowledge of $f(x)$ as well as $g(x)$.

Algorithm 5.4: A Computational Efficient VI based off-policy IRL algorithm for an input-affine nonlinear system with completely unknown dynamics

---

1. **Step 1: Initialization.** Select an initial value function $V^0(x) = 0$ and a control law $u^0 = 0$.
2. **Step 2.** Use knowledge of V^i and u^i , apply arbitrary control law into the system, find V^{i+1} and u^{i+1} using (5.47).

$$\begin{aligned} & V^i(x(t + T_s)) - V^{i+1}(x(t)) \\ &= - \int_t^{t+T_s} (Q(x) + (u^i)^T R u^i) d\tau \\ &\quad + \int_t^{t+T_s} 2(-u^{i+1})^T R (u - u^i) d\tau \end{aligned} \quad (5.47)$$

where T_s is the sampling time.

3. **Step 3:** Let $i \leftarrow i + 1$ and Go back to **Step 2** until convergence.
-

Theorem 5.7. Under the assumption that T_s is sufficiently small, Algorithm 5.4 is equivalent to Algorithm 5.3. They share the same convergence property.

Proof. Differentiating the value functional $V(x, t)$ along the system trajectory given by $\dot{x} = f(x) + g(x)u$, where the input u is an arbitrary function.

$$\dot{V} = (V_x)^T (f(x) + g(x)u) + \frac{\partial V}{\partial t} \quad (5.48)$$

Integrating (5.48) from t to $t + T_s$ gives (5.49)

$$\begin{aligned} & V(x(t + T_s), t + T_s) - V(x, t) \\ &= \int_t^{t+T_s} \left(\frac{\partial V}{\partial t} + \frac{\partial V^T}{\partial x} (f(x) + g(x)u^i) \right) d\tau \\ &+ \int_t^{t+T_s} 2V_x^T g(x)(u - u^i) d\tau \end{aligned} \quad (5.49)$$

Using the backward Euler method with small T_s , the second term in the left hand side can be approximated in (5.50). Notice that

$$\begin{aligned} & \int_t^{t+T_s} (2V_x^T(\tau)g(\tau)(u - u^i)) d\tau \\ &= \int_t^{t+T_s} (2(V_x(x(t), t))^T g(x)(u - u^i)) d\tau \end{aligned} \quad (5.50)$$

Substituting (5.42), (5.43) and (5.50) into (5.49) gives

$$\begin{aligned} & V(x(t + T_s), t + T_s) - V(x, t) \\ &= \int_t^{t+T_s} -(Q(x) + (u^i)^T R u^i) d\tau \\ &+ \int_t^{t+T_s} 2(-u^{i+1})^T R (u - u^i) d\tau \end{aligned} \quad (5.51)$$

Using the definition of $V(x(t + T_s), t + T_s)$ and $V(x, t)$ obtains

$$\begin{aligned} & V^i(x(t + T_s)) - V^{i+1}(x(t)) \\ &= - \int_t^{t+T_s} (Q(x) + (u^i)^T R u^i) d\tau \\ &+ \int_t^{t+T_s} 2(-u^{i+1})^T R (u - u^i) d\tau \end{aligned} \quad (5.52)$$

□

5.7.1 Simulation Verification of Algorithm 5.4 using the same linear system in Section 5.5

Comparing **Algorithm 5.4** with **Algorithm 5.2**, it is obvious that additional neural networks are not needed to approximate $V_x^{i+1}f(x)$. Moreover, the value update and policy improvement are done within one single step. In this section, **Algorithm 5.4** is applied to the same linear system shown in previous section .

The value function V^{i+1} and control law u^{i+1} are approximated by

$$\hat{V}^{i+1} = (W_1^{i+1})^T [x_1^2 \quad x_1 x_2 \quad x_2^2]^T, \quad \hat{u}^{i+1} = (W_2^{i+1})^T x$$

The evaluation of W_1 and W_2 depicted in Figure 5.3 with $T_s = 0.1$ and in Figure 5.4 with $T_s = 0.2$. In both cases, starting from $W_2^0 = 0$, $\{W_2^i\}_{i=0}^\infty$ finally converges to $W_2^* = -[0.0125 \ 2.3168]^T$, which verifies **Theorem 5.7**.

FIGURE 5.3: Evaluation of W_1 and W_2 in a linear system with $T_s=0.1$ under **Algorithm 5.4**

5.7.2 Simulation Verification of Algorithm 5.4 using the same nonlinear system in Section 5.5

In this section, we apply **Algorithm 5.4** to the same input-affine nonlinear system example in Section 5.5.

FIGURE 5.4: Evaluation of W_1 and W_2 in a linear system with $T_s=0.2$ under **Algorithm 5.4**

The value function V^{i+1} and control law u^{i+1} are approximated by

$$\hat{V}^{i+1} = (W_1^{i+1})^T \varphi(x) \quad (5.53a)$$

$$\hat{u}^{i+1} = (W_2^{i+1})^T \phi(x) \quad (5.53b)$$

where W_1^{i+1} , W_2^{i+1} are the weight vectors and $\varphi(x)$, $\phi(x)$ are the activation functions. In this case study, $\varphi(x)$, $\phi(x)$ are selected as follows.

$$\varphi(x) = [x_1^2 \quad x_1 x_2 \quad x_2^2 \quad x_1^4 \quad \dots \quad x_2^4] \quad (5.54a)$$

$$\phi(x) = x_1 * [0 \quad x_1 \quad 2x_2 \quad 0 \quad 3x_1^2 x_2 \quad 2x_1^2 x_2 \quad 3x_1 x_2^2 \quad 4x_2^3] \quad (5.54b)$$

As shown in Figure 5.5, starting from $W_2^0 = 0$, $\{W_2^i\}_{i=0}^\infty$ finally converges to $W_2^* = [0 \quad 0.017 \quad -0.498 \quad 0 \quad 0.043 \quad 0 \quad 0.036 \quad -0.544]^T$, which verifies **Theorem 5.7**.

The state trajectories with the initial state condition $x_1 = 1, x_2 = -1$ under u^* via **Algorithm 5.1** and NN approximated $\hat{u}^*$ via **Algorithm 5.4** are shown in Figure 5.6. It indicates that there is a small acceptable gap between state trajectory under u^* in **Algorithm 5.1** and u^* in **Algorithm 5.4**.

FIGURE 5.5: Evaluation of W_2 with sampling time $T_s = 0.1$ in Algorithm 5.4

FIGURE 5.6: State trajectory under u^* in Algorithm 5.1 and u^* in Algorithm 5.4

5.8 Conclusion

This chapter has presented a detailed convergence analysis for VI-based IRL algorithm when solving an optimal control problem for an input-affine continuous-time nonlinear system. It has been proved that the selection of the initial value function

V^0 as well as the sample time T_s will result in different modes of convergence, i.e., a PI-mode with a monotonically decreasing convergence, a VI-mode with a monotonically increasing convergence, and a mixed mode by combining these two features. Moreover, a VI based off-policy IRL algorithm is proposed to solve the optimal control problems with completely unknown system dynamics, and its convergence has been proved to be equivalent to the modes observed in VI based on-policy IRL algorithm with small T_s . Extensive numerical simulations are provided to illustrate the feasibility and effectiveness of the proposed VI based off-policy IRL algorithm.

Chapter 6

Some On-going and Future work

6.1 Future Work Topic : Off-policy Value Iteration for H_∞ Regulation of Nonlinear System

6.1.1 Background

Consider the continuous time nonlinear system described as

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t) + k(x(t))d(t), \quad (6.1a)$$

$$x(0) = x_0, \quad (6.1b)$$

with the system state $x(t) \in \mathbf{R}^n$, the control variable $u(t) \in \mathbf{R}^m$, the external disturbance $d \in \mathbf{R}^q$ the system dynamics $f(x(t)) \in \mathbf{R}^n$, the input dynamics $g(x(t)) \in \mathbf{R}^{n \times m}$ and the disturbance matrix $k(x(t)) \in \mathbf{R}^{n \times q}$. It is assumed that $f(x)$, $g(x)$ and $k(x)$ are unknown Lipchitz functions on a compact set $\Omega \in \mathbf{R}^n$, which contains the origin such that $x = 0$ is an equilibrium point of the system. Assume that system (6.1) is controllable with at least one continuous control law u which can asymptotically stabilize the system on the compact set Ω .

The optimal performance function using H_∞ technique is defined in (6.2),

$$V(x(t)) = \min_u \max_d \int_t^\infty (Q(x(\tau)) + u(\tau)^T R u(\tau) - \gamma^2 d^T d) d\tau \quad (6.2)$$

where $Q(x)$ is a positive definite function of x , i.e., $\forall x \neq 0, Q(x) > 0$. $Q(x)$ equals 0 if and only if $x = 0$. R is a positive definite matrix $\in \mathbf{R}^{n \times n}$

A value iteration based IRL for nonlinear optimal control with partial information of the system dynamics is given in Algorithm 6.2 [58] without the assumption of initial stable control law. The convergence of Algorithm 6.1 is based on the monotone property of $\{V^i\}_0^\infty$ sequence [58].

Algorithm 6.1: A VI based on-policy IRL for H_∞ control problem with partial information of system dynamics [58] [69]

---

1. **Step 1: Initialization.** Let $i = 0$ and select an initial value function $V^0(x)$ with $V^0(0) = 0$, and the control law is shown as follows,

$$\begin{aligned} u^0 &= -\frac{1}{2} R^{-1} g^T V_x^0 \\ d^0 &= -\frac{1}{2} \gamma^{-2} k^T V_x^0 \end{aligned} \quad (6.3)$$

2. **Step2:** For each iteration i , use the knowledge V^i and the control law u^i to determine the value function V^{i+1} using (6.4).

$$V^{i+1}(x(t)) = \int_t^{t+T_s} (Q(x) + (u^i)^T R u^i - \gamma^2 (d^i)^T d^i) d\tau + V^i(x(t+T_s)) \quad (6.4)$$

where T_s is the sampling time.

3. **Step 3:** Update the control law and disturbance law at the $i + 1$ iteration u^{i+1} using,

$$\begin{aligned} u^{i+1} &= -\frac{1}{2} R^{-1} g^T V_x^{i+1} \\ d^{i+1} &= \frac{1}{2} \gamma^{-2} k^T V_x^{i+1} \end{aligned} \quad (6.5)$$

4. **Step 4:** Let $i \leftarrow i + 1$, and go to **Step 2** until convergence.
-

6.1.2 Proposed VI based IRL for H_∞ Control of a Completely Unknown System

It is obvious that VI based IRL for H_∞ control in Algorithm 6.1 still needs the knowledge $g(x), k(x)$ in Step 3 (6.5). Motivated by our work in Chapter 5, we propose VI based off policy IRL for H_∞ control of a completely unknown system in **Algorithm 6.2**.

Algorithm 6.2: A VI based off-policy IRL algorithm for H_∞ control of input-affine nonlinear system with completely unknown dynamics

---

1. **Step 1: Initialization.** Select an initial value function $V^0(x) = 0$ and a control law $u^0 = 0$.
2. **Step 2.** Use knowledge of V^i and u^i , apply arbitrary control law into the system, find V^{i+1} and u^{i+1} using (6.6).

$$\begin{aligned}
 & V^i(x(t + T_s)) - V^{i+1}(x(t)) \\
 &= - \int_t^{t+T_s} (Q(x) + (u^i)^T R u^i - \gamma^2 (d^i)^T d^i) d\tau \\
 &+ \int_t^{t+T_s} 2(-u^{i+1})^T R (u - u^i) d\tau \\
 &+ \int_t^{t+T_s} 2\gamma^2 (d^{i+1})^T (d - d^i) d\tau
 \end{aligned} \tag{6.6}$$

where T_s is the sampling time.

3. **Step 3:** Let $i \leftarrow i + 1$ and Go back to **Step 2** until convergence.
-

6.1.3 Preliminary Results

Mathematical proofs are being carried out to prove the convergence of **Algorithm 6.2**. Meanwhile, **Algorithm 6.2** is also applied to the same linearized model of a DC-DC converter in Chapter 4 given by

$$\dot{x} = Ax + Bu + Kd \tag{6.7}$$

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 416 & -83 & 0 \\ 0 & -1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} -2564 \\ 0 \\ 0 \end{bmatrix} \quad (6.8)$$

$$K = \begin{bmatrix} 0 \\ -833 \\ 0 \end{bmatrix}, \quad (6.9)$$

The control problem is formulated to find u^* to minimize the performance function and d^* to maximize performance function defined below

$$V(x(t)) = \min_u \max_d \int_t^\infty (Q(x(\tau)) + u(\tau)^T R u(\tau) - \gamma^2 d(\tau)^T d(\tau)) d\tau \quad (6.10)$$

where $Q(x) = x^T \text{diag}([0.1, 0.1, 500])x$, $R = 0.1$, $\gamma = 10$.

The control policy u^i during the iteration process is approximated using $u^i = (W^i)^T(x)$. Start from initial control policy $u = (W^0)^T x$ with $W^0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} x$ and $V^0 = 0$ and perform Algorithm 6.2. The Algorithm 6.2 converges to the same control policy $u^\infty = \begin{bmatrix} 1.4 & 0.9 & -70 \end{bmatrix} x \equiv W^T x$.

FIGURE 6.1: Evaluation of W_1 under VI IRL Algorithm 6.2 and PI IRL Algorithm 4.2

Figure 6.1 gives the hypothesis that V^i derived in **Algorithm 6.2** seems to have the monotonically increasing property, which will inspire our mathematical proof in future.

6.2 Chapter Conclusion

In this section, we consider the further development of VI based off-policy IRL for H_∞ control of a completely unknown nonlinear system. Preliminary simulation results suggest the monotonically increasing property of V^i . Mathematical proofs are being carried out to prove the convergence of **Algorithm 6.2**.

Other potential future works also include more general results in adaptive asymptotic tracking with disturbance rejection and hardware implementations of VI based off-policy IRL in electrical machines and power converter systems.

Chapter 7

Conclusion

In the first part of this thesis, IRL has been successfully applied to H_∞ control of a 2-kW electrical machine, which shows the superiority of IRL controllers over other classical controllers with the distinct advantage of learning an optimal control law without prior knowledge of the system model. Some technical limitations are observed, including large number of neural networks in H_∞ IRL control as well as the requirement of an initial stabilizing control law.

In the second part of this thesis, two main theoretical improvements are made accordingly.

1. Firstly, existing IRL based H_∞ control algorithms require the update of disturbance policy at each iteration due to the fact that the standard H_∞ control covers all disturbances (i.e. worst case disturbance). However, the pattern or characteristic of the disturbance may be available and the main goal is to develop an effective and computationally inexpensive method to address given disturbance. A modified IRL method is proposed for intermediate H_∞ control problem with no disturbance update. The main goal is to develop an effective approach to achieve disturbance attenuation for the given time dependent disturbance instead of all disturbances. Compared with the conventional off-policy IRL for standard H_∞ control, the proposed method does not need disturbance update while solving the intermediate H_∞ control problem. The number of unknown variables and the number of neural networks during learning process are further reduced and hence is more computationally efficient.

2. Most IRL works are developed based on policy iteration where an initial stable controller is needed to start the whole iteration process. A value iteration based off-policy IRL control is proposed to overcome the limitations in policy iteration based IRL where an initial stable control law is always needed. It has been proved that the selection of the initial value function V^0 as well as the sample time T_s will result in different modes of convergence, i.e., a PI-mode with a monotonically decreasing convergence, a VI-mode with a monotonically increasing convergence, and another mixed mode by combining these two features. Moreover, a VI based off-policy IRL algorithm is proposed to solve the optimal control problems with completely unknown system dynamics, and its convergence has been proved to be equivalent to the modes observed in VI based on-policy IRL algorithm with small T_s . Extensive numerical simulations are provided to illustrate the feasibility and effectiveness of the proposed VI based off-policy IRL algorithm.

List of Author's Awards, Patents, and Publications¹

1. Yu Yang, Gajanayake Chandana, Loder David, Gupta Amit, “Short-circuit Detection Methods of Solid State Power Controller with decoupled capacitor” filed with US Patent and Trademark Office, Nov. 2020
2. Gajanayake Chandana, Yu Yang, Loder David, Gupta Amit, “Fault Detection for a Solid State Power Controller ” , filed with US Patent and Trademark Office, Nov. 2020
3. Gajanayake Chandana, Yu Yang, Loder David, Gupta Amit, “Novel topology of Solid-State Power Controller with Two Mid-caps” , filed with US Patent and Trademark Office, Nov. 2020.
4. Yang Yu, Yicheng Zhang, Rong Su ”Value Iteration based Off-Policy Integral Reinforcement Learning for a Completely Unknown Continuous Time Non-linear Systems ” under review by IEEE Transactions on Neural Networks and Learning Systems.
5. Yang Yu, Zuo Shan, Xinan Zhang, Rong Su, Frank lewis ”Data-driven Intermediate H_∞ Control of DC-DC Boost Converter Using Off policy Integral Reinforcement Learning” ready to be submitted to IEEE Transactions on Control Systems Technology.
6. Yicheng Zhang, Yang Yu, Rong Su, and Jian Chen. ”Power Scheduling in More Electric Aircraft based on An Optimal Adaptive Control Strategy.” accepted by IEEE Transactions on Industrial Electronics (2019).

---

¹The superscript * indicates joint first authors

7. Yang Yu, Shuai Wang, Rong Su V. Viswanathan, S. Ramakrishna, C. J. Gajanayake, and Amit K. Gupta. "Application of Off-policy Integral Reinforcement Learning for H_∞ , Input Constrained Control of Permanent Magnet Synchronous Machine." In 2019 IEEE Applied Power Electronics Conference and Exposition (APEC), pp. 2570-2576. IEEE, 2019.
8. Yang Yu, Xiaohe Ma, Rong Su, Tseng King Jet, V. Viswanathan, Chandana Jayampathi Gajanayake, S. RamaKrishna, and A. K. Gupta. "Application of integral reinforcement learning for optimal control of a high speed flux-switching permanent magnet machine." In IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society, pp. 2702-2707. IEEE, 2016.
9. Yang Yu, Shuai Wang, Rong Su, Tseng Kingjet, V. Viswanathan, G. Chandana, S. RamaKrishna, and Amit Gupta. "Excitation control for a high speed switched reluctance generator based on polynomial approximation." In 2016 12th IEEE International Conference on Control and Automation (ICCA), pp. 293-298. IEEE, 2016.

Bibliography

- [1] Giampaolo Buticchi, Serhiy Bozhko, Marco Liserre, Patrick Wheeler, and Kamal Al-Haddad. On-board microgrids for the more electric aircraft—technology review. *IEEE Transactions on Industrial Electronics*, 66(7):5588–5599, 2018. [1](#)
- [2] Pat Wheeler and Sergei Bozhko. The more electric aircraft: Technology and challenges. *IEEE Electrification Magazine*, 2(4):6–12, 2014. [1](#)
- [3] JA Rosero, JA Ortega, E Aldabas, and LARL Romeral. Moving towards a more electric aircraft. *IEEE Aerospace and Electronic Systems Magazine*, 22(3):3–9, 2007. [1](#)
- [4] Bulent Sarlioglu and Casey T Morris. More electric aircraft: Review, challenges, and opportunities for commercial transport aircraft. *IEEE transactions on Transportation Electrification*, 1(1):54–64, 2015. [1](#)
- [5] AA AbdElhafez and AJ Forsyth. A review of more-electric aircraft. In *ASAT-13, Paper No. ASAT-13-EP-01*, 2009. [1](#)
- [6] airbus. <https://www.airbus.com/innovation/zero-emission/electric-flight/efan-x.html>, 2020. [2](#)
- [7] Sandro Gunter, Giampaolo Buticchi, Giovanni De Carne, Chunyang Gu, Marco Liserre, He Zhang, and Chris Gerada. Load Control for the DC Electrical Power Distribution System of the More Electric Aircraft. *IEEE Transactions on Power Electronics*, 34(4):3937–3947, 2019. [2](#)
- [8] Wikipedia. <https://en.wikipedia.org/wiki/jet-engine>. [2](#)
- [9] Ellis Fh Chong, Shanmukha Ramakrishna, Vaiyapuri Viswanathan, and Shuai Wang. Segmented electrical machine, March 14 2019. US Patent App. 16/124,240. [3](#)

- [10] Jmag. <https://www.jmag-international.com/products/jmag-rt>, 2015. 3, 4
- [11] Paul J Werbos, WT Miller, and RS Sutton. A menu of designs for reinforcement learning over time. *Neural networks for control*, pages 67–95, 1990. 3
- [12] Dimitri P Bertsekas and John N Tsitsiklis. Neuro-dynamic programming: an overview. In *Proceedings of 1995 34th IEEE Conference on Decision and Control*, volume 1, pages 560–564. IEEE, 1995. 3
- [13] Paul Werbos. Approximate dynamic programming for realtime control and neural modelling. *Handbook of intelligent control: neural, fuzzy and adaptive approaches*, pages 493–525, 1992. 3
- [14] Warren B Powell. *Approximate Dynamic Programming: Solving the curses of dimensionality*, volume 703. John Wiley & Sons, 2007. 3, 65
- [15] Asma Al-Tamimi, Frank L Lewis, and Murad Abu-Khalaf. Discrete-time nonlinear hjb solution using approximate dynamic programming: Convergence proof. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 38(4):943–949, 2008. 3
- [16] Draguna Vrabie, O Pastravanu, Murad Abu-Khalaf, and Frank L Lewis. Adaptive optimal control for continuous-time linear systems based on policy iteration. *Automatica*, 45(2):477–484, 2009. 3, 7, 41, 65
- [17] Danil V Prokhorov and Donald C Wunsch. Adaptive critic designs. *IEEE transactions on Neural Networks*, 8(5):997–1007, 1997. 4
- [18] Yu Jiang and Zhong-Ping Jiang. Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics. *Automatica*, 48(10):2699–2704, 2012. 7, 20, 65
- [19] Draguna Vrabie and Frank Lewis. Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems. *Neural Networks*, 22(3):237–246, 2009. 8, 11, 41, 65, 73, 74, 77
- [20] F. L. Lewis and D. Vrabie. Reinforcement learning and adaptive dynamic programming for feedback control. *IEEE Circuits and Systems Magazine*, 9(3):32–50, Third 2009. 8, 65

- [21] Murad Abu-Khalaf and Frank L Lewis. Nearly optimal control laws for non-linear systems with saturating actuators using a neural network hjb approach. *Automatica*, 41(5):779–791, 2005. [8](#)
- [22] Randal W Beard, George N Saridis, and John T Wen. Galerkin approximations of the generalized hamilton-jacobi-bellman equation. *Automatica*, 33(12):2159–2177, 1997. [8](#), [10](#)
- [23] Jae Young Lee, Jin Bae Park, and Yoon Ho Choi. Integral q-learning and explorized policy iteration for adaptive optimal control of continuous-time linear systems. *Automatica*, 48(11):2850–2859, 2012. [11](#)
- [24] Yu Jiang and Zhong-Ping Jiang. Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics. *Automatica*, 48(10):2699 – 2704, 2012. [11](#)
- [25] Y. Jiang and Z. Jiang. Robust adaptive dynamic programming and feedback stabilization of nonlinear systems. *IEEE Transactions on Neural Networks and Learning Systems*, 25(5):882–893, 2014. [12](#)
- [26] Y. Jiang and Z. Jiang. Global adaptive dynamic programming for continuous-time nonlinear systems. *IEEE Transactions on Automatic Control*, 60(11):2917–2929, 2015. [12](#)
- [27] Biao Luo, Huai-Ning Wu, and Tingwen Huang. Off-policy reinforcement learning for control design. *IEEE transactions on cybernetics*, 45(1):65–76, 2015. [12](#)
- [28] Geyang Xiao, Huaguang Zhang, Yanhong Luo, and He Jiang. Data-driven optimal tracking control for a class of affine non-linear continuous-time systems with completely unknown dynamics. *IET Control Theory & Applications*, 10(6):700–710, 2016. [13](#), [14](#)
- [29] H. Modares, F. L. Lewis, and Z. P. Jiang. h_∞ tracking control of completely unknown continuous-time systems via off-policy reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems*, 26(10):2550–2562, Oct 2015. [13](#), [24](#), [26](#), [47](#), [48](#)
- [30] Hamidreza Modares and Frank L Lewis. Linear quadratic tracking control of partially-unknown continuous-time systems using reinforcement learning. *IEEE Transactions on Automatic Control*, 59(11):3051–3056, 2014. [15](#)

- [31] W. Gao and Z. Jiang. Adaptive dynamic programming and adaptive optimal output regulation of linear systems. *IEEE Transactions on Automatic Control*, 61(12):4164–4169, 2016. [16](#)
- [32] W. Gao and Z. Jiang. Learning-based adaptive optimal tracking control of strict-feedback nonlinear systems. *IEEE Transactions on Neural Networks and Learning Systems*, 29(6):2614–2624, 2018. [16](#)
- [33] Jie Huang. *Nonlinear output regulation: theory and applications*. SIAM, 2004. [17](#)
- [34] Marcin Morawiec. The adaptive backstepping control of permanent magnet synchronous motor supplied by current source inverter. *IEEE Transactions on Industrial Informatics*, 9(2):1047–1055, 2013. [19](#)
- [35] Matthias Preindl and Silverio Bolognani. Model predictive direct speed control with finite control set of pmsm drive systems. *IEEE Transactions on Power Electronics*, 28(2):1007–1015, 2013. [19](#)
- [36] Han Ho Choi, Nga Thi-Thuy Vu, and Jin-Woo Jung. Digital implementation of an adaptive speed regulator for a pmsm. *IEEE Transactions on Power Electronics*, 26(1):3–8, 2011. [19](#)
- [37] M Nasir Uddin and Md Muminul Islam Chy. Online parameter-estimation-based speed control of pm ac motor drive in flux-weakening region. *IEEE Transactions on Industry Applications*, 44(5):1486–1494, 2008. [19](#)
- [38] Murat Karabacak and H Ibrahim Eskikurt. Speed and current regulation of a permanent magnet synchronous motor via nonlinear and adaptive backstepping control. *Mathematical and Computer Modelling*, 53(9):2015–2030, 2011. [19](#)
- [39] Seok-Kyoon Kim. Robust adaptive speed regulator with self-tuning law for surfaced-mounted permanent magnet synchronous motor. *Control Engineering Practice*, 61:55–71, 2017. [19](#), [34](#)
- [40] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, and F.L. Lewis. Adaptive optimal control for continuous-time linear systems based on policy iteration. *Automatica*, 45(2):477 – 484, 2009. [20](#), [41](#)

- [41] Lal Bahadur Prasad, Hari Om Gupta, and Barjeev Tyagi. Application of policy iteration technique based adaptive optimal control design for automatic voltage regulator of power system. *International Journal of Electrical Power & Energy Systems*, 63:940–949, 2014. [20](#), [41](#)
- [42] Hiroyuki Suda and Masaki Yamakita. On adaptive optimal control for hybrid systems. In *Control Applications (CCA), 2013 IEEE International Conference on*, pages 859–864. IEEE, 2013. [20](#), [42](#)
- [43] Anake Pomprapa, Steffen Leonhardt, and Berno JE Misgeld. Optimal learning control of oxygen saturation using a policy iteration algorithm and a proof-of-concept in an interconnecting three-tank system. *Control Engineering Practice*, 59:194–203, 2017. [20](#), [42](#)
- [44] Jilong Zhao, Mingyao Lin, Long Jin, and Wei Zhang. Vector control study of axial field flux-switching permanent magnet machine. In *Electrical Machines and Systems (ICEMS), 2014 17th International Conference on*, pages 1091–1095. IEEE, 2014. [21](#)
- [45] R. Cai, R. Zheng, M. Liu, and M. Li. Robust control of pmsm using geometric model reduction and μ -synthesis. *IEEE Transactions on Industrial Electronics*, 65(1):498–509, 2018. [22](#)
- [46] Z. Deng and X. Nian. Robust control of permanent magnet synchronous motors. *IEEE/CAA Journal of Automatica Sinica*, 2(2):143–150, 2015. [22](#)
- [47] Hamidreza Modares, Frank L. Lewis, and Mohammad-Bagher Naghibi-Sistani. Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems. *Automatica*, 50(1):193 – 202, 2014. [24](#)
- [48] Biao Luo, Huai-Ning Wu, Tingwen Huang, and Derong Liu. Reinforcement learning solution for hjb equation arising in constrained optimal control problem. *Neural Networks*, 71:150 – 158, 2015. [25](#)
- [49] Hamidreza Modares, Frank L Lewis, and Zhong-Ping Jiang. h_∞ tracking control of completely unknown continuous-time systems via off-policy reinforcement learning. *IEEE transactions on neural networks and learning systems*, 26(10):2550–2562, 2015. [26](#)

- [50] Yue Fu and Tianyou Chai. Online solution of two-player zero-sum games for continuous-time nonlinear systems with completely unknown dynamics. *IEEE transactions on neural networks and learning systems*, 27(12):2577–2587, 2015. [26](#)
- [51] Francisco Huerta, Daniel Pizarro, Santiago Cobreces, Francisco J Rodriguez, Carlos Giron, and Ana Rodriguez. Lqg servo controller for the current control of *lcl* grid-connected voltage-source converters. *IEEE Transactions on Industrial Electronics*, 59(11):4272–4284, 2012. [28](#), [41](#), [55](#)
- [52] James Larminie, Andrew Dicks, and Maurice S McDonald. *Fuel cell systems explained*, volume 2. J. Wiley Chichester, UK, 2003. [40](#)
- [53] H. El Fadil and F. Giri. Robust nonlinear adaptive control of multiphase synchronous buck power converters. *Control Engineering Practice*, 17(11):1245 – 1254, 2009. [41](#)
- [54] A. G. Beccuti, S. Mariethoz, S. Cliquennois, S. Wang, and M. Morari. Explicit model predictive control of dc–dc switched-mode power supplies with extended kalman filtering. *IEEE Transactions on Industrial Electronics*, 56(6):1864–1874, June 2009. [41](#)
- [55] Y. Xie, R. Ghaemi, J. Sun, and J. S. Freudenberg. Implicit model predictive control of a full bridge dc–dc converter. *IEEE Transactions on Power Electronics*, 24(12):2704–2713, Dec 2009. [41](#)
- [56] Frank L Lewis, Draguna Vrabie, and Vassilis L Syrmos. *Optimal control*. John Wiley & Sons, 2012. [41](#), [48](#)
- [57] Yongliang Yang, Liming Wang, Hamidreza Modares, Dawei Ding, Yixin Yin, and Donald Wunsch. Data-driven integral reinforcement learning for continuous-time non-zero-sum games. *IEEE Access*, 7:82901–82912, 2019. [41](#), [74](#)
- [58] Huaguang Zhang, Geyang Xiao, Yang Liu, and Lei Liu. Value iteration-based h infinity controller design for continuous-time nonlinear systems subject to input constraints. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2018. [41](#), [65](#), [89](#)

- [59] Huaguang Zhang, Xiaohong Cui, Yanhong Luo, and He Jiang. Finite-horizon h infinity tracking control for unknown nonlinear systems with saturating actuators. *IEEE transactions on neural networks and learning systems*, 29(4):1200–1212, 2017. [41](#)
- [60] C. Olalla, I. Queinnec, R. Leyva, and A. El Aroudi. Optimal state-feedback control of bilinear dc–dc converters with guaranteed regions of stability. *IEEE Transactions on Industrial Electronics*, 59(10):3868–3880, Oct 2012. [55](#)
- [61] Richard S Sutton and Andrew G Barto. Reinforcement learning: An introduction. 2011. [65](#)
- [62] J. J. Murray, C. J. Cox, G. G. Lendaris, and R. Saeks. Adaptive dynamic programming. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 32(2):140–153, May 2002. [65](#)
- [63] Tao Bian, Yu Jiang, and Zhong-Ping Jiang. Adaptive dynamic programming and optimal control of nonlinear nonaffine systems. *Automatica*, 50(10):2624–2632, 2014. [65](#)
- [64] J. Y. Lee, J. B. Park, and Y. H. Choi. On integral value iteration for continuous-time linear systems. In *2013 American Control Conference*, pages 4215–4220, June 2013. [65](#), [66](#), [75](#)
- [65] T. Bian and Z. Jiang. Value iteration, adaptive dynamic programming, and optimal control of nonlinear systems. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 3375–3380, Dec 2016. [65](#), [66](#)
- [66] Hanguang Su, Huaguang Zhang, Kun Zhang, and Wenzhong Gao. Online reinforcement learning for a class of partially unknown continuous-time nonlinear systems via value iteration. *Optimal Control Applications and Methods*, 39(2):1011–1028, 2018. [65](#), [66](#)
- [67] D. Vrabie, M. Abu-Khalaf, F. L. Lewis, and Y. Wang. Continuous-time adp for linear systems with partially unknown dynamics. In *2007 IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning*, pages 247–253, April 2007. [65](#), [67](#), [68](#), [72](#)
- [68] Jae Young Lee, Jin Bae Park, and Yoon Ho Choi. On integral generalized policy iteration for continuous-time linear quadratic regulations. *Automatica*, 50(2):475–489, 2014. [65](#)

-
- [69] Geyang Xiao, Huaguang Zhang, Kun Zhang, and Yinlei Wen. Value iteration based integral reinforcement learning approach for h controller design of continuous-time nonlinear systems. *Neurocomputing*, 285:51 – 59, 2018. [65](#), [89](#)
- [70] Tao Bian and Zhong-Ping Jiang. Value iteration and adaptive dynamic programming for data-driven adaptive optimal control design. *Automatica*, 71:348 – 360, 2016. [65](#)