Adversarial robustness of deep reinforcement learning

Qu, Xinghua

2021

Qu, X. (2021). Adversarial robustness of deep reinforcement learning. Doctoral thesis, Nanyang Technological University, Singapore. <https://hdl.handle.net/10356/154587>

<https://hdl.handle.net/10356/154587>

<https://doi.org/10.32657/10356/154587>

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

Downloaded on 26 Apr 2023 18:06:23 SGT

**NANYANG
TECHNOLOGICAL
UNIVERSITY**

SINGAPORE

ADVERSARIAL ROBUSTNESS OF DEEP REINFORCEMENT LEARNING

XINGHUA QU

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

2021

ADVERSARIAL ROBUSTNESS OF DEEP REINFORCEMENT LEARNING

XINGHUA QU

School of Computer Science and Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment 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.

28 Dec 2021

Date

linghua gu

Xinghua Qu

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.

28 Dec 2021

Date

NTU NTU NTU NTU NTU NTU NTU NTU
NTU NTU NTU NTU NTU NTU NTU NTU
NTU NTU NTU NTU NTU NTU NTU NTU
NTU NTU NTU NTU NTU NTU NTU NTU

Yew-Soon Ong

Authorship Attribution Statement

This thesis contains material from 2 published journal papers and 1 submitted paper that is currently under review. The authorship attributions of each paper is illustrated below.

Chapter 3 is published as Xinghua Qu, Zhu Sun, Yew-Soon Ong, Abhishek Gupta, and Pengfei Wei, “Minimalistic attacks: How little it takes to fool deep reinforcement learning policies”, IEEE Transactions on Cognitive and Developmental Systems, 1-12 (2020). DOI: 10.1109/TCDS.2020.2974509.

The contributions of the coauthors are as follows:

- Prof Yew-Soon Ong and Dr Abhishek Gupta provided instructive comments on the idea of the work.
- I co-designed the study with Dr Zhu Sun and I performed all the laboratory work. I also analyzed the data and the results.
- I prepared the manuscript draft. The manuscript was revised by Dr Abhishek Gupta, Dr. Zhu Sun, Prof Yew-Soon Ong and Dr. Pengfei Wei.

Chapter 4 is published as Xinghua Qu, Yew-Soon Ong, Abhishek Gupta, “Frame-Correlation Transfers Trigger Economical Attacks on Deep Reinforcement Learning Policies”, IEEE Transactions on Cybernetics, 1-14 (2021). DOI: 10.1109/TCYB.2020.3041265.

The contributions of the coauthors are as follows:

- Prof Yew-Soon Ong and Dr Abhishek Gupta provided instructive comments on the idea of the work.
- I co-designed the study with Dr Abhishek Gupta and I performed all the laboratory work. I also analyzed the data and the results.

- I prepared the manuscript draft. The manuscript was revised by Prof Yew-Soon Ong and Dr. Abhishek Gupta.

Chapter 5 is submitted as Xinghua Qu, Yew-Soon Ong, Abhishek Gupta, Zhu Sun, “Adversarial Agnostic Robust Deep Reinforcement Learning”, Transaction on Neural Networks and Learning Systems. (2021).

The contributions of the coauthors are as follows:

- Prof Yew-Soon Ong and Dr Abhishek Gupta provided instructive comments on the idea of the work.
- I designed the study and performed all the laboratory work. I also analyzed the data and the results.
- I prepared the manuscript draft. The manuscript was revised by Dr. Abhishek Gupta and Dr. Zhu Sun.

28 Dec 2021

Date

Xinghua Qu

Xinghua Qu

Acknowledgments

First and foremost, I would like to express my sincere gratitude and respect to my supervisor Prof. Ong Yew-Soon, not just for the great encouragement and help on my research, but also the shared wisdom about family and life from him. I also would like to greatly thank my co-supervisor Dr. Abhishek Gupta for his continuous support, insightful discussions and critical thinking. What I have learned from them will benefit my entire career and life.

Secondly, I wish to thank my collaborators: Dr. Pengfei Wei, Dr. Zhu Sun, Prof. Yaqing Hou and Prof. Xiaobo Shen for the discussion on research idea and paper writing. I also thank my Thesis Advisory Committee (TAC) members (i.e., Prof Mahardhika Pratama and Dr. Chi Keong Goh) for their insightful comments and suggestions on my research. Thanks Prof Jie Zhang for the help and support. On that note, I thank all my group members in NTU, including Feng Liang, Yaqing Hou, Kavitesh Kumar Bali, Bingshui Da, Ray Lim, Wesley Tan, Alvin Chan, Bill Pung for the created happy environment. I also thank the research fellows and visiting scholars in our group, including Harold Chan, Zhenkun Wang, Haitao Liu, Zhengyang Li, Wenzhe Duan, Hao Li, Xiaobo Shen, Xingguang Peng, Tiantian He, Zhendong Guo, Wu Lin, Yu Xie, Xiaobin Han, Lu Bai, Lei Zhou, Zhefeng Chen for the joyful time we had together. In particular, I wish to thank Prof Bo Liu from the academy of mathematics and systems science in Chinese Academy of Science for his encouragement and help at the beginning of my Phd study. Many thanks all my friends in NTU for the happiness they brought to me. They include Tianbo Li, Xin Ju, Xiaohu Wu, Yuqing Zhu, Yunfei Li, Jianping Hu, Qian Chen, Qing Cai, Haiyan Yin, Shuo Chen, Xiaoming Li, Qiang Zhou, Wei Qiu. Thanks Kesavan Asaithambi for the technique support in the computational intelligence lab.

Most importantly, I would like to sincerely thank all my families, including my wife, my parents, my sister, and my nephew, for their constant support, encouragement and love all the time. I love you all.

Contents

Statement of Originality	i
Supervisor Declaration Statement	iii
Authorship Attribution Statement	v
Acknowledgements	vii
List of Figures	xiii
List of Tables	xvi
Abstract	1
1 Introduction	5
2 Background and Related Works	11
2.1 Deep Reinforcement Learning	11
2.2 Adversarial Attacks in DRL	14
2.3 Adversarial Attacks Transfer	17
2.4 Adversarial Robustness Improvements in DRL	19
2.5 Summary	21
3 Minimalistic Attacks: How Little It Takes To Fool Deep Reinforcement Learning	23
3.1 Black-box Policy Access	26
3.2 Fractional-State Adversary	29
3.3 Tactically-Chanced Attack	31
3.4 Experiments and Analysis	34
3.4.1 Experiment Setup	36
3.4.2 Results for Fractional-State Adversary	37

3.4.3	Results for Tactically-Chanced Attack	37
3.4.4	Attack Initialization Analysis	41
3.5	Summary	44
4	Frame-Correlation Transfers Trigger Economical Attacks on Deep Reinforcement Learning Policies	45
4.1	Motivations	47
4.1.1	Derivative-Free Attack Generation	47
4.1.2	Knowledge Transfer in Optimization	48
4.2	Frame-Correlation Transfer (FCT)	50
4.2.1	Anterior Case Transfer (AC-T)	51
4.2.2	Random Projection based Transfer (RP-T)	52
4.2.3	Principle Components based Transfer (PC-T)	55
4.2.4	Increasing Computational Complexity of the Three FCTs	57
4.3	Experiments	59
4.3.1	Experiment Setup	59
4.3.2	Convergence Trends Under Frame-Correlation Transfer	60
4.3.3	Efficiency Analysis of Frame-Correlation Transfers	64
4.3.4	Computation Time Analysis	65
4.4	Summary	67
5	Adversary Agnostic Robust Deep Reinforcement Learning	69
5.1	Preliminaries	71
5.2	Adversarial Robustness without Adversaries in DRL	73
5.3	Adversary Agnostic Policy Distillation (A2PD)	78
5.3.1	Prescription Gap Maximization (PGM)	78
5.3.2	Jacobian Regularization (JR)	79
5.4	Theoretical Analysis	80
5.5	Experiments	82
5.5.1	Experimental Setup	82
5.5.2	Robustness Evaluation of Policy π_{θ^S}	87
5.5.3	Robustness Evaluation of Policy π_{θ}^S on Minimalistic Attacks	90

5.5.4	Robustness Comparison between Teacher Policy π_{θ^T} and Student Policy π_{θ^S}	92
5.5.5	Analysis on Hyper-parameter β	93
5.5.6	Analysis on Hyperparameter η in $\mathcal{L}_{pgm}(\theta^S)$	96
5.5.7	Analysis on Prescription Gap	99
5.5.8	Analysis on Computational Efficiency	99
5.6	Summary	101
6	Conclusions and Future Works	103
6.1	Conclusions	103
6.2	Future Works	105
6.2.1	Limited Policy Query Based Adversarial Attack	105
6.2.2	Theoretical Guaranteed Adversarial Defense	106
6.2.3	Robustness Certificate of DRL	107
6.2.4	General Framework for Discrete and Continuous RL tasks	108
	List of Publications	109
	List of Submissions	111
	References	113

List of Figures

2.1	Markov Decision Process in RL.	12
2.2	The taxonomy of studies in adversarial robustness of deep reinforcement learning	21
3.1	The visualization of the single pixel attack on Breakout.	30
3.2	Full-state adversary versus the FSA	31
3.3	The attack uncertainty ζ_t on Pong.	33
3.4	The six representative Atari games [9] considered in the experiments . . .	35
3.5	The neural network structure for playing Atari games	36
3.6	The relationship between attacked frames and TCA threshold ζ^* , where the line and shaded area illustrate the mean and standard deviation of 30 independent runs respectively.	38
3.7	The relationship between attacked frames and total frames, where the line and shaded area illustrate the mean and standard deviation of 30 independent runs respectively.	39
3.8	The results of adversarial attack with different FSA size n (i.e. number of attacked pixels), where the line and shaded area illustrate the mean and standard deviation of 30 independent runs respectively.	40
3.9	The results of adversarial attack with different TCA threshold ζ^* , where the line and shaded area illustrate the mean and standard deviation of 30 independent runs respectively.	41
3.10	The impact of attacked frames for accumulated reward on Breakout . . .	42
3.11	The comparison between random initialization (GA-RI) and warm start initialization (GA-WSI), where the line and shaded area illustrate the mean and standard deviation of 10 independent runs respectively.	43

4.1	The flowchart of sequential transfer in optimization.	49
4.2	A schematic for retrieving pre-computed adversarial perturbations for transfer across similar frames. A low-dimensional feature extraction can enable efficient source retrieval.	50
4.3	The anterior case transfer (AC-T).	51
4.4	The random projection based transfer (RP-T).	54
4.5	The principle components based transfer (PC-T).	55
4.6	The comparison of convergence trends of: 1) NO-T (no transfer), 2) AC-T (anterior case transfer), 3) RP-T (random projection based transfer), and 4) PC-T (principal components based transfer).	62
4.7	The comparison of policy queries needed to generate successful untargeted attacks using different types of frame-correlation transfers. Note that <i>NO-T</i> [70] represents the derivative free generation of adversarial attacks but without transfer.	63
5.1	Policy distillation in reinforcement learning	72
5.2	The visualization of the initial state $\mathbf{s}_t$ of the five selected Atari games. .	84
5.3	The adversarial robustness comparison between teacher policy π_{θ^T} and student policy π_{θ^S}	89
5.4	The robustness evaluation on minimalistic attacks. The reward is evaluated for 50 episodes to calculate the mean (i.e., the height of the histogram) and standard deviation (i.e., the error bar on each histogram).	91
5.5	The analysis on hyperparameter β for Bank Heist, Boxing and Freeway. With each β , the distilled policy is evaluated for 50 episodes to calculate the mean (i.e., the height of the histogram) and standard deviation (i.e., the error bar on each histogram).	94
5.6	The analysis on hyperparameter β for Pong and Road Runner. With each β , the distilled policy is evaluated for 50 episodes to calculate the mean (i.e., the height of the histogram) and standard deviation (i.e., the error bar on each histogram).	95
5.7	The convergence of A2PD training with different β	97
5.8	The convergence of A2PD training with different η in (Eq. 5.12).	98

5.9	Comparison of prescription gap between teacher policy π_{θ^T} and student policy π_{θ^S}	100
-----	---	-----

List of Tables

4.1	Comparison of computational complexities of FCTs.	58
4.2	The structure of the policy network.	59
4.3	Computation time comparison on generating adversarial perturbations with different methods. FGSM is the fast gradient sign method [39]; PGD-4, PGD-10 and PGD-20 represent the projected gradient descent with iteration number being 4, 10 and 20 respectively; CW indicates the Carlini & Wagner attack [11]; NO-T represents the minimalistic attack [70]. Note that, the value in the table is calculated as the averaged time for generating adversarial perturbations on 200 frames. The smallest and second smallest values are highlighted by bold underline and bold, respectively. The last row indicates the average time across the six tested games.	66
5.1	Hyper-parameters settings for training teacher policy π_{θ^T}	83
5.2	Hyper-parameters settings for our proposed policy distillation of π_{θ^S} . . .	84
5.3	The comparison of averaged accumulated reward. In evaluating A2PD, the averaged accumulated rewards is over 50 episodes. Note that, NA in table represents the evaluation is not reported in previous studies. Natural reward indicates the reward obtained without any attack, which corresponds to the $\mathcal{R}_B$ in Eq. 5.20. The results of vanilla DQN and <i>DQN Adv. Training</i> [4] are imported from [99].	86
5.4	The comparison of relative robustness. The relative robustness value $\mathcal{M}(\mathcal{R}_\delta, \mathcal{R}_{w/o \delta})$ is calculated based on Eq. 5.20), where $\mathcal{R}_\delta$ is the reward under PGD ($K = 10, \epsilon = 1/255$) that is shaded with light blue in Table 5.3, $\mathcal{R}_{w/o \delta}$ is the corresponding natural reward without attack. The largest value of $\mathcal{M}(\mathcal{R}_\delta, \mathcal{R}_{w/o \delta})$ is highlighted with bold blue text. . . .	88

5.5	Computation time comparison on five games. The value is the average over 50 epochs. The unit of time is second.	101
-----	--	-----

Abstract

Over the past decades, the advancements in deep reinforcement learning (DRL) have demonstrated that deep neural network (DNN) policies can be trained to prescribe near-optimal actions in many complex tasks. Unfortunately, DNN policies are shown to be vulnerable to adversarial perturbations in the input states, which creates obstacles for the real-world deployments of RL agents, especially on those security-sensitive tasks. Different adversarial attacks to understand the vulnerability and corresponding defense approaches to resist against attacks have been proposed. Although some achievements and interesting findings have been observed, existing adversarial attacks are deemed to be less realistic due to the extensive assumptions utilized, such as white-box policy access and full-state adversary setting. Moreover, existing adversarial defense approaches are largely built on the adversarial training, an adversary dependent defense that is deemed to be less realistic in the wild. Given the research gaps, investigating more realistic adversarial robustness evaluation procedures (i.e., through adversarial attacks) and accordingly designing robust policies have been becoming significant but under developed topics in DRL.

In this dissertation, **firstly**, we propose a minimalistic attack in Chapter 3 by taking a more restrictive view towards adversary generation - with the goal of unveiling the limits of a DRL model's vulnerability. To this end, we define **three key settings**: (1) *black-box policy access*: where the attacker only has access to the input (state) and output (action probability) of an RL policy; (2) *fractional-state adversary*: where only several pixels

are perturbed, with the extreme case being a single-pixel adversary; and (3) *tactically-chanced attack*: where only significant frames are tactically chosen to be attacked. We formulate the adversarial attack by accommodating the three key settings, and explore their potency on six Atari games by examining four fully trained state-of-the-art policies. In Breakout, for example, we surprisingly find that: (i) all policies showcase significant performance degradation by merely modifying 0.01% of the input state, and (ii) the policy trained by DQN is totally deceived by perturbing only 1% frames.

Secondly, considering the computational complexity of the minimalistic attacks (in Chapter 3) caused by treating every frame in isolation, Chapter 4 showcases the first study of how transferability across frames could be exploited for boosting the creation of minimal yet powerful attacks in image-based RL. In doing so, we introduce three types of frame-correlation transfers (i.e., anterior case transfer, random projection based transfer, and principal components based transfer) with varying degrees of computational complexity in generating adversaries via a genetic algorithm. We empirically demonstrate the trade-off between the complexity and potency of the transfer mechanism by exploring four fully trained state-of-the-art policies on six Atari games. Our frame-correlation transfers dramatically speed up the attack generation compared to existing methods, often significantly reducing the required computation time; thus, shedding light on the real threat of real-time attacks in RL.

Last but not the least, to alleviate the vulnerability issue of DRL, in Chapter 5, we propose an adversary agnostic defense approach in order to increase the robustness of existed DRL policies. Particularly, to increase the robustness of DRL policies, previous approaches assume that the knowledge of adversaries can be added into the training process to achieve the corresponding generalization ability on these perturbed observations. However, such an assumption not only makes the robustness improvement more expensive, but may also leave a model less effective to other kinds of attacks in the wild. In

contrast, we propose an *adversary agnostic* robust DRL paradigm that does not require learning from adversaries. To this end, we first theoretically derive that robustness could indeed be achieved independently of the adversaries based on a policy distillation setting. Motivated by this finding, we propose a new policy distillation loss with two terms: 1) a prescription gap maximization loss aiming at simultaneously maximizing the likelihood of the action selected by the teacher policy and the entropy over the remaining actions; 2) a corresponding Jacobian regularization loss that minimizes the magnitude of gradient with respect to the input state. The theoretical analysis shows that our distillation loss guarantees to increase the prescription gap and the adversarial robustness. Furthermore, experiments on five Atari games firmly verify the superiority of our approach in terms of boosting adversarial robustness compared to other state-of-the-art methods.

Most importantly, we hope this dissertation will provide a useful starting point for both evaluation and improvement of adversarial robustness in DRL.

Keywords: *Deep Reinforcement Learning, Adversarial Attack, Adversarial Defense, Policy Distillation.*

Chapter 1

Introduction

The advancements in deep reinforcement learning (DRL ¹) have demonstrated that deep neural networks (DNNs) as powerful function approximators can be trained to prescribe near-optimal actions on many complex tasks (e.g., Atari games [56], robotics control [46] and motor control [87]). Although remarkable achievements have been documented, many studies on adversarial attacks [39, 47, 71, 94] have shown that DRL policies are vulnerable to adversarial perturbations in state observations. This inspires the studies on both the **evaluation** and **improvement** of DRL’s adversarial robustness.

In evaluating the adversarial robustness in DRL, various adversarial attack approaches [39, 47, 98] have been proposed. These approaches have successfully shown many interesting findings about the vulnerability of DRL policies when human imperceptible input perturbations exist. For instance, Huang et al. [39] indicate that a single step fast gradient sign method across a decision sequence could significantly decrease the accumulated reward. Lin et al. [47] designed a strategically timed attack to determine the when should an attack appear, while the basic attack strategy is based on the existing CW attack [11]. Chen et al. [14] summarized popular adversarial attack methods in RL based on white-box assumption. Although the effectiveness has shown, existing works towards adversary generation are found to make some common simplifying assumptions,

¹In this dissertation, DRL and RL are interchangeably used.

namely (1) white-box policy access: where the adversarial examples are computed by back-propagating targeted changes in the prescribed action distribution through known neural network architectures and weights, (2) full-state adversary: where the adversary may perturb almost all pixels of an image representing an input state, and (3) fully-chanced attack: where the attacker strikes the policy at every time step (e.g., at every frame of a video game).

Given that most prior works analyze the effects of perturbing every pixel of every frame assuming white-box policy access, we propose to take a more restrained view towards adversary generation - with the goal of exploring the limits of a DNN model's vulnerability in RL. Hence, Chapter 3 focuses on *minimalistic attacks* by only considering adversarial examples that perturb limited number of pixels in selected frames, and under restricted black-box policy access. In other words, we intend to unveil how little it really takes to successfully fool state-of-the-art RL policies. In doing so, we define **three key settings**: (1) *black-box policy access*: where the attacker only has access to the input (state) and output (action probability) of an RL policy; (2) *fractional-state adversary (FSA)*: where only several pixels are perturbed, with the extreme case being a single-pixel adversary; and (3) *tactically-chanced attack (TCA)*: where only significant frames are tactically chosen to be attacked. To accommodate the three restrictive settings for adversarial attack generation in deep RL, we formulate the task as a black-box optimization problem with a novel objective function to generate the FSA under black-box settings, and propose an entropy-based uncertainty measurement to achieve the TCA, where details are illustrated in Chapter 3.

However, it is noteworthy that in such minimalistic attack generation process on RL, the black-box optimization is executed **from scratch** on every frame (i.e., time-step). Therefore, the computation complexity involved may still preclude the practical realization of such attacks in time-sensitive RL tasks, since the limited time-window between consecutive frames would not allow a time-consuming attack generation.

To alleviate this issue, the central thesis of Chapter 4 is that the exploitation of correlations (similarities) between frames (through attack transfer) could greatly speed up, and thus increase the plausibility of adversarial attacks in RL. In contrast to previous approaches that generate adversarial attacks on different frames in isolation, we induce different types of *frame-correlation transfers* (each of differing degrees of computational complexity) as ways of boosting the efficiency of the adversary generation, with the goal of highlighting the threat of real-time attack realizability in *image-based* RL applications. In particular, we induce three types of frame-correlation transfers with increasing computational complexity: *anterior case transfer* that simply reuses the generated adversary from the most recent previously attacked frame as a related source solution; *random projection based transfer* that reuses adversaries from the most similar previously encountered source frame identified via a random low-dimensional mapping of the input space; *principle components based transfer* that reuses adversaries from the most similar source frame identified via optimized (variance maximizing) low-dimensional linear projections of the input space. Note that, the frame correlations reflect the consistency of environmental dynamics. Inherent similarities are naturally expected to exist among frames and their corresponding adversarial perturbations. Moreover, the presence of such correlations marks a clear distinction of RL from supervised learning tasks (e.g., classification), where the inputs are typically assumed to be i.i.d (that is, *independently* and *identically* distributed).

Given Chapter 3 and Chapter 4 have revealed the vulnerability of existed DRL policies, the following task is to improve their robustness in order to resist against the adversarial attacks that possibly appear in the state observations. To enhance the adversarial robustness of DRL policies, many previous studies have investigated adding the adversaries into the training process to achieve the corresponding generalization ability on these perturbed observations. For instance, Mirman et al. [53] and Fischer et al. [25]

utilized adversarial training [67] based policy distillation [74] to obtain an accurate and robust student policy, where the robustness is learned from the added adversarial attacks that are generated by fast gradient sign method and projected gradient descent, respectively. Recently, based on a derivation of state-adversarial Markov decision process (SA-MDP), Zhang et al. [99] proposed a robustness regularization to align the policy’s prescriptions *with* and without adversarial perturbation; thus still being adversary dependent. In a nutshell, these approaches require additional procedures of incorporating adversaries in the training, which, however, makes improvements on robustness less flexible (e.g., robust policy depending on one particular attack may fail when another attack happens), and more expensive from both computation (in calculating the adversaries) and implementation (in deploying adversaries for training in the wild) perspectives.

Instead of learning robust policy from adversaries, under the policy distillation (PD) paradigm, we prove in Chapter 5.2 that such robustness can also be achieved without relying on adversaries. Specifically, a robust student policy can be learned through maximizing the student policy’s prescription gap between the teacher selected action and the remaining actions under attack. Most importantly, we further derive that maximizing the prescription gap under attack can be transformed to simultaneously maximizing the prescription **without attack** and minimizing the Jacobian with respect to input states; this provides us the possibility of achieving adversarial robustness without learning from adversaries.

Inspired by this derivation, in Chapter 5.3 we propose an *adversary agnostic PD* (*A2PD*) loss that includes two parts: 1) prescription gap maximization (PGM) loss, and 2) Jacobian regularization (JR) loss. The PGM loss is different from most previous distillation loss functions (e.g., cross-entropy) that merely maximize the probability of the action selected by teacher policy. In contrast, we also maximize the entropy of those actions not selected by the teacher policy, which enforces the student policy to have a larger

prescription gap in order to resist perturbations in state observations. More importantly, the entropy term is weighted by the probability of the selected action; this allows the training to focus on PD accuracy at the beginning and pursue entropy maximization in the end. Last but not the least, in order to further improve the adversarial robustness, we also minimize the magnitude of Jacobian with respect to the input state, which is calculated based on PGM loss.

The itemized contributions of this dissertation can be encapsulated as follows:

- For the first time, Chapter 3 proposes a minimalistic attack that can deceive RL policies under three restrictive settings, viz., black-box policy access, fractional-state adversary, and tactically-chanced attack. In our minimalistic attack, a Shannon entropy-based uncertainty measurement is proposed to sparingly select the most vulnerable frames to be attacked. Experiments on extensive Atari games show that on Breakout: (i) with only *a single pixel* ($\approx 0.01\%$ of the state) attacked, the trained policies show significant performance degradation; and (ii) by merely attacking around 1% frames, the policy trained by DQN is significantly deceived.
- In Chapter 4, to alleviate the computation budget of minimalistic attack in Chapter 3, we propose three different frame correlation transfer methods (i.e., anterior case transfer, random projection based transfer, and principal components based transfer) with varying degrees of computational complexity in generating adversaries. Experiments validate that through the use of frame correlation transfers, the computation cost for generating adversarial example is significantly reduced, viz., the number of policy queries required in some cases is *zero*.

- For the first time in DRL, Chapter 5 proposes the adversary agnostic robust DRL that achieves robustness without relying on learning from adversaries. To this end, based on the PD paradigm, we theoretically derive that the robustness of student policy can be indeed learned independently of the adversaries. Inspired by that, we design the adversary agnostic PD (A2PD) loss function that contains: 1) a PGM loss for simultaneously maximizing the probability of the action prescribed by the teacher policy as well as the entropy of unwanted actions; and 2) a JR loss that minimizes the norm of Jacobian with respect to the input state. More importantly, the proposed A2PD has been shown to defeat the minimalistic attacks proposed in Chapter 3.

Chapter 2

Background and Related Works

In this chapter, we provide an overview of the background and related works that are related to the adversarial robustness in reinforcement learning. Specifically, Chapter 2.1 make a brief introduction about Markov decision process in reinforcement learning. Chapter 2.2 illustrates the definition and recent progress of adversarial robustness evaluation, namely, adversarial attack in RL. Chapter 2.3 presents the related studies on attack transfer and highlights our contributions on frame correlation transfers. Chapter 2.4 presents a literature review for improving the adversarial robustness of deep reinforcement learning policies.

2.1 Deep Reinforcement Learning

Deep reinforcement learning (DRL) is a sub-field of machine learning, which usually utilize deep neural network (DNN) as the function approximator to solve a reinforcement learning (RL) task. An RL problem [69, 38, 101, 68] is commonly formulated as a Markov Decision Process (MDP), where the RL agent interacts with a particular environment based on a set of possible actions and corresponding state transitions. The decision making procedure of an MDP is illustrated in Fig. 2.1, where r_t and s_t represent the reward value and state obtained at step t from the environment, respectively, and a_t indicates the action prescribed by a policy π_θ that is parameterized by θ . According to

Figure 2.1: Markov Decision Process in RL.

action a_t , the agent interacts with the environment, transitioning from the current state s_t to the next state s_{t+1} and receiving a new reward r_t . This process terminates when a decision making process is completed. The goal of an RL algorithm [76] is thus to optimize policy parameters θ (e.g., the weights of a neural network) by maximizing the expected accumulated reward,

$$\theta^* = \max_{\theta} \mathbb{E}[\sum_{t=0}^T \gamma^t r_t | \pi_{\theta}] \quad (\text{Eq. 2.1})$$

where $\sum_{t=0}^T \gamma^t r_t$ is the discounted accumulated reward value; and γ is the discount factor in the range $(0, 1)$.

To find the optimal policy parameters θ^* , many different RL algorithms have been proposed. We select several representative ones for demonstration, including DQN [55], PPO [77], A2C [54], ACKTR [93].

- **Deep Q-Networks (DQN) [55]:** Instead of predicting the probability of each action, DQN computes the *Q value* for each available action. Such *Q* value represents the approximated accumulated reward from the current frame. Based on such approximation, DQN is trained via minimizing the temporal difference loss (i.e., squared Bellman error). In our approach, the corresponding policy for DQN is achieved by selecting the action with maximum *Q* value. To keep the DQN-trained

policy consistent with the other three algorithms, a soft-max layer is added at the end of the network.

- **Proximal Policy Optimization (PPO)**[77]: PPO is an off-policy method using policy gradient, and it pursues a balance among ease of implementation, sample complexity, and ease of tuning. PPO is proposed based on trust region policy optimization (TRPO) that solves a constrained optimization problem so as to alleviate performance instability. However, PPO handles this issue in a different manner as compared to TRPO; it involves a penalty term that indicates the Kullback–Leibler divergence between the old policy action prediction and the new one. This operation computes an update at each step, minimizing the cost function while restricting the updating step to be relatively small.
- **Advantage Actor-Critic (A2C)** [83]: A2C is an actor-critic method that learns both a policy and a state value function (i.e., also called critic). In A2C, the advantage A represents the extra reward if the action a_t is taken, and is formulated as $A = Q(s_t, a_t) - V(s_t)$, where $Q(s_t, a_t)$ is the state-action value and $V(s_t)$ is the state value. This formulation reduces variance of the policy gradient and thus increases stability of the gradient estimation.
- **Actor Critic using Kronecker-Factored Trust Region (ACKTR)** [93]: ACKTR is an extension of the natural policy gradient, which optimizes both the actor and the critic using Kronecker-factored approximate curvature (K-FAC) with trust region. To the best of knowledge, ACKTR is the first scalable trust region natural gradient method for actor-critic methods. The efficacy of ACKTR suggests that Kronecker-factored natural gradient approximations in RL is a promising framework.

Although notable performances have been achieved by these algorithms on many challenging tasks (e.g., video games [55] and board games [78, 79]), recent studies have revealed that the policies trained by these algorithms are easily fooled by adversarial perturbations [39, 47, 98], as introduced next.

2.2 Adversarial Attacks in DRL

Adversarial attacks (examples) are first introduced by Szegedy et al. [84], which represent the perturbed images that are visually indistinguishable from the original images yet significantly misleading the well trained machine learning models. Since then, deep learning (DL) models have been revealed to be vulnerable against many different input perturbations (i.e., adversaries) [97, 43] in supervised learning tasks (e.g., image classification).

In general, adversarial attacks can be categorized as targeted or untargeted attack based on the definition of objective. Namely, untargeted attack aims to mislead a DNN model into making incorrect predictions without caring what attack outcome exactly is. In contrast, targeted attack forces the DNN model to output specific target class. To effectively generate both targeted and untargeted attacks, many studies have been investigated in the supervised learning domain. For instance, Goodfellow et al. [30] proposed a fast gradient sign method (FGSM), where a small perturbation based on the sign of input gradient from a single back-propagation can degrade the performance of a well-behaved classifier. Inspired by FGSM, Madry et al. [51] investigated a multi-step attack generation, namely projected gradient descent (PGD), which repeatedly update the adversarial example using the basic FGSM. Given the PGD providing much stronger attacks, it has been usually chosen as a baseline to compare with in the literature. After Szegedy et al. [84] and Madry et al. [51], a variety of other adversarial attack approaches, such as Carlini & Wagner attack [11], Jacobian based saliency map attack (JSMA) [65] and

deep fool [58], have been proposed to generate input perturbations that are sufficient to fool well trained deep neural network (DNN) classifiers while being imperceptible to the human eye. Moreover, Ilyas et al. [41] explains that the existence of adversarial examples in image classification is probably caused by training on non-robust features.

Besides image classification tasks, there also have studies on adversarial attacks for other domains, such as speech recognition [5], machine translation [92] and text summarization [100]. These generated adversaries can easily deceive a well-trained DL model to predict incorrectly, by simply adding small input perturbations that are visually or semantically imperceptible to human beings. As such, it is worth noting that most previous interest in adversarial attack generation has been focused on supervised learning tasks. In contrast, adversarial attacks on deep reinforcement learning (DRL) have been relatively less investigated thus far. Moreover, due to our adversarial attack in RL focuses on diminishing the accumulated reward through input perturbations instead of achieving a targeted decision trajectory, we mount our attack on the untargeted scenario.

In the context of DRL, Huang et al. [39] were among the first to analyze the vulnerability of DNN policies, where they utilized the FGSM [30]) to generate extensive adversarial perturbations (i.e., perturbing nearly every pixel of every frame) by back-propagating the divergence loss of the action distribution through known DNN architectures and weights. Lin et al. [47] explored strategically-timed attacks by partially perturbing only selected frames, but the adversary is generated by a gradient-based approach (i.e., C&W attack [11]) and is still deemed as being extensive due to the number of pixels that are adjusted. In other applications, Trejo et al. [88] and Solis et al. [80] have considered the interactions between attackers and defenders for adapting patrolling strategies using RL in Stackelberg security games.

The aim of an adversarial attack in RL is to find an optimal adversary δ_t^* that corresponds to a minimized accumulated reward. The general formulation of an adversarial

attack in RL can therefore be formulated as,

$$\min_{\delta_t} \sum_{t=1}^T r(\pi, s_t + \delta_t) : \forall t \|\delta_t\| \leq C \quad (\text{Eq. 2.2})$$

where T is the length of an episode, and $r(\pi, s_t)$ is a reward function which returns the reward value given state s_t and DNN policy π . δ_t represents the adversary generated at time-step t , with its norm $\|\delta_t\|$ bounded by C , that represents the visibility of a generated adversarial perturbation δ_t . From an attacker’s perspective, the generated adversarial perturbations are intended to be as humanly imperceptible as possible while deceiving a well trained RL policy π . In other words, such constraint C comes from the objectives of attacker; thus naturally being known by the attack designer. In Eq. 2.2, the objective is that the perturbed input state $s_t + \delta_t$ would lead to a sub-optimal action selection, and thereby threaten the optimality of the accumulated reward value. Alternatively stated, the action prediction a_t^p on the perturbed state $s_t + \delta_t$ may differ from the originally unperturbed action a_t^o ; this in turn jeopardizes the reward value r_t due to the sub-optimal action taken. We note that such an assumption naturally corresponds to the definition of an untargeted attack [30], which can be stated as $a_t^o \neq a_t^p$.

To optimize the adversary δ_t in Eq. 2.2, many prior studies utilize gradient-based approaches (e.g., fast gradient sign method [30]) to back-propagate a divergence loss in action distribution through the policy network to compute the gradient with respect to the input state s_t ; the resultant gradient is denoted as $\nabla_{s_t} J(\pi, \theta, \Delta a_t, s_t)$. Here, θ represents the weights of the neural network; Δa_t is the change in action space; J indicates the loss function (e.g., cross-entropy loss). Moreover, the imperceptibility of added perturbation is an important considering. In minimalistic attacks, the imperceptibility is implemented through minimal number of perturbed pixels, which can be deemed as a L_0 norm constraint. The assumption is that the less number of pixel perturbed, the harder perturbation could be observed. Solving such L_0 norm constrained problem however is

NP-hard. Previous approaches (e.g., SparseFool [57]) solve such NP-hard problem via converting it as a L_1 constrained problem under the white-box setting. For black-box setting in our study, we utilize genetic algorithm to heuristically find promising solutions.

It is worth noting that a necessary prerequisite for a gradient-based approach is the complete knowledge of a policy’s structure and model parameters θ . This prerequisite however does not always hold, as most systems do not release their internal specifications, only allowing a model to be queried as an oracle. As such, in order to truly unveil the vulnerability of such systems, the derivative-free, black-box optimization approaches may naturally be more suited. On the other hand, it is also noteworthy that using derivative free attack generation requires a number of queries for the attacked DRL policies, which may not be suitable for the RL tasks that has a strict time window between consecutive frames. For such scenario, the attacks transfer across the frames in an MDP is a promising approach as shown in Chapter 4.

2.3 Adversarial Attacks Transfer

The transferability of adversarial attack has been well studied in supervised learning tasks. In general, previous studies on transferability mainly refer to the ability of an attack against one model to be effective against other models. Particularly, the attack transferability was originally observed in early studies on adversarial examples [30, 84]. In particular, Szegedy et al. [84] is among the first to observe the adversarial examples are transferable across different classifiers. Such adversarial transferability is adopted in a practical black box attack [64], where a surrogate model is trained to reduce the number of model queries. Later on, since the increased requirements on deploying machine learning models from server to the terminal devices, more interests on attack transferability has been gained. For instance, Xie et al. [95] propose to improve the transferability of adversarial examples by creating diverse input patterns. Demontis et al. [19] provide

a comprehensive analysis with the goal of investigating the transferability of both test-time evasion and training-time poisoning attacks. Dong et al. [21] showcased that the adversarial transferability can be boosted via integrating the momentum term into the iterative process for attacks, with the goal of stabilizing update directions and escaping from poor local maxima during the iterations. Josh et al. [86] leverage domain randomization, a widely-used generalization improving approach, to generate adversarial attacks that enable vulnerability transfer. However, most previous studies on attack transferability is mainly for supervised learning tasks (e.g., image classification); seldom study has been done in the reinforcement learning setting. Moreover, such transferability mainly captures the ability of an attack against a machine-learning model to be effective against another different model [19]. This is different from our transferability that captures the correlation across different frames in an MDP.

In RL, Huang et al. [39] mentioned the adversarial transferability based on FGSM, which however requires access to the training environment of the target model. Xiao et al. [94] investigated the adversarial transferability through a surrogate policy that is trained based on behavior cloning. These studies for transferability are mainly based on white-box policy access setting. Under black-box setting, Chapter 3 investigates a minimalistic attack with black-box policy access, which only perturbs a small number of pixels in a few selected frames utilizing derivative-free optimization. However, Chapter 3 does not comprehensively account for the scope of exploiting frame correlations. For this reason, even though [70] marks an advance over prior forms of adversarial attacks in RL, its adversary generation process may still be insufficient for unveiling the realizability of real-time attacks across many time-sensitive RL applications.

In summary, existing studies in the literature have mainly focused on generating extensive adversarial examples utilizing gradient-based approaches, and more importantly, not paying sufficient attention to the presence of correlations between frames for attack

transfer in RL. To the best of our knowledge, there is no comprehensive study on the potency of frame-correlation transfers in RL for reducing computational complexity of generating selective adversarial perturbations. It is expected that the exploitation of such correlations (through attack transfer) can greatly speed up, and thus increase the plausibility of, adversarial attacks in image-based RL applications.

In contrast, Chapter 4 marks the first study on utilizing the attack transferability across a Markov decision process to accelerate the attack generation. Therefore, from this view, our attack transferability in DRL also falls in the domain of transfer optimization. Given the vulnerability of current DRL policies, a following question is how can we increase the adversarial robustness of DRL policies in order to resist against the attacks appearing at test time.

2.4 Adversarial Robustness Improvements in DRL

Robustness of RL has been studied for long due to the main components (i.e., states, actions, transition dynamics and rewards) could exist uncertainty or errors. For instance, the robust Markov decision process (RMDP) [62] improves the robustness of policies by involving the worst perturbation matrices towards the transition probabilities. Xu and Mannor [96] extend such RMDP to a distributional setting, viz., the uncertainty is described by a sequence of nested sets, each of which corresponds to a probabilistic guarantee for a different confidence level. Osogami [63] further considers a partially observable Markov decision process, where the cumulative reward is maximized for the worst case where uncertain parameters are only known in a given region. These studies however are mainly based on the perturbation in transition dynamics, which is different from our research scope on the robustness towards state observation. There are some other studies [28] in improve robustness based on a minimax game between two policies,

where one policy’s action is expected to impact the other’s state observation and action prediction. This however is till different from our research scope.

To resist against adversarial attacks on state observations in DRL, there have been several works that study the adversarial robustness improvement by adversarial training [98]. In specific, adversarial training utilize data augmentation to involve the generated adversarial examples into the training data. This thereby helps the model to be generalizable towards similar attacks. Mandlekar et al. [52] applied adversarial training on policy gradient algorithm by leveraging a simple FGSM to generate adversarial examples, but they just tested on the RL tasks (e.g., Mujoco locomotion) that have relatively low dimensional input state. Behzadan and Munir [4] adopted similar adversarial training to several Atari games, but their results suggest that the performance of adversarially trained policies got struggled when attacks are included in the training data. Pattanaik et al. [66] introduced much stronger attacks that are achieved by projected gradient descent (PGD) in adversarial training on Atari games. However, the rebustness improvements come with significant performance drops. To obtain better robustness, Mirman et al. [53] and Fischer et al. [25] proposed a adversarial training based policy distillation to build a more robust student policy, where FGSM and PGD are utilized to generate adversarial attacks during training. Recently, Zhang et al. [99] proposed a robustness regularization to align the policy’s prescriptions with and without adversarial perturbation; the robustness improvement is therefore still dependent on adversaries.

In sum, those approaches require additional procedures to operate adversaries during training; this makes improvements on adversarial robustness less flexible and more expensive in computation and real-world implementation. In contrast, our policy distillation approach is able to learn a robust student policy that does not rely on any information of the adversaries.

The ***broader impact*** of our adversary agnostic approach for improving adversarial robustness would be highlighted in those safety critical applications, since our robustness

Figure 2.2: The taxonomy of studies in adversarial robustness of deep reinforcement learning

can be achieved without requiring the expensive and dangerous adversarial examples to be involved in the training process. For instance, in autonomous driving, previous adversary dependent approaches may need to witness traffic accidents (as the results of perturbed action selection) with the goal of improving the robustness against input perturbations. *In contrast*, our approach alleviates this issue via avoiding utilizing those dangerous and expensive data. Therefore, our adversary agnostic robust DRL provides a more realistic solution for improving the adversarial robustness in the wild.

2.5 Summary

Given the vulnerability issues of DRL policies and the limitations of current attack dependent defenses illustrated above, this dissertation is inspired to study both the evaluation and improvement of the adversarial robustness in DRL. The paradigm of our study is shown in Fig. 2.2, where in our studies, the robustness evaluation and improvement mainly focus on the black-box attack and attack agnostic defense, respectively.

Chapter 3

Minimalistic Attacks: How Little It Takes To Fool Deep Reinforcement Learning

Due to RL systems are deployed in various safety-critical applications, it is imperative to understand how vulnerable they are to adversarial attacks. Most prior works on DRL adversarial attacks mainly analyze the effects of perturbing every pixel of every frame assuming white-box policy access. These assumptions however might be unrealistic in some real scenarios. By contrast, this chapter¹ we propose to take a more restrained view towards adversary generation - with the goal of exploring the limits of a DNN model's vulnerability in RL. In specific, we focus on *minimalistic attacks* by only considering adversarial examples that perturb limited number of pixels in selected frames, and under restricted black-box policy access. In other words, we intend to unveil how little it really takes to successfully fool state-of-the-art RL policies. In what follows, we briefly introduce the three restrictive settings that characterize our study.

- **Black-box Policy Access (BPA).** Most previous studies focus on a white-box setting [98], that allows full access to a policy network for back-propagation. However, most systems do not release their internal configurations (i.e., network struc-

¹The work in this chapter has been published in IEEE Transactions on Cognitive & Development Systems, 2020

ture and weights), only allowing the model to be queried; this makes the white-box assumption too optimistic from an attacker’s perspective [13]. In contrast, we use a BPA setting where the attacker treats the policy as an oracle with access only to its input and outputs.

- **Fractional-State Adversary (FSA).** In the FSA setting, the adversary only perturbs a small fraction of the input state. This, in the extreme situation, corresponds to the single-pixel attack shown in Figure 3.1, where perturbing a single pixel of the input state is found to change the action prescription from ‘RIGHT’ to ‘LEFT’. In contrast, most previous efforts [98] are mainly based on a full-state adversary (i.e., the number of modified pixels is fairly large, usually spanning the entire frame).
- **Tactically-Chanced Attack (TCA).** In previously studied RL adversarial attacks [39, 66, 67], the adversary strikes the policy on every frame of an episode; this is a setting termed as the fully-chanced attack. Contrarily, we investigate a relatively restrictive case where the attacker only strikes at a few selected frames - a setting we term as *tactically-chanced attack*, where a minimal number of frames is explored to strategically deceive the policy network.

The proposed restrictive settings are deemed to be significant for safety-critical RL applications, such as in the computer vision based manufacturing systems [75] that deploys DRL agents. In such domains, there may exist a temporal gap between the acquisition of the input and the action execution; thereby, providing a time window in which a tactical attack could take place. Moreover, the restrictive nature of the attacks makes them practically imperceptible, greatly reducing the chance of identifying and rectifying them. With this in mind, the present chapter explores the vulnerability of deep RL models to

restrictive adversarial attacks, with particular emphasis on image-based tasks (i.e., Atari games).

The key challenge of this research, therefore, lies in assessing how to effectively accommodate the three restrictive settings for adversarial attack generation in deep RL. To this end, we design a mathematical program with a novel objective function to generate the FSA under black-box settings, and propose an entropy-based uncertainty measurement to achieve the TCA. The optimization variables are defined as the discrete 2-D coordinate location(s) and perturbation value(s) of the selectively attacked pixel(s), and the designed mathematical program guarantees a successful deception of the policy as long as a positive objective value is found. The optimization procedure, for a given input frame, is then carried out by a simple genetic algorithm (GA) [37, 18].

In minimalistic attacks, the imperceptibility is controlled through minimal number of perturbed pixels, which can be deemed as a L_0 norm constraint. The assumption is that the less number of pixel perturbed, the harder perturbation could be observed. Solving such L_0 norm constrained problem however is NP-hard. Previous approaches (e.g., SparseFool [57]) solve such a NP-hard problem via converting it as a L_1 constrained problem under the white-box setting. For black-box setting in our study, we utilize the genetic algorithm to heuristically find promising solutions.

Furthermore, the Shannon entropy of the action distribution is specified as the attack uncertainty, and is utilized to only select a few salient frames to be tactically attacked. We demonstrate the sufficiency of the three restrictive settings on four state-of-the-art deep RL policies pre-trained on six Atari games.

To sum up, the contributions of this chapter can be encapsulated as follows:

- We unveil how little it takes to deceive an RL policy by considering three restrictive settings, namely, black-box policy access, fractional-state adversary, and tactically-chanced attack;

- We formulate the RL adversarial attack as a black-box optimization problem comprising a novel objective function and discrete FSA optimization variables. We also propose a Shannon entropy-based uncertainty measurement to sparingly select the most vulnerable frames to be attacked;
- We explore the three restrictive settings on the policies trained by four state-of-the-art RL algorithms (i.e., DQN [55], PPO [77], A2C [54], ACKTR [93]) on six representative Atari games (i.e., Pong, Breakout, SpaceInvaders, Seaquest, Qbert, BeamRider);
- Surprisingly, we find that on Breakout: (i) with only *a single pixel* ($\approx 0.01\%$ of the state) attacked, the trained policies show significant performance degradation; and (ii) by merely attacking around 1% frames, the policy trained by DQN is significantly deceived.

3.1 Black-box Policy Access

Most previous studies focus on a white-box setting [98], that allows full access to a policy network for back-propagation. However, most systems do not release their internal configurations (i.e., network structure and weights), only allowing the model to be queried; this makes the white-box assumption too optimistic from an attacker’s perspective [13]. In contrast, we use a BPA setting where the attacker treats the policy as an oracle with access only to its input and outputs. We adopt the commonly used black-box definition [13, 82] from supervised learning, where the attacker can only query the policy. In other words, the attacker is unable to compute the gradient $\nabla_{s_t} \pi(\cdot | s_t, \delta_t)$, but only has the privilege to query a targeted policy so as to obtain useful information for crafting adversarial examples. We note that such a setting has been rarely investigated in RL adversarial attacks.

We realize the proposed BPA setting by formulating the adversarial attack as a black-box optimization problem. To this end, we first define a measure $\mathcal{D}(\cdot)$ to quantify the discrepancy between the original action distribution $\pi(\cdot|s_t)$ (produced by an RL policy without input perturbation) and the corresponding perturbed distribution $\pi(\cdot|s_t + \delta_t)$. Assuming a finite set of m available actions $a_t^1, a_t^2, \dots, a_t^m$, the probability distribution over actions is represented as $\pi(\cdot|s_t) = [p(a_t^1), p(a_t^2), \dots, p(a_t^m)]$, where $\sum_{j=1}^m p(a_t^j) = 1$. Typically, a deterministic policy selects the action $o = \arg \max_j p(a_t^j)$. With this, the black-box attacker considers the discrepancy measure as the optimization objective function, where the state s_t is perturbed by adversary δ_t such that the measure $\mathcal{D}(\cdot)$ is maximized.

The overall problem formulation can thus be stated as,

$$\max_{\delta_t} \mathcal{D}(\pi(\cdot|s_t), \pi(\cdot|s_t + \delta_t)) : \forall t \|\delta_t\| \leq L. \quad (\text{Eq. 3.1})$$

The above mathematical program changes the action distribution to $\pi(\cdot|s_t + \delta_t)$, from the original (optimal) action distribution $\pi(\cdot|s_t)$. *If the change in action distribution leads to a change in action selection at any point in the trajectory of an RL agent, then, based on Bellman's principle of optimality [83], the resulting sub-trajectory shall lead to a sub-optimal cumulative reward as a consequence of the attack.* It is based on this principle that we propose to substitute the reward minimization problem of Eq. 2.2 with the discrepancy maximization formulation in Eq. 3.1.

The exact choice of the discrepancy measure $\mathcal{D}(\cdot)$ is expected to have a significant impact on the attack performance, as different measures shall capture different variations in the action distributions. Previous works [39, 94, 47] apply the Euclidean norm (e.g., L_1, L_2, L_∞) between $\pi(\cdot|s_t + \delta_t)$ and $\pi(\cdot|s_t)$. However, such L_p norm cannot guarantee a successful untargeted attack, since maximizing the L_p norm may not guarantee that the action selection has been altered by the perturbed state. Thus, a successful untargeted

attack under deterministic action selection must ensure that $\arg \max_j \pi(\cdot|s_t + \delta_t)_j \neq \arg \max_j \pi(\cdot|s_t)_j$, where $\pi(\cdot|s_t)_j = p(a_t^j)$.

To enable a more consistent discrepancy measure for untargeted attacks, we design the following function $\tilde{\mathcal{D}}$ based on the query feedback of the policy,

$$\begin{aligned} \tilde{\mathcal{D}} &= \max_{j \neq o} \pi(\cdot|s_t + \delta_t)_j - \pi(\cdot|s_t + \delta_t)_o \\ \text{where: } o &= \arg \max_j \pi(\cdot|s_t)_j. \end{aligned} \tag{Eq. 3.2}$$

This formulation is different from the Euclidean norm, guaranteeing a successful untargeted attack if $\tilde{\mathcal{D}}$ is positive. To support this claim, we refer to the theorem and proof below.

Theorem 1. Suppose the discrepancy measure $\tilde{\mathcal{D}}$ in Eq. 3.2 is positive and policy π is a deterministic, i.e., action a_t^o is chosen such that $o = \arg \max_j \pi(\cdot|s_t)_j$. Then, the adversarial example δ_t for policy π at state s_t is guaranteed to be a successful untargeted attack, i.e.,

$$\arg \max_j [\pi(\cdot|s_t + \delta_t)]_j \neq \arg \max_j [\pi(\cdot|s_t)]_j.$$

Proof. We use the symbol o and p to represent the index of the selection action from state s_t and perturbed state $s_t + \delta_t$, respectively. As the policy π is deterministic, o and p are given as,

$$\begin{aligned} o &= \arg \max_j [\pi(\cdot|s_t)]_j, \\ p &= \arg \max_j [\pi(\cdot|s_t + \delta_t)]_j. \end{aligned} \tag{Eq. 3.3}$$

If the discrepancy measure $\tilde{\mathcal{D}}$ is positive, then we have,

$$\begin{aligned} \tilde{\mathcal{D}} > 0 &\Rightarrow \max_{j \neq o} \pi(\cdot|s_t + \delta_t)_j - \pi(\cdot|s_t + \delta_t)_o > 0 \\ &\Rightarrow \max_{j \neq o} \pi(\cdot|s_t + \delta_t)_j > \pi(\cdot|s_t + \delta_t)_o \\ &\Rightarrow \max_j \pi(\cdot|s_t + \delta_t)_j = \max_{j \neq o} \pi(\cdot|s_t + \delta_t)_j. \end{aligned} \tag{Eq. 3.4}$$

Therefore, the perturbed action index p is given by,

$$\begin{aligned} p &= \arg \max_j \pi(\cdot | s_t + \delta_t)_j = \arg \max_{j \neq o} \pi(\cdot | s_t + \delta_t)_j \neq o \\ &\Rightarrow \arg \max_j [\pi(\cdot | s_t + \delta_t)]_j \neq \arg \max_j [\pi(\cdot | s_t)]_j. \blacksquare \end{aligned}$$

With the aforementioned discrepancy measure $\tilde{\mathcal{D}}$, the mathematical program for generating attacks is given by,

$$\begin{aligned} \max_{\delta_t} \max_{j \neq o} \pi(\cdot | s_t + \delta_t)_j - \pi(\cdot | s_t + \delta_t)_o \\ \text{where: } \forall t \quad \|\delta_t\| \leq L, \quad o = \arg \max_j \pi(\cdot | s_t)_j \end{aligned} \tag{Eq. 3.5}$$

When resolving the optimization problem in Eq. 3.5, δ_t is determined according to its parameterisation. In this chapter, δ_t is limited to perturbing only a small fraction of the input state (i.e., fractional-state adversary), as described next.

3.2 Fractional-State Adversary

In the FSA setting, the adversary only perturbs a small fraction of the input state. This, in the extreme situation, corresponds to the single-pixel attack shown in Figure 3.1, where perturbing a single pixel of the input state is found to change the action prescription from ‘RIGHT’ to ‘LEFT’. In contrast, most previous efforts [98] are mainly based on a full-state adversary (i.e., the number of modified pixels is fairly large, usually spanning the entire frame).

In comparison to the full-state adversary depicted in Fig. 3.2(a), FSA only perturbs a fraction of the state; shown in Fig. 3.2(b). The extreme scenario for FSA is merely a single-pixel attack, which is deemed to be more physically realizable for the attacker than a full-state one. For instance, pasting a sticker or a simple fading of color in the “STOP” sign could easily lead to a FSA in Figure 3.2(b). Such FSA setting has been investigated in image classification tasks, such as one-pixel attack [82] and adversarial patch [10], yet seldom mentioned in the RL setting.

Figure 3.1: The visualization of the single pixel attack on Breakout.

To achieve the FSA setting, we parameterize the adversary δ_t by its corresponding pixel coordinates (i.e., $\mathbf{x}_t, \mathbf{y}_t$) and perturbation value $\mathbf{p}_t$ as follows,

$$\begin{aligned} \delta_t &\leftarrow \mathcal{P}(\mathbf{x}_t, \mathbf{y}_t, \mathbf{p}_t) \\ &\leftarrow \mathcal{P}(x_t^1, y_t^1, p_t^1, \dots, x_t^n, y_t^n, p_t^n) \end{aligned} \quad (\text{Eq. 3.6})$$

where n is the number of pixels that are attacked in a frame, x_t^i, y_t^i are coordinate values of the i^{th} perturbed pixel, and p_t^i is the adversarial perturbation value of the i^{th} pixel.

With Eq. 3.6, the final black-box optimization problem is stated as,

$$\begin{aligned} &\max_{\mathbf{x}_t, \mathbf{y}_t, \mathbf{p}_t} \max_{j \neq o} [\pi(\cdot | s_t + \mathcal{P}(\mathbf{x}_t, \mathbf{y}_t, \mathbf{p}_t))]_j - [\pi(\cdot | s_t + \mathcal{P}(\mathbf{x}_t, \mathbf{y}_t, \mathbf{p}_t))]_o \\ \text{where: } &\forall t \ o = \arg \max_j [\pi(\cdot | s_t)]_j; \\ &\forall t \ 0 \leq x_t^i \leq \mathcal{I}_{\mathbf{x}}, \ x_t^i \in \mathbb{N}, \ i = [1, 2, \dots, n]; \\ &\forall t \ 0 \leq y_t^i \leq \mathcal{I}_{\mathbf{y}}, \ y_t^i \in \mathbb{N}, \ i = [1, 2, \dots, n]; \\ &\forall t \ 0 \leq u_t^i + p_t^i \leq \mathcal{I}_{\mathbf{p}}, \ p_t^i \in \mathbb{N}, \ i = [1, 2, \dots, n]; \end{aligned} \quad (\text{Eq. 3.7})$$

where $\mathcal{I}_{\mathbf{x}}, \mathcal{I}_{\mathbf{y}}$ are the integral upper bounds of $\mathbf{x}_t$ and $\mathbf{y}_t$ respectively, u_t^i is the unperturbed value of the i^{th} pixel in s_t , $\mathcal{I}_{\mathbf{p}}$ is the maximum allowable perturbed pixel value, and n is

Figure 3.2: Full-state adversary versus the FSA

the FSA size that controls the number of pixels to be attacked given any input state s_t . The extreme case is $n = 1$, implying that there is only *one pixel* attacked in an input frame. To the best of our knowledge, such one-pixel attack has never been explored in the context of RL adversarial attacks.

To obtain the optimal $\mathbf{x}_t, \mathbf{y}_t, \mathbf{p}_t$, we utilize a simple genetic algorithm (GA) [37, 7, 18] (i.e., a *derivative-free* evolutionary computation [42, 49, 48, 23] method for black-box optimization). The pseudo code of the GA is illustrated in Algorithm 1 (lines 7-17). A population pop containing λ individuals is evolved, where each individual $\delta_t^i, i \in [1, \lambda]$ is a candidate adversarial example that is evaluated by the objective function in Eq. 3.7. The top- $\lambda\beta$ individuals with respect to the objective value are selected as elites to survive in the new generation (line 10). Based on standard crossover (line 12) and mutation (line 13) operations, another population pop' is reproduced to replace the old pop . This process repeats until a specified maximum number of function evaluations M^{eval} is reached.

3.3 Tactically-Chanced Attack

To explore adversarial attacks on a restricted number of frames, we design tactically-chanced attacks (TCA) where the attacker strategically strikes *only* salient frames that

Algorithm 1: Attack Generation via GA

Input: s_t – state at time-step t ; n – number of pixels for attack; λ – population size; β – selection rate; γ – rate of mutation; M^{eval} – maximum number of function evaluations; ζ^* – TCA threshold value

- 1 Load the policy π pre-trained by a particular RL algorithm;
- 2 Obtain the RL input state s in a particular Atari game;
- 3 Calculate the attack uncertainty ζ_t by Eq. 3.8;
- 4 **if** $\zeta_t \geq \zeta^*$ **then**
 - 5 // No adversarial attack
 - 5 $\delta_t = \text{none}$;
- 6 **else**
 - 7 // Explore the n pixel attack
 - 7 Initialize the population pop with randomly generating λ individuals;
 - 8 **repeat**
 - 9 Evaluate the discrepancy measure $\tilde{\mathcal{D}}$ of each individual;
 - 10 Save the top- $\lambda\beta$ individuals in pop' ;
 - 11 **for** $j=1:\frac{\lambda(1-\beta)}{2}$ **do**
 - 12 Crossover: generate δ_c^j with one-point crossover [37];
 - 13 Mutation: $\delta_m^j = \delta^j + \gamma \cdot \epsilon_j$, $\epsilon_j \in \mathcal{N}(0, 1)$;
 - 14 Append δ_c and δ_m to pop' ;
 - 15 $pop = pop'$
 - 16 **until** the maximum objective evaluations M^{eval} reached;
 - 17 Return the optimal FSA $\delta^* = [\mathbf{x}^*, \mathbf{y}^*, \mathbf{p}^*]$;

are likely to be most contributing to the accumulated reward. In contrast, most existing approaches [39] apply adversarial examples on every frame, which is referred to as fully-chanced attack. Our proposed TCA is more restrictive, realizable, and threatening, as can further be highlighted from three different perspectives: (1) due to the communication budget restriction, the attacker may not be able to strike the policy in a fully-chanced fashion; (2) a tactically-chanced attack is less likely to be detected by the defender; and (3) only striking the salient frames improves the attack efficiency, as many frames contribute trivially to the accumulated reward.

To this end, we define the normalized Shannon entropy of the action distribution $\pi(\cdot|s_t) = [p(a_t^1), p(a_t^2), \dots, p(a_t^m)]$ as a measure of attack uncertainty. Specifically, the

Figure 3.3: The attack uncertainty ζ_t on Pong.

attack uncertainty (ζ_t) of each frame is given by,

$$\zeta_t = - \sum_{i=0}^m \frac{p(a_t^i) \cdot \log p(a_t^i)}{\log m} \quad (\text{Eq. 3.8})$$

where $p(a_i)$ is the probability of action a_i ; m is the dimensionality of the action space. Note that ζ_t must lie within $(0, 1]$. Under this definition, the frame with relatively low ζ_t indicates that the policy has high confidence in its prescribed action. Hence, we assume that attacking such frames will effectively disrupt the policy.

We demonstrate the attack uncertainty on Pong in Figure 3.3; similar trends can be observed on the other games as well. In the figure, the frames with smaller ζ_t , marked by brown rectangles, depict that the ball is close to the paddle. On the other hand, the blue rectangle marked ones with larger ζ_t (close to 1) indicate that the ball is distant from the paddle. Attacking the brown marked frames are intuitively more effective in fooling the policy, as they are likely to lead to a more considerable reward loss. In contrast, an attack may be relatively inconsequential for the blue marked frames. As the ball is

distant from the paddle, attacking such frames will have little impact on the cumulative reward.

More importantly, the TCA setting may pose more practical threat to RL agents deployed in the wild. Compared to fully chanced attack that add perturbation at every time-step, TCA only appears occasionally when it is believed as harmful enough. This may significantly make the possible adversarial defense or detection harder due to such a less predictable pattern of TCA. For instance, in self-driving systems, if attack happens every second, the system or even passengers can easily notice that. In contrast, the attacks appear off and on under the TCA setting will be less perceptible but still threat the safety of driving behavior.

Therefore, we formulate TCA by defining a TCA threshold (ζ^*), as shown by the red dashed line in Figure(3.3); this threshold controls the proportion of attacked frames, such that only those frames with an uncertainty value below ζ^* are perturbed. In the experimental section, we analyze the sensitivity of our approach to the choice of ζ^* by varying its values; this also helps us to explore how little it actually takes to deceive a policy (from the perspective of the number of frames attacked). The adversary δ_t under TCA setting is thus given by,

$$\delta_t = \begin{cases} \mathcal{P}(\mathbf{x}_t^*, \mathbf{y}_t^*, \mathbf{p}_t^*), & \text{if } \zeta_t < \zeta^*, \\ \text{none}, & \text{if } \zeta_t \geq \zeta^*. \end{cases} \quad (\text{Eq. 3.9})$$

Eq. 3.9 implies that if the attack uncertainty (ζ_t) is smaller than ζ^* , an adversary shall be generated by solving the optimization problem in Eq. 3.7. Otherwise, the attacker will tactically hide without wasting compute resources on trivial frames.

3.4 Experiments and Analysis

We evaluate the three restrictive settings on six Atari games in OpenAI Gym [9] with various difficulty levels, including Pong, Breakout, SpaceInvaders, Seaquest, Qbert, and

Figure 3.4: The six representative Atari games [9] considered in the experiments

BeamRider². Following the setting from [55], these games are played by RL agents with looking at the screen images. The game screen shots are shown in Figure 3.4. For instance, in Breakout as shown by Figure 3.4 (b), the player controls the paddle with four actions (i.e., 'LEFT', 'RIGHT', 'FIRE', 'NOOP'). More details for playing other games can refer to [9]. For each game, the policies are trained by four start-of-the-art RL algorithms, including DQN [55], PPO2 [77], A2C [54], ACKTR [93]. The network structure is taken from [56], and is kept the same for all the four RL algorithms.

²Our code is available at <https://github.com/RLMA2019/RLMA>

Figure 3.5: The neural network structure for playing Atari games

3.4.1 Experiment Setup

We utilize the fully-trained policies from the RL baseline zoo [73], where each policy is trained for 4×10^7 time-steps. The other parameter settings (e.g., learning rate, discount factor and max gradient norm) for each training algorithm follow the default settings in the RL baseline zoo [73]. Each policy follows the same pre-processing steps and neural network architecture (i.e., shown as Figure 3.5) as in [55]. The input state s_t of the neural network is the concatenation of the last four screen images, where each image is resized to 84×84 . The pixel value of the grey scale image is in the range $[0, 255]$ stepped by 1. The output of the policy is a distribution over possible actions for PPO2, A2C, ACKTR, and an estimation of Q values for DQN.

To calculate the attack uncertainty for DQN, the soft-max operation is applied to normalize the output. Moreover, given the image size and pixel value, the constraints in Eq. 3.7 are set as $\mathcal{I}_x = 84$, $\mathcal{I}_y = 84$, $\mathcal{I}_p = 255$. In the GA, the population size λ , maximum number of objective function evaluations M^{eval} , selection rate β and mutation rate γ are set to 10, 400, 0.2 and 0.1 respectively. However, following Theorem 1, the optimization process is terminated when a positive function value is found. To investigate the impact of FSA size n , we alter its value from 1 to 10 in steps of 1. In other experiments, we analyze the effect of the TCA threshold ζ^* by setting different values in $[0, 1]$. This leads to different proportions of attacked frames.

3.4.2 Results for Fractional-State Adversary

We investigate the impact of FSA size n , where $n = 1$ corresponds to the extreme case that only one pixel is perturbed. For a fair comparison with respect to different algorithms and games, in the experiments reported in Fig. 3.8 we have set the TCA threshold ζ^* as the mean of all frames' attack uncertainty values. Fig. 3.8 reveals the performance (i.e. accumulated reward) drop of the policies trained by the four RL algorithms on the six Atari games. Several interesting observations can be noted.

(1) Overall, the FSA size n is positively related to the performance drop. That is, the FSA with larger n is able to deceive the policy to a larger extent. (2) On most of the games, the policies are almost completely fooled with $n \leq 4$. This indicates a small FSA size is often sufficient to achieve a successful adversarial attack, indicating the efficiency of FSA. (3) The results indicate that high performance in the absence of attacks is not a guarantee of robustness against adversarial attacks. For instance, in SpaceInvaders, the original performance of A2C is higher than DQN, but the performance of A2C drops faster than that of DQN as shown in Figure 3.8(c). (4) We *surprisingly* find that: on Breakout, with only a single pixel attacked, the policies trained by all the four RL algorithms are deceived, with the accumulated reward dropping by more than 50%; the visualisation for the single-pixel attack on Breakout is shown in Figure 3.1. Such single pixel attack corresponds to an approximate perturbation proportion of 0.01% (i.e., $1/(84 \times 84)$) of the total number of pixels in a frame. On Qbert, the accumulated reward of the policy trained by DQN drops from 10000 to 0 with two pixels attacked.

3.4.3 Results for Tactically-Chanced Attack

We study the tactically-chanced attack (TCA) by altering the TCA threshold ζ^* . Different ζ^* values lead to different proportions of attacked frames. In our exploration, we set the ζ^* in the range of $(0, 1]$, where $\zeta^* = 0$ and $\zeta^* = 1$ correspond to 0% and 100%

Figure 3.6: The relationship between attacked frames and TCA threshold ζ^* , where the line and shaded area illustrate the mean and standard deviation of 30 independent runs respectively.

proportion of attacked frames, respectively. Recall that according to the results in Figure 3.8, most of the policies are sufficiently fooled with no more than 4 pixels attacked. Hence, for a fair comparison, we set the FSA size $n = 4$ in subsequent experiments. The results are displayed in Figure 3.9, where we observe that in general the performance (i.e., accumulated reward) decreases for higher TCA threshold ζ^* values. This demonstrates that the policies are more easily fooled with more frames attacked. In addition, we make some other instructive observations.

(1) In Figure 3.9 (a) on Pong and Figure 3.9 (d) on Seaquest, the policies trained by PPO2 are more sensitive to the increasing of TCA threshold values. In contrast, the performance of policies trained by DQN appears to be more stable with the increasing values of TCA threshold, until it is close to 1. We infer that this observation is because DQN provides Q values instead of directly predicting action probabilities; the resultant increased variance of the derived action distribution leads to much fewer attacked frames under a given threshold value. (2) In Figure 3.9 (b) on Breakout, the accumulated rewards of the policies trained by PPO2, A2C and ACKTR decrease to 0 when the TCA

Figure 3.7: The relationship between attacked frames and total frames, where the line and shaded area illustrate the mean and standard deviation of 30 independent runs respectively.

threshold value is higher than 0.3. This indicate that a small TCA threshold value is enough to completely deceive those policies. **(3)** In Figure 3.9 (f) on Qbert, when the TCA threshold value is higher than 0.6, all policies can only obtain a reward less than 2000. Especially for the policy trained by PPO2, the accumulated reward drops from 14000 to 2000.

We further analyze the exact number of attacked frames with different TCA threshold values. As shown in Figure 3.6, the number of attacked frames naturally increases with the prescribed threshold value. In Figure 3.6 (b), on Breakout, when the TCA threshold value is smaller than 0.2, less than 50 frames are attacked. A similar finding is also observed on Beamrider, where less than 100 frames are attacked when the TCA threshold value is smaller than 0.2.

We also analyze the relationship between the number of attacked frames and the number of total frames in Figure 3.7. We find similar trends on the six games, where the total number of frames in the game decreases when more frames are attacked. This observation is particularly apparent for Breakout in Figure 3.7 (b), where attacking less

Figure 3.8: The results of adversarial attack with different FSA size n (i.e. number of attacked pixels), where the line and shaded area illustrate the mean and standard deviation of 30 independent runs respectively.

than 50 frames considerably reduces the total number of frames. This results from the fact that the attacked frames may cause (i) termination of the game, or (ii) a loss of life. Both outcomes greatly shorten the episode, resulting in a smaller number of total frames. For instance, if the paddle in Breakout is deceived to miss the ball, the agent would lose $1/5$ life.

To explore a more restrictive setting, we additionally examine the single-pixel ($n = 1$) attack on limited frames. We illustrate the results on Breakout in Figure 3.10, and similar trends can be observed on other games as well. From this figure, we find that by only attacking six frames on average, the policy trained by DQN is totally fooled with reward decreasing from 224 to 13.7. Here, six frames correspond to an attack ratio of $6/535 \approx 1\%$, where 535 is the averaged number of total frames. Similar findings are also observed on policies trained by PPO2, A2C and ACKTR. For instance, the policy trained by PPO2 shows a significant reward decline with only 27 frames attacked under the same TCA setting as other policies.

Figure 3.9: The results of adversarial attack with different TCA threshold ζ^* , where the line and shaded area illustrate the mean and standard deviation of 30 independent runs respectively.

3.4.4 Attack Initialization Analysis

The GA population initialization plays an important role in efficiently generating successful adversarial examples, and thus highlights the practicality of the approach. To this end, we study the effects of random initialization versus warm starting based on prior data. In RL, the simplest form of prior data originates from the correlation between states in a Markov Decision Process (MDP); this also distinguishes RL from other supervised learning tasks (e.g., classification), where the predictions on different data points are independent. Thus, with the existence of correlations between adjacent frames in mind, we set up different types of population initializations in the GA: (1) **GA-RI**: random initialization (RI) of pop_0 ; (2) **GA-WSI**: warm start initialization (WSI) of pop_0 by inserting the optimal adversary x_p^* found in the previous attacked frame.

We randomly attack four frames for all the six games. On each attacked frame, the GA optimization is executed 10 times independently on both types of initialization. The comparison results are shown in Figure 3.11. In general, we clearly see that GA-WSI

Figure 3.10: The impact of attacked frames for accumulated reward on Breakout

significantly speeds up the optimization convergence in finding successful adversarial examples (i.e., corresponding to positive objective function values). In particular, on many frames (e.g., Pong: ξ_1 , ξ_2 , ξ_4 ; Breakout: ξ_2 ; Space Invaders: ξ_3 ; Qbert: ξ_3 , ξ_4), the GA-RI takes significantly more time (or is even unable) to obtain a positive objective value; whereas GA-WSI achieves it easily.

*We interestingly find that, in some cases, GA-WSI provides a successful adversary without the need of any optimization. This observation underlines the real possibility of launching successful adversarial attacks on deep RL policies in **real-time** at little/no computational cost, a threat that raises severe concerns about their deployment especially in safety critical applications.* As such, it points to two critical directions for future research. On the one hand, there is a need for more comprehensive exploration of procedures of learning from experiential frames, that could lead to the generation of potent adversaries on-the-fly. This motivates the study on frame correlation transfer in Chapter 4. On the other hand, a significant void in the trustworthiness of existing RL policies is revealed, one that could possibly be addressed through more robust training algorithms.

Figure 3.11: The comparison between random initialization (GA-RI) and warm start initialization (GA-WSI), where the line and shaded area illustrate the mean and standard deviation of 10 independent runs respectively.

3.5 Summary

This chapter explores *minimalistic scenarios* for adversarial attack in deep reinforcement learning (RL), comprising (1) black-box policy access where the attacker only has access to the input (state) and output (action probability) of an RL policy; (2) fractional-state adversary where only a small number of pixels are perturbed, with the extreme case of one-pixel attack; and (3) tactically-chanced attack where only some significant frames are selected to be attacked. We verify these settings on policies that are trained by four state-of-the-art RL algorithms on six Atari games. We surprisingly find that: (i) with only ***a single pixel*** ($\approx 0.01\%$ of the state) attacked, the trained policies can be significantly deceived; and (ii) by merely attacking around 1% frames, the policy trained by DQN is totally fooled on certain games. Our immediate future work will focus on further investigating different initialization schemes that could better exploit the correlations across different attacked frames. Our goal is to thoroughly analyze the plausibility of generating adversarial attacks in real-time for time-sensitive RL applications.

Chapter 4

Frame-Correlation Transfers Trigger Economical Attacks on Deep Reinforcement Learning Policies

As shown in Chapter 2.2, most previous adversarial attack generation procedures [39, 47] are based on the input gradient that is obtained by back-propagating through a neural network policy, where the basic assumptions are 1) the neural network is transparent to an attacker, and 2) the adversarial perturbation can be added on every pixel of the input. However, these assumptions do not always hold in many real-world systems.

In Chapter 3, a minimalistic attack with the afore-stated strong assumptions being relaxed [70] has been investigated. In particular, a genetic algorithm (GA) [37] is used for the generation of selective perturbations (on selective pixels and frames), treating the policy as a black-box. The method does not require the weights and structure of the evaluated policy to be known. However, it is noteworthy that in this attack generation process on RL, the GA is executed *from scratch* on every frame (i.e., time-step). Therefore, the computation complexity involved may still preclude the practical realization of such attacks in time-sensitive RL tasks, since the limited time-window between consecutive frames would not allow a time-consuming attack generation.

To alleviate this issue, our central thesis of this chapter¹ is that the exploitation of

¹The work in this chapter has been published in IEEE Transactions on Cybernetics, 2021

correlations (similarities) between frames (through attack transfer) could greatly speed up, and thus increase the plausibility of, adversarial attacks in RL. In contrast to previous approaches, we induce different types of *frame-correlation transfers* (each of differing degrees of computational complexity) as ways of boosting the efficiency of the adversary generation, with the goal of highlighting the threat of real-time attack realizability in *image-based* RL applications. Note that, the frame correlations reflect the consistency of environmental dynamics. Inherent similarities are naturally expected to exist among frames and their corresponding adversarial perturbations. Moreover, the presence of such correlations marks a clear distinction of RL from supervised learning tasks (e.g., classification), where the inputs are typically assumed to be i.i.d (that is, *independently* and identically distributed).

In doing so, the following design choices have been made. **(1)** We formulate the adversary generation task as a derivative-free optimization problem that is solved by a population-based GA [37]. We involve an optimization formulation with a novel objective function that ensures to successfully deceive the DNN policy if a positive objective function value is found. **(2)** The optimization variables are specified as the discrete 2-D coordinate location(s) and perturbation value(s) of the selectively attacked pixel(s). Further, we utilize the Shannon entropy of the action distribution to keep only selective frames to be attacked. **(3)** We induce three types of frame-correlation transfers with increasing computational complexity: *anterior case transfer* that simply reuses the generated adversary from the most recent previously attacked frame as a related source solution; *random projection based transfer* that reuses adversaries from the most similar previously encountered source frame identified via a random low-dimensional mapping of the input space; *principle components based transfer* that reuses adversaries from the most similar source frame identified via optimized (variance maximizing) low-dimensional linear projections of the input space.

Our main contributions of this chapter can be summarized as:

- For the first time, we show that the presence of correlations between frames (as suggested by the consistency of RL environmental dynamics) can be exploited to significantly boost attack generation.
- We induce three types of FCTs (i.e., anterior case transfer, random projection based transfer, and principal components based transfer) with varying degrees of computational complexity in generating adversaries.
- Based on extensive experimental comparisons on six well-known Atari games and four representative RL algorithms, we validate that through the use of FCTs, the adversarial examples can surprisingly be generated for free, i.e., the number of policy queries required to generate successful adversaries is nearly *zero*. This indicates that our attack can be approximately launched in *real-time* at *little computational cost*. Our results thus highlight the severe vulnerability of existing DRL policies.

4.1 Motivations

In this part, we present an overview of derivative-free attack generation using genetic algorithm, and its potential enhancement with the notion of knowledge transfer in optimization.

4.1.1 Derivative-Free Attack Generation

To generate the attacks without accessing a policy’s structure and weights, the creation of δ_t is normally formulated as a derivative-free optimization problem. Specifically for image based RL tasks (e.g., video games), such formulation is finding the optimal discrete variables (e.g., coordinates and perturbation values on particular pixels) that produce noisy images to fool the trained policies. In solving such derivative-free optimization

problem, genetic algorithm (GA) [37, 85] is a commonly used approach. In Chapter 3, we have investigated the feasibility of applying GA to generate perturbations only on selected pixels.

To obtain the optimal $\mathbf{x}_t, \mathbf{y}_t, \mathbf{p}_t$, we utilize a simple genetic algorithm (GA) [37, 7, 18] (i.e., illustrated in Algorithm 1 from line 7 to 17) for the black-box optimization. In GA, a population *pop* containing λ individuals is evolved, where each individual $\delta_t^i, i \in [1, \lambda]$ is a candidate adversarial example that is evaluated by the objective function in Eq. 3.7. The top- $\lambda\beta$ individuals with respect to the objective value are selected as elites to survive in the new generation (line 10). Based on standard crossover (line 12) and mutation (line 13) operations, another population *pop'* is reproduced to replace the old *pop*. This process repeats until a specified maximum number of function evaluations M^{eval} is reached.

However, in Chapter 3, we did not comprehensively study the potential impact of exploiting similarities between optimization, arising from the correlation between frames. We note that these correlations reflect the consistency of environmental dynamics in MDPs, and thus are expected to naturally hold transferable information across frames. Incorporating such information into derivative free attack generation promises to significantly accelerate the convergence behavior of the optimization process [34]. This naturally inspires us involving the transfer optimization into the attack generation in DRL.

4.1.2 Knowledge Transfer in Optimization

As opposed to traditional optimization solvers that start a search from scratch assuming zero prior knowledge about the task at hand, the notion of knowledge transfer [33, 16] in optimization puts forth the idea of leveraging data acquired from problems that have been solved in the past to accelerate the search in an ongoing (target) task of interest. Such *transfer optimization* [34, 20] is inspired by the similar observations of human-like

Figure 4.1: The flowchart of sequential transfer in optimization.

problem-solving as motivated the relatively well-known field of *transfer learning* [90, 91, 3] - where a predictive model utilizes available data from one or more related source tasks to improve the approximation of a target function given only limited target data. However, the emerging topic of transfer optimization is different in the sense that it focuses on improving the efficiency of finding optimal *prescriptions* based on prior (source) data, rather than being limited to predictive modeling.

For the kind of applications considered in this chapter, the derivative free optimization based adversary generation is to be carried out one after the other with respect to input frames that appear sequentially. Therefore, we focus on the paradigm of *sequential transfer* that is illustrated in Figure 4.1, based on its formalization in [34]. A basic assumption in sequential transfer is that the source tasks have already been addressed before tackling the target task. As a result, the extracted information from all source tasks is already available in a knowledge database, with the key challenge being to define a knowledge transfer operation that is able to efficiently make use of this (possibly large) database. With regard to image-based DRL applications in particular, the source knowledge consists of previously encountered input frames and their corresponding optimized adversarial perturbations. Thus, in this domain, efficient transfer can be achieved

Figure 4.2: A schematic for retrieving pre-computed adversarial perturbations for transfer across similar frames. A low-dimensional feature extraction can enable efficient source retrieval.

through the low-cost retrieval of the most similar source frames, such that their derived perturbations could be directly reused in the attacked target frame.

4.2 Frame-Correlation Transfer (FCT)

In the basic attack methodology, the adversary is generated from scratch without leveraging frame-correlations in MDP, resulting in the demand for high computation time and resources in solving the black-box optimization problem of Eq. 3.7. Therefore, such attacks may not pose a realistic threat to most time-sensitive RL systems, as the short time window of state transition would preclude any extensive computation for adversary generation.

To analyze more realistic attacks that could lead to tangible threats even to time-sensitive RL systems, in this chapter we study the potential to speed up the adversary generation process by exploiting what we refer to as *frame-correlation* transfers. The basic rationale behind our approach is based on the fact that correlations between input frames reflect the consistency of environmental dynamics. Therefore, inherent similarities are naturally expected to exist across different frames, and by extension, between their corresponding adversarial perturbations as well.

The argumentation above highlights the importance of examining the vulnerability of DRL policies to the direct transfer and reuse of pre-computed adversarial attacks across

Figure 4.3: The anterior case transfer (AC-T).

similar frames. To this end, we consider a realization of the notion of sequential transfer in optimization, as illustrated in the context of adversarial attacks in Figure 4.2. In particular, for generating the adversary δ_{k+1} on frame s_{k+1} (marked as *target* task), the data acquired from previous frames $\{s_1, s_2, \dots, s_k\}$ and their corresponding generated adversaries $\{\delta_0, \delta_1, \dots, \delta_k\}$ (marked as *source* tasks) is taken into account. The key challenge is to define a knowledge transfer strategy that retrieves similar source frames as efficiently as possible (bearing in mind the real-time nature of many RL applications), after which the corresponding generated adversaries could be directly reused in the attacked target frame s_{k+1} .

We investigate *three different transfer strategies* with increasing levels of computational complexity. The first is a simple rule-based transfer strategy, while the other two involve the extraction of low-dimensional features enabling similarity measurements for efficient retrieval of the most relevant high-dimensional source frames. These strategies are detailed next.

4.2.1 Anterior Case Transfer (AC-T)

AC-T is a rule-based strategy that directly reuses the generated adversary from the most recent attacked frame as the related source. The basic idea of AC-T is shown in Figure 4.3; to generate the adversary δ_{k+1} on frame s_{k+1} , the adversary δ_k computed on the

immediately preceding frame s_k is directly injected as an initialization for derivative-free adversary generation.

The assumption of AC-T is that neighboring frames share larger similarity than non-neighboring ones. Note that in an MDP, the frame s_{k+1} is controlled by a conditional transition function $T(s_{k+1}|s_k, a_k)$ [83]. Although the transition function is hidden in the environment and unknown to the agent, it nevertheless implies a close relationship between s_k and s_{k+1} . We visualize this relationship in Fig.4.3, where s_k and s_{k+1} are largely the same with only a slight difference in the positions of the ball and the paddle. Based on this rationale supporting the AC-T setting, the adversary δ_k is transferred as a promising initialization in the population of a genetic algorithm (GA) optimizing δ_{k+1} on the current frame s_{k+1} .

AC-T, though conceptually simple and straightforward for an attacker to implement, is limited in the sense that it only considers the adversarial perturbation from the proceeding frame s_k to be transferred. As such, it totally ignores a large number of other historical frames $\{s_1, s_2, \dots, s_{k-1}\}$ that have previously been attacked and may bear similarity with the current target frame of interest. A more comprehensive approach could thus be to mine the most similar historical frames from the available source knowledge base, such that their corresponding pre-computed attacks could then be transferred for reuse (as illustrated in Figure 4.2).

4.2.2 Random Projection based Transfer (RP-T)

RP-T considers the transfer of adversarial perturbations from the most similar previously encountered source frame identified via a low dimensional mapping produced by a random projection model. Random projection [27] (RP) is a *training free* approach that is computationally efficient yet sufficiently accurate for low dimensional feature extraction of otherwise high dimensional data. It is worth noting that the cost of RP is significantly

lower than other well-known dimensionality reduction approaches (e.g., principle component analysis and singular value decomposition), yet it can serve to extract effective features that sufficiently measure the actual distances between data points with little distortion [2].

A schematic of RP-T is shown in Figure 4.4, where RP is randomly generated. To elaborate, we consider the low dimension projections $\mathbf{s}^f = \{s_1^f, s_2^f, \dots, s_k^f\}$ of the original states $\mathbf{s} = \{s_1, s_2, \dots, s_k\}$, where k is the size of the database of historically attacked frames. The dimensions of $\mathbf{s}$ and $\mathbf{s}^f$ are d and d_f respectively, where we assume $d_f \ll d$. The rationale behind the effectiveness of the generated low-dimensional features, with regard to their use in Euclidean distance based similar source image retrieval, is supported by the following Johnson and Lindenstrauss (JL) Theorem [2].

(Johnson and Lindenstrauss Theorem). *Given any $\epsilon \in (0, 1)$ and any integer k , let d_f be a positive integer such that,*

$$d_f \geq 4(\epsilon^2/2 - \epsilon^3/3)^{-1} \ln(k).$$

Then for any set $\mathbf{V}$ of k data points in $\mathbf{R}^d$, there is a map $g : \mathbf{R}^d \rightarrow \mathbf{R}^{d_f}$ such that for all $u, v \in \mathbf{V}$,

$$(1 - \epsilon) \|u - v\|^2 \leq \|g(u) - g(v)\|^2 \leq (1 + \epsilon) \|u - v\|^2$$

Furthermore, this map can be found in randomized polynomial time.

An elementary proof of JL theorem can be found in [17]. Simply put, JL theorem suggests that if data points of high dimensionality are mapped randomly to a reduced subspace, the distance between points will be *approximately* preserved. In our proposed RP-T, we thus utilize a Gaussian random matrix $\mathbf{M}_{d_f \times d}$ to linearly project the original

Figure 4.4: The random projection based transfer (RP-T).

d -dimensional states $\mathbf{s}$ to d_f -dimensional features. The projection can be written as,

$$\begin{aligned}
 s_i^f &= g(s_i) \\
 &= \mathbf{M}_{d_f \times d} s_i.
 \end{aligned}
 \quad (\text{Eq. 4.1})$$

$$\mathbf{M}_{d_f \times d} = \begin{bmatrix} m_{1,1} & m_{1,2} & \cdots & m_{1,d} \\ m_{2,1} & m_{2,2} & \cdots & m_{2,d} \\ \vdots & \vdots & \vdots & \vdots \\ m_{d_f,1} & m_{d_f,2} & \cdots & m_{d_f,d} \end{bmatrix}, m_{i,j} \sim \mathcal{N}(\mu, \sigma^2),$$

where each entry of the random matrix $\mathbf{M}_{d_f \times d}$ is drawn from a gaussian distribution $\mathcal{N}(\mu, \sigma^2)$ with mean μ and standard deviation σ . In the original d -dimensional space, we would write the Euclidean distance between the input state of the current target task s_{k+1} , and any prior $s_i, i \in \{1, 2, \dots, k\}$ in the source knowledge base as $\|s_{k+1} - s_i\|^2$. Following the elementary proof of the JL theorem in [17], there is a *sizeable probability* that $\|s_{k+1} - s_i\|^2$ when transformed to an appropriate low-dimensional space shall satisfy,

$$\left\| s_{k+1}^f - s_i^f \right\|^2 = \beta \cdot \frac{d_f}{d} \|s_{k+1} - s_i\|^2, \quad (\text{Eq. 4.2})$$

where $\beta \in [1 - \epsilon, 1 + \epsilon], \epsilon \in (0, 1)$.

Figure 4.5: The principle components based transfer (PC-T).

This result comes from derivations showing that the distribution of β is sharply concentrated around its mean, where $\beta_{mean} = 1$, with the probability of β lying outside $[1 - \epsilon, 1 + \epsilon]$ being $O(1/k^2)$. Accordingly, given an appropriate dimension d_f , we can obtain a random projection that suffices for us to retrieve a sufficiently similar source frame by solving,

$$i^* = \arg \min_{i=1}^k \left\| s_{k+1}^f - s_i^f \right\|^2. \quad (\text{Eq. 4.3})$$

Given the optimized index i^* , the adversary δ_{i^*} is then transferred as a potentially potent attack for initialization of the GA population optimizing δ_{k+1} on the new state s_{k+1} .

4.2.3 Principle Components based Transfer (PC-T)

Although RP-T is indeed computationally efficient, the randomization (of the projection matrix) involved inevitably forfeits any assurance of effective source retrieval. PC-T on the other hand transfers adversarial perturbations from the most similar source frame identified by projecting the data to an optimized (variance maximizing) subspace identi-

fied via principle component analysis (PCA). The schematic of PC-T is shown in Figure 4.5, where an optimized matrix $\mathbf{M}_{d_f \times d}^*$ is now learned from the accumulated source database (i.e., states $\mathbf{s} = \{s_1, s_2, \dots, s_k\}$) using PCA [45]. Note that an assumption of the database for the three transfers (i.e., AC-T, RP-T and PC-T) is that source frames gradually accumulate with more successful attacks found. Namely, at the beginning, the optimization of finding successful attacks starts from scratch, which may take a number of iterations. With more successful attacks recorded, our designed frame correlation transfers can speed up the optimization; thus finally reduce the computation on average.

In PCA, given k data points of input states $\mathbf{s}$, as a $k \times d$ matrix, we want to find the best d_f -dimensional feature approximation for the states $\mathbf{s}^f$, such that the ground truth distance between data points can be optimally preserved in the reduced dimensional space. To this end, PCA leverages orthogonal transformation [44] to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. PCA is an optimal data projection in the mean-square sense [8]; the squared error introduced in the projection is minimized over all projections onto a d_f -dimensional subspace.

The pseudo code of PCA is provided in Algorithm 2, where $\mathbf{M}_{d_f \times d}^*$ is constructed by selecting the eigenvectors with top- d_f eigenvalues. Based on $\mathbf{M}_{d_f \times d}^*$, low-dimensional feature vectors $\{\mathbf{M}_{d_f \times d}^* s_1, \mathbf{M}_{d_f \times d}^* s_2, \dots, \mathbf{M}_{d_f \times d}^* s_k\}$ can be extracted. Accordingly, the best source task index i^* is determined by minimizing the Euclidean distance in the transformed space, as given by,

$$i^* = \arg \min_{i=1}^k \left\| \mathbf{M}_{d_f \times d}^* s_{k+1} - \mathbf{M}_{d_f \times d}^* s_i \right\|^2, \quad (\text{Eq. 4.4})$$

Thereafter, with the optimized source state index i^* , the corresponding adversary δ_{i^*} is transferred as a promising starting point for the adversary generation of δ_{k+1} on the target state s_{t+1} .

Algorithm 2: Principal Component Analysis

- Input:** $\mathbf{s} = \{s_1, s_2, \dots, s_k\}^T$ – Source data; d_f – the dimensionality in feature space.
- 1 Normalize the data: $s_i \leftarrow s_i - \frac{1}{k} \sum_{j=1}^k s_j$;
 - 2 Calculate the co-variance matrix $\mathbf{ss}^T$;
 - 3 Calculate eigenvalue decomposition of $\mathbf{ss}^T$ to obtain the eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_k$;
 - 4 Get the eigenvectors $(\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_{d_f})$ that correspond to the largest d_f eigenvalues.
 - 5 **Return** the matrix $\mathbf{M}_{d_f \times d}^* = [\mathbf{m}_1; \mathbf{m}_2; \dots; \mathbf{m}_{d_f}]$
-

4.2.4 Increasing Computational Complexity of the Three FCTs

Our main objective behind introducing three different types of FCTs is to showcase the trade-off between their respective computational complexities and the resulting transfer potency in attacks towards DRL policies. On one hand, there are potential applications of RL in (for example) the domains of medical diagnosis and treatment [59] where there is a reasonably large time window between state transitions such that a computationally intensive but powerful adversary could intervene to misdirect intermediate actions. At the other extreme are RL applications of a highly time-sensitive nature where actions in response to input signals must be taken within a fraction of a second (e.g., steering / breaking control in a self-driving vehicle when a sudden obstacle appears), automatically precluding adversary generation of high computational complexity. There will of course be a spectrum of possibilities between these two extremes where an adversary may choose to compromise (to varying degrees) on the guarantee of attack potency in favour of timeliness of the attack. With that in mind, the comparison of computational complexities of the three different FCTs are shown in Table 4.1.

AC-T marks the simplest case, due to its rule-based transfer methodology that directly reuses the prior state as the optimal source task. Naturally, the cost of retrieval does not scale with the size of the source knowledge base, thus the constant computational complexity is $\mathcal{O}(1)$. Such a transfer rule is however highly limiting, as it entirely

Table 4.1: Comparison of computational complexities of FCTs.

FCT type	Computational complexity
AC-T	$\mathcal{O}(1)$
RP-T	$\mathcal{O}(d \cdot d_f + k \cdot d_f)$
PC-T	$\mathcal{O}((k \cdot d^2 + d^3) + (k + 1)d \cdot d_f + k \cdot d_f)$

ignores the database of previously seen and attacked source frames. In comparison, RP-T has a higher computational complexity of $\mathcal{O}(d \cdot d_f + k \cdot d_f)$, but provides a way to take advantage of the available source database; recall that $d_f \ll d$. In our implementation of RP-T, the random projection matrix $\mathbf{M}_{d_f \times d}^*$ is generated once and then fixed, such that when a target task occurs, the source frames have already been projected to the low d_f dimensional space. Thus, the computational complexity only comes from two components: the projection of the target frame (i.e., $d \cdot d_f$) and from Euclidean distance measurements in the reduced subspace (i.e., $k \cdot d_f$). As per JL theorem, RP-T possesses a practically useful probabilistic performance guarantee, but it is yet not theoretically optimal due to the randomization of the projection matrix. It is this lacuna that is filled by PC-T, which utilizes optimized (variance maximizing) low-dimensional feature extraction in order to further enhance the efficacy of source retrieval. However, PC-T has much higher computational complexity as a significant cost to be paid in order to ensure its optimality; every-time a new frame is attacked and added to the source database, the principle components may be re-learned, requiring the newly encountered target as well as the prior source frames to be projected into the newly derived subspace. Therefore, the time complexity of PC-T would scale as $\mathcal{O}((k \cdot d^2 + d^3) + (k + 1) \cdot d \cdot d_f + k \cdot d_f)$, where the three components $(k \cdot d^2 + d^3)$, $(k + 1)d \cdot d_f$ and $k \cdot d_f$ come from PCA model learning, source & target frame projections, and Euclidean distance measurements in the reduced subspace, respectively.

Table 4.2: The structure of the policy network.

Input s_t :	84 * 84 * 4 images
layer 1	Convolution $8 \times 8 \times 32$; Stride 4; Relu; Batch norm
layer 2	Convolution $4 \times 4 \times 64$; Stride 2; Relu; Batch norm
layer 3	Convolution $3 \times 3 \times 64$; Stride 2; Relu; Batch norm
layer 4	Fully-connected layer 512; Relu
Output $\mathcal{A}(\pi, s_t)$:	Dimension of actions; softmax.

4.3 Experiments

To demonstrate the potency of the investigated frame-correlation transfers (i.e., AC-T, RP-T and PC-T), we examine four fully trained DRL policies (i.e., DQN [55], PPO2 [77], A2C [54], ACKTR [93]) on six Atari games (i.e., Pong, Breakout, SpaceInvaders, Seaquest, Qbert, and BeamRiderare) in OpenAI Gym [9].

4.3.1 Experiment Setup

Table 4.2 shows the architecture of the policy network, which is kept same for PPO2, A2C and ACKTR. For DQN trained policy, we add an additional softmax operation on the output layer to normalize the predicted Q values into $[0, 1]$. The input state s_t of the policy network is the concatenation of the latest four gray scale screen images; each image is pre-processed to a resolution of 84×84 . Such pre-processing is referred from [55], and kept the same for all the policies. The integer pixel value is in the range $[0, 255]$, and it is linearly scaled to the range $[0, 1]$. The output of the policy is the distribution $\mathcal{A}(\pi, s_t)$ in action space for PPO2, A2C, ACKTR, and an estimation of Q values for DQN. To carry out a fair comparison, all the policies (π) are obtained from the RL baseline zoo², where each policy is trained for 1×10^7 steps. The constraints in Eq. 3.7 are set as

²<https://github.com/araffin/rl-baselines-zoo>

$\mathcal{I}_x = 84$, $\mathcal{I}_y = 84$, $\mathcal{I}_p = 1$, respectively. A simple genetic algorithm (GA) is chosen as the derivative-free optimizer, in which the maximum objective evaluations, population size, selection rate and mutation rate are set as 400, 10, 0.2 and 0.1, respectively. To get the knowledge base (that stores state s_t and the corresponding successful adversarial perturbation δ_t) as source data for our proposed FCTs, we first run the NO-T to get 500 successful attacks. Thereby, these stored data will behave as the source data. During each FCT running, if further successful attacks are found, they will be stored to augment the source data.

Before investigating the potency of FCT, we first validate the basic attack generation methodology in Chapter 4.2. In doing so, we analyze the impact of n in Eq. 3.6 by varying it from 1 to 10. We also analyze the impact of different proportions of attacked frames by varying the value of the prescription entropy threshold ζ_t^* in $[0, 1]$. For FCT, we analyze the impact of the three proposed transfer strategies (i.e., AC-T, RP-T and PC-T) on the convergence speed (measured in terms of the number of model queries required to generate a successful untargeted attack) of adversary generation. In the FCT experiments, we set $d_f = 20$ for both RP-T and PC-T. Moreover, in generating the random matrix in Eq. 4.1, the mean μ and standard deviation σ of Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$ are set to be 0 and 0.05 respectively, following the setting in the reference [50].

4.3.2 Convergence Trends Under Frame-Correlation Transfer

Based on the basic attack generation methodology, we study the potency of frame-correlation transfers (FCTs) on six Atari games. Four different strategies are compared in this chapter, including: (1) no transfer (NO-T), (2) anterior case transfer (AC-T), (3) random projection based transfer (RP-T), and (4) principal components based transfer (PC-T). NO-T means that the derivative-free GA starts the attack generation from scratch without being aided by any biased initializing from source frames. On each game,

we show the optimization convergence trend on four randomly selected frames. To make a trustworthy comparison, on each attacked frame, we repeat the GA optimization 30 times for each strategy. GA is a stochastic search algorithm, thus multiple runs are needed to derive statistically significant results. In the evolutionary computation literature [102, 72, 81], 30 independent runs are often deemed as a standard for providing a robust comparison on the convergence characteristics of the algorithm. Therefore, to provide a fair and trustworthy comparison, we have carried out 30 independent runs throughout. The results are shown in Figure 4.6, where the line and shaded area indicate the value of mean and standard deviation, respectively. In Figure 4.6, $\xi_1, \xi_2, \xi_3, \xi_4$ are four frames that we randomly select to be attacked. All the policies on the six games are trained by DQN. Note that *NO-T* [70] represents the derivative free generation of adversarial attacks but without transfer.

From Figure 4.6, we observe that the convergence speed of NO-T is generally much slower than the other three FCTs. This shows that incorporation of FCTs can indeed dramatically speed up the generation of potent adversaries. Notably, PC-T and RP-T are found to converge at a much faster rate than AC-T. What is more, PC-T performs more effectively than RP-T in the majority of cases, with this being attributable to its optimized feature extraction (albeit at a significantly higher computational cost of having to first learn this feature extractor).

We also notice that in most cases (e.g., Pong ξ_3, ξ_4 , Breakout ξ_1 - ξ_4 , BeamRider ξ_1 - ξ_4), the initial objective function from PC-T and RP-T are already positive, which indicates that our FCTs are in fact capable of generating successful untargeted adversarial attacks (as discussed in Section IV-A) without the need for repeatedly running an optimization search procedure. These observations demonstrate that FCTs could indeed lead to the very real threat of powerful adversarial attacks that have been generated on-the-fly, thus revealing the practical risk even to time-sensitive RL applications.

Figure 4.6: The comparison of convergence trends of: 1) NO-T (no transfer), 2) AC-T (anterior case transfer), 3) RP-T (random projection based transfer), and 4) PC-T (principal components based transfer).

Figure 4.7: The comparison of policy queries needed to generate successful untargeted attacks using different types of frame-correlation transfers. Note that *NO-T* [70] represents the derivative free generation of adversarial attacks but without transfer.

4.3.3 Efficiency Analysis of Frame-Correlation Transfers

In order to further highlight the aforementioned threat, we compare the exact number of policy queries needed for generating successful adversaries via the different FCTs (i.e., NO-T, AC-T, RP-T and PC-T). We recall that a successful candidate adversary is one that produces a positive objective function value in the formulation of Eq. 3.7 during the course of the GA search. On each type of FCT, we repeat the GA run 30 times. The statics and swarm plot are shown in Figure 4.7, where the bar indicates the mean value. Notably, the cost of attack generation indicates the budget on computing the adversarial attack δ_t . Therefore, in our FCT based attack, the computational cost is quantified as the calculation after one particular attack on historical frames is injected. In other words, if one injected attack δ_t can directly fool a policy, the computational cost is deemed to be 0 since no additional computation is needed.

Generally from Figure 4.7, we find that the policy queries required by our proposed FCTs are consistently far less than that of NO-T (i.e., without transfer). This once again demonstrates that FCTs significantly speed up the convergence rate of adversary generation. Even for the simplest AC-T, the numbers of policy queries (i.e., 82.26, 0.0, 49.35, 0.0, 3.23, 0.0 on the six games, respectively) are considerably lower than that of NO-T (i.e., 261.29, 43.67, 185.81, 50.0, 114.19, 63.55). However, the average number of policy queries in AC-T is much higher in turn than RP-T and PC-T. Among the three FCTs, PC-T is found to be the most effective in retrieving relevant source frames, so much so that it takes *zero* policy queries in five out of the six games to produce a successful untargeted attack. In contrast, RP-T consumes zero policy queries in four out of the six games, while AC-T is able to achieve similar success in three of the games. Despite the clear superiority of PC-T (as revealed in Figure 4.7), we must bear in mind that this form of transfer comes at a higher computational cost in learning optimized features for source-target similarity measurement. This cost may however not always be feasible in

time-sensitive RL applications, leading an attacker to consider RP-T and AC-T which can also produce strong (yet efficient) adversaries. Nevertheless, our observations shed light on the general fact that minimalistic adversarial attacks on DRL policies can be nearly launched in *real-time* at little computational cost. Furthermore, Figure 4.7 suggests that the attacks on different games have different level of difficulties. For instance, on Pong, all four evaluated scenarios need more iterations compared to other games. This makes it the only one game (out of the six tested) that does not have the zero policy query for PC-T.

4.3.4 Computation Time Analysis

In order to demonstrate the efficiency of our proposed frame correlation transfers, we further compare the computation time on generating adversarial perturbations with different attack approaches. We utilize each attack approach to generate adversarial perturbations on 200 different frames, and calculate their averaged time of attack generation. Note that, all the evaluation is based on Intel(R) Xeon(R) CPU E7-8880 v4 @ 2.20GHz. The implementations of FGSM, PGD and CW attack are from IBM Adversarial Robustness Toolbox [60]. The computation time of our FCT based attacks is a summation of 1) the attack transfer (i.e., the distance calculation in feature space for source retrieval) and 2) additional steps of optimization needed to find a successful untargeted attack. In our implementation, the construction of the data projection matrix for RP-T as well as PC-T is carried out offline (i.e., before the target frame is observed). Therefore, the online computation for RP-T and PC-T include 1) feature extraction of the target frame using the existed model, and 2) calculating the nearest source frame in the low-dimensional feature space. Note that the latter operation can be easily parallelized to further reduce the cost of transfer. According to the results shown in Table 4.3, the computation time of our FCT (i.e., AC-T, RP-T and PC-T) based attack generation is significantly less than

Table 4.3: Computation time comparison on generating adversarial perturbations with different methods. FGSM is the fast gradient sign method [39]; PGD-4, PGD-10 and PGD-20 represent the projected gradient descent with iteration number being 4, 10 and 20 respectively; CW indicates the Carlini & Wagner attack [11]; NO-T represents the minimalistic attack [70]. Note that, the value in the table is calculated as the averaged time for generating adversarial perturbations on 200 frames. The smallest and second smallest values are highlighted by bold underline and bold, respectively. The last row indicates the average time across the six tested games.

Games	Attack approaches (time unit is second)								
	AC-T	RP-T	PC-T	NO-T [70]	FGSM [39]	PGD-4 [51]	PGD-10 [51]	PGD-20 [51]	CW [47]
Pong	0.0549	0.0342	0.0331	1.9009	<u>0.0096</u>	0.0171	0.0358	0.0672	2.2169
Breakout	<u>0.0000</u>	0.0004	0.0005	0.6545	0.0086	0.0163	0.0338	0.0633	2.2375
SpaceInvaders	0.0053	<u>0.0003</u>	0.0004	1.8950	0.0088	0.0163	0.0337	0.0624	2.2500
Seaquest	0.0253	0.0038	<u>0.0030</u>	1.3048	0.0090	0.0166	0.0346	0.0652	2.2349
BeamRider	<u>0.0032</u>	0.0051	0.0062	0.8891	0.0095	0.0170	0.0348	0.0656	2.2642
Qbert	0.0045	0.0031	<u>0.0006</u>	1.1606	0.0095	0.0169	0.0348	0.0654	2.3054
<i>Average</i>	0.0155	0.0078	<u>0.0073</u>	1.3008	0.0092	0.0167	0.0346	0.0649	2.2515

other start-of-the-arts. In comparison with NO-T (i.e., GA optimization of adversarial perturbation but without transfer), our AC-T enables real-time attack generation (i.e., computation time required being nearly zero) on Breakout. Furthermore, on most games (i.e., Breakout, SpaceInvaders, BeamRider and Qbert), our FCT based attack generation is an order of magnitude faster than most of the gradient based attack generation approaches (e.g., PGD-10 and PGD-20). The last row of Table IV underscores how, on average across all games, the cost of optimization (in NO-T and AC-T) outweighs the relatively small cost of relevant source retrieval and transfer (in RP-T and PC-T). These comparisons demonstrate again that our FCT based adversarial attacks on DRL policies could be launched in **real-time** at *little computational cost*, which raises severe safety concerns about their deployment especially in safety-critical applications.

4.4 Summary

This chapter unveils how transferability across frames can be exploited for the efficient creation of minimal yet powerful adversarial attacks in image-based RL applications. We propose and analyze three types of frame-correlation transfers with increasing levels of computational complexity (i.e., anterior case transfer, random projection based transfer, and principal components based transfer) to generate small-scale perturbations via derivative-free methods. It is worth noting that it is the presence of such exploitable correlations that marks a key distinction of RL from the growing literature on attacks towards supervised learning models, where the inputs are typically assumed to be i.i.d. We carry out experiment studies on six Atari games and four RL training algorithms with particular emphasis on showcasing the trade-off between computational complexity of the attack and its resultant potency. Most cases surprisingly (and perhaps alarmingly) indicate that frame-correlation transfers significantly accelerate the adversary generation process, so much so that often *zero* black-box policy queries are needed to come up with a

successful adversary. Our research therefore reveals the practical safety risk (against adversaries) posed even in time-sensitive RL applications, motivating future research efforts towards enhancing the resilience of state-of-the-art DRL policies. Given the vulnerability revealed by this chapter and Chapter 3, our following target is to increase the robustness of DRL policies with the goal of resisting the unknown attacks at test time.

Chapter 5

Adversary Agnostic Robust Deep Reinforcement Learning

Previous studies [25, 99] as well as Chapter 3 and Chapter 4 have shown that DRL policies can be deceived by perturbations on state observations that appear at test time but are unknown during training. To enhance the adversarial robustness of DRL policies, many studies have investigated adding the adversaries into the training process to achieve the corresponding generalization ability on these perturbed observations. For instance, Mirman et al. [53] and Fischer et al. [25] utilized adversarial training [67] based policy distillation [74] to obtain an accurate and robust student policy, where the robustness is learned from the added adversarial attacks that are generated by fast gradient sign method and projected gradient descent, respectively. Recently, based on a derivation of state-adversarial Markov decision process (SA-MDP), Zhang et al. [99] proposed a robustness regularization to align the policy’s prescriptions *with* and without adversarial perturbation; thus still being adversary dependent. In a nutshell, these approaches require additional procedures of incorporating adversaries in the training, which, however, makes improvements on robustness less flexible (e.g., robust policy depending on one particular attack may fail when another attack happens), and more expensive from both computation (in calculating the adversaries) and implementation (in deploying adversaries for training in the wild) perspectives.

Instead of learning robust policy from adversaries, under the policy distillation (PD) paradigm, we prove in Chapter 5.2 that such robustness can also be achieved without relying on adversaries¹. Specifically, a robust student policy can be learned through maximizing the student policy’s prescription gap between the teacher selected action and the remaining actions under attack. Most importantly, we further derive that maximizing the prescription gap under attack can be transformed to simultaneously maximizing the prescription **without attack** and minimizing the Jacobian with respect to input states; this provides us the possibility of achieving adversarial robustness without learning from adversaries.

Inspired by this derivation, we propose an *adversary agnostic PD (A2PD)* loss that includes two parts: 1) prescription gap maximization (PGM) loss, and 2) Jacobian regularization (JR) loss. The PGM loss is different from most previous distillation loss functions (e.g., cross-entropy) that merely maximize the probability of the action selected by teacher policy. In contrast, we also maximize the entropy of those actions not selected by the teacher policy, which enforces the student policy to have a larger prescription gap in order to resist perturbations in state observations. More importantly, the entropy term is weighted by the probability of the selected action; this allows the training to focus on PD accuracy at the beginning and pursue entropy maximization in the end. Last but not the least, in order to further improve the adversarial robustness, we also minimize the magnitude of Jacobian with respect to the input state, which is calculated based on PGM loss.

Our main contributions of this chapter can be summarized as:

- For the first time in DRL, we propose the adversary agnostic robust DRL that achieves robustness without relying on learning from adversaries. To this end,

¹The work in this chapter has been submitted to IEEE Transactions on Neural Networks & Learning Systems, 2021

based on the PD paradigm, we theoretically derive that the robustness of student policy can be indeed learned independently of the adversaries.

- To this end, we design the adversary agnostic PD (A2PD) loss function that contains: 1) a PGM loss for simultaneously maximizing the probability of the action prescribed by teacher policy as well as the entropy of unwanted actions; 2) a JR loss that minimizes the norm of Jacobian with respect to the input state.
- The theoretical analysis proves that our proposed PD loss guarantees to increase the prescription gap and the adversarial robustness. Meanwhile, experiments on five Atari games show that the robustness of the student policies trained by A2PD loss is significantly improved.

In this chapter, we start with providing preliminaries on deep reinforcement learning and policy distillation (PD). Based on PD paradigm, we theoretically derive how to achieve a robust student policy in resisting against adversaries but without relying on adversaries. Inspired by this derivation, we thereby propose a novel policy distillation loss $\mathcal{L}_{A2PD}$, consisting of a prescription gap maximization loss and a Jacobian regularization loss. Finally, we theoretically prove that our distillation loss can increase the prescription gap and the adversarial robustness.

5.1 Preliminaries

We consider a finite-horizon Markov decision process (MDP) that consists of a 4-tuple $(\mathcal{S}, \mathcal{A}, r, p)$, where $\mathcal{S}$ denotes the state space; $\mathcal{A}$ means the action space with size $|\mathcal{A}|$; $r(s_t)$ is the reward function when state s_t transits to s_{t+1} given action a_t ; and p represents the state transition function, e.g., $p(s_{t+1}|s_t, a_t)$, that is controlled by the environment. The aim of RL² algorithm (e.g., DQN [56]) is to maximize the expected accumulative

²In this dissertation, DRL and RL are interchangeably used.

Figure 5.1: Policy distillation in reinforcement learning

reward $\mathcal{R}(\pi_\theta) = \mathbb{E}[\sum_{t=0}^T \gamma^t r(s_t) | \pi_\theta]$ following a policy π_θ , where π is parameterized by θ ; γ is the discount factor. However, $\mathcal{R}(\pi_\theta)$ can be significantly degraded when an adversarial example $\delta_t : \mathcal{S} \rightarrow \mathcal{S}$ exists in state s_t . Note that in this study $\pi_\theta(s_t)$ represents a prescribed distribution in action space of the policy π_θ on state s_t ; $\pi_\theta(s_t, a)$ is the prescription on action a given policy π_θ and state s_t . In the adversarial attack setting, δ_t is added on the original state s_t in order to perturb the prescribed action distribution $\pi_\theta(s_t + \delta_t)$. Therefore, the perturbed action $a_t = \arg \max_a \pi_\theta(s_t + \delta_t, a)$ may be sub-optimal, thus reducing the reward of π_θ . The expected accumulative reward with perturbation δ_t is denoted as $\mathcal{R}(\pi_\theta) = \mathbb{E}[\sum_{t=0}^T \gamma^t r(s_t, \delta_t) | \pi_\theta]$. To improving the robustness in resisting against δ_t , adversarial training based policy distillations (PD) [25] have been used.

We follow the problem setting of PD [74] that is illustrated as Fig. 5.1. In PD, a teacher policy π_{θ^T} (e.g., Q value approximator) is first learned by RL algorithms. The aim of PD is to learn a student policy π_{θ^S} that can mimic the behavior of its teacher policy π_{θ^T} . Therefore, PD is formulated to minimise the loss function $\mathcal{L}(\theta^S)$ that measures the difference between the prescription

from student policy $\pi_{\theta^S}(s_t)$ and that from the pre-trained teacher policy $\pi_{\theta^T}(s_t)$, which is shown as

$$\mathcal{L}(\theta^S) = \mathbb{E}_{s_t \sim \mathcal{S}} [\mathcal{D}(\pi_{\theta^S}(s_t), \pi_{\theta^T}(s_t))], \quad (\text{Eq. 5.1})$$

where $\mathcal{D}$ is a distance measurement and it usually adopts the Kullback–Leibler (KL) divergence [15] or cross entropy [25].

Although PD has documented many success stories on reward improvement [74] and policy network compression [15], the adversarial robustness of the student policy π_{θ^S} has been less investigated so far. Recent advancements (e.g., Fischer et al. [25]) have studied the adversarial defense by involving adversarial training in PD. In doing so, the loss function for a robust student policy $\mathcal{L}_R(\theta^S)$ is reformulated as,

$$\mathcal{L}_R(\theta^S) = \mathbb{E}_{s_t \sim \mathcal{S}} \left[\max_{\delta_t} \mathcal{D}(\pi_{\theta^S}(s_t + \delta_t), \pi_{\theta^T}(s_t)) \right], \quad \|\delta_t\| \leq \epsilon, \quad (\text{Eq. 5.2})$$

where the norm value of δ_t is bounded by ϵ . In order to generate adversarial examples δ_t , many attack models (e.g., FGSM [53] and PGD [25]) have been applied. In particular, Zhang et al. [99] proposed the SA-MDP with a robustness regularization to align the policy’s prescriptions with and without adversarial perturbation; thus the regularization is still dependent on the adversaries δ_t . In summary, these approaches for robustness require accessing and operating on the adversaries; this makes adversarial robustness improvement less flexible. That is to say, the robustness obtained depending on one particular attack may fail when the agent faces another attack. Furthermore, the generation of adversarial examples may also lead to higher computational cost. Hence, a natural question to ask is: can we build a distillation paradigm that is capable of improving robustness without learning from adversaries in the training of student policy?

5.2 Adversarial Robustness without Adversaries in DRL

The aim of our policy distillation is to find a student policy π_{θ^S} that can maximize the accumulated reward $\mathcal{R}(\pi_{\theta^S})$ even with adversarial perturbation δ_t on state s_t , while the

distillation training is independent on adversaries. With this in mind, we note that the expected reward starting from s_t is denoted by state value $V(s_t)$:

$$V(s_t) = \mathbb{E} \left[\sum_{k=1}^T \gamma^k r(s_{t+k}) | \pi_{\theta^s} \right], \quad (\text{Eq. 5.3})$$

where γ is the discount factor. Thus, given adversarial perturbation δ_t on state s_t , the adversarial state value $V(s_t + \delta_t)$ is formulated by,

$$V(s_t + \delta_t) = \mathbb{E} \left[\sum_{k=1}^T \gamma^k r(s_{t+k} + \delta_t) | \pi_{\theta^s} \right]. \quad (\text{Eq. 5.4})$$

Note that δ_t is only applied to s_t , reflecting the impact for state value after adding δ_t on state s_t .

Notably, Theorem 2 (i.e., Theorem 1 in Constrained Policy Optimization [1]) below connects the difference in returns (or constraint returns) between two arbitrary policies to an average divergence between them. More details can refer to [1].

Theorem 2. [Theorem 1 in Constrained Policy Optimization [1]]

For any function $f : S \rightarrow \mathbf{R}$ and any policies π' and π , define $\delta_f(s, a, s') \doteq \mathcal{R}(s, a, s') + \gamma f(s') - f(s)$,

$$\begin{aligned} \epsilon_f^{\pi'} &\doteq \max_s |\mathbf{E}_{a \sim \pi', s' \sim \mathcal{P}}[\delta_f(s, a, s')]|, \\ L_{\pi, f}(\pi') &\doteq \mathbf{E}_{s \sim d^\pi, a \sim \pi, s' \sim P} \left[\left(\frac{\pi'(a|s)}{\pi(a|s)} - 1 \right) \delta_f(s, a, s') \right] \\ D_{\pi, f}^\pm(\pi') &\doteq \frac{L_{\pi, f}(\pi')}{1 - \gamma} \pm \frac{2\gamma\epsilon_f^{\pi'}}{(1 - \gamma)^2} \mathbf{E}_{s \sim d^\pi} [D_{TV}(\pi' || \pi)[s]] \end{aligned}$$

where $D_{TV}(\pi' || \pi)[s] = (1/2) \sum a |\pi'(a|s) - \pi(a|s)|$ is the total variational divergence between action distribution at s . To following bounds hold:

$$D_{\pi, f}^+(\pi') \geq J(\pi') - J\pi \geq D_{\pi, f}^-(\pi')$$

Furthermore, the bounds are tight (when $\pi' = \pi$, all three expressions are identically zero).

Later on, Zhang et al. [99] extend this theorem to the RL setting as shown by the Theorem 3 below.

Theorem 3. [Theorem 5 in [99]] Given a policy π for a non-adversarial MDP and its value function is $V_\pi(s)$. Under the optimal adversary v^* in state-adversarial MDP, for all $s \in \mathcal{S}$ we have

$$\max_{s \in \mathcal{S}} \{V_\pi(s) - V_{\pi_{ov^*}(\pi)}(s)\} \leq \alpha \max_{s \in \mathcal{S}} \max_{\hat{s} \in B(s)} D_{TV}(\pi(\cdot|s), \pi(\cdot|\hat{s}))$$

where $D_{TV}(\pi(\cdot|s), \pi(\cdot|\hat{s}))$ is the total variation distance between $\pi(\cdot|s)$ and $\pi(\cdot|\hat{s})$, and $\alpha := 2 \left[1 + \frac{\gamma}{(1-\gamma)^2} \max_{s_t, a_t, s_{t+1} \in \mathcal{S} \times \mathcal{A} \times \mathcal{S}} [r(s_t, a_t)] \right]$ is a constant independent on π_{θ^S} ; s_{t+1} indicates the next state at step $t + 1$.

Accordingly, under our definition, the difference between $V(s_t)$ and $V(s_t + \delta_t)$ can be bounded as,

$$\max_{s_t \in \mathcal{S}} \{V(s_t) - V(s_t + \delta_t)\} \leq \alpha \max_{s_t \in \mathcal{S}} \max_{\|\delta_t\| \leq \epsilon} \mathcal{D}(\pi_{\theta^S}(s_t), \pi_{\theta^S}(s_t + \delta_t)), \quad (\text{Eq. 5.5})$$

where $\mathcal{D}(\pi_{\theta^S}(s_t), \pi_{\theta^S}(s_t + \delta_t))$ is the distance in action space between the student policy prescription $\pi_{\theta^S}(s_t)$ without adversary and the prescription $\pi_{\theta^S}(s_t + \delta_t)$ with adversary. Eq. 5.5 indicates that the *state value gap between $V(s_t)$ and $V(s_t + \delta_t)$ has an upper bound relying on the distance $\mathcal{D}(\pi_{\theta^S}(s_t), \pi_{\theta^S}(s_t + \delta_t))$ in action space*. This motivates us to improve the adversarial robustness of the student policy by directly minimizing $\mathcal{D}(\pi_{\theta^S}(s_t), \pi_{\theta^S}(s_t + \delta_t))$ during policy distillation.

Recalling the discrete action selection in RL, given a deterministic policy π_{θ^S} , the action is selected as $\arg \max_a \pi_{\theta^S}(s_t, a)$. The distance $\mathcal{D}(\pi_{\theta^S}(s_t), \pi_{\theta^S}(s_t + \delta_t))$ can thus be defined as,

$$\mathcal{D}(\pi_{\theta^S}(s_t), \pi_{\theta^S}(s_t + \delta_t)) = \begin{cases} 0, & \arg \max_a \pi_{\theta^S}(s_t, a) = \\ & \arg \max_a \pi_{\theta^S}(s_t + \delta_t, a) . \\ 1, & \text{otherwise} \end{cases} \quad (\text{Eq. 5.6})$$

Note that in policy distillation, the student policy is trained to be consistent with its teacher policy, viz., selecting the same action a^T from π_{θ^T} as

$$\arg \max_a \pi_{\theta^T}(s_t, a) = a^T = \arg \max_a \pi_{\theta^S}(s_t, a). \quad (\text{Eq. 5.7})$$

Therefore, to minimize the distance in Eq. 5.6, $a^T = \arg \max_a \pi_{\theta^S}(s_t + \delta_t, a)$ is to be ensured. In other words, we need to encourage the student policy π_{θ^S} to choose the action a^T selected by the teacher policy π_{θ^T} , even with the adversary δ_t existing on state s_t . Accordingly, the following proposition is put forth for a robust student policy based on a pre-trained teacher policy π_{θ^T} .

Proposition 1. (*Robust student policy*) We assume that π_{θ^S} and π_{θ^T} are deterministic policies. The optimal action chosen by the teacher policy π_{θ^T} is $a^T = \arg \max_a \pi_{\theta^T}(s_t, a)$. Given bounded adversarial perturbations $\delta_t, \|\delta_t\| \leq \epsilon$ on state s_t , we define the prescription gap of student policy π_{θ^S} as,

$$\begin{aligned} \mathcal{G}_{\theta^S}(s_t + \delta_t, a^T) &= \min_{\delta_t, \|\delta_t\| \leq \epsilon} [\pi_{\theta^S}(s_t + \delta_t, a^T) - \pi_{\theta^S}(s_t + \delta_t, a)] \\ &\quad, \forall a \in \mathcal{A} \cap a \neq a^T. \end{aligned} \quad (\text{Eq. 5.8})$$

Then, a robust distilled student policy π_{θ^S} **must guarantee** $\mathcal{G}_{\theta^S}(s_t + \delta_t, a^T) > 0$.

Proposition 1 indicates that as long as $\mathcal{G}_{\theta^S}(s_t + \delta_t, a^T) > 0$, the action selected by the distilled student policy under any adversarial perturbation $\|\delta_t\| \leq \epsilon$ on state s_t is still a^T . Hence, during policy distillation, the value of $\pi_{\theta^S}(s_t + \delta_t, a^T) - \pi_{\theta^S}(s_t + \delta_t, a)$ needs to be maximized in order to boost adversarial robustness. According to Taylor expansion, we have

$$\begin{aligned} \pi_{\theta^S}(s_t + \delta_t, a^T) &= \pi_{\theta^S}(s_t, a^T) + \delta_t \nabla_{s_t} \pi_{\theta^S}(s_t, a^T) + \omega_1, \\ \pi_{\theta^S}(s_t + \delta_t, a) &= \pi_{\theta^S}(s_t, a) + \delta_t \nabla_{s_t} \pi_{\theta^S}(s_t, a) + \omega_2, \end{aligned} \quad (\text{Eq. 5.9})$$

where ω_1 and ω_2 are truncation errors. For ease of analysis we make a common assumption, viz., $\omega_1 - \omega_2 = 0$. Thereby, Eq. 5.8 can be transformed as,

$$\begin{aligned}
 \mathcal{G}_{\theta^S}(s_t + \delta_t, a^T) &= \min_{\delta_t, \|\delta_t\| \leq \epsilon} [(\pi_{\theta^S}(s_t, a^T) + \delta_t \nabla_{s_t} \pi_{\theta^S}(s_t, a^T)) - \\
 &\quad (\pi_{\theta^S}(s_t, a) + \delta_t \nabla_{s_t} \pi_{\theta^S}(s_t, a))] \\
 &= [\pi_{\theta^S}(s_t, a^T) - \pi_{\theta^S}(s_t, a)] + \\
 &\quad \min_{\delta_t, \|\delta_t\| \leq \epsilon} [\delta_t \nabla_{s_t} \pi_{\theta^S}(s_t, a^T) - \delta_t \nabla_{s_t} \pi_{\theta^S}(s_t, a)] \quad (\text{Eq. 5.10}) \\
 &= [\pi_{\theta^S}(s_t, a^T) - \pi_{\theta^S}(s_t, a)] + \\
 &\quad \min_{\delta_t, \|\delta_t\| \leq \epsilon} \delta_t \nabla_{s_t} [\pi_{\theta^S}(s_t, a^T) - \pi_{\theta^S}(s_t, a)] \\
 &= \mathcal{G}_{\theta^S}(s_t, a^T) + \min_{\delta_t, \|\delta_t\| \leq \epsilon} \delta_t \nabla_{s_t} \mathcal{G}_{\theta^S}(s_t, a^T).
 \end{aligned}$$

Thus, in order to maximize the prescription gap $\mathcal{G}_{\theta^S}(s_t + \delta_t, a^T)$, the first term $\mathcal{G}_{\theta^S}(s_t, a^T)$ in Eq. 5.10) should be maximized. Note that δ_t is optimized by an attacker to impact $\mathcal{G}_{\theta^S}(s_t, a^T)$ **negatively**; thus being unable to be controlled. Although δ_t can not be controlled by π_{θ^S} , we can alternatively control its impact weight (i.e., the Jacobian $\nabla_{s_t} \mathcal{G}_{\theta^S}(s_t, a^T)$). Namely, to improve the robustness, the influence of the second term in Eq. 5.10) can be reduced by minimizing the magnitude of the Jacobian $\|\nabla_{s_t} \mathcal{G}_{\theta^S}(s_t, a^T)\|$.

In sum, to maximize the prescription gap $\mathcal{G}_{\theta^S}(s_t + \delta_t, a^T)$ with attack, we can alternatively maximize $\mathcal{G}_{\theta^S}(s_t, a^T)$ **without** attack and simultaneously minimize the magnitude of Jacobian $\|\nabla_{s_t} \mathcal{G}_{\theta^S}(s_t, a^T)\|$. It is noteworthy that our robustness improvement derived in Eq. 5.10) is different from SA-MDP [99], although both works are motivated by a similar bound relationship as shown in Eq. 5.5. The goal of SA-MDP [99] is to align the prescriptions with adversary and without adversary; thus querying and generating the perturbation δ_t is still required. On the contrary, we can ignore the usage of any knowledge from δ_t by directly regularizing the prescription gap and the corresponding input gradient. Guided by this finding, we devise our adversary agnostic policy distillation (A2PD) loss in what follows.

5.3 Adversary Agnostic Policy Distillation (A2PD)

To enable the adversary agnostic robust DRL, our policy distillation loss $\mathcal{L}_{A2PD}(\theta^S)$ is proposed as,

$$\mathcal{L}_{A2PD}(\theta^S) = \mathcal{L}_{pgm}(\theta^S) + \beta \mathcal{L}_{jr}(\theta^S), \quad (\text{Eq. 5.11})$$

where $\mathcal{L}_{pgm}(\theta^S)$ is the prescription gap maximization loss that not only maximizes the likelihood on action a^T selected by teacher policy π_{θ^T} , but also maximizes the entropy on the remaining actions; $\mathcal{L}_{jr}(\theta^S)$ is the Jacobian regularization loss that aims to boot the robustness via minimizing the magnitude of input gradient on s_t back-propagated from $\mathcal{L}_{pgm}(\theta^S)$. Recalling Eq. 5.10), minimizing $\mathcal{L}_{pgm}(\theta^S)$ and $\mathcal{L}_{jr}(\theta^S)$ corresponds to maximizing $\mathcal{G}_{\theta^S}(s_t, a^T)$ and minimizing $\nabla_{s_t} \mathcal{G}_{\theta^S}(s_t, a^T)$, respectively. The weight β controls the strength of $\mathcal{L}_{jr}(\theta^S)$. We illustrate the details of $\mathcal{L}_{pgm}(\theta^S)$ and $\mathcal{L}_{jr}(\theta^S)$ as follows.

5.3.1 Prescription Gap Maximization (PGM)

With the goal of maximizing the prescription gap between action a^T and the remaining actions in mind, we devise the PGM loss $\mathcal{L}_{pgm}(\theta^S)$ as,

$$\mathcal{L}_{pgm}(\theta^S) = -\pi_{\theta^S}(s_t, a^T)^\eta \cdot \left[- \sum_{a=1, a \neq a^T}^{|\mathcal{A}|} \left(\frac{\pi_{\theta^S}(s_t, a)}{1 - \pi_{\theta^S}(s_t, a^T)} \right) \log \left(\frac{\pi_{\theta^S}(s_t, a)}{1 - \pi_{\theta^S}(s_t, a^T)} \right) \right], \quad (\text{Eq. 5.12})$$

where $\pi_{\theta^S}(s_t, a)$ is the prescription on action a ; $\eta \in (0, 1)$ is a constant. The rationale behind Eq. 5.12 is that minimizing $\mathcal{L}_{pgm}(\theta^S)$ enables to simultaneously maximize the likelihood $\pi_{\theta^S}(s_t, a^T)$ on the action a^T (i.e., the action selected by the teacher policy π_{θ^T}) and the entropy over the remaining actions $-\sum_{a=1, a \neq a^T}^{|\mathcal{A}|} \left(\frac{\pi_{\theta^S}(s_t, a)}{1 - \pi_{\theta^S}(s_t, a^T)} \right) \log \left(\frac{\pi_{\theta^S}(s_t, a)}{1 - \pi_{\theta^S}(s_t, a^T)} \right)$. The entropy maximization results in a smaller maximum over action a , $a \in \mathcal{A}, a \neq a^T$. Hence, by maximizing $\pi_{\theta^S}(s_t, a^T)$ at the same time, we can facilitate a larger prescription gap $\mathcal{G}_{\theta^S}(s_t, a^T)$. Note that, the entropy calculation is weighted by $\frac{1}{1 - \pi_{\theta^S}(s_t, a^T)}$; this makes the

distillation training focus on maximizing $\pi_{\theta^S}(s_t, a^T)$ at the beginning when $\pi_{\theta^S}(s_t, a^T)$ is small. In contrast, when $\pi_{\theta^S}(s_t, a^T)$ increases during training, $\mathcal{L}_{pgm}(\theta^S)$ gradually shifts attention to the entropy maximization. In addition, η balances the maximization on $\pi_{\theta^S}(s_t, a^T)$ and entropy regularization, which is analyzed in experimental studies.

5.3.2 Jacobian Regularization (JR)

As derived in Eq. 5.10), a robust policy distillation also requires minimizing the Jacobian on the input s_t as additional regularization. The concept of JR was introduced by Drucker and Le Cun [22] in double backpropagation to enhance generalization performance, where they trained neural networks not only by minimizing the gradient on weights but the gradient with respect to the input features. Hoffman et al. [36] and Chan et al. [12] utilized JR to regularize the stability and interpretability of image classifiers. However, how to effectively exploit JR in RL adversarial robustness, especially in the policy distillation process, has so far remained under-explored. With that in mind, we thus propose the JR loss,

$$\mathcal{L}_{jr}(\theta^S) = \left\| \frac{\partial \mathcal{L}_{pgm}(\theta^S)}{\partial s_t} \right\|_F, \quad (\text{Eq. 5.13})$$

where $\frac{\partial \mathcal{L}_{pgm}(\theta^S)}{\partial s_t}$ indicates the Jacobian on state s_t w.r.t. the loss function $\mathcal{L}_{pgm}(\theta^S)$; F represents the Frobenius norm. It is worth noting that most start-of-the-art attack algorithms (e.g., FGSM [31]) are on the basis of utilizing the Jacobian, thus minimizing the magnitude of Jacobian intuitively provides weaker gradient information; this makes a harder generation of δ_t for an attacker. In addition, according to the analysis in [24], if we maximize the prescription gap, it is able to alleviate the issue of gradient masking. A more detailed analysis on the improvement of adversarial robustness via minimizing the magnitude of Jacobian is provided in Theorem 5.

5.4 Theoretical Analysis

To support the our loss design for robust policy distillation, we analyze the policy prescription gap and the resultant improvement on adversarial robustness in the following theoretical analysis.

Theorem 4. (*Policy prescription gap maximization*) Given a particular prescription $\pi_{\theta^S}(s_t, a^T)$ by student policy π_{θ^S} on the action a^T , if the PGM loss $\mathcal{L}_{pgm}(\theta^S)$ is minimized, it is guaranteed that the prescription gap $\mathcal{G}_{\theta^S}(s_t, a^T)$ in Eq. 5.10 is maximized. Moreover, if $\pi_{\theta^S}(s_t, a^T) > \frac{1}{|\mathcal{A}|}$ where $|\mathcal{A}|$ is the size of action space, $\mathcal{G}_{\theta^S}(s_t, a^T)$ is ensured to be positive.

The proof follows from the fact that maximum entropy is attained when the distribution over actions is uniform, which is shown as below.

Proof. Given the condition that $\pi_{\theta^S}(s_t, a^T)$ is a particular prescription, we define $\pi_{\theta^S}(s_t, a^T) = C$ as a constant. Then the PGM loss in Eq. 5.12 can be rewritten as,

$$\mathcal{L}_{pgm}(\theta^S) = -C^\eta \cdot \left[- \sum_{a=1, a \neq a^T}^{|\mathcal{A}|} \left(\frac{\pi_{\theta^S}(s_t, a)}{1-C} \right) \log \left(\frac{\pi_{\theta^S}(s_t, a)}{1-C} \right) \right], \quad (\text{Eq. 5.14})$$

As $C \in (0, 1)$, we can get $1-C \in (0, 1)$ and $C^\eta \in (0, 1)$ where η is a positive constant.

Therefore, minimizing $\mathcal{L}_{pgm}(\theta^S)$ is equal to maximizing the entropy $h(\pi_{\theta^S}(s_t, a))$,

$$h(\pi_{\theta^S}(s_t, a)) = - \sum_{a=1, a \neq a^T}^{|\mathcal{A}|} \pi_{\theta^S}(s_t, a) \log(\pi_{\theta^S}(s_t, a)). \quad (\text{Eq. 5.15})$$

Given $\pi_{\theta^S}(s_t, a^T) = C$, we have $\sum_{a=1, a \neq a^T} \pi_{\theta^S}(s_t, a) = 1-C$. According to the information theorem [32], the maximum of $h(\pi_{\theta^S}(s_t, a))$ is obtained when the distribution of $\pi_{\theta^S}(s_t, a)$ is uniform; this results in a minimum $\pi_{\theta^S}(s_t, a) = \frac{1-C}{|\mathcal{A}|-1}$. Thereby, we can get the maximized prescription gap

$$\begin{aligned} \mathcal{G}_{\theta^S}(s_t, a^T) &= C - \frac{1-C}{|\mathcal{A}|-1} \\ &= \frac{|\mathcal{A}|}{|\mathcal{A}|-1} C - \frac{1}{|\mathcal{A}|-1}. \end{aligned} \quad (\text{Eq. 5.16})$$

If $\pi_{\theta^S}(s_t, a^T) = C > \frac{1}{|\mathcal{A}|}$, then we can get $\mathcal{G}_{\theta^S}(s_t, a^T) > 0$. ■

Theorem 5. (*Adversarial robustness*) Given a student policy π_{θ^S} , if both the PGM loss $\mathcal{L}_{pgm}(\theta^S)$ and JR loss $\mathcal{L}_{jr}(\theta^S)$ are minimized, we can guarantee an improvement on adversarial robustness.

The basic idea for proof is that the minimized $\left\| \frac{\partial \mathcal{L}_{pgm}(\theta^S)}{\partial s_t} \right\|$ minimizes the impact of adversary δ_t on the PGM loss. The detail of proof is shown as below.

Proof. In our policy distillation, the PGM loss $\mathcal{L}_{pgm}(\theta^S)$ is optimized to simultaneously maximize the probability on action a^T and the prescription gap $\mathcal{G}_{\theta^S}(s_t, a^T)$ as shown in the proof of Theorem 4. Therefore, in order to keep the policy π_{θ^S} robust to adversary δ_t , $\|\delta_t\| \leq \epsilon$, we have to ensure the PGM loss is still minimized with adversary δ_t . For ease of analysis, we rewrite the PGM loss $\mathcal{L}_{pgm}(\theta^S)$ as a function of s_t ,

$$\mathcal{L}_{pgm}(s_t) = -\pi_{\theta^S}(s_t, a^T)^\eta \cdot \left[- \sum_{a=1, a \neq a^T}^{|\mathcal{A}|} \left(\frac{\pi_{\theta^S}(s_t, a)}{1 - \pi_{\theta^S}(s_t, a^T)} \right) \log \left(\frac{\pi_{\theta^S}(s_t, a)}{1 - \pi_{\theta^S}(s_t, a^T)} \right) \right]. \quad (\text{Eq. 5.17})$$

The PGM loss under adversary δ_t is $\mathcal{L}_{pgm}(s_t + \delta_t)$. Using Taylor expansion, we have

$$\mathcal{L}_{pgm}(s_t + \delta_t) = \mathcal{L}_{pgm}(s_t) + \delta_t \frac{\partial \mathcal{L}_{pgm}(s_t)}{\partial s_t} + \omega_{pgm}, \quad (\text{Eq. 5.18})$$

$$\forall \delta_t, \|\delta_t\| \leq \epsilon.$$

ω_{pgm} is the truncation error in Taylor expansion, thus it can be assumed small enough to be ignored. Note that δ_t is generated by the attacker to negatively impact our distilled policy, which makes δ_t not controllable from the distillation training perspective. Therefore, to minimize the impact of $\delta_t \frac{\partial \mathcal{L}_{pgm}(s_t)}{\partial s_t}$, we can alternatively minimize the magnitude of Jacobian $\left\| \frac{\partial \mathcal{L}_{pgm}(s_t)}{\partial s_t} \right\|$.

Hence, given that the $\mathcal{L}_{pgm}(s_t)$ **without** adversary is minimized, if we minimize the norm of Jacobian $\left\| \frac{\partial \mathcal{L}_{pgm}(s_t)}{\partial s_t} \right\|$, we can still achieve a minimized PGM loss $\mathcal{L}_{pgm}(s_t + \delta_t)$

with adversary δ_t . Therefore, even with adversarial attack δ_t , the loss $\mathcal{L}_{pgm}(s_t + \delta_t)$ can still be minimized, which leads to a maximized $\mathcal{G}_{\theta^S}(s_t, a^T) > 0$ as proved in Theorem 4. Accordingly, we can prove that the adversarial robustness is improved. ■

It is noteworthy that the mathematical deviation in Eq. 5.10 is to show **how** can we achieve adversarial robustness without considering the information of attack δ_t . However, it is hard to be optimized directly. Therefore, we design the loss in Eq. 5.12 and Eq. 5.13 to achieve the goal direvied in Eq. 5.11. In addition, the PGM loss that we defined in Eq. 5.17 is just one particular form for maximizing $\mathcal{G}_{\theta^S}(s_t, a^T)$ through simultaneously maximizing the likelihood $\pi_{\theta^S}(s_t, a^T)$ on the action a^T (i.e., the action selected by the teacher policy π_{θ^T}) and the entropy over the remaining actions $-\sum_{a=1, a \neq a^T}^{|\mathcal{A}|} (\frac{\pi_{\theta^S}(s_t, a)}{1 - \pi_{\theta^S}(s_t, a^T)}) \log(\frac{\pi_{\theta^S}(s_t, a)}{1 - \pi_{\theta^S}(s_t, a^T)})$. Our contribution of the loss design here is to showcase an adversarial agnostic routine of achieving adversarial robustness in DRL settings.

5.5 Experiments

5.5.1 Experimental Setup

We test our approach³ on five Atari games (i.e., Freeway, Bank Heist, Pong, Boxing and Road Runner) that are utilized in the state-of-the-arts [25, 99]. The visualization of each game screen is shown in Fig. 5.2. For each game, the state is a 4-stack of consecutive frames, where each frame is pre-processed to size 84×84 . The pre-processing applies the environment wrapper (based on Arcade learning environment) in Rainbow. After preprocessing, the pixel value from $[0, 255]$ is normalized to $[0, 1]$.

Teacher Policy Training. In our evaluation, the teacher policy is trained by RAINBOW-DQN [35] that is deemed to be the start-of-the-art with combining many ingredients, such

³Our code will be released upon acceptance.

Table 5.1: Hyper-parameters settings for training teacher policy π_{θ^T} .

Parameters	Settings	Descriptions
Optimizer	Adam	–
Batch size	32	–
Learning rate	0.000625	Adam learning rate
ϕ_1	0.9	Adam decay rate 1
ϕ_2	0.999	Adam decay rate 2
Adam-eps	1.5×10^{-4}	Adam epsilon
Start steps	2×10^4	Number of steps before starting training
Environment ID	123	The random seed in Arcade environment
T-max	1×10^7	Number of training steps
Max-episode-len	108×10^3	Maximum episode length in game frames
h	4	Number of consecutive states processed
Hidden-size	512	Network hidden size
σ	0.1	Initial standard deviation of noisy linear layers
Atoms	51	Discretised size of value distribution
V-min	-10	Minimum of value distribution support
V-max	10	Maximum of value distribution support
Memory-length	1×10^6	The length of replay buffer
Target-update	1×10^4	The frequency of updating target network

as distributional RL [6], dueling networks [89] and noisy nets [26]. Each teacher policy is trained with 4 million frames on a particular game (see source code⁴), which costs 12-40 hours on Nvidia 2080Ti. The other parameter settings for teacher policy training are provided in Table 5.1. More detailed explanations of each parameter can refer to [35].

Student Policy Distillation Training. In our implementation, the network structure of student policy π_{θ^S} uses the Nature-DQN structure [56]. To train the student policy, we collect 1×10^5 state-prescription pairs $[s_t, \pi_{\theta^T}(s_t)]$ from the teacher policy θ^T , where 90% of the collected data is treated as training and the remaining 10% as testing. We use Adam

⁴<https://github.com/Kaixhin/Rainbow>

(5.2.a) BankHeist (5.2.b) Boxing (5.2.c) Freeway (5.2.d) Pong (5.2.e) RoadRunner

Figure 5.2: The visualization of the initial state $\mathbf{s}_t$ of the five selected Atari games.

as the optimizer, and the implementation is based on Keras. The rest hyperparameter settings for policy distillation are provided in Table 5.2.

Table 5.2: Hyper-parameters settings for our proposed policy distillation of π_{θ^S} .

Parameters	Settings	Descriptions
Batch size	32	—
Learning rate	0.00004	Adam learning rate
ϕ_1	0.9	Adam decay rate 1
ϕ_2	0.999	Adam decay rate 2
Adam-eps	1×10^{-7}	Adam epsilon
Max-epoch	1000	The maximum number of epochs
Patience	60	Early stop number
β	0.1	The weight of $\mathcal{L}_{jr}(\theta^S)$ in Eq. 5.11
η	1/3	Discount factor in Eq. 5.12

Evaluation of Robustness. To align with the compared state-of-the-arts [25, 99, 4], we use the untargeted Projected Gradient Decent (PGD) attack. However, to evaluate our achieved robustness more generally, we further tested our distilled policies using two additional attacks, including Jacobian saliency map attack (JSMA) [65] and fast gradient sign method (FGSM) [31]. The PGD attack performs K -iteration updates of adversary

δ_t , given by:

$$\begin{aligned} s^{k+1} &= s^k + \frac{\epsilon}{K} \mathcal{P} \left(\frac{\partial \mathcal{H}(\pi_{\theta s}(s^k), a^T)}{\partial s^k} \right), \\ s^0 &= s_t, \quad k = 0, 1, \dots, K-1, \end{aligned} \tag{Eq. 5.19}$$

where s^{k+1} is the attacked input state with adversarial perturbation inside; $\mathcal{H}(\pi_{\theta s}(s^k), a^T)$ is the cross-entropy loss between student policy prescription $\pi_{\theta s}(s^k)$ and the one-hot vector encoded based on action a^T selected by teacher policy. $\mathcal{P}$ is an operator for projecting the input gradient into a constrained norm ball. ϵ and $\frac{\epsilon}{K}$ are the total norm constraint and the norm constraint for each iteration step, respectively. We explore three different values of $K \in \{4, 10, 50\}$, and two different values of $\epsilon \in \{1/255, 3/255\}$. Note that $\epsilon = 3/255$ is a stronger attack that has not been evaluated in previous studies [25, 99]. The implementation of all the evaluated attacks is based on the adversarial robustness toolbox [61].

Evaluation of Relative Robustness. In previous evaluations of adversarial robustness of DRL, the accumulated rewards under attacks (i.e., usually PGD attacks) are treated as the evaluation metric. However, in evaluating the reward under attack, the performance of baseline model has a significant impact; this inspires us to further design an auxiliary evaluation metric named as relative robustness $\mathcal{M}(\mathcal{R}_\delta, \mathcal{R}_{w/o \delta})$,

$$\mathcal{M}(\mathcal{R}_\delta, \mathcal{R}_{w/o \delta}) = \frac{\mathcal{R}_\delta}{\mathcal{R}_{w/o \delta}} \cdot 100\% \tag{Eq. 5.20}$$

where $\mathcal{R}_\delta$ and $\mathcal{R}_{w/o \delta}$ are the reward with and without attack, respectively. Hence, $\mathcal{M}(\mathcal{R}_\delta, \mathcal{R}_{w/o \delta})$ provides a percentage variation of accumulated rewards with respect to the baseline performance $\mathcal{R}_{w/o \delta}$, which should be considered when comparisons are based on different baseline policies. As another contribution of this study, we contend that $\mathcal{M}(\mathcal{R}_\delta, \mathcal{R}_{w/o \delta})$ should be treated as a complement metric for evaluating adversarial robustness.

Table 5.3: The comparison of averaged accumulated reward. In evaluating A2PD, the averaged accumulated rewards is over 50 episodes. Note that, NA in table represents the evaluation is not reported in previous studies. Natural reward indicates the reward obtained without any attack, which corresponds to the $\mathcal{R}_B$ in Eq. 5.20. The results of vanilla DQN and DQN Adv. Training [4] are imported from [99].

Approaches	Attack Setting	Bank Heist	Boxing	Freeway	Pong	Road Runner
DQN (Vanilla)	Natural Reward	1308.40 $\pm$ 24.1	70.9 $\pm$ 34.1	34.0 $\pm$ 0.2	21.0 $\pm$ 0.0	45534.0 $\pm$ 7066.0
	PGD Attack Reward ($K = 10, \epsilon = \frac{1}{255}$)	564.0 $\pm$ 21.2	4.8 $\pm$ 5.5	0.0 $\pm$ 0.0	-21.0 $\pm$ 0.0	0.0 $\pm$ 0.0
DQN Adv. Training [4] (attack 50% frames)	Natural Reward	1126.0 $\pm$ 70.9	NA	25.4 $\pm$ 0.8	10.1 $\pm$ 6.6	22944.0 $\pm$ 6532.5
	PGD Attack Reward ($K = 10, \epsilon = \frac{1}{255}$)	9.4 $\pm$ 13.6	NA	0.0 $\pm$ 0.0	-21.0 $\pm$ 0.0	14.0 $\pm$ 34.7
Imitation Learning [25] (PGD Adv. Training)	Natural Reward	238.66	80.67	32.93	19.73	12106.67
	PGD Attack Reward ($K = 4, \epsilon = \frac{1}{255}$)	18.13	50.87	32.53	19.67	5753.33
SA-DQN [99] (PGD)	Natural Reward	1245.2 $\pm$ 14.5	NA	33.9 $\pm$ 0.4	21.0 $\pm$ 0.0	34032.0 $\pm$ 3845.0
	PGD Attack Reward ($K = 10, \epsilon = \frac{1}{255}$)	1006.0 $\pm$ 226.4	NA	23.7 $\pm$ 2.3	21.0 $\pm$ 0.0	20402.0 $\pm$ 7551.1
SA-DQN [99] (Convex)	Natural Reward	1235.4 $\pm$ 9.8	NA	30.0 $\pm$ 0.4	21.0 $\pm$ 0.0	44638.0 $\pm$ 7367.0
	PGD Attack Reward ($K = 10, \epsilon = \frac{1}{255}$)	1232.4 $\pm$ 16.2	NA	30.0 $\pm$ 0.0	21.0 $\pm$ 0.0	44732.0 $\pm$ 8059.5
	PGD Attack Reward ($K = 50, \epsilon = \frac{1}{255}$)	1234.6 $\pm$ 16.6	NA	30.0 $\pm$ 0.0	21.0 $\pm$ 0.0	44678.0 $\pm$ 6954.0
A2PD (<i>Adversary Agnostic</i>)	Natural Reward	1617.4 $\pm$ 34.8	74.2 $\pm$ 21.2	33.9 $\pm$ 0.3	20.7 $\pm$ 0.1	29252.0 $\pm$ 6610.7
	PGD Attack Reward ($K = 10, \epsilon = \frac{1}{255}$)	1620.8 $\pm$ 22.8	81.9 $\pm$ 16.9	34.0 $\pm$ 0.0	21.0 $\pm$ 0.0	32076.0 $\pm$ 7910.7
	PGD Attack Reward ($K = 50, \epsilon = \frac{1}{255}$)	1620.0 $\pm$ 40.6	85.2 $\pm$ 3.7	33.9 $\pm$ 0.4	21.0 $\pm$ 0.0	31078.0 $\pm$ 6848.3
	PGD Attack Reward ($K = 10, \epsilon = \frac{3}{255}$)	1622.0 $\pm$ 35.0	79.0 $\pm$ 16.2	33.5 $\pm$ 0.9	18.2 $\pm$ 3.7	32396.0 $\pm$ 5623.0
	PGD Attack Reward ($K = 50, \epsilon = \frac{3}{255}$)	1606.2 $\pm$ 53.9	77.6 $\pm$ 11.0	32.8 $\pm$ 2.1	17.1 $\pm$ 4.0	30622.0 $\pm$ 7275.7
	FGSM Attack Reward	1624.8 $\pm$ 30.8	78.4 $\pm$ 22.85	33.8 $\pm$ 0.3	20.3 $\pm$ 0.3	31324.0 $\pm$ 5096.3
	JSMA Attack Reward	1615.4 $\pm$ 43.6	74.5 $\pm$ 20.9	33.9 $\pm$ 0.2	18.7 $\pm$ 2.3	30084.0 $\pm$ 6886.3

5.5.2 Robustness Evaluation of Policy π_{θ^s}

We evaluate the robustness of the policies trained by our adversary agnostic PD (A2PD), and compare our results with the state-of-the-art methods, including (1) vanilla DQN, (2) DQN adversarial training [4], (3) imitation learning with adversarial training [25], (4) SA-DQN with PGD [99], and (5) SA-DQN with convex relaxation [99]. To provide a comprehensive comparison on the adversarial robustness, we evaluate our distilled policies based on a wide range of attacks, including PGD attacks (with $K \in \{10, 50\}, \epsilon \in \{1/255, 3/255\}$), fast gradient sign method (FGSM) and Jacobian saliency map attack (JSMA). The results are presented in Table 5.3, where the performance under same attack is shaded with same color. For instance, the performance under PGD attack with $K = 10, \epsilon = 1/255$ and PGD attack with $K = 50, \epsilon = 1/255$ are shaded by light blue and light brown, respectively. Note that the top 1-3 rewards under attacks are highlighted by bold blue, blue and bold, respectively.

In general, the accumulated rewards of our distilled policies π_{θ^s} under different attacks are larger than those from previous approaches; this indicates that our A2PD achieves significant robustness improvements although without adversary involved in the training. Specifically, several interesting findings are noted. **(1)** Under PGD ($K = 10, \epsilon = 1/255$) attack (i.e., shaded by light blue), our π_{θ^s} achieves a much higher reward than all the start-of-the-arts. Especially on Bank Heist, our A2PD achieves the average reward of 1632.4, whilst in previous studies the corresponding reward values are less than 1250. Another noteworthy finding is on Freeway, where our A2PD achieve the reward 34.0 ± 0.0 , the maximum that any policy can get even without attack. On Road Runner, the reward of our A2PD is smaller than that of SA-DQN (convex), which however is mainly caused by the fact that our natural reward 29252.0 ± 6610.7 is significantly smaller than 44638.0 ± 7367.0 in SA-DQN. Further comparison and illustration of this issue (i.e., based on the relative robustness) is provided in Table 5.4. **(2)** The similar superiority

Table 5.4: The comparison of relative robustness. The relative robustness value $\mathcal{M}(\mathcal{R}_\delta, \mathcal{R}_{w/o\ \delta})$ is calculated based on Eq. 5.20), where $\mathcal{R}_\delta$ is the reward under PGD ($K = 10, \epsilon = 1/255$) that is shaded with light blue in Table 5.3, $\mathcal{R}_{w/o\ \delta}$ is the corresponding natural reward without attack. The largest value of $\mathcal{M}(\mathcal{R}_\delta, \mathcal{R}_{w/o\ \delta})$ is highlighted with **bold blue** text.

Approach	BankHeist	Boxing	Freeway	Pong	Road Runner
DQN (Vanilla)	43.1%	6.8%	0%	-100%	0%
DQN (Adv Training)[4]	0.8%	NA	0%	-207.9%	0.06%
Imitation Learning [25] (PGD Adv Training)	7.6%	63.1%	98.8%	99.7%	47.5%
SA-DQN [99] (PGD)	80.8%	NA	69.9%	100%	59.9%
SA-DQN (Convex) [99]	99.8%	NA	100%	100%	100.2%
A2PD (Adversary Agnostic)	100.2%	110.4%	100.3%	100%	109.7%

Figure 5.3: The adversarial robustness comparison between teacher policy $\pi_{\theta T}$ and student policy $\pi_{\theta S}$.

of our A2PD can also be found in PGD ($K = 50, \epsilon = 1/255$) attack (i.e., shaded by light brown) versus SA-DQN (convex) [99], where our A2PD outperforms SA-DQN (convex) on 4/5 games. On Road Runner, the natural reward of SA-DQN is much larger; thus making sense that our A2PD performs slightly worse under attack. **(3)** In addition to $K \in \{10, 50\}$ with $\epsilon = 1/255$, we further evaluated our A2PD under 3 times stronger PGD attacks with $\epsilon = 3/255$. According to Table 5.3, the policies learned by our A2PD still behave robustly. For instance, the policy learned on Bank Heist achieves the reward of 1622.0 ± 35.0 , which is still slightly larger than the corresponding natural reward 1617.4 ± 34.8 . **(4)** Besides the PGD attack, we also evaluate the policies trained by our A2PD on FGSM and JSMA, which further showcases the ability of A2PD for achieving adversarial robustness. **(5) Relative robustness $\mathcal{M}(\mathcal{R}_\delta, \mathcal{R}_{w/o \delta})$:** In most of previous studies, only the reward under PGD ($K = 10, \epsilon = 1/255$) attack is evaluated, but the corresponding baselines (i.e., natural reward) are different. Therefore, we further compare the relative robustness defined in Eq. 5.20). The comparison shown in Table 5.4 indicates that on all the games, our A2PD achieves the highest relative robustness. More importantly, the rewards from our A2PD are greater than or equal to 100% on all the five games. In summary, all the aforementioned robustness of our approach is achieved without leveraging any knowledge of adversaries; this thereby provides a totally new routine for achieving robustness in RL.

5.5.3 Robustness Evaluation of Policy π_θ^S on Minimalistic Attacks

Given Chapter 3 and Chapter 4 have revealed that DRL policies can be easily fooled by minimalistic attack based perturbation, we further investigate the robustness of our A2PD trained π_θ^S on such scenario. In doing so, we evaluate our distilled policies based on a range of minimalistic attacks with $n \in \{1, 5, 10, 15\}$, where n is the number of perturbed pixels as shown in Chapter 3. The accumulated reward and comparisons with

Figure 5.4: The robustness evaluation on minimalistic attacks. The reward is evaluated for 50 episodes to calculate the mean (i.e., the height of the histogram) and standard deviation (i.e., the error bar on each histogram).

the corresponding teacher policies on the five games under different n -pixel attack are shown in Figure 5.4.

In general, the student policies trained by our A2PD achieve significantly higher rewards than the teacher policies with $n \geq 1$. In particular, on Bank Heist, Freeway and Road Runner, when $n \geq 5$, the average accumulated rewards of the teacher policy decrease to 0, while our A2PD trained student policy still can achieve the reward that is larger than 20. On Boxing, when $n \geq 5$, the reward of teacher policy decrease to a negative value, while the student policy from A2PD performs robustly with achieving positive reward values. On Pong, although when $n \in \{10, 15\}$ both teacher and student policies are completely deceived by the minimalistic attacks, the policies trained by A2PD still behave much more robust than the teacher policy when $n \in \{1, 5\}$.

These observations are consistent with our theoretical design. In particular, recalling the optimization objective Eq. 3.2 in Chapter 3, the aim of minimalistic attacks is *minimize* $\tilde{\mathcal{D}}$, viz.,

$$\begin{aligned} \tilde{\mathcal{D}} &= \max_{j \neq o} \pi(\cdot | s_t + \delta_t)_j - \pi(\cdot | s_t + \delta_t)_o \\ \text{where: } o &= \arg \max_j \pi(\cdot | s_t)_j. \end{aligned} \tag{Eq. 5.21}$$

However, from our prescription gap maximization loss design in Eq. 5.17, the student policies trained by A2PD *maximize* $\tilde{\mathcal{D}}$. Therefore, our A2PD naturally behaves more robust for minimalistic attacks.

5.5.4 Robustness Comparison between Teacher Policy π_{θ^T} and Student Policy π_{θ^S}

In this part, we compare the robustness between our own teacher policy π_{θ^T} and student policy π_{θ^S} . To this end, we evaluate both π_{θ^T} and π_{θ^S} under four PGD attacks (i.e., $K \in \{10, 50\}$ and $\epsilon \in \{1/255, 3/255\}$). The results are depicted in Fig. 5.3. As a whole, the rewards of teacher policies decrease significantly under attacks, whereas the rewards

of student policies keep remarkably robust under attacks. For instance, in Fig. 5.3 (c) on Freeway and Fig. 5.3 (e) on Road Runner, all the rewards of teacher policy under attacks are decreased to nearly 0 (i.e., the bar is hardly observable). Similarly, in Fig. 5.3 (d) on Pong, the rewards under attacks decrease to -21 that is the minimum reward value any policy could achieve. *In contrast*, on Freeway, Bank Heist and Pong, we observe that the student policies achieve approximately same rewards with their teacher policies and keep stable under attacks; this suggests that our distillation can achieve robustness without sacrificing performance on rewards. Although on Boxing and Road Runner the student policies cannot get rewards as high as their teacher policies, they still perform significantly more robust than the corresponding teacher policies. This firmly shows that our A2PD dramatically enhances the robustness even the data is collected from non-robust teacher policies.

5.5.5 Analysis on Hyper-parameter β

We analyze the impact of β in Eq. 5.11, which controls the weight of JR loss. In particular, we test six different values for β (i.e., $\beta \in \{0.001, 0.01, 0.1, 1, 10, 100\}$) on the five Atari games. The results are shown in Fig. 5.5. In general, we observe that on most cases the best performance is achieved with $\beta = 0.01$. Moreover, when β is too large, the rewards (with and without attacks) of distilled policies decrease accordingly. These empirical observations are consistent with our analysis in Theorem 5, where the adversarial robustness is based on maximizing the prescription gap even with adversaries in state observations. Therefore, a too large β will overemphasize the contribution of the JR loss, thus impacting the optimization towards PGM loss that controls the accuracy of imitating the teacher policy’s action selection a^T .

In particular, on Bank Heist, the rewards of the distilled policies keep stable when $\beta \leq 1$, while dramatically decreases to nearly 0 with $\beta \in \{10, 100\}$; thus the corresponding

Figure 5.5: The analysis on hyperparameter β for Bank Heist, Boxing and Freeway. With each β , the distilled policy is evaluated for 50 episodes to calculate the mean (i.e., the height of the histogram) and standard deviation (i.e., the error bar on each histogram).

Figure 5.6: The analysis on hyperparameter β for Pong and Road Runner. With each β , the distilled policy is evaluated for 50 episodes to calculate the mean (i.e., the height of the histogram) and standard deviation (i.e., the error bar on each histogram).

bars are invisible in Fig. 5.5(a). In Fig. 5.5(b), the highest reward under attack is achieved when $\beta = 0.01$ and decreases to negative values when $\beta = 100$. On Freeway, compared to other games, the robustness achieved is relatively less sensitive to the value of β , with a slight reward decrease when $\beta \geq 10$. The opposite situation occurs on Pong and Road Runner. On Pong, when $\beta = 100$, the natural reward and the rewards under attacks are -21. On Road Runner, the distilled policy with $\beta = 0.01$ keeps robust for all the attacks, while other values of β cannot achieve such consistent robustness; such comparison is most obvious on PGD ($K = 10, \epsilon = 3/255$) attack.

Moreover, we also evaluate the impact of β on the convergence process of A2PD training; this further unveils the relationship between our achieved performance and the varying value of β . The results are shown in Fig. 5.7, from which we find that in general the A2PD training convergence becomes slower with the increasing of β . This finding is also consistent with our theoretical analysis, viz., the β only controls the Jacobian minimization. Therefore, when β is too large, it hurts the optimization towards PGM loss (i.e., responsible for the accuracy of A2PD training). In particular, on most games (i.e., Bank Heist, Freeway, Pong and Road Runner), when $\beta \geq 10$, there are significant validation accuracy drops. Especially with $\beta = 100$, the validation accuracy on Road Runner is even less than 50%. Contrarily, the validation accuracy on most games gradually converges to the optimum if $\beta \leq 0.1$.

5.5.6 Analysis on Hyperparameter η in $\mathcal{L}_{pgm}(\theta^S)$

In Eq. 5.12), the hyperparameter η is designed to balance the maximization of $\pi_{\theta^S}(s_t, a^T)$ and the entropy regularization; thus it is expected to impact the convergence of A2PD training. Therefore, we further analyze the hyperparameter η with evaluating different values from $\{1/9, 1/6, 1/3, 1\}$. In the training, we utilize early stop with setting patience as 60. The maximum epoch value is 1000. The convergence plots across the five games

 Figure 5.7: The convergence of A2PD training with different β .

 Figure 5.8: The convergence of A2PD training with different η in (Eq. 5.12).

are shown in Fig. 5.8. From the results, we can generally see that the convergence becomes faster with the decreasing of η . In particular, the convergence processes with $\eta \leq 1/3$ keep similar, while the convergence with $\eta = 1$ got stuck. This suggests that the maximization of $\pi_{\theta^S}(s_t, a^T)$ and the entropy regularization in Eq. 5.12) can be well balanced with the setting $\eta \in [1/9, 1/3]$ on Atari games.

5.5.7 Analysis on Prescription Gap

To examine our claim on improving adversarial robustness via maximizing the prescription gap between optimal action and other sub-optimal actions, we compare the prescription gap between the teacher policy π_{θ^T} and student policy π_{θ^S} as shown in Fig. 5.9. Assuming a prescription distribution $P = [p_0, p_1, \dots, p_N]$, the prescription gap is calculated as $p_{i^*} - \max_{j \neq i^*} p_j$, where $i^* = \arg \max_i p_i$. From Fig. 5.9, on all the five games tested, the prescription gap values of student policies are far larger than those of teacher policies. Together with the above robustness analysis, it further demonstrates that the robustness of DRL policies is indeed improved via maximizing the prescription gap based on our policy distillation loss in Eq. 5.11. Most importantly, in achieving the adversarial robustness for DRL policies by maximizing the prescription gap, no knowledge of the adversaries is required, which thus enables a more realistic solution for improving adversarial robustness compared to adversarial training based approaches.

5.5.8 Analysis on Computational Efficiency

To support our claim about computational efficiency, we compare the averaged training time of our A2PD and PGD based adversarial training. Note that, the maximum epoch and patience for early stop are set same for all the student policy training, thus we can compare the averaged training time on each epoch. The statistics of training time is obtained on NVIDIA RTX 2080 Ti. The results are shown in Table 5.5, where the

Figure 5.9: Comparison of prescription gap between teacher policy π_{θ^T} and student policy π_{θ^S}

Table 5.5: Computation time comparison on five games. The value is the average over 50 epochs. The unit of time is second.

Time on Games	A2PD	Adversarial Training		
		PGD-4	PGD-10	PGD-20
<i>Time (Bank Heist)</i>	18.20s	23.29s	36.89s	64.76s
<i>Time (Boxing)</i>	17.23s	25.73s	40.12s	67.32s
<i>Time (Freeway)</i>	19.73s	26.72s	43.54s	68.57s
<i>Time (Pong)</i>	19.34s	26.52s	41.47s	68.36s
<i>Time (Road Runner)</i>	17.41s	27.83s	43.26s	71.51s
<i>Average</i>	18.38s	26.02s	41.09s	68.10s

average training time (seconds) from 50 epochs on the five games are presented. In general, we can see that our A2PD requires significantly less computation time than all the adversarial training approaches. Specifically, in average on all games, our A2PD only requires 27% of the training time in PGD-20 adversarial training.

5.6 Summary

For the first time, we propose an adversary agnostic robust deep reinforcement learning paradigm (i.e., based on policy distillation) that achieves robust policy to resist against adversaries but without relying on any information of adversaries during training. To this end, we theoretically derive that the robustness of student policy can be indeed learned independent with the adversaries. Accordingly, we propose a novel PD loss function that contains: 1) a PGM loss for simultaneously maximizing the probability of the action prescribed by teacher policy as well as the entropy of unwanted actions; 2) a JR loss that minimizes the norm of Jacobian with respect to the input state. The theoretical analysis proves that our propose PD loss guarantees to increase the prescription gap and the adversarial robustness. Meanwhile, experiments on five Atari games show that the

robustness of the student policies trained via our PD loss is significantly improved.

Chapter 6

Conclusions and Future Works

6.1 Conclusions

In this thesis, I investigate the adversarial robustness of deep reinforcement learning from the perspectives of both robustness evaluation and robustness improvement. Particularly, in Chapter 2, we provide the background and related works related to reinforcement learning and adversarial robustness.

In Chapter 3, we focus on the minimalistic attacks that mainly explores how little it takes to deceive an RL policy. To this end, we we define *three key settings*: (1) *black-box policy access*: where the attacker only has access to the input (state) and output (action probability) of an RL policy; (2) *fractional-state adversary (FSA)*: where only several pixels are perturbed, with the extreme case being a single-pixel adversary; and (3) *tactically-chanced attack (TCA)*: where only significant frames are tactically chosen to be attacked. To accommodate the three restrictive settings for adversarial attack generation in deep RL, we formulate the task as a black-box optimization problem with a novel objective function to generate the FSA under black-box settings, and propose an entropy-based uncertainty measurement to achieve the TCA. We formulate the adversarial attack by accommodating the three key settings, and explore their potency on six Atari games by examining four fully trained state-of-the-art policies. In Breakout, for example, we surprisingly find that: (i) all policies showcase significant performance degradation by

merely modifying 0.01% of the input state, and (ii) the policy trained by DQN is totally deceived by perturbing only 1% frames.

In Chapter 4, considering the computational complexity of the minimalistic attacks caused by treating every frame in isolation, we showcase the first study of how transferability across frames could be exploited for boosting the creation of minimal yet powerful attacks in image-based RL. In doing so, we introduce three types of frame-correlation transfers (i.e., anterior case transfer, random projection based transfer, and principal components based transfer) with varying degrees of computational complexity in generating adversaries via a genetic algorithm. We empirically demonstrate the trade-off between the complexity and potency of the transfer mechanism by exploring four fully trained state-of-the-art policies on six Atari games. In general, our frame-correlation transfers dramatically speed up the attack generation compared to existing methods, often reducing the computation time required to nearly zero; thus, shedding light on the real threat of real-time attacks in RL.

Given the adversarial robustness issue, in Chapter 5, we propose an attack agnostic defense approach in order to increase the robustness of existed DRL policies. Different from previous approaches assuming that the knowledge of adversaries can be added into the training process to achieve the corresponding generalization ability on these perturbed observations, we introduce an *adversary agnostic* robust DRL paradigm that does not require learning from adversaries. Note that the assumption of involving adversarial attacks as training data not only makes the robustness improvement more expensive, but may also leave a model less effective to other kinds of attacks in the wild. Therefore, our adversary agnostic robust DRL paradigm is a more realistic approach. In particular, we first theoretically derive that robustness could indeed be achieved independently of the adversaries based on a policy distillation setting. Motivated by this finding, we propose a new policy distillation loss with two terms: 1) a prescription gap maximization

loss aiming at simultaneously maximizing the likelihood of the action selected by the teacher policy and the entropy over the remaining actions; 2) a corresponding Jacobian regularization loss that minimizes the magnitude of gradient with respect to the input state. The theoretical analysis shows that our distillation loss guarantees to increase the prescription gap and the adversarial robustness. Furthermore, experiments on five Atari games firmly verify the superiority of our approach in terms of boosting adversarial robustness compared to other state-of-the-art methods.

6.2 Future Works

Future works will be mainly focused on four aspects, via.,

- more realistic adversarial with limited policy queries;
- adversarial defense approaches with theoretical guarantee;
- adversarial robustness certificates in DRL settings;
- more general framework on tasks with both discrete and continuous actions.

6.2.1 Limited Policy Query Based Adversarial Attack

Although our minimalistic attack in Chapter 2.2 already showcased that the DRL policies can be deceived in restricted settings, the computation budget is still an issue for real world deployments of such attacks. Namely, many time sensitive RL tasks may not allow the timeslot for generating minimalistic attacks. To alleviate such an issue, the following frame correlation transfer in Chapter 4 indicates that the computation can be significantly reduced with transferring the knowledge among different frame. This however is still based on the assumption that prior successful attacks exist, thus the knowledge can be leveraged. Therefore, one future direction towards revealing more

realistic vulnerability threat for DRL policies is to enable limited query based black-box attack. To this end, the black-box gradient estimation technique [40], such as natural evolution strategy, is a promising method worth to try.

6.2.2 Theoretical Guaranteed Adversarial Defense

Modern deep reinforcement learning (DRL) typically utilizes deep neural networks (DNNs) as function approximators which is trained in an end-to-end style [56]. Although DRL has shown remarkable performance on various complex tasks (e.g., board games [79], video games [56] and robotics control [46]), many studies on adversarial attacks [39, 47, 71, 94] indicate that DRL policies can be easily deceived by the perturbations (e.g., human imperceptible pixel noises and inaccurate sensor measurement) on the *observations of states*. Given the end-to-end inference of DRL, the perturbed observations of input states are directly fed into the DNN policies, hence the perturbations consequently propagate to the output (i.e., action prescriptions) influencing the reward.

We claim that such vulnerabilities to state observations are mainly caused by the DNN policies failing to extract *useful and robust* features. Such claim is inspired by a recent work from Ilyas et al. [41], where the vulnerability is explained as the result of non-robust features (i.e., highly predictive, yet brittle and incomprehensible to humans). On the other hand, the robust features are expected to be uncorrelated to the human incomprehensible noises. In a nutshell, both robust and non-robust features can be beneficial to the end-to-end reward maximization, but non-robust features can be easily modified by adversarial perturbations. Therefore, a following question to ask is:

how can we build a “robustified” policy that only operates on robust features given a well-trained policy?

To answer this question, we will use a robust information bottleneck for extracting the robust features under the policy distillation setting. To this end, first, we can utilize a

variational policy network structure that supports the calculation of Fisher information. Note that such variational network structure has empirically shown to give some sense of robustness. However, we showcase both theoretically and empirically that by introducing randomization can only give limited robustness. To further ensure the robustness, we can regularize the Fisher information in order to motivate the policy operating only on robust features.

6.2.3 Robustness Certificate of DRL

During the past years, the studies on adversarial attack and defense have continually being in an arm-race, which makes it difficult to deliver the machine learning models that can be robust on any possible attacks. Given that fact, to enable safe and trustworthy deployments of DRL policies, it is important to offer a robustness certificate that can provide a performance guarantee under possible adversarial attacks. Such robustness certificate should be able to provide the lower-bound performance of the DRL policies under adversarial attack. From this perspective, with a given DRL policy, we have to design an optimal adversary that leads to the worst performance, namely, an absolute lower bound of the expected cumulative reward an agent can receive.

In doing so, we can train another adversarial RL agent to generate the optimal adversarial attacks. However, different from vanilla RL training to maximize the expected accumulated reward, the goal of this adversarial agent is to minimize the expected accumulated reward instead. In that sense, if an RL algorithm can ensure the agent to achieve the maximum reward in the vanilla training, the adversarial agent can thus be trained to obtain the minimum reward; thereby ensuring getting the lower bound of the reward for DRL policies.

6.2.4 General Framework for Discrete and Continuous RL tasks

Our proposed approaches are mainly suitable for RL tasks with discrete action space. For instance, the frame importance factor proposed in minimalistic is based on an entropy calculated from discrete action probabilities. The prescription gap in Chapter 5 is also based on the similar discrete action space. In specific, our approaches for both attack and defense are evaluated on Atari games, that are benchmark environments for discrete tasks. For continuous tasks, our proposed attacks and defense need to be adjusted accordingly. However, we believe the motivation and philosophy behind the proposed ideas can be inspiring for future research. For instance, the tactically chanced attack in continuous tasks could be based on the magnitude of action prediction. Namely, for those frames with larger actions (e.g., forces in robotics), we can enable the attacker to add perturbations in input states. On the contrary, for the frames with smaller predicted forces, the perturbations are easy to be detected since the agent itself keeps quiet at the moment. Thus similar tactics of skipping attack from such frames could also increase an attacker’s imperceptibility on RL tasks with continuous action space.

List of Publications

- **Xinghua Qu**, Abhishek Gupta, Yew-Soon Ong, Zhu Sun. Adversary Agnostic Robust Deep Reinforcement Learning. IEEE Transactions on Neural Network & Learning Systems (TNNLS, Impact Factor: 10.45), 2021.
- **Xinghua Qu**, Ong Yew Soon, Abhishek Gupta. Frame Correlation Transfers Trigger Economical Attacks on Deep Reinforcement Learning Policies. IEEE Transactions on Cybernetics (TCYB, Impact Factor: 11.079), 2021.
- **Xinghua Qu**, Zhu Sun, Ong Yew Soon, Pengfei Wei, Abhishek Gupta. Minimalistic Attacks: How Little It Takes to Fool Deep Reinforcement Learning Policies. IEEE Transactions on Cognitive and Developmental Systems (TCDS), 2020.
A*STAR¹ Research Highlight Award².
- **Xinghua Qu**, Yew-Soon Ong, Yaqing Hou, Xiaobo Shen. Memetic Evolution Strategy for Reinforcement Learning. IEEE Congress on Evolutionary Computation (CEC), 2019.
- Hang Xu, **Xinghua Qu**, Zinovi Rabinovich. Spiking Pitch Black: Poisoning an Unknown Environment to Attack Unknown Reinforcement Learners. International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2022).
- Pengfei Wei, Yiping Ke, **Xinghua Qu**, Tze-Yun Leong. Subdomain Adaptation with Manifolds Discrepancy Alignment. IEEE Transactions on Cybernetics (TCYB), 2021.

¹Agency for Science, Technology and Research

²<https://research.a-star.edu.sg/articles/highlights/expecting-the-unexpected-from-ai/>

- Hongyang Liu, Zhu Sun, **Xinghua Qu**, Fuyong Yuan. Top-aware Recommender Distillation with Deep Reinforcement Learning. Information Science, 2021.
- Pengfei Wei, **Xinghua Qu**, Yiping Ke, Tze-Yun Leong, Yew Soon Ong. Adaptive Knowledge Transfer based on Transfer Neural Kernel Network. Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS), 2020.
- Zhu Sun, Di Yu, Hui Fang, Jie Yang, **Xinghua Qu**, Jie Zhang, Cong Geng. Are We Evaluating Rigorously? Benchmarking Recommendation for Reproducible Evaluation and Fair Comparison. ACM Conference on Recommender Systems (RecSys), 2020.

List of Submissions

- **Xinghua Qu**, Yew-Soon Ong, Abhishek Gupta, Pengfei Wei, Zhu Sun. Importance Adaptive Policy Distillation. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI).
- **Xinghua Qu**, Pengfei Wei, Mingyong Gao, Zhu Sun, Yew-Soon Ong, Zejun Ma. Synthesising Audio Adversarial Examples for Automatic Speech Recognition. The International Conference on Learning Representations (ICLR-2022).
- Pengfei Wei, Thanh Vinh Vo, **Xinghua Qu**, Yew-Soon Ong, Zejun Man. Transfer Kernel Learning for Multi-source Transfer Regression. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI).
- Pengfei Wei, **Xinghua Qu**, Yew-Soon Ong, Zejun Ma. An Improved Transfer Model: Randomized Transferable Machine. IEEE Transactions on Neural Network & Learning Systems (TNNLS).
- Zhendong Guo, Yew-Soon Ong, **Xinghua Qu**. Generative Self-Transfer Optimisation. IEEE Transactions on Cybernetics (TCYB).

References

- [1] Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *Proceedings of the 34th International Conference on Machine Learning (ICML) - Volume 70*, pages 22–31. JMLR. org, 2017.
- [2] Dimitris Achlioptas. Database-friendly random projections. In *Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 274–281, 2001.
- [3] Lu Bai, Yew-Soon Ong, Tiantian He, and Abhishek Gupta. Multi-task gradient descent for multi-task learning. *Memetic Computing*, pages 1–15, 2020.
- [4] Vahid Behzadan and Arslan Munir. Whatever does not kill deep reinforcement learning, makes it stronger. *arXiv preprint arXiv:1712.09344*, 2017.
- [5] Yonatan Belinkov and James Glass. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72, 2019.
- [6] Marc G Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. In *International Conference on Machine Learning (ICML)*, pages 449–458, 2017.
- [7] Dimitri P Bertsekas. Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334, 1997.
- [8] Ella Bingham and Heikki Mannila. Random projection in dimensionality reduction: applications to image and text data. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 245–250, 2001.

- [9] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.
- [10] Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017.
- [11] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57. IEEE, 2017.
- [12] Alvin Chan, Yi Tay, Yew Soon Ong, and Jie Fu. Jacobian adversarially regularized networks for robustness. In *International Conference on Learning Representations (ICLR)*, 2019.
- [13] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *10th ACM Workshop on Artificial Intelligence and Security*, pages 15–26. ACM, 2017.
- [14] Tong Chen, Jiqiang Liu, Yingxiao Xiang, Wenjia Niu, Endong Tong, and Zhen Han. Adversarial attack and defense in reinforcement learning-from ai security view. *Cybersecurity*, 2(1):1–22, 2019.
- [15] Wojciech M Czarnecki, Razvan Pascanu, Simon Osindero, Siddhant Jayakumar, Grzegorz Swirszcz, and Max Jaderberg. Distilling policy distillation. In *The 22nd International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 1331–1340, 2019.
- [16] Bingshui Da, Abhishek Gupta, and Yew-Soon Ong. Curbing negative influences online for seamless transfer evolutionary optimization. *IEEE transactions on cybernetics*, 49(12):4365–4378, 2018.
- [17] Sanjoy Dasgupta and Anupam Gupta. An elementary proof of a theorem of johnson and lindenstrauss. *Random Structures & Algorithms*, 22(1):60–65, 2003.

- [18] Kusum Deep, Krishna Pratap Singh, Mitthan Lal Kansal, and C Mohan. A real coded genetic algorithm for solving integer and mixed integer optimization problems. *Applied Mathematics and Computation*, 212(2):505–518, 2009.
- [19] Ambra Demontis, Marco Melis, Maura Pintor, Matthew Jagielski, Battista Biggio, Alina Oprea, Cristina Nita-Rotaru, and Fabio Roli. Why do adversarial attacks transfer? explaining transferability of evasion and poisoning attacks. In *28th {USENIX} Security Symposium ({USENIX} Security 19)*, pages 321–338, 2019.
- [20] Thanh Pham Dinh, Binh Huynh Thi Thanh, Trung Tran Ba, and Long Nguyen Binh. Multifactorial evolutionary algorithm for solving clustered tree problems: competition among cayley codes. *Memetic Computing*, 12(3):185–217, 2020.
- [21] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9185–9193, 2018.
- [22] Harris Drucker and Yann Le Cun. Improving generalization performance using double backpropagation. *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, 3(6):991–997, 1992.
- [23] Liang Feng, Yew-Soon Ong, Siwei Jiang, and Abhishek Gupta. Autoencoding evolutionary search with learning across heterogeneous problems. *IEEE Transactions on Evolutionary Computation*, 21(5):760–772, 2017.
- [24] Chris Finlay and Adam M Oberman. Scaleable input gradient regularization for adversarial robustness. *arXiv preprint arXiv:1905.11468*, 2019.
- [25] Marc Fischer et al. Online robustness training for deep reinforcement learning. *arXiv preprint arXiv:1911.00887*, 2019.
- [26] Meire Fortunato, Mohammad Gheshlaghi Azar, Bilal Piot, Jacob Menick, Matteo Hessel, Ian Osband, Alex Graves, Volodymyr Mnih, Remi Munos, Demis Hassabis, et al. Noisy networks for exploration. In *International Conference on Learning Representations (ICLR)*, 2018.

- [27] Dmitriy Fradkin and David Madigan. Experiments with random projections for machine learning. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 517–522, 2003.
- [28] A Gleave, M Dennis, N Kant, C Wild, S Levine, and S Russell. Adversarial policies: Attacking deep reinforcement learning. *The International Conference on Learning Representations (ICLR)*, 2020.
- [29] Ian Goodfellow, Nicolas Papernot, Sandy Huang, Yan Duan, Pieter Abbeel, and Jack Clark. Attacking machine learning with adversarial examples. *OpenAI*. <https://blog.openai.com/adversarial-example-research>, 2017.
- [30] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.
- [31] Ian Goodfellow et al. Explaining and harnessing adversarial examples. *International Conference on Learning Representations (ICLR)*, 2015.
- [32] Robert M Gray. *Entropy and information theory*. Springer Science & Business Media, 2011.
- [33] Abhishek Gupta and Yew-Soon Ong. Sequential knowledge transfer across problems. In *Memetic Computation*, pages 63–82. Springer, 2019.
- [34] Abhishek Gupta, Yew-Soon Ong, and Liang Feng. Insights on transfer optimization: Because experience is the best teacher. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2(1):51–64, 2017.
- [35] Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. In *Thirty-Second AAAI Conference on Artificial Intelligence (AAAI)*, 2018.
- [36] Judy Hoffman, Daniel A Roberts, and Sho Yaida. Robust learning with jacobian regularization. *arXiv preprint arXiv:1908.02729*, 2019.

- [37] John H Holland. Genetic algorithms. *Scientific american*, 267(1):66–73, 1992.
- [38] Yaqing Hou, Yew-Soon Ong, Liang Feng, and Jacek M Zurada. An evolutionary transfer reinforcement learning framework for multiagent systems. *IEEE Transactions on Evolutionary Computation (TEVC)*, 21(4):601–615, 2017.
- [39] Sandy Huang, Nicolas Papernot, Ian Goodfellow, Yan Duan, and Pieter Abbeel. Adversarial attacks on neural network policies. *International Conference on Learning Representations (ICLR) Workshop*, 2017.
- [40] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International Conference on Machine Learning*, pages 2137–2146. PMLR, 2018.
- [41] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. *arXiv preprint arXiv:1905.02175*, 2019.
- [42] Yaochu Jin, Jürgen Branke, et al. Evolutionary optimization in uncertain environments-a survey. *IEEE Transactions on evolutionary computation*, 9(3):303–317, 2005.
- [43] HM Dipu Kabir, Abbas Khosravi, Saeid Nahavandi, and Abdollah Kavousi-Fard. Partial adversarial training for neural network-based uncertainty quantification. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2019.
- [44] Simon K Kearsley. On the orthogonal transformation used for structural comparisons. *Acta Crystallographica Section A: Foundations of Crystallography*, 45(2):208–210, 1989.
- [45] Kyungnam Kim. Face recognition using principle component analysis. In *International Conference on Computer Vision and Pattern Recognition*, volume 586, page 591, 1996.
- [46] Petar Kormushev et al. Reinforcement learning in robotics: Applications and real-world challenges. *Robotics*, 2(3):122–148, 2013.

- [47] Yen-Chen Lin, Zhang-Wei Hong, Yuan-Hong Liao, Meng-Li Shih, Ming-Yu Liu, and Min Sun. Tactics of adversarial attack on deep reinforcement learning agents. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 3756–3762, 2017.
- [48] Bo Liu, Ling Wang, and Yi-Hui Jin. An effective pso-based memetic algorithm for flow shop scheduling. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(1):18–27, 2007.
- [49] Bo Liu, Ling Wang, Yi-Hui Jin, Fang Tang, and De-Xian Huang. Improved particle swarm optimization combined with chaos. *Chaos, Solitons & Fractals*, 25(5):1261–1271, 2005.
- [50] Kun Liu, Hillol Kargupta, and Jessica Ryan. Random projection-based multiplicative data perturbation for privacy preserving distributed data mining. *IEEE Transactions on knowledge and Data Engineering*, 18(1):92–106, 2005.
- [51] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.
- [52] Ajay Mandlekar, Yuke Zhu, Animesh Garg, Li Fei-Fei, and Silvio Savarese. Adversarially robust policy learning: Active construction of physically-plausible perturbations. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3932–3939. IEEE, 2017.
- [53] Fischer Mirman et al. Distilled agent dqn for provable adversarial robustness. *arXiv*, 2018.
- [54] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International Conference on Machine Learning (ICML)*, pages 1928–1937, 2016.

- [55] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.
- [56] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.
- [57] Apostolos Modas, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. Sparse-fool: a few pixels make a big difference. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9087–9096, 2019.
- [58] Seyed-Mohsen Moosavi-Dezfooli et al. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2574–2582, 2016.
- [59] Stelmo Magalhães Barros Netto, Vanessa Rodrigues Coelho Leite, Aristófanês Corrêa Silva, Anselmo Cardoso de Paiva, and Areolino de Almeida Neto. Application on reinforcement learning for diagnosis based on medical image. *Reinforcement learning*, page 379, 2008.
- [60] Maria-Irina Nicolae, Mathieu Sinn, Minh Ngoc Tran, Beat Buesser, Ambrish Rawat, Martin Wistuba, Valentina Zantedeschi, Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, et al. Adversarial robustness toolbox v1. 0.0. *arXiv preprint arXiv:1807.01069*, 2018.
- [61] Maria-Irina Nicolae, Mathieu Sinn, Minh Ngoc Tran, Beat Buesser, Ambrish Rawat, Martin Wistuba, Valentina Zantedeschi, Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, Ian Molloy, and Ben Edwards. Adversarial robustness toolbox v1.2.0. *CoRR*, 1807.01069, 2018.
- [62] Arnab Nilim and Laurent El Ghaoui. Robustness in markov decision problems with uncertain transition matrices. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 839–846. Citeseer, 2003.

- [63] Takayuki Osogami. Robust partially observable markov decision process. In *International Conference on Machine Learning*, pages 106–115. PMLR, 2015.
- [64] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against deep learning systems using adversarial examples. *arXiv preprint arXiv:1602.02697*, 1(2):3, 2016.
- [65] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 372–387. IEEE, 2016.
- [66] Anay Pattanaik, Zhenyi Tang, Shuijing Liu, Gautham Bommannan, and Girish Chowdhary. Robust deep reinforcement learning with adversarial attacks. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, pages 2040–2042. International Foundation for Autonomous Agents and Multiagent Systems, 2018.
- [67] Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 2817–2826. JMLR. org, 2017.
- [68] Xinghua Qu, Yew-Soon Ong, Abhishek Gupta, and Zhu Sun. Defending adversarial attacks without adversarial attacks in deep reinforcement learning. *arXiv preprint arXiv:2008.06199*, 2020.
- [69] Xinghua Qu, Yew-Soon Ong, Yaqing Hou, and Xiaobo Shen. Memetic evolution strategy for reinforcement learning. In *2019 IEEE Congress on Evolutionary Computation (CEC)*, pages 1922–1928. IEEE, 2019.
- [70] Xinghua Qu, Zhu Sun, Yew Soon Ong, Abhishek Gupta, and Pengfei Wei. Minimalistic attacks: How little it takes to fool deep reinforcement learning policies. *IEEE Transactions on Cognitive and Developmental Systems*, 2020.

- [71] Xinghua Qu, Zhu Sun, Yew Soon Ong, Abhishek Gupta, and Pengfei Wei. Minimalistic attacks: How little it takes to fool deep reinforcement learning policies. *IEEE Transactions on Cognitive and Developmental Systems (TCDS)*, 2020.
- [72] Xinghua Qu, Ran Zhang, Bo Liu, and Huifeng Li. An improved tlbo based memetic algorithm for aerodynamic shape optimization. *Engineering Applications of Artificial Intelligence*, 57:1–15, 2017.
- [73] Antonin RAFFIN. Reinforcement learning baseline zoo. <https://github.com/araffin/rl-baselines-zoo>, 2018.
- [74] Andrei A Rusu, Sergio Gomez Colmenarejo, Caglar Gulcehre, Guillaume Desjardins, James Kirkpatrick, Razvan Pascanu, Volodymyr Mnih, Koray Kavukcuoglu, and Raia Hadsell. Policy distillation. *arXiv preprint arXiv:1511.06295*, 2015.
- [75] Gerrit Schoettler, Ashvin Nair, Jianlan Luo, Shikhar Bahl, Juan Aparicio Ojea, Eugen Solowjow, and Sergey Levine. Deep reinforcement learning for industrial insertion tasks with visual inputs and natural rewards. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5548–5555. IEEE, 2020.
- [76] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International Conference on Machine Learning (ICML)*, pages 1889–1897, 2015.
- [77] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
- [78] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484, 2016.

- [79] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *Nature*, 550(7676):354, 2017.
- [80] César US Solis, Julio B Clempner, and Alexander S Poznyak. Handling a kullback-leibler divergence random walk for scheduling effective patrol strategies in stackelberg security games. *Kybernetika*, 55:618–640, 2019.
- [81] Jost Tobias Springenberg, Aaron Klein, Stefan Falkner, and Frank Hutter. Bayesian optimization with robust bayesian neural networks. In *Advances in neural information processing systems*, pages 4134–4142, 2016.
- [82] Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 2019.
- [83] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.
- [84] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.
- [85] Mao Tan, Hua-Li Yang, and Yong-Xin Su. Genetic algorithms with greedy strategy for green batch scheduling on non-identical parallel machines. *Memetic Computing*, 11(4):439–452, 2019.
- [86] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 23–30. IEEE, 2017.
- [87] Arne Traue, Gerrit Book, Wilhelm Kirchgässner, and Oliver Wallscheid. Toward a reinforcement learning environment toolbox for intelligent electric motor control. *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, 2020.

- [88] Kristal K Trejo, Julio B Clempner, and Alexander S Poznyak. Adapting attackers and defenders patrolling strategies: A reinforcement learning approach for stackelberg security games. *Journal of Computer and System Sciences*, 95:35–54, 2018.
- [89] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado Hasselt, Marc Lanctot, and Nando Freitas. Dueling network architectures for deep reinforcement learning. In *International Conference on Machine Learning (ICML)*, pages 1995–2003, 2016.
- [90] Pengfei Wei, Yiping Ke, Xinghua Qu, and Tze-Yun Leong. Subdomain adaptation with manifolds discrepancy alignment. *arXiv:2005.03229*, 2020.
- [91] Pengfei Wei, Xinghua Qu, Yiping Ke, Tze-Yun Leong, and Yew Soon Ong. Adaptive knowledge transfer based on transfer neural kernel network. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, pages 1485–1493, 2020.
- [92] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google’s neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.
- [93] Yuhuai Wu, Elman Mansimov, Roger B Grosse, Shun Liao, and Jimmy Ba. Scalable trust-region method for deep reinforcement learning using kronecker-factored approximation. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5279–5288, 2017.
- [94] Chaowei Xiao, Xinlei Pan, Warren He, Jian Peng, Mingjie Sun, Jinfeng Yi, Bo Li, and Dawn Song. Characterizing attacks on deep reinforcement learning. *arXiv preprint arXiv:1907.09470*, 2019.
- [95] Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2730–2739, 2019.

- [96] Huan Xu and Shie Mannor. Distributionally robust markov decision processes. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2505–2513, 2010.
- [97] Erkun Yang, Tongliang Liu, Cheng Deng, and Dacheng Tao. Adversarial examples for hamming space search. *IEEE transactions on cybernetics*, 2018.
- [98] Xiaoyong Yuan, Pan He, Qile Zhu, and Xiaolin Li. Adversarial examples: Attacks and defenses for deep learning. *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, 30(9):2805–2824, 2019.
- [99] Huan Zhang, Hongge Chen, Chaowei Xiao, Bo Li, Duane Boning, Cho-Jui Hsieh, et al. Robust deep reinforcement learning against adversarial perturbations on state observations. *Advances in Neural Information Processing Systems (NeurIPS) (NeurIPS)*, 33, 2020.
- [100] Wei Emma Zhang, Quan Z Sheng, Ahoud Alhazmi, and Chenliang Li. Adversarial attacks on deep-learning models in natural language processing: A survey. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(3):1–41, 2020.
- [101] Zhen Zhang, Yew-Soon Ong, Dongqing Wang, and Binqiang Xue. A collaborative multiagent reinforcement learning method based on policy gradient potential. *IEEE transactions on cybernetics (TCYB)*, 2019.
- [102] Zexuan Zhu, Yew-Soon Ong, and Manoranjan Dash. Markov blanket-embedded genetic algorithm for gene selection. *Pattern Recognition*, 40(11):3236–3248, 2007.