Boosting Transferable Adversarial Attacks against Deep Reinforcement Learning

Zexin Li[1] **Ruili Yao**[2] **Yiming Zeng**[3] **Xiaoxue Gao**[4]*

[1]Nanyang Technological University

[2]University of California, Riverside [3]University of Connecticut

[4]School of Artificial Intelligence,

The Chinese University of Hong Kong, Shenzhen

zexin.li@ntu.edu.sg

Abstract

Most adversarial attacks on deep reinforcement learning (DRL) assume white-box access to the victim policy, which rarely holds in practice. This paper studies *transfer-based black-box* attacks on DRL: the attacker crafts observation perturbations on a white-box surrogate agent and feeds them to an unknown victim. We formulate the attack as return minimization under a per-step perturbation budget. We first show that transplanting transferable image-classification attacks (FGSM, MI-FGSM, and NI-FGSM) with a per-step objective yields perturbations that transfer but are no stronger than random noise of the same budget. We then propose a *trajectory-level* attack that optimizes a sequence of perturbations over a receding horizon through a differentiable model of the environment and a temperature-smoothed surrogate policy, with the same optimizers. On CartPole-v1 with ten DQN and DDQN agents and 100 surrogate–victim pairs, the trajectory-level attack outperforms per-step attacks and random noise in the white-box, cross-model, and cross-algorithm settings.

1 Introduction

Deep reinforcement learning (DRL) has enabled complex control from learned representations (Mnih et al., 2015; Peng et al., 2018; Kahn et al., 2018), but its policies remain vulnerable to small observation perturbations (Huang et al., 2017; Lin et al., 2017; Zhang et al., 2020). In a sequential system, a corrupted observation affects both the current action and the states encountered later. Evaluating this vulnerability therefore requires measuring cumulative return, beyond whether an attack changes an individual action.

Many attacks assume access to the victim's parameters. For an inaccessible deployed policy, a practical alternative is to craft perturbations on a white-box surrogate and transfer them to the victim (Huang et al., 2017; Pan et al., 2020). Prior work demonstrates shared adversarial features across MDPs (Korkmaz, 2022) and natural black-box vulnerabilities even after adversarial training (Korkmaz, 2023). These findings establish that transfer is possible, but leave an important design question: how should an attacker exploit sequential dynamics to make transferred perturbations more damaging?

A natural baseline applies transferable image-classification attacks independently at each step. However, changing the surrogate's preferred action is only a proxy for reducing the victim's return. A well-trained controller may correct an isolated error, while a perturbation tailored to one decision boundary may become ineffective on another policy. On CartPole, we find that per-step FGSM, MI-FGSM, and NI-FGSM transfer across agents but do not outperform random corner noise at the same budget. This motivates optimizing the consequences of a sequence of perturbations rather than repeatedly targeting a single action.

We optimize perturbation sequences through differentiable environment dynamics over a receding horizon. A temperature-smoothed surrogate policy permits differentiable rollouts, and momentum or Nesterov updates seek trajectories approaching failure. Only the first perturbation is applied before replanning. The objective thus incorporates future state evolution while retaining the same per-step perturbation budget as the baselines.

Our contributions are a return-minimization formulation linking per-step and trajectory objectives (§3.1), a trajectory-level attack using FGSM, MI-FGSM, and NI-FGSM (§3.2), and an evaluation over ten DQN/DDQN agents and 100 surrogate–victim pairs (§4). Trajectory attacks outperform per-step attacks and random noise in white-box, cross-model, and cross-algorithm settings.

*Corresponding author. Assistant Professor, School of Artificial Intelligence, The Chinese University of Hong Kong, Shenzhen.

Proceedings of the Workshop on Secure, Safe, and Privacy-Preserving Speech and Audio AI (TrustAudio 2026), pages 50–56
November 10, 2026 ©2026 Association for Computational Linguistics

2 Related Work

Transferable attacks. Iterative optimization (Kurakin et al., 2018; Madry et al., 2018), momentum (Dong et al., 2018), Nesterov look-ahead (Lin et al., 2020), variance tuning (Wang and He, 2021), and input diversity (Xie et al., 2019) improve transfer. Applications include face recognition (Li et al., 2023b), speech (Gao et al., 2024), text generation (Li et al., 2023a), efficiency attacks (Chen et al., 2023, 2026), and backdoors (Cheng et al., 2026; Guo et al., 2022).

Adversarial attacks on DRL. Huang et al. (2017) studied FGSM (Goodfellow et al., 2015) attacks and transfer in Atari (Bellemare et al., 2013); Lin et al. (2017) planned target actions through learned dynamics. Other work learns white-box adversaries (Zhang et al., 2020; Sun et al., 2022), uses domain randomization for transfer (Pan et al., 2020), or manipulates robot teams (Nikkhoo et al., 2023; Li et al., 2026a). Shared adversarial directions (Korkmaz, 2022) and natural black-box vulnerabilities (Korkmaz, 2023) motivate our focus on trajectory optimization.

Diagnosing and evaluating robustness. Korkmaz and Brown-Cohen (2023) detect adversarial directions using local quadratic approximations of the policy loss. Korkmaz (2024) analyzes how unstable directions evolve across states and time, revealing limitations of robust training. Korkmaz (2026) shows that enforcing local invariance can impair counterfactual value learning.

Sequence-level objectives. SPEAR (Li et al., 2026b) uses order-aware symbolic trajectory alignment to reward language-model reasoning distillation. It shares our emphasis on sequence structure, but addresses distillation rather than adversarial control.

3 Method

3.1 Problem Formulation

We model the task as a Markov decision process $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, r, \gamma)$. The victim is a value-based agent with Q-network $Q_v(\cdot, \cdot; \theta_v)$ and greedy policy $\pi_v(s) = \arg\max_a Q_v(s, a; \theta_v)$. At each step t, the attacker observes the true state s_t and injects a perturbation δ_t into the victim's observation (Figure 1), so that the victim acts on $\tilde{s}_t = s_t + \delta_t$ while the environment evolves according to the true state. The attacker aims to minimize the victim's expected return under a per-step ℓ_∞ budget ϵ:

$$\begin{aligned} \min_{\{\delta_t\}_{t=0}^{H}} \; & J(\delta) = \mathbb{E}\Big[\textstyle\sum_{t=0}^{H} \gamma^t r(s_t, a_t)\Big] \\ \text{s.t.} \; & a_t = \pi_v(s_t + \delta_t), \quad \|\delta_t\|_\infty \le \epsilon, \\ & s_{t+1} \sim P(\cdot \mid s_t, a_t), \end{aligned} \tag{1}$$

where H is the episode horizon. In the black-box setting, θ_v is unknown and the attacker cannot query π_v while crafting perturbations. It trains a surrogate Q-network $Q_s(\cdot, \cdot; \theta_s)$ in the same environment, possibly with a different seed or algorithm, and uses it as a stand-in for Q_v. Even with a surrogate, Eq. (1) is hard to optimize: the return is a non-differentiable function of a long sequence of discrete actions and stochastic transitions.

Per-step relaxation. The standard relaxation, which we use as a baseline, optimizes each perturbation independently. We choose a lure action $a^\dagger$ whose persistent execution drives the system toward failure (for CartPole, always pushing the cart in the same direction), and at each step we solve

$$\delta_t^* = \arg\max_{\|\delta\|_\infty \le \epsilon} \log \operatorname{softmax}\big(Q_s(s_t + \delta, \cdot)\big)_{a^\dagger} \tag{2}$$

with FGSM (Goodfellow et al., 2015), MI-FGSM (Dong et al., 2018), or NI-FGSM (Lin et al., 2020). This objective is myopic: one wrong action changes the state only slightly and the victim's next decision can undo it, so a per-step attack must succeed at almost every step, which is hard under a small budget and harder after transfer.

3.2 Trajectory-Level Transfer Attack

We instead optimize perturbations with respect to their effect on the future trajectory. We assume the attacker has a differentiable model f of the environment dynamics, $\hat{s}_{k+1} = f(\hat{s}_k, u_k)$, where u_k is a continuous relaxation of the action. For CartPole, f is the known physical model; in general, it can be learned from interaction data, as in the enchanting attack (Lin et al., 2017). To make the rollout differentiable, we replace the surrogate's greedy action with a temperature-smoothed policy

$$\tilde{\pi}_s(a \mid x) = \operatorname{softmax}\big(Q_s(x, \cdot)/\kappa\big)_a, \tag{3}$$

and apply the expected action $u_k = \mathbb{E}_{a \sim \tilde{\pi}_s(\cdot \mid \hat{s}_k + \delta_k)}[a]$ (for CartPole, the expected force). At step t, the attacker optimizes a sequence of K perturbations $\Delta_t = (\delta_t, \ldots, \delta_{t+K-1})$ to maximize the proximity of the predicted states to

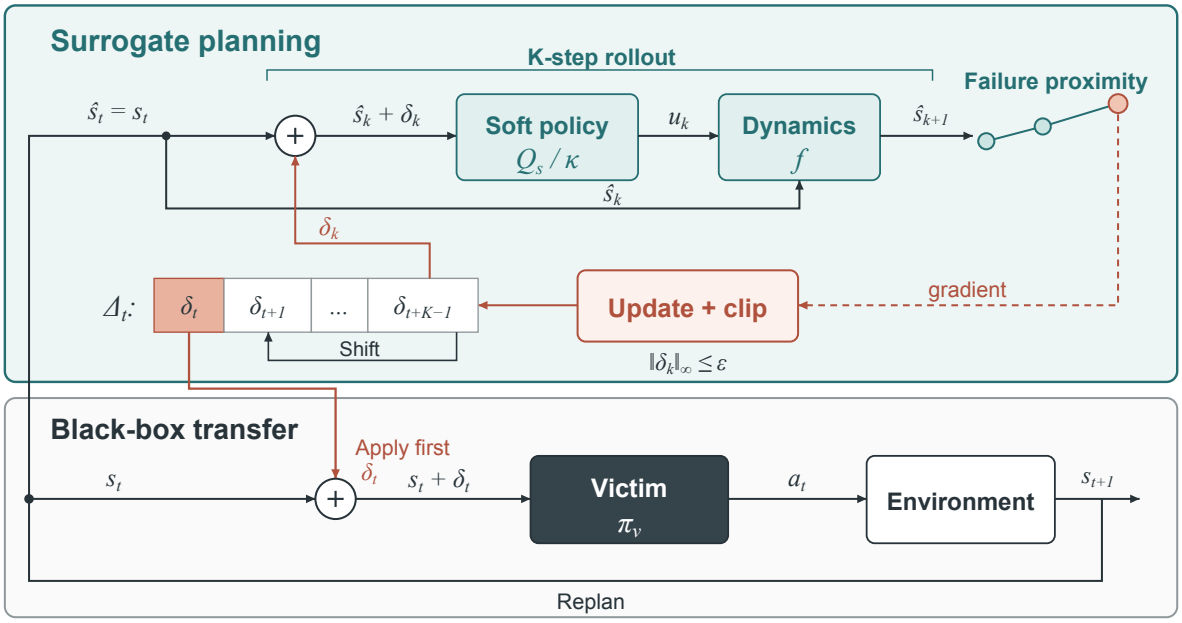


Figure 1: Trajectory-level transfer attack. The attacker optimizes a bounded perturbation sequence through a temperature-smoothed surrogate and differentiable dynamics to maximize predicted failure proximity. Only the first perturbation is applied to the victim's observation; the remainder warm-starts the next plan. The environment evolves from the true state. Dashed arrows denote gradients; no victim parameters or queries are used during crafting.

the failure region:

$$\max_{\|\delta_k\|_\infty \le \epsilon} \mathcal{L}_{\text{traj}}(\Delta_t) = \sum_{k=1}^{K} \phi(\hat{s}_{t+k}), \quad \hat{s}_t = s_t, \tag{4}$$

where $\phi(s) = \max\{(\theta/\theta_{\max})^2, (x/x_{\max})^2\}$ measures how close the pole angle θ and the cart position x are to their termination thresholds. Following model-predictive control, only δ_t is applied, and the remaining perturbations warm-start the optimization at the next step.

We solve Eq. (4) with the same operators as in the per-step case, applied to the whole sequence Δ_t. With step size $\alpha = \epsilon/T$ and momentum decay μ, MI-FGSM accumulates an ℓ_1-normalized gradient momentum,

$$g_{j+1} = \mu\, g_j + \frac{\nabla_\Delta \mathcal{L}_{\text{traj}}(\hat{\Delta}_j)}{\|\nabla_\Delta \mathcal{L}_{\text{traj}}(\hat{\Delta}_j)\|_1}, \tag{5}$$

with $\hat{\Delta}_j = \Delta_j$ for MI-FGSM and the Nesterov look-ahead $\hat{\Delta}_j = \Delta_j + \alpha\mu\, g_j$ for NI-FGSM, followed by

$$\Delta_{j+1} = \text{Clip}_\epsilon\big(\Delta_j + \alpha \cdot \text{sign}(g_{j+1})\big). \tag{6}$$

FGSM takes a single step of size ϵ. Algorithm 1 summarizes the full attack.

Algorithm 1 Trajectory-Level Transfer Attack

Require: Surrogate Q-network Q_s, black-box victim policy π_v, dynamics model f, budget ϵ, horizon K, iterations T, momentum decay μ, temperature κ, operator $\mathcal{O} \in \{\text{FGSM}, \text{MI}, \text{NI}\}$
Ensure: Perturbation sequence that minimizes the victim's return J in Eq. (1)
Observe s_0; initialize $\Delta \leftarrow \mathbf{0}_K$
for $t = 0, 1, \ldots, H$ **do**
  $\alpha \leftarrow \epsilon/T$, $\Delta_0 \leftarrow \Delta$, $g_0 \leftarrow \mathbf{0}$
  for $j = 0, \ldots, T-1$ **do**
    $\hat{\Delta}_j \leftarrow \Delta_j + \alpha\mu\, g_j$ (NI) or Δ_j (MI)
    Roll out f from s_t with $\tilde{\pi}_s$ (Eq. (3)) under $\hat{\Delta}_j$
    Compute $\mathcal{L}_{\text{traj}}$ via Eq. (4) and backpropagate
    Update g_{j+1} via Eq. (5) and Δ_{j+1} via Eq. (6)
  end for ▷ FGSM: one step of size ϵ
  Victim acts $a_t \leftarrow \pi_v(s_t + \delta_t)$, $\delta_t = \Delta_T[0]$ ▷ Transfer
  Observe r_t, s_{t+1}; **break** if terminal
  $\Delta \leftarrow \Delta_T$ shifted by one step ▷ Warm start
end for

Transfer rationale. The objective uses dynamics shared by surrogate and victim to target failure over multiple steps, rather than a single surrogate action. Temperature smoothing reduces sensitivity to the surrogate's exact decision boundary; momentum stabilizes optimization (Dong et al., 2018). Their individual contributions require separate ablations.

Table 1: CartPole victim returns (lower is stronger): mean and standard deviation over five victims (white-box), 20 ordered same-algorithm pairs (cross-model), or 25 pairs per direction (cross-algorithm). Random baselines are evaluated directly at the same budget. Middle/bottom groups use per-step/trajectory objectives; bold marks each column's strongest attack.

	White-box				Cross-model transfer				Cross-algorithm transfer			
Budget	0.05		0.1		0.05		0.1		0.05		0.1	
Victim	DQN	DDQN	DQN	DDQN	DQN	DDQN	DQN	DDQN	DDQN	DQN	DDQN	DQN
No attack	500.0±0	485.4±24	500.0±0	485.4±24	500.0±0	485.4±24	500.0±0	485.4±24	485.4±24	500.0±0	485.4±24	500.0±0
Uniform noise	375.0±130	304.7±193	172.7±53	247.0±193	375.0±130	304.7±193	172.7±53	247.0±193	304.7±193	375.0±130	247.0±193	172.7±53
Sign noise	240.5±92	284.9±204	74.3±38	121.2±133	240.5±92	284.9±204	74.3±38	121.2±133	284.9±204	240.5±92	121.2±133	74.3±38
FGSM	322.0±140	336.0±179	262.5±177	214.7±163	330.9±138	314.7±204	173.4±128	274.6±185	302.1±212	411.2±124	163.4±138	364.5±140
MI-FGSM	315.8±157	277.0±200	159.5±174	125.4±188	338.9±140	317.0±209	147.7±120	211.2±190	317.7±214	400.9±127	148.6±160	324.0±166
NI-FGSM	304.3±163	286.3±201	153.5±176	125.4±189	360.5±138	306.7±206	158.2±121	215.6±189	319.2±204	400.2±130	169.7±184	322.3±164
Traj-FGSM	113.0±108	243.9±211	56.5±52	30.6±23	**190.0**±147	**240.0**±205	77.3±54	43.3±30	**259.1**±214	**181.8**±126	**41.6**±27	82.3±74
Traj-MI-FGSM	**73.5**±49	**157.3**±178	19.5±13	15.7±6	203.4±154	268.1±217	60.6±95	**41.1**±40	266.9±216	186.3±162	47.5±47	**53.1**±57
Traj-NI-FGSM	74.1±44	173.1±177	**18.5**±12	**13.9**±4	202.9±149	271.6±219	**59.6**±78	45.5±40	268.9±214	189.3±154	55.1±50	58.7±62

4 Experiments

4.1 Experimental Setup

Environment and agents. We use CartPole-v1 from Gymnasium (Brockman et al., 2016; Barto et al., 1983). The observation is a raw 4-D vector (cart position, cart velocity, pole angle, and pole angular velocity), there are two actions (push left or right), and the agent receives a reward of $+1$ per step, up to a maximum return of 500. We train DQN (Mnih et al., 2015) and Double DQN (DDQN) (Van Hasselt et al., 2016) agents with five seeds each, which gives ten models. All agents use a two-layer MLP (128 units), Adam (learning rate 5×10^{-4}), batch size 128, soft target updates, and 600 training episodes, and we keep the best checkpoint under periodic greedy evaluation. Figure 2 shows the training curves. Without attack, eight agents obtain the maximum return of 500, and the other two average 489.3 and 437.8.

Attacks. We compare the per-step FGSM, MI-FGSM, and NI-FGSM attacks (Eq. (2), lure action $a^\dagger$ = push left) with their trajectory-level counterparts Traj-FGSM, Traj-MI-FGSM, and Traj-NI-FGSM (Algorithm 1). All iterative attacks use $T = 10$ iterations and $\mu = 1.0$. The trajectory attacks use horizon $K = 20$ and temperature $\kappa = 2$ and roll out the CartPole equations of motion. We also report two random baselines with the same budget: *Uniform* noise drawn from $[-\epsilon, \epsilon]^4$, and *Sign* noise that places every coordinate at a random corner, $\pm\epsilon$. Budgets are given in raw observation units, and we report $\epsilon \in \{0.05, 0.1\}$.

Protocol. We evaluate three settings: *white-box* (the surrogate is the victim), *cross-model* (same algorithm, different seed; 20 ordered pairs per algorithm), and *cross-algorithm* (DQN↔DDQN; 25 pairs per direction). Each evaluation runs 10 episodes with fixed environment seeds, and we report the victim's average return (lower is stronger).

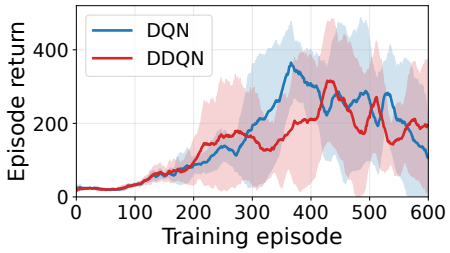


Figure 2: CartPole training returns over five seeds: 20-episode moving average with one standard deviation shaded. Periodic greedy evaluation selects each checkpoint.

4.2 Main Results

Per-step attacks transfer but are weak. At $\epsilon = 0.1$, white-box MI-FGSM and NI-FGSM reduce DQN returns to 159.5 and 153.5, versus 262.5 for FGSM (Table 1). Cross-model MI-FGSM achieves 147.7, but remains weaker than Sign noise (74.3). Cross-algorithm transfer is asymmetric: DDQN-crafted attacks leave DQN returns above 320.

Trajectory-level attacks are much stronger. In the white-box setting at a budget of 0.1, Traj-NI-FGSM reduces the return to 18.5 on DQN and 13.9 on DDQN, 4–9× lower than Sign noise. At the smaller budget of 0.05, where all per-step attacks leave the victims above a return of 277, Traj-MI-FGSM reaches 73.5 on DQN and 157.3 on DDQN.

Transfer across seeds and algorithms. For cross-model transfer at a budget of 0.1, the trajectory attacks reduce DQN and DDQN victims to

59.6 and 41.1, compared with 147.7 and 211.2 for the best per-step attack and 74.3 and 121.2 for Sign noise. They also remove the asymmetry of cross-algorithm transfer: perturbations crafted on DDQN reduce DQN victims to 53.1, compared with more than 320 for per-step attacks. This supports our hypothesis that an objective defined through the shared dynamics transfers better than one defined through individual actions. At a budget of 0.05, they remain the strongest in every column, with a smaller margin (e.g., 190.0 compared with 240.5 for Sign noise).

Momentum and look-ahead. At $\epsilon = 0.1$, momentum and look-ahead reduce white-box DQN returns from 56.5 (Traj-FGSM) to about 19. The three trajectory operators perform similarly under transfer, suggesting the trajectory objective drives most of the improvement.

5 Conclusion

On CartPole, per-step attacks transfer but are weaker than random corner noise. Optimizing perturbation sequences through differentiable dynamics strengthens white-box and transfer attacks, motivating defenses against temporally coordinated perturbations. Future work will address high-dimensional observations and learned dynamics.

Limitations

Our evaluation is limited to CartPole's low-dimensional observations, known dynamics, and DQN/DDQN agents. The attack assumes access to the true state and a differentiable dynamics model; its effectiveness with visual inputs, learned or inaccurate dynamics, other environments, and adversarially trained victims remains untested. Reported standard deviations describe variability across victims or pairs, but shared models prevent treating all pairs as independent samples for significance testing. Ablations of the planning horizon, temperature, and warm start, as well as measurements of attack latency, remain future work.

Ethical Statement

This work studies adversarial attacks to support robustness evaluation of reinforcement learning systems. The reported experiments use trained agents in simulated CartPole. The method has dual-use potential: transferable observation perturbations could be misused to disrupt deployed controllers, with possible physical harm in safety-critical applications. Such testing should therefore be restricted to systems for which explicit authorization has been obtained and conducted in simulation or controlled environments with appropriate safeguards.

Acknowledgments

This research is supported by UDF01004676.

References

Andrew G. Barto, Richard S. Sutton, and Charles W. Anderson. 1983. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13(5):834–846.

Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. 2013. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. 2016. OpenAI gym. *arXiv preprint arXiv:1606.01540*.

Yiming Chen, Simin Chen, Zexin Li, Wei Yang, Cong Liu, Robby T. Tan, and Haizhou Li. 2023. Dynamic transformers provide a false sense of efficiency. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7164–7180.

Yiming Chen, Zexin Li, Xianghu Yue, Robby T. Tan, and Haizhou Li. 2026. NaturalSloth: Revisiting denial-of-service attacks on large language models. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Yezheng Cheng, Zexin Li, Jiaqi Wu, Zhihong Zhang, and Simin Chen. 2026. VLMSysTrojan: Stealthy system-aware backdoor attacks against vision-language models. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. 2018. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9185–9193.

Xiaoxue Gao, Zexin Li, Yiming Chen, Cong Liu, and Haizhou Li. 2024. Transferable adversarial attacks against ASR. *IEEE Signal Processing Letters*.

Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*.

Junfeng Guo, Ang Li, and Cong Liu. 2022. Backdoor detection and mitigation in competitive reinforcement learning. *arXiv preprint arXiv:2202.03609*.

Sandy Huang, Nicolas Papernot, Ian Goodfellow, Yan Duan, and Pieter Abbeel. 2017. Adversarial attacks on neural network policies. *arXiv preprint arXiv:1702.02284*.

Gregory Kahn, Adam Villaflor, Bosen Ding, Pieter Abbeel, and Sergey Levine. 2018. Self-supervised deep reinforcement learning with generalized computation graphs for robot navigation. *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–8.

Ezgi Korkmaz. 2022. Deep reinforcement learning policies learn shared adversarial features across MDPs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 7229–7238.

Ezgi Korkmaz. 2023. Adversarial robust deep reinforcement learning requires redefining robustness. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 8369–8377.

Ezgi Korkmaz. 2024. Understanding and diagnosing deep reinforcement learning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pages 25265–25275. PMLR.

Ezgi Korkmaz. 2026. How to lose inherent counterfactuality in reinforcement learning. In *International Conference on Learning Representations*.

Ezgi Korkmaz and Jonah Brown-Cohen. 2023. Detecting adversarial directions in deep reinforcement learning to make robust decisions. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pages 17534–17543. PMLR.

Alexey Kurakin, Ian J Goodfellow, and Samy Bengio. 2018. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pages 99–112. Chapman and Hall/CRC.

Yufei Li, Zexin Li, Yingfan Gao, and Cong Liu. 2023a. White-box multi-objective adversarial attack on dialogue generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1778–1792.

Zexin Li, Bangjie Yin, Taiping Yao, Junfeng Guo, Shouhong Ding, Simin Chen, and Cong Liu. 2023b. Sibling-attack: Rethinking transferable adversarial attacks against face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24626–24637.

Zexin Li, Ziliang Zhang, Hyoseung Kim, and Cong Liu. 2026a. PIMbot: A self-adaptive attack framework for adversarial manipulation of multi-robot reinforcement learning. *arXiv preprint arXiv:2605.23027*.

Zhuochun Li, Yuelyu Ji, Yiming Zeng, and Daqing He. 2026b. SPEAR: Distilling domain-adaptive reasoning skeletons via sequential symbolic alignment in reinforcement learning. *arXiv preprint arXiv:2608.26550*.

Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and John E. Hopcroft. 2020. Nesterov accelerated gradient and scale invariance for adversarial attacks. In *International Conference on Learning Representations*.

Yen-Chen Lin, Zhang-Wei Hong, Yuan-Hong Liao, Meng-Li Shih, Ming-Yu Liu, and Min Sun. 2017. Tactics of adversarial attack on deep reinforcement learning agents. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 3756–3762.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards deep learning models resistant to adversarial attacks. *International Conference on Learning Representations (ICLR)*.

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. 2015. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533.

Shahab Nikkhoo, Zexin Li, Aritra Samanta, Yufei Li, and Cong Liu. 2023. PIMbot: Policy and incentive manipulation for multi-robot reinforcement learning in social dilemmas. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5630–5636.

Xinlei Pan, Yulong Cao, Xindi Wu, Eric Zelikman, Chaowei Xiao, Yanan Sui, Rudrasis Chakraborty, and Ronald S Fearing. 2020. Transferable adversarial attacks on deep reinforcement learning with domain randomization. In *CVPR Workshops*.

Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. 2018. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Transactions on Graphics (TOG)*, 37(4):143.

Yanchao Sun, Ruijie Zheng, Yongyuan Liang, and Furong Huang. 2022. Who is the strongest enemy? towards optimal and efficient evasion attacks in deep RL. In *International Conference on Learning Representations (ICLR)*.

Hado Van Hasselt, Arthur Guez, and David Silver. 2016. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30.

Xiaosen Wang and Kun He. 2021. Enhancing the transferability of adversarial attacks through variance tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1924–1933.

Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. 2019. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2730–2739.

Huan Zhang, Hongge Chen, Chaowei Xiao, Bo Li, Mingyan Liu, Duane Boning, and Cho-Jui Hsieh. 2020. Robust deep reinforcement learning against adversarial perturbations on state observations. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 21024–21037.