# DiffFP: Learning Behaviors from Scratch via Diffusion-based Fictitious-Play

**Akash Karthikeyan** , **Yash Vardhan Pant**
University of Waterloo
{a9karthi, yash.pant}@uwaterloo.ca

## Abstract

Reinforcement learning has demonstrated success in learning strategic behaviors in multi-agent settings. However, achieving robust performance in dynamic, continuous state-action games remains a significant challenge. Agents must anticipate diverse and potentially unseen opponent strategies to ensure adaptability in multi-agent environments. These challenges often lead to slow convergence or even failure to converge to a Nash equilibrium. To address these challenges, we propose *DiffFP*, a fictitious-play (FP) framework that estimates the best response to unseen opponents while learning a robust and multimodal behavioral policy. Specifically, we approximate the best response using a *diffusion-policy* that leverages generative modeling to learn adaptive and multimodal strategies. Through extensive empirical evaluation, we demonstrate that the proposed FP framework converges towards an approximate Nash equilibrium in continuous state-action zero-sum games. We validate our method on complex multi-agent environments, including racing and multi-particle dynamic games. Our results show that the learned policies are robust against diverse opponents and outperform baseline reinforcement learning policies. Our approach achieves upto *3*× *faster convergence* and *30*× *higher success rates* on average against RL baselines, demonstrating its robustness to opponent strategies and stability across training iteration.

## 1 Introduction

Multi-agent reinforcement learning (MARL) has achieved remarkable success in domains like Go, Atari, chess, shogi, and StarCraft 2 [18, 22, 23, 25]. In these systems, agents learn strategies via policy optimization techniques parameterized by deep neural networks, demonstrating the transformative impact of MARL in complex, dynamic environments.

A common approach to learning robust policies in interactive, competitive settings is *self-play*, where agents iteratively improve by training against copies of themselves or evolving opponents. Fictitious Play (FP) [2] is a game-theoretic algorithm that formalizes this idea. In FP, each agent computes the *best response* to the empirical average of its opponents' past actions. Under certain conditions, the sequence of strategies converges to a Nash equilibrium. Recent advances have extended FP to more complex settings, including extensive-form and Markov games [10, 20]. Fictitious Self Play (FSP) [10] thus introduces a learning-based class of algorithms that approximate Extensive-Form Fictitious Play by using reinforcement learning to estimate best responses and supervised learning to update the average strategy.

In practice, computing an exact best response in continuous and high-dimensional environments is challenging. Reinforcement learning (RL) methods are typically employed to approximate the best responses; however, standard RL policies tend to be unimodal. This unimodality limits their ability to capture the inherent multimodality of complex tasks, often resulting in policies that overfit to recent opponent behaviors while neglecting previously encountered strategies.

**Related Work.** Recent work has demonstrated the power of diffusion models as expressive generative tools for capturing diverse data distributions, with notable success in behavioral cloning, imitation learning, and planning [1, 3, 4, 11, 13, 19, 24]. However, most existing approaches focus on learning from demonstrations, while applications to learning from scratch in online Reinforcement Learning (RL) remain relatively under-explored. Although several methods have explored the use of diffusion models in offline RL [14, 27], their integration into online settings poses significant challenges due to the dynamic nature of value estimation as policies evolve. Moreover, diffusion policies are defined implicitly through a stochastic process rather than an explicit parametric mapping; it is non-trivial to apply policy gradient methods: unlike traditional policy-based RL algorithms. Furthermore, we aim to extend this framework to multi-agent and dynamic environments, where interactions and opponent adaptations introduce additional complexity.

**Contributions.** In this work, we propose a FP-based framework that leverages diffusion policies to estimate best responses. This integration enables (i) learning from scratch via an iterative fictitious-play, (ii) capturing multimodal action and behavior distributions, and (iii) improving sample efficiency and convergence toward an approximate Nash equilibrium. By harnessing the expressiveness of diffusion models, our approach addresses key limitations of traditional RL-

based best-response approximations in continuous spaces, resulting in more robust and adaptable MARL policies.

## 2 Preliminaries and Problem Statement

We consider a Partially Observable Markov Games (POMG) [7] defined by the tuple $(\mathcal{I}, \mathcal{S}, b^0, \{\mathcal{A}_i\}, \{\mathcal{O}_i\}, \mathcal{T}, \{R_i\})$, where:

- $\mathcal{I} = \{1, \cdots, n\}$ is the finite set of agents;
- $\mathcal{S}_i \subseteq \mathbb{R}^{d_s}$ is the continuous state space for agent $i$, and $\mathcal{S} = \prod_{i \in \mathcal{I}} \mathcal{S}_i$ is the joint state space, with $\overrightarrow{s} = (s_1, \cdots, s_n) \in \mathcal{S}$ denoting the joint state;
- $b^0 \in \Delta(\mathcal{S})$ denotes the initial state distribution;
- $\mathcal{A}_i \subseteq \mathbb{R}^{d_a}$ is the continuous action space for agent $i$, and $\mathcal{A} = \prod_{i \in \mathcal{I}} \mathcal{A}_i$ is the joint action space, with $\overrightarrow{a} = (a_1, \cdots, a_n) \in \mathcal{A}$;
- $\mathcal{O}_i$ is the continuous observation space for agent $i$, and $\mathcal{O} = \prod_{i \in \mathcal{I}} \mathcal{O}_i$ is the joint observation space, with $\overrightarrow{o} = (o_1, \cdots, o_n) \in \mathcal{O}$;
- $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the (joint) transition function;
- $R_i : \mathcal{S} \times \mathcal{A} \to [r_{\min}, r_{\max}] \subset \mathbb{R}$ is the bounded reward function for agent $i$, where $r_{\min} \leq R_i(\overrightarrow{s}, \overrightarrow{a}) \leq r_{\max}$ for all $(\overrightarrow{s}, \overrightarrow{a}) \in \mathcal{S} \times \mathcal{A}$.

**A (stochastic) policy** for agent $i$ is defined as a mapping $\pi^i : \mathcal{O}_i \to \Delta(\mathcal{A}_i)$, where $\mathcal{O}_i$ is the observation space of agent $i$ and $\Delta(\mathcal{A}_i)$ denotes the set of probability distributions over the action space $\mathcal{A}_i$. The joint strategy of all agents is denoted as $\boldsymbol{\pi} = (\pi^1, \ldots, \pi^n) = (\pi^i, \boldsymbol{\pi}^{-i})$, where $\boldsymbol{\pi}^{-i}$ represents the strategy profile of all agents except agent $i$.

We consider a two-player game without loss of generality, noting that this formulation extends naturally to multi-agent settings with more than two agents or teams. The joint strategy is denoted by $\boldsymbol{\pi} = (\pi^{\text{ego}}, \pi^{\text{opp}})$, where $\pi^{\text{ego}}$ and $\pi^{\text{opp}}$ represent the behavioral strategies of the ego and opponent agents, respectively.

**Definition 1** (Best Response)**.** *Given the opponent's strategy $\boldsymbol{\pi}^{-i}$, the best-response strategy $\pi_i^{\text{BR}}$ for agent $i$ is defined as*

$$\pi_i^{\text{BR}} \in \arg\max_{\pi^i \in \Pi^i} f_i(\pi^i, \boldsymbol{\pi}^{-i}), \tag{1}$$

*where $\Pi^i$ denotes the set of admissible policies for agent $i$, and $f_i : \boldsymbol{\pi} \mapsto \mathbb{R}$ is a continuous payoff function representing the expected cumulative reward for agent $i$.*

**Definition 2** ($\epsilon$-Nash Equilibrium)**.** *In a zero-sum game, the reward functions satisfy $R^{ego} = -R^{opp}$, yielding a minimax structure where one agent's gain equals the other's loss. The objective is to find equilibrium strategies that minimize exploitability in the presence of an adversarial opponent. A joint strategy $\boldsymbol{\pi}^* = (\pi^{ego,*}, \pi^{opp,*})$ constitutes an $\epsilon$-Nash equilibrium if, for all $i \in \{ego, opp\}$, the following condition holds:*

$$f_i\big(\pi^{i,*}, \boldsymbol{\pi}^{-i,*}\big) \geq f_i\big(\pi^i, \boldsymbol{\pi}^{-i,*}\big) - \epsilon, \quad \forall\, \pi^i \in \Pi^i. \tag{2}$$

*The deviation from exact equilibrium is quantified by the exploitability of agent $i$, defined as*

$$\epsilon_i = f_i\big(\pi_i^{\text{BR}}, \boldsymbol{\pi}^{-i}\big) - f_i\big(\pi^i, \boldsymbol{\pi}^{-i}\big), \tag{3}$$

*and the total exploitability is given by $\epsilon = \sum_{i \in \mathcal{I}} \epsilon_i$.*

**Algorithm 1** Fictitious-Play

1: **Initialize** $\pi_0^i$ randomly for each agent $i \in \{\text{ego}, \text{opp}\}$.
2: **for** $k = 0, \ldots, K-1$ **do**
3:  **for** each agent $i \in \{\text{ego}, \text{opp}\}$ **do**
4:  Compute best response: $\text{BR}_{k+1}^i \leftarrow \text{BR}(\boldsymbol{\pi}_k^{-i})$.
5:  Update strategy: $\pi_{k+1}^i \leftarrow \frac{k}{k+1}\, \pi_k^i + \frac{1}{k+1}\, \text{BR}_{k+1}^i$.
6:  **end for**
7: **end for**

**Problem Statement** We consider a two-agent zero-sum POMG, where agents interact in continuous state and action spaces. Each agent $i \in \{\text{ego}, \text{opp}\}$ seeks to maximize its expected return $J_i(\pi^i, \boldsymbol{\pi}^{-i})$ while anticipating worst-case behavior from its opponent. This induces a minimax optimization problem over joint policies.

$$J_i(\pi^i, \boldsymbol{\pi}^{-i}) := \mathbb{E}_{\substack{s_0 \sim b^0,\ \overrightarrow{a_t} \sim \boldsymbol{\pi}(\cdot|\overrightarrow{o}_t), \\ \overrightarrow{s_{t+1}} \sim \mathcal{T}(\cdot|\overrightarrow{s_t}, \overrightarrow{a_t})}} \left[ \sum_{t=0}^{T} \gamma^t\, R_i(\overrightarrow{s_t}, \overrightarrow{a_t}) \right],$$

The goal of each agent is to solve the following minimax optimization:

$$\pi^{*,i} = \arg\max_{\pi^i \in \Pi^i} \min_{\boldsymbol{\pi}^{-i} \in \Pi^{-i}} J_i(\pi^i, \boldsymbol{\pi}^{-i}). \tag{4}$$

## 3 DiffFP: Method

To solve the minimax problem in continuous zero-sum games (Section 2), we adopt a Fictitious Play-style learning framework, as described in Algorithm 1. We model the best response using a Diffusion Policy (DP) to represent stochastic strategies directly in action space. At each iteration, a diffusion-based policy is trained to approximate the best response to a fixed opponent strategy.

### 3.1 Fictitious Play

Fictitious Play [2] is an iterative algorithm in which agents update their strategies based on the empirical distribution of their opponents' past strategies. At each iteration $k$, each agent $i$ computes a best response $\text{BR}_{k+1}^i$ to the current average opponent strategy $\boldsymbol{\pi}_k^{-i}$ and updates its own strategy accordingly, as shown in line 5 of Algorithm 1.

One of the primary challenges in fictitious play is computing the best response policy, especially in environments with continuous action spaces. To address this, we adopt a generalized weakened fictitious play framework [15], which allows for approximate best responses.

### 3.2 Approximating Best Response with RL

Finding an exact best response is often computationally intractable, particularly in large or continuous spaces. Prior work [9] demonstrates that instead of performing full-width backups, one can approximate best responses using reinforcement learning (RL), and estimate the average strategy via supervised learning (SL). Following [29], we model best-response policies using RL and represent the average strategy as a mixture of past best responses. We propose using a diffusion policy to approximate the best response. At each iteration of Fictitious Play (FP), the best-response computation

is formulated against a fixed but unknown average opponent policy, denoted by $\boldsymbol{\pi}_k^{-i}$. This opponent policy is assumed to be stochastic and fixed during the best-response phase, but unknown to the ego agent. Rather than approximating the average policy via supervised learning [9], we implement the averaging rule directly by sampling from past best-response policies, weighted by their corresponding update coefficients. This corresponds to Line 5 in Algorithm 1.

**Diffusion Policy (DP).** Unlike unimodal Gaussian policy networks, DP's do not maintain an explicit likelihood function. To learn expressive, multimodal policies for individual agents, we follow the framework introduced in [28], which combines a behavior cloning loss [11] with double Q-learning [8] to guide policy improvement. The policy improvement step is modeled via action gradients that refine sampled actions stored in the replay buffer.

**Implementation.** In practice, we parameterize both the diffusion actor and critic networks using multilayer perceptrons (MLPs). The actor learns to iteratively denoise actions, while the critic estimates the corresponding target value functions.

# 4 Simulation Studies

We aim to answer the following questions through our simulation experiments:

- **Convergence.** Can DiffFP learn an approximately unexploitable policy profile? Specifically, in continuous and stochastic environments, can it converge to an approximate Nash equilibrium?
- **Efficiency.** What is the benefit of using a DiffFP for continuous state-action games, in terms of stability and sample efficiency?
- **Robustness.** Can our method learn diverse yet robust strategies that generalize to unseen opponents?

**Baselines.** We evaluate DiffFP against several reinforcement learning (RL) algorithms, adapting them as (approximate) best-response within the FP (see Algorithm 1).

▷ **QSM** [21]: A score-based actor model trained by aligning the score function with the state-action gradient field, enabling policy learning via score-matching objective.

▷ **SAC** [6]: maximum entropy RL algorithm that estimates a stochastic policy by maximizing a trade-off between expected return and entropy.

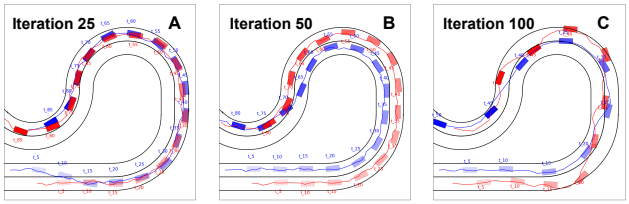


Figure 1: **Training Progression.** Agent trajectories sampled from different FP iterations, illustrating evolving behaviors. Attacking agent (blue) is always initialized behind the defending agent (red). As training progresses, agents learn to navigate more efficiently: note the reduction in completion time from 80 to 45 timesteps, despite identical initial positions.

▷ **TD3** [5]: A deterministic actor-critic method that addresses overestimation bias in Q-learning through clipped double Q-learning and target policy smoothing.

**Metrics.** We evaluate best responses using the total exploitability of the policy (see Equation 3). In practice, this is computed as the difference in episodic rewards between the best-response policy against the average opponent and the average policy against the same opponent. Additionally, for the racing task, we report normalized cumulative rewards to assess overall performance.

## 4.1 Racing Environment

We consider a multi-agent racing scenario [16], where agents must learn strategic behaviors such as overtaking, defending, and navigating efficiently along a track. The environment is modeled as a general-sum game [12, 26], where agents are rewarded based on minimizing or maximizing relative arc-length distances, subject to track boundary constraints and penalties for driving slowly. Additionally, we also evaluate robustness against unseen opponents across different track configurations.

**Results.** Figure 2 shows the exploitability convergence comparison. DiffFP consistently outperforms baselines, achieving faster and more stable convergence. Exploitability measures how much worse a joint policy profile is compared to an approximate Nash equilibrium; a policy is considered less exploitable when it is closer to a Nash strategy.

We also compare against the generative baseline QSM, which exhibits high instability due to sensitivity to value function errors. This leads to noisy exploitability curves, especially in multi-agent settings where non-stationary opponent strategies make value estimation difficult. Moreover, exploitability alone is insufficient, as policies may converge to suboptimal local minima. Figure 3 reports normalized episodic cumulative rewards. DiffFP is the only method that consistently reaches near-optimal returns across opponent initializations. Although SAC and TD3 show apparent exploitability convergence, their actual performance remains poor when measured by cumulative rewards.

We also present agent trajectories from different FP iterations in Figure 1. In panel A, the blue agent (attacker) trails the red agent (defender). As training progresses (panel B), the attacker improves its racing strategy and performs a lane switch, prompting the defender to adapt. Near convergence (panel C), both agents exhibit competitive behaviors: actively attacking and defending resulting in faster lap times while attempting to push each other toward the track limits.

## 4.2 Multiple Particle Environment

To demonstrate the generalizability of our approach and its extension to broader multi-agent setups, we conduct experiments on the Multiple Particle Environment (MPE) suite [17]. This benchmark is particularly challenging due to sparse rewards and the presence of multiple viable strategies.

- **MPE-Adversary:** Ego agents must coordinate around fixed landmarks while simultaneously acting as decoys to mislead adversaries.

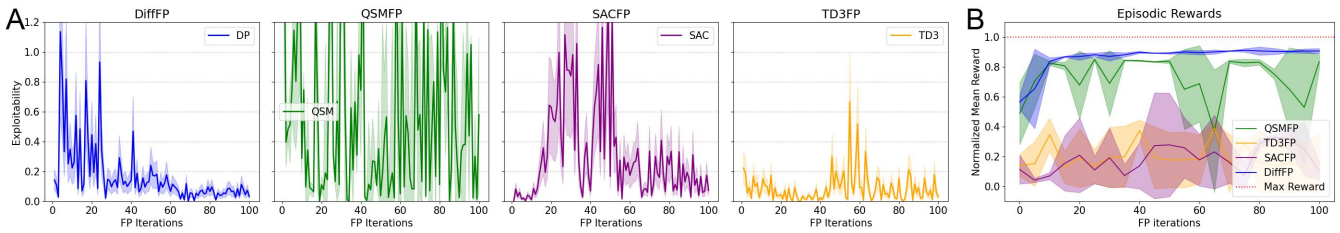


Figure 2: **A. Exploitability convergence on the Racing task.** We report the mean and standard deviation of exploitability as a function of FP iterations, computed over 10 episodes per iteration. **B. Normalized Episodic Cumulative Rewards.** Mean normalized rewards reported over 10 environments per FP iteration. This metric reflects policy performance and stability throughout training.

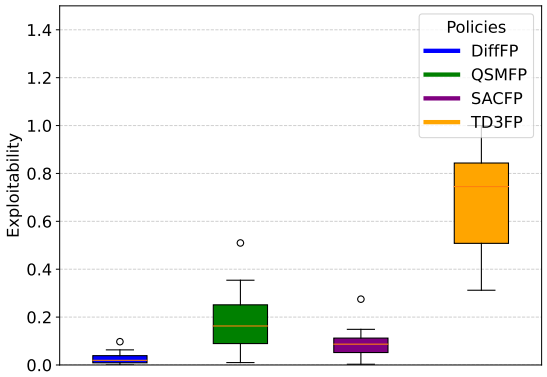


Figure 3: **Exploitability.** We report exploitability (see Eq. 3) computed over 100 evaluation runs. Lower values indicate better robustness, with the proposed DiffFP achieving the lowest exploitability.

| Ego | Adv | Ego Wins | Adv Wins | Draws |
|---|---|---|---|---|
| **Games where DiffFP is Ego** | | | | |
| DiffFP | SACFP | 79 | 10 | 11 |
| DiffFP | TD3FP | 75 | 11 | 14 |
| DiffFP | QSMFP | 75 | 12 | 13 |
| **Games where DiffFP is Adversary** | | | | |
| SACFP | DiffFP | 62 (17↓) | 16 (6↑) | 22 (11↑) |
| TD3FP | DiffFP | 50 (25↓) | 18 (7↑) | 32 (18↑) |
| QSMFP | DiffFP | 63 (12↓) | 27 (15↑) | 10 (3↓) |

Table 1: **Robustness Against Unseen Opponents.** Head-to-head evaluation in the **MPE-Adversary** environment. Colored numbers indicate relative gains (↑) or losses (↓) when ego and adversary roles are swapped.

**Results.** We report exploitability across 100 novel scenarios in Figure 2. DiffFP achieves the lowest exploitability with minimal variance, attributable to its generative modeling framework. Here, policy robustness emerges from exposure to a diverse range of adversaries during training. The combination of generative sampling and action gradient ascent enables effective exploration and synthesis of multimodal solutions.

The sparse reward setting further increases task difficulty. Despite this, DiffFP maintains strong performance due to its stochastic policy representation and gradient-based augmentation, which facilitate better exploration and more accurate estimation of the dynamic reward landscape under limited feedback.

To assess zero-shot generalization, we conduct symmetric matchups between all models, as shown in Table 1. In this task, the ego agent wins if it reaches the true goal without being intercepted; the adversary wins only if it intercepts the goal-directed ego agent (catching a decoy is considered a failure). When playing as the ego agent, DiffFP consistently outperforms all baselines, winning the majority of games. Even when acting as the adversary, it improves baseline win rates and increases the number of draws, demonstrating robust performance across both roles. For example, against SAC, DiffFP achieves a 79% win rate as ego compared to SAC's 62% when roles are reversed a relative gain of ↑ 27%. As adversary, DiffFP improves its win rate from 10% to 16% (↑ 60%) and doubles the draw rate from 11% to 22% (↑ 100%).

## 5 Conclusion

We study the problem of learning policies in dynamic, continuous state-action games. We propose DiffFP, a fictitious-play framework that models the best response using a diffusion policy to capture diverse, multimodal action distributions. Extensive simulation results show that DiffFP achieves faster convergence, more stable training, and learns policies closer to equilibrium, while maintaining robust performance against diverse and unseen opponents.

## References


[1] Anurag Ajay, Yilun Du, Abhi Gupta, Joshua B. Tenenbaum, Tommi S. Jaakkola, and Pulkit Agrawal. Is conditional generative modeling all you need for decision making? In *ICLR*, 2023.

[2] George W. Brown. Iterative solution of games by fictitious play. In T. C. Koopmans, editor, *Activity Analysis of Production and Allocation*. Wiley, New York, 1951.

[3] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Proceedings of Robotics: Science and Systems (RSS)*, 2023.

[4] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[5] Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pages 1587–1596. PMLR, 2018.

[6] Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018.

[7] Eric A. Hansen, Daniel S. Bernstein, and Shlomo Zilberstein. Dynamic programming for partially observable stochastic games. In *National Conference on Artifical Intelligence*, 2004.

[8] Hado Hasselt. Double q-learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 23, 2010.

[9] J Heinrich and D Silver. Deep reinforcement learning from self-play in imperfect-information games, 2016.

[10] Johannes Heinrich, Marc Lanctot, and David Silver. Fictitious self-play in extensive-form games. In *International conference on machine learning*, pages 805–813. PMLR, 2015.

[11] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[12] Junling Hu and Michael P. Wellman. Nash q-learning for general-sum stochastic games. *J. Mach. Learn. Res.*, 4(null):1039–1069, December 2003.

[13] Michael Janner, Yilun Du, Joshua B Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. *arXiv preprint arXiv:2205.09991*, 2022.

[14] Bingyi Kang, Xiao Ma, Chao Du, Tianyu Pang, and Shuicheng Yan. Efficient diffusion policies for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 36, 2024.

[15] David S Leslie and Edmund J Collins. Generalised weakened fictitious play. *Games and Economic Behavior*, 56(2):285–298, 2006.

[16] Edouard Leurent. An environment for autonomous driving decision-making. https://github.com/eleurent/highway-env, 2018.

[17] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *Neural Information Processing Systems (NIPS)*, 2017.

[18] Matej Moravčík, Martin Schmid, Neil Burch, Viliam Lisỳ, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael Bowling. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356(6337):508–513, 2017.

[19] Tim Pearce, Tabish Rashid, Anssi Kanervisto, Dave Bignell, Mingfei Sun, Raluca Georgescu, Sergio Valcarcel Macua, Shan Zheng Tan, Ida Momennejad, Katja Hofmann, and Sam Devlin. Imitating human behaviour with diffusion models, 2023.

[20] Julien Perolat, Bilal Piot, and Olivier Pietquin. Actor-critic fictitious play in simultaneous move multistage games. In Amos Storkey and Fernando Perez-Cruz, editors, *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pages 919–928. PMLR, 09–11 Apr 2018.

[21] Michael Psenka, Alejandro Escontrela, Pieter Abbeel, and Yi Ma. Learning a diffusion model policy from rewards via q-score matching. *arXiv preprint arXiv:2312.11752*, 2023.

[22] Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020.

[23] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

[24] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[25] Oriol Vinyals, Igor Babuschkin, Wojciech Marian Czarnecki, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 2019.

[26] Mingyu Wang, Zijian Wang, John Talbot, J. Christian Gerdes, and Mac Schwager. Game-theoretic planning for self-driving cars in multivehicle competitive scenarios. *IEEE Transactions on Robotics*, 37(4):1313–1325, 2021.

[27] Zhendong Wang, Jonathan J Hunt, and Mingyuan Zhou. Diffusion policies as an expressive policy class for offline reinforcement learning. In *The Eleventh International Conference on Learning Representations*, 2022.

[28] Long Yang, Zhixiong Huang, Fenghao Lei, Yucun Zhong, Yiming Yang, Cong Fang, Shiting Wen, Binbin Zhou, and Zhouchen Lin. Policy representation via diffusion probability model for reinforcement learning. *arXiv preprint arXiv:2305.13122*, 2023.

[29] Shuo Yang, Hongrui Zheng, Cristian-Ioan Vasile, George Pappas, and Rahul Mangharam. Stlgame: Signal temporal logic games in adversarial multi-agent systems. In *7th Annual Learning for Dynamics\& Control Conference*, pages 1102–1114. PMLR, 2025.