# EMAgnet: Parameter-Space EMA Regularization for Policy Gradient Self-Play in Large Games

**Tristan Maidment[1], JB Lanier[1], Chase McDonald, Nathan Tsang[1], Eugene Vinitsky[3], Roy Fox[2], Albert Wang, Wesley N. Kerr[1]**

[1]*Riot Games* [2]*University of California, Irvine* [3]*New York University*

## Abstract

Recent work has established that regularized policy gradient methods such as PPO, when used in self-play, can match or exceed specialized game-theoretic algorithms for solving two-player zero-sum imperfect-information games. The uniform distribution has emerged as a strong policy regularization target for this purpose, but it regularizes equally toward all actions regardless of their viability. We introduce EMAgnet, which instead regularizes toward an exponential moving average (EMA) of the last-iterate policy's parameters, providing an adaptive regularization target that evolves with the agent's improving strategy. We evaluate EMAgnet on both standard two-player zero-sum benchmarks and modified benchmarks with exploration challenges and large numbers of strictly dominated strategies. Relative to PPO self-play with uniform-magnet regularization under both linear and power-law annealing schedules, EMAgnet achieves lower exploitability in the majority of tested environments, with consistent performance gains across games containing strictly dominated strategies.

## 1. Introduction

Solving two-player zero-sum imperfect-information games (IIGs) has driven notable advances in AI, from Poker [3, 4] and Stratego [21, 26] to real-time strategy games like StarCraft [28] and Dota [1]. Of these, multiple results have relied on self-play training stabilized through regularization toward target policies [21, 26], and recent work has demonstrated that with appropriate regularization, generic policy gradient methods can match or exceed other more specialized game-theoretic approaches [23, 25]. Given this growing reliance on regularization, the choice of target policy becomes a key design decision.

Sokota et al. [25] and Rudolph et al. [23] establish the uniform distribution, implemented as an entropy bonus, as a straightforward and effective regularization target. Rudolph et al. [23] show PPO self-play outperforms more specialized game-solving methods such as R-NaD [21], PSRO [9], and NFSP [6]. However, the uniform target is strategically agnostic as it regularizes equally toward all strategies regardless of whether they are viable or strictly dominated. In games with large strategy spaces where most options are bad, the uniform target wastes regularization budget on irrelevant strategies (Figure 1a,b). As games grow in complexity, the fraction of the strategy space that is strategically relevant tends to shrink, making this limitation increasingly consequential. In tabular settings, Sokota et al. [25] explored a continuously moving "magnet" (the regularization target) that trails behind the current policy, demonstrating faster convergence than annealing uniform regularization strength.

We introduce **EMAgnet**, which extends this concept to deep RL by maintaining a parameter-space exponential moving average (EMA) of the policy's own network weights as the regularization

Corresponding authors: tmaidment@riotgames.com, wkerr@riotgames.com

© T. Maidment, J. Lanier, C. McDonald, N. Tsang, E. Vinitsky, R. Fox, A. Wang & W.N. Kerr.

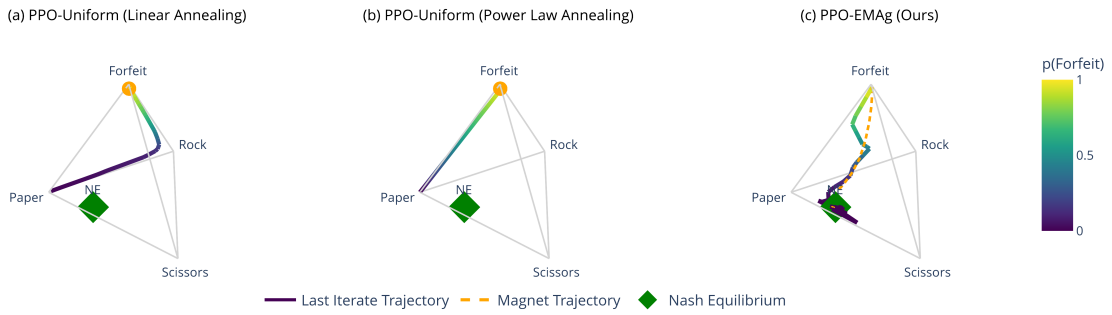


Figure 1: Self-play policy trajectories in Control Biased RPS [11], where agents must solve grid-world navigation tasks to execute each RPS action or else forfeit. **(a,b)** Regularizing toward uniform forces the policy to use strictly dominated strategies that fail navigation and forfeit. By the time annealed regularization is weak enough to avoid forfeiting, the policy fails to explore and find the Nash equilibrium (green diamond). **(c)** PPO-EMAg applies constant regularization toward an EMA of the last iterate (dashed orange), which regularizes toward viable actions the policy has chosen in the past, enabling convergence.

target (Figure 1c). The EMA magnet continuously adapts to the policy while adding minimal complexity to the standard PPO training loop. As the policy learns to avoid dominated strategies, the magnet also gradually stops regularizing toward them. At the same time, the EMA naturally accumulates a smoothed mixture over strategies encountered during self-play cycling, encouraging the policy to maintain coverage over strategically relevant options. This dual property, forgetting bad strategies while remembering good ones, is what distinguishes EMAgnet from a fixed regularization target.

We extend the tabular moving magnet concept from Sokota et al. [25] to deep RL with PPO [24] via parameter-space EMA regularization. We denote this new algorithm as **PPO-EMAg**, and we evaluate it against PPO self-play with uniform-magnet regularization under both linear annealing [23] and power-law annealing [26] across standard two-player zero-sum benchmarks and modified benchmarks containing a large number of strictly dominated strategies [11]. PPO-EMAg matches or outperforms uniform baselines on standard benchmarks and outperforms them in games containing strictly dominated strategies.

## 2. Preliminaries

We consider two-player zero-sum games formalized as finite-horizon partially observable stochastic games (POSGs). A game is defined by the tuple

$$\langle \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{R}, \mathcal{T}, \Omega, T \rangle,$$

where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{O}$ is the observation space, $\mathcal{R}^i : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function for player $i$ with $\mathcal{R}^i = -\mathcal{R}^{-i}$, $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ is the transition function, $\Omega : \mathcal{S} \to \mathcal{O}$ is the observation function, and $T$ is the episode horizon. From a sequence of observations and actions, each player constructs an information state $z \in \mathcal{Z} = \cup_t (\mathcal{O} \times \mathcal{A})^t \times \mathcal{O}$ that is sufficient for optimal decision-making.

Each player $i \in \{1, 2\}$ acts according to a stochastic policy $\pi_i : \mathcal{Z} \to \Delta(\mathcal{A})$. We measure a joint policy $\pi = (\pi_1, \pi_2)$ by its *exploitability*, the mean incentive across players to deviate to a best response. A joint policy is a Nash equilibrium if and only if its exploitability is zero.

## 3. Method

All methods in this work build on Proximal Policy Optimization [24] in symmetric self-play. We represent the policy with a neural network $\pi_\theta$ and use a shared parameterization for both players, with player identity encoded in the observation. Our baseline, **PPO-Uniform** [23], augments the standard clipped PPO objective $\mathcal{L}_{\text{PPO}}(\theta)$ with an entropy bonus $\lambda_H H(\pi_\theta(\cdot \mid z))$ that regularizes the policy toward the uniform distribution, where $\lambda_H > 0$ may be held fixed or annealed over training. We propose replacing this fixed uniform target with an adaptive one.

### 3.1. PPO-EMAg

The tabular moving magnet of Sokota et al. [25] updates the magnet via a geometric average in policy space at each information state, $\rho_{t+1}(h) \propto \rho_t(h)^{1-\tilde{\eta}} \pi_{t+1}(h)^{\tilde{\eta}}$. With neural network policies, maintaining per-information-state policy averages is impractical. **PPO-EMAg** instead performs an arithmetic average in parameter space, replacing the uniform magnet with an exponential moving average (EMA) of the policy parameters. The objective becomes

$$\mathcal{L}_{\text{PPO-EMAg}}(\theta) = \mathcal{L}_{\text{PPO}}(\theta) + \lambda_{\text{KL}}\, \mathbb{E}_{z \sim \mathcal{T}}\big[D_{\text{KL}}\big(\pi_{\theta_{\text{mag}}}(\cdot \mid z) \,\|\, \pi_\theta(\cdot \mid z)\big)\big], \tag{1}$$

where $\theta_{\text{mag}}$ denotes the magnet parameters and $\lambda_{\text{KL}} > 0$ controls the regularization strength. After each PPO update, the magnet parameters are updated as

$$\theta_{\text{mag}} \leftarrow (1 - \tau)\, \theta_{\text{mag}} + \tau\, \theta, \tag{2}$$

with step size $\tau \in (0, 1]$. The magnet is initialized to the same random weights as the policy, $\theta_{\text{mag}} \leftarrow \theta$. The full training procedure is summarized in Algorithm 1 (Appendix).

The key property of PPO-EMAg is that the regularization target adapts with the policy's improving strategy. As the policy learns to avoid strictly dominated strategies, the EMA magnet also gradually stops regularizing toward those strategies. In contrast, the uniform magnet in PPO-Uniform always regularizes toward all actions equally, regardless of their strategic relevance. This distinction becomes significant in games with large strategy spaces containing many suboptimal options, where the uniform magnet wastes regularization budget on irrelevant strategies.

## 4. Experiments

We evaluate PPO-EMAg against two PPO-Uniform baselines across three families of two-player zero-sum games with progressively higher proportions of strictly dominated strategies. Our baselines are PPO-Uniform with linear annealing [23] and PPO-Uniform with power-law annealing [26]. All methods share a fixed compute budget and hyperparameter sweep procedure per environment (Appendix D). For PPO-EMAg, we report exploitability for both the last-iterate policy and the EMA magnet policy, as the magnet often achieves lower exploitability than the last iterate (see Table 1 in the Appendix for full numerical results). We first test on standard games (§4.1), then on games augmented with strictly dominated strategies (§4.2), and finally on games where the vast majority of the strategy space is dominated (§4.3).

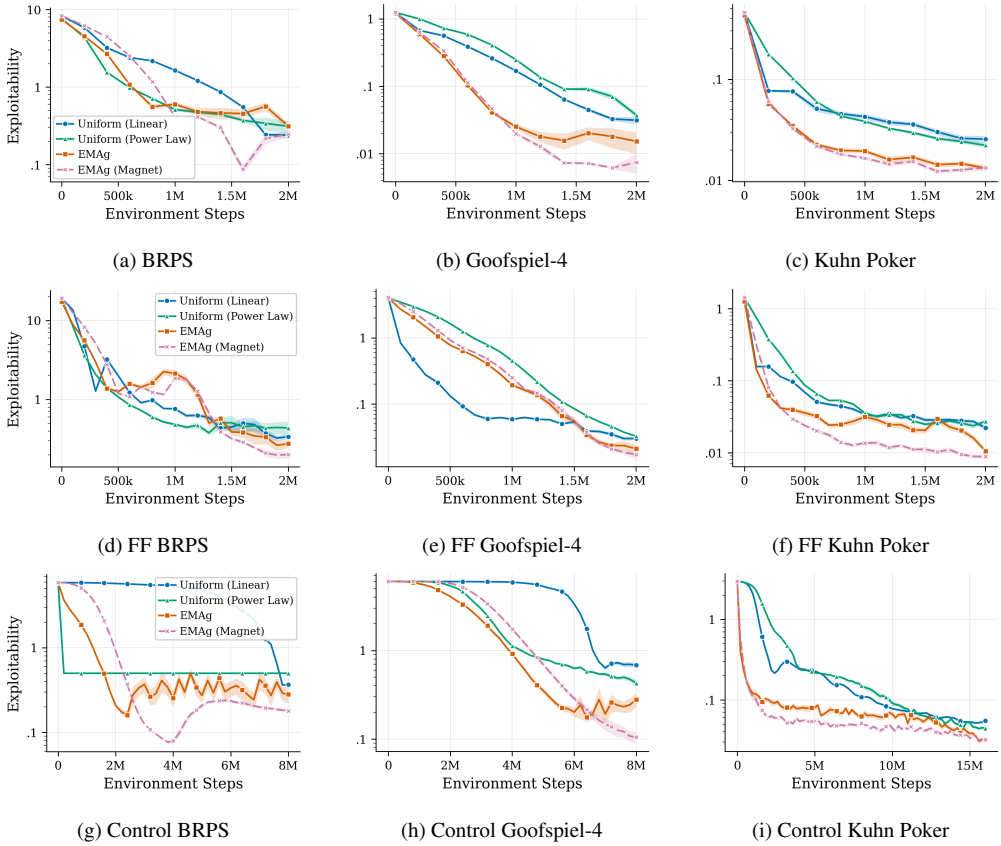


Figure 2: Exploitability over environment steps for each game variant. Top row (a–c): standard games. Middle row (d–f): FF variants with a strictly dominated forfeit action added. Bottom row (g–i): control variants where most strategies are dominated. Best hyperparameter configuration per method (selected via Bayesian sweep), mean across 24 seeds with standard error bands. PPO-EMAg's last-iterate and magnet policies both outperform baselines in FF and control variants, with the magnet consistently reaching lower exploitability than the last iterate. In the control games (g–i), PPO-EMAg also converges significantly faster than both baselines.

### 4.1. Standard Games

We evaluate on three two-player zero-sum games with exact exploitability computation from OpenSpiel [10]: Biased RPS, 4-Card Goofspiel, and Kuhn Poker. Figure 2(a–c) shows exploitability over environment steps. All methods converge to low exploitability across the three games. In Goofspiel-4 and Kuhn Poker, both the PPO-EMAg last iterate and magnet outperform the baselines, with the magnet achieving the lowest exploitability in both games. In Biased RPS, all methods reach comparable final exploitability. PPO-EMAg is competitive with all baselines in the standard game formulations tested.

### 4.2. Forfeit (FF) Games

We apply the forfeit transformation from Lanier et al. [11] to each of the three base games. Every decision node is augmented with a forfeit action. In a game with utilities bounded in $[u_{\min}, u_{\max}]$, the forfeiting player receives $u_{\min} - 1$ and the opponent receives $-(u_{\min} - 1)$, making forfeiting strictly worse than any base-game outcome. The FF variants add strictly dominated strategies to each game while preserving the strategic structure for non-forfeit play.

Figure 2(d–f) shows exploitability curves for the FF variants. The EMA magnet achieves the lowest final exploitability across all three FF games. A key comparison is between Biased RPS and FF Biased RPS. In standard Biased RPS, PPO-EMAg performs comparably to the baselines. In FF Biased RPS, the only change is the addition of a strictly dominated forfeit action, yet PPO-EMAg now outperforms all baselines. In FF Goofspiel and FF Kuhn, both the last iterate and magnet outperform the baselines, with clear separation by the end of training. The uniform magnet continues to regularize equally toward forfeit throughout training, while the EMA magnet adapts to reduce regularization toward it.

### 4.3. Control Games

The control game transformation from Lanier et al. [11] embeds each base-game decision within a multi-step gridworld navigation task. At each decision point, the acting player is placed at a starting position on a grid containing one designated action square per available base-game action. The player navigates using directional movement actions (left, right, up, down, stay) for a fixed number of steps. To execute a base-game action, the player must be on its corresponding action square when the timer expires; if the player is not on any action square, the forfeit action is selected. Each player acts independently during navigation and cannot observe the opponent's grid position.

Unlike the FF variants, where only a single dominated action is added, the control variants produce a strategy space in which the vast majority of strategies are strictly dominated, as most navigation sequences fail to reach any action square. This structure reflects many real-world competitive games where high-level strategic mixing occurs over a small subset of viable options, but executing each option requires a long sequence of coordinated actions.

Because players act independently during navigation, control-game policies are analytically reducible to their corresponding FF game [11]. For each base-game information state, the navigation policy's induced distribution over base-game actions (and forfeit) can be computed by evaluating it over all grid positions and timer values, yielding an equivalent mixed strategy from which exact exploitability is computed. Full details on the control game configurations used in our experiments are provided in Appendix E.

Figure 2(g–i) shows exploitability curves for the control variants. PPO-EMAg outperforms all baselines across all three control environments, with the magnet again achieving the lowest exploitability. The advantage is particularly striking in terms of convergence speed. In Control BRPS (g), Uniform Linear fails to learn for approximately 7M steps before dropping late in training, while Uniform Power Law remains at high exploitability throughout. PPO-EMAg converges by 2M steps. In Control Goofspiel (h), Uniform Linear barely improves over 6M steps while PPO-EMAg reaches low exploitability much earlier. Control Kuhn (i) shows a similar pattern, with PPO-EMAg converging faster than uniform-magnet methods.

**Simplex analysis.** Since control-game policies are reducible to the FF game, we can project the learned policies in Control BRPS onto the probability simplex over {rock, paper, scissors, forfeit}

and visualize their trajectories (Figure 1). All trajectories start near the forfeit vertex (yellow, high p(forfeit)). With uniform annealing (Figure 1a,b), strong early regularization pulls the policy toward all actions including forfeit, causing it to spend training time on strictly dominated strategies. As regularization anneals toward zero, the policy escapes forfeit but now lacks the regularization pressure needed to explore the full strategy space. It settles near a pure strategy without exploring the alternatives that best respond to that strategy. This reveals a fundamental dilemma with annealed uniform regularization: when strong, it wastes budget on dominated strategies, and when weak, it provides insufficient force to encourage mixing.

PPO-EMAg (Figure 1c) avoids this dilemma. The policy gradually reduces its mass on the forfeit action with small, interior cycles among the more viable strategies. Because the EMA magnet (dashed orange) tracks the policy, it stops regularizing toward forfeit once the policy learns to avoid it, while constant (non-annealed) KL regularization continues to encourage mixing throughout training. Once the policy reaches the subspace of non-dominated strategies, it continues cycling near the Nash equilibrium rather than collapsing to a pure strategy.

## 5. Discussion

We introduced EMAgnet, extending the tabular moving magnet concept from Sokota et al. [25] to deep RL with PPO by regularizing toward a parameter-space exponential moving average of the policy's own weights. PPO-EMAg is competitive with uniform-magnet baselines on standard game-solving benchmarks and outperforms them in games containing strictly dominated strategies. Our simplex analysis in Control BRPS (Figure 1) illustrates a limitation of uniform regularization. In games with strictly dominated strategies, a uniform magnet may either waste budget on dominated strategies or lose the regularization force needed for mixing. The EMA magnet avoids this by maintaining its regularization strength while adapting its target.

As games grow in complexity, the fraction of the strategy space that is strategically relevant tends to shrink. In large-scale competitive games, most possible action sequences are suboptimal, and an agent that allocates regularization budget to these options pays an increasing cost. PPO-EMAg offers a simple mechanism for adapting the regularization target to the agent's evolving strategy, requiring only a single EMA update per training step beyond the standard PPO loop.

Understanding when and why PPO-EMAg is most effective likely depends on factors beyond dominated strategy density. In future work, we plan to investigate how structural properties of games, such as the balance of transitive vs. cyclic structure in a game and the number and nature of cycles, affect the relative benefit of adaptive regularization.

## References

[1] Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, et al. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680*, 2019.

[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] Noam Brown and Tuomas Sandholm. Superhuman AI for heads-up no-limit poker: Libratus beats top professionals. *Science*, 359(6374):418–424, 2018.

[4] Noam Brown, Anton Bakhtin, Adam Lerer, and Qucheng Gong. Combining deep reinforcement learning and search for imperfect-information games. In *Advances in Neural Information Processing Systems*, volume 33, pages 17057–17069, 2020.

[5] Yun Kuen Cheung and Georgios Piliouras. Vortices instead of equilibria in minmax optimization: Chaos and butterfly effects of online learning in zero-sum games. In *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 807–834. PMLR, 2019.

[6] Johannes Heinrich and David Silver. Deep reinforcement learning from self-play in imperfect-information games, 2016.

[7] Daniel Hennes, Dustin Morrill, Shayegan Omidshafiei, Rémi Munos, Julien Perolat, Marc Lanctot, Audrunas Gruslys, Jean-Baptiste Lespiau, Paavo Parmas, Edgar Duéñez-Guzmán, et al. Neural replicator dynamics: Multiagent learning via hedging policy gradients. In *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems*, pages 492–501, 2020.

[8] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

[9] Marc Lanctot, Vinicius Zambaldi, Audrunas Gruslys, Angeliki Lazaridou, Karl Tuyls, Julien Perolat, David Silver, and Thore Graepel. A unified game-theoretic approach to multiagent reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[10] Marc Lanctot, Edward Lockhart, Jean-Baptiste Lespiau, Vinicius Zambaldi, Satyaki Upadhyay, Julien Pérolat, Sriram Srinivasan, Finbarr Timbers, Karl Tuyls, Shayegan Omidshafiei, Daniel Hennes, Dustin Morrill, Paul Muller, Timo Ewalds, Ryan Faulkner, János Kramár, Bart De Vylder, Brennan Saeta, James Bradbury, David Ding, Sebastian Borgeaud, Matthew Lai, Julian Schrittwieser, Thomas Anthony, Edward Hughes, Ivo Danihelka, and Jonah Ryan-Davis. OpenSpiel: A framework for reinforcement learning in games. *CoRR*, abs/1908.09453, 2019. URL http://arxiv.org/abs/1908.09453.

[11] JB Lanier, Nathan Monette, Pierre Baldi, and Roy Fox. Data-augmented game starts for accelerating self-play exploration in imperfect information games. *preprint*, 2026.

[12] Hojoon Lee, Hyeonseo Cho, Hyunseung Kim, Donghu Kim, Dugki Min, Jaegul Choo, and Clare Lyle. Slow and steady wins the race: Maintaining plasticity with hare and tortoise networks. *arXiv preprint arXiv:2406.02596*, 2024.

[13] Timothy Paul Lillicrap, Jonathan James Hunt, Alexander Pritzel, Nicolas Manfred Otto Heess, Tom Erez, Yuval Tassa, David Silver, and Daniel Pieter Wierstra. Continuous control with deep reinforcement learning, September 15 2020. US Patent 10,776,692.

[14] Siqi Liu, Luke Marris, Daniel Hennes, Josh Merel, Nicolas Heess, and Thore Graepel. NeuPL: Neural population learning. In *International Conference on Learning Representations*, 2022.

[15] 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.

[16] Stephen McAleer, John Banister Lanier, Roy Fox, and Pierre Baldi. Pipeline PSRO: A scalable approach for finding approximate Nash equilibria in large games. In *Advances in Neural Information Processing Systems*, volume 33, pages 20238–20248, 2020.

[17] Stephen McAleer, John Banister Lanier, Kevin A. Wang, Pierre Baldi, Tuomas Sandholm, and Roy Fox. Toward optimal policy population growth in two-player zero-sum games. In *International Conference on Learning Representations*, 2024.

[18] Stephen Marcus McAleer, John Banister Lanier, Kevin A. Wang, Pierre Baldi, and Roy Fox. XDO: A double oracle algorithm for extensive-form games. In *Advances in Neural Information Processing Systems*, 2021.

[19] Stephen Marcus McAleer, Gabriele Farina, Marc Lanctot, and Tuomas Sandholm. Escher: Eschewing importance sampling in games by computing a history value function to estimate regret. In *The Eleventh International Conference on Learning Representations*, 2023.

[20] Daniel Morales-Brotons, Thijs Vogels, and Hadrien Hendrikx. Exponential moving average of weights in deep learning: Dynamics and benefits. *arXiv preprint arXiv:2411.18704*, 2024.

[21] Julien Pérolat, Bart De Vylder, Daniel Hennes, Eugene Tarassov, Florian Strub, Vincent de Boer, Paul Muller, Jerome T. Connor, Neil Burch, Thomas Anthony, Stephen McAleer, Romuald Elie, Sarah H. Cen, Zhe Wang, Audrunas Gruslys, Aleksandra Malysheva, Mina Khan, Sherjil Ozair, Finbarr Timbers, Tobias Pohlen, Tom Eccles, Mark Rowland, Marc Lanctot, Jean-Baptiste Lespiau, Bilal Piot, Shayegan Omidshafiei, Edward Lockhart, Laurent Sifre, Nathalie Beauguerlange, Rémi Munos, David Silver, Satinder Singh, Demis Hassabis, and Karl Tuyls. Mastering the game of Stratego with model-free multiagent reinforcement learning. *Science*, 378(6623):990–996, 2022.

[22] Alexandre Ramé, Johan Ferret, Nino Vieillard, Robert Dadashi, Léonard Hussenot, Pierre-Louis Cedoz, Pier Giuseppe Sessa, Sertan Girgin, Arthur Douillard, and Olivier Bachem. Warp: On the benefits of weight averaged rewarded policies. *arXiv preprint arXiv:2406.16768*, 2024.

[23] Max Rudolph, Nathan Lichtle, Sobhan Mohammadpour, Alexandre Bayen, J. Zico Kolter, Amy Zhang, Gabriele Farina, Eugene Vinitsky, and Samuel Sokota. Reevaluating policy gradient methods for imperfect-information games. In *International Conference on Learning Representations (ICLR)*, 2026.

[24] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017.

[25] Samuel Sokota, Ryan D'Orazio, J Zico Kolter, Nicolas Loizou, Marc Lanctot, Ioannis Mitliagkas, Noam Brown, and Christian Kroer. A unified approach to reinforcement learning, quantal response equilibria, and two-player zero-sum games. In *The Eleventh International Conference on Learning Representations*, 2023.

[26] Samuel Sokota, Eugene Vinitsky, Hengyuan Hu, J. Zico Kolter, and Gabriele Farina. Superhuman AI for Stratego using self-play reinforcement learning and test-time search. *arXiv preprint arXiv:2511.07312*, 2025.

[27] Eric Steinberger, Adam Lerer, and Noam Brown. DREAM: Deep regret minimization with advantage baselines and model-free learning, 2020.

[28] Oriol Vinyals, Igor Babuschkin, Wojciech M. Czarnecki, et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature*, 575:350–354, 2019.

[29] Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*, pages 23965–23998. PMLR, 2022.

[30] Lunjun Zhang and Jimmy Ba. EMA policy gradient: Taming reinforcement learning for LLMs with EMA anchor and Top-k KL. *arXiv preprint arXiv:2602.04417*, 2026.

## Appendix A. Related Work

### A.1. Two-Player Zero-Sum Game Solving

A central challenge in two-player zero-sum imperfect-information games is that naive self-play with policy gradient methods can cycle or diverge rather than converge to equilibrium [5]. This has motivated a variety of algorithmic frameworks designed to stabilize learning. Population-based methods maintain a growing set of policies and mix over them. Fictitious play [2] and its deep RL successor NFSP [6] average over best responses, while PSRO [9, 16–18] and NeuPL [14] solve an empirical metagame over a policy population. Regret-minimization approaches such as DREAM [27] and ESCHER [19] adapt counterfactual regret minimization to function approximation. A third family of regularized policy-gradient methods, including NeuRD [7], R-NaD [21], and magnetic mirror descent [MMD, 25], stabilizes last-iterate convergence through explicit regularization terms in the policy objective. Our work builds on this last family, proposing a new form of regularization target that adapts over the course of training.

Sokota et al. [25] introduced MMD and showed that policy gradients with strong entropy regularization toward a uniform "magnet" policy converge to quantal-response equilibria, achieving performance competitive with CFR in tabular settings. They also explored a *moving magnet* variant in which the magnet trails behind the current policy rather than remaining fixed, demonstrating faster convergence than annealing the regularization temperature, though only in tabular settings. Rudolph et al. [23] subsequently demonstrated that generic policy-gradient methods such as PPO, when run in a high uniform regularization regime, are competitive with or superior to all FP-, DO-, and CFR-based deep RL approaches across five large imperfect-information games. This result established PPO self-play with uniform-magnet regularization as a strong and simple baseline. More recently, Sokota et al. [26] achieved superhuman performance in Stratego by, among other innovations, annealing regularization coefficients according to power-law schedules over training, which avoids premature entropy collapse while permitting stronger convergence late in training. R-NaD [21] takes a different approach, regularizing via reward shaping toward a periodically updated reference policy. At scale, DeepNash gradually transitions between regularization targets using linear interpolation and uses an EMA of the policy parameters to approximate fixed points. However, the regularization targets themselves remain discrete snapshots set at iteration boundaries, and the algorithm requires a complex multi-phase structure with separate dynamics and update stages.

Existing regularization targets thus range from fixed and strategically uninformed (the uniform distribution) to adaptive but discrete (R-NaD's periodic snapshots). The tabular moving magnet of Sokota et al. [25] demonstrated that a continuously moving target can yield faster convergence than a fixed uniform one. We propose a parameter-space EMA that provides continuous adaptation with minimal additional complexity over the uniform baseline, extending the moving magnet concept to deep RL.

### A.2. Weight Averages in Deep Learning

Weight-space averaging has had significant empirical successes in deep learning and its use has taken many forms. For example, in supervised learning benchmark tasks Izmailov et al. [8] demonstrated that their approach (Stochastic Weight Averaging; SWA) of averaging the weights at specified intervals during stochastic gradient descent resulted in a model that exhibited improved gen-

eralization. In a similar vein, Wortsman et al. [29] showed that an average of diverse fine-tuned models ("model soups") improves both performance and generalization.

EMAs have become an increasingly common approach to weight averaging. Morales-Brotons et al. [20] provided a detailed study of the behavior and training dynamics of EMA models. Their work showed that EMA models often exhibit improved generalization and robustness, as well as calibration, consistency, and performance in transfer learning. Beyond yielding a stronger model on its own, the EMA of the weights has also proven useful as a slowly-moving reference for the training network, improving performance, stability, and plasticity in reinforcement learning for large language models [22, 30] and standard single- and multi-agent reinforcement learning [12, 13, 15].

## Appendix B. PPO-EMAg Training Procedure

EMAgnet requires only a few simple modifications to standard policy gradient methods. We describe the full procedure for PPO-EMAg in Algorithm 1. The magnet policy is initialized to a copy of the policy's parameters $\theta$. Data collection remains unchanged from standard PPO, and the only change to the training objective is the addition of the KL loss term in Equation 1. At the end of each epoch, the parameters of the magnet policy $\theta_{\text{mag}}$ are updated via the EMA update using the current $\theta$.

**Algorithm 1:** PPO-EMAg self-play

**Input:** POSG $\mathcal{G}$, joint policy $\pi_\theta$, magnet parameters $\theta_{\text{mag}} \leftarrow \theta$, EMA step size $\tau$, PPO hyperparameters

```
while not converged do
    Initialize empty trajectory buffer 𝒯;
    for episode = 1, ..., K do
        Roll out self-play episode using π_θ;
        Add trajectory data to 𝒯;
    end
    for PPO epoch = 1, ..., E do
        Update θ using L_PPO-EMAg(θ) on minibatches from 𝒯;
        θ_mag ← (1 − τ) θ_mag + τ θ ;          // EMA magnet update
    end
end
```

## Appendix C. Summary of Results

The final numerical values for exploitability across tasks and algorithms are shown in Table 1. The results for the last-iterate and magnet policy are shown separately, and we compare to uniform regularization baselines.

There are no statistically significant differences for results in Biased RPS. In all remaining environments, at least one of the two PPO-EMAg policies (last-iterate or magnet) achieves the lowest exploitability by a statistically significant margin.

In several cases we observe a significant performance gain by the magnet over the last-iterate policy, although it is not consistent across all settings. In Biased RPS, FF Biased RPS, FF Kuhn

Poker, and Control Goofspiel we observe significantly lower exploitability in the magnet relative to the last iterate. The strong performance of the EMA itself is in line with prior empirical work that has demonstrated strong performance in a number of tasks from EMA policies [20].

Table 1: Exploitability (lower is better) across standard, FF, and control game benchmarks. Reported as mean $\pm$ 95% CI (Student's $t$, $n = 24$ seeds). A † on the magnet cell indicates that it differs significantly ($p < 0.05$) from the last iterate.

| **Game** | **Uniform** (Linear) | **Uniform** (Power Law) | **PPO-EMAg** (Last Iterate) | **PPO-EMAg** (Magnet) |
|---|---|---|---|---|
| *Standard Benchmarks* | | | | |
| Biased RPS | $\mathbf{0.240 \pm 0.043}$ | $0.311 \pm 0.118$ | $0.314 \pm 0.049$ | $0.249 \pm 0.046^{\dagger}$ |
| Goofspiel-4 | $0.031 \pm 0.009$ | $0.037 \pm 0.007$ | $\mathbf{0.015 \pm 0.011}$ | $\mathbf{0.007 \pm 0.005}$ |
| Kuhn Poker | $0.026 \pm 0.004$ | $0.022 \pm 0.003$ | $\mathbf{0.013 \pm 0.002}$ | $\mathbf{0.013 \pm 0.002}$ |
| *FF Games* | | | | |
| FF Biased RPS | $0.338 \pm 0.058$ | $0.436 \pm 0.184$ | $0.275 \pm 0.058$ | $\mathbf{0.201 \pm 0.041}^{\dagger}$ |
| FF Goofspiel-4 | $0.030 \pm 0.003$ | $0.033 \pm 0.002$ | $\mathbf{0.021 \pm 0.006}$ | $\mathbf{0.017 \pm 0.005}$ |
| FF Kuhn Poker | $0.022 \pm 0.003$ | $0.027 \pm 0.004$ | $\mathbf{0.011 \pm 0.001}$ | $\mathbf{0.009 \pm 0.001}^{\dagger}$ |
| *Control Games* | | | | |
| Control BRPS | $0.367 \pm 0.010$ | $0.500 \pm 0.000$ | $0.281 \pm 0.125$ | $\mathbf{0.179 \pm 0.013}$ |
| Control Goofspiel-4 | $0.683 \pm 0.101$ | $0.429 \pm 0.078$ | $\mathbf{0.280 \pm 0.100}$ | $\mathbf{0.105 \pm 0.033}^{\dagger}$ |
| Control Kuhn | $0.055 \pm 0.004$ | $0.044 \pm 0.004$ | $\mathbf{0.032 \pm 0.004}$ | $\mathbf{0.032 \pm 0.004}$ |

## Appendix D. Hyperparameter Sweeps

To ensure a fair comparison across algorithms and environments, we allocate a standardized and fixed compute budget to hyperparameter searches for each method. This means that our claims and results do not preclude further improvements in any of the algorithms tested, but rather suggest that any further gains in performance or reductions in exploitability require additional resources.

We use Bayesian optimization for hyperparameter selection, choosing configurations that minimize final exploitability. We sweep over a specified set of parameters for each algorithm, allocating 700 samples from the parameter space with three random seeds per parameter set. All runs are trained to completion (i.e., no early stopping). The parameters tuned for each algorithm are shown in Table 2.

For power-law schedules, we use the following formulation to scale the entropy and learning rates:

$$(1 + \rho C)^{-q}, \tag{3}$$

where $C$ is a scaling coefficient, $q$ is the power-law exponent, and $\rho$ is the current training progress in terms of the fraction of total updates completed.

Table 2: Parameter specifications for each algorithm used across environments.

| **Parameter** | Values | **Sampling Method** |
|---|---|---|
| *Uniform (Linear)* | | |
| Entropy Coefficient | [1e-4, 32.0] | Log Uniform |
| Anneal Entropy | {true, false} | Choice |
| Learning Rate | [1e-7, 0.01] | Log Uniform |
| Anneal Learning Rate | {true, false} | Choice |
| Clip Coefficient | [1e-4, 0.4] | Log Uniform |
| GAE $\lambda$ | [0.6, 1.0] | Uniform |
| VF Coefficient | [0.1, 5.0] | Uniform |
| *Uniform (Power Law)* | | |
| Learning Rate Power Law $C$ | [1, 1e5] | Log Uniform |
| Learning Rate Power Law $q$ | [0.1, 2.0] | Uniform |
| Entropy Power Law $C$ | [1, 1e5] | Log Uniform |
| Entropy Power Law $q$ | [0.1, 2.0] | Uniform |
| Entropy Coefficient | [1e-4, 32.0] | Log Uniform |
| Learning Rate | [1e-7, 0.01] | Log Uniform |
| Clip Coefficient | [1e-4, 0.4] | Log Uniform |
| GAE $\lambda$ | [0.6, 1.0] | Uniform |
| VF Coefficient | [0.1, 5.0] | Uniform |
| *PPO-EMAg* | | |
| EMAg $\lambda_{\text{KL}}$ | [0.01, 32.0] | Log Uniform |
| EMAg $\tau$ | [1e-5, 0.1] | Log Uniform |
| Entropy Coefficient | [1e-4, 0.1] | Log Uniform |
| Anneal Entropy | {true, false} | Choice |
| Learning Rate | [1e-7, 0.01] | Log Uniform |
| Anneal Learning Rate | {true, false} | Choice |
| Clip Coefficient | [1e-4, 0.4] | Log Uniform |
| GAE $\lambda$ | [0.6, 1.0] | Uniform |
| VF Coefficient | [0.1, 5.0] | Uniform |

## Appendix E. Control Game Details

We apply the forfeit and control game transformations from Lanier et al. [11] to Biased RPS, 4-Card Goofspiel, and Kuhn Poker. In each control variant, the base game is first augmented with a forfeit action (the FF transformation), then each decision node is replaced with a gridworld navigation task (the control transformation).

**Observation space.** At each navigation step, the acting player observes a vector consisting of their normalized grid position (2 dimensions), the fraction of time remaining (1 dimension), and the base-game information state tensor. The opponent's grid position is not observed, which is what enables the analytic reduction to the FF game.

**Action space.** During navigation, the player selects from five movement actions: left, right, up, down, and stay. The time limit equals the number of steps available to reach an action square. When

the timer expires, the action square the player occupies determines the base-game action taken; if the player is not on any action square, the forfeit action is taken in the base game.

**Exploitability computation.** Because players act independently during navigation, the control policy's induced distribution over base-game actions can be computed exactly. For each base-game information state, we evaluate the navigation policy over all grid positions and timer values to compute the probability of terminating on each action square or forfeiting. These probabilities define an equivalent mixed strategy over base-game actions plus forfeit, from which we compute exact exploitability in the base (FF) game. The cost of this reduction is proportional to the number of information states, grid cells, and timer steps, and is orders of magnitude smaller than the cost of self-play training.

**Configurations used.** Table 3 lists the specific grid configurations used for each control game in our experiments.

Table 3: Control game configurations. Each action square corresponds to one base-game action. Forfeit is the default when no action square is reached.

| **Game** | **Grid Size** | **Timer (steps)** | **Dist. to Action Sq.** | **Action Squares** |
|---|---|---|---|---|
| Control BRPS | $5 \times 5$ | 4 | 4 | 3 (rock, paper, scissors) |
| Control Goofspiel-4 | $7 \times 7$ | 5 | 5 | 4 (one per card) |
| Control Kuhn | $5 \times 5$ | 4 | 4 | 2 (bet, pass) |

## Camera-Ready Edits Summary

Relative to the original submission, this camera-ready version (i) adds author names and affiliations, (ii) makes minor wording clarifications, and (iii) applies minor grammatical and formatting corrections. The clarifications most notably concern the definition of exploitability, the description of the control-game mechanics, and the related work discussion, where one citation was also updated. The methods, experiments, and results are unchanged.