TIME EFFICIENT SWAP REGRET MINIMIZATION

ASHLEY YU

Abstract. No-regret learning algorithms provide principled frameworks for multi-agent decision-making, with swap regret minimization enabling convergence to correlated equilibria, a stronger solution concept than the coarse correlated equilibria achieved by external regret algorithms. The classical Blum-Mansour (BM) algorithm achieves optimal $O(\sqrt{NT \log N})$ swap regret bounds, but computing the stationary distribution of an $N \times N$ Markov chain at each iteration requires $O(N^3)$ time complexity that severely limits scalability.

We propose a novel approach that replaces exact stationary distribution computation with efficient sampling-based estimation, reducing per-iteration complexity from $O(N^3)$ to $O(N)$ while maintaining the fundamental structure of the original algorithm.

Date: November 2025.

1. Introduction

No-regret learning formalizes repeated decision-making in adversarial or multi-agent settings, where agents iteratively choose actions and observe outcomes. Over time, they adjust their strategies based on feedback, aiming to match or outperform the best fixed action in hindsight. The central performance measure is *regret*. The difference between the learner's cumulative loss and that of the best comparator. An algorithm is said to be *no-regret* if this quantity grows sublinearly with the number of rounds Cesa-Bianchi and Lugosi (2006).

One of the most fundamental notions is **external regret** (ExtReg), which compares the learner's performance to the best fixed action in hindsight:

$$\mathsf{ExtReg}(T) = \sum_{t=1}^{T} \left\langle \mathbf{x}^{(t)}, \ell^{(t)} \right\rangle - \min_{\mathbf{x}^* \in \Delta_N} \sum_{t=1}^{T} \left\langle \mathbf{x}^*, \ell^{(t)} \right\rangle .$$

Algorithms like Multiplicative Weights Update (MWU) achieve $\mathsf{ExtReg}(T) = O(\sqrt{T \log N})$ with $O(N)$ computation per round Cesa-Bianchi and Lugosi (2006). This efficiency enables applications ranging from minimax strategies in two-player zero-sum games to learning *coarse correlated equilibria* (CCE) in general games Nash (1951); Syrgkanis et al. (2015).

Correlated equilibrium (CE) and *coarse correlated equilibrium* (CCE) are solution concepts for multi-agent interactions Aumann (1974); Hart and Mas-Colell (2000). Consider n players with joint action space $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$ and loss functions $\ell_i : \mathcal{A} \to [0, 1]$. Let D be a distribution over joint actions. Then:

- D is a **coarse correlated equilibrium** (CCE) if for all players i and actions $a_i' \in \mathcal{A}_i$:

$$\mathbb{E}_{\mathbf{a} \sim D}[\ell_i(\mathbf{a})] \leq \mathbb{E}_{\mathbf{a} \sim D}[\ell_i(a_i', \mathbf{a}_{-i})]$$

- D is a **correlated equilibrium** (CE) if for all players i, actions a_i in the support of D_i (the marginal on $\mathcal{A}_i$), and deviations $a_i' \in \mathcal{A}_i$:

$$\mathbb{E}_{\mathbf{a} \sim D}[\ell_i(\mathbf{a}) \mid a_i] \leq \mathbb{E}_{\mathbf{a} \sim D}[\ell_i(a_i', \mathbf{a}_{-i}) \mid a_i]$$

The key distinction is that CE conditions on the recommended action a_i, requiring obedience even with knowledge of one's own action. CCE only requires obedience to unconditional deviations.

However, CCEs often provide weak guarantees in multi-agent settings. Stronger solution concepts like **correlated equilibrium** (CE) require minimizing **swap regret** (SwapReg), which benchmarks against the best *action mapping* $\phi : [N] \to [N]$ in hindsight Blum and Mansour (2007):

$$\mathsf{SwapReg}(T) = \sum_{t=1}^{T} \left\langle \mathbf{x}^{(t)}, \ell^{(t)} \right\rangle - \min_{\phi \in \Phi} \sum_{t=1}^{T} \left\langle \phi \circ \mathbf{x}^{(t)}, \ell^{(t)} \right\rangle .$$

While Blum–Mansour's (BM) classical approach minimizes SwapReg using N parallel instances of an external regret minimizer, it suffers from a critical computational bottleneck: each iteration requires solving for the stationary distribution $\mathbf{x}^{(t)}$ of a Markov chain defined by transition matrix $\mathbf{Q}^{(t)}$, incurring $O(N^\omega)$ time, where ω refers to the optimal time complexity of matrix multiplication Blum and Mansour (2007); Peng and Rubinstein (2023).

1.1. **Monte Carlo-Based Blum-Mansour.** We propose MC-BM, a novel algorithm that overcomes the computational bottleneck by replacing exact stationary distribution computation with an efficient *Monte Carlo (MC) sampling* procedure. The key innovation is recognizing that we don't need exact stationary distributions, only approximate samples that are "close enough" to preserve swap-regret guarantees. Instead of solving for the stationary distribution analytically at cost $O(N^3)$ per iteration, we approximate it by running the Markov chain defined by $\mathbf{Q}^{(t)}$ for a small number of steps k (where $k \ll N$) and collecting m samples. Each sample is obtained by starting from a random initial state and transitioning according to $\mathbf{Q}^{(t)}$ for k steps. The empirical distribution of these samples serves as our approximation to the stationary distribution.

This MC-based approach dramatically reduces computational complexity. While exact methods require $O(N^3)$ per iteration, our sampling procedure requires only $O(k \cdot m \cdot N)$ per round. For constant choices of k and m, this achieves $O(N)$ per-iteration cost, matching the efficiency of external regret minimization while targeting the stronger swap regret guarantee.

In this paper, we first formalize the trade-offs between ExtReg, SwapReg, and computational efficiency (Section 2). There, we will detail CM-BM's design and analyze the relationship between mixing time, sampling parameters, and regret accumulation. After that, we present experimental results (Section 3). These results provide compelling evidence that CM-BM achieves sublinear swap regret with dramatically improved computational efficiency, motivating new theoretical approaches for analyzing MCMC-based approximations in online learning.

2. Preliminaries

2.1. **Online Learning Setup.** Consider a repeated decision process over T rounds. At each round t, the learner chooses distribution $\mathbf{x}^{(t)} \in \Delta_N$ over N actions, the adversary reveals loss vector $\ell^{(t)} \in [0,1]^N$, after which the learner suffers loss $\left\langle \mathbf{x}^{(t)}, \ell^{(t)} \right\rangle$. The learner aims to minimize regret relative to comparator classes.

2.2. **Regret Definitions.**

Definition 1 (External Regret (ExtReg))**.**

$$\mathsf{ExtReg}(T) = \sum_{t=1}^{T} \left\langle \mathbf{x}^{(t)}, \ell^{(t)} \right\rangle - \min_{i \in [N]} \sum_{t=1}^{T} \ell_i^{(t)}$$

Definition 2 (Swap Regret ($\mathsf{SwapReg}$))**.**

$$\mathsf{SwapReg}(T) = \sum_{t=1}^{T} \left\langle \mathbf{x}^{(t)}, \ell^{(t)} \right\rangle - \min_{\phi:[N]\to[N]} \sum_{t=1}^{T} \ell^{(t)}_{\phi(i)} x^{(t)}_i$$

where ϕ ranges over all action modifications.

2.3. **Follow-the-Regularized-Leader (FTRL).** FTRL balances loss minimization with regularization:

$$\mathbf{x}^{(t)} = \arg\min_{\mathbf{x}\in\Delta_N} \left(\sum_{\tau=1}^{t-1} \left\langle \mathbf{x}, \ell^{(\tau)} \right\rangle + \frac{1}{\eta} R(\mathbf{x}) \right)$$

Theorem 1 (FTRL External Regret Bound)**.** *For $R(\mathbf{x}) = \sum_{i=1}^{N} x_i \log x_i$ and $\eta = \sqrt{2\log N/T}$,*

$$\mathsf{ExtReg}(T) \le \sqrt{2T \log N}$$

Proof. By the FTRL lemma, for any $\mathbf{x}^* \in \Delta_N$:

$$\sum_{t=1}^{T} \left\langle \mathbf{x}^{(t)} - \mathbf{x}^*, \ell^{(t)} \right\rangle \le \frac{R(\mathbf{x}^*) - R(\mathbf{x}^{(1)})}{\eta} + \frac{\eta}{2} \sum_{t=1}^{T} \left\| \ell^{(t)} \right\|_\infty^2$$

With $R_{\max} - R_{\min} \le \log N$ and $\left\|\ell^{(t)}\right\|_\infty^2 \le 1$:

$$\mathsf{ExtReg}(T) \le \frac{\log N}{\eta} + \frac{\eta T}{2} = \sqrt{2T \log N} \quad \square$$

$\square$

2.4. **Blum-Mansour (BM) Algorithm.** BM reduces swap regret minimization to external regret minimization:

Algorithm 1 Blum-Mansour

Require: External regret algorithm $\mathcal{A}$, actions $[N]$
1: Initialize N copies $\mathcal{A}_1, \ldots, \mathcal{A}_N$ of $\mathcal{A}$
2: **for** $t = 1$ **to** T **do**
3: **for** $i = 1$ **to** N **do**
4: Feed $\mathcal{A}_i$ loss $x^{(t)}[i] \cdot \ell^{(t)}$
5: Receive $q_i^{(t)} \in \Delta_N$ from $\mathcal{A}_i$
6: **end for**
7: Construct $\mathbf{Q}^{(t)} = [q_1^{(t)} \| \cdots \| q_N^{(t)}]$
8: Compute stationary $\mathbf{x}^{(t)}$ satisfying $\mathbf{x}^{(t)} = \mathbf{Q}^{(t)}\mathbf{x}^{(t)}$
9: Play $\mathbf{x}^{(t)}$, observe $\ell^{(t)}$
10: **end for**

Theorem 2 (BM Swap Regret Bound)**.** *For $\mathcal{A}$ achieving $\mathsf{ExtReg}_i(T) \le \sqrt{C_i \log N}$ with $C_i = \sum_t x^{(t)}[i]$,*

$$\mathsf{SwapReg}(T) \le \sqrt{NT \log N}$$

Proof. Let ϕ^* be optimal modifier. The regret decomposes as:

$$\mathsf{SwapReg}(T) = \sum_{i=1}^{N} \underbrace{\left(\sum_{t=1}^{T} x^{(t)}[i] \ell_i^{(t)} - \min_j \sum_{t=1}^{T} x^{(t)}[i] \ell_j^{(t)} \right)}_{\text{Regret of } \mathcal{A}_i}$$

Each $\mathcal{A}_i$ has regret $\le \sqrt{(\sum_t x^{(t)}[i]) \log N} = \sqrt{C_i \log N}$. By Cauchy-Schwarz:

$$\mathsf{SwapReg}(T) \le \sqrt{\log N} \sum_{i=1}^{N} \sqrt{C_i} \le \sqrt{N \log N} \sqrt{\sum_{i=1}^{N} C_i} = \sqrt{NT \log N} \quad \square$$

$\square$

2.5. **The Computational Bottleneck.** The main computational challenge in Blum-Mansour is computing the stationary distribution $\mathbf{x}^{(t)}$ satisfying $\mathbf{x}^{(t)} = \mathbf{Q}^{(t)}\mathbf{x}^{(t)}$ at each round. Standard methods include linear system solving, which finds the eigenvector for eigenvalue 1 via Gaussian elimination at cost $O(N^3)$, power iteration, which computes $\lim_{k\to\infty}(\mathbf{Q}^{(t)})^k\mathbf{x}$ requiring many iterations each costing $O(N^2)$, and matrix inversion using $(I - \mathbf{Q}^{(t)} + \mathbf{1}\mathbf{1}^\top)^{-1}\mathbf{1}$ at cost $O(N^\omega)$ where $\omega \approx 2.37$. Over T rounds, this stationary computation dominates the overall complexity, making BM impractical for large action spaces despite its strong theoretical guarantees.

3. Monte Carlo Blum Mansour

The classical Blum-Mansour (BM) algorithm guarantees sublinear swap regret but requires computing a stationary distribution at each iteration, costing $O(N^3)$ per round. In this section, we introduce *MC-Based Blum-Mansour* (MC-BM), which replaces exact stationary distribution computation with Markov chain sampling. We approximate the stationary distribution $\mathbf{x}^{(t)}$ of the transition matrix $\mathbf{Q}^{(t)}$ by running the Markov chain for k steps and collecting m samples. This reduces complexity from $O(N^3)$ to $O(k \cdot m \cdot N)$ per round.

Algorithm 2 MC-BM

Input: External regret algorithm $\mathcal{A}$, actions $[N]$, MC steps k, number of samples m
Initialize: N copies $\mathcal{A}_1, \ldots, \mathcal{A}_N$ of $\mathcal{A}$
for $t = 1$ to T **do**
    for $i = 1$ to N **do**
        Feed $\mathcal{A}_i$ loss $x^{(t)}[i] \cdot \ell^{(t)}$
        Receive distribution $q_i^{(t)} \in \Delta_N$ from $\mathcal{A}_i$
    end for
    Construct $\mathbf{Q}^{(t)} = [q_1^{(t)} \| \cdots \| q_N^{(t)}]$
    // MC sampling to approximate stationary distribution
    for $j = 1$ to m **do**
        Sample $a_0^{(j)} \sim \text{Uniform}([N])$
        for $s = 1$ to k **do**
            Sample $a_s^{(j)} \sim \mathbf{Q}^{(t)}[a_{s-1}^{(j)}, \cdot]$
        end for
        Set $\hat{a}^{(j)} \leftarrow a_k^{(j)}$
    end for
    Set $\mathbf{x}^{(t)} \leftarrow$ empirical distribution of $\{\hat{a}^{(1)}, \ldots, \hat{a}^{(m)}\}$
    Play $\mathbf{x}^{(t)}$, observe $\ell^{(t)}$
end for

3.1. **Algorithm Description.** Instead of solving $\mathbf{x}^{(t)} = \mathbf{Q}^{(t)}\mathbf{x}^{(t)}$ exactly, we approximate the stationary distribution by sampling. For each of m independent trials, we start from a uniformly random state $a_0^{(j)}$ and run the Markov chain defined by $\mathbf{Q}^{(t)}$ for k steps, obtaining a final state $\hat{a}^{(j)}$. The empirical distribution of these m samples gives our approximation $\mathbf{x}^{(t)}$. Running a Markov chain for k steps produces samples approximately distributed according to the stationary distribution, with accuracy depending on the chain's mixing time.

3.2. **Rationale and Advantages.** MC-BM replaces exact computation with sampling to overcome the computational bottleneck in BM. Computing the stationary distribution in exact BM requires solving a linear system via Gaussian elimination at cost $O(N^3)$, power iteration requiring many $O(N^2)$ matrix-vector products, or matrix inversion at cost $O(N^\omega)$. Each sample in MC-BM requires only k Markov chain transitions, each costing $O(N)$ with alias method preprocessing for categorical sampling.

We do not need exact stationary distributions, only approximations that preserve the regret decomposition structure underlying BM's theoretical guarantees. As the Markov chain runs, the distribution of samples converges to the true stationary distribution. The mixing time of $\mathbf{Q}^{(t)}$ determines how many steps k are required, while the number of samples m controls the variance of the empirical distribution.

Exact BM becomes impractical for $N > 50$ due to cubic scaling. MC-BM maintains efficiency for hundreds or thousands of actions, which is important for applications in large-scale

strategic games, online advertising, or resource allocation where action spaces are naturally large.

3.3. **Computational Complexity and Extensions.** Each Markov chain step requires sampling from a categorical distribution, which costs $O(N)$ per step using alias method preprocessing (one-time $O(N)$ cost per distribution $q_i^{(t)}$). Running m chains for k steps requires $O(m \cdot k \cdot N)$ per round. For constant k and m, this achieves $O(N)$ per-iteration cost, matching the efficiency of external regret algorithms while targeting swap regret. MC-BM has time complexity $O(T \cdot N)$ compared to exact BM's $O(T \cdot N^3)$.

Several extensions are possible. Adaptive sampling can increase k and m when detecting slow convergence or high variance in samples. Mixing time estimation can monitor the spectral gap or use heuristics to automatically set k based on the mixing properties of $\mathbf{Q}^{(t)}$. Variance reduction methods like antithetic variates or control variates can reduce the variance of the empirical distribution for fixed m. Hybrid approaches can combine MC-BM with optimistic updates or other refinements to improve convergence rates.

3.4. **Theoretical Considerations.**

Conjecture 1 (MC-BM Swap Regret)**.** *For MC steps $k = \Omega(\log T)$ and samples $m = \Omega(\log T)$, MC-BM achieves*

$$\mathbb{E}[\mathsf{SwapReg}(T)] = O(\sqrt{NT \log N} + T \cdot \mathit{poly}(N) \cdot e^{-\gamma k})$$

where γ is the minimum spectral gap of $\{\mathbf{Q}^{(t)}\}_{t=1}^{T}$.

The main theoretical challenge is bounding the error from approximating the stationary distribution. In exact BM, the regret decomposes cleanly across the N external regret minimizers because $\mathbf{x}^{(t)}$ is exactly stationary: $\mathbf{x}^{(t)} = \mathbf{Q}^{(t)}\mathbf{x}^{(t)}$. When we approximate $\mathbf{x}^{(t)}$ via sampling, this stationarity property holds only approximately, introducing an error term that accumulates over T rounds.

The approximation error depends on two factors: the mixing time of the Markov chain, which determines how quickly the distribution of samples converges to stationarity (controlled by k and the spectral gap γ), and the sampling variance, which measures how well the empirical distribution of m samples approximates the true distribution (scaling as $O(1/\sqrt{m})$). The question is whether these errors remain sublinear when summed over T rounds, preserving the $O(\sqrt{T})$ swap regret bound.

A significant challenge is that $\mathbf{Q}^{(t)}$ changes at each round (it depends on the history through the evolving regret minimizers $\mathcal{A}_i$). This prevents direct application of standard Markov chain analysis, which assumes a fixed transition matrix. New techniques for analyzing time-varying Markov chains and their cumulative approximation error are needed to establish MC-BM's theoretical guarantees.

3.5. **Summary.** MC-BM modifies the Blum-Mansour algorithm to replace exact stationary distribution computation with Markov chain sampling, reducing computational cost from $O(N^3)$ to $O(N)$ per iteration. The sampling-based approximation preserves the structure of BM while making swap regret minimization practical for large action spaces. Section **??** demonstrates that these design choices lead to competitive swap regret performance in practice.

4. Results

In this section, we present our empirical findings comparing MC-BM to exact Blum-Mansour (BM). Experiments were performed on Kuhn Poker, random normal-form games with varying action space sizes, and subgames of the Diplomacy environment.

For random games, we generated two-player normal-form games with payoff matrices sampled uniformly from $[0, 1]$. We tested action spaces of size $N = 128$, $N = 256$, and $N = 512$ to evaluate scalability. We also tested Diplomacy, which is an extensive form game with normal-form subgames. There are 7 players and each chooses their action simultaneously. We used the neural network from Gray et al. (2020), which finds the 11-14 best actions for each power. We modified it to obtain utility matrices. We obtained over 500 two-player subgames for 6 pairs of powers. The remaining 4 powers' actions were randomly selected from the filtered actions. We also obtained several 7-player subgames. For MC-BM, we used $k = 1000$ MC steps with $m = 1$ sample unless otherwise specified.

Unless otherwise specified, each trial was repeated multiple times with different random seeds, and we plot the average experimental over the number of iterations T.

4.1. **Kuhn Poker.** In figure 1, we measure swap regret over T iterations for both exact BM and MC-BM. Both algorithms achieve sublinear swap regret and converge to correlated equilibrium. MC-BM exhibits slightly higher swap regret than exact BM, but the difference remains small relative to the overall regret scale. This demonstrates that while the MC-based approximation introduces some error, it remains comparable to exact BM and preserves the key convergence properties.

4.2. **Random Games.** We test performance on randomly generated normal-form games with larger action spaces. Figures 2, 3, and 4 show results for games with $N = 128$, $N = 256$, and $N = 512$ actions per player. In all experiments, MC-BM converges to correlated equilibrium with swap regret slightly higher than exact BM. The gap remains small. These results show that MC-BM provides a good trade-off between speed and solution quality.

4.3. **Diplomacy Subgames.** We next evaluate performance on selected two-player subgames extracted from the complex, multi-player game Diplomacy.

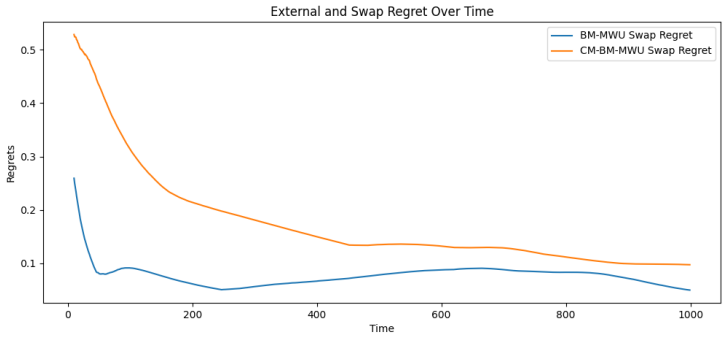

FIGURE 1. Kuhn Poker: BM vs. MC-BM. MC-BM achieves comparable swap regret to exact BM while maintaining $O(N)$ per-iteration complexity instead of $O(N^3)$.

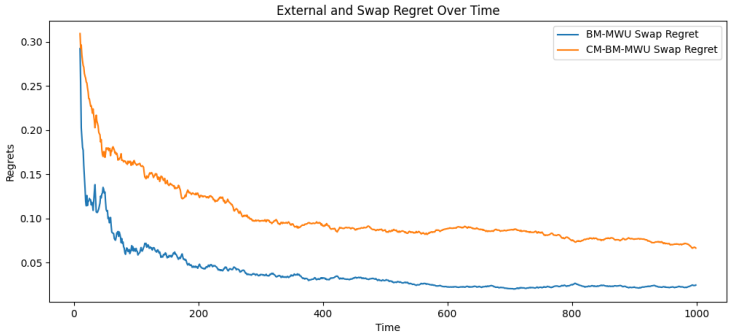

FIGURE 2. Random game ($N = 128$): BM vs. MC-BM. MC-BM maintains comparable swap regret while achieving significant computational savings over exact BM.

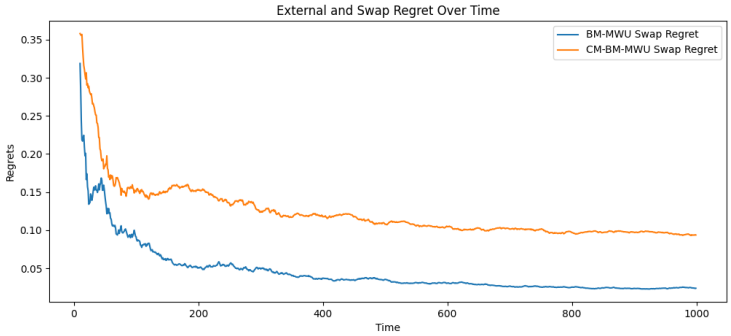

FIGURE 3. Random game ($N = 256$): BM vs. MC-BM. MC-BM continues to perform comparably to exact BM while the computational advantage becomes more pronounced.

Figures 5, 6, and 7 show results comparing exact BM and MC-BM. The patterns here closely mirror those observed in Kuhn Poker. Swap regret for MC-BM consistently exceeds that of exact BM, with the gap stabilizing after initial convergence, but it still remains minimal.

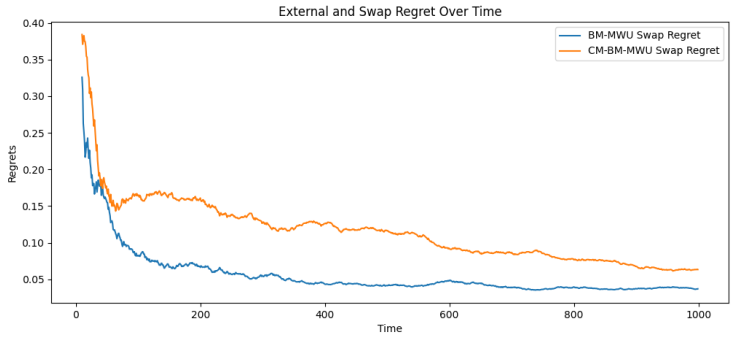

FIGURE 4. Random game ($N = 512$): BM vs. MC-BM. MC-BM scales to large action spaces where exact BM becomes computationally prohibitive, while maintaining comparable swap regret guarantees.

Two key observations distinguish these strategic games from random games. First, the swap regret curves exhibit more oscillatory behavior in early iterations, particularly visible in the first 200 iterations. This likely reflects the structured nature of Diplomacy payoff matrices compared to random matrices. Second, the relative performance gap between MC-BM and exact BM appears similar to that observed in Kuhn Poker and random games, confirming that the MC-based approximation error generalizes across different game types.

The final convergence values vary across matchups. This reflects different strategic complexities. Some subgames converge to lower regret values (around 0.013-0.015 in the first two figures). Others maintain slightly higher equilibrium regret (around 0.018 in the third figure). In all cases, MC-BM tracks exact BM's convergence behavior. This suggests that the approximation preserves the essential learning dynamics. Across different Diplomacy settings, MC-BM consistently showed bounded swap regret. It maintained the computational efficiency advantage that becomes critical for scaling to larger games.

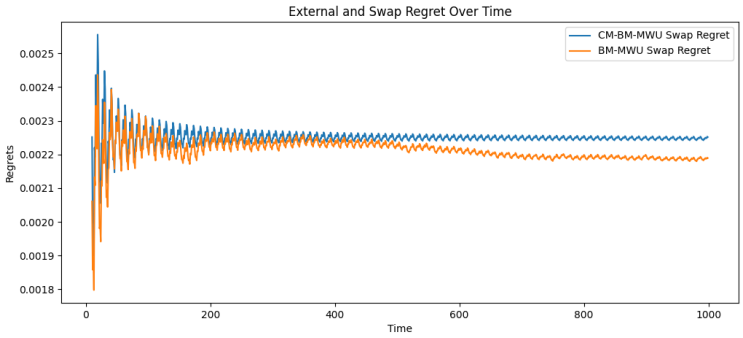

FIGURE 5. Diplomacy subgame: BM vs. MC-BM. MC-BM demonstrates comparable convergence, showing that the MC-based approximation works effectively in strategic scenarios.

4.4. **Summary.** Our experiments reveal two key findings. First, MC-BM delivers swap regret performance on par with exact BM across multiple domains, including Kuhn Poker,

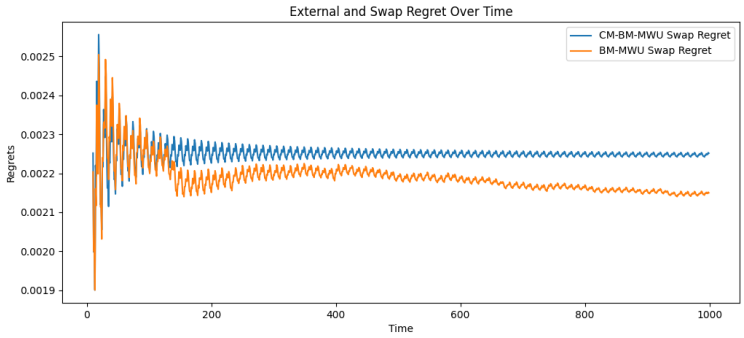

FIGURE 6. Diplomacy subgame: BM vs. MC-BM. MC-BM again achieves comparable swap regret, reinforcing the effectiveness of sampling-based approximation in practice.

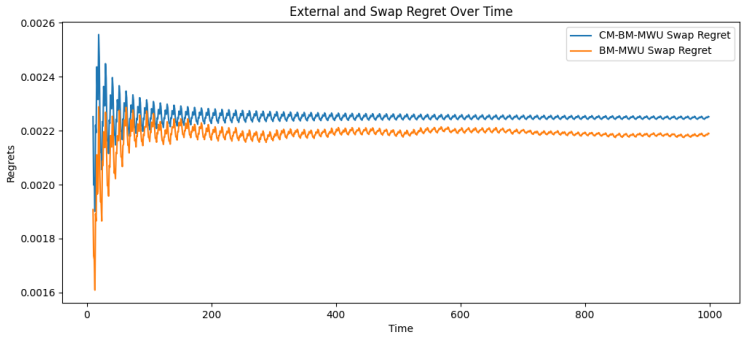

FIGURE 7. Diplomacy subgame: BM vs. MC-BM. Sampling-based approximation in MC-BM leads to comparable convergence across different strategic matchups.

random games, and Diplomacy subgames. Though Monte Carlo approximation leads to marginally higher swap regret, the difference remains small and practically acceptable. Second, MC-BM's computational benefits scale dramatically as action spaces expand. With 512 actions, MC-BM runs over 1000 times faster than exact BM, making the minor performance trade-off highly worthwhile.

5. CONCLUSION

We have presented a new swap regret minimization algorithm, MC-Based Blum-Mansour (MC-BM), designed to overcome the computational bottleneck in exact Blum-Mansour. By replacing exact stationary distribution computation with Markov chain sampling, MC-BM achieves dramatic computational speedups over exact BM while maintaining comparable swap regret performance. Moreover, our results indicate that the MC-based approximation introduces only a small, bounded performance gap in practice, with external regret remaining nearly identical across all experiments.

Extensive experiments on Kuhn Poker, random games, and Diplomacy subgames confirm that MC-BM achieves swap regret comparable to exact BM, thereby enabling practical correlated equilibrium computation in settings where exact methods are prohibitively expensive. The computational efficiency improvement is particularly strong for large action spaces. These findings show the potential for MC-BM to be applied in broader multi-agent AI settings, as scalable swap regret minimization is vital for achieving stronger equilibrium guarantees in these settings.

Looking ahead, it remains an open theoretical question to establish rigorous regret bounds for MC-BM that account for the MC-based approximation error. Future work should analyze how mixing time, spectral gap, and sampling parameters (k and m) affect the accumulated approximation error over T rounds. Additional research directions include adaptive sampling schemes that automatically adjust k based on observed mixing properties, variance reduction techniques to improve sample efficiency, and integration with optimistic or predictive updates to potentially accelerate convergence. We believe that continued research along these lines will help bridge the gap between the strong theoretical guarantees of swap regret minimization and the computational efficiency required for real-world multi-agent systems.

References

Aumann, R. J. (1974). Subjectivity and correlation in randomized strategies. *Journal of Mathematical Economics*, 1(1):67–96.

Blum, A. and Mansour, Y. (2007). From external to internal regret. *Journal of Machine Learning Research*, 8(6):1307–1324.

Cesa-Bianchi, N. and Lugosi, G. (2006). *Prediction, Learning, and Games*. Cambridge University Press.

Gray, J., Lerer, A., Bakhtin, A., and Brown, N. (2020). Human-level performance in no-press diplomacy via equilibrium search. *arXiv preprint arXiv:2010.02923*.

Hart, S. and Mas-Colell, A. (2000). A simple adaptive procedure leading to correlated equilibrium. *Econometrica*, 68(5):1127–1150.

Nash, J. (1951). Non-cooperative games. *Annals of Mathematics*, 54(2):286–295.

Peng, B. and Rubinstein, A. (2023). Fast swap regret minimization and applications to approximate correlated equilibria.

Syrgkanis, V., Agarwal, A., Luo, H., and Schapire, R. E. (2015). Fast convergence of regularized learning in games. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 28, pages 2989–2997.