# Bounding Regret in Simulated Games

**Steven Jecmen** [1] **Erik Brinkman** [1] **Arunesh Sinha** [1]

## Abstract

We present a bandit-style problem arising from a specific problem in agent-based modeling of games. In this preliminary work, we provide some initial heuristic algorithms and compare against some baselines.

## 1. Introduction

Empirical game-theoretic analysis has been widely employed for analyzing large-scale complex games (Wellman, 2006). These large, complex games include stock markets and cyber-security games. In such games, the payoffs of an action profile (mixed or pure) are themselves random variables and the expected value has to be estimated via a simulation of the underlying game dynamics. Then, using a heuristic search in the space of action profiles, a candidate Nash equilibrium action profile is found. The regret of an action profile is the maximum payoff gain any player can achieve by switching from playing the profile to playing some other strategy. Finding the regret of an action profile is an important step in the empirical analysis of a game; for instance, a candidate profile can be confirmed as a Nash equilibrium if it is found to have zero regret.

For games where all payoffs are known deterministically, finding the regret is simple. However, for simulated games, not all payoffs are known and it can be expensive even to sample all pure-action profiles in support to find the payoff of a single mixed-action profile. This difficulty is still present when the game has a nice structure, like being symmetric; that is, all players have same action spaces $S$ and the payoff depends on how many players played an action (and not who played that action). Empirical game-theoretic analysis is often performed for symmetric games (Wellman, 2006).

We seek to provide an algorithm that outputs a tight confidence interval containing the true regret as quickly as possible. We additionally aim to provide a theoretical upper bound on regret for large symmetric games that is known to be within some fixed distance of the true regret. This goal, at first, may appear to be similar to the "combinatorial pure exploration of bandits" problem (Chen et al., 2014). However, there are subtle differences, which we elaborate upon in related work.

In this paper, we present a preliminary algorithm that finds an upper bound of fixed width on the regret of a mixed-action profile for a simulated game, minimizing the number of samples taken from the simulator. We compare this performance to the performance of several different methods of distributing samples. Our future goal is to perform a theoretical analysis of the guarantees of our algorithm or any improved version of the same.

## 2. Simulation-Based Game

In a simulation-based game, a simulator acts as an oracle for player utility functions, taking as input a strategy profile, an assignment of one strategy to every player, and returning an observation of the payoff each player accrued from that profile in simulation. Since these games typically incorporate stochastic factors (uncertainty in environment and agent private information), the payoff-vector observation is actually a sample from some underlying distribution of payoff outcomes. It is known that every finite symmetric game has a symmetric equilibrium (Nash, 1951), where a symmetric equilibrium is one in which every player plays the same (mixed) action. Hence, solving such games typically involves searching for a symmetric equilibrium. Various techniques of finding an approximate symmetric Nash equilibrium for such games have been proposed in literature.

We analyze a symmetric $n$ player game with finite action space $S$ (the same for every player). We restrict ourselves to symmetric mixed actions $\sigma$, where $\sigma_i$ denotes the probability of choosing $i \in S$. The quality of an approximate equilibrium is measured by regret, the largest gain any player can achieve through unilateral deviation. We wish to compute regret for a given fixed $\sigma$, which will be implicit in all our notation henceforth. Denote by $P_i$ the random utility of a player when this player plays $i \in S$ and others play according to $\sigma$ (a "deviating payoff"). Formally, regret of a

[1]Computer Science and Engineering, University of Michigan, Ann Arbor, Michigan, United States. Correspondence to: Steven Jecmen <sjecmen@umich.edu>.

*Published at the Exploration in Reinforcement Learning Workshop at the $35^{th}$ International Conference on Machine Learning*, Stockholm, Sweden. 

symmetric action profile $\sigma$ in a symmetric game is given by:

$$R = \max_{i \in S} E[G_i] \text{ where } G_i = P_i - \sum_{j \in S} \sigma_j P_j$$

$G_i$ is the gain of deviating to $i$. We view $\nu_i = E[G_i]$ as the expected value of the $i^{th}$ super-arm that is formed by a linear combination of underlying arms whose expected values are given by $\mu_i = E[P_i]$. At any time step we obtain a sample from one arm (not super-arm).

Our sampling primitive is only the deviating payoff $P_i$ from a mixed-action profile. While other information can be gained from most simulators (such as deviating payoffs from the underlying pure-action profiles or non-deviating mixed-action profile payoffs), it is unclear how well this information could be used in constructing bounds on the relevant values. We believe that our primitive is one reasonable way of approaching the problem and is simpler to work with than the alternatives.

**Problem Statement** : Our goal is to find an interval $[L_R, U_R]$ such that $R \in [L_R, U_R]$ and $U_R - L_R < z$ for some small fixed $z$ within the fewest samples possible.

## 3. Related Work

Our problem is a bandit-style problem, yet different from any setting studied till now. First, we are not concerned about costs over rounds; thus, our goal is not classic stochastic bandit regret minimization. More closely related to our problem is the problem of best arm identification (Jamieson & Nowak, 2014), but our problem differs as we aim to identify the best super-arm. The problem of combinatorial pure exploration in bandits (Chen et al., 2014; Gabillon et al., 2016) also has super-arms that are subsets of arms, and the reward of each super-arm is the sum of its subset of arms. Again, our problem differs as super-arm rewards are linear combinations of rewards of each arm. More importantly, the approach to combinatorial pure exploration works by choosing arms that lie in the symmetric difference of two subsets of arms, whereas for our case this symmetric difference of two subsets of arms maybe always empty (e.g. when the support size of the given mixed action is all the actions). Further, our goal is to not identify the best super-arm, but provide a desired confidence interval for the best super-arm expected reward in the shortest time possible.

## 4. Preliminary Analysis and Motivation for Algorithm

Recall Hoeffding's inequality, where $\overline{X}$ is the mean of $n$ i.i.d. random variables with mean $\mu$ and subgaussian parameter $g^2$, $P[\overline{X} - \mu \geq t] \leq e^{-\frac{t^2 n}{2g^2}}$.

Thus, if $P_i$ is subgaussian with parameter $g^2$ and mean $\mu_i$,

$$P\left[\overline{P}_{i,t} + \sqrt{\frac{2g^2}{N_{i,t}} \ln \frac{1}{\delta}} \leq \mu_i\right] \leq \delta$$

$$P\left[\overline{P}_{i,t} - \sqrt{\frac{2g^2}{N_{i,t}} \ln \frac{1}{\delta}} \geq \mu_i\right] \leq \delta$$

where $\overline{P}_{i,t}$ is the sample mean and $N_{i,t}$ is the number of samples taken from action $i$ up to (and including) time step $t$. It is also easy to check that $\overline{P}_{i,t} - \mu_i$ is subgaussian with parameter $g^2/N_{i,t}$ and mean 0.

Also of interest are bounds on the gains $G_{i,t}$ from deviating, the difference in payoff to a player who deviates from playing $\boldsymbol{\sigma}$ to playing one of their pure strategies $i$. Denote the support of the given mixed action as $U$. We construct bounds on these as well, using the following set of equations.

$$\nu_i = \mu_i - \sum_{j \in U} \sigma_j \mu_j$$

$$\nu_i = (1 - \sigma_i)\mu_i - \sum_{j \in U \setminus i} \sigma_j \mu_j$$

$$\overline{G}_{i,t} = (1 - \sigma_i)\overline{P}_{i,t} - \sum_{j \in U \setminus i} \sigma_j \overline{P}_{j,t}$$

$$\overline{G}_{i,t} - \nu_i = (1 - \sigma_i)(\overline{P}_{i,t} - \mu_i) - \sum_{j \in U \setminus i} \sigma_j (\overline{P}_{j,t} - \mu_j)$$

From properties of addition of subgaussians, the last line above yields that $\overline{G}_{i,t} - \nu_i$ is subgaussian with parameter $g^2 \sum_{k \in S \cup \{i\}} c_{k,i}^2 / N_k$, where $c_{k,i} = \sigma_k$ if $k \in S \setminus i$ else $c_{k,i} = 1 - \sigma_k$. Thus, we get

$$P\left[\overline{G}_{i,t} + \sqrt{2g^2 \ln \frac{1}{\delta} \sum_{k \in U \cup \{i\}} \frac{c_{k,i}^2}{N_{k,t}}} \leq \nu_i\right] \leq \delta$$

$$P\left[\overline{G}_{i,t} - \sqrt{2g^2 \ln \frac{1}{\delta} \sum_{k \in U \cup \{i\}} \frac{c_{k,i}^2}{N_{k,t}}} \geq \nu_i\right] \leq \delta$$

**Ideal Sampling from Best Deviation Oracle**: Consider the scenario when we know the best deviation $i^*$. In such a scenario, we would want to minimize the quantity $\sqrt{2g^2 \ln \frac{1}{\delta} \sum_{k \in U \cup \{i^*\}} \frac{c_{k,i^*}^2}{N_{k,t}}}$ for a given number of time steps $t$ where $t = \sum_{k \in S \cup \{i^*\}} N_{k,t}$. Using the generalized arithmetic-harmonic mean inequality we get that $\sum_{k \in U \cup \{i^*\}} \frac{c_{k,i^*}^2}{N_{k,t}} \geq 1/t$ with equality only when $\frac{c_{k,i^*}}{N_{k,t}} = \lambda$ for some constant $\lambda$. While this inequality is for real numbers, given large enough $N_{k,t}$'s we can approximately satisfy this equality condition, thereby minimizing the bound.

**Algorithm 1** SAUCB: fixed time setting

Sample from all strategies once to initialize $N_{i,0}$, $\overline{P}_{i,0}$ for all $i \in S$
$B_{i,0} \leftarrow \sqrt{2g^2 \ln \frac{1}{\delta} \sum_{k \in U \cup \{i\}} \frac{c_{k,i}^2}{N_{k,0}}}$ for all $i \in S$
$t \leftarrow 1$
**while** $t \leq T$ **do**
  $i_t^* \leftarrow \arg\max_{i \in S} \overline{G}_{i,t-1} + B_{i,t-1}$
  $l \leftarrow \arg\max_{j \in S} \frac{c_{j,i_t^*}}{N_{j,t-1}}$
  Sample strategy $l$ and update $N_{l,t}$, $\overline{P}_{l,t}$
  $B_{i,t} \leftarrow \sqrt{2g^2 \ln \frac{1}{\delta} \sum_{k \in U \cup \{i\}} \frac{c_{k,i}^2}{N_{k,t}}}$ for all $i \in S$
  $t \leftarrow t + 1$
**end while**

**Algorithm 2** SAUCB: early stop setting

Initialize $\mathcal{K}$, the K time steps to check whether to stop
Initialize in the same manner as fixed time setting
$t \leftarrow 1$
**loop**
  Sample in the same manner as fixed time setting
  **if** $t \in \mathcal{K}$ and $\max_{i \in S} \overline{G}_{i,t} + B_{i,t} - \max_{j \in S} \overline{G}_{j,t} - B_{j,t} \leq z$ **then**
    return $\left[\max_{j \in S} \overline{G}_{j,t} - B_{j,t}, \max_{i \in S} \overline{G}_{i,t} + B_{i,t}\right]$
  **end if**
  $t \leftarrow t + 1$
**end loop**

Thus, this suggests that the $t$ samples should be distributed among the actions in $U \cup \{i^*\}$ as

$$N_{k,t} = \frac{c_{k,i^*}}{\sum_{k \in U \cup \{i^*\}} c_{k,i^*}} t$$

In our algorithm, this proportional allocation forms the basis of choosing which action to sample from a given super-arm.

**Naive Sampling**: A naive sampling approach is to choose each action equally often in a round robin fashion. Thus, every action is sampled approximately $t/|S|$ times, which gives the bounds for each deviation gain (super-arm). Then, the bound for the best deviation gain at time $t$ provides the confidence interval we obtain after $t$ time steps.

**Modified UCB**: We also consider as a baseline a modified version of the UCB algorithm for best-arm identification. For the first half of the iterations, this algorithm samples from the action with the highest upper bound on its deviating payoff, as in UCB. The algorithm then selects one action as the best deviation and for the remaining samples, picks according to the optimal distribution described above for that deviation. In the "early stop" setting (described below), this process is repeated during each period between checks.

## 5. Algorithm

We consider two settings for the problem. In one, which we call the "fixed time" setting, sampling continues for a fixed total number $T$ of iterations. In the other, which we call the "early stop" setting, sampling occurs only until a fixed maximum number of samples is taken, the bound width is checked a fixed number of times throughout in an attempt to stop sampling as early as possible, and Bonferroni correction (a known correction for multiple tests) is used to avoid the problem of repeated testing. Since our objective is to minimize the number of samples taken to achieve a bound of fixed width, the early stop setting is more practically applicable. However, the fixed time setting is simpler to analyze and performance should translate between settings.

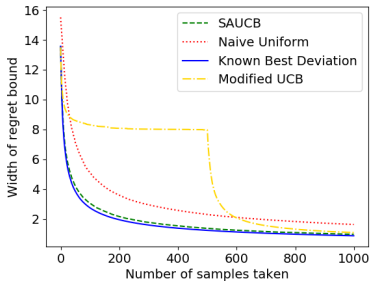
Our algorithm (called SAUCB, due to the concept of super-arms and its similarities to UCB) samples deviating payoffs and maintains an upper bound on regret with confidence level $1 - \alpha$ (where $\alpha$ is a free parameter), sampling in order to minimize this bound. We do this by maintaining bounds on the gains from deviating to each strategy, and at each iteration greedily sampling the deviating payoff to the strategy that reduces the current upper bound on regret by the most. Following the analysis in Section 4, we therefore sample from the strategy which maximizes $\frac{c_{j,i_t^*}}{N_{j,t-1}}$. We present here our algorithms for both settings.

In the fixed time setting, input parameter $T$ specifies the total number of iterations the algorithm will run for. In the early stop setting, input parameters $z$ and $K$ specify the desired bound width and number of checks of this stopping criterion respectively. In addition, we require that sampling stop after a finite number of iterations, since the number of checks of the stopping criterion is fixed. This total number of iterations $T$ in the early stop setting is calculated as the number of samples that one would have to take such that there must be at least one deviation such that the bound width is below $z$; this means that any algorithm will return a bound by iteration $T$ at the latest, and so we space the $K$ checks of the stopping criteria uniformly on $[1, T]$.

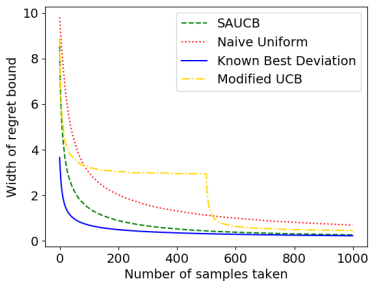
$\delta$, the probability of error in any bound on $\nu_i$ for any action $i \in S$, is calculated using the union bound as $\delta = \frac{\alpha}{2*|\mathcal{S}|}$ in the fixed time setting, so as to provide a $1-\alpha$ guarantee that the confidence interval for every deviation is correct. In the early stop setting, we use Bonferroni correction and choose $\delta = \frac{\alpha}{2*|\mathcal{S}|*K}$. This ensures that the regret bounds returned in the early stop setting are correct with probability $1 - \alpha$.

| Experiment | Means $\boldsymbol{\mu}$ of each deviation | Mixed strategy $\sigma$ |
|---|---|---|
| 1 | [10, 9.9, 4.5, 4, 3.5, 3, 2.5, 2, 1.5, 1] | [0, 0, 0, 0, 0.25, 0, 0, 0, 0.5, 0.25] |
| 2 | [10, 9.5, 4.5, 4, 3.5, 3, 2.5, 2, 1.5, 1] | [0.75, 0, 0, 0, 0.25, 0, 0, 0, 0, 0] |
| 3 | [10, 9, 8, 7, 6, 5, 4, 3, 2, 1] | [0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1] |

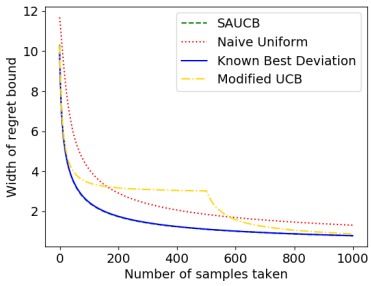
*Table 1.* Individual experiment parameters



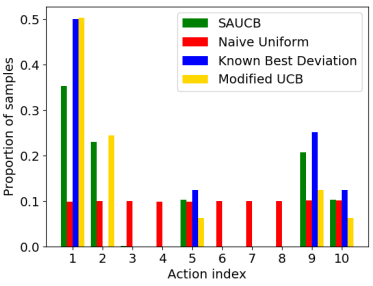

(a) Exp. 1: Bound width



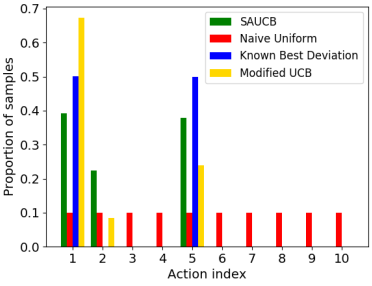

(b) Exp. 2: Bound width



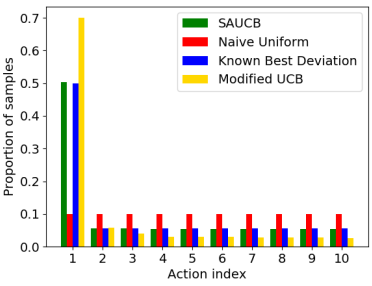

(c) Exp. 3: Bound width




(d) Exp. 1: Sample distribution




(e) Exp. 2: Sample distribution




(f) Exp. 3: Sample distribution

| Approach | Exp. 1 | Exp. 2 | Exp. 3 |
|---|---|---|---|
| Naive Uniform | 15348.29 | 2822.59 | 10614.0 |
| Known Best | 4924.36 | 2122.0 | 4245.0 |
| Modified UCB | 6898.75 | 2122.0 | 6368.0 |
| SAUCB | 6410.46 | 2122.0 | 4245.0 |

*Table 2.* Samples taken to achieve desired bound

## 6. Experiments

In this section, we present the results of the aforementioned algorithms on some synthetic experiments. We assume a game with 10 actions. In lieu of simulated games, deviating payoffs are sampled from independent Gaussian distributions with identical standard deviation $std = 2$ in all experiments. Parameters $\alpha = 0.05$, $T = 1000$ (in the fixed time setting), $z = 0.5$ (in the early stop setting), and $K = 10$ (in the early stop setting) are used in all experiments. Results are averaged across 1000 trials in the fixed time setting and 100 trials in the early stop setting. We perform three experiments; the deviating payoff means $\boldsymbol{\mu}$ and mixed-action profile $\sigma$ for each are specified in Table 1. Experiment 1 is the hard case where the top two deviation means are close.

Results from the fixed time setting are shown in figures (a) through (f) and results from the early stop setting are shown in Table 2. In general, SAUCB performs close to the ideal "known best deviation" baseline and superior to uniform or modified UCB, both in terms of convergence (a-c) or proportion of samples allocated to each action (d-f). The closeness to "known best deviation" is also seen in Table 2, with the difference occurring in the hard case of Exp. 1.

## References

Chen, Shouyuan, Lin, Tian, King, Irwin, Lyu, Michael R, and Chen, Wei. Combinatorial pure exploration of multi-armed bandits. In *NIPS*, pp. 379–387. 2014.

Gabillon, Victor, Lazaric, Alessandro, Ghavamzadeh, Mohammad, Ortner, Ronald, and Bartlett, Peter. Improved learning complexity in combinatorial pure exploration

bandits. In *Artificial Intelligence and Statistics*, 2016.

Jamieson, Kevin and Nowak, Robert. Best-arm identification algorithms for multi-armed bandits in the fixed confidence setting. In *Information Sciences and Systems (CISS)*. IEEE, 2014.

Nash, John. Non-cooperative games. *Annals of mathematics*, 1951.

Wellman, Michael P. Methods for empirical game-theoretic analysis. In *21st national conference on Artificial intelligence-Volume 2*, 2006.